GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

Advance Communication Engineering

For 6th Semester

ELECTRONICS AND TELECOMMUNICATION

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Mr. Sarada Prasanna Singh

(Lecturer in Electronics & Telecommunication Engineering)

SATTELITE SYSTEM

CHAPTER 1

1.1 Introduction:-

Satellites offer a number of features not readily available with other means of communications. Because very large areas of the earth are visible from a satellite, the satellite can form the star point of a communications net, simultaneously linking many users who may be widely separated geographically. The same feature enables satellites to provide communications links to remote communities in sparsely populated areas that are difficult to access by other means. Of course, satellite signals ignore political boundaries as well as geographic ones, which may or may not be a desirable feature.

Satellites are also used for remote sensing, examples being the detection of water pollution and the monitoring and reporting of2 Chapter One weather conditions. Some of these remote sensing satellites also form a vital link in search and rescue operations for downed aircraft and the like. Satellites are specifically made for telecommunication purpose. They are used for mobile applications such as communication to ships, vehicles, planes, hand-held terminals and for TV and radio broadcasting. They are responsible for providing these services to an assigned region (area) on the earth. The power and bandwidth of these satellites depend upon the preferred size of the footprint, complexity of the traffic control protocol schemes and the cost of ground stations. A satellite works most efficiently when the transmissions are focused with a desired area. When the area is focused, then the emissions do not go outside that designated area and thus minimizing the interference to the other systems. This leads more efficient spectrum usage.

Satellite’s antenna patterns play an important role and must be designed to best cover the designated geographical area (which is generally irregular in shape). Satellites should be designed by keeping in mind its usability for short and long term effects throughout its life time. The earth station should be in a position to control the satellite if it drifts from its orbit it is subjected to any kind of drag from the external forces.

1.2 History of Satellite Communications

The first artificial satellite used solely to further advances in global communications was a balloon named Echo 1. Echo 1 was the world's first artificial communications satellite capable of relaying signals to other points on Earth. The first American satellite to relay communications was Project SCORE in 1958, which used a tape recorder to store and forward voice messages. It was used to send a Christmas greeting to the world from U.S. President Dwight D. Eisenhower. NASA launched the Echo satellite in 1960; the 100-foot (30 m) aluminised PET film balloon served as a passive reflector for radio communications. Courier 1B, built by Philco, also launched in 1960, was the world's first active repeater satellite. The first communications satellite was Sputnik 1. Put into orbit by the Soviet Union on October 4, 1957, it was equipped with an onboard radio-transmitter that worked on two frequencies: 20.005 and 40.002 MHz. Sputnik 1 was launched as a step in the exploration of space and rocket development. While incredibly important it was not placed in orbit for the purpose of sending data from one point on earth to another. And it was the first artificial satellite in the steps leading to today's satellite communications. Telstar was the second active, direct relay communications satellite. Belonging to AT&T as part of a multi-national agreement between AT&T, Bell Telephone Laboratories, NASA, the British General Post Office, and the French National PTT (Post Office) to develop satellite communications, it was launched by NASA from Cape Canaveral on July 10, 1962, the first privately sponsored space launch. Relay 1 was launched on December 13, 1962, and became the first satellite to broadcast across the Pacific on November 22, 1963.

CHAPTER 2: ORBITAL MECHANICS

Satellites (spacecraft) orbiting the earth follow the same laws that govern the motion of the planets around the sun. From early times much has been learned about planetary motion through careful observations. Johannes Kepler (1571–1630) was able to derive empirically three laws describing planetary motion. Later, in 1665, Sir Isaac Newton (1642–1727) derived Kepler's laws from his own laws of mechanics and developed the theory of gravitation.

2.1 Kepler's Laws of Planetary Motion

Kepler's First Law:- *Kepler's first law* states that the path followed by a satellite around the primary will be an ellipse. An ellipse has two focal points shown as *F1* and *F2* in Fig.1.

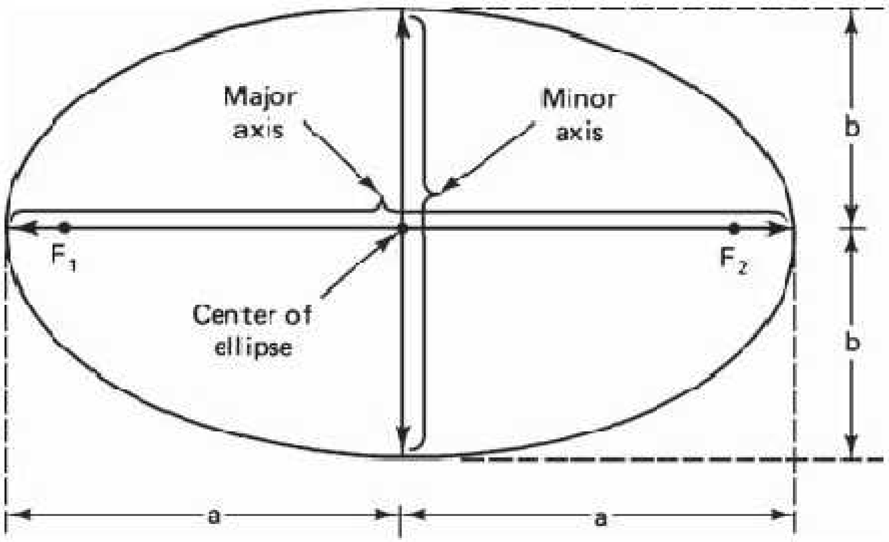


Fig.2.1 The foci *F1* and *F2*, the semimajor axis *a*, and the semiminor axis *b* of an ellips

The foci *F* The eccentricity and the semimajor axis are two of the orbital parameters specified for satellites (spacecraft) orbiting the earth. For an elliptical orbit, 0<*e*< 1. When *e* = 0, the orbit becomes circular.

Kepler's Second Law:- *Kepler's second law* states that, for equal time intervals, a satellite will sweep out equal areas in its orbital plane, focused at the barycenter. The center of mass of the two-body system, termed the *barycenter*, is always centered on one of the foci.

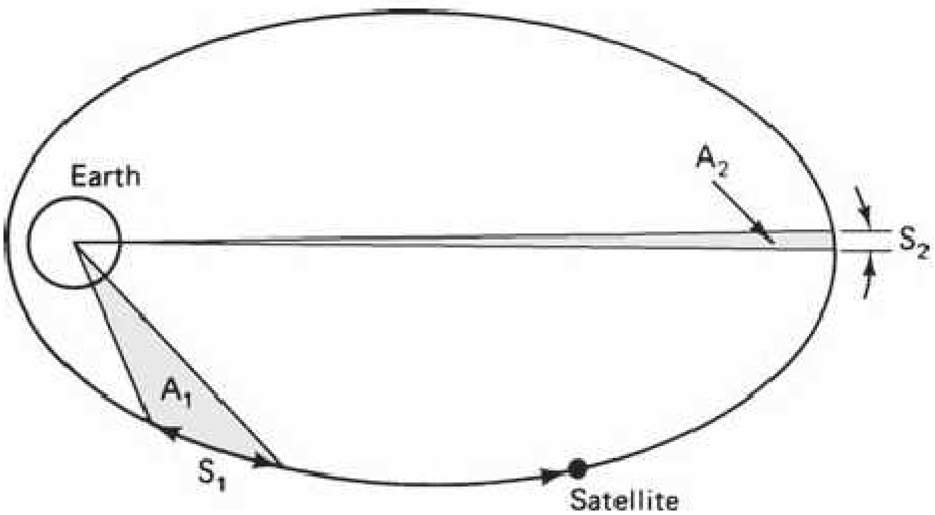


Figure 2.2. Kepler’s second law. The areas *A*1 and *A*2 swept out in unit time are equal.

Kepler’s Third Law:- *Kepler’s third law* states that the square of the periodic time of orbit is proportional to the cube of the mean distance between the two bodies. The mean distance is equal to the semimajor axis *a*. For the
artificial satellites orbiting the earth, Kepler’s third law can be written as follows

$$a^3 = \frac{\text{«}}{n^2} \qquad \ldots (1)$$

where *n* is the mean motion of the satellite in radians per second and *μ* is the earth’s geocentric gravitational constant.

$$\text{«} = 3.986005 \times 10^{14}\, m^3 / s^3 \qquad \ldots (2)$$

The importance of Kepler’s third law is that it shows there is a fixed relationship between period and semimajor axis.

2.2 Satellite Orbits

There are many different satellite orbits that can be used. The ones that receive the most attention are the geostationary orbit used as they are stationary above a particular point on the Earth. The orbit that is chosen for a satellite depends upon its application. These orbits are given in table 1.

Geostationary or geosynchronous earth orbit (GEO)

A satellite in a geostationary orbit appears to be stationary with respect to the earth, hence the name *geostationary*. GEO satellites are synchronous with respect to earth. Looking from a fixed point from Earth, these satellites appear to be stationary. These satellites are placed in the space in such a way that only three satellites are sufficient to provide connection throughout the surface of the Earth. GEO satellite travels eastward at the same rotational speed as the earth in circular orbit with zero inclination.

A geostationary orbit is useful for communications because ground antennas can be aimed at the satellite without their having to track the satellite's motion. This is relatively inexpensive. In applications that require a large number of ground antennas, such as DirectTVdistribution, the savings in ground equipment can more than outweigh the cost and complexity of placing a satellite into orbit.

Table: 1

STELLITE ORBIT NAME	ORBIT	SATELLITE ORBIT ALTITUDE (KM ABOVE EARTH'S SURFACE)	APPLICATION
Low Earth Orbit	LEO	200 - 1200	Satellite phones, Navstar or Global Positioning (GPS) system
Medium Earth Orbit	MEO	1200 - 35790	High-speed telephone signals
Geosynchronous Orbit	GSO	35790	Satellite Television
Geostationary Orbit	GEO	35790	Direct broadcast television

Low Earth Orbit (LEO) satellites

A low Earth orbit (LEO) typically is a circular orbit about 200 kilometres (120 mi) above the earth's surface and, correspondingly, a period (time to revolve around the earth) of about 90 minutes. Because of their low altitude, these satellites are only visible from within a radius of roughly 1000 kilometers from the sub-satellite point. In addition, satellites in low earth orbit change their position relative to the ground position quickly. So even for local applications, a large number of satellites are needed if the mission requires uninterrupted connectivity. s. LEO systems try to ensure a high elevation for every spot on earth to provide a high quality communication link. Each LEO satellite will only be visible from the earth for around ten minutes.

Low-Earth-orbiting satellites are less expensive to launch into orbit than geostationary satellites and, due to proximity to the ground, do not require as high signal strength (Recall that signal strength falls off as the square of the distance from the source, so the effect is dramatic). Thus there is a trade off between the number of satellites and their cost. In addition, there are important differences in the onboard and ground equipment needed to support the two types of missions. One general problem of LEOs is the short lifetime of about five to eight years due to atmospheric drag and radiation from the inner Van Allen belt1.

Medium Earth Orbit (MEO) satellites

A MEO satellite is in orbit somewhere between 8,000 km and 18,000 km above the earth's surface. MEO satellites are similar to LEO satellites in functionality. MEO satellites are visible for much longer periods of time than LEO satellites, usually between 2 to 8 hours. MEO satellites have a larger coverage area than LEO satellites. A MEO satellite's longer duration of visibility and wider footprint means fewer satellites are needed in a MEO network than a LEO network. One disadvantage is that a MEO satellite's distance gives it a longer time delay and weaker signal than a LEO satellite, though not as bad as a GEO satellite. Due to the larger distance to the earth, delay increases to about 70–80 ms. so these satellites need higher transmit power and special antennas for smaller footprints.

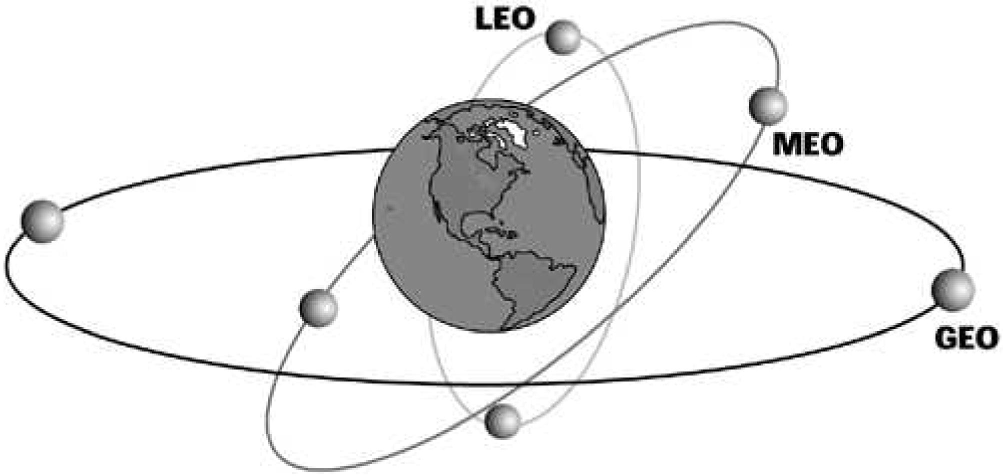


Fig. 2.3 Satellite Orbits

2.3 Spacing and Frequency Allocation

Allocating frequencies to satellite services is a complicated process which requires international coordination and planning. This is carried out under the supervision of the *International Telecommunication Union* (ITU). This frequency allocation is done based on different areas. So this world is divided into three areas.

Area 1:- : Europe, Africa, Soviet Union, and Mongolia

Area 2: North and South America and Greenland

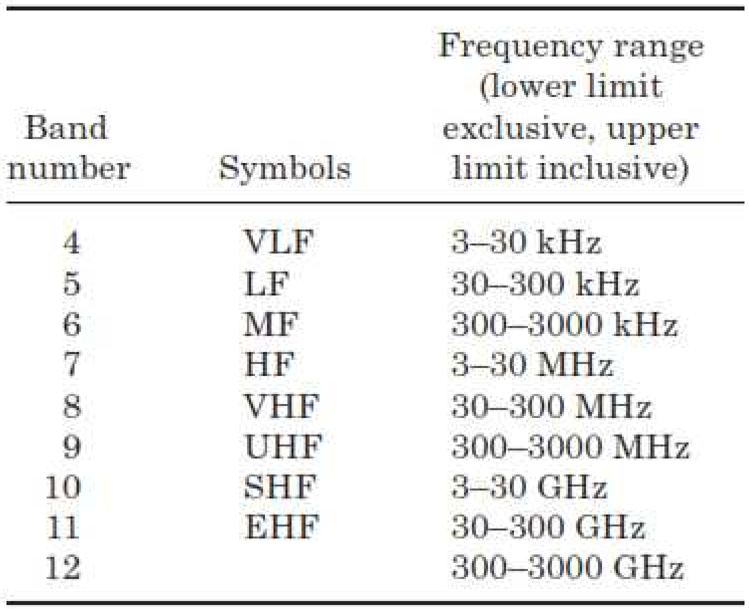
Area 3: Asia (excluding area 1 areas), Australia, and the south-west Pacific

Within these regions, frequency bands are allocated to various satellite services, although a given service may be allocated different frequency bands in different regions. Some of the services provided by satellites are:

- ***Fixed satellite service* (FSS)**

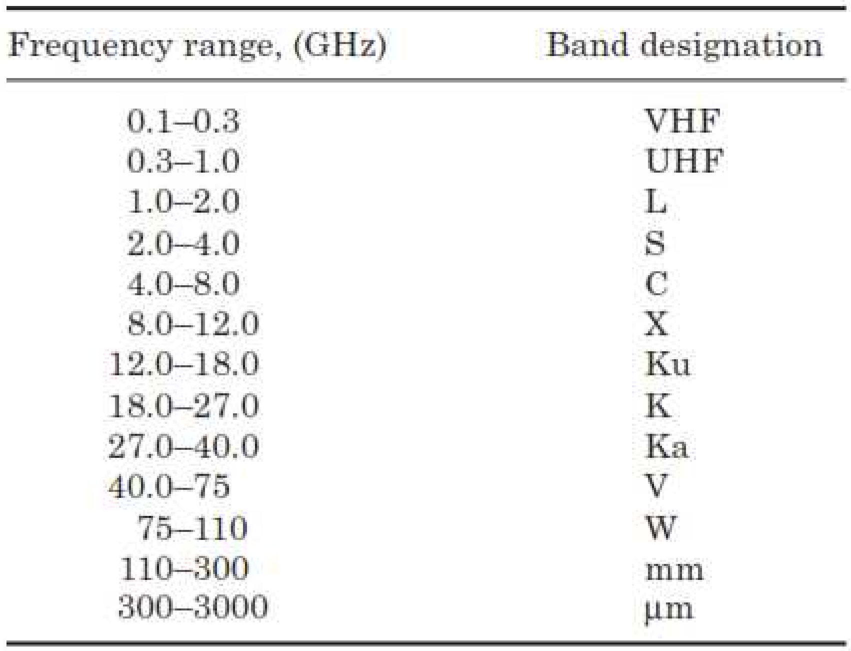
  The FSS provides links for existing telephone networks as well as for transmitting television signals to cable companies for distribution over cable systems. Broadcasting satellite services are intended mainly for direct broadcast to the home, sometimes referred to as *direct broadcast satellite* (DBS) service [in Europe it may be known as *direct-to-home* (DTH) service]. Mobile satellite services would include land mobile, maritime mobile, and aeronautical mobile. Navigational satellite services include *global positioning systems* (GPS), and satellites intended for the meteorological services often provide a search and rescue service.

TABLE 2: ITU Frequency Band Designations



Band number	Symbols	Frequency range (lower limit exclusive, upper limit inclusive)
4	VLF	3–30 kHz
5	LF	30–300 kHz
6	MF	300–3000 kHz
7	HF	3–30 MHz
8	VHF	30–300 MHz
9	UHF	300–3000 MHz
10	SHF	3–30 GHz
11	EHF	30–300 GHz
12		300–3000 GHz

TABLE 3: Frequency Band Designations



Frequency range, (GHz)	Band designation
0.1–0.3	VHF
0.3–1.0	UHF
1.0–2.0	L
2.0–4.0	S
4.0–8.0	C
8.0–12.0	X
12.0–18.0	Ku
18.0–27.0	K
27.0–40.0	Ka
40.0–75	V
75–110	W
110–300	mm
300–3000	μm

- ***Broadcasting satellite service* (BSS)**

  Provides Direct Broadcast to homes. E.g. Live Cricket matches etc.
- ***Mobile satellite services***
  - Land Mobile
  - Maritime Mobile
  - Aeronautical mobile
- ***Navigational satellite services***
  - Include Global Positioning systems
- ***Meteorological satellite services***
  - They are often used to perform Search and Rescue service.

2.4 Look Angle Determination

The satellite look angle refers to the angle that one would look for a satellite at a given time from a specified position on the Earth. The look angles for the ground station antenna are Azimuth and Elevation angles. They are required at the antenna so that it points directly at the satellite. Look angles are calculated by considering the elliptical orbit. These angles change in order to track the satellite.

Azimuth angle:- The azimuth angle is an angle measured from North direction in the local horizontal plane.

Elevation angle:- The elevation angle is the angle measured perpendicular to the horizontal plane (in the vertical plane) to the line-of-sight to the satellite.

The three pieces of information that are needed to determine the look angles for the geostationary orbit are

1. The earth-station latitude, denoted here by ª$_E$

2. The earth-station longitude, denoted here by ¤$_E$

3. The longitude of the subsatellite point, denoted here by ¤$_{SS}$ (this is just referred to as the satellite longitude)

4. ES: Position of Earth Station
5. SS: Sub-Satellite Point
6. S: Satellite
7. d: Range from ES to S
8. σ: angle to be determined

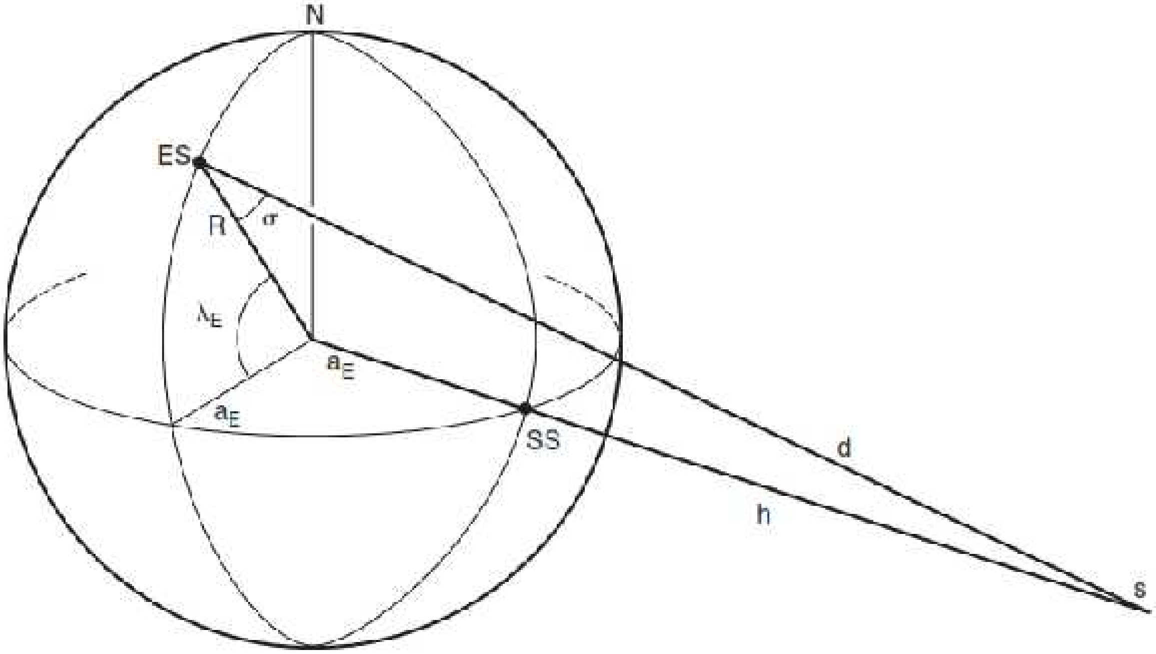


Fig. 2.4:- The geometry used in determining the look angles for a geostationary satellite.

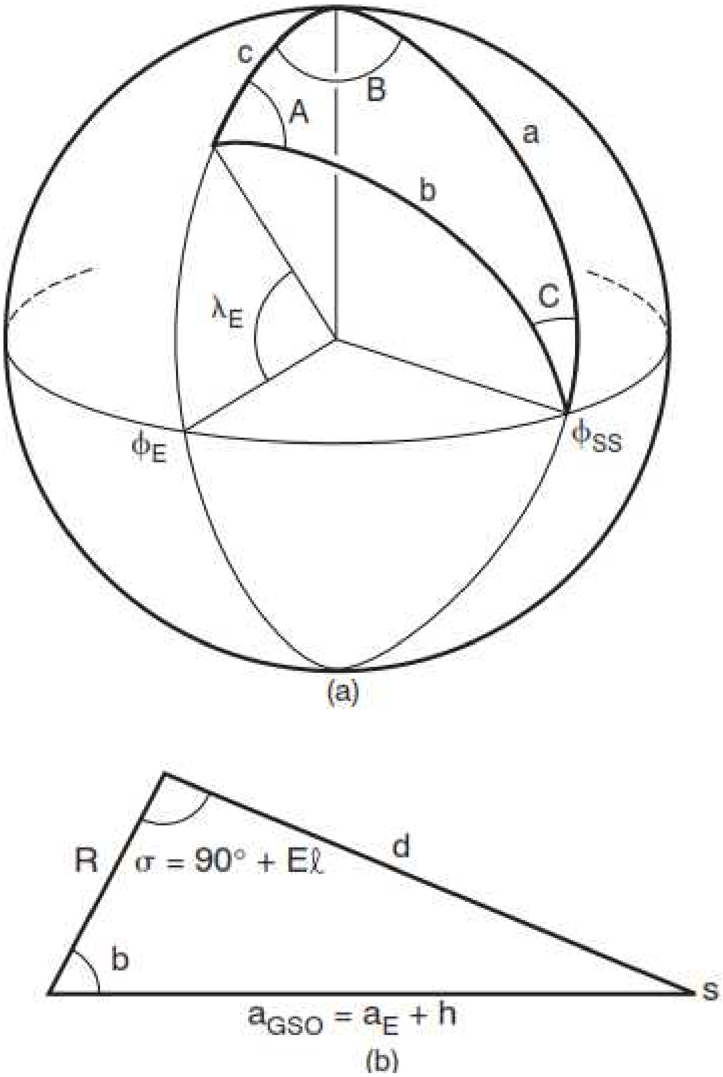


Figure 4.5 (*a*) The spherical geometry related to Fig. 4.4. (*b*) The plane triangle obtained from Fig. 4.4.

There are six angles in all defining the spherical triangle. The three angles *A*, *B*,and *C* are the angles between the planes. Angle *A* is the angle between the plane containing *c* and the plane containing *b.* Angle *B* is the angle between the plane containing *c* and the plane containing *a*.

Considering figure 5 (b), it's a spherical triangle. All sides are the arcs of a great circle. Three sides of this triangle are defined by the angles subtended by the centre of the earth.

Side a: angle between North Pole and radius of the sub-satellite point.

Side b: angle between radius of Earth and radius of the sub-satellite point.

Side c: angle between radius of Earth and the North Pole.

a = 90^o and such a spherical triangle is called quadrantal triangle. c = 90^o – λ

Angle B is the angle between the plane containing c and the plane containing a.

Thus, $B = ¤_E - ¤_{SS}$

Angle A is the angle between the plane containing b and the plane containing c.

Angle C is the angle between the plane containing a and the plane containing b.

Thus,

$a=90^0$

$c=90^0 - \lambda_E$

$B=f_E - f_{SS}$

Thus, b = arcos (cos B cos a_E

A = arc sin (sin |B| / sin b)

2.5 Orbital Perturbation

The *keplerian orbit* described so far is ideal in the sense that it assumes that the earth is a uniform spherical mass and that the only force acting is the centrifugal force resulting from satellite motion balancing the gravitational pull of the earth. In practice, other forces which can be significant are the gravitational forces of the sun and the moon and atmospheric drag. The gravitational pulls of sun and moon have negligible effect on low-orbiting satellites, but they do affect satellites in the geostationary orbit.

There are two types of perturbation:-

1- **Gravitational:- when considering third body interaction and the non-spherical shape of the earth.**

The earth is very far away from perfectly spherical. This depends on the earth rotation, earth gravitational potential.

2- **Non-gravitational:- like Atmospheric drag, solar radiation pressure and tidal friction.**

For near-earth satellites, below about 1000 km, the effects of atmospheric drag are significant. Because the drag is greatest at the perigee, the drag acts to reduce the velocity at this point, with the result that the satellite does not reach the same apogee height on successive revolutions.

CHAPTER: 3 SATELLITES

3.1 Satellite Launching

A satellite is sent into space on top of a rocket. When a satellite is put into space, we say that it is "launched." The rocket that is used to launch a satellite is called a "launch vehicle." This satellite launching needs the earth stations in order to operate the satellite operation. The satellite launching can be divided into four stages.

1- **First Stage:-** The first stage of the launch vehicle contains the rockets and fuel that are needed to lift the satellite and launch vehicle off the ground and into the sky.

2- **Second Stage:-** The second stage contains smaller rockets that ignite after the first stage is finished. The rockets of the second stage have their own fuel tanks. The second stage is used to send the satellite into space.

3- **Third Stage (Upper Stage):-** The upper stage of the launch vehicle is connected to the satellite itself, which is enclosed in a metal shield, called a "fairing." The fairing protects the satellite while it is being launched and makes it easier for the launch vehicle to travel through the resistance of the Earth's atmosphere.

4- Fourth Stage (Firing):- Once the launch vehicle is out of the Earth's atmosphere, the satellite separates from the upper stage. The satellite is then sent into a "transfer orbit" that sends the satellite higher into space. Once the satellite reaches its desired orbital height, it unfurls its solar panels and communication antennas, which had been stored away during the flight. The satellite then takes its place in orbit with other satellites and is ready to provide communications to the public.

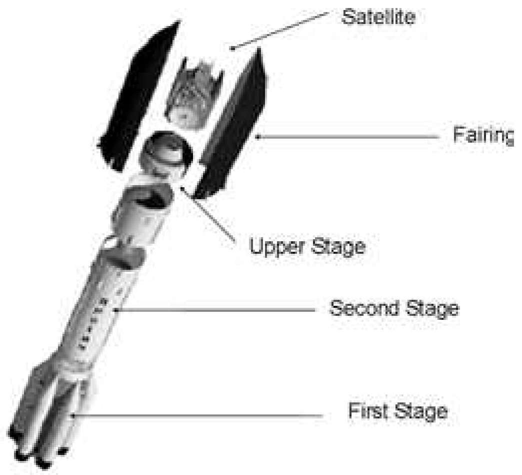


Figure 3.1 Steps of Satellite Launching

The launch process can be divided into two phases: the launch phase and the orbit injection phase.

1- **The Launch Phase**

   The launch vehicle places the satellite into the transfer orbit. An eliptical orbit that has at its farthest point from earth (apogee) the geosynchronous elevation of 22,238 miles and at its nearest point (perigee) an elevation of usually not less than 100 miles.

2- **The Orbit Injection Phase**

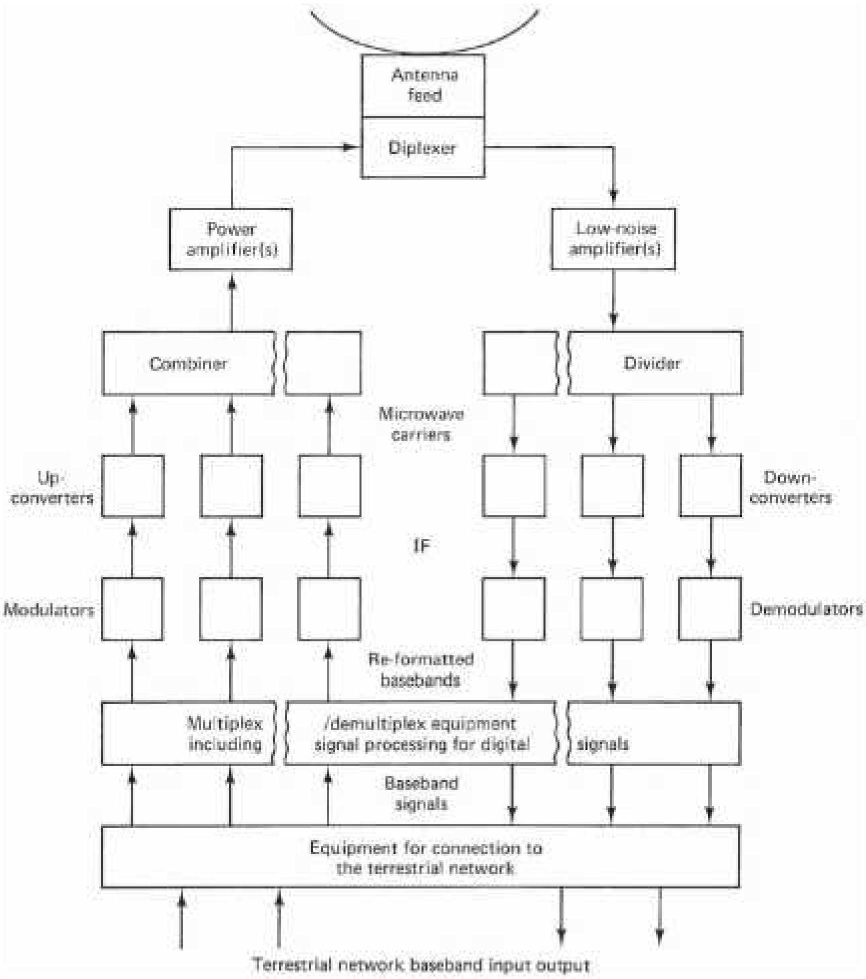
   The energy required to move the satellite from the elliptical transfer orbit into the geosynchronous orbit is supplied by the satellite's apogee kick motor (AKM). This is known as the orbit injection phase.

3.2 Earth Station

The earth segment of a satellite communications system consists of the transmit and receive earth stations. The station's antenna functions in both, the transmit and receive modes, but at different frequencies.

An earth station is generally made up of a multiplexor, a modem, up and downconverters, a high power amplifier (HPA) and a low noiseamplifier (LNA). Almost all transmission to satellites is d igital, and the digital data streams are combined in a multiplexor and fed to a modemthat modula tes a carrier frequency in the 50 to 180 MHz range. An upconverter bumps the carrier into the gi gahertz range, which goes to the HPA and antenna.

For receiving, the LNA boosts the signals to the downconverter, which lowers the freque ncy and sends itto the modem. The modemdemodulates the carrier, and the digital output goes to the demultiplexing device and then to its destinations. See earth station on board vessel and base station. A detailed block diagram is shown in fig. 3.2.



and dish.

Figure 3.2:- Block diagram of a transmit-receive earth station

3.3 Satellite Sub-systems

A satellite communications system can be broadly divided into two segments—a ground segment and a space segment. The space segment will obviously include the satellites, but it also includes the ground facilities needed to keep the satellites operational, these being referred to as the *tracking, telemetry, and command* (TT&C) facilities. In many networks it is common practice to employ a ground station solely for the purpose of TT&C.

In a communications satellite, the equipment which provides the connecting link between the satellite's transmit and receive antennas is referred to as the *transponder*. The transponder forms one of the main sections of the payload, the other being the antenna subsystems.

PAYLOAD:- The payload comprises of a Repeater and Antenna subsystem and performs the primary function of communication.

1- **REPEATER:-** It is a device that receives a signal and retransmits it to a higher level and/or higher power onto the other side of the obstruction so that the signal can cover longer distance.
2- **Transparent Repeater:-** It only translates the uplink frequency to an appropriate downlink frequency. It does so without processing the baseband
   signal. The main element of a typical transparent repeater is a single beam satellite. Signals from antenna and the feed system are fed into the low-noise amplifier through a bandpass filter.
3- **Regenerative Repeater :-** A repeater, designed for digital transmission, in which digital signals are amplified, reshaped, retimed, and retransmitted.
   Regenerative Repeater can also be called as a device which regenerates incoming digital signals and then retransmits these signals on an outgoing circuit.
4- **Antennas :-** The function of an antenna of a space craft is to receive signals
   and transmit signals to the ground stations located within the coverage area of the satellite. The choice of the antenna system is therefore governed by the size and shape of the coverage area. Consequently, there is also a limit to the minimum size of the antenna footprint.

3.4 Satellite System Link Models

System Link Budget calculations basically relate two quantities, the transmit power and the receive power, and show in detail how the difference between these two powers is accounted for. Link-power budget calculations also need the additional losses and noise factor which is incorporated with the transmitted and the received signals. Along with losses, this unit also discusses the system noise parameters. Various components of the system add to the noise in the signal that has to be transmitted.

3.4.1 EQUIVALENT ISOTROPIC RADIATED POWER

The key parameter in link-power budget calculations is the equivalent isotropic radiated power factor, commonly denoted as EIRP. Is the amount of power that a theoretical isotropic antenna (which evenly distributes power in all directions) would emit to produce the peak power density observed in the direction of maximum antenna gain. EIRP can be defined as the power input to one end of the transmission link and the problem to find the power received at the other end.

EIRP = G Ps

Where,
G - Gain of the Transmitting antenna and G is in decibels.
Ps- Power of the sender (transmitter) and is calculated in watts.

[EIRP] = [G] + [Ps] dBW

3.4.2 TRANSMISSION LOSSES:-

As EIRP is thought of as power input of one end to the power received at the other, the problem here is to find the power which is received at the other end. Some losses that occur in the transmitting – receiving process are constant and their values can be pre – determined.

3.4.2.1 Free-Space Transmission Losses (FSL)

This loss is due to the spreading of the signal in space. Going back to the power flux density equation

$$\cdot_{m} = P_{s} / 4\circledR r^{2}$$

The power that is delivered to a matched receiver is the power flux density. It is multiplied by the effective aperture of the receiving antenna. Hence, the received power is:

$$P_{R} = \cdot_{M} A_{eff}$$

$$= \frac{EIRP \bullet \text{a}^{2} G_{R}}{4\circledR r^{2}}$$

Where
r- distance between transmitter and receiver, G_R - power gain at the receiver
In decibels, the above equation becomes:

$$[P_R] = [EIRP] + [G_R] - 10\log\left(\frac{4\circledR r}{\text{a}}\right)^{2}$$

$$[FSL] = 10\log\left(\frac{4\circledR r}{\text{a}}\right)^{2}$$

$$[P_R] = [EIRP] + [G_R] - [FSL]$$

3.4.2.2 Feeder Losses (RFL):- This loss is due to the connection between the satellite receiver device and the receiver antenna is improper. Losses here occur is connecting wave guides, filers and couplers. The receiver feeder loss values are added to free space loss.

3.4.2.1 Antenna Misalignment Losses (AML):- To attain a good communication link, the earth station‟s antenna and the communicating satellite‟s antenna must face each other in such a way that the maximum gain is attained.

3.4.2.1 Fixed Atmospheric (AA) and Ionospheric losses (PL):-The gases present in the atmosphere absorb the signals. This kind of loss is usually of a fraction of decibel in quantity. Along with the absorption losses, the ionosphere introduces a good amount of depolarization of signal which results in loss of signal.

3.5 Link Equations

The EIRP can be considered as the input power to a transmission link. Due to the above discussed losses, the power at the receiver that is the output can be considered as a simple calculation of EIRP– losses.

Losses = [FSL] + [RFL] + [AML] + [AA] + [PL]

The received power that is P

$$\left[P_R\right] = \left[EIRP\right] + \left[G_R\right] - [Losses]$$

Where;

$\left[P_R\right]$ - Received power in dB, [EIRP] - equivalent isotropic radiated power in dBW.

$[G_R]$- Isotropic power gain at the receiver and its value is in dB.

[FSL]-Free-space transmission loss in dB.

[RFL] -Receiver feeder loss in dB.

[AA] -Atmospheric absorption loss in dB.

[AML] -Antenna misalignment loss in dB.

[PL] - Depolarization loss in dB.

CHAPTER 4: MODULATION AND MULTIPLEXING TECHNIQUES

4.1 Multiple Access

Multiple accesses is defined as the technique where in more than one pair of earth stations can simultaneously use a satellite transponder. A multiple access scheme is a method used to distinguish among different simultaneous transmissions in a cell. A radio resource can be a different time interval, a frequency interval or a code with a suitable power level.

If the different transmissions are differentiated for the frequency band, it will bedefined as the Frequency Division Multiple Access (FDMA). Whereas, if transmissions are distinguished on the basis of time, then it is considered as Time Division Multiple Access (TDMA). If a different code is adopted to separate simultaneous transmissions, it will be Code Division Multiple Access (CDMA).

4.1.1 Frequency Division Multiple Access (FDMA)

Frequency Division Multiple Access or FDMA is a channel access method used in multiple-access protocols as a channelization protocol. FDMA gives users an individual allocation of one or several frequency bands, or channels. It is particularly commonplace in satellite communication.

- In FDMA all users share the satellite transponder or frequency channel simultaneously but each user transmits at single frequency.
- FDMA can be used with both analog and digital signal.
- FDMA requires high-performing filters in the radio hardware.
- FDMA is not vulnerable to the timing problems that TDMA has. Since a predetermined frequency band is available for the entire period of communication, stream data (a continuous flow of data that may not be packetized) can easily be used with FDMA.
- Each user transmits and receives at different frequencies as each user gets a unique frequency slots.

4.1.2 Time Division Multiple Access (TDMA)

Time division multiple access (TDMA) is a channel access method for shared medium networks. It allows several users to share the same frequency channel by dividing the signal into different time slots. This allows multiple stations to share the same transmission medium (e.g. radio frequency channel) while using only a part of its channel capacity.

- Shares single carrier frequency with multiple users.
- Slots can be assigned on demand in dynamic TDMA.
- Less stringent power control than CDMA due to reduced intra cell interference
- Higher synchronization overhead than CDMA
- Cell breathing (borrowing resources from adjacent cells) is more complicated than in CDMA.
- Frequency/slot allocation complexity.

4.1.3 Code Division Multiple Access (CDMA)

Code division multiple access (CDMA) is a channel access method used by various radio communication technologies. CDMA is an example of multiple access, which is where several transmitters can send information simultaneously over a single communication channel. This allows several users to share a band of frequencies (see bandwidth). CDMA is used as the access method in many mobile phone standards such as cdmaOne, CDMA2000 (the 3G evolution of cdmaOne), and WCDMA (the 3G standard used by GSM carriers), which are often referred to as simply *CDMA*.

- One of the early applications for code division multiplexing is in the Global Positioning System (GPS). This predates and is distinct from its use in mobile phones.
- The Qualcomm standard IS-95, marketed as cdmaOne.
- The Qualcomm standard IS-2000, known as CDMA2000, is used by several mobile phone companies, including the Globalstar satellite phone network.
- The UMTS 3G mobile phone standard, which uses W-CDMA.
- CDMA has been used in the Omni TRACS satellite system for transportation logistics.

4.2 Direct Broadcast Satellite Services

Direct-broadcast satellite (DBS) is a type of artificial satellite which usually sends satellite television signals for home reception through geostationary satellites. The type of satellite television which uses direct-broadcast satellites is known as direct-broadcast satellite television (DBSTV) or direct-to-home television (DTHTV). This has initially distinguished the transmissions directly intended for home viewers from cable television distribution services that are sometimes carried on the same satellite.

A DBS subscriber installation consists of a dish antenna two to three feet (60 to 90 centimeters) in diameter, a conventional TV set, a signal converter placed next to the TV set, and a length of coaxial cable between the dish and the converter. The dish intercepts microwave signals directly from the satellite. The converter produces output that can be viewed on the TV receiver. Broadcast services include audio, television, and Internet services. Direct broadcast television, which is digital TV, is the subject of this chapter. A Typical DBS system block diagram is shown in fig. 4.1. The home receiver consists of two units—an outdoor unit and an indoor unit.

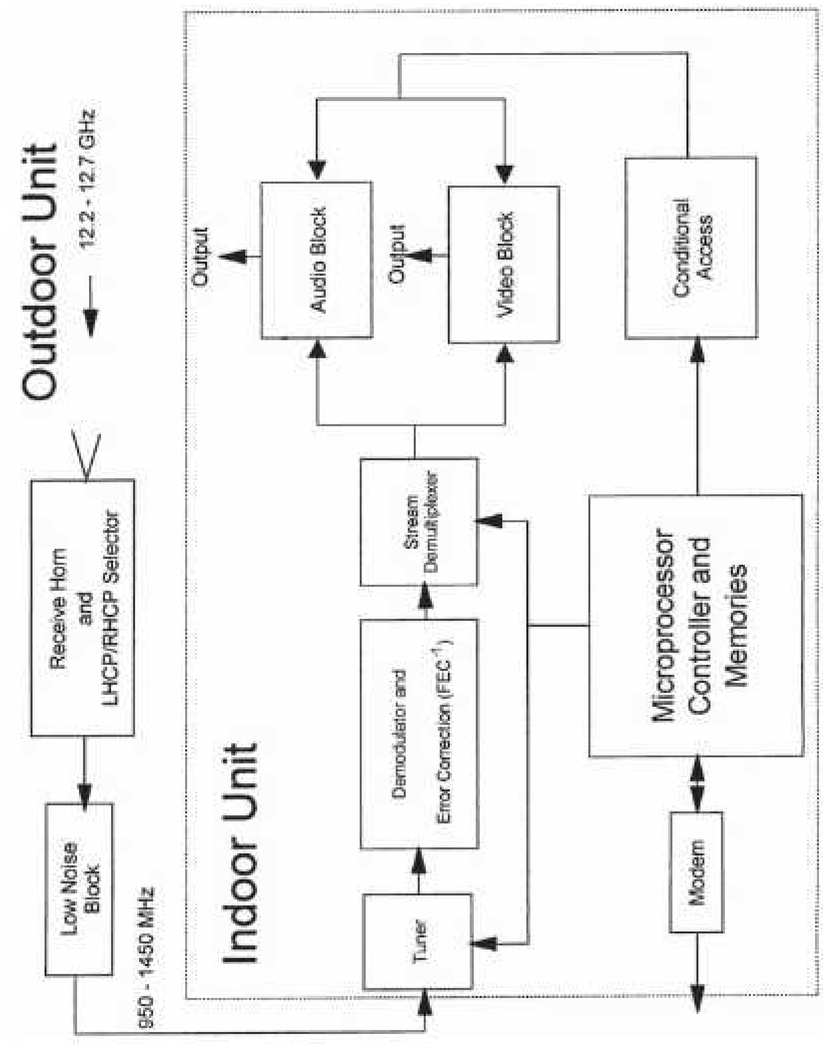


Fig. 4.1 Block schematic for the outdoor unit (ODU) and IDU unit of DBS.

The Home Receiver Outdoor Unit (ODU):- The downlink signal, covering the frequency range 12.2 to 12.7 GHz, is focused by the antenna into the receive horn. The horn feeds into a polarizer that can be switched to pass either left-hand circular or right-hand circular polarized signals. The low-noise block that follows the polarizer contains a *low-noise amplifier* (LNA) and a down converter. The down converter converts the 12.2- to 12.7-GHz band to 950 to 1450 MHz, a frequency range better suited to transmission through the connecting cable to the indoor unit.

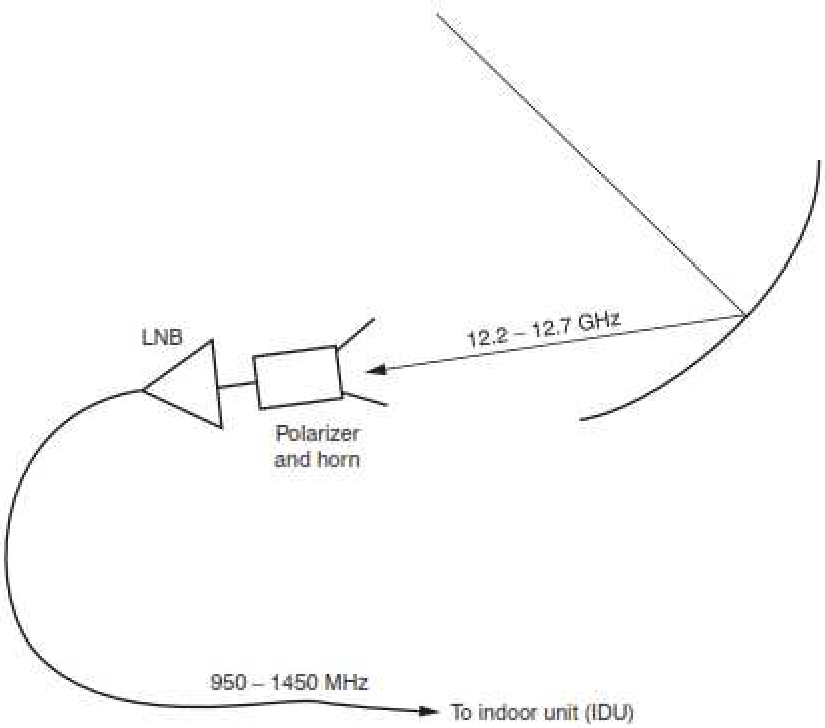


Fig. 4.2 Block schematic for the outdoor unit (ODU)

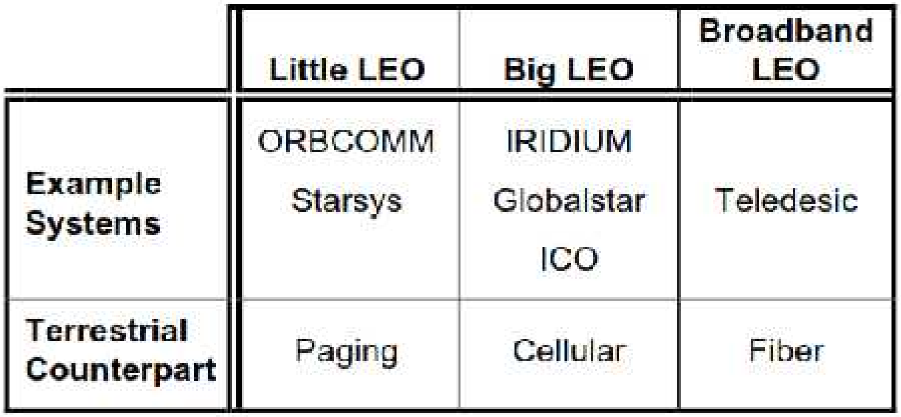
The Home Receiver Indoor Unit (IDU):- The transponder frequency bands shown in Fig. 16.2 are down converted to be in the range 950 to 1450 MHz, but of course, each transponder retains its 24-MHz bandwidth. The IDU must be able to receive any of the 32 transponders, although only 16 of these will be available for a single polarization. The tuner selects the desired transponder. It should be recalled that the carrier at the center frequency of the transponder is QPSK modulated by the bit stream, which itself may consist of four to eight TV programs TDM. Following the tuner, the carrier is demodulated, the QPSK modulation being converted to a bit stream. Error correction is carried out in the decoder block labeled FEC.

4.3 Application of LEO

The evolution from geo-stationary to low-Earthorbit (LEO) satellites has resulted in a number of proposed global satellite systems, which can be grouped into three distinct types - Little LEOs, Big LEOs, and Broadband LEOs. These systems can best be distinguished by reference to their terrestrial counterparts: paging, cellular, and fiber, as shown in Table 4.1. On the ground, paging, cellular, and fiber services are complementary, not competitive, because they offer fundamentally different kinds of services. Similarly, the Little LEOs, Big LEOs, and Broadband

LEOs are complementary rather than competitive because they are providing distinctly different services targeted at different markets, and have different pricing structures.

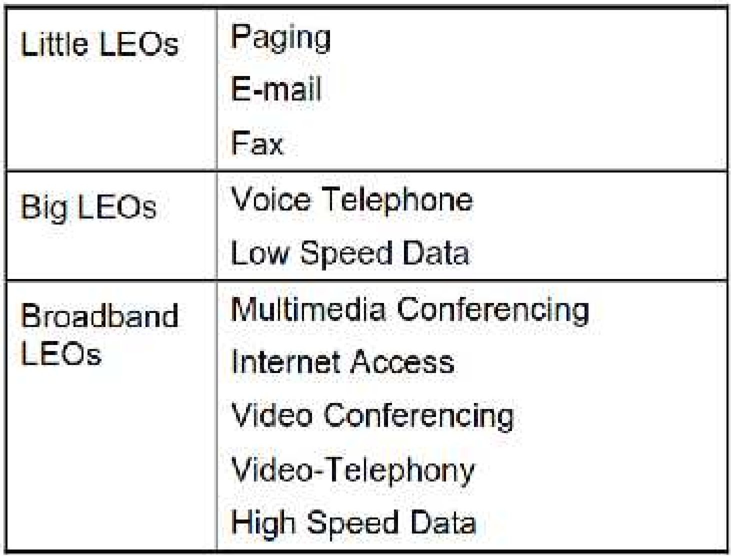
Table 4.1 Terrestrial Counterparts



	Little LEO	Big LEO	Broadband LEO
Example Systems	ORBCOMM Starsys	IRIDIUM Globalstar ICO	Teledesic
Terrestrial Counterpart	Paging	Cellular	Fiber

Typical applications of the various types of LEO systems are shown in Table 2. Of course the Big LEOs can support the Little LEO applications, and the Broadband LEOs can support both the Big and Little LEO applications.

Table 4.2 Application of LEO Satellites



Little LEOs	Paging E-mail Fax
Big LEOs	Voice Telephone Low Speed Data
Broadband LEOs	Multimedia Conferencing Internet Access Video Conferencing Video-Telephony High Speed Data

4.4 MEO and GEO Satellites

Medium Earth orbit (**MEO**), sometimes called **intermediate circular orbit** (**ICO**), is the region of space around the Earth above low Earth orbit (altitude of 2,000 kilometres (1,243 mi)) and below geostationary orbit (altitude of 35,786 kilometres (22,236 mi)).The most common use for satellites in this region is for navigation, communication, and geodetic/space environment science.[1] The most common altitude is approximately 20,200 kilometres (12,552 mi)), which yields an orbital period of 12 hours, as used, for example, by the Global Positioning System (GPS). Other satellites in Medium Earth Orbit include Glonass (with an altitude of 19,100 kilometres (11,868 mi)) and Galileo (with an altitude of 23,222 kilometres (14,429 mi)) constellations.[*citation needed*] Communications satellites that cover the North and South Pole are also put in MEO.

Geostationary satellites appear to be fixed over one spot above the equator. Receiving and transmitting antennas on the earth do not need to track such a satellite. These antennas can be fixed in place and are much less expensive than tracking antennas. These satellites have revolutionized global communications, television broadcasting and weather forecasting, and have a number of important defenseand intelligence applications.

One disadvantage of geostationary satellites is a result of their high altitude: radio signals take approximately 0.25 of a second to reach and return from the satellite, resulting in a small but significant signal delay. This delay increases the difficulty of telephone conversation and reduces the performance of common network protocols such as TCP/IP, but does not present a problem with non-interactive systems such as television broadcasts. There are a number of proprietary satellite data protocols that are designed to proxy TCP/IP connections over long-delay satellite links—these are marketed as being a partial solution to the poor performance of native TCP over satellite links. TCP presumes that all loss is due to congestion, not errors, and probes link capacity with its "slow-start" algorithm, which only sendspackets once it is known that earlier packets have been received. Slow start is very slow over a path using a geostationary satellite. There are approximately 600 geosynchronous satellites, some of which are not operational

Table 4.3 Application of MEO and GEO Satellites

Satellites	Application
Medium Earth Orbit	High-speed telephone signals
Geosynchronous Orbit	Satellite Television
Geostationary Orbit	Direct broadcast television

References

1- http://www.williamcraigcook.com/satellite/index.html

2- http://transition.fcc.gov/cgb/kidszone/satellite/kidz/into_space.html

3- Satellite Communications Dennis Roddy 3rd edition, McGraw Hill publication.

4- http://www.radio-electronics.com/info/satellite/satellite-orbits/satellites-orbit-definitions.php.

5- www.wikipedia.com

6- LEOs - THE COMMUNICATIONS SATELLITES of 21Century by Mark A. Sturza

OPTICAL FIBRE SYSTEM

An optical fiber (or optical fibre) is a flexible, transparent fiber made of extruded glass (silica) or plastic, slightly thicker than a human hair. It can function as a waveguide, or “light pipe”, to transmit light between the two ends of the fiber.The field of applied science and engineering concerned with the design and application of optical fibers is known as fiber optics.

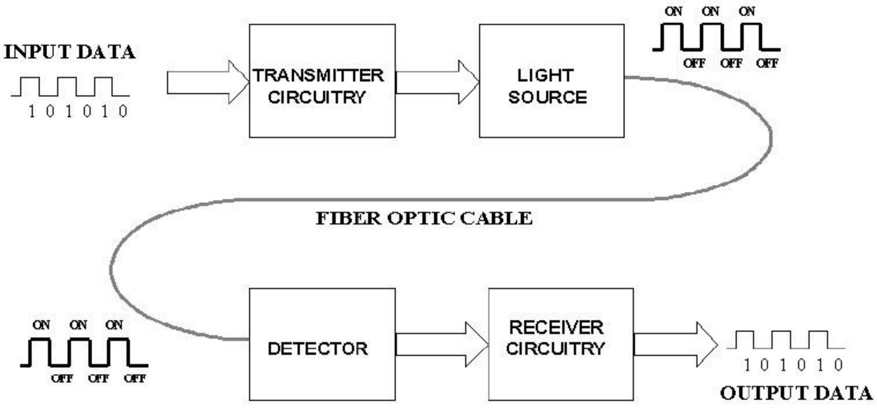
Optical fibers are widely used in fiber-optic communications, where they permit transmission over longer distances and at higher bandwidths (data rates) than wire cables. Fibers are used instead of metal wires because signals travel along them with less loss and are also immune to electromagnetic interference. Fibers are also used for illumination, and are wrapped in bundles so that they may be used to carry images, thus allowing viewing in confined spaces. Specially designed fibers are used for a variety of other applications, including sensors and fiber lasers.

Optical fibers typically include a transparent core surrounded by a transparent cladding material with a lower index of refraction. Light is kept in the core by total internal reflection. This causes the fiber to act as a waveguide. Fibers that support many propagation paths or transverse modes are called multi-mode fibers (MMF), while those that only support a single mode are called single-mode fibers (SMF). Multi-mode fibers generally have a wider core diameter, and are used for short-distance communication links and for applications where high power must be transmitted. Single-mode fibers are used for most communication links longer than 1,000 meters (3,300 ft).

How a Fiber Optic Communication Works?

Unlike copper wire based transmission where the transmission entirely depends on electrical signals passing through the cable, the fiber optics transmission involves transmission of signals in the form of light from one point to the other. Furthermore, a fiber optic communication network consists of transmitting and receiving circuitry, a light source and detector devices like the ones shown in the figure.

When the input data, in the form of electrical signals, is given to the transmitter circuitry, it converts them into light signal with the help of a light source. This source is of LED whose amplitude, frequency and phases must remain stable and free from fluctuation in order to have efficient transmission. The light beam from the source is carried by a fiber optic cable to the destination circuitry wherein the information is transmitted back to the electrical signal by a receiver circuit.



The Receiver circuit consists of a photo detector along with an appropriate electronic circuit, which is capable of measuring magnitude, frequency and phase of the optic field. This type of communication uses the wave lengths near to the infrared band that are just above the visible range. Both LED and Laser can be used as light sources based on the application.

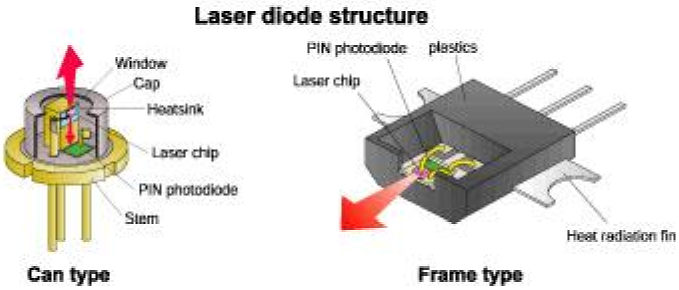
3 Basic Elements of a Fiber Optic Communication System

There are three main basic elements of fiber optic communication system. They are

1. Compact Light Source
2. Low loss Optical Fiber
3. Photo Detector

Accessories like connectors, switches, couplers, multiplexing devices, amplifiers and splices are also essential elements in this communication system.

1. Compact Light Source

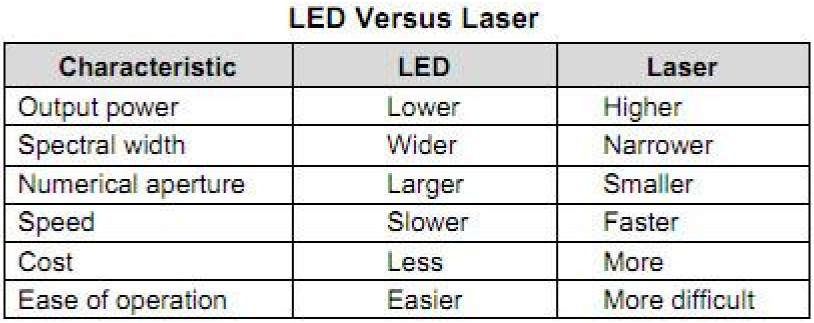


Laser Diodes

Depending on the applications like local area networks and the long haul communication systems, the light source requirements vary. The requirements of the sources include power, speed, spectral line width, noise, ruggedness, cost, temperature, and so on. Two components are used as light sources: light emitting diodes (LED's) and laser diodes.

The light emitting diodes are used for short distances and low data rate applications due to their low bandwidth and power capabilities. Two such LEDs structures include Surface and Edge Emitting Systems. The surface emitting diodes are simple in design and are reliable, but due to its broader line width and modulation frequency limitation edge emitting diode are mostly used. Edge emitting diodes have high power and narrower line width capabilities.

For longer distances and high data rate transmission, Laser Diodes are preferred due to its high power, high speed and narrower spectral line width characteristics. But these are inherently non-linear and more sensitive to temperature variations.



LED Versus Laser

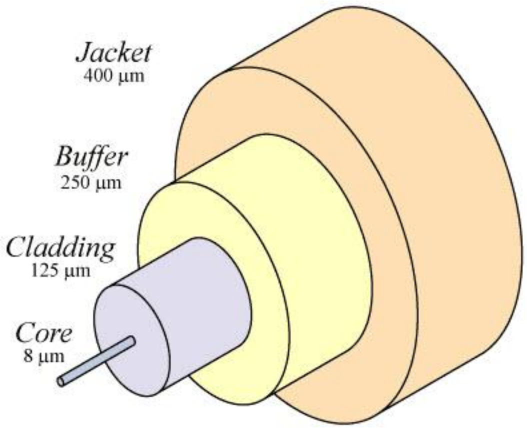
Characteristic	LED	Laser
Output power	Lower	Higher
Spectral width	Wider	Narrower
Numerical aperture	Larger	Smaller
Speed	Slower	Faster
Cost	Less	More
Ease of operation	Easier	More difficult

LED vs Laser Diodes

Nowadays many improvements and advancements have made these sources more reliable. A few of such comparisons of these two sources are given below. Both these sources are modulated using either direct or external modulation techniques.

2. Low Loss Optical Fiber

Optical fiber is a cable, which is also known as cylindrical dielectric waveguide made of low loss material. An optical fiber also considers the parameters like the environment in which it is operating, the tensile strength, durability and rigidity. The Fiber optic cable is made of high quality extruded glass (si) or plastic, and it is flexible. The diameter of the fiber optic cable is in between 0.25 to 0.5mm (slightly thicker than a human hair).



A Fiber Optic Cable consists of four parts.

- Core
- Cladding
- Buffer
- Jacket

Core

The core of a fiber cable is a cylinder of plastic that runs all along the fiber cable's length, and offers protection by cladding. The diameter of the core depends on the application used. Due to internal reflection, the light travelling within the core reflects from the core, the cladding boundary. The core cross section needs to be a circular one for most of the applications.

Cladding

Cladding is an outer optical material that protects the core. The main function of the cladding is that it reflects the light back into the core. When light enters through the core (dense material) into the cladding(less dense material), it changes its angle, and then reflects back to the core.

Buffer

The main function of the buffer is to protect the fiber from damage and thousands of optical fibers arranged in hundreds of optical cables. These bundles are protected by the cable's outer covering that is called jacket.

JACKET

Fiber optic cable's jackets are available in different colors that can easily make us recognize the exact color of the cable we are dealing with. The color yellow clearly signifies a single mode cable, and orange color indicates multimode.

2 Types of Optical Fibers

Single-Mode Fibers: Single mode fibers are used to transmit one signal per fiber; these fibers are used in telephone and television sets. Single mode fibers have small cores.

Multi-Mode Fibers: Multimode fibers are used to transmit many signals per fiber; these signals are used in computer and local area networks that have larger cores.

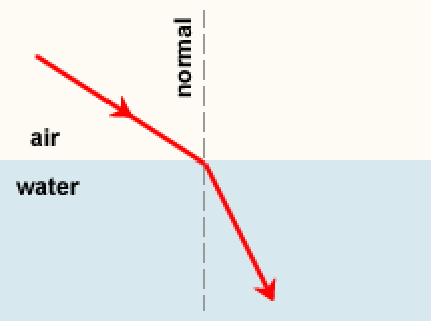
3. Photo Detectors

The purpose of photo detectors is to convert the light signal back to an electrical signal. Two types of photo detectors are mainly used for optical receiver in optical communication system: PN photo diode and avalanche photo diode. Depending on the application's wavelengths, the material composition of these devices vary. These materials include silicon, germanium, InGaAs, etc.

Basic optical laws

Refraction of light

As a light ray passes from one transparent medium to another, it changes direction; this phenomenon is called refraction of light. How much that light ray changes its direction depends on the refractive index of the mediums.



Refractive Index

Refractive index is the speed of light in a vacuum (abbreviated **c**, **c**=299,792.458km/second) divided by the speed of light in a material (abbreviated **v**). Refractive index measures how much a material refracts light. Refractive index of a material, abbreviated as **n**, is defined as

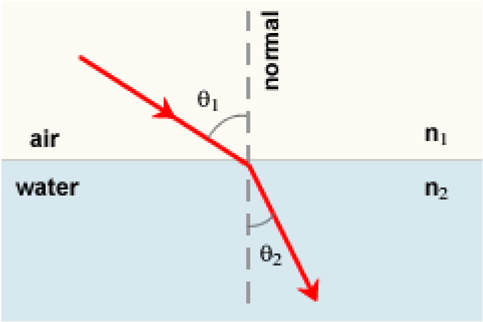
n=c/v

Snell’s Law

In 1621, a Dutch physicist named Willebrord Snell derived the relationship between the different angles of light as it passes from one transparent medium to another. When light passes from one transparent material to another, it bends according to Snell's law which is defined as:

$$\boldsymbol{n_1 \sin(\theta_1) = n_2 \sin(\theta_2)}$$

where:
$\boldsymbol{n_1}$ is the refractive index of the medium the light is leaving
$\boldsymbol{\theta_1}$ is the incident angle between the light beam and the normal (normal is 90° to the interface between two materials)
$\boldsymbol{n_2}$ is the refractive index of the material the light is entering
$\boldsymbol{\theta_2}$ is the refractive angle between the light ray and the normal

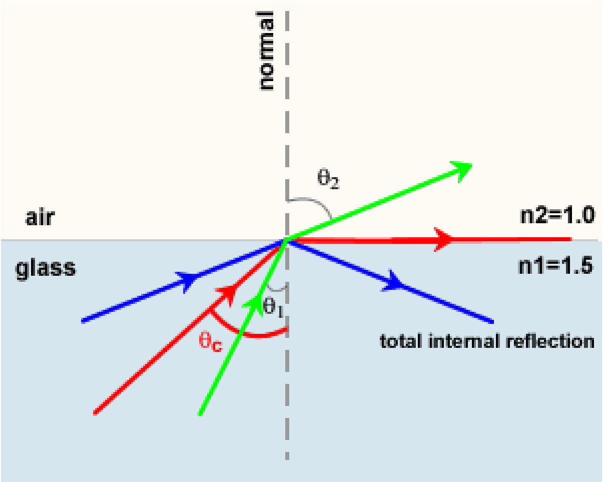


Note:

For the case of $\theta_1 = 0°$ (i.e., a ray perpendicular to the interface) the solution is $\theta_2 = 0°$ regardless of the values of n_1 and n_2. That means a ray entering a medium perpendicular to the surface is never bent.

The above is also valid for light going from a dense (higher n) to a less dense (lower n) material; the symmetry of Snell's law shows that the same ray paths are applicable in opposite direction.

Total Internal Reflection



When a light ray crosses an interface into a medium with a higher refractive index, it bends towards the normal. Conversely, light traveling cross an interface from a higher refractive index medium to a lower refractive index medium will bend away from the normal.

This has an interesting implication: at some angle, known as the **critical angle θ_c**, light traveling from a higher refractive index medium to a lower refractive index medium will be refracted at 90°; in other words, refracted along the interface.

If the light hits the interface at any angle larger than this critical angle, it will not pass through to the second medium at all. Instead, all of it will be reflected back into the first medium, a process known as **total internal reflection**.

The critical angle can be calculated from Snell's law, putting in an angle of 90° for the angle of the refracted ray θ_2. This gives θ_1:

$$\theta_1 = \arcsin[(n_2/n_1)*\sin(\theta_2)]$$

Since

$\theta_2 = 90°$

So

$\sin(\theta_2) = 1$

Then

$\theta_c = \theta_1 = \arcsin(n_2/n_1)$

For example, with light trying to emerge from glass with n_1=1.5 into air (n_2 =1), the critical angle θ_c is arcsin(1/1.5), or 41.8°.

For any angle of incidence larger than the critical angle, Snell's law will not be able to be solved for the angle of refraction, because it will show that the refracted angle has a sine larger than 1, which is not possible. In that case all the light is totally reflected off the interface, obeying the law of reflection.

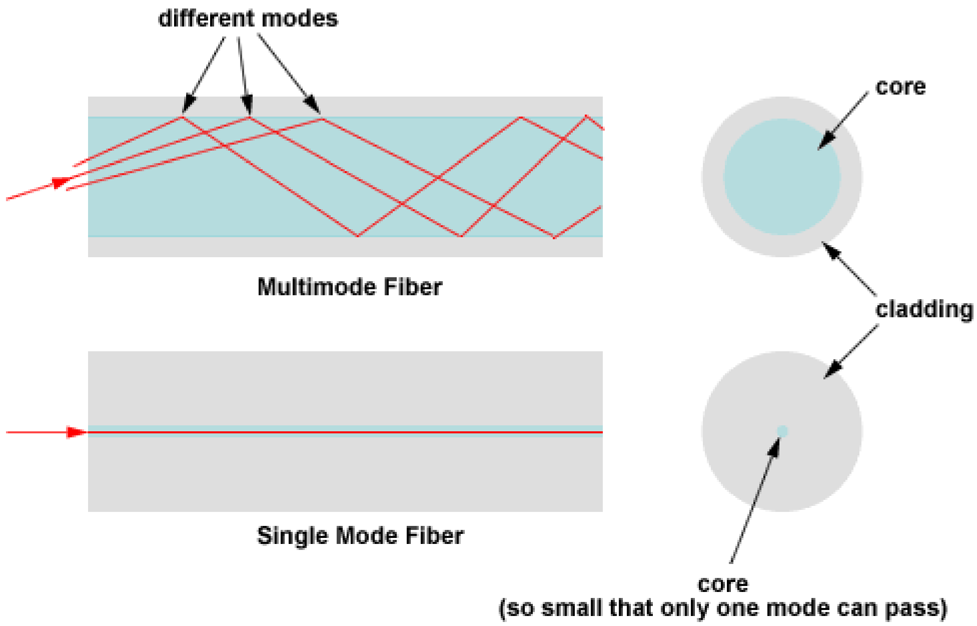
Optical Fiber Mode

What is Fiber Mode?

An optical fiber guides light waves in distinct patterns called *modes*. Mode describes the distribution of light energy across the fiber. The precise patterns depend on the wavelength of light transmitted and on the variation in refractive index that shapes the core. In essence, the variations in refractive index create boundary conditions that shape how light waves travel through the fiber, like the walls of a tunnel affect how sounds echo inside.

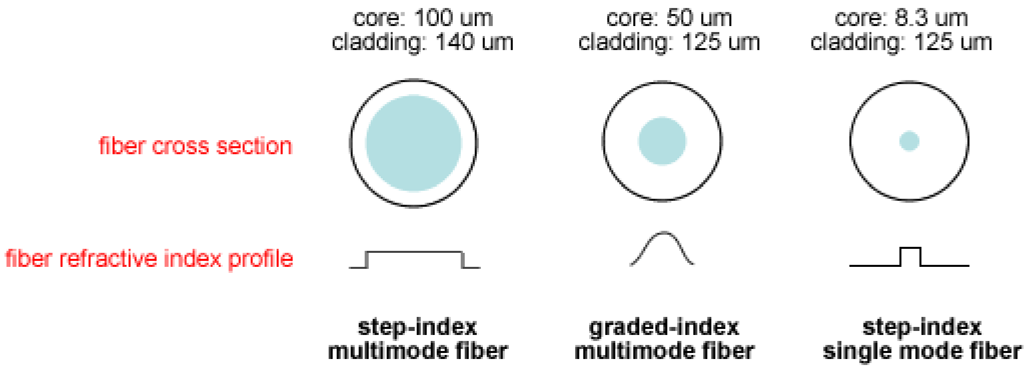
We can take a look at large-core step-index fibers. Light rays enter the fiber at a range of angles, and rays at different angles can all stably travel down the length of the fiber as long as they hit the core-cladding interface at an angle larger than critical angle. These rays are different modes.

Fibers that carry more than one mode at a specific light wavelength are called multimode fibers. Some fibers have very small diameter core that they can carry only one mode which travels as a straight line at the center of the core. These fibers are single mode fibers. This is illustrated in the following picture.



Optical Fiber Index Profile

Index profile is the refractive index distribution across the core and the cladding of a fiber. Some optical fiber has a step index profile, in which the core has one uniformly distributed index and the cladding has a lower uniformly distributed index. Other optical fiber has a graded index profile, in which refractive index varies gradually as a function of radial distance from the fiber center. Graded-index profiles include power-law index profiles and parabolic index profiles. The following figure shows some common types of index profiles for single mode and multimode fibers.



Multimode Fibers

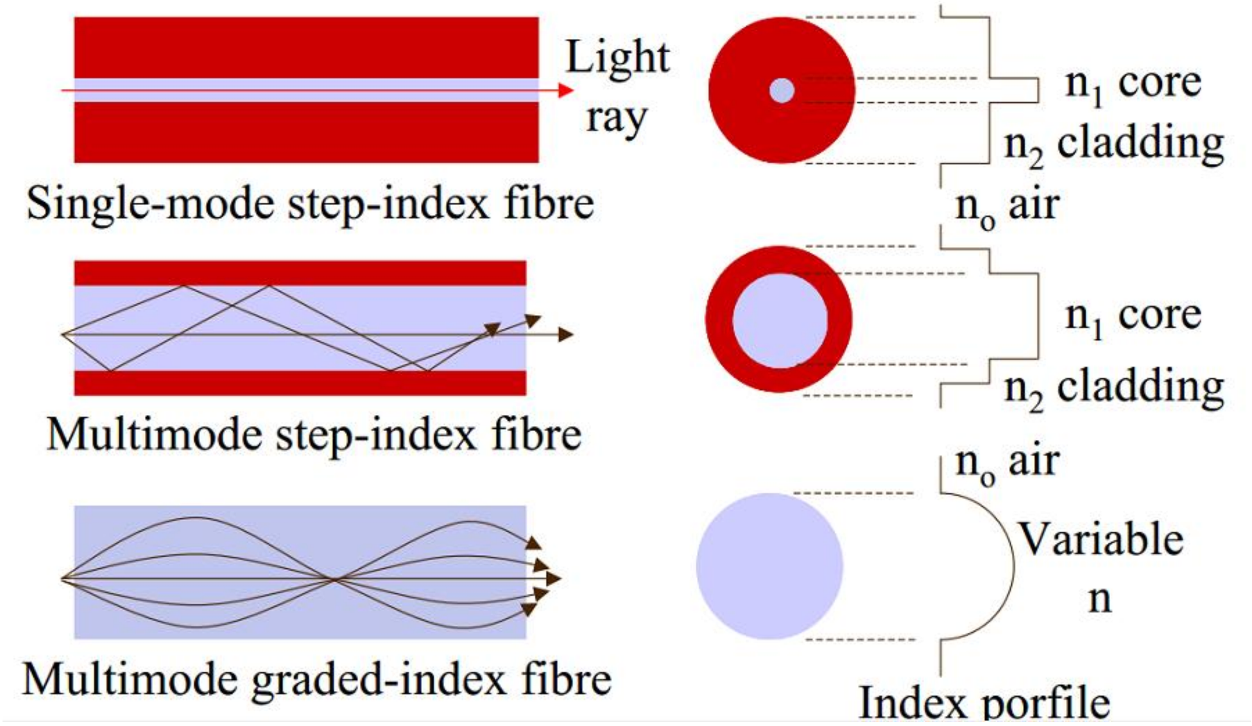
As their name implies, multimode fibers propagate more than one mode. Multimode fibers can propagate over 100 modes. The number of modes propagated depends on the core size and numerical aperture (NA).

As the core size and NA increase, the number of modes increases. Typical values of fiber core size and NA are 50 to 100 micrometer and 0.20 to 0.29, respectively.

Single Mode Fibers

The core size of single mode fibers is small. The core size (diameter) is typically around 8 to 10 micrometers. A fiber core of this size allows only the fundamental or lowest order mode to propagate around a 1300 nanometer (nm) wavelength. Single mode fibers propagate only one mode, because the core size approaches the operational wavelength. The value of the normalized frequency parameter (V) relates core size with mode propagation.

In single mode fibers, V is less than or equal to 2.405. When V = 2.405, single mode fibers propagate the fundamental mode down the fiber core, while highorder modes are lost in the cladding. For low V values (<1.0), most of the power is propagated in the cladding material. Power transmitted by the cladding is easily lost at fiber bends. The value of V should remain near the 2.405 level.



Multimode Step Index Fiber

Core diameter range from 50-1000 m .Light propagate in many different ray paths, or modes, hence the name multimode Index of refraction is same all across the core of the fiber Bandwidth range 20-30 MHz . Multimode Graded Index Fiber The index of refraction across the core is gradually changed from a maximum at the center to a minimum near the edges, hence the name “Graded Index” Bandwidth ranges from 100MHz-Km to 1GHz-Km

Pulse dispersion in a step index optical fiber is given by

$$\text{pulse dispersion} = \frac{\triangle\, n_1\, \ell}{c}$$

where

$\triangle$is the difference in refractive indices of core and cladding.

n1is the refractive index of core

$\ell$is the length of the optical fiber under observation

$$c = 3 \times 10^8\,\mathrm{ms}^{-1}$$

Graded-Index Multimode Fiber

Contains a core in which the refractive index diminishes gradually from the center axis out toward the cladding. The higher refractive index at the center makes the light rays moving down the axis advance more slowly than those near the cladding. Due to the graded index, light in the core curves helically rather than zigzag off the cladding, reducing its travel distance. The shortened path and the higher speed allow light at the periphery to arrive at a receiver at about the same time as the slow but straight rays in the core axis. The result: digital pulse suffers less dispersion. This type of fiber is best suited for local-area networks.

Pulse dispersion in a graded index optical fiber is given by

$$\text{Pulse dispersion} = \frac{k\delta n\ n_1\ l}{c},$$

where

$\delta n$is the difference in refractive indices of core and cladding,

$n_1$is the refractive index of the cladding,

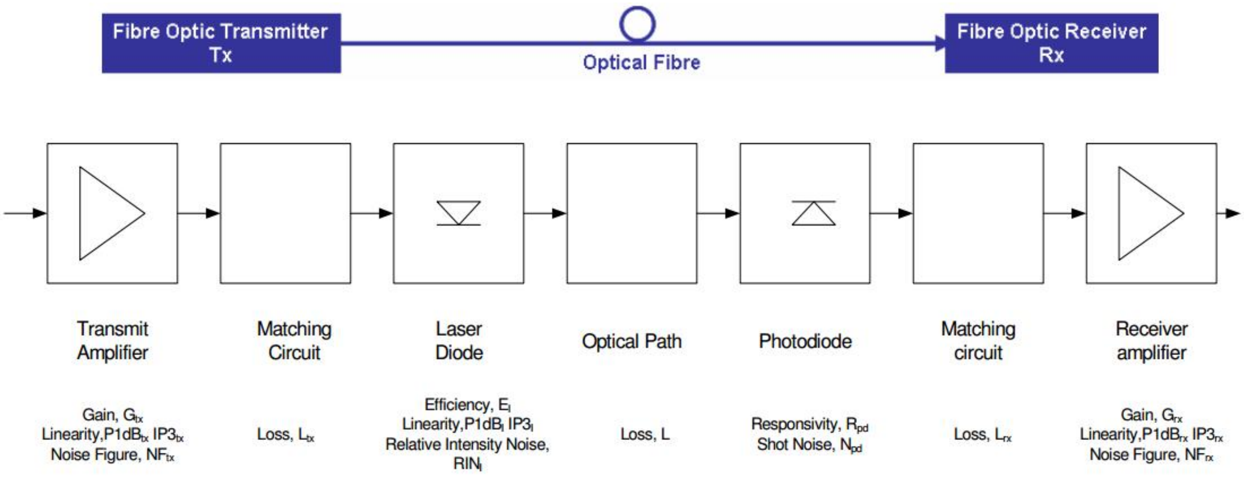
$l$is the length of the fiber taken for observing the pulse dispersion,

$c \approx 3 \times 10^8 \text{ m/s}$is the speed of light, and

k is the constant of graded index profile.

Fibre Optic Link Budget

The FOL budget provides the design engineer with quantitative performance information about the FOL.It is determined by computing the FOL power budget and overall link gain.



Fibre Optic Power Budget

The FOL power budget (PB) is simply the difference between the maximum and minimum signals that the FOL can transport.

Fibre Optic Link Gain

FOL link gain is a summation of gains and losses derived from the different elements of the FOL as shown in above figure . Gains and losses attributed to the Tx, Rx, optical fibre and connectors, as well as any additional in-line components such as splitters, multiplexers, splices etc, must be taken into accounts when computing the linkloss budget.

In the case of a simple point-to-point link described in Above figure , and resistively matched (50 ohms) components,

the link gain (G) is expressed as:-

$G = T + R - 2LO$ (1)

Where T is the gain of the Tx, R is the gain of theRx, and LO is the insertion loss attributed to the fibre link. Note the factor of two in this last optical term, meaning that for each dB optical loss there is a corresponding 2dB RF loss.

To calculate LO the following information is needed.

Standard Corning SMF28 single mode fibre has an insertion loss 0.2dB/km at 1310nm and 0.15dB/km at 1550nm. Optical connectors such as FC/APC typically have an insertion loss of 0.25dB. Optical splices introduce a further 0.25dB loss. Refer to TIA 568 standard forInterfacility and Premise cable specifications.

Output Noise Power

The output noise power of an analogue FOL must alsobe considered when quantifying the overall link budget. The measured output noise power is defined as:-

Output Noise Power = ONF + 10log10 (BW)

Where ONF (Optical Noise Floor) is the noise output of the link on its own, defined in a bandwidth of 1Hz,and BW is the bandwidth of the service transported over fibre. In a real installation, the NF, or Noise Figure is used to define the noise performance of the fibre optic link and is related to the output noise floor as follows:

ONF = -174dBm + NF + G (3)

-174dBm, is the noise contribution from an ideal 1ohm resistive load at zero degrees Kelvin.

The measured output noise power is given as:-

= -174dBm + NF + G + 10log10 (MBW)

ATTENUATION ON OPTICAL FIBER

The signal on optical attenuates due to following mechanisms.

1. Intrinsic loss in the fiber material.
2. Scattering due to micro irregularities inside the fiber.
3. Micro-bending losses due to micro-deformation of the fiber.
4. Bending or radiation losses on the fiber.

The first two losses are intrinsically present in any fiber and the last two depend on the environment in which the fiber is laid.

Material Loss

(a) Due to impurities: The material loss is due to the impurities present in glass used for making fibers. Inspite of best purification efforts, there are always impurities like Fe, Ni, Co, Al which are present in the fiber material. The Fig. shows attenuation due to various molecules inside glass as a function of wavelength. It can be noted from the figure that the material loss due to impurities reduces substantially beyond about 1200nm wavelength.

(b)Due to OH molecule: In addition, the OH molecule diffuses in the material and causes absorption of light. The OH molecule has main absorption peak somewhere in the deep infra-red wavelength region. However, it shows substantial loss in the range of 1000 to 2000nm.

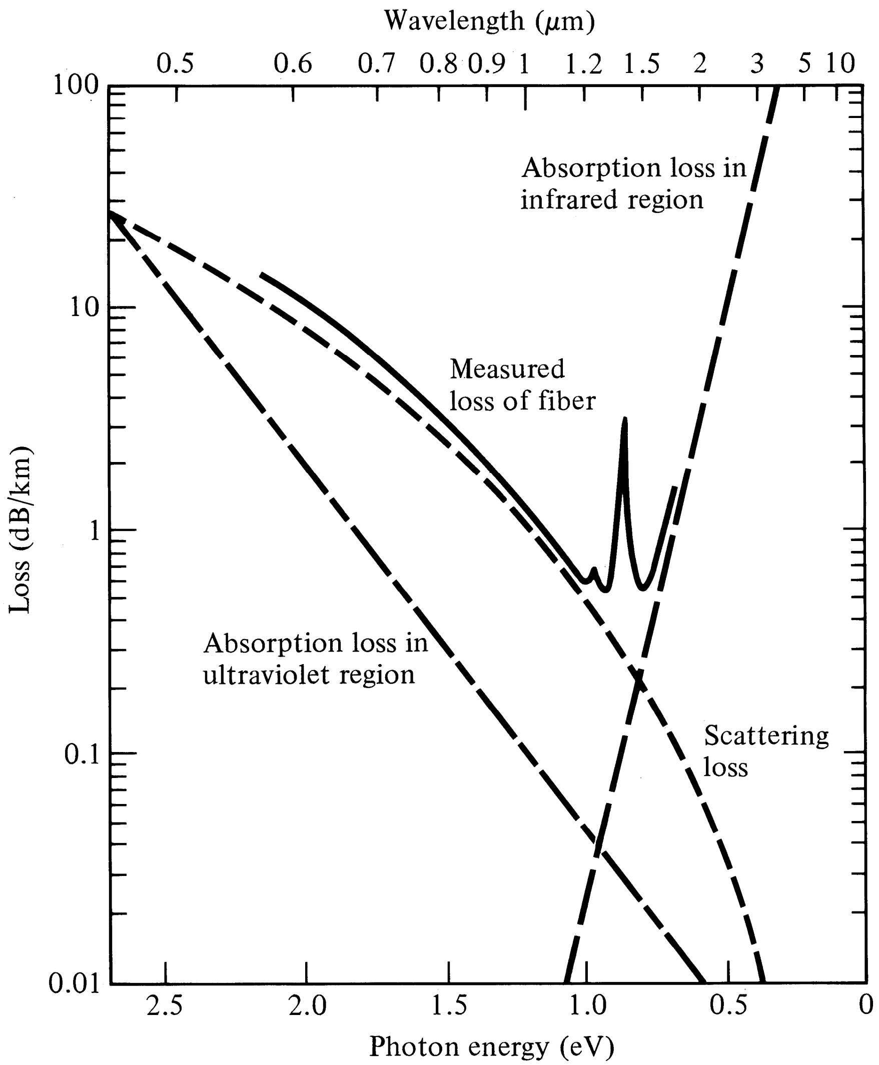
(b) Due to infra-red absorption : Glass intrinsically is a good infra-red absorber. As we increase the wavelength the infra-red loss increases rapidly.

SCATTERING LOSS

The scattering loss is due to the non-uniformity of the refractive index inside the core of the fiber. The refractive index of an optical fiber has fluctuation of the order of 10^{-4} over spatial scales much smaller than the optical wavelength. These fluctuations act as scattering centres for the light passing through the fiber. The process is, Rayleigh Scattering . A very tiny fraction of light gets scattered and therefore contributes to the loss.

The Rayleigh scattering is a very strong function of the wavelength. The scattering loss varies as λ^{-4}. This loss therefore rapidly reduces as the wavelength increases. For each doubling of the wavelength, the scattering loss reduces by a factor of 16. It is then clear that the scattering loss at 1550nm is about factor of 16 lower than that at 800nm.

The following Fig. shows the infrared, scattering and the total loss as a function of wavelength.

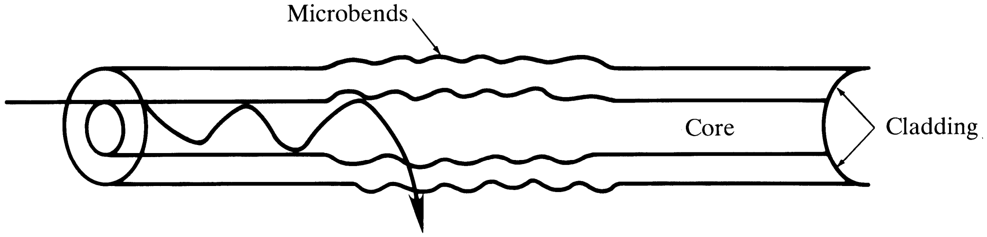


It is interesting to see that in the presence of various losses, there is a natural window in the optical spectrum where the loss is as low as 0.2-0.3dB/Km. This window is from 1200nm to 1600nm.

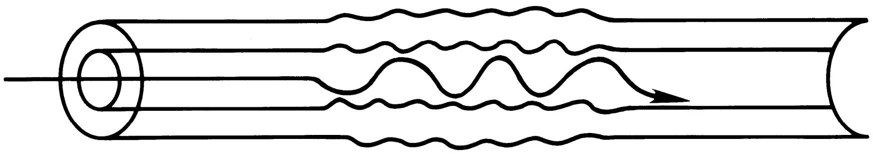
There is a local attenuation peak around 1400nm which is due to OH absorption. The low-loss window therefore is divided into sub-windows, one around 1300nm and other around 1550nm. In fact these are the windows which are the II and III generation windows of optical communication.

MICRO-BENDING LOSSES

While commissioning the optical fiber is subjected to micro-bending as shown in Fig.



Power loss from higher-order modes



Power coupling to higher-order modes

The analysis of micro-bends is a rather complex task. However, just for basic understanding of how the loss takes place due to micro-bending, we use following arguments.

In a fiber without micro-bends the light is guided by total internal reflection (ITR) at the core-cladding boundary. The rays which are guided inside the fiber has incident angle greater than the critical angle at the core-cladding interface. In the presence of micro-bends however, the direction of the local normal to the core-cladding interface deviates and therefore the rays may not have angle of incidence greater than the critical angle and consequently will be leaked out.

A part of the propagating optical energy therefore leaks out due to micro-bends.

Depending upon the roughness of the surface through which the fiber passes, the micro-bending loss varies.

Typically the micro-bends increase the fiber loss by 0.1-0.2 dB/Km.

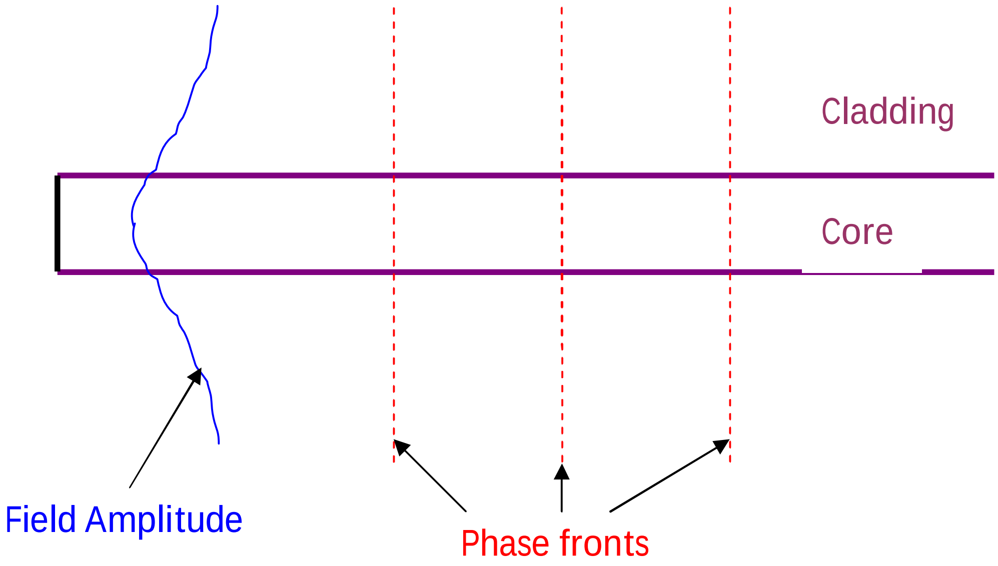
RADIATION OR BENDING LOSS

While laying the fiber the fiber may undergo a slow bend. In micro-bend the bending is on micron scale, whereas in a slow bend the bending is on cm scale. A typical example of a slow bend is a formation of optical fiber loop.

The loss mechanism due to bending loss can be well understood using modal propagation model.

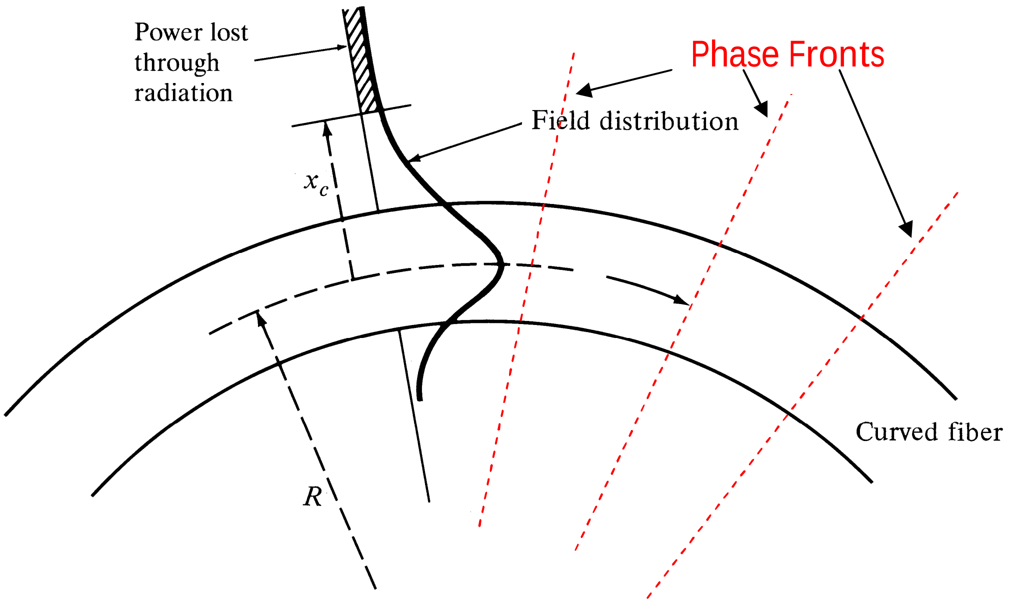
As we have seen, the light inside a fiber propagates in the form of modes. The modal fields decay inside the cladding away from the core cladding interface. Theoretically the field in the cladding is finite no matter how far away we are from the core-cladding interface. Now look at the amplitude and phase distribution for the fibers which are straight and which are bent over an circular arc as shown in Fig.

Phase Fronts in a Straight Fiber



It can be noted that for the straight the phase fronts are parallel and each point on the phase front travels with the same phase velocity.

Phase Fronts for a Bent Fiber



However, as soon the fiber is bent (no matter how gently) the phase fronts are no more parallel. The phase fronts move like a fan pivoted to the center of curvature of the bent fiber (see Fig.). Every point on the phase front consequently does not move with same velocity. The velocity increases as we move radially outwards the velocity of the phase front increases. Very quickly we reach to a distance x_c from the fiber where the velocity tries to become greater than the velocity of light in the cladding medium.

Since the velocity of energy can not be greater than velocity of light, the energy associated with the modal field beyond x_c gets detached from the mode and radiates away. This is called the bending or the radiation loss.

Following important things can be noted about the bending loss.

1. The radiation loss is present in every bent fiber no matter how gentle the bend is.
2. Radiation loss depends upon how much is the energy beyond x_c.
3. For a given modal field distribution if x_c reduces, the radiation loss increases. The x_c reduces as the radius of curvature of the bent fiber reduces, that is the fiber is sharply bent.
4. The number of modes therefore reduces in a multimode fiber in presence of bends.

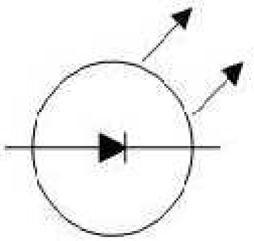
Light Emitting Diodes:-

INTRODUCTION

Over the past 25 years the light-emitting diode (LED) has grown from a laboratory curiosity to a broadly used light source for signaling applications . In 1992 LED production reached a level of approximately 25 billion chips , and $2 . 5 billion worth of LED-based components were shipped to original equipment manufacturers .

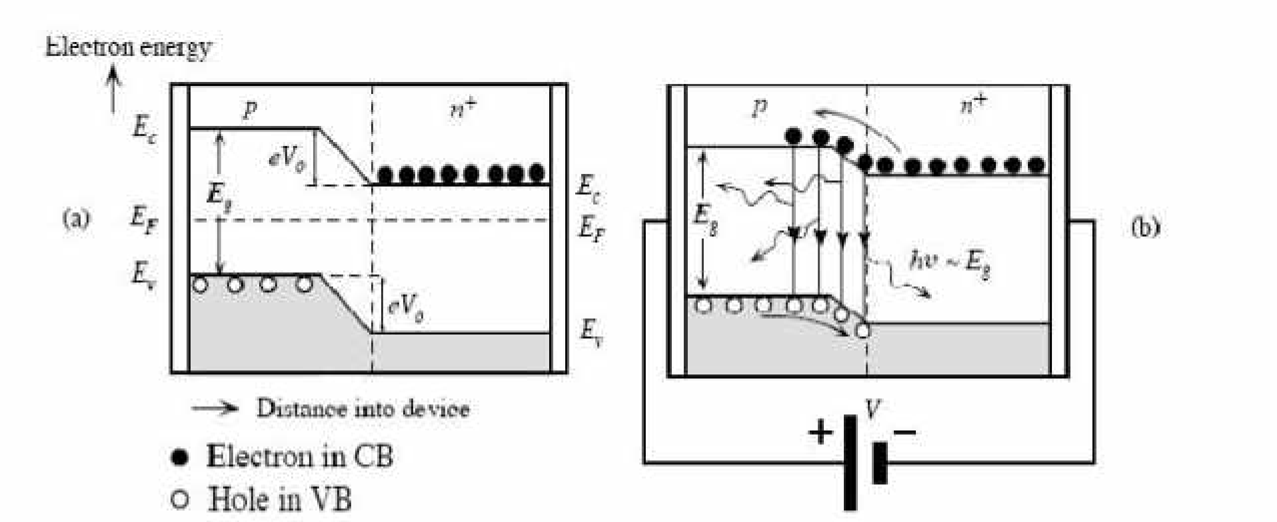
This article covers light-emitting diodes from the basic light-generation processes todescriptions of LED products . First , we will deal with light-generation mechanisms and light extraction . Four major types of device structures—from simple grown or dif fused homojunctions to complex double heterojunction devices are discussed next , followed by a description of the commercially important semiconductors used for LEDs , from the pioneering GaAsP system to the AlGaInP system that is currently revolutionizing LED technology . Then processes used to fabricate LED chips are explained—the growth of GaAs and GaP substrates ; the major techniques used for growing the epitixal material in which the light-generation processes occur ; and the steps required to create LED chips up to the point of assembly . Next the important topics of quality and reliability—in particular , chip degradation and package-related failure mechanisms—will be addressed . Finally , LED-based products , such as indicator lamps , numeric and alphanumeric displays , optocouplers , fiber-optic transmitters , and sensors , are described . This article covers the mainstream structures , materials , processes , and applications in use today . It does not cover certain advanced structures , such as quantum well or strained layer devices , The reader is also referred to for current information on edge-emitting LEDs , whose fabrication and use are similar to lasers .

Schematic:



Theory:

A Light emitting diode (LED) is essentially a pn junction diode. When carriers are injected across a forward-biased junction, it emits incoherent light. Most of the commercial LEDs are realized using a highly doped n and a p Junction.



(a) The energy band diagram of a pn^+ (heavily n-type doped) junction without any bias. Built-in potential V_o prevents electrons from diffusing from n^+ to p side. (b) The applied bias reduces V_o and thereby allows electrons to diffuse or be injected into the p-side. Recombination around the junction and within the diffusion length of the electrons in the p-side leads to photon emission.

Figure 1: p-n+ Junction under Unbiased and biased conditions

To understand the principle, let’s consider an unbiased pn+ junction (Figure1 shows the pn+ energy band diagram). The depletion region extends mainly into the p-side. There is a potential barrier from Ec on the n-side to the Ec on the p-side, called the built-in voltage, V0. This potential barrier prevents the excess free electrons on the n+ side from diffusing into the p side. When a Voltage V is applied across the junction, the built-in potential is reduced from V0 to V0 – V. This allows the electrons from the n+ side to get injected into the p-side. Since electrons are the minority carriers in the p-side, this process is called minority carrier injection. But the hole injection from the p side to n+ side is very less and so the current is primarily due to the flow of electrons into the p-side. These electrons injected into the p-side recombine with the holes. This recombination results in spontaneous emission of photons (light). This effect is called injection electroluminescence. These photons should be allowed to escape from the device without being reabsorbed.

The recombination can be classified into the following two kinds

- Direct recombination
- Indirect recombination

Direct Recombination:

In direct band gap materials, the minimum energy of the conduction band lies directly above the maximum energy of the valence band in momentum space energy . In this material, free electrons at the bottom of the conduction band can recombine directly with free holes at the top of the valence band, as the momentum of the two particles is the same. This transition from conduction band to valence band involves photon emission (takes care of the principle of energy conservation). This is known as direct recombination. Direct recombination occurs spontaneously. GaAs is an example of a direct band-gap material.

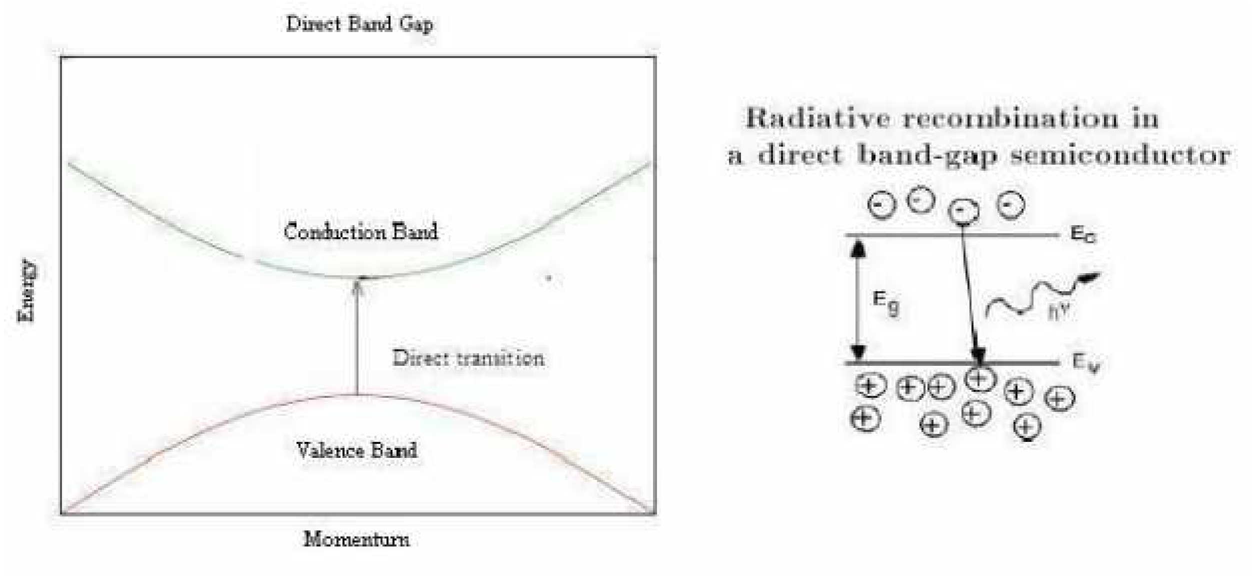


Figure 2: Direct Bandgap and Direct Recombination

Indirect Recombination:

In the indirect band gap materials, the minimum energy in the conduction band is shifted by a k-vector relative to the valence band. The k-vector difference represents a difference in momentum. Due to this difference in momentum, the probability of direct electronhole recombination is less.

In these materials, additional dopants(impurities) are added which form very shallow donor states. These donor states capture the free electrons locally; provides the necessary momentum shift for recombination. These donor states serve as the recombination centers. This is called Indirect (non-radiative) Recombination.

Nitrogen serves as a recombination center in GaAsP. In this case it creates a donor state, when SiC is doped with Al, it recombination takes place through an acceptor level. when SiC is doped with Al, it recombination takes place through an acceptor level.

The indirect recombination should satisfy both conservation energy, and momentum.

Thus besides a photon emission, phonon emission or absorption has to take place.

GaP is an example of an indirect band-gap material.

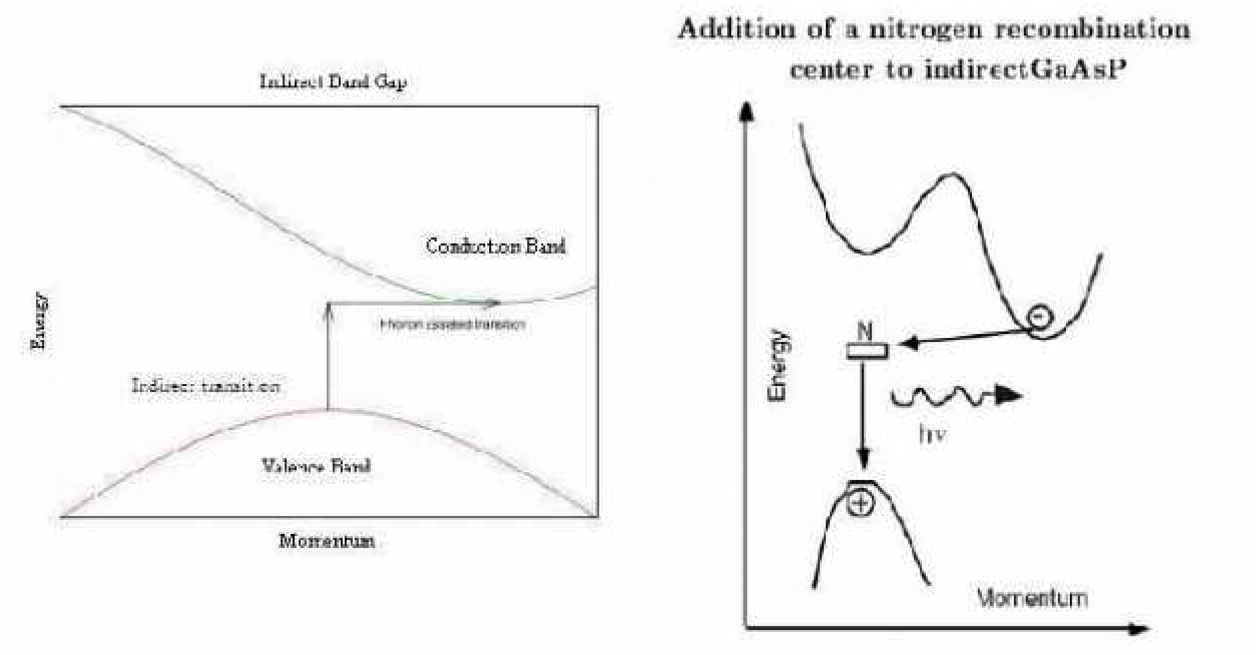


Figure 3: Indirect Bandgap and NonRadiative recombination

The wavelength of the light emitted, and hence the color, depends on the band gap energy of the materials forming the p-n junction.

The emitted photon energy is approximately equal to the band gap energy of the semiconductor. The following equation relates the wavelength and the energy band gap.

$$h\nu = Eg$$
$$hc/\lambda = Eg$$
$$\lambda = hc/ Eg$$

Where h is Plank’s constant, c is the speed of the light and Eg is the energy band gap Thus, a semiconductor with a 2 eV band-gap emits light at about 620 nm, in the red. A 3 eV band-gap material would emit at 414 nm, in the violet.

LED Materials:

An important class of commercial LEDs that cover the visible spectrum are the III-V. ternary alloys based on alloying GaAs and GaP which are denoted by GaAs1-

yPy. InGaAlP is an example of a quarternary (four element) III-V alloy with a direct band gap.

The LEDs realized using two differently doped semiconductors that are the same material is called a homojunction. When they are realized using different bandgap materials they are called a heterostructure device. A heterostructure LED is brighter than a homoJunction LED.

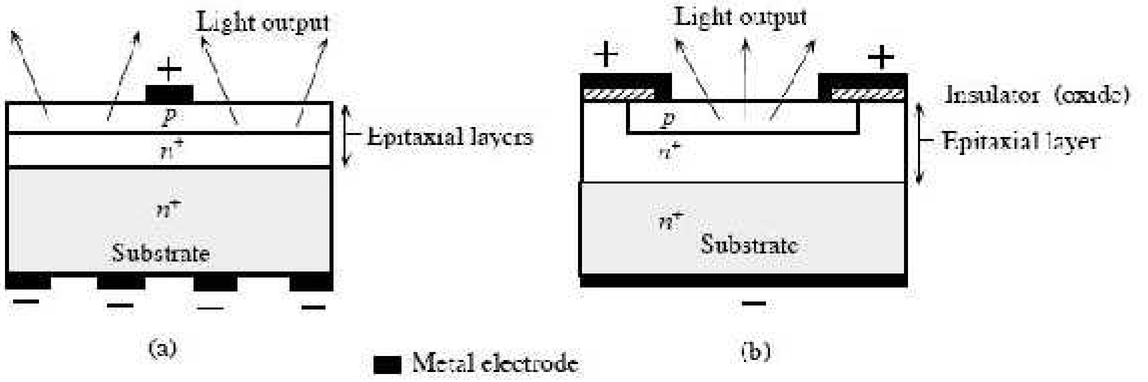
LED Structure:

The LED structure plays a crucial role in emitting light from the LED surface. The LEDs are structured to ensure most of the recombinations takes place on the surface by the following two ways.

- By increasing the doping concentration of the substrate, so that additional free minority charge carriers electrons move to the top, recombine and emit light at the surface.

• By increasing the diffusion length $L = \sqrt{D\tau}$, where D is the diffusion coefficient and τ is the carrier life time. But when increased beyond a critical length there is a chance of re-absorption of the photons into the device.

The LED has to be structured so that the photons generated from the device are emitted without being reabsorbed. One solution is to make the p layer on the top thin, enough to create a depletion layer. Following picture shows the layered structure. There are different ways to structure the dome for efficient emitting.



A schematic illustration of typical planar surface emitting LED devices. (a) *p*-layer grown epitaxially on an n^+ substrate. (b) First n^+ is epitaxially grown and then *p* region is formed by dopant diffusion into the epitaxial layer.

LED structure

LEDs are usually built on an n-type substrate, with an electrode attached to the p-typelayer deposited on its surface. P-type substrates, while less common, occur as well. Manycommercial LEDs, especially GaN/InGaN, also use sapphire substrate.

LED efficiency:

A very important metric of an LED is the external quantum efficiency ηext. It quantifies the efficeincy of the conversion of electrical energy into emitted optical energy. It is defined as the light output divided by the electrical input power. It is also defined as the product of Internal radiative efficiency and Extraction efficiency.

ηext = Pout(optical) / IV

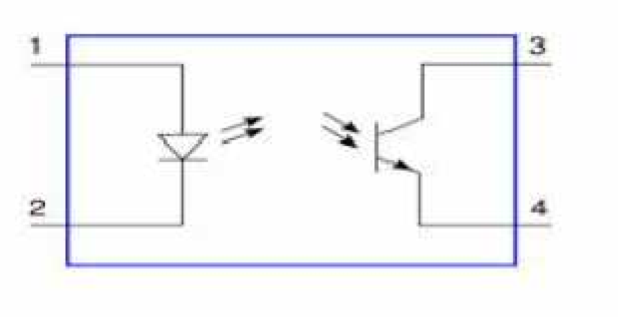
For indirect bandgap semiconductors ηext is generally less than 1%, where as for a direct band gap material it could be substantial.

ηint = rate of radiation recombination/ Total recombination

The internal efficiency is a function of the quality of the material and the structure and composition of the layer.

Applications: LED have a lot of applications. Following are few examples.

- Devices, medical applications, clothing, toys
- Remote Controls (TVs, VCRs)
- Lighting
- Indicators and signs
- Optoisolators and optocouplers
- Swimming pool lighting



Optocoupler schematic showing LED and phototransistor

Advantages of using LEDs:

• LEDs produce more light per watt than incandescent bulbs; this is useful inbattery powered or energy-saving devices.

• LEDs can emit light of an intended color without the use of color filters that traditional lighting methods require. This is more efficient and can lower initial costs.

• The solid package of the LED can be designed to focus its light. Incandescent and fluorescent sources often require an external reflector to collect light and direct itin a usable manner.

• When used in applications where dimming is required, LEDs do not change their color tint as the current passing through them is lowered, unlike incandescent lamps, which turn yellow.

• LEDs are ideal for use in applications that are subject to frequent on-off cycling, unlike fluorescent lamps that burn out more quickly when cycled frequently, or High Intensity Discharge (HID) lamps that require a long time before restarting.

• LEDs, being solid state components, are difficult to damage with external shock.Fluorescent and incandescent bulbs are easily broken if dropped on the ground.

• LEDs can have a relatively long useful life. A Philips LUXEON k2 LED has a life time of about 50,000 hours, whereas Fluorescent tubes typically are rated at about 30,000 hours, and incandescent light bulbs at 1,000–2,000 hours.

• LEDs mostly fail by dimming over time, rather than the abrupt burn-out of incandescent bulbs.

• LEDs light up very quickly. A typical red indicator LED will achieve full brightness in microseconds; Philips Lumileds technical datasheet DS23 for the Luxeon Star states "less than 100ns." LEDs used in communications devices can have even faster response times.

• LEDs can be very small and are easily populated onto printed circuit boards.

• LEDs do not contain mercury, unlike compact fluorescent lamps.

Disadvantages:

• LEDs are currently more expensive, price per lumen, on an initial capital cost basis, than more conventional lighting technologies. The additional expense partially stems from the relatively low lumen output and the drive circuitry and power supplies needed. However, when considering the total cost of ownership (including energy and maintenance costs), LEDs far surpass incandescent or halogen sources and begin to threaten the future existence of compact fluorescent lamps.

• LED performance largely depends on the ambient temperature of the operating environment. Over-driving the LED in high ambient temperatures may result in overheating of the LED package, eventually leading to device failure. Adequate heat-sinking is required to maintain long life .

• LEDs must be supplied with the correct current. This can involve series resistors or current-regulated power supplies.

• LEDs do not approximate a "point source" of light, so they cannot be used in applications needing a highly collimated beam. LEDs are not capable of providing divergence below a few degrees. This is contrasted with commercial ruby lasers with divergences of 0.2 degrees or less. However this can be corrected by using lenses and other optical devices.

Laser diodes:-

Laser diodes (also called .injection lasers.) are in effect anspecialised form of LED. Just like a LED, they.re a form of P-N junction diode with a thin depletion layer where electrons and holes collide to create light photons, when the diode is forward biased.

The difference is that in this case the .active. part of the depletion layer (i.e., where most of the current flows) is made quite narrow, to concentrate the carriers. The endsof this narrow active region are also highly polished, or coated with multiple very thin reflective layers to act as mirrors, so it forms a resonant optical cavity.

The forward current level is also increased, to the point where the current density reaches a critical level where carrier population inversion. occurs. This means there are more holes than electrons in the conduction band, and more electrons than holes in the valence band . or in other words, a very large excess population of electrons and holes which can potentially combine to release photons. And when this happens, the creation of new photons can be triggered not just by random collisions of electrons and holes, but lso by the influence of passing photons. Passing photons are then able to stimulate the production of more photons, without themselves being absorbed. So laser action is able to occur: Light Amplification by Stimulated Emission of Radiation. And the important thing to realise is that the photons that are triggered by other passing photons have the same wavelength, and arealso in phase with them. In other words, they end up .in sync. and forming continuous-wave coherent radiation.

Because of the resonant cavity, photons are thus able to travel back and forth from one end of the active region to the other, triggering the production of more and more photons in sync with themselves. So quite a lot of coherent light energy is generated.

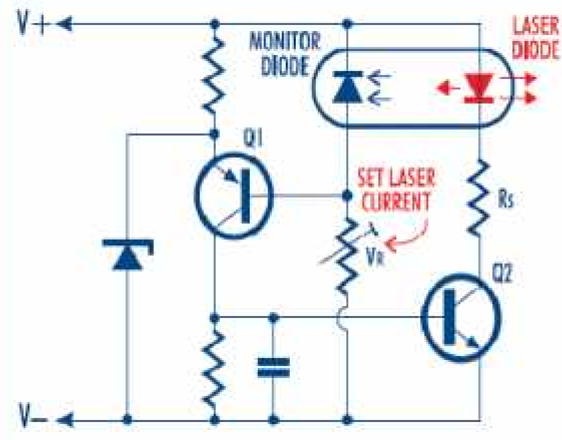


Fig.7: How the monitor diode is used to control the laser operating current and optical output, in a simple feedback circuit.

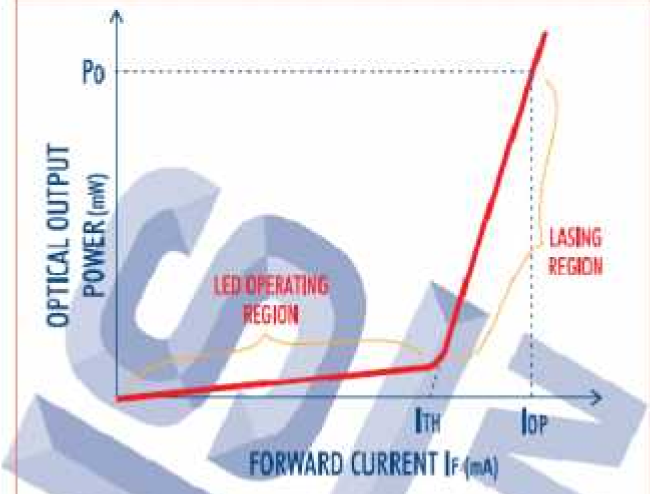


Fig.8: Lasing occurs only at current levels above a critical value, ITH.

And as the ends of the cavity are not totally reflective (typically about 90-95%), some of this coherent light can leave the laser chip . to form its output beam.

Because a laser.s light output is coherent, it is very low in noise and also more suitable for use as a .carrier. for data communications. The bandwidth also tends to be narrower and better defined than LEDs, making them more suitable for optical systems where light beams need to be separated or manipulated on the basis of wavelength.

The very compact size of laser diodes makes them very suitable for use in equipment like CD, DVD and MiniDisc players and recorders. As their light is reasonably well collimated (although not as well as gas lasers) and easily focussed, they.re also used in optical levels, compact handheld laser pointers, barcode scanners etc. There are two main forms of laser diode: the horizontal type, which emits light from the polished ends of the chip, and the vertical or .surface emitting. type. They both operate in the way just described, differing mainly in terms of the way the active light generating region and resonant cavity are formed inside the chip. Because laser diodes have to be operated at such a high current density, and have a very low forward resistance when lasing action occurs, they are at risk of destroying themselves due to thermal runaway. Their operating light density can also rise to a level where the end mirrors can begin melting. As a result their electrical operation must be much more carefully controlled than a LED. This means that not only must a laser diode.s current be regulated by a .constant current. circuit rather than a simple series resistor, but optical negative feedback must generally be used as well . to ensure that the optical output is held to a constant safe level.

To make this optical feedback easier, most laser diodes have a silicon PIN photodiode built right into the package, arranged so that it automatically receives a fixed proportion of the laser.s output. The output of this monitor diode can then be used to control the current fed through the laser by the constant current circuit, for stable and reliable operation. Fig.6 shows a typical .horizontal. type laser chip mounted in its package, with the monitor photodiode mounted on the base flange below it so the diode receives the light output from the .rear. of the laser chip.

Fig.7 (page 3) shows a simple current regulator circuit used to operate a small laser diode, and you can see how the monitor photodiode is connected. The monitor diode is shunting the base forward bias for transistor Q1, which has

its emitter voltage fixed by the zener diode. So as the laseroutput rises, the monitor diode current increases, reducing the conduction of Q1 and hence that of transistor Q2, which controls the laser current. As a result, the laser current is automatically stabilised to a level set by adjustable resistor VR.

Laser diode parameters

Perhaps the key parameter for a laser diode is the threshold current (ITH), which is the forward current level where lasing actually begins to occur. Below that current level the device delivers some light output, but it operates only as a LED rather than a laser. So the light it does produce in this mode is incoherent. Another important parameter is the rated light output (Po), which is the highest recommended light output level (in milliwatts) for reliable continuous operation. Not surprisingly there.s an operating current level (IOP) which corresponds to this rated light output (Fig.8). There.s also the corresponding current output from the feedback photodiode, known as the monitor current level (Im). Other parameters usually given for a laser diode are its peak lasing wavelength, using given in nanometres (nm); and its beam divergence angles (defined as the angle away from the beam axis before the light intensity drops to 50%), in the X and Y directions (parallel to, and normal to the chip plane).

Laser safety

Although most of the laser diodes used in electronic equipment have quite low optical output levels . typically less than 5mW (milliwatts) . their output is generally concentrated in a relatively narrow beam. This means that it is still capable of causing damage to a human or animal eye, and particularly to its light-sensitive retina.

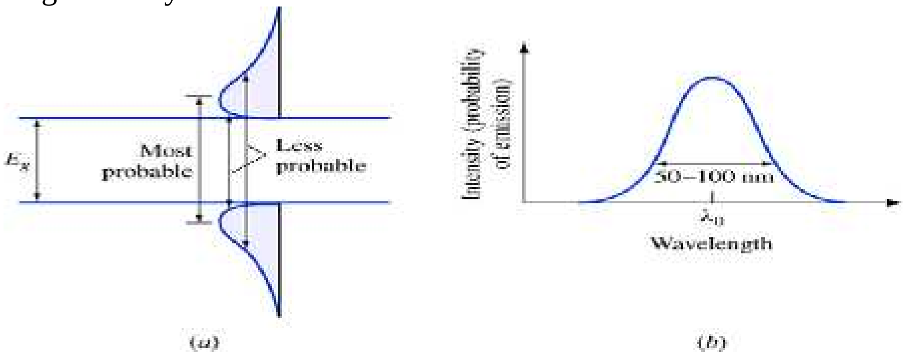
Infra-red (IR) lasers are especially capable of causing eye damage, because their light is not visible. This prevents the eye.s usual protective reflex mechanisms (iris contraction, eyelid closure) from operating. So always take special care when using devices like laser pointers, and especially when working on equipment which includes IR lasers, to make sure that the laser beam cannot enter either your own, or anyone else.s eyes. If you need to observe the output from a laser, either use protective filter goggles or use an IR-sensitive CCD type video camera. Remember that eye damage is often irreversible, especially when it.s damage to the retina.

•Light Emitting Diode

•Light is mostly monochromatic (narrow energy spread comparable to the distribution of electrons/hole populations in the band edges)

•Light is from spontaneous emission (random events in time and thus phase).

•Light diverges significantly



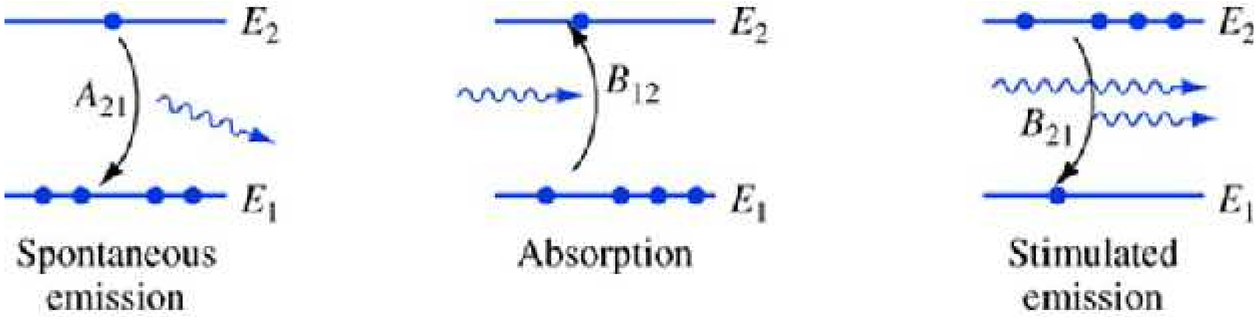
LASER

•Light is essentially single wavelength (highly monochromatic)

•Light is from "stimulated emission" (timed to be in phase with other photons

•Light has significantly lower divergence (Semiconductor versions have more than gas lasers though).

Spontaneous Light Emission



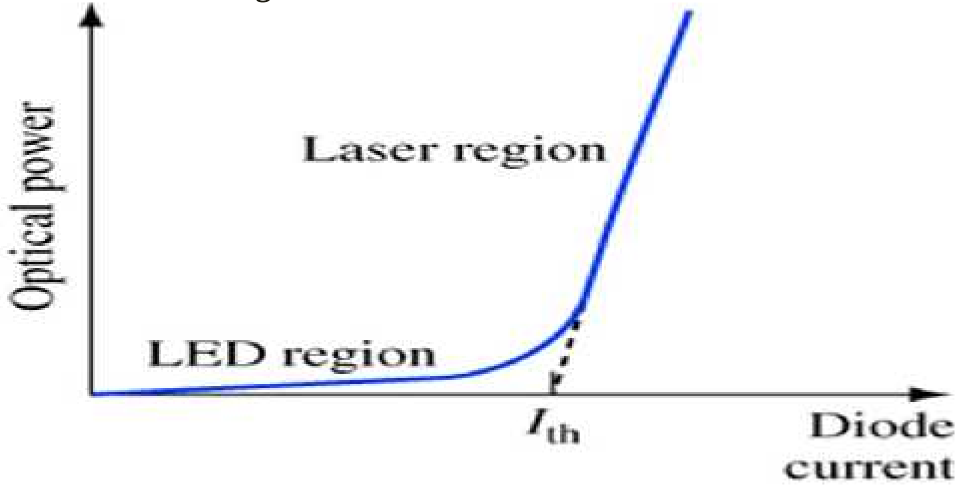
• We can add to our understanding of absorption and spontaneous radiation due to random recombination another form of radiation – Stimulated emission.

• Stimulated emission can occur when we have a "population inversion", i.e. when we have injected so many minority carriers that in some regions there are more "excited carriers" (electrons) than "ground state" carriers (holes).

• Given an incident photon of the band gap energy, a second photon will be "stimulated" by the first photon resulting in two photons with the same energy (wavelength) and phase.

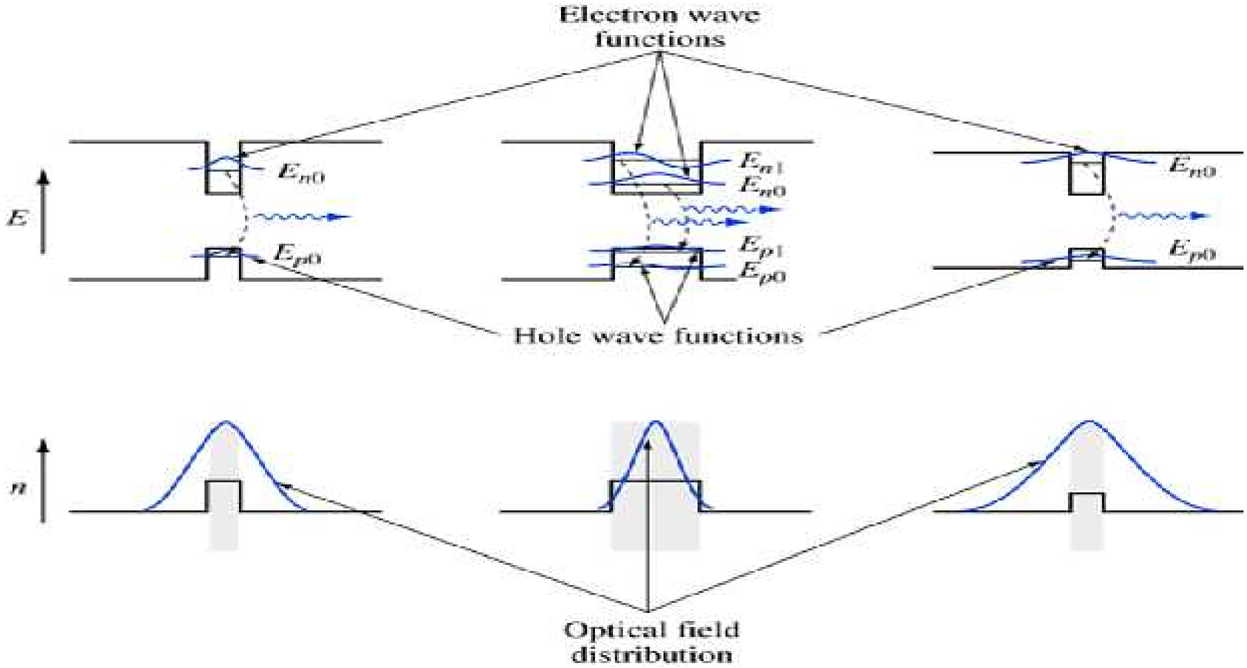
- **This phase coherence results in minimal divergence of the optical beam resulting in a directed light source.**

Spontaneous vs Stimulated Light Emission:



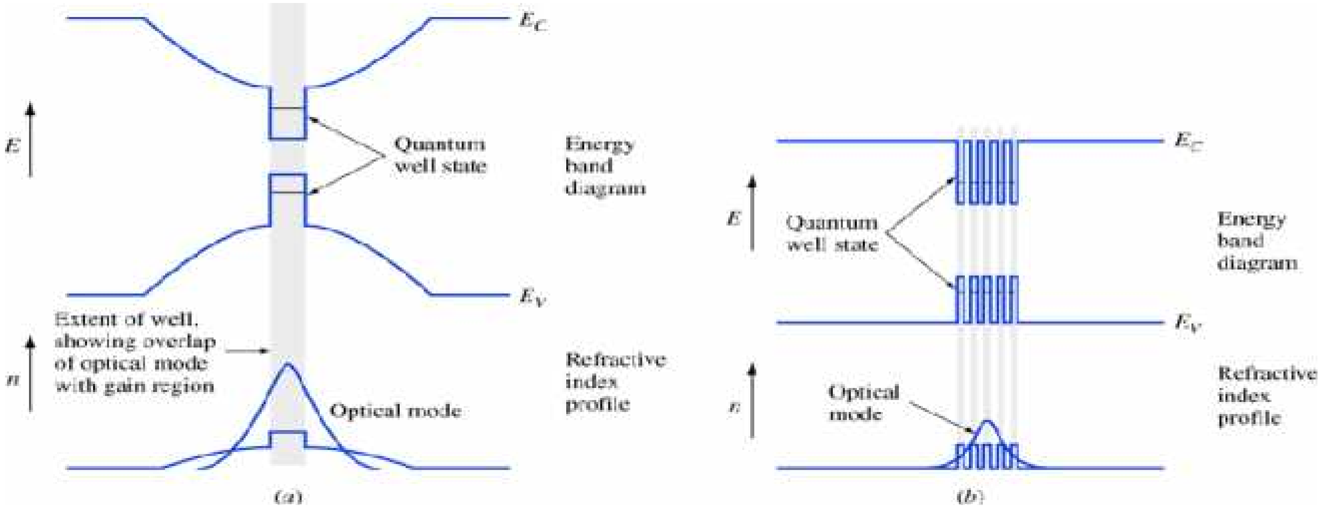
The power-current curve of a laser diode. Below threshold, the diode is an LED. Above threshold, the population is inverted and the light output increases rapidly.

LASER Wavelength Design:



Adjusting the depth and width of quantum wells to select the wavelength of emission is one form of band-gap engineering. The shaded areas indicate the width of the well to illustrate the degree of confinement of the mode.

Advanced LASER Wavelength Design:



(a) A GRINSCH structure helps funnel the carriers into the wells to improve the probability of recombination. Additionally, the graded refractive index helps confine the optical mode in the nearwell region. Requires very precise control over layers due to grading. Almost always implemented via MBE

(b) A multiple quantum well structure has improves carrier capture.

Sometimes the two are combined to give a “digitally graded” device where only two compositions are used but the well thicknesses are varied to implement an effective “index grade”

Photodetectors:-

These are **Opto-electric devices** i.e. to convert the optical signal back into electrical impulses.

The light detectors are commonly made up of semiconductor material.

When the light strikes the light detector a current is produced in the external circuit proportional to the intensity of the incident light.

Optical signal generally is **weakened** and distorted when it emerges from the end of the fiber, **the photodetector must meet following strict performance requirements.**

A **high sensitivity** to the emission wavelength range of the received light signal.

A **minimum** addition of **noise** to the signal.

A **fast response** speed to handle the desired data rate.

Be **insensitive** to **temperature** variations.

Be **compatible** with the physical dimensions of the **fiber.**

Have a **Reasonable cost** compared to other system components.

Have a long **operating lifetime.**

Some important parameters while discussing photodetectors:

Quantum Efficiency

It is the ratio of primary electron-hole pairs created by incident photon to the photon incident on the diode material.

Detector Responsivity

This is the ratio of output current to input optical power. Hence this is the efficiency of the device.

Spectral Response Range

This is the range of wavelengths over which the device will operate.

Types of Light Detectors

_ PIN Photodiode

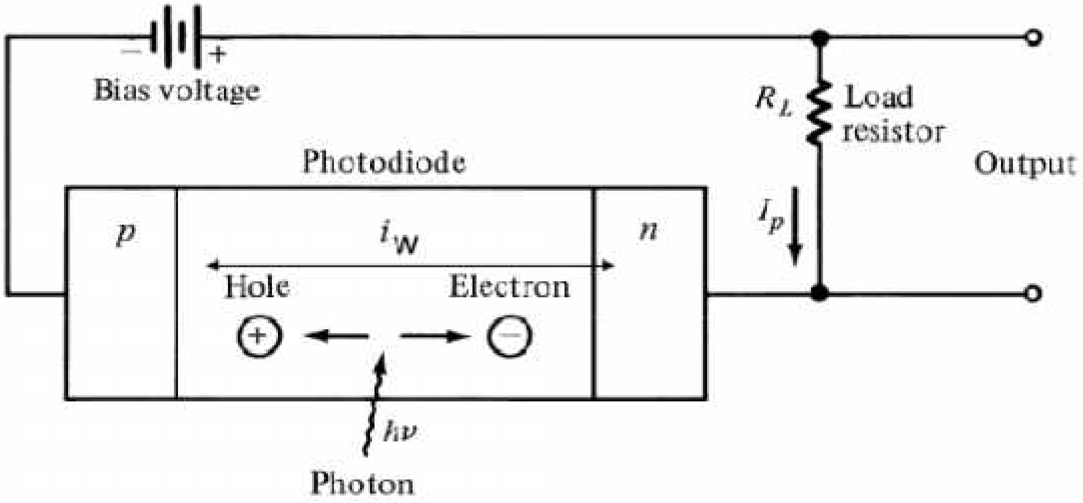
_ Avalanche Photodiode

The Pin Photodetector:-

The **device structure** consists of **p and n** semiconductor regions separated by a very **lightly n-doped intrinsic (i) region.**

In **normal operation** a reverse-bias voltage is applied across the device so that **no free electrons or holes** exist in the **intrinsic region**.

Incident photon having energy **greater than or equal** to the **bandgap energy** of the semiconductor material, **give up itsenergy** and **excite an electron** from the valence band to the conduction band.



The high electric field present in the depletion region causes photogenerated carriers to separate and be collected across the reverse – biased junction. This gives rise to a current flow in an external circuit, known as **photocurrent**.

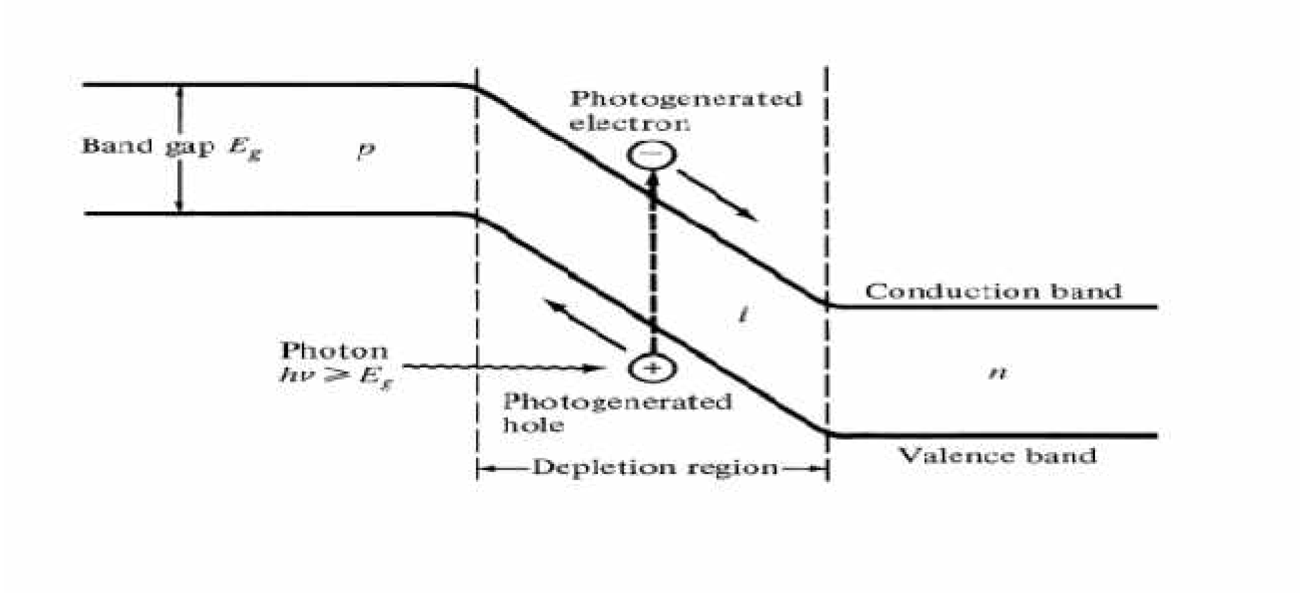
Photocarriers:

Incident photon, generates free (mobile) **electron-hole pairs in the intrinsic region**. These charge carriers are known as **photocarriers,** since they are generated by a photon.

Photocurrent:

The electric field across the device causes the **photocarriers to be swept out of the intrinsic region**, thereby giving rise to a **current flow in an external circuit**. This current flow is known as the **photocurrent.**

Energy-Band diagram for a *pin* photodiode:



An incident photon is able to boost an electron to the conduction band only if it has an energy that is greater than or equal to the bandgap energy

Thus, a particular semiconductor material can be used only over a limited wavelength range.

$$\lambda_c = \frac{hc}{E_g}$$

As the charge carriers flow through the material some of them recombine and disappear.

The charge carriers move a distance Ln or Lp for electrons and holes before recombining. This distance is known as diffusion length

The time it take to recombine is its life time _n or _p respectively.

$$L_n = (D_n \tau_n)^{1/2} \quad \text{and} \quad L_p = (D_p \tau_p)^{1/2}$$

Where Dn and Dp are the diffusion coefficients for electrons and holes respectively.

Photocurrent:-

As a photon flux penetrates through the semiconductor, it will be absorbed.

If *P*in is the optical power falling on the photo detector at *x*=0 and *P*(*x*) is the power level at a distance *x* into the material then the incremental change be given as

$$dP(x) = -\alpha_s(\lambda)P(x)dx$$

where _s(_) is the photon absorption coefficient at a wavelength _. So that

$$P(x) = P_{in}\exp(-\alpha_s x)$$

Optical power absorbed, *P*(*x*), in the depletion region can be written in terms of incident optical power, *Pin* :

$$P(x) = P_{in}(1 - e^{-\alpha_s(\lambda)x})$$

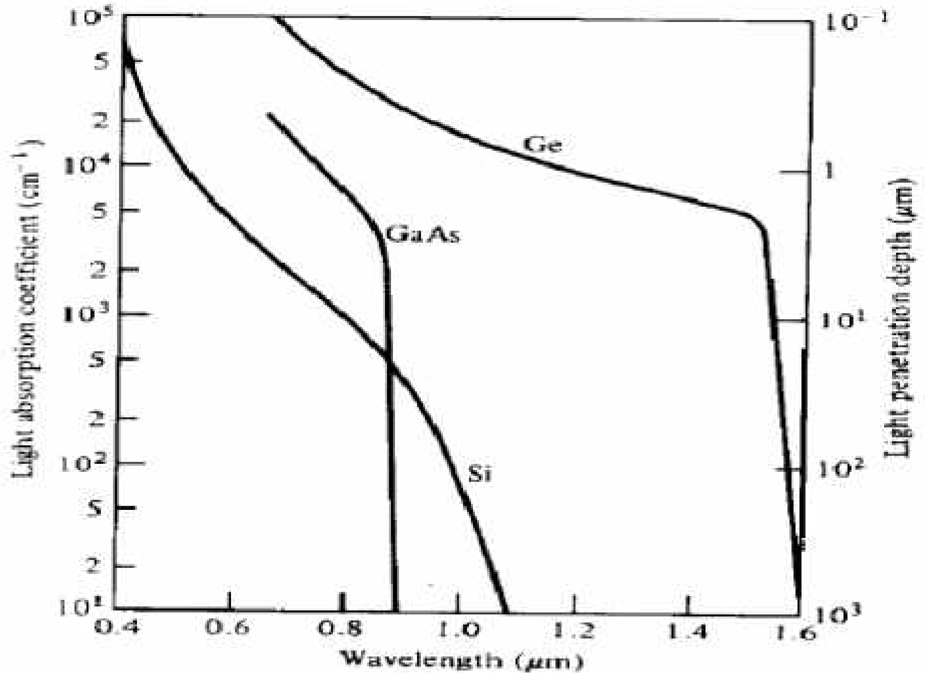
Absorption coefficient a*s* (l) strongly depends on wavelength. The upper wavelength cutoff for any semiconductor can be

$$\lambda_c\,(\mu\mathrm{m}) = \frac{1.24}{E_g\,(\mathrm{eV})}$$

Taking entrance face reflectivity into consideration, the absorbed power in the width of depletion region, *w*, becomes:

$$(1-R_f)P(w) = P_{in}(1-e^{-\alpha_s(\lambda)w})(1-R_f)$$

Optical Absorption Coefficient



The primary photocurrent resulting from absorption is:

$$\boxed{I_p = \frac{q}{h\nu} P_{in}(1-e^{-\alpha_s(\lambda)w})(1-R_f)}$$

Quantum Efficiency:

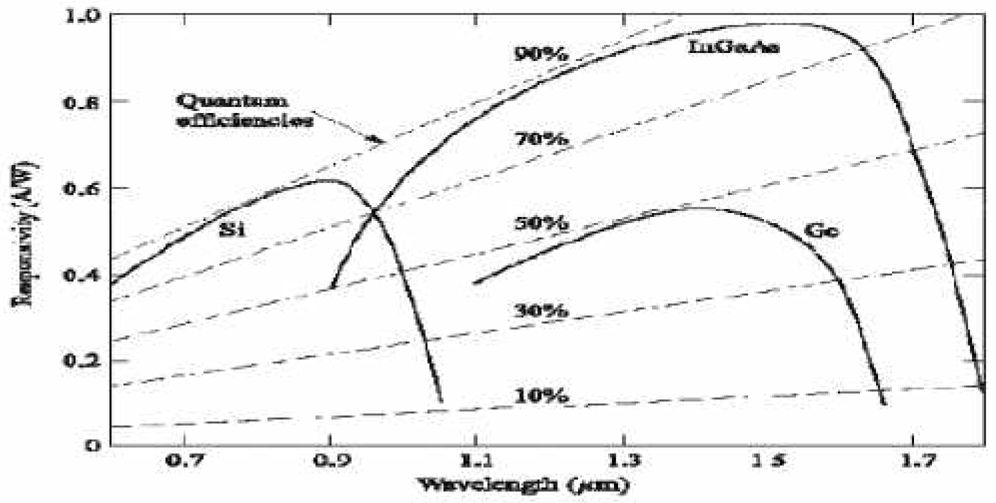
$$\eta = \frac{\text{\# of electron - hole photogener ated pairs}}{\text{\# of incident photons}}$$

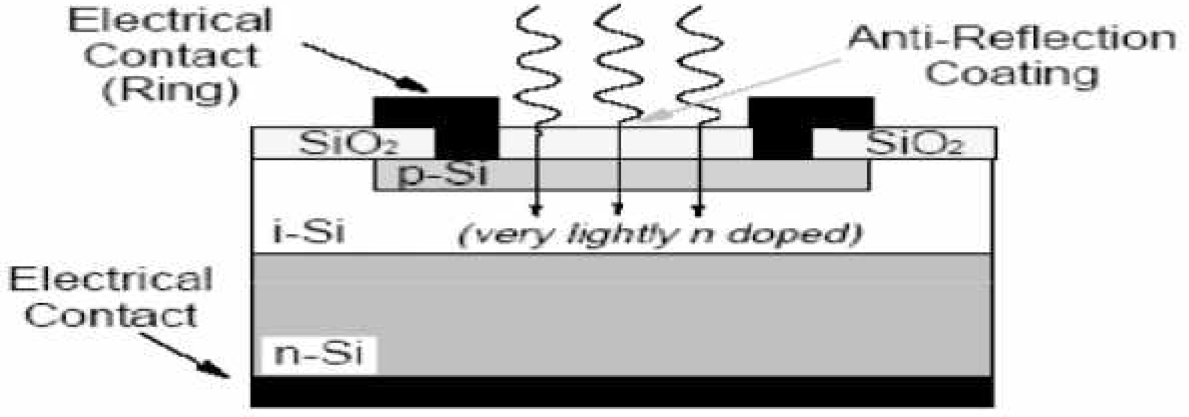
$$\boxed{\eta = \frac{I_p / q}{P_{in} / h\nu}}$$

Responsivity:

$$\boxed{\Re - \frac{I_p}{P_{in}} - \frac{\eta q}{h\nu} \quad [\text{A/W}]}$$

Responsivity vs. wavelength

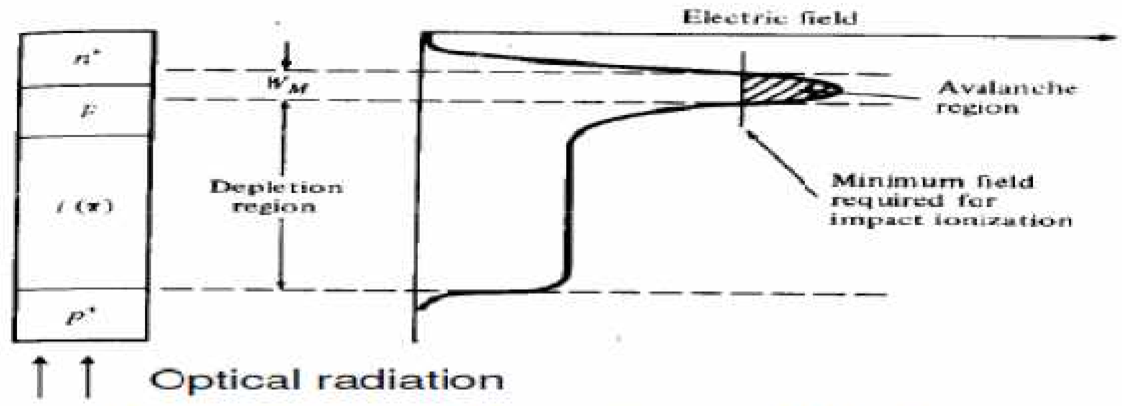




Typical Silicon P-I-N Diode Schematic

Avalanche Photodiode (APD):

APDs internally multiply the primary photocurrent before it enters to following circuitry. In order to carrier multiplication take place, the photogenerated carriers must traverse along a high field region. In this region, photogenerated electrons and holes gain enough energy toionize bound electrons in VB upon colliding with them. This multiplication is known as impact ionization. The newly created carriers in the presence of high electric field result in more ionization called avalanche effect.



Reach-Through APD structure (RAPD) showing the electric fields in depletion region and multiplication region.

Responsivity of APD:

The multiplication factor (current gain) *M* for all carriers generated in the photodiode is defined as:

$$M = \frac{I_M}{I_P}$$

where *IM* is the average value of the total multiplied output current & *Ip* is the primary photocurrent.

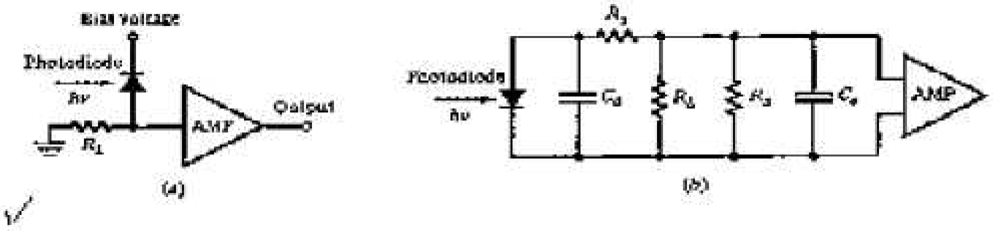
The responsivity of APD can be calculated by considering the current gain as:

$$\boxed{\Re_{\text{APD}} = \frac{\eta q}{h\nu} M = \Re_0 M}$$

Photodetector Noise & S/N:-

Detection of weak optical signal requires that the photodetector and its following amplification circuitry be optimized for a desired signal-to-noise ratio.

It is the noise current which determines the minimum optical power level that can be detected. This minimum detectable optical power defines the **sensitivity** of photodetector. That is the optical power that generates a photocurrent with the amplitude equal to that of the total noise current (*S/N=1*)



$$\frac{S}{N}=\frac{\text{signal power from photocurre nt}}{\text{photodetec tor noise power + amplifier noise power}}$$

$$\boxed{\frac{S}{N}=\frac{\left\langle i_P{}^2\right\rangle M^2}{2q(I_P+I_D)BM^2F(M)+2qI_LB+4k_BTB/R_L}}$$

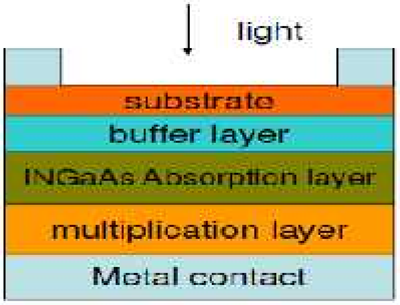
Since the noise figure $F(M)$ increases with M, there always exists an optimum value of M that maximizes the S/N. For sinusoidally modulated signal with m=1 and

$$F(M)\approx M^x$$

$$M_{\text{opt}}^{x+2}=\frac{2qI_L+4k_BT/R_L}{xq(I_P+I_D)}$$

Structures for InGaAs APDs:-

Separate-absorption-and multiplication (SAM) APD



InGaAs APD superlattice structure (The multiplication region is composed of several layers of InAlGaAs quantum wells separated by InAlAs barrier layers.

Fiber Optic System Design:-

There are many factors that must be considered to ensure that enough light reaches the receiver. Without the right amount of light, the entire system will not operate properly.

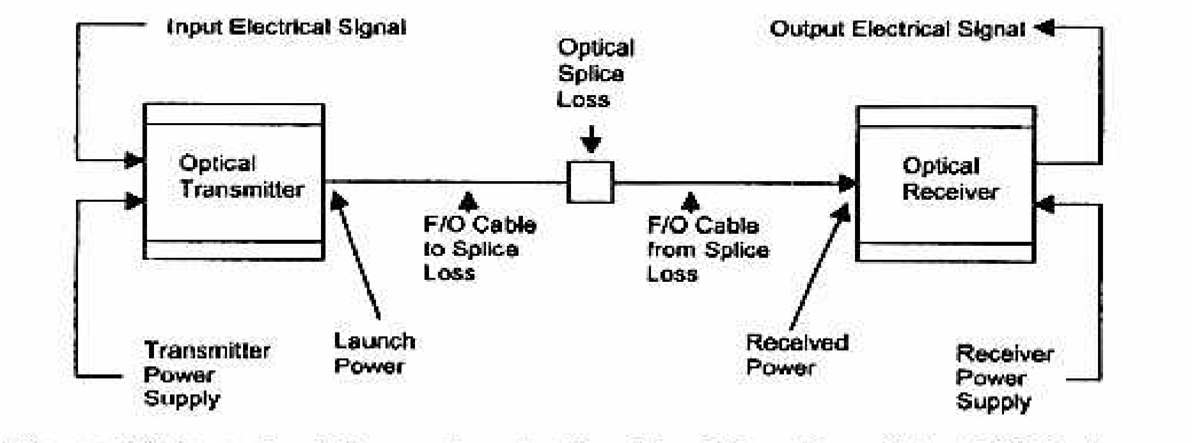


Figure 12, Important Parameters to Consider When Specifying F/O Systems

Fiber Optic System Design- Step-by-Step:-

Select the most appropriate optical transmitter and receiver combination based upon the signal to be transmitted

Determine the operating power available (AC, DC, etc.).

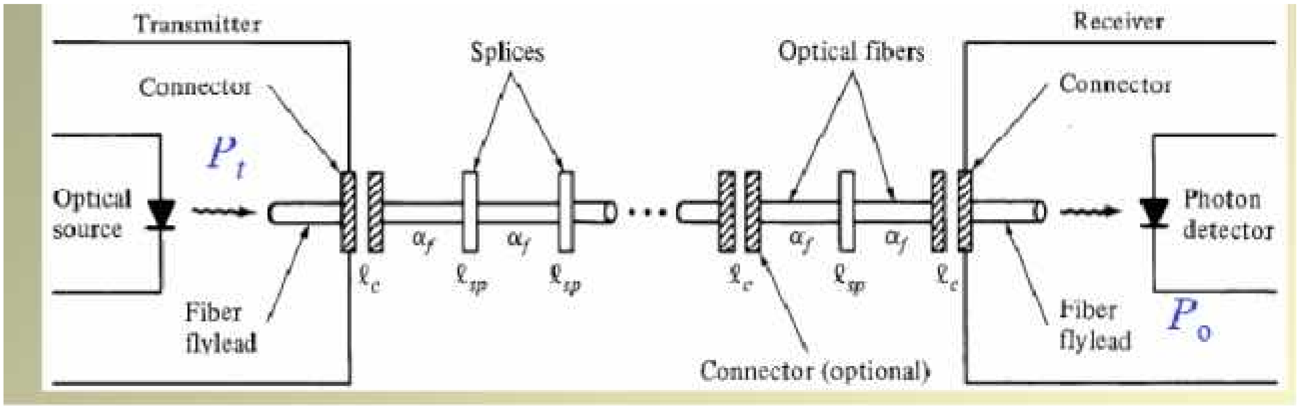
Determine the special modifications (if any) necessary (Impedances, bandwidths, connectors, fiber size, etc.).

Carry out system link power budget.

Carry out system rise time budget (I.e. bandwidth budget).

If it is discovered that the fiber bandwidth is inadequate for transmitting the required signal over the necessary distance, then either select a different transmitter/receiver (wavelength) combination, or consider the use of a lower loss premium fiber

Link Power Budget:-



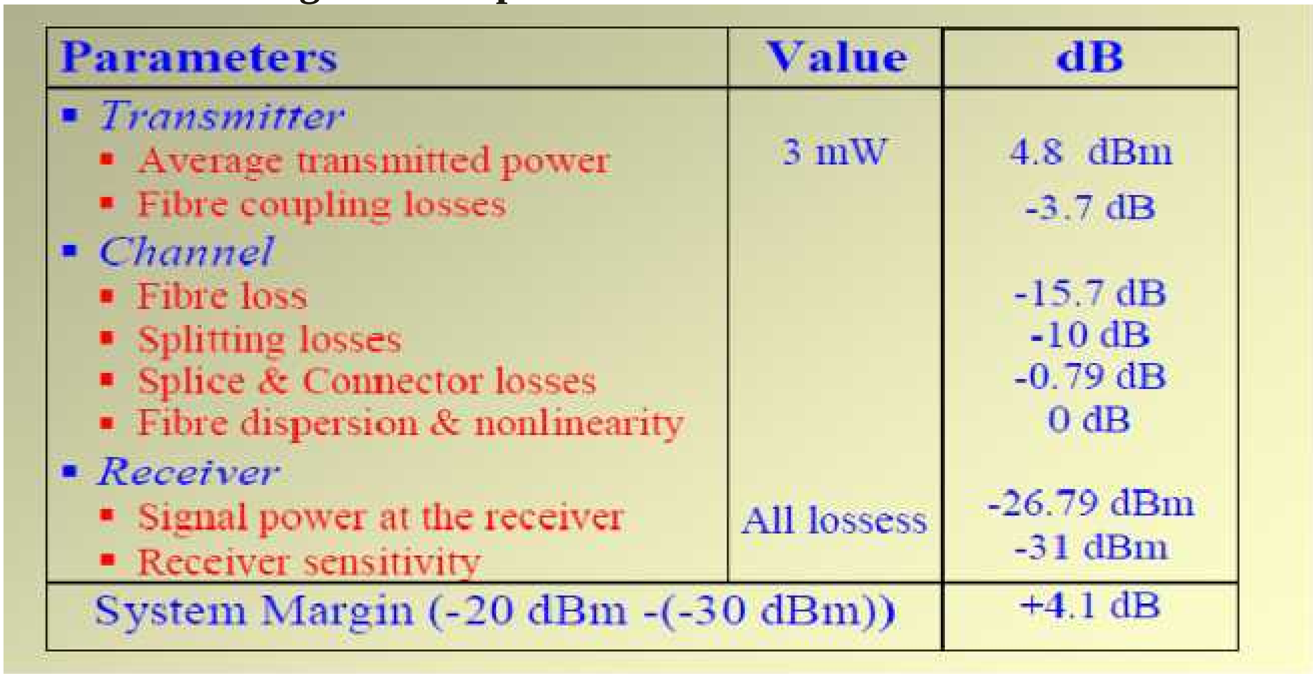
Total loss $LT = \alpha f\ L + lc + lsp$

$$Pt - Po = LT + SM$$

Po = Receiver sensitivity (i.e. minimum power requirement)

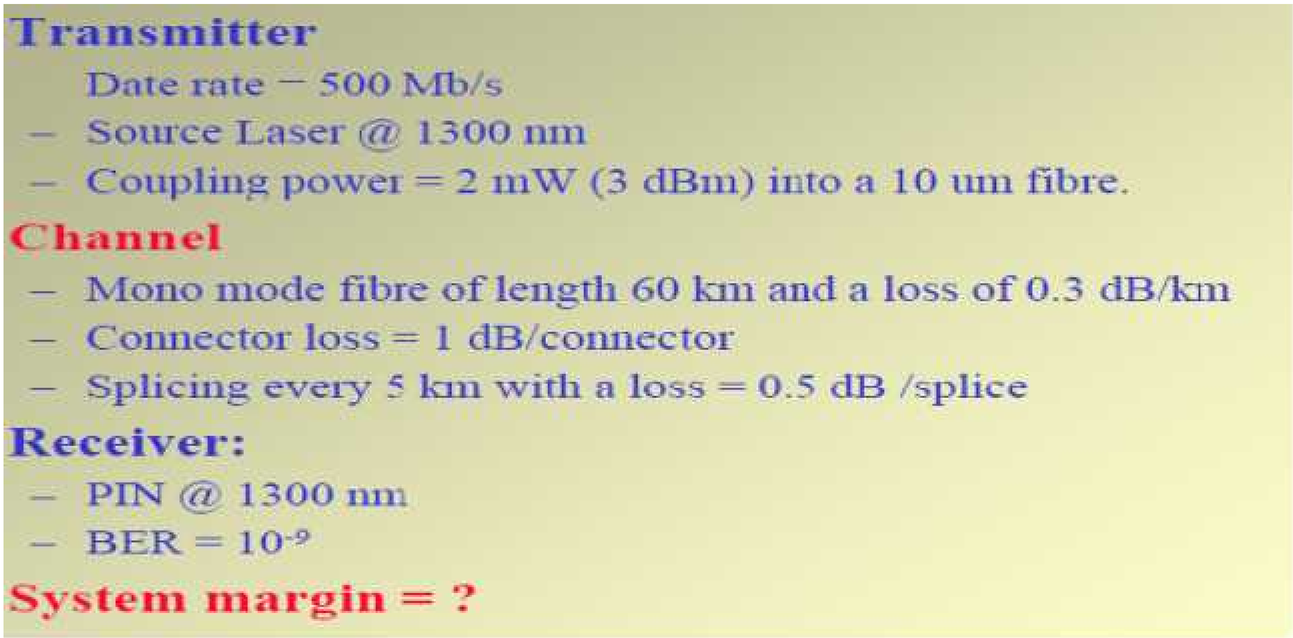
SM = System margin (to ensure that small variation the system operating parameters do not result in an unacceptable decrease in system performance)

Link Power Budget - Example 1:-



Parameters	Value	dB
▪ *Transmitter*		
▪ Average transmitted power	3 mW	4.8 dBm
▪ Fibre coupling losses		-3.7 dB
▪ *Channel*		
▪ Fibre loss		-15.7 dB
▪ Splitting losses		-10 dB
▪ Splice & Connector losses		-0.79 dB
▪ Fibre dispersion & nonlinearity		0 dB
▪ *Receiver*		
▪ Signal power at the receiver	All lossess	-26.79 dBm
▪ Receiver sensitivity		-31 dBm
System Margin (-20 dBm -(-30 dBm))		+4.1 dB

Link Power Budget - Example 2:-



Transmitter

Date rate – 500 Mb/s

– Source Laser @ 1300 nm

– Coupling power = 2 mW (3 dBm) into a 10 um fibre.

Channel

– Mono mode fibre of length 60 km and a loss of 0.3 dB/km

– Connector loss = 1 dB/connector

– Splicing every 5 km with a loss = 0.5 dB /splice

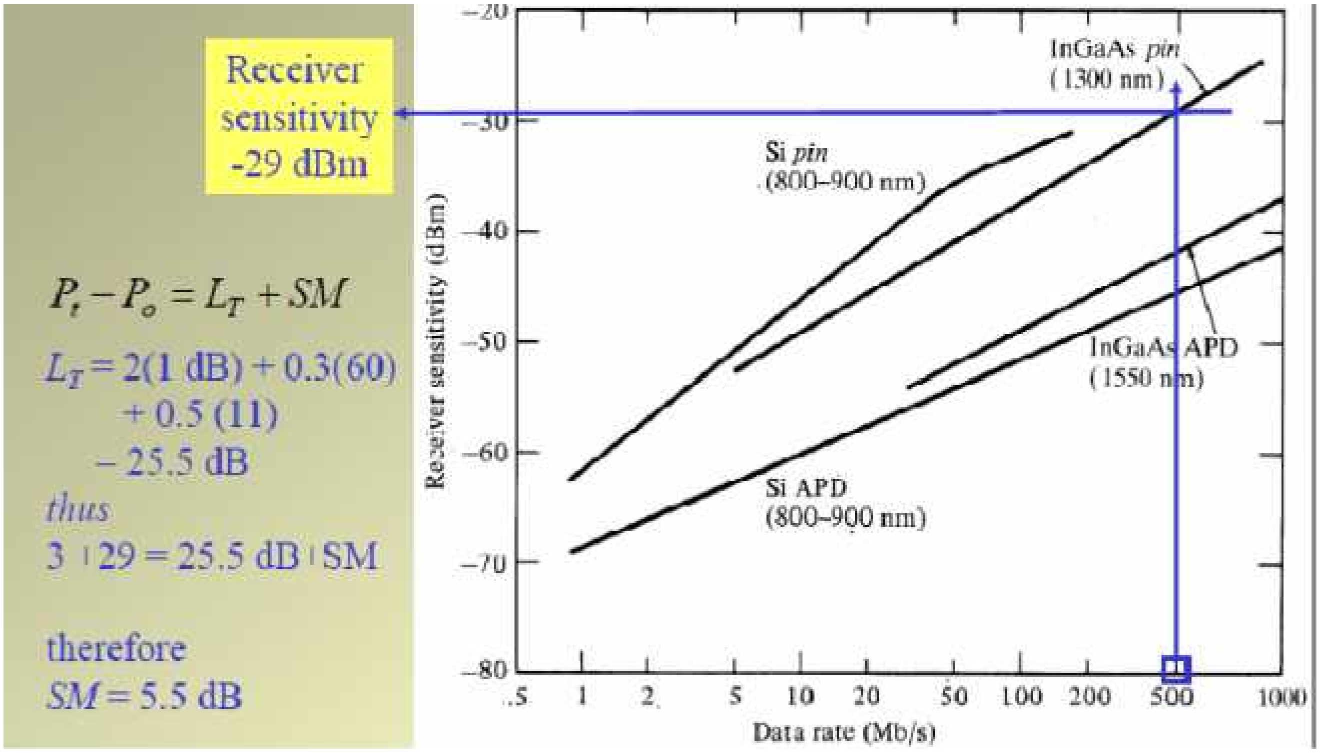
Receiver:

– PIN @ 1300 nm

– BER = 10^{-9}

System margin = ?

Link Power Budget - Example 2 *contd.*:-



Receiver sensitivity -29 dBm

$P_t - P_o = L_T + SM$

$L_T = 2(1\text{ dB}) + 0.3(60)$
$+ 0.5(11)$
$- 25.5\text{ dB}$

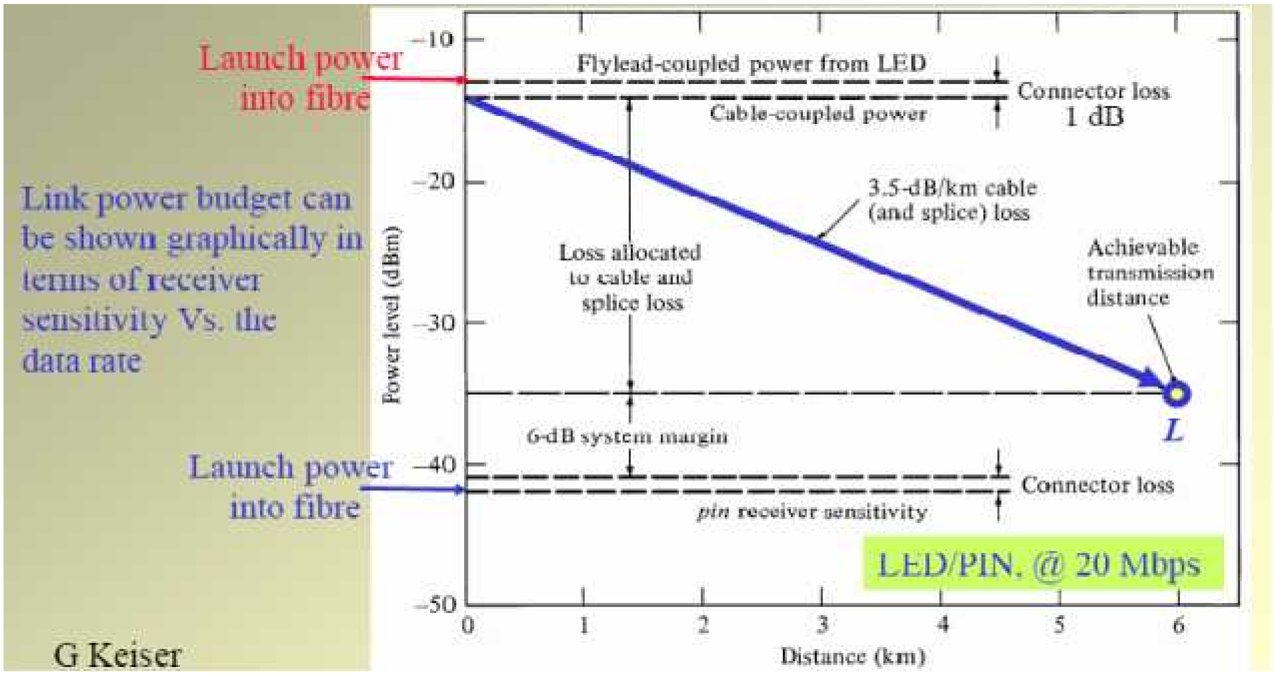
thus

3 ı29 = 25.5 dB ı SM

therefore

SM = 5.5 dB

Link-Power Budget - Example 3:-



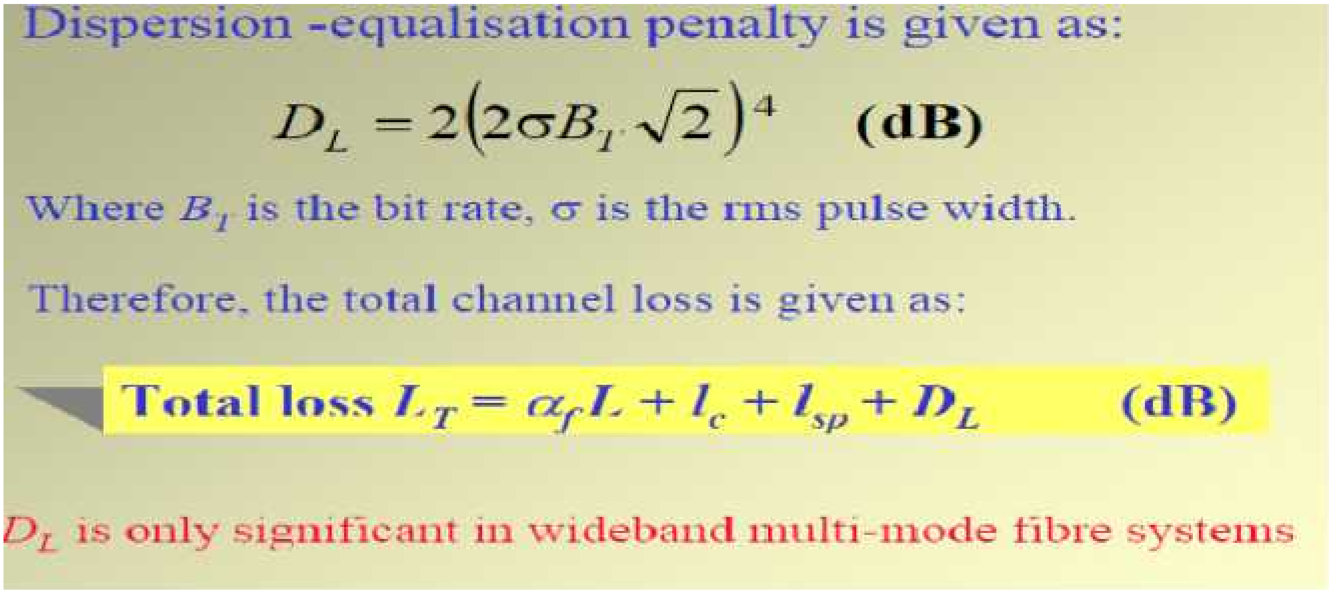
Launch power into fibre

Link power budget can be shown graphically in terms of receiver sensitivity Vs. the data rate

Launch power into fibre

LED/PIN, @ 20 Mbps

G Keiser



Dispersion -equalisation penalty is given as:

$$D_L = 2\left(2\sigma B_I \sqrt{2}\right)^4 \quad \textbf{(dB)}$$

Where B_I is the bit rate, σ is the rms pulse width.

Therefore, the total channel loss is given as:

$$\textbf{Total loss } L_T = \alpha_f L + l_c + l_{sp} + D_L \qquad \textbf{(dB)}$$

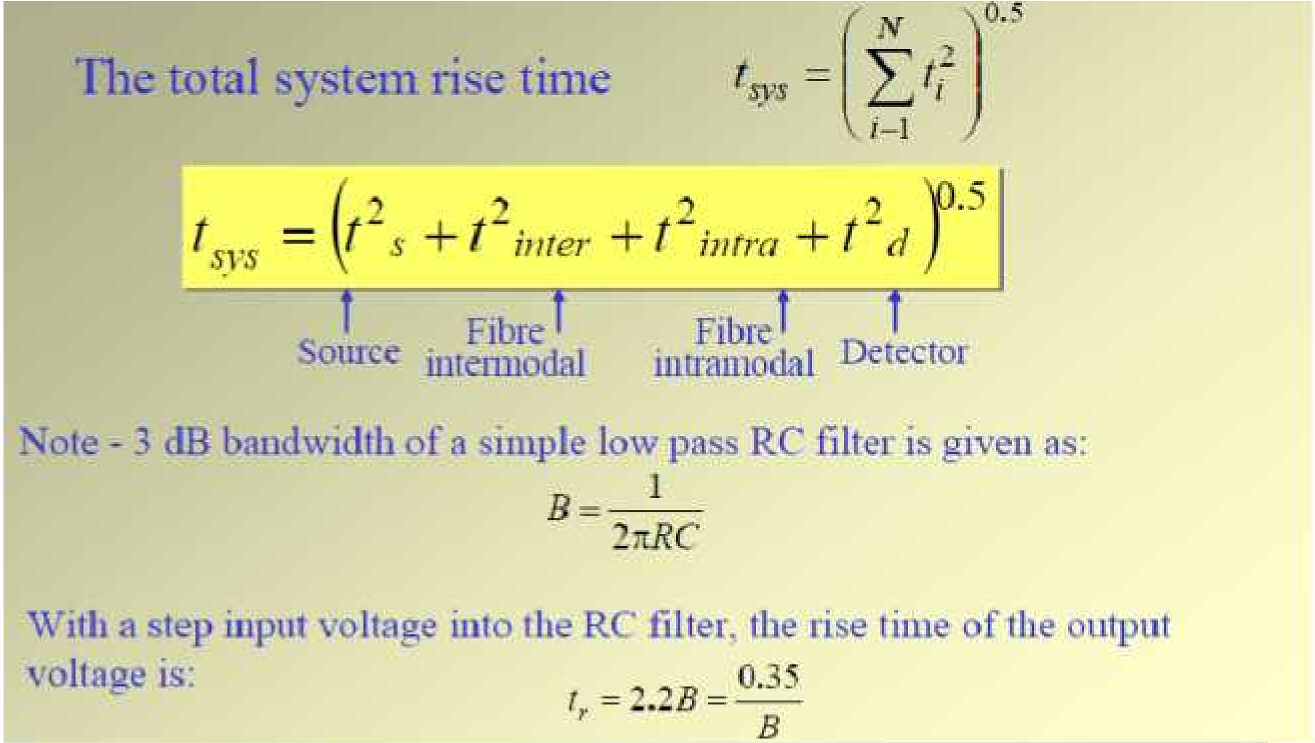
D_L is only significant in wideband multi-mode fibre systems

Rise Time Budget:-

The system design must also take into account the temporal response of the system components. The total loss LT (given in the power budget section) is determined in the absence of the any pulse broadening due to dispersion.

Finite bandwidth of the system (transmitter, channel, receiver) may results in pulse spreading (i.e. intersymbol interference), giving a **reduction in the receiver sencitivity**. I.e. worsening of BER or SNR

The additional loss penalty is known as **dispersion equalisation or ISI penalty.**



The total system rise time $t_{sys} = \left(\sum_{i-1}^{N} t_i^2\right)^{0.5}$

$$t_{sys} = \left(t^2{}_s + t^2{}_{inter} + t^2{}_{intra} + t^2{}_d\right)^{0.5}$$

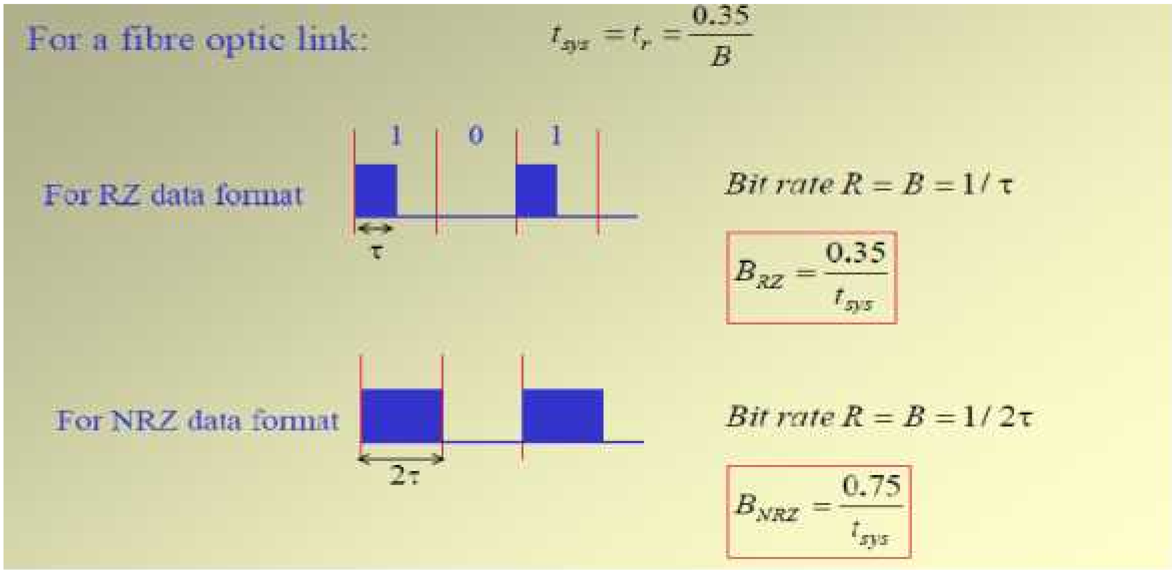
Source
Fibre intermodal
Fibre intramodal
Detector

Note - 3 dB bandwidth of a simple low pass RC filter is given as:

$$B = \frac{1}{2\pi RC}$$

With a step input voltage into the RC filter, the rise time of the output voltage is:

$$t_r = 2.2B = \frac{0.35}{B}$$



For a fibre optic link: $t_{sys} = t_r = \frac{0.35}{B}$

For RZ data format

Bit rate $R = B = 1/\tau$

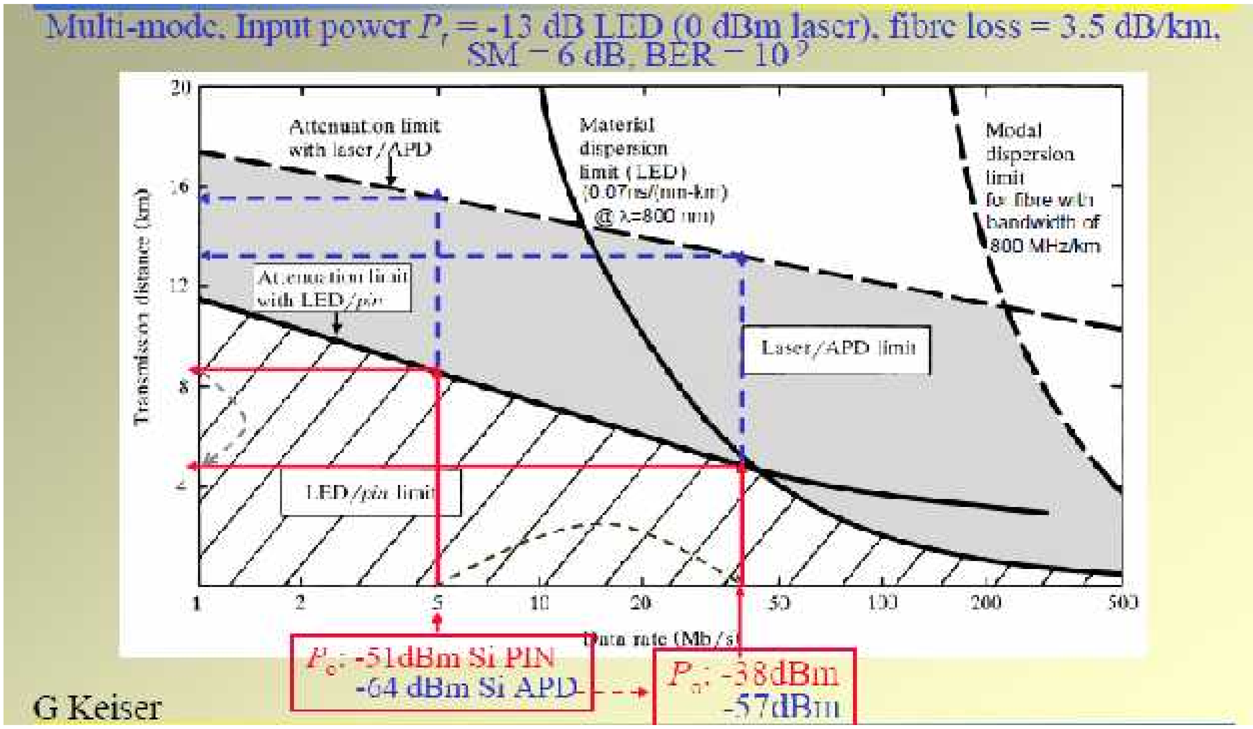
$$\boxed{B_{RZ} = \frac{0.35}{t_{sys}}}$$

For NRZ data format

Bit rate $R = B = 1/2\tau$

$$\boxed{B_{NRZ} = \frac{0.75}{t_{sys}}}$$

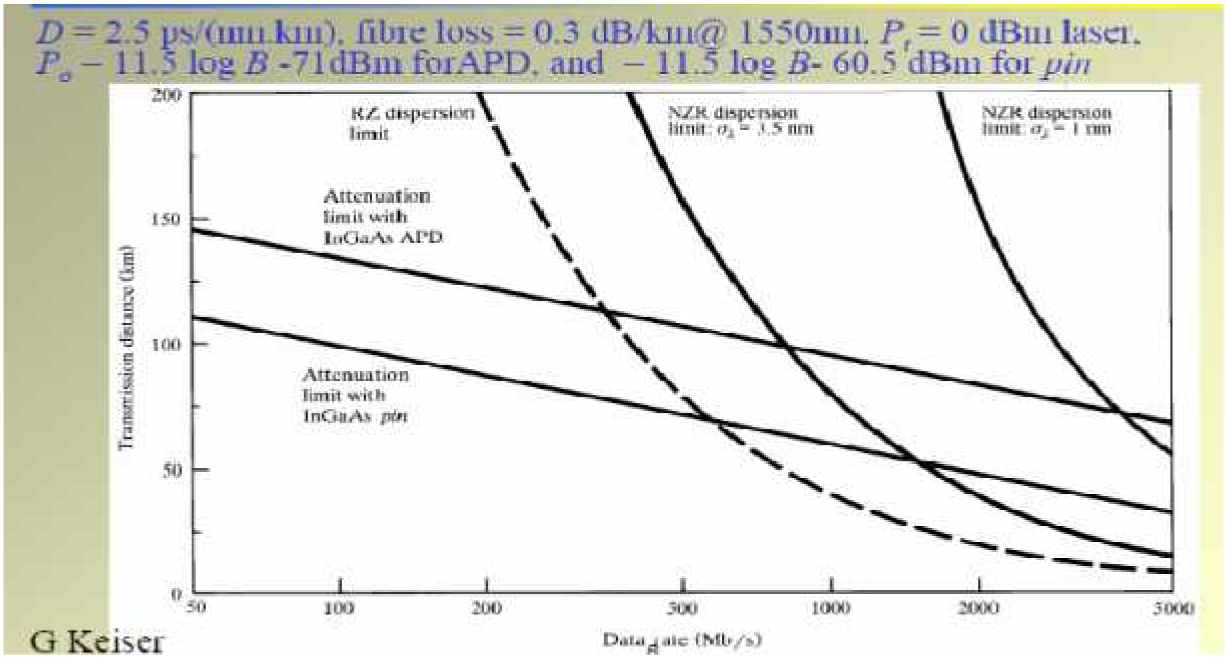
Transmission Distance -1st window:-



Multi-mode, Input power P_t = -13 dB LED (0 dBm laser), fibre loss = 3.5 dB/km, SM = 6 dB, BER = 10^{-9}

G Keiser

Transmission Distance -3rd window:-



$D = 2.5$ ps/(nm.km), fibre loss = 0.3 dB/km@ 1550nm, $P_t = 0$ dBm laser, $P_o - 11.5 \log B$ -71dBm forAPD, and $-11.5 \log B$- 60.5 dBm for *pin*

Analogue System:-

The system must have sufficient bandwidth to pass the HIGEST FREQUENCIES. Link Power budget is the same as in digital systems Rise Time budget is also the same, except for the system bandwidth which is defined as:

$$B_{sys} = \frac{0.35}{t_{sys}}$$

CHAPTER 1

INTRODUCTION TO RADAR SYSTEM

1.1 Introduction:-

Radar is an electromagnetic system for the detection and location of objects. It operates by transmitting a particular type of waveform, a pulse-modulated sine wave for example, and detects the nature of the echo signal. Radar is used to extend the capability of one's senses for observing the environment, especially the sense of vision.

An elementary form of radar consists of a transmitting antenna emitting electromagnetic radiation generated by an oscillator of some sort, a receiving antenna, and an energy-detecting device, or receiver.

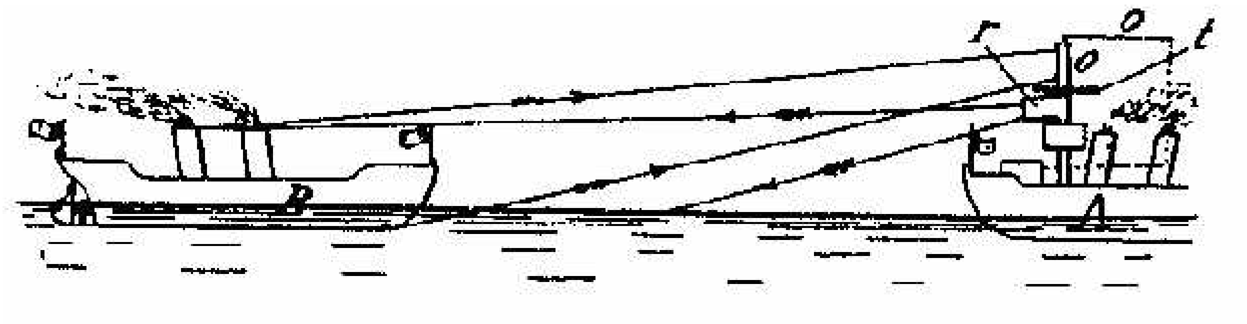
A portion of the transmitted signal is intercepted by a reflecting object (target) and is reradiated in all directions. It is the energy reradiated in the back direction that is of prime interest to the radar. The receiving antenna collects the returned energy and delivers it to a receiver, where it is processed to detect the presence of the target and to extract its location and relative velocity.

The distance to the target is determined by measuring the time taken for the radar signal to travel to the target and back. The direction, or angular position, of the target may be determined from the direction of arrival of the reflected wave- front. The usual method of measuring the direction of arrival is with narrow antenna beams. If relative motion exists between target and radar, the shift in the carrier frequency of the reflected wave (Doppler Effect) is a measure of the target's relative (radial) velocity and may be used to distinguish moving targets from stationary objects. In radars which continuously track the movement of a target, a continuous indication of the rate of change of target position is also available.

1.2 History Background

James Clerk Maxwell (1831 –1879) - predicted the existence of radio waves in his theory of electromagnetism. In 1886, Hertz experimentally tested the theories of Maxwell and demonstrated the similarity between radio and light waves. Hertz showed that radio waves could be reflected itself. Heinrich Hertz, in 1886, experimentally tested the theories of Maxwell and demonstrated the similarity between radio and light waves. Hertz showed that radio waves could be reflected by metallic and dielectric bodies. Due to these reflections occurred through metallic bodies given a start to the development of radar systems.

In 1903 a German engineer by the name of Hiilsmeyer experimented with the detection of radio waves reflected from ships. He obtained a patent in 1904 in several countries for an radio waves reflected from ships as shown in fig.1.



(a)

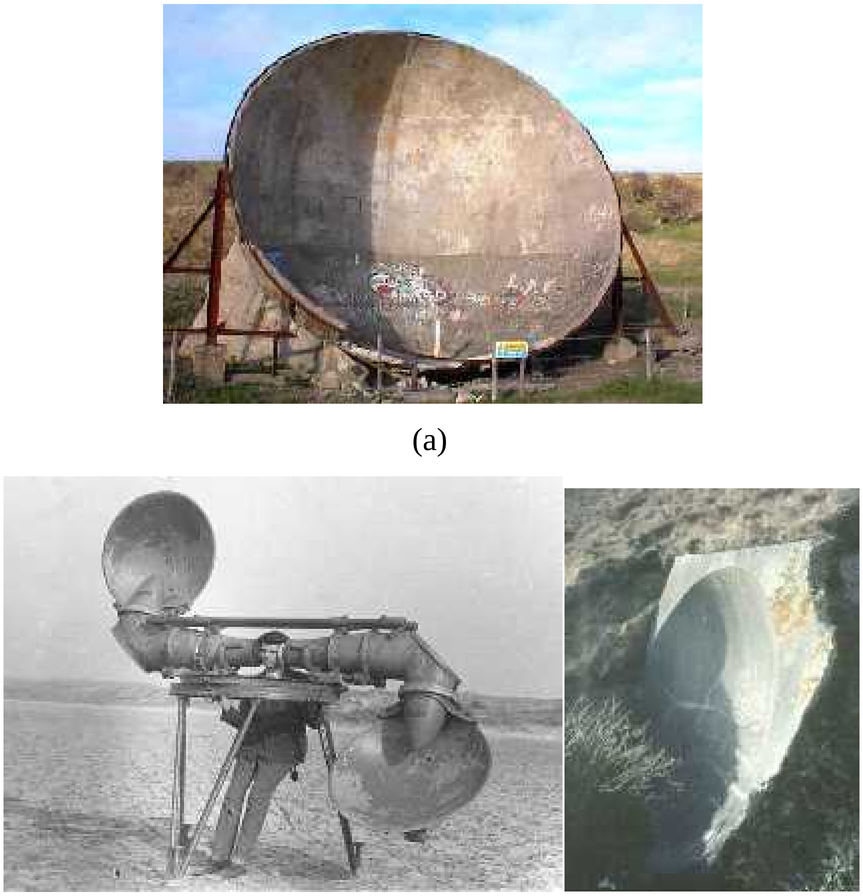
(b)

Fig. 1 (a) Detection of wooden ship in 1904 **(b)** Hülsmeyer 1904, who detected the first object through radar

In the autumn of 1922 A. H. Taylor and L. C. Young of the Naval Research Laboratory detected a wooden ship using a CW wave-interference radar with separated receiver and transmitter. The wavelength was 5 m. The first application of the pulse technique to the measurement of distance was in the basic scientific investigation by Breit and Tuve in 1925 for measuring the height of the ionosphere. However, more than a decade was to elapse before the detection of aircraft by pulse radar was demonstrated.

The first detection of aircraft using the wave-interference effect was made in June, 1930, by L. A. tlyland of the Naval Research Laboratory.' It was made accidentally while he was working with a direction-finding apparatus located in an aircraft on the ground. The transmitter at a frequency of 33 MHz was located 2 miles away, and the beam crossed an air lane from L. Hyland of the Naval Research Laboratory. It was made accidentally while he was working with a direction-finding apparatus located in an aircraft on the ground. The transmitter at a frequency of 33 MHz was located 2 miles away, and the beam crossed an air lane from a nearby airfield.

Before the advent of radar, the only practicable means of detection of aircraft was acoustic, and a network of acoustic detectors was built in the 1920s and 1930s around the south and east coast of the UK, some of which still remain. In calm air conditions, detection ranges of up to 25km were achievable.



(a)

(b)

(c)

Fig. 2 Different types of Acoustic Radars from 1920-1930

Radar Applications:-

In aviation, aircraft are equipped with radar devices that warn of aircraft or other obstacles in or approaching their path, display weather information, and give accurate altitude readings. The first commercial device fitted to aircraft was a 1938 Bell Lab unit on some United Air Lines aircraft. Such aircraft can land in fog at airports equipped with radar-assisted ground-controlled approach systems in which the plane's flight is observed on radar screens while operators radio landing directions to the pilot.

Marine radars are used to measure the bearing and distance of ships to prevent collision with other ships, to navigate, and to fix their position at sea when within range of shore or other fixed references such as islands, buoys, and lightships. In port or in harbour, vessel traffic service radar systems are used to monitor and regulate ship movements in busy waters.

Normal radar functions:

1. Range (from pulse delay)
2. Velocity (from Doppler frequency shift)
3. Angular direction (from antenna pointing)

Signature analysis and inverse scattering:

4. Target size (from magnitude of return)
5. Target shape and components (return as a function of direction)
6. Moving parts (modulation of the return)
7. Material composition

The complexity (cost & size) of the radar increases with the extent of the functions that the radar performs.

CHAPTER 2: BASIC PRINCIPLES OF RADAR

A radar system has a transmitter that emits radio waves called *radar signals* in moving or stationary target directions. When these come into contact with an object they are usually reflected or scattered in many directions. Radar signals are reflected especially well by materials of considerable electrical conductivity especially by most metals, by seawater and by wet ground. Some of these make the use of radar altimeters possible. The radar signals that are reflected back towards the transmitter are the desirable ones that make radar work. If the object is *moving* either toward or away from the transmitter, there is a slight equivalent change in the frequency of the radio waves, caused by the Doppler effect.

The basic principle of the radar is shown in fig. 2.1. A transmitter generates an electromagnetic signal that is radiated by the antenna into space. A portion of the transmitted electromagnetic energy is reflected back by the target towards the radar. Based on the received target echo signal the receiver made decision for the position, range and direction of the target. The term radar is a contraction of the words radio detection and ranging.

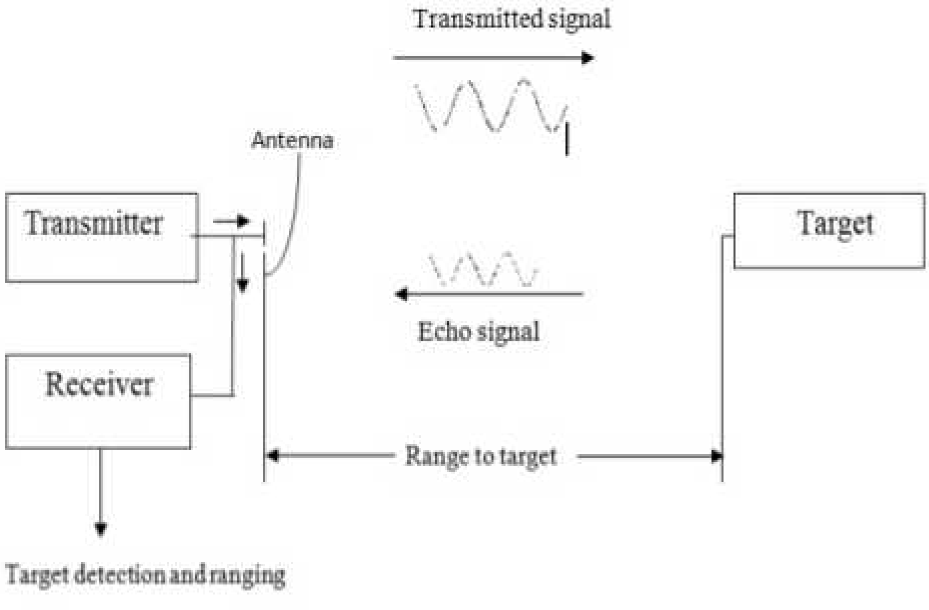


Fig. 2.1 Basic Principles of the Radar

The basic terminology used for radar is discussed as follows.

Range:- The range of the target is observed by measuring the time (T_R) it takes for the radar signal to travel to the target and return back to the radar. Thus the time for the signal to travel to the target located at range (*R*) and the return back to the radar is *2R/C*. The range of the target can be given as:

$$R = \frac{cT_R}{2} \qquad \ldots (1)$$

with the range in kilometers or in nautical miles, and T in microseconds.

$$\begin{aligned} R(km) &= 0.15T_R\ (\text{«}s) \\ R(nmi) &= 0.081T_R\ (\text{«}s) \end{aligned} \qquad \ldots (2)$$

Maximum Unambiguous Range:- Once a signal is radiated into space by a radar, enough time must elapse to allow all echo signal to return to the radar before the transmission of next pulse. The rate at which the pulses are transmitted, is determined by the longest range of the target. If the time between pulses T_P is too short, an echo signal from the long range target might arrive after the transmission of the next pulse. The echo that arrives after the transmission of next pulse is called as *second-time-around-echo (or multiple-time-around-echo*). Such an echo would appear to be at a closer range than actual, this range measurement will be misleading for range calculation, if it is not known that this is second time echo. The range beyond which the target appears as second-time-around-echoes is the *maximum unambiguous range,* R_{un} and is given by

$$\begin{aligned} R_{un} &= \frac{cT_P}{2} = \frac{c}{2f_p} \\ f_p &= \frac{1}{T_P} \\ Duty\,cycle &= \frac{2}{T_P} \end{aligned} \qquad \ldots (3)$$

Where T_P is the pulse repletion time and f_p is the pulse repetition frequency.

A problem with pulsed radars and range measurement is how to unambiguously determine the range to the target if the target returns a strong echo. This problem arises because of the fact that pulsed radars typically transmit a sequence of pulses. The radar receiver measures the time between the leading edges of the last transmitting pulse and the echo pulse. It is possible that an echo will be received from a long range target after the transmission of a second transmitting pulse.

In this case, the radar will determine the wrong time interval and therefore the wrong range. The measurement process assumes that the pulse is associated with the second transmitted pulse and declares a much reduced range for the target. This is called range ambiguity and occurs where there are strong targets at a range in excess of the pulse repetition time. The pulse repetition time defines a maximum unambiguous range. To increase the value of the unambiguous range, it is necessary to increase the PRT, this means: to reduce the PRF.

Echo signals arriving after the reception time are placed either into the transmit time where they remain unconsidered since the radar equipment isn't ready to receive during this time, or into the following reception time where they lead to measuring failures (ambiguous returns).

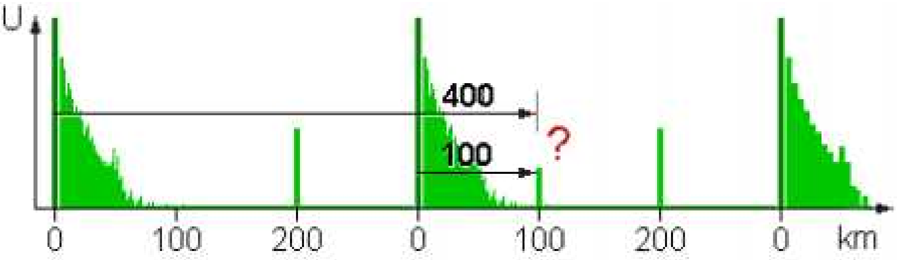


Fig. 2.2 a second-time-arounr-echo in a distance of 400 km assumes a wrong range of 100 km

Pulse Repetition Frequency (PRF):- The rate at which the pulses are transmitted towards the target from the radar is called as the pulse repletion frequency, f_p .

$$f_p = \frac{1}{T_P} \qquad \ldots (4)$$

Pulse Repetition Period:- The time interval at which the pulses are periodically transmitted towards the target from the radar is called as the pulse repletion period, T_P is given by in terms of prf.

$$T_p = \frac{1}{f_P} \qquad \ldots (5)$$

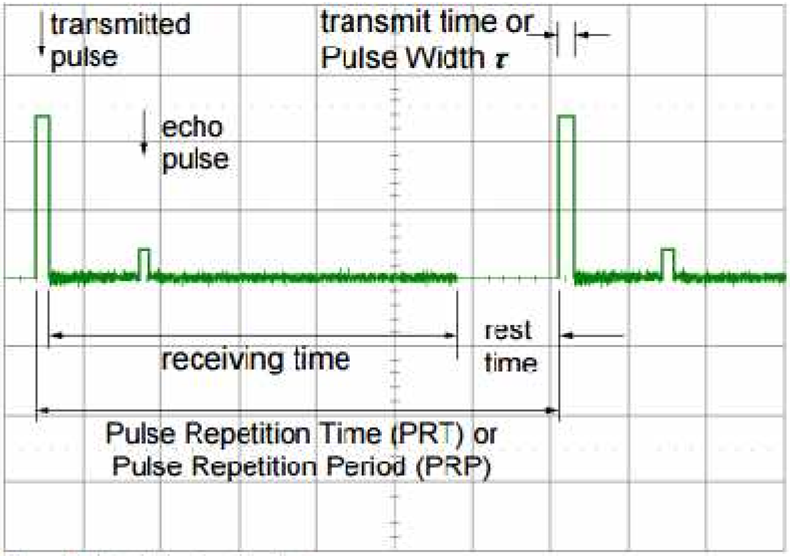

transmitted pulse
transmit time or Pulse Width τ
echo pulse
receiving time
rest time
Pulse Repetition Time (PRT) or
Pulse Repetition Period (PRP)

Fig. 2.3 A typical radar time line

Duty Cycle:- The duty cycle of the radar waveform is described as the ratio of the total time the radar is radiating to the total time it could have radiated.

$$Duty\,cycle = \frac{p_{av}}{P_T} \qquad \ldots (6)$$

$$Duty\,cycle = \frac{^2}{T_P} = {}^2 f_p \qquad \ldots (7)$$

Where ^{2} is pulse width of the transmitted pulse and T_P is the pulse repetition period.

Peak Power of the Radar:- The maximum power of the radar antenna, that can be transmitted for the maximum unambiguous range target detection in particular direction.

Average Power of the Radar:- The average power of the radar antenna, that can be transmitted for the maximum unambiguous range target detection in all the direction (for isotropic antenna).

Radar Wave forms:- Typical radar utilizes various waveforms for target detection.

- **Pulse waveform:-** A radar uses rectangular pulse wave form with pulse width of 1microsecond, pulse repletion period 1 millisecond.
- **Continuous waveform:-** A very long continuous waveform are required for some long range radars to achieve sufficient energy for small target detection.

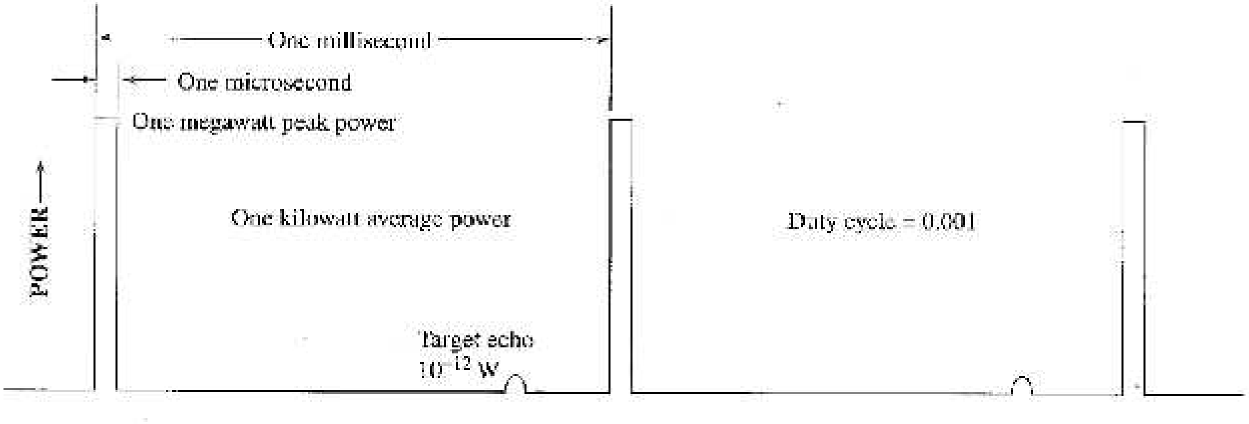


Fig.2.4 Example of typical pulse waveform for medium range air surveillance radar

CHAPTER: 3 RADAR RANGE EQUATION

3.1 Introduction:-

The radar range relates the radar range with the characteristics of transmitter, receiver antenna, target and environment. The radar range equation is useful to understand the maximum range of the radar that can be detected by the radar with their performance parameters. One of the simpler equations of radar theory is the radar range equation.

3.2 BASIC RADAR RANGE EQUATIONS

The transmitted power P_t is radiated by an isotropic antenna, the power density at distance R can be given as:

$$\text{Power demnsity at range R from an isotropic antenna} = \frac{P_t}{4® R^2} \quad \text{(Watt/square meter)} \qquad \ldots (3.1)$$

The maximum gain of the antenna can be defined as:

$$G = \frac{\text{max power density radiated byan antenna}}{\text{power density radiated by a lossless isotropic antenna}} \qquad \ldots (3.2)$$

Thus the power density at target from a directive antenna can be given as:

$$\text{Power density at range R from } a \text{ directive antenna} = \frac{P_t G}{4® R^2} \qquad \ldots (3.3)$$

The target receives a portion of the incident energy and reflected it in various directions. Thus the radar cross section of the target determines the power density returned back to the radar.

The reflected power from the target through its cross section (target cross section) can be given as:

$$\text{Reflected power from the target towards the radar} = \frac{P_t G}{4® R^2} \bullet \frac{\pm}{4® R^2} \qquad \ldots (3.4)$$

The radar antenna receives a portion of the reflected power from the target cross section. the received power can be given as:

$$P_r = \frac{P_t G}{4® R^2} \bullet \frac{\pm}{4® R^2} \bullet A_e \qquad \ldots (3.5)$$

$$A_e = {}^{\circ}{}_{a} \bullet A \qquad \ldots (3.6)$$

Where A_e is the effective area of the receiving antenna, A is the physical antenna area and ${}^{\circ}{}_{a}$ is the antenna aperture efficiency. The maximum range of the radar (Rmax) can be defined as the maximum distance beyond which radar cannot detect the target. So the received signal power can be given as the minimum detectable signal.

$$S_{\min} = \frac{P_t G}{4\circledR R^2} \bullet \frac{\pm}{4\circledR R_{\max}{}^2} \bullet A_e \qquad \ldots (3.7)$$

$$R_{\max} = \left[\frac{P_t G}{4\circledR} \bullet \frac{\pm}{4\circledR} \bullet \frac{A_e}{S_{\min}} \right]^{1/4} \qquad \ldots (3.8)$$

This is the fundamental form of radar range equation. If the antenna is used for both the transmission and receiving purpose, then the transmitted gain (G) can be given in terms of the effective area (A_e).

$$G = \frac{4\circledR A_e}{\mathrm{a}^2} \qquad \ldots (3.9)$$

Now the maximum radar range can be given as follows.

$$R_{\max} = \left[\frac{P_t G^2 \mathrm{a}}{(4\circledR)^3} \bullet \pm \bullet \frac{A_e}{S_{\min}} \right]^{1/4} \quad \text{(When G is constant)} \qquad \ldots (3.10)$$

$$R_{\max} = \left[\frac{P_t}{(4\circledR)^3} \bullet \pm \bullet \frac{A_e{}^2}{S_{\min}} \right]^{1/4} \quad \text{(When } A_e \text{ is constant)} \qquad \ldots (3.11)$$

These three forms of radar range equations [2.8, 2.10 and 2.11] are based on the effective area (A_e) and transmitter antenna gain (G).

3.3 Radar Block Diagram

The operation of a typical pulse radar may be described with the aid of the block diagram shown in Fig. 1.2. The transmitter may be an oscillator. such as a magnetron. that is "pulsed" (turned on and off) by the modulator to generate a repetitive train of pulses. The magnetron has prohahly heen the most widely used of the various microwave generators for radar. A typical radar for the detection of aircraft at ranges of 100 or 200 nmi might employ a peak power of the order of a megawatt. an average power of several kilowatts, a pulse width of several microseconds. and a

pulse repetition frequency of several hundred pulses per second. The waveform generated by the transmitter travels via a transmission line to the antenna.
where it is radiated into space.

A single antenna is generally used for both transmitting and receiving. The receiver must be protected from damage caused by the high power of the transmitter. This is the function of the duplexer. The receiver is usually of the superheterodyne type. The first stage might be a low-noise RF amplifier. such as a parametric amplifier or a low-noise transistor. However. it is not always desirable to employ a low-noise first stage in radar.

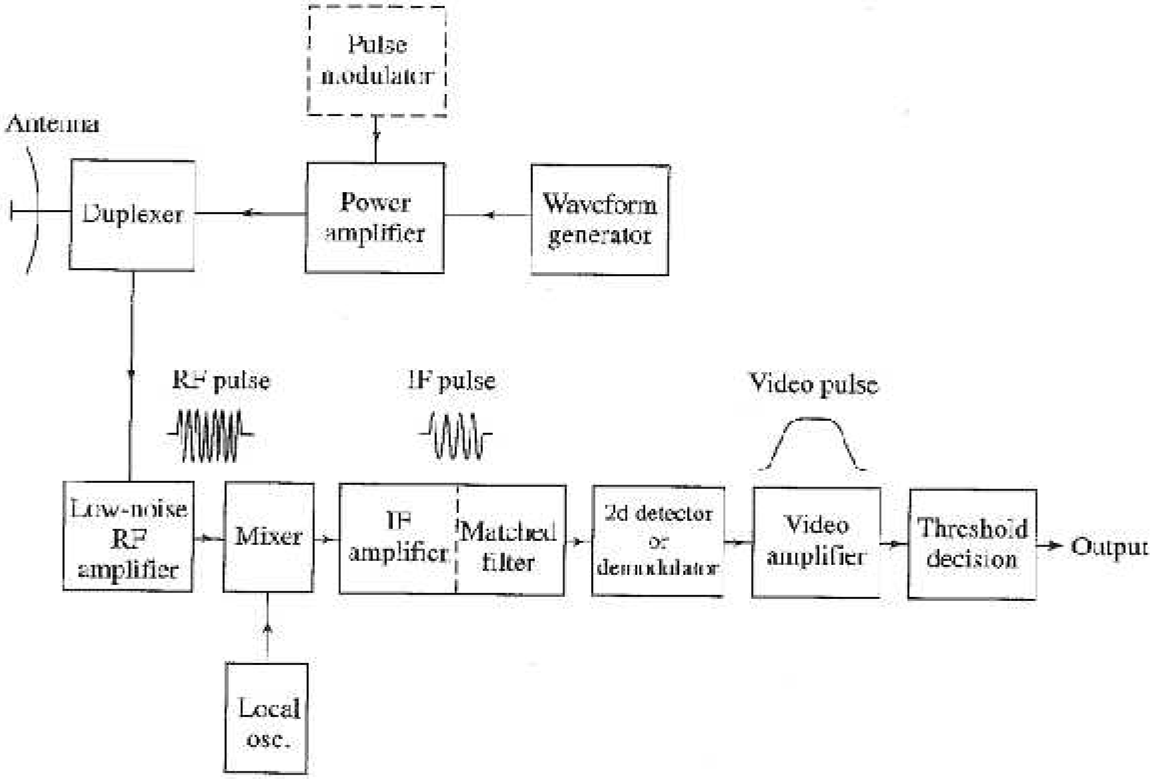


Fig. 3.1 Radar Block Diagram

The mixer and local oscillator (LO) convert the RF signal to an intermediate frequency (IF). A "typical" IF amplifier for an air-surveillance radar might have a center frequency of 30 or 60 MHz and a bandwidth of the order of one megahertz.

The IF amplifier should be designed as a matchted filter; i.e., its frequency-response function H(f) should maximize the peak-signal-to-mean-noise-power ratio at the output.

After maximizing the signal-to-noise ratio in the IF amplifier, the pulse modulation is extracted by the second detector and amplified by the video amplifier to a level where it can be properly displayed, usually on a cathode-ray tube (CRT). Timing signals are also supplied to the indicator to provide the range zero. Angle information is obtained from the pointing direction of the antenna.

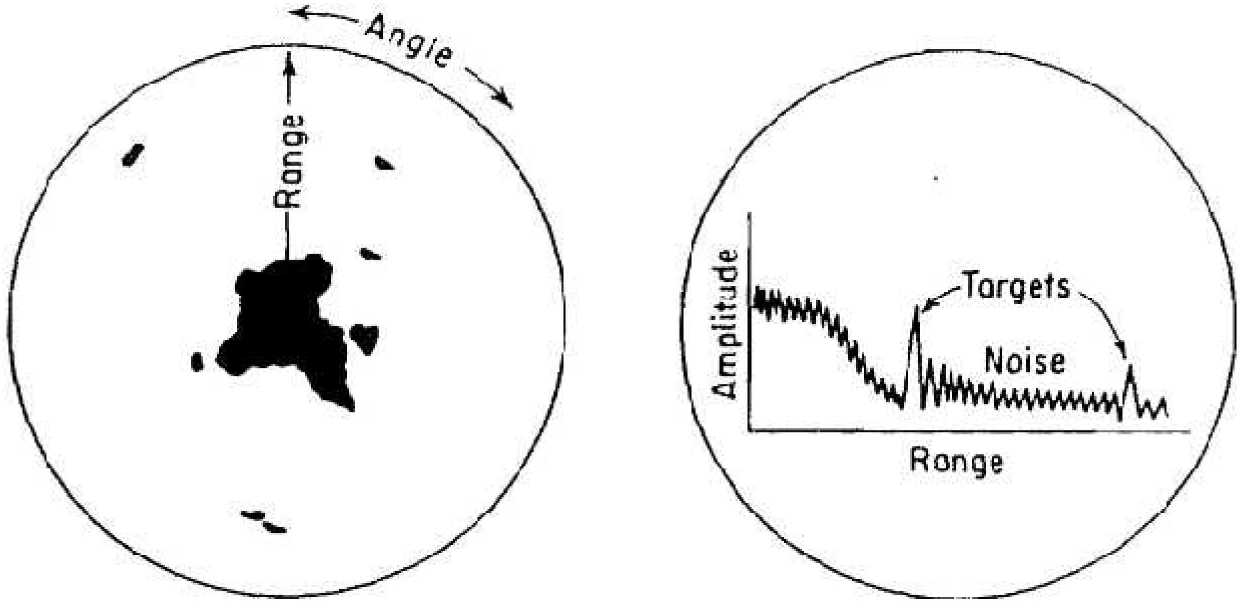


Fig. 3.2 (a) PPI presentation displaying range vs. angle (intensity modulation); (b) A scope presentation displaying amplitude vs. range (deflection modulation).

A common form of radar antenna is a reflector with a parabolic shape, fed (illuminated) from a point source at its focus. The parabolic reflector focuses the energy into a narrow beam, just as does a searchlight or an automobile headlamp. The beam may be scanned in space by mechanical pointing of the antenna. Phased-array antennas have also been used for radar. In a phased array the beam is scanned by electronically varying the phase of the currents across the aperture.

3.3 Radar’s Electromagnetic Spectrum

Conventional radars generally have been operated at frequencies extending from about 220 MHz to 35 GHz, a spread of more than seven octaves. These are not necessarily the limits, since radars can be, and have been, operated at frequencies outside either end of this range.Skywave HF over-the-horizon (OTH) radar might be at frequencies as low as 4 or 5 MHz, and Groundwave HF radars as low as 2 MHz. At the other end of the spectrum, millimeter radars Have operated at 94 GHz. Laser radars operate at even higher frequencies.

The place of radar frequencies in the electromagnetic spectrum is shown in Fig. 3.3. Some of the nomenclature employed to designate the various frequency regions is also shown. Early in the development of radar, a letter code such as S, X, L, etc., was employed to Designate radar frequency bands. Although its original purpose was to guard military secrecy, the designations were maintained, probably out of habit as well as the need for some convenient short nomenclature. This usage has continued and is now an accepted practice of radar engineers.

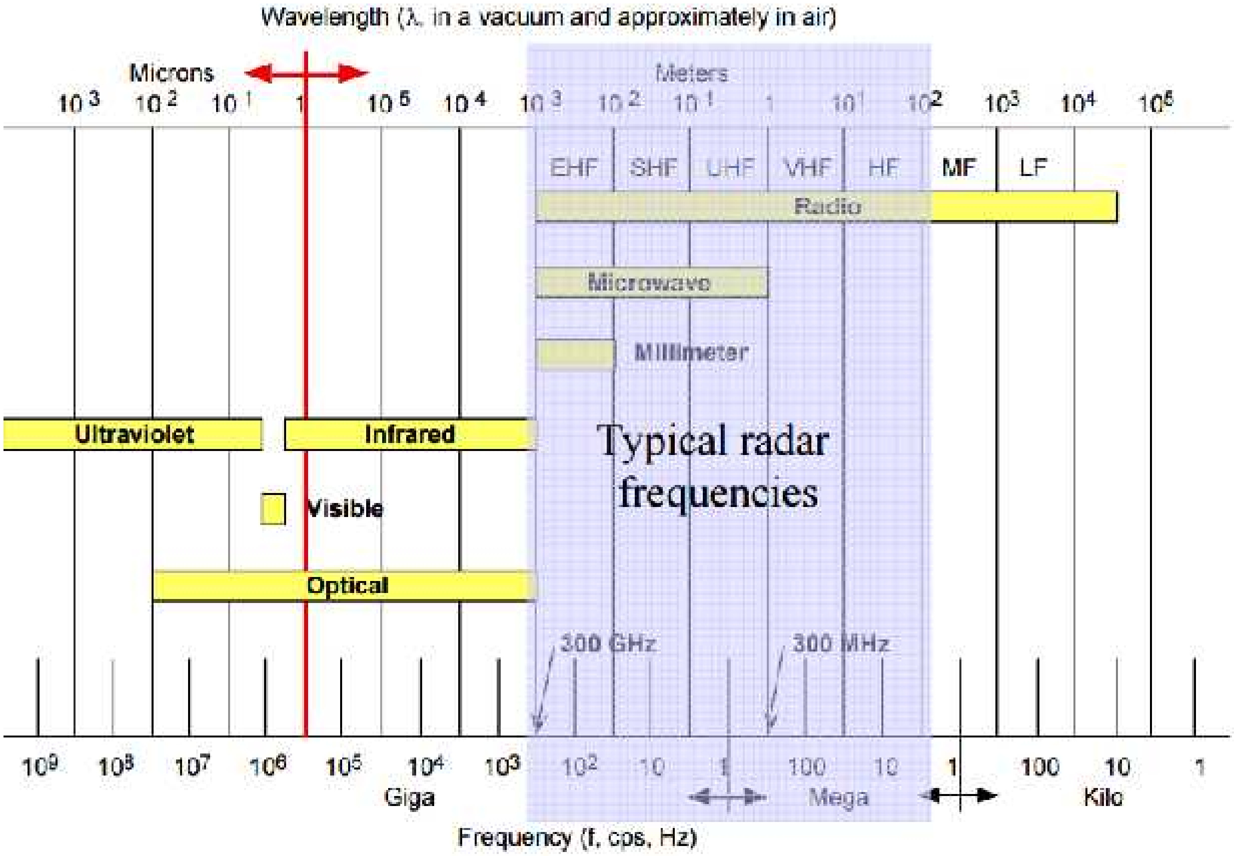
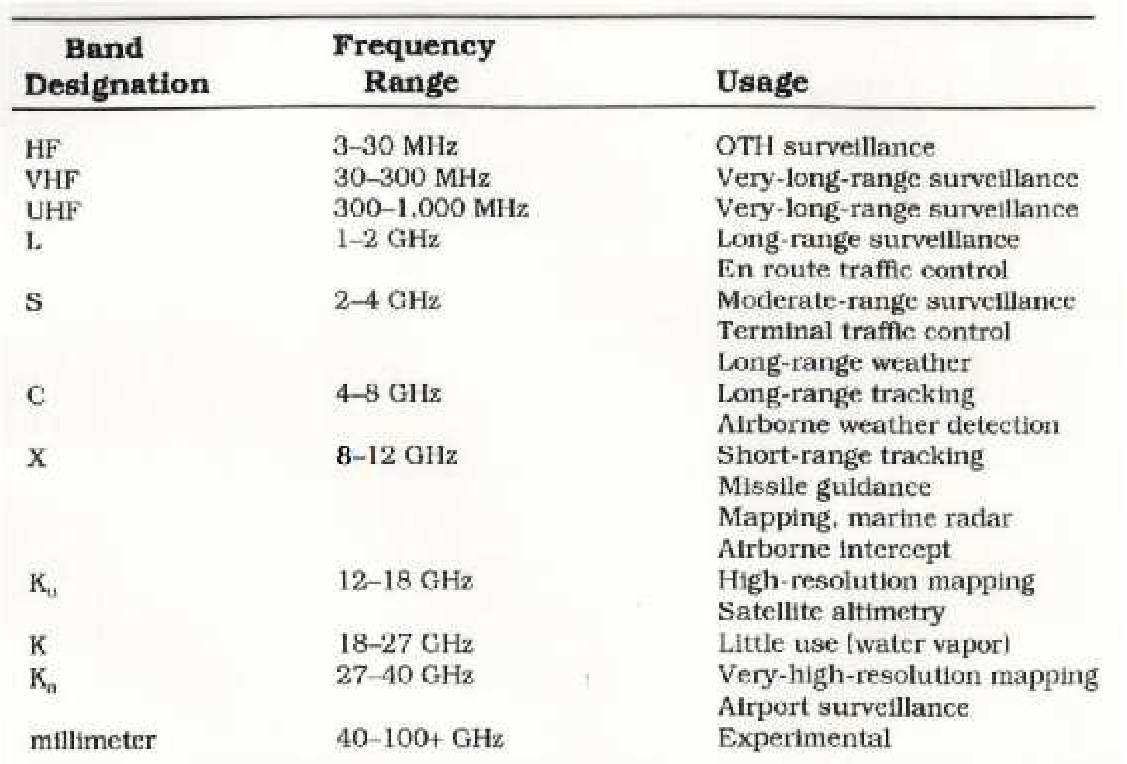


Fig. 3.3 Frequency spectrum for radar frequencies

Table 3.1 lists the radar-frequency letter-band nomenclature adopted by the IEEE. These are related to the specific bands assigned by the International Telecommunications Union for radar. For example, although the nominal frequency range for L band is 1000 to 2000 MHz, an L-band radar is thought of as being confined within the region from 1215 to 1400 MHz since that is the extent of the assigned band.

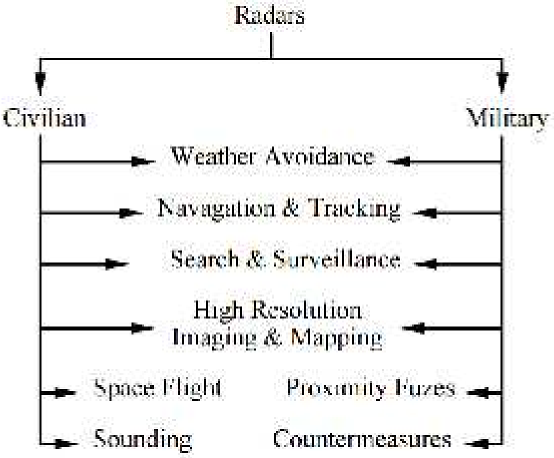
Table 3.1 Radar Bands and their Usage



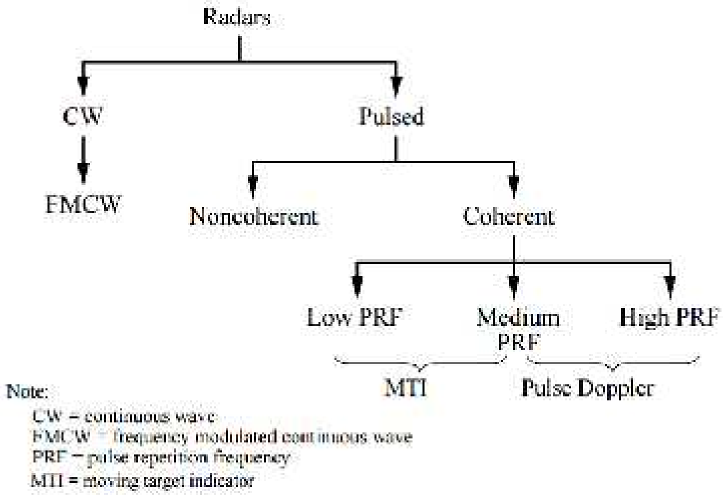
Band Designation	Frequency Range	Usage
HF	3–30 MHz	OTH surveillance
VHF	30–300 MHz	Very-long-range surveillance
UHF	300–1,000 MHz	Very-long-range surveillance
L	1–2 GHz	Long-range surveillance En route traffic control
S	2–4 GHz	Moderate-range surveillance Terminal traffic control Long-range weather
C	4–8 GHz	Long-range tracking Airborne weather detection
X	8–12 GHz	Short-range tracking Missile guidance Mapping, marine radar Airborne intercept
K_u	12–18 GHz	High-resolution mapping Satellite altimetry
K	18–27 GHz	Little use (water vapor)
K_a	27–40 GHz	Very-high-resolution mapping Airport surveillance
millimeter	40–100+ GHz	Experimental

3.4 Radar classification

Radar can be classified based on the function and the waveforms



(a)


Radars
CW
Pulsed
FMCW
Noncoherent
Coherent
Low PRF
Medium PRF
High PRF
MTI
Pulse Doppler
Note:
CW = continuous wave
FMCW = frequency modulated continuous wave
PRF = pulse repetition frequency
MTI = moving target indicator

(b)

Fig. 3.3 Radar can be classified based on the (a) function and (b) waveforms

In practice, however, the simple radar equation does not predict the range performance of actual radar equipments to a satisfactory degree of accuracy. The predicted values of radar range are usually optimistic. In some cases the actual range might be only half that predicted. Part of this discrepancy is due to the failure of Eq. (3.10) to explicitly include the various losses that can occur throughout the system or the loss in performance usually experienced when electronic equipment is operated in the field rather than under laboratory-type conditions &.another important factor that must be considered in the radar equation is the statistical or unpredictable nature of several of the parameters. The minimum detectable signal Smin and the target cross section (σ) are both statistical in nature and must be expressed in statistical terms.

3.5 MINIMUM DETECTABLE SIGNAL

The ability of a radar receiver to detect a weak echo signal is limited by the noise energy that occupies the same portion of the frequency spectrum as does' the signal energy. The weakest signal the receiver can detect is called. the minimum detectable signal. The specification of the minimum detectable signal is sometimes difficult because of its statistical nature and because the criterion for deciding whether a target is present or not may not be too well defined.

Detection is based on establishing a threshold level at the output of the receiver. If the Receiver output exceeds the threshold, a signal is assumed to be present. This is called threshold detection.

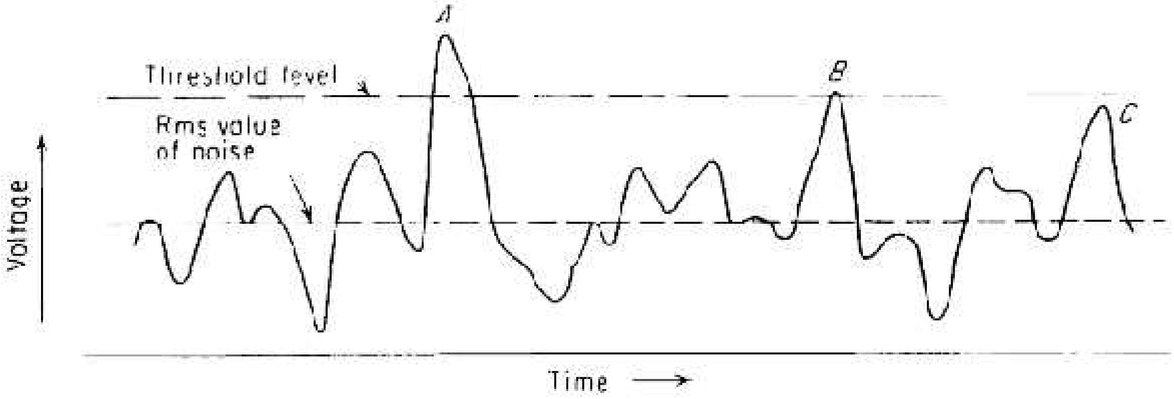


Fig. 3.4 Typical envelope of the radar receiver output as a function of time. A, and B, and C represent signal plus noise. A and B would be valid detections, but C is a missed detection.

A target is said to be detected if the envelope crosses the threshold. if the signal is large such as at A, it is not difficult to decide that a target is present. But consider the two signals at B and C, representing target echoes of equal amplitude. The noise voltage accompanying the signal at B is large enough so that the combination of signal plus noise exceeds the threshold.

Weak signals such as C would not be lost if the threshold level were lower. But too low a threshold increases the likelihood that noise alone will rise above the threshold and be taken for a real signal. Such an occurrence is called a false alarm.

CHAPTER 4: CONTINUOUS WAVE AND FREQUENCY MODULATED RADAR

4.1 THE DOPPLER EFFECT

It is well known in the fields of optics and acoustics that if either the source of oscillation or the observer of the oscillation is in motion, an apparent shift in frequency will result. This is the doppler effect and is the basis of CW radar.

If R is the distance from the radar to target, the total number of wavelengths (λ) contained in the two-way path between the radar and the target is 2R/λ. The distance R and the wavelength (λ), are assumed to be measured in the same units. Since one wavelength corresponds to an angular excursion of 2π radians, the total angular excursion ϕ made by the electromagnetic wave during its transit to and from the target is $4® R / ª$.

If target is in motion the range R and phase φ is continually changing. Thus the change in phase with respect to time can be given as frequency.

$$\frac{d¤}{dt} = \frac{4®}{ª}\frac{dR}{dt} \qquad \ldots (1)$$

Range with respect to time can be defined as the radial velocity of the target. Thus the Doppler angular frequency can be given as:

$$µ_d = 2® f_d = \frac{4®}{ª} v_r \qquad \ldots (2)$$

Where f_d is Doppler frequency and v_r is the radial velocity of the target with respect to radar. The Doppler frequency can be related with transmitter frequency f_0.

$$f_d = \frac{2v_r}{ª} = \frac{2v_r f_0}{c} \qquad \ldots (3)$$

When v_r is given in knots then the Doppler frequency can be given as:

$$f_d = \frac{1.03 v_r\ (knots)}{ª\,(m)} \qquad \ldots (4)$$

The relative velocity may be written $v_r = v\cos¯$ where v is the target speed and θis the Angle made by the target trajectory and the line joining radar and target. When θ = 0, the doppler frequency is maximum. The doppler is zero when the trajectory is perpendicular to the radar line of sight (θ= 90°).

A plot of doppler frequency shifts as a function of radial velocity and the radar frequency bands is given in fig. 4.2. This figure illustrates that as the target radial velocity get increases the Doppler frequency shifts get increases with higher radar frequencies.

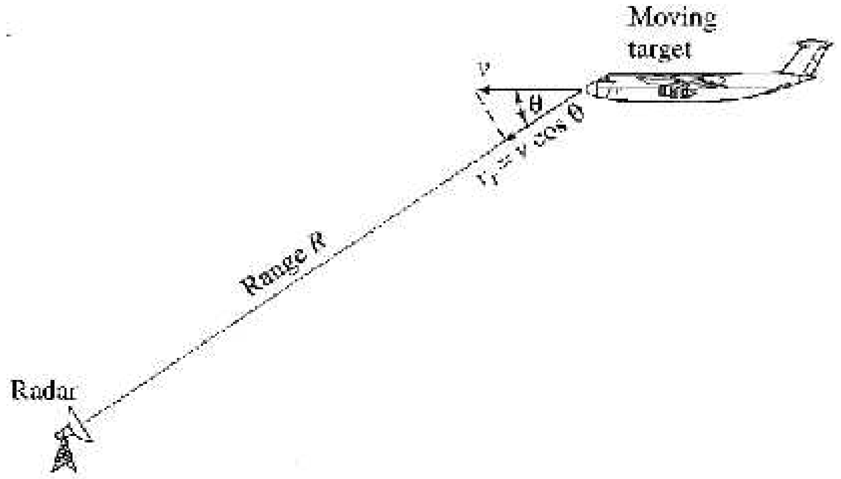


Fig. 4.1 Geometry of Radar and target in deriving the Doppler shifts

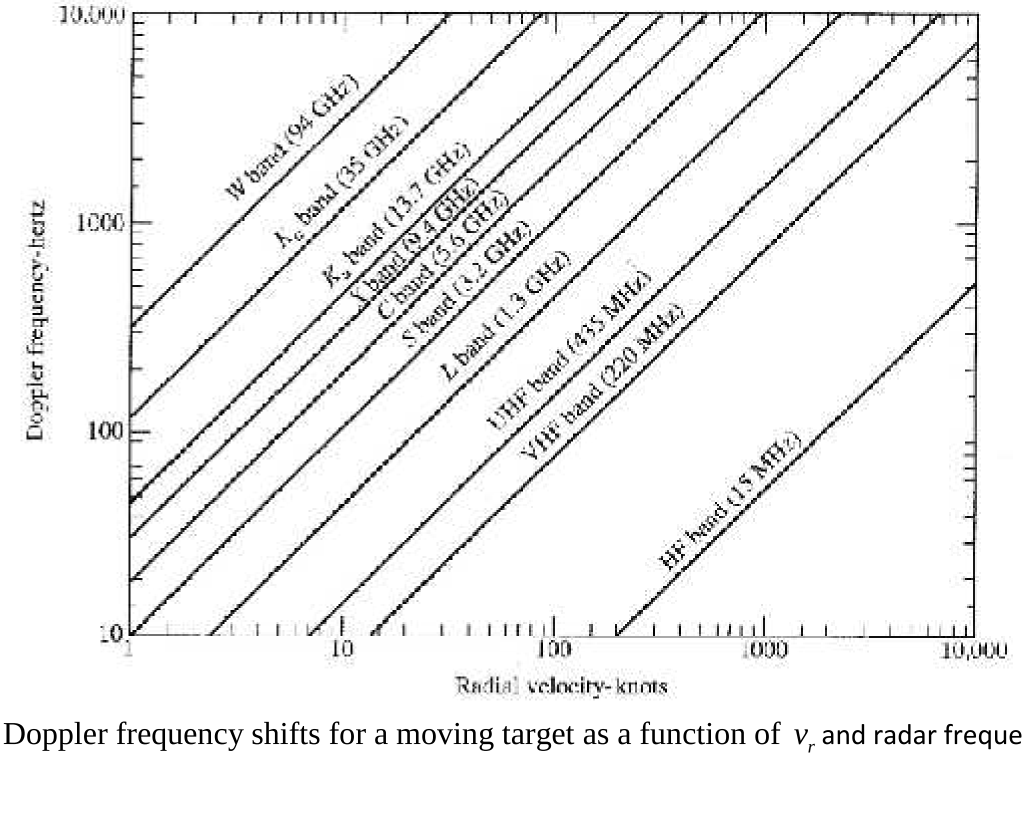


Fig. 4.2 Doppler frequency shifts for a moving target as a function of v_r and radar frequency band.

4.2 Continuous Wave Radar (CW Radar):-

A block diagram of simple CW radar is shown in Fig. 4.3. The transmitter generates a continuous (unmodulated) oscillation of frequency f_0, which is radiated by the antenna. A portion of the radiated energy is intercepted by the target and is scattered, some of it in the direction of the radar, where it is collected by the receiving antenna.

If the target is in motion with a velocity v_r relative to the radar, the received signal will be shifted in frequency from the transmitted frequency f_0 by an amount $\pm f_d$ as given by Eq. (4).

- The plus sign associated with the doppler frequency applies if the distance between target and radar is decreasing (closing target), that is, when the received signal frequency is greater than the transmitted signal frequency.
- The minus sign applies if the distance is increasing (receding target).

The received echo signal at a frequency $f_0 \pm f_d$ enters the radar via the antenna and is heterodyned in the detector (mixer) with a portion of the transmitter signal/o to produce a doppler beat note of frequency f_d The sign f_d is lost in this process.

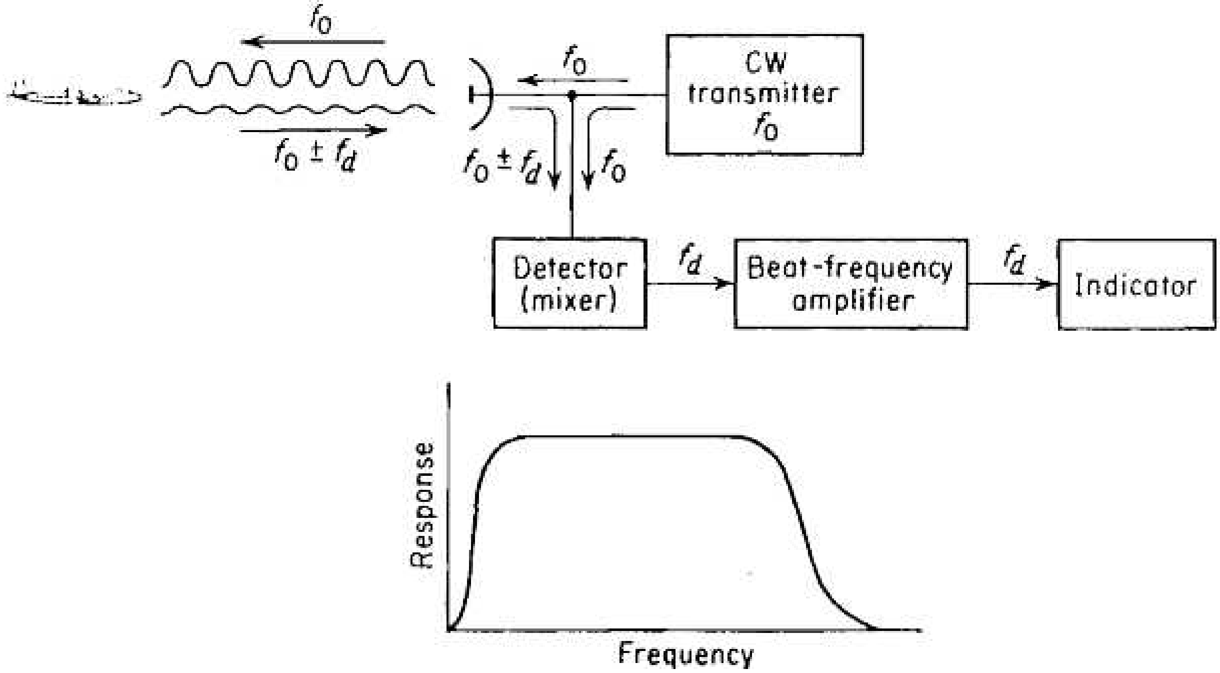


Fig. 4.3 CW Radar with frequency response

Pulse Radar: Pulse radar that extracts the Doppler frequency-shifted echo signal. A simple way to convert the CW radar to the pulse radar by turning on and off CW oscillator to generate pulses. This way of generation of pulses removes the reference signal, which is required to recognize the Doppler shifts. One way to introduce the reference signal is shown in fig. 4.4. Here the power amplifier is turned on and off to generate the high power pulses. The received echo signal is mixed with the output of CW oscillator, which acts as coherent reference to allow the recognition of any change in the frequency. Here coherent means that the transmitted pulses are synchronously used as reference signal. The change in frequency is detected through Doppler filter.

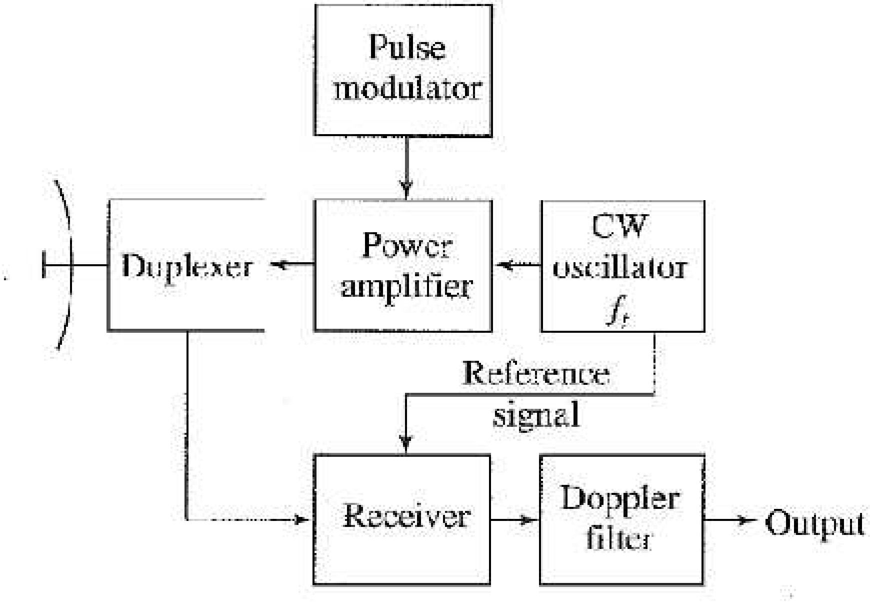


Fig. 4.4 Block diagram of simple Pulse Radar

Sweep to sweep subtraction:

The bipolar video (signal has positive and negative values) from two successive sweeps of MTI radar is shown in fig. 4.5. If one sweep is subtracted from the previous sweep, fixed clutter echoes will get cancel, and will not detected. On the other hand, moving target change its amplitude from sweep to sweep due the Doppler frequency shift. If one sweep is subtracted from other, the result will be canceled residue as shown in fig. 3.5.

Subtraction of the echoes from two successive sweeps is accomplished in delay line cancellers as shown in fig. 4.6. The delay-line canceller acts as a filter to eliminate the doc component of fixed targets and to pass the a-c components of moving targets. The video portion of the receiver is divided into two channels. One is a normal video channel. In the other, the video signal experiences a time delay equal to one pulse-repetition period (equal to the reciprocal of the pulse-repetition frequency). The outputs from the two channels are subtracted from one another. The fixed targets with unchanging amplitudes from pulse to pulse are canceled on subtraction.

However, the amplitudes of the moving-target echoes arc not constant from pulse 10 pulse, and subtraction results in an uncanceled residue.

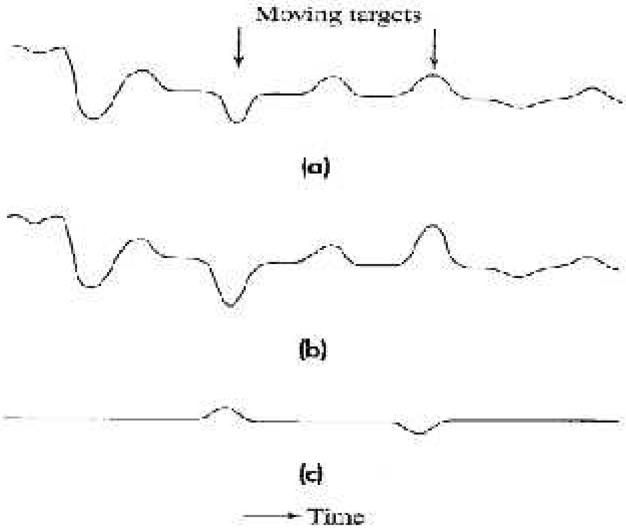


Fig. 4.5 Sweep to sweep subtraction

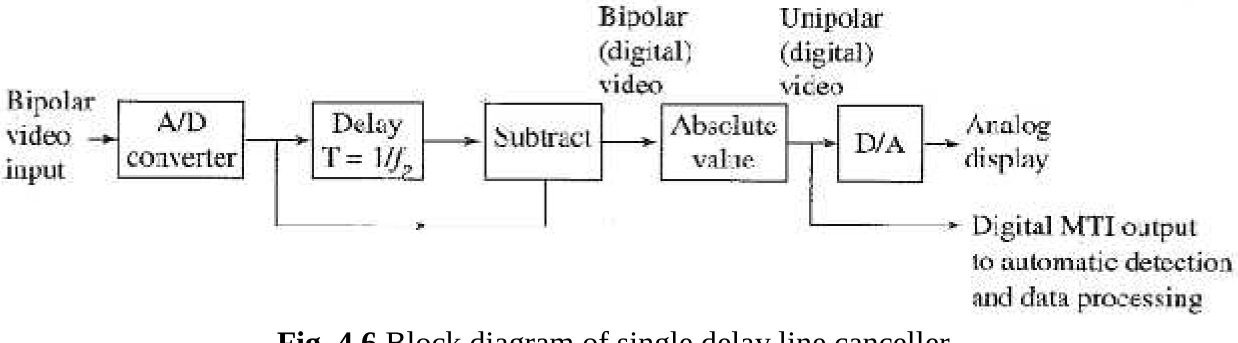


Fig. 4.6 Block diagram of single delay line canceller

MTI Radar Block Diagram:-

The doppler frequency shift [Eq. (3.2)] produced by a moving target may be used in a pulse radar. just as in the CW radar discussed in Chap. 3, to determine the relative velocity of a target or to separate desired moving targets from undesired stationary objects (clutter). Such a pulse radar that utilizes the doppler frequency shift as a means for discriminating moving from fixed targets is called an MTI (moving target indication) or a pulse doppler radar.

The block diagram of a more common MTI radar employing a power amplifier is shown in Fig. 4.5. The significant difference between this MTI configuration is the manner in which the reference signal is generated. In Fig. 4.7, the coherent reference is supplied by an oscillator called the coho, which stands for coherent oscillator.

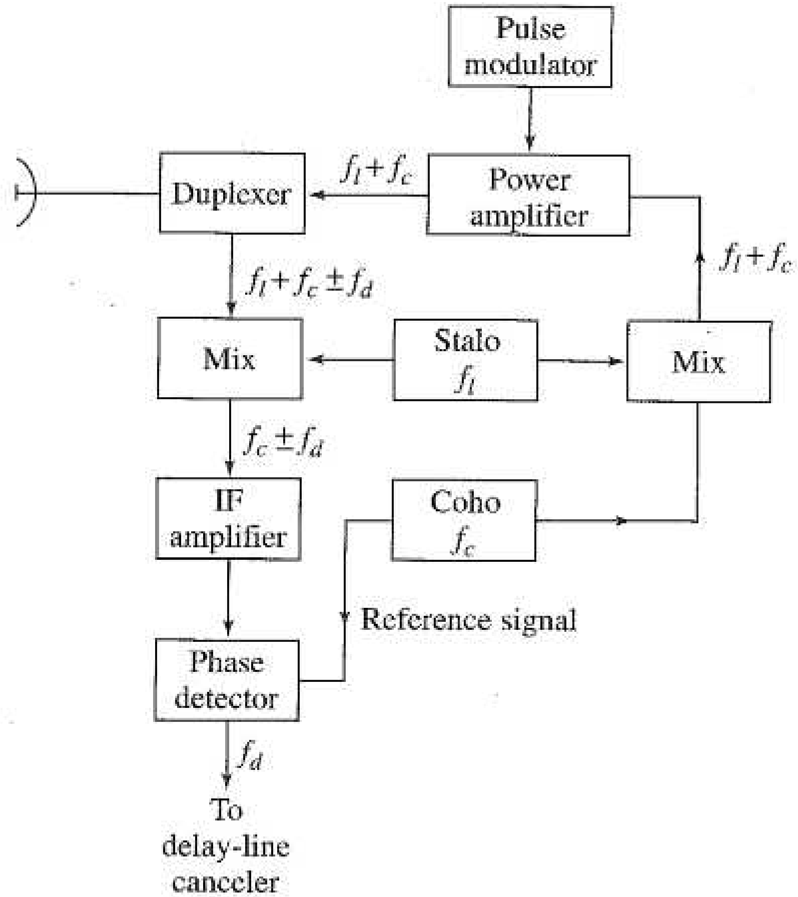


Fig. 4.7 Block diagram of MTI radar with power-amplifier transmitter

- The coho is a stable oscillator whose frequency is the same as the intermediate frequency used in the receiver. In addition to providing the refcrence signal. the output of the coho. fc is also mixed with the local-oscillator frequency fl.

- The local oscillator must also be a stable oscillator and is called stalo, for stable local oscillator.

- The stalo, coho, and the mixer in which they are combined plus any low-level amplification are called the receiver-exciter because of the dual role they serve in both the receiver and the transmitter.

- The characteristic feature of coherent MTI radar is that the transmitted signal must be coherent (in phase) with the reference signal in the receiver.

- The reference signal from the coho and the I F echo signal are both fed into a mixer called the phase detector. The phase detector differs from the normal amplitude detector since its output is proportional to the phase difference between the two input signals.

Delay Line Canceller:-

The simple MTI delay-line canceller shown in Fig. 4.6 is an example of a time-domain filter.The capability of this device depends on the quality of the medium used as the delay line. Thedelay line must introduce a time delay equal to the pulse repetition interval. For typical ground-based air-surveillance radars this might be several milliseconds. Delay times of this magnitude cannot be· achieved with practical electromagnetic transmission lines. By converting the electromagnetic signal to an acoustic signal it is possible to utilize delay lines of a delay line must introduce a time delay equal.

One of the advantages of a time-domain delay-line canceler as compared to the more conventional frequency-domain filter is that a single network operates at all ranges and does not require a separate filter for each range resolution cell. Frequency-domain doppler filter banks are of interest in some forms of MTI and pulse-doppler radar.

Frequency Response of Delay Line canceller

The delay-line canceler acts as a filter which rejects the d-c component of clutter. Because of its periodic nature, the filter also rejects energy in the vicinity of the pulse repetition frequency and its harmonics.

The signal from a target at range R_0, the output of the phase detector can be given as:

$$V_1 = k\sin(2® f_d t - ¤_0) \qquad \ldots (5)$$

Where f_d is Doppler frequency, $¤_0$ constant phase of $4® R_0 / ª$. The signal from the previous radar transmission is similar, which is delayed by time T_P

$$V_2 = k\sin[2® f_d (t - T_P) - ¤_0] \qquad \ldots (6)$$

Everything else is assumed to remain essentially constant over the interval T_P so that k is the same for both pulses. The output from the subtractor is

$$V = V_1 - V_2 = 2k\sin(® f_d T_P)\cos\left[2® f_d (t - \frac{T_P}{2}) - ¤_0\right] \qquad \ldots. (7)$$

The magnitude of the relative frequency-response of the delay-line canceler [ratio of the amplitude of the output from the delay-line canceler, $2k\sin(® f_d T_P)$, to the amplitude of the normal radar video k] is shown in Fig. 4.8.

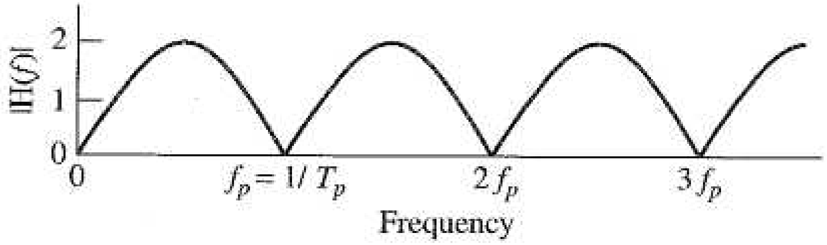


Fig. 4.8 Frequency response of the single delay-line canceller; T = delay time $=1/f_p$

Blind Speed:-

The response of the single-delay-line canceler will be zero whenever the argument $® f_d T_P$ in the amplitude factor of Eq. (7) is 0, π, 2 π, ... , etc., or when

$$f_d = \frac{2V_r}{ª} = \frac{n}{T_P} = nf_p \quad n = 0,1,2,3,...... \qquad \ldots (8)$$

The delay-line cancela not only eliminates the d-c component caused by clutter (n = 0), but unfortunately it also rejects any moving target whose doppler frequency happens to be the same as the prf or a multiple

there of. Those relative target velocities which result in zero MTI response are called blind speed and can be given as:

$$v_n = \frac{nª}{2T_p} = \frac{nª f_p}{2} \quad n = 0,1,2,3,...... \qquad \ldots (9)$$

where v_n is the nth blind speed. If λ is measured in meters,fp in Hz, and the relative velocity in knots, the blind speeds are

$$v_n = \frac{n^{a} f_p}{1.02} \approx n^{a} f_p \qquad \ldots (10)$$

The blind speeds are one of the limitations of pulse MTI radar which do not occur with CW radar. They are present in pulse radar because doppler is measured by discrete samples (pulses) at the prf rather than continuously.

Pulse Doppler Radar:-

A pulse radar that extracts the doppler frequency shift for the purpose of detecting moving targets in the presence of clutter is either an MTI radar or a pulse doppler radar.

The distinction between them is based on the fact that in a sampled measurement system like a pulse radar, ambiguities can arise in both the doppler frequency (relative velocity) and the range (time delay) measurements. Range ambiguities are avoided with a low sampling rate (low pulse repetition frequency), and doppler frequency ambiguities are avoided with a high sampling rate. However, in most radar applications the sampling rate, or pulse repetition frequency, cannot be selected to avoid both types of measurement ambiguities.

The pulse doppler radar is more likely to use range-gated doppler filter-banks than delay-line cancelers. Also, a power amplifier such as a klystron is more likely to be used than a delay-line cancelers. A pulse doppler radar operates at a higher duty cycle than does an MTI. Although it is difficult to generalize, the MTI radar seems to be the more widely used of the two, but pulse doppler is usually more capable of reducing clutter. .

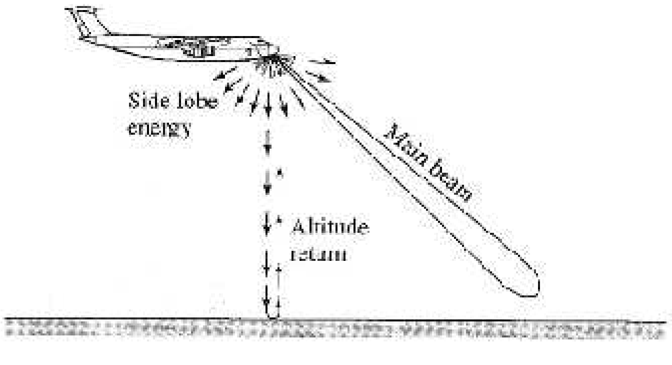
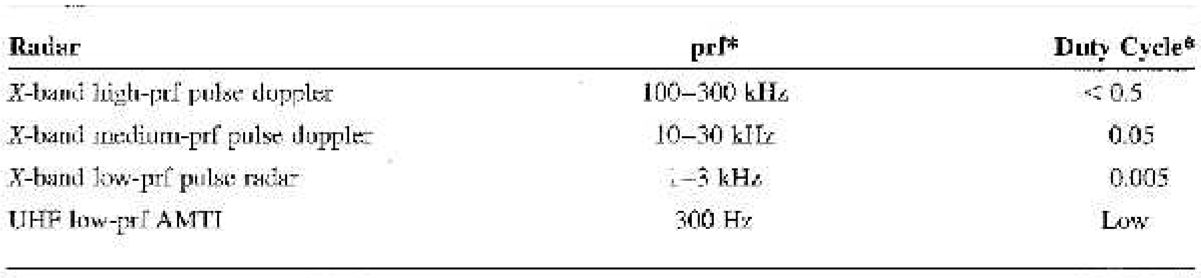


Fig. 4.9 Sketch of airborne Pulse Doppler radar

- A radar that increases its prf high enough to avoid the problems of blind speeds is called as Pulse radar.
- A high-prf pulase Doppler radar is one with no blind speeds with in the Doppler space.
- A medium-prf pulase Doppler radar is one get operated at slightly lower prf and accepts both range and Doppler ambiguities.
- A brief comparison between different Doppler pulse radar is given in table 4.1

Table. 4.1:- Comparison of different pulse Doppler radar



Radar	prf*	Duty Cycle*
X-band high-prf pulse doppler	100–300 kHz	< 0.5
X-band medium-prf pulse doppler	10–30 kHz	0.05
X-band low-prf pulse radar	1–3 kHz	0.005
UHF low-prf AMTI	300 Hz	Low

References

1- www.wikipedia.com

2- Introduction to Radar Systems by Merrill I. Skolnik, 3rd Edition, PHI Publications.

MOBILE COMMUNICATION

Module 1

Introduction

Communication is one of the integral parts of science that has always been a focus point for exchanging information among parties at locations physically apart. After its discovery, telephones have replaced the telegrams and letters. Similarly, the term `mobile' has completely revolutionized the communication by opening up innovative applications that are limited to one's imagination. Today, mobile communication has become the backbone of the society. All the mobile system technologies have improved the way of living. Its main plus point is that it has privileged a common mass of society. In this chapter, the evolution as well as the fundamental techniques of the mobile communication is discussed. The first wireline telephone system was introduced in the year 1877. Mobile communication systems as early as 1934 were based on Amplitude Modulation (AM) schemes and only certain public organizations maintained such systems. With the demand for newer and better mobile radio communication systems during the World War II and the development of Frequency Modulation (FM) technique by Edwin Armstrong, the mobile radio communication systems began to witness many new changes. Mobile telephone was introduced in the year 1946. However, during its initial three and a half decades it found very less market penetration owing to high costs and numerous technological drawbacks. But with the development of the cellular concept in the 1960s at the Bell Laboratories, mobile communications began to be a promising field of expanse which could serve wider populations. Initially, mobile communication was restricted to certain official users and the cellular concept was never even dreamt of being made commercially available. Moreover, even the growth in the cellular networks was very slow. However, with the development of newer and better technologies starting from the 1970s and with the mobile users now connected to the Public Switched Telephone Network (PSTN), there has been an astronomical growth in the cellular radio and the personal communication systems. Advanced Mobile Phone System (AMPS) was the first U.S. cellular telephone system and it was deployed in 1983. Wireless services have since then been experiencing a 50% per year growth rate. The number of cellular telephone users grew from 25000 in 1984 to around 3 billion in the year 2007 and the demand rate is increasing day by Day.

Mobile Telephony Developement

The first wireline telephone system was introduced in the year 1877. Mobile communication systems as early as 1934 were based on Amplitude Modulation (AM) schemes and only certain public organizations maintained such systems. With the demand for newer and better mobile radio communication systems during the World War II and the development of Frequency Modulation (FM) technique by Edwin Armstrong, the mobile radio communication systems began to witness many new changes. Mobile telephone was introduced in the year 1946. However, during its initial three and a half decades it found very less market penetration owing to high 1 Figure The worldwide mobile subscriber chart. costs and numerous technological drawbacks. But with the development of the cellular concept in the 1960s at the Bell Laboratories, mobile communications began to be a promising field of expanse which could serve wider populations. Initially, mobile communication was restricted to certain official users and the cellular concept was never

even dreamt of being made commercially available. Moreover, even the growth in the cellular networks was very slow. However, with the development of newer and better technologies starting from the 1970s and with the mobile users now connected to the Public Switched Telephone Network (PSTN), there has been an astronomical growth in the cellular radio and the personal communication systems. Advanced Mobile Phone System (AMPS) was the first U.S. cellular telephone system and it was deployed in 1983. Wireless services have since then been experiencing a 50% per year growth rate. The number of cellular telephone users grew from 25000 in 1984 to around 3 billion in the year 2007 and the demand rate is increasing day by
day. A schematic of the subscribers is shown in Fig.

Cellular Concept:

The power of the radio signals transmitted by the BS decay as the signals travel away from it. A minimum amount of signal strength (let us say, x dB) is needed in order to be detected by the MS or mobile sets which may the hand-held personal units or those installed in the vehicles. The region over which the signal strength lies above this threshold value x dB is known as the coverage area of a BS and it must be a circular region, considering the BS to be isotropic radiator. Such a circle, which gives this actual radio coverage, is called the foot print of a cell (in reality, it is amorphous). It might so happen that either there may be an overlap between any two such side by side circles or there might be a gap between the coverage areas of two adjacent circles. This is shown in Figure 1. Such a circular geometry, therefore, cannot serve as a regular shape to describe cells. We need a regular shape for cellular design over a territory which can be served by 3 regular polygons, namely, equilateral triangle, square and regular hexagon, which can cover the entire area without any overlap and gaps. Along with its regularity, a cell must be designed such that it is most reliable too, i.e., it supports even the weakest mobile with occurs at the edges of the cell. For any distance between the center and the farthest point in the cell from it, a regular hexagon covers the maximum area. Hence regular hexagonal geometry is used as the cells in mobile communication.

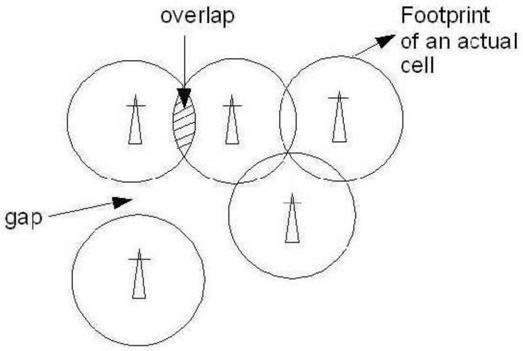


Fig 1:Footprint of cells showing the overlaps and gaps.

Frequency Reuse:

Frequency reuse, or, frequency planning, is a technique of reusing frequencies and channels within a communication system to improve capacity and spectral efficiency. Frequency reuse is one of the fundamental concepts on which commercial wireless systems are based that involve the partitioning of an RF radiating area into cells. The increased capacity in a commercial wireless network, compared with a network with a single transmitter, comes from the fact that the same radio frequency can be reused in a different area for a completely different transmission. Frequency reuse in mobile cellular systems means that frequencies allocated to

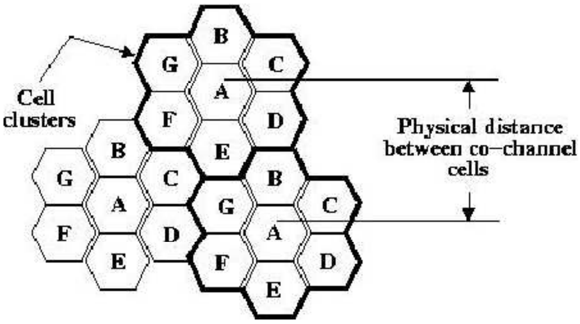


Fig 2: Frequency reuse technique of a cellular system.

the services are reused in a regular pattern of cells, each covered by one base station. The repeating regular pattern of cells is called cluster. Since each cell is designed to use radio frequencies only within its boundaries, the same frequencies can be reused in other cells not far away without interference, in another cluster. Such cells are called `co-channel' cells. The reuse of frequencies enables a cellular system to handle a huge number of calls with a limited number of channels. Figure 3.2 shows a frequency planning with cluster size of 7, showing the co-channels cells in different clusters by the same letter. The closest distance between the co-channel cells (in different clusters) is determined by the choice of the cluster size and the layout of the cell cluster. Consider a cellular system with S duplex channels available for use and let N be the number of cells in a cluster. If each cell is allotted K duplex channels with all being allotted unique and disjoint channel groups we have S = KN under normal circumstances. Now, if the cluster are repeated M times within the total area, the total number of duplex channels, or, the total number of users in the system would be T = MS = KMN. Clearly, if K and N remain constant, then

$T \propto M$ and, if T and K remain constant, then

$$N \propto \frac{1}{M}.$$

Hence the capacity gain achieved is directly proportional to the number of times a cluster is repeated, as shown in (3.1), as well as, for a fixed cell size, small N =25 decreases the size of the cluster with in turn results in the increase of the number of clusters and hence the capacity. However for small N, co-channel cells are located much closer and hence more interference. The value of N is determined by calculating the amount of interference that can be tolerated for a sufficient quality communication. Hence the smallest N having interference below the tolerated limit is used. However, the cluster size N cannot take on any value and is given only by the following equation

$$N = i^2 + ij + j^2, \qquad i \geq 0, j \geq 0,$$

Where i and j are integer numbers.

Channel Assignment Strategies

With the rapid increase in number of mobile users, the mobile service providers had to follow strategies which ensure the effective utilization of the limited radio spectrum. With increased capacity and low interference being the prime objectives, a frequency reuse scheme was helpful in achieving these objectives. A variety of channel assignment strategies have been followed to aid these objectives. Channel assignment strategies are classified into two types: fixed and dynamic, as discussed below.

Fixed Channel Assignment (FCA)

In fixed channel assignment strategy each cell is allocated a fixed number of voice channels. Any communication within the cell can only be made with the designated unused channels of that particular cell. Suppose if all the channels are occupied, then the call is blocked and subscriber has to wait. This is simplest of the channel assignment strategies as it requires very simple circuitry but provides worst channel utilization. Later there was another approach in which the channels were borrowed from adjacent cell if all of its own designated channels were occupied. This was named as borrowing strategy. In such cases the MSC supervises the borrowing process and ensures that none of the calls in progress are interrupted.

Dynamic Channel Assignment (DCA)

In dynamic channel assignment strategy channels are temporarily assigned for use in cells for the duration of the call. Each time a call attempt is made from a cell the corresponding BS requests a channel from MSC. The MSC then allocates a channel to the requesting the BS. After the call is over the channel is returned and kept in a central pool. To avoid co-channel interference any channel that in use in one cell can only be reassigned simultaneously to another cell in the system if the distance between the two cells is larger than minimum reuse distance. When compared to the FCA, DCA has reduced the likelihood of blocking and even increased the trunking capacity of the network as all of the channels are available to all cells, i.e., good quality of service. But this type of assignment strategy results in heavy load on switching center at heavy traffic condition.

Handoff Process

When a user moves from one cell to the other, to keep the communication between the user pair, the user channel has to be shifted from one BS to the other without interrupting the call, i.e., when a MS moves into another cell, while the conversation is still in progress, the MSC automatically transfers the call to a new FDD channel without disturbing the conversation. This process is called as handoff. A schematic diagram of handoff is given in Figure Processing of handoff is an important task in any cellular system. Handoffs must be performed successfully and be imperceptible to the users. Once a signal level is set as the minimum acceptable for good voice quality (Prmin), then a slightly stronger level is chosen as the threshold (PrH)at which handoff has to be made, as shown in Figure 3.4. A parameter, called power margin, defined as

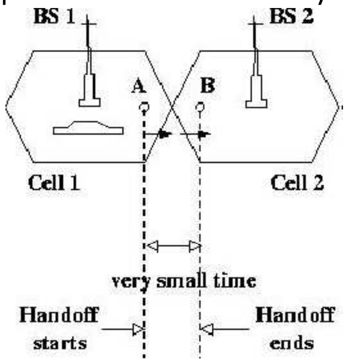


Fig 3:Handoff scenario at two adjacent cell boundary.

$$\Delta = P_{r_H} \qquad P_{r_{min}}$$

is quite an important parameter during the handoff process since this margin can neither be too large nor too small. If Δis too small, then there may not be enough time to complete the handoff and the call might be lost even if the user crosses the cell boundary. If Δ is too high o the other hand, then MSC has to be burdened with unnecessary handoffs. This is because MS may not intend to enter the other cell. Therefore Δ should be judiciously chosen to ensure imperceptible handoffs and to meet other objectives

Interference & System Capacity

Susceptibility and interference problems associated with mobile communications equipment are because of the problem of time congestion within the electromagnetic spectrum. It is the limiting factor in the performance of cellular systems. This interference can occur from clash with another mobile in the same cell or because of a call in the adjacent cell. There can be interference between the base stations operating at same frequency band or any other non-cellular system's energy leaking inadvertently into the frequency band of the cellular system. If there is an interference in the voice channels, cross talk is heard will appear as noise between the users. The interference in the control channels leads to missed and error calls because of digital signaling. Interference is more severe in urban areas because of the greater RF noise and greater density of mobiles and base stations. The interference can be
divided into 2 parts: co-channel interference and adjacent channel interference.

Co-channel interference (CCI)

For the efficient use of available spectrum, it is necessary to reuse frequency bandwidth over relatively small geographical areas. However, increasing frequency reuse also increases interference, which decreases system capacity and service quality. The cells where the same set of frequencies is used are call co-channel cells. Co-channel interference is the cross talk between two different radio transmitters using the same radio frequency as is the case with the co-channel cells. The reasons of CCI can be because of either adverse weather conditions or poor frequency planning or overly crowded radio spectrum. If the cell size and the power transmitted at the base stations are same then CCI will become independent of the transmitted power and will depend on radius of the cell (R) and the distance between the interfering co-channel cells (D). If D/R ratio is increased, then the effective distance between the co-channel cells will increase 34 and interference will decrease. The parameter Q is called the frequency reuse ratio and is related to the cluster size. For hexagonal geometry

$$Q - D/R - \sqrt{3N}$$

From the above equation, small of `Q' means small value of cluster size `N' and increase in cellular capacity. But large `Q' leads to decrease in system capacity but increase in transmission quality. Choosing the options is very careful for the selection of `N', the proof of which is given in the first section. The Signal to Interference Ratio (SIR) for a mobile receiver which monitors the forward channel can be calculated as

$$\frac{S}{I} = \frac{S}{\sum_{i=1}^{i_0} I_i}$$

where i0 is the number of co-channel interfering cells, S is the desired signal power from the baseband station and Ii is the interference power caused by the i-th interfering co-channel base station. In order to solve this equation from power calculations, we need to look into the signal power characteristics. The average power in the mobile radio channel decays as a power law of the distance of separation between transmitter and receiver. The expression for the received power Pr at a distance d can be approximately calculated as

$$P_r = P_0(\frac{d}{d_0})^{-n}$$

and in the dB expression as

$$P_r(dB) = P_0(dB) - 10n\log(\frac{d}{d_0})$$

where P0 is the power received at a close-in reference point in the far field region at a small distance do from the transmitting antenna, and `n' is the path loss exponent. Let us calculate the SIR for this system. If Di is the distance of the i-th interferer from the mobile, the received power at a given mobile due to i-th interfering cell is proportional to (Di)□n (the value of 'n' varies between 2 and 4 in urban cellular systems). Let us take that the path loss exponent is same throughout the coverage area and the transmitted power be same, then SIR can be approximated as

$$\frac{S}{I} - \frac{R^{\ n}}{\sum_{i=1}^{i_0} D_i^{-n}}$$

where the mobile is assumed to be located at R distance from the cell center. If we consider only the first layer of interfering cells and we assume that the interfering base stations are equidistant from the reference base station and the distance between the cell centers is 'D' then the above equation can be converted as

$$\frac{S}{I} = \frac{(D/R)^n}{i_0} = \frac{(\sqrt{3N})^n}{i_0}$$

which is an approximate measure of the SIR. Subjective tests performed on AMPS cellular system which uses FM and 30 kHz channels show that sufficient voice quality can be obtained by SIR being greater than or equal to 18 dB. If we take n=4 , the value of 'N' can be calculated as 6.49. Therefore minimum N is 7. The above equations are based on hexagonal geometry and the distances from the closest interfering cells can vary if different frequency reuse plans are used. We can go for a more approximate calculation for co-channel SIR. This is the example of a 7 cell reuse case. The mobile is at a distance of D-R from 2 closest interfering cells and approximately D+R/2, D, D-R/2 and D+R distance from other interfering cells in the first tier. Taking n = 4 in the above equation, SIR can be approximately calculated as

$$\frac{S}{I} - \frac{R^{-4}}{2(D-R)^{-4} + (D+R)^{-4} + (D)^{-4} + (D+R/2)^{-4} + (D-R/2)^{-4}}$$

which can be rewritten in terms frequency reuse ratio Q as

$$\frac{S}{I} = \frac{1}{2(Q-1)^{-4} + (Q+1)^{-4} + (Q)^{-4} + (Q+1/2)^{-4} + (Q-1/2)^{-4}}$$

Using the value of N equal to 7 (this means Q = 4.6), the above expression yields that worst case SIR is 53.70 (17.3 dB). This shows that for a 7 cell reuse case the worst case SIR is slightly less than 18 dB. The worst case is when the mobile is at the corner of the cell i.e., on a vertex as shown in the Figure 3.6. Therefore N = 12 cluster size should be used. But this reduces the capacity by 7/12 times. Therefore, co-channel interference controls link performance, which in a way controls frequency reuse plan and the overall capacity of the cellular system. The effect of co-channel interference can be minimized by optimizing the frequency assignments of the base stations and their transmit powers. Tilting the base-station antenna to limit the spread of the signals in the system can also be done.

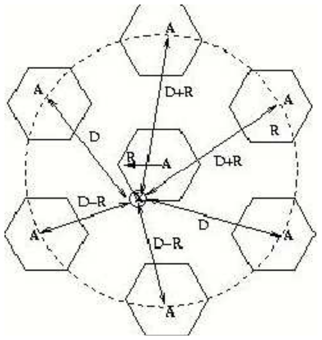


Fig 4:First tier of co-channel interfering cells

Adjacent Channel Interference (ACI)

This is a different type of interference which is caused by adjacent channels i.e. channels in adjacent cells. It is the signal impairment which occurs to one frequency due to presence of another signal on a nearby frequency. This occurs when imperfect receiver filters allow nearby frequencies to leak into the pass band. This problem is enhanced if the adjacent channel user is transmitting in a close range compared to the subscriber's receiver while the receiver attempts to receive a base station on the channel. This is called near-far effect. The more adjacent channels are packed into the channel block, the higher the spectral efficiency, provided that the performance degradation can be tolerated in the system link budget. This effect can also occur if a mobile close to a base station transmits on a channel close to one being used by a weak mobile. This problem might occur if the base station has problem in discriminating the mobile user from the "bleed over" caused by the close adjacent channel mobile. Adjacent channel interference occurs more frequently in small cell clusters and heavily used cells. If the frequency separation between the channels is kept large this interference can be reduced to some extent. Thus assignment of channels is given such that they do not form a contiguous band of frequencies within a particular cell and frequency separation is maximized. Efficient assignment strategies are very much important in making the interference as less as possible. If the frequency factor is small then distance between the adjacent channels cannot put the interference level within tolerance limits. If a mobile is 10 times close to the base station than other mobile and has energy spill out of its pass band, then SIR for weak mobile is approximately

$$\frac{S}{I} = 10^{-n}$$

which can be easily found from the earlier SIR expressions. If n = 4, then SIR is 52 dB. Perfect base station filters are needed when close-in and distant users share the same cell. Practically, each base station receiver is preceded by a high Q cavity filter in order to remove adjacent channel interference. Power control is also very much important for the prolonging of the battery life for the subscriber unit but also reduces reverse channel SIR in the system.

Power control is done such that each mobile transmits the lowest power required to maintain a good quality link on the reverse channel.

Cell-Splitting

Cell Splitting is based on the cell radius reduction and minimizes the need to modify the existing cell parameters. Cell splitting involves the process of sub-dividing a congested cell into smaller cells, each with its own base station and a corresponding reduction in antenna size and transmitting power. This increases the capacity of a cellular system since it increases the number of times that channels are reused. Since the new cells have smaller radii than the existing cells, inserting these smaller cells, known as microcells, between the already existing cells results in an increase of capacity due to the additional number of channels per unit area. There are few challenges in increasing the capacity by reducing the cell radius. Clearly, if cells are small, there would have to be more of them and so additional base stations will be needed in the system. The challenge in this case is to introduce the new base stations without the need to move the already existing base station towers. The other challenge is to meet the generally increasing demand that may vary quite rapidly between geographical areas of the system. For instance, a city may have highly populated areas and so the demand must be supported by cells with the smallest radius. The radius of cells will generally increase as we move from urban to sub urban areas, because the user density decreases on moving towards sub-urban areas. The key factor is to add as minimum number of smaller cells as possible wherever an increase in demand occurs. The gradual addition of the smaller cells implies that, at least for a time, the cellular system operates with cells of more than one size. Figure 5 shows a cellular layout with seven-cell clusters. Consider that the cells in the center of the diagram are becoming congested, and cell A in the center has reached its maximum capacity. Figure also shows how the smaller cells are being superimposed on the original layout. The new smaller cells have half the cell radius of the original cells. At half the radius, the new cells will have one-fourth of the area and will consequently need to support one-fourth the number of subscribers. Notice that one of the new smaller cells lies in the center of each of the larger cells. If we assume that base stations are located in the cell centers, this allows the original base stations to be maintained even in the new system layout. However, new base stations will have to be added for new cells that do not lie in the center of the larger cells. The organization of cells into clusters is independent of the cell radius, so that the cluster size can be the same in the small-cell layout as it was in the large-cell layout.

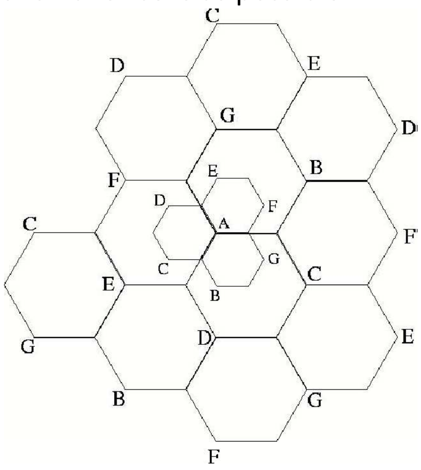


Fig 5:Splitting of congested seven-cell clusters.

Also the signal-to-interference ratio is determined by cluster size and not by cell radius. Consequently, if the cluster size is maintained, the signal-to-interference ratio will be the same after cell splitting as it was before. If the entire system is 41 replaced with new half-radius cells, and the cluster size is maintained, the number of channels per cell will be exactly as it was before, and the number of subscribers per cell will have been reduced. When the cell radius is reduced by a factor, it is also desirable to reduce the transmitted power. The transmit power of the new cells with radius half that of the old cells can be found by examining the received power PR at the new and old cell boundaries and setting them equal. This is necessary to maintain the same frequency re-use plan in the new cell layout as well. Assume that PT1 and PT2 are the transmit powers of the larger and smaller base stations respectively. Then, assuming a path loss index n=4, we have power received at old cell boundary = PT1=R4 and the power received at new cell boundary = PT2=(R=2)4. On equating the two received powers, we get PT2 = PT1 / 16. In other words, the transmit power must be reduced by 12 dB in order to maintain the same S/I with the new system lay-out. At the beginning of this channel splitting process, there would be fewer channels in the smaller power groups. As the demand increases, more and more channels need to be accommodated and hence the splitting process continues until all the larger cells have been replaced by the smaller cells, at which point splitting is complete within the region and the entire system is rescaled to have a smaller radius per cell. If a cellular layout is replaced entirety by a new layout with a smaller cell radius, the signal-to-interference ratio will not change, provided the cluster size does not change. Some special care must be taken, however, to avoid co-channel interference when both large and small cell radii coexist. It turns out that the only way to avoid interference between the large-cell and small-cell systems is to assign entirely different sets of channels to the two systems. So, when two sizes of cells co-exist in a system, channels in the old cell must be broken down into two groups, one that corresponds to larger cell reuse requirements and the other which corresponds to the smaller cell reuse requirements. The larger cell is usually dedicated to high speed users as in the umbrella cell approach so as to minimize the number of hand-offs.

Sectoring

Sectoring is basically a technique which can increase the SIR without necessitating an increase in the cluster size. Till now, it has been assumed that the base station is located in the center of a cell and radiates uniformly in all the directions behaving as an omni-directional antenna. However it has been found that the co-channel interference in a cellular system may be decreased by replacing a single omni-directional antenna at the base station by several directional antennas, each radiating within a specified sector. In the Figure 3.8, a cell is shown which has been split into three 120o sectors. The base station feeds three 120o directional antennas, each of which radiates into one of the three sectors. The channel set serving this cell has also been divided, so that each sector is assigned one-third of the available number cell of channels. This technique for reducing co-channel interference wherein by using suit-

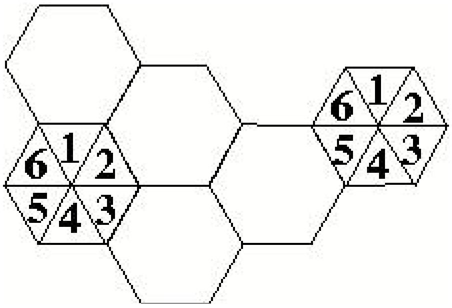


Fig 6:A seven-cell cluster with 60o sectors

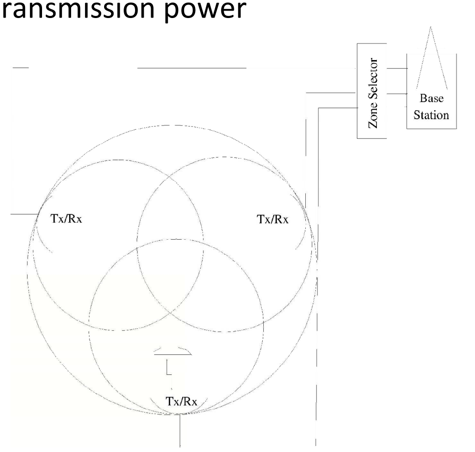
able directional antennas, a given cell would receive interference and transmit with a fraction of available co-channel cells is called 'sectoring'. In a seven-cell-cluster layout with 120o sectored cells, it can be easily understood that the mobile units in a particular sector of the center cell will receive co-channel interference from only two of the first-tier co-channel base stations, rather than from all six. Likewise, the base station in the center cell will receive co-channel interference from mobile units in only two of the co-channel cells. Hence the signal to interference ratio is now modified to

$$\frac{S}{I} = \frac{(\surd 3N)^n}{2}$$

where the denominator has been reduced from 6 to 2 to account for the reduced number of interfering sources. Now, the signal to interference ratio for a seven-cell cluster layout using 120o sectored antennas can be found from equation above to be 23.4 dB which is a significant improvement over the Omni-directional case where the worst-case S/I is found to be 17 dB (assuming a path-loss exponent, n=4). Some cellular systems divide the cells into 60o sectors. Similar analysis can be performed on them as well.

Microcell Zone Concept:

The increased number of handoffs required when sectoring is employed results in an increased load on the switching and control link elements of the mobile system. To overcome this problem, a new microcell zone concept has been proposed. As shown in Figure 7 this scheme has a cell divided into three microcell zones, with each of the three zone sites connected to the base station and sharing the same radio equipment. It is necessary to note that all the microcell zones, within a cell, use the same frequency used by that cell; that is no handovers occur between microcells. Thus when a mobile user moves between two microcell zones of the cell, the BS simply switches the channel to a different zone site and no physical re-allotment of channel takes place. Locating the mobile unit within the cell: An active mobile unit sends a signal to all zone sites, which in turn send a signal to the BS. A zone selector at the BS uses that signal to select a suitable zone to serve the mobile unit - choosing the zone with the strongest signal. Base Station Signals: When a call is made to a cellular phone, the system already knows the cell location of that phone. The base station of that cell knows in which zone, within that cell, the cellular phone is located. Therefore when it receives the signal, the base station transmits it to the suitable zone site. The zone site receives the cellular signal from the base station and transmits that signal to the mobile phone after amplification. By confining the power transmitted to the mobile phone, co-channel interference is reduced between the zones and the capacity of system is increased. Benefits of the micro-cell zone concept: 1) Interference is reduced in this case as compared to the scheme in which the cell size is reduced. 2) Handoffs are reduced (also compared to decreasing the cell size) since the microcells within the cell operate at the same frequency; no handover occurs when the mobile unit moves between the microcells. 3) Size of the zone apparatus is small. The zone site equipment being small can be mounted on the side of a building or on poles. 4) System capacity is increased. The new microcell knows where to locate the mobile unit in a particular zone of the cell and deliver the power to that zone. Since the signal power is reduced, the microcells can be closer and result in an increased system capacity. However, in a microcellular system, the transmitted power to a mobile phone within a microcell has to be precise; too much power results in interference between microcells, while with too little power the signal might not reach the mobile phone. This is a drawback of microcellular systems, since a change in the surrounding (a new building, say, within a microcell) will require a change of the transmission power



.

Fig 7: The micro-cell zone concept

Trunked Radio System

In the previous sections, we have discussed the frequency reuse plan, the design trade-offs and also explored certain capacity expansion techniques like cell-splitting and sectoring. Now, we look at the relation between the number of radio channels a cell contains and the number of users a cell can support. Cellular systems use the concept of trunking to accommodate a large number of users in a limited radio spectrum. It was found that a central office associated with say, 10,000 telephones 47 requires about 50 million connections to connect every possible pair of users. However, a worst case maximum of 5000 connections need to be made among these telephones at any given instant of time, as against the possible 50 million connections. In fact, only a few hundreds of lines are needed owing to the relatively short duration of a call. This indicates that the resources are shared so that the number of lines is much smaller than the number of possible connections. A line that connects switching offices and that is shared among users on an as-needed basis is called a trunk. The fact that the number of trunks needed to make connections between offices is much smaller than the maximum number that could be used suggests that at times there might not be sufficient facilities to allow a call to be completed. A call that cannot be completed owing to a lack of resources is said to be blocked. So one important to be answered in mobile cellular systems is: How many channels per cell are needed in a cellular telephone system to ensure a reasonably low probability that a call will be blocked?

In a trunked radio system, a channel is allotted on per call basis. The performance of a radio system can be estimated in a way by looking at how efficiently the calls are getting connected and also how they are being maintained at handoffs. Some of the important factors to take into consideration are (i) Arrival statistics, (ii)Service statistics, (iii)Number of servers/channels.

Module 2

Mobile Radio Propagation :

There are two basic ways of transmitting an electro-magnetic (EM) signal, through a guided medium or through an unguided medium. Guided mediums such as coaxial cables and fiber optic cables, are far less hostile toward the information carrying EM signal than the wireless or the unguided medium. It presents challenges and conditions which are unique for this kind of transmissions. A signal, as it travels through the wireless channel, undergoes many kinds of propagation effects such as refection, diffractions and scattering, due to the presence of buildings, mountains and other such obstructions. Refection occurs when the EM waves impinge on objects which are much greater than the wavelength of the traveling wave. Diffraction is a phenomena occurring when the wave interacts with a surface having sharp
irregularities. Scattering occurs when the medium through the wave is traveling contains objects which are much smaller than the wavelength of the EM wave. These varied phenomena's lead to large scale and small scale propagation losses. Due to the inherent randomness associated with such channels they are best described with the help of statistical models. Models which predict the mean signal strength for arbitrary transmitter receiver distances are termed as large scale propagation models. These are termed so because they predict the average signal strength for large Tx-Rx separations, typically for hundreds of kilometers.

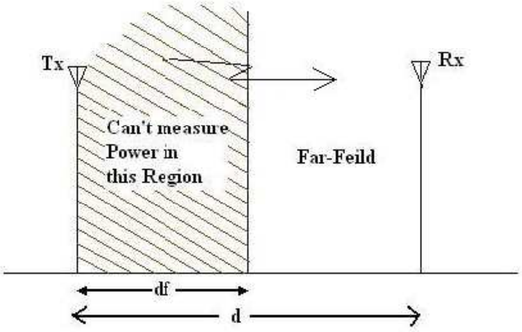


Fig 8: Free space propagation model, showing the near and far fields.

Basic Methods of Propagation:

Diffraction

Diffraction is the phenomenon due to which an EM wave can propagate beyond the horizon, around the curved earth's surface and obstructions like tall buildings. As the user moves deeper into the shadowed region, the received field strength decreases. But the diffractions field still exists an it has enough strength to yield a good signal. This phenomenon can be explained by the Huygen's principle, according to which, every point on a wave front acts as point sources for the production of secondary wavelets, and they combine to produce a new wave front in the direction of propagation. The propagation of secondary wavelets in the shadowed region results in diffractions. The field in the shadowed region is the vector sum of the electric field components of all the secondary wavelets that are received by the receiver.

Scattering

The actual received power at the receiver is somewhat stronger than claimed by the models of refection and diffractions. The cause is that the trees, buildings and lampposts scatter energy in all directions. This provides extra energy at the receiver. Roughness is tested by a Rayleigh criterion, which defines a critical height hc of surface protuberances for a given angle of incidence θi, given by,

$$h_c = \frac{\lambda}{8 sin\theta_i}.$$

A surface is smooth if its minimum to maximum protuberance h is less than hc,
and rough if protuberance is greater than hc. In case of rough surfaces, the surface refection coefficient needs to be multiplied by a scattering loss factor θS, given by

$$\rho s = exp(-8(\frac{\pi\sigma_h sin\theta_i}{\lambda})^2)$$

where Δh is the standard deviation of the Gaussian random variable h. The following result is a better approximation to the observed value

$$\rho s = exp(-8(\frac{\pi\sigma_h sin\theta_i}{\lambda})^2)I_0[-8(\frac{\pi\sigma_h sin\theta_i}{\lambda})^2]$$

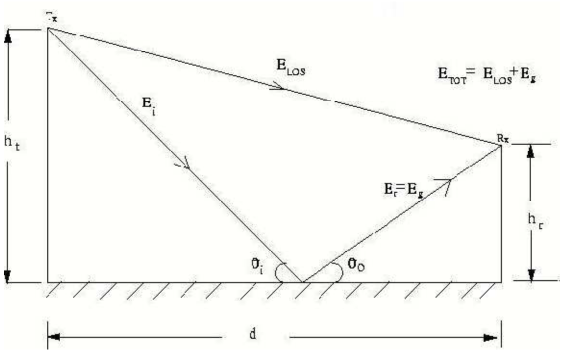


Fig 9: Two-ray refection model

which agrees very well for large walls made of limestone. The equivalent refection coefficient is given by,

$$\Gamma_{rough} = \rho_S \Gamma.$$

Outdoor Propagation Models

There are many empirical outdoor propagation models such as Longley-Rice model, Durkin's model, Okumura model, Hata model etc. Longley-Rice model is the most commonly used model within a frequency band of 40 MHz to 100 GHz over different terrains. Certain modifications over the rudimentary model like an extra urban factor (UF) due to urban clutter near the receiver is also included in this model. Below, we discuss some of the outdoor models, followed by a few indoor models too.

Okumura Model

The Okumura model is used for Urban Areas is a Radio propagation model that is used for signal prediction. The frequency coverage of this model is in the range of 200 MHz to 1900 MHz and distances of 1 Km to 100 Km .It can be applicable for base station effective antenna heights (ht) ranging from 30 m to 1000 m. Okumura used extensive measurements of base station-to-mobile signal attenuation throughout Tokyo to develop a set of curves giving median attenuation relative to free space (Amu) of signal propagation in irregular terrain. The empirical path loss formula of Okumura at distance d parameterized by the carrier frequency fc is given by

$$P_L(d)dB = L(f_c, d) + A_{mu}(f_c, d) - G(h_t) - G(h_r) - G_{AREA}$$

where L(fc; d) is free space path loss at distance d and carrier frequency fc, Amu(fc; d) is the median attenuation in addition to free-space path loss across all environments(ht) is the base station antenna height gain factor,G(hr) is the mobile antenna height gain factor,GAREA is the gain due to type of environment. The values of Amu(fc; d) and GAREA are obtained from Okumura's empirical plots. Okumura derived empirical formulas for G(ht) and G(hr) as follows:

$$G(h_t) = 20\log_{10}(h_t/200), \qquad 30m < h_t < 1000m$$

$$G(h_r) = 10\log_{10}(h_r/3), \qquad h_r \leq 3m$$

$$G(h_r) = 20\log_{10}(h_r/3), \qquad 3m < h_r < 10m$$

Correlation factors related to terrain are also developed in order to improve the models accuracy. Okumura's model has a 10-14 dB empirical standard deviation between the path loss predicted by the model and the path loss associated with one of the measurements used to develop the model.

Multipath & Small-Scale Fading

Multipath signals are received in a terrestrial environment, i.e., where diffierent forms of propagation are present and the signals arrive at the receiver from transmitter via a variety of paths. Therefore there would be multipath interference, causing multipath fading. Adding the effiect of movement of either Tx or Rx or the surrounding clutter to it, the received overall signal amplitude or phase changes over a small amount of time. Mainly this causes the fading.

Fading

The term fading, or, small-scale fading, means rapid uctuations of the amplitudes, phases, or multipath delays of a radio signal over a short period or short travel distance. This might be so severe that large scale radio propagation loss effects might be ignored.

Multipath Fading Effects

In principle, the following are the main multipath effects:

1. Rapid changes in signal strength over a small travel distance or time interval.
2. Random frequency modulation due to varying Doppler shifts on different multipath signals.
3. Time dispersion or echoes caused by multipath propagation delays.

Factors Influencing Fading

The following physical factors influence small-scale fading in the radio propagation channel:

(1) Multipath propagation { Multipath is the propagation phenomenon that results in radio signals reaching the receiving antenna by two or more paths. The effects of multipath include constructive and destructive interference, and phase shifting of the signal.

(2) Speed of the mobile { The relative motion between the base station and the mobile results in random frequency modulation due to different doppler shifts on each of the multipath components.

(3) Speed of surrounding objects { If objects in the radio channel are in motion, they induce a time varying Doppler shift on multipath components. If the surrounding objects move at a greater rate than the mobile, then this effect dominates fading.

(4) Transmission Bandwidth of the signal { If the transmitted radio signal bandwidth is greater than the \bandwidth" of the multipath channel (quanti- ffied by coherence bandwidth), the received signal will be distorted.

Types of Small-Scale Fading

The type of fading experienced by the signal through a mobile channel depends on the relation between the signal parameters (bandwidth, symbol period) and the channel parameters (rms delay spread and Doppler spread). Hence we have four different types of fading. There are two types of fading due to the time dispersive nature of the channel.

Fading Effects due to Multipath Time Delay Spread

Flat Fading

Such types of fading occurs when the bandwidth of the transmitted signal is less than the coherence bandwidth of the channel. Equivalently if the symbol period of the signal is more than the rms delay spread of the channel, then the fading is at fading. So we can say that at fading occurs when

$$B_S \ll B_C$$

where BS is the signal bandwidth and BC is the coherence bandwidth. Also

$$T_S \gg \sigma_\tau$$

where TS is the symbol period and σ_T is the rms delay spread. And in such a case, mobile channel has a constant gain and linear phase response over its bandwidth. Frequency Selective Fading Frequency selective fading occurs when the signal bandwidth is more than the coherence bandwidth of the mobile radio channel or equivalently the symbols duration of the signal is less than the rms delay spread.

$$B_S \gg B_C$$

$$T_S \ll \sigma_\tau$$

At the receiver, we obtain multiple copies of the transmitted signal, all attenuated and delayed in time. The channel introduces inter symbol interference. A rule of thumb for a channel to have at fading is if

$$\frac{\sigma_\tau}{T_S} \leq 0.1$$

Fading Effects due to Doppler Spread

Fast Fading

In a fast fading channel, the channel impulse response changes rapidly within the symbol duration of the signal. Due to Doppler spreading, signal undergoes frequency dispersion leading to distortion. Therefore a signal undergoes fast fading if

$$T_S \gg T_C$$

where TC is the coherence time and

$$B_S \gg B_D$$

where BD is the Doppler spread. Transmission involving very low data rates suffer from fast fading.

Slow Fading

In such a channel, the rate of the change of the channel impulse response is much less than the transmitted signal. We can consider a slow faded channel a channel in which channel is almost constant over at least one symbol duration. Hence

$$T_S \ll T_C$$

$$B_S \gg B_D$$

We observe that the velocity of the user plays an important role in deciding whether the signal experiences fast or slow fading.

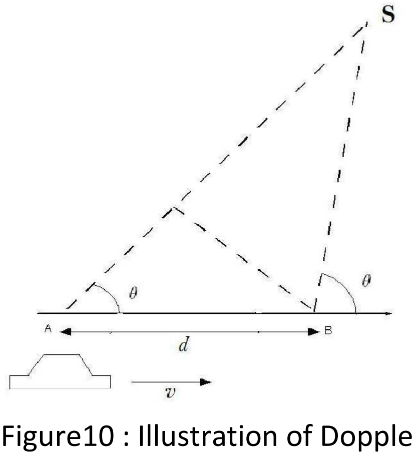


Figure10 : Illustration of Doppler effect.

Doppler Shift

The Doppler effect (or Doppler shift) is the change in frequency of a wave for an observer moving relative to the source of the wave. In classical physics (waves in a medium), the relationship between the observed frequency f and the emitted frequency fo is given by:

$$f = \left(\frac{v \pm v_r}{v \pm v_s}\right) f_0$$

where v is the velocity of waves in the medium, vs is the velocity of the source relative to the medium and vr is the velocity of the receiver relative to the medium. In mobile communication, the above equation can be slightly changed according to our convenience since the source (BS) is fixed and located at a remote elevated level from ground. The expected Doppler shift of the EM wave then comes out to be

$\pm \frac{v_r}{c} f_0$ or, $\pm \frac{v_r}{\lambda}$.

As the BS is located at an elevated place, a cos factor would also be multiplied with this. The exact scenario, as given in Figure10 , is illustrated below. Consider a mobile moving at a constant velocity v, along a path segment length d between points A and B, while it receives signals from a remote BS source S. The difference in path lengths traveled by the wave from source S to the mobile at points A and B is

$\Delta l = d \cos\theta = v \Delta t \cos\theta$, where Δt

Where Δt is the time required for the mobile

to travel from A to B, and ff is assumed to be the same at points A and B since the source is assumed to be very far away. The phase change in the received signal due to the difference in path lengths is therefore

$$\Delta\varphi = \frac{2\pi \Delta l}{\lambda} = \frac{2\pi v \Delta t}{\lambda} \cos\theta$$

and hence the apparent change in frequency, or Doppler shift (fd) is

$$f_d = \frac{1}{2\pi} . \frac{\Delta\varphi}{\Delta t} = \frac{v}{\lambda} . \cos\theta.$$

Different Modulation Techniques:

BPSK

In binary phase shift keying (BPSK), the phase of a constant amplitude carrier signal is switched between two values according to the two possible signals m1 and m2 corresponding to binary 1 and 0, respectively. Normally, the two phases are separated by 180o. If the sinusoidal carrier has an amplitude A, and energy per bit $E_b = \frac{1}{2} A_c^2 T_b$ then the transmitted BPSK signal is

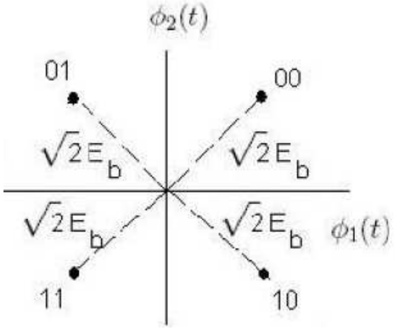
$$s_{BPSK}(t) = m(t) \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \theta_c).$$

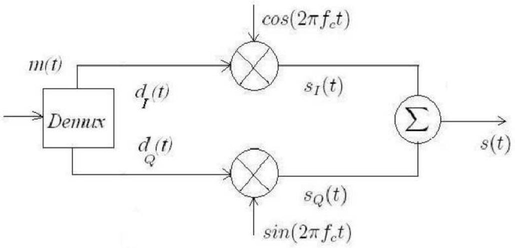
A typical BPSK signal constellation diagram is shown in Figure. The probability of bit error for many modulation schemes in an AWGN channel is found using the Q-function of the distance between the signal points. In case of BPSK,

$$P_{eBPSK} = Q(\sqrt{\frac{2E_b}{N_0}}).$$

QPSK

The Quadrature Phase Shift Keying (QPSK) is a 4-ary PSK signal. The phase of the carrier in the QPSK takes 1 of 4 equally spaced shifts. Although QPSK can be viewed as a quaternary modulation, it is easier to see it as two independently modulated quadrature carriers. With this interpretation, the even (or odd) bits are





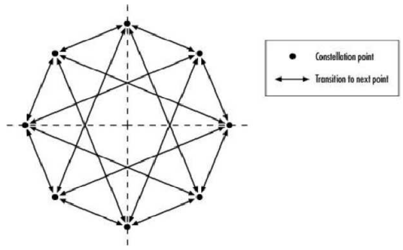
used to modulate the in-phase component of the carrier, while the odd (or even) bits are used to modulate the quadrature-phase component of the carrier. The QPSK transmitted signal is defined by:

$$s_i(t) = A\cos(\omega t + (i-1)\pi/2), i = (1, 2, 3, 4)$$

and the constellation disgram is shown in Figure

Offset-QPSK

As in QPSK, as shown in Figure 6.3, the NRZ data is split into two streams of odd and even bits. Each bit in these streams has a duration of twice the bit duration,



Tb, of the original data stream. These odd (d1(t)) and even bit streams (d2(t)) are then used to modulate two sinusoidals in phase quadrature,and hence these data streams are also called the in-phase and and quadrature phase components. After modulation they are added up and transmitted. The constellation diagram of Offset- QPSK is the same as QPSK. Offset-QPSK differs from QPSK in that the d1(t) and d2(t) are aligned such that the timing of the pulse streams are offset with respect to each other by Tb seconds. From the constellation diagram it is observed that a signal point in any quadrant can take a value in the diagonally opposite quadrant only when two pulses change their polarities together leading to an abrupt 180 degree phase shift between adjacent symbol slots. This is prevented in O-QPSK and the allowed phase transitions are 90 degree. Abrupt phase changes leading to sudden changes in the signal amplitude in the time domain corresponds to significant out of band high frequency components in the frequency domain. Thus to reduce these sidelobes spectral shaping is done at baseband. When high effciency power ampliffers, whose non-linearity increases as the effciency goes high, are used then due to distortion, harmonics are generated and this leads to what is known as spectral regrowth.

Since sudden 180 degree phase changes cannot occur in OQPSK, this problem is reduced to a certain extent.
DQPSK like QPSK can be thought to be carried in the phase of a single modulated carrier or on the amplitudes of a pair of quadrature carriers. The
modulated signal during the time slot of kT < t < (k + 1)T given by:

$$s(t) = cos(2\pi f_c t + \psi_{k+1})$$

Here

$$\psi_{k+1} = \psi_k + \Delta\psi_k \text{ and } \Delta\psi_k \text{ can take values } \pi/4 \text{ for } 00,\ 3\pi/4 \text{ for } 01,\ -3\pi/4$$

This corresponds to eight points in the signal constellation but at any instant of time only one of the four points are possible: the four points on axis or the four points axis. The constellation diagram along with possible transitions are shown in Figure

Angle Modulation (FM and PM)

There are a number of ways in which the phase of a carrier signal may be varied in accordance with the baseband signal; the two most important classes of angle modulation being frequency modulation and phase modulation. Frequency modulation (FM) involves changing of the frequency of the carrier signal according to message signal. As the information in frequency modulation is in the frequency of modulated signal, it is a nonlinear modulation technique. In this method, the amplitude of the carrier wave is kept constant (this is why FM is called constant envelope). FM is thus part of a more general class of modulation known as angle modulation. Frequency modulated signals have better noise immunity and give better performance in fading scenario as compared to amplitude modulation.Unlike AM, in an 114 FM system, the modulation index, and hence bandwidth occupancy, can be varied to obtain greater signal to noise performance.This ability of an FM system to trade bandwidth for SNR is perhaps the most important reason for its superiority over AM. However, AM signals are able to occupy less bandwidth as compared to FM signals, since the transmission system is linear. An FM signal is a constant envelope signal, due to the fact that the envelope of the carrier does not change with changes in the modulating signal. The constant envelope of the transmitted signal allows effcient Class C power ampliffers to be used for RF power ampliffcation of FM. In AM, however, it is critical to maintain linearity between the applied message and the amplitude of the transmitted signal, thus linear Class A or AB amplifiers, which are not as power efficient, must be used. FM systems require a wider frequency band in the transmitting media (generally several times as large as that needed for AM) in order to obtain the advantages of reduced noise and capture effect. FM transmitter and receiver equipment is also more complex than that used by amplitude modulation systems. Although frequency modulation systems are tolerant to certain types of signal and circuit nonlinearities, special attention must be given to phase characteristics. Both AM and FM may be demodulated using inexpensive non-coherent detectors. AM is easily demodulated using an envelope detector whereas FM is demodulated using a discriminator or slope detector. In FM the instantaneous frequency of the carrier signal is varied linearly with the baseband message signal m(t), as shown in following equation:

$$s_{FM}(t) = A_c \cos[2\pi f_c t + \theta(t)] = A_c \cos[2\pi f_c t + 2\pi k_f \int m(\eta) d\eta]$$

where Ac, is the amplitude of the carrier, fc is the carrier frequency, and kf is the frequency deviation constant (measured in units of Hz/V). Phase modulation (PM) is a form of angle modulation in which the angle _(t) of the carrier signal is varied linearly with the baseband message signal m(t), as shown in equation below.

$$s_{PM}(t) = A_c \cos(2\pi f_c t + k_\theta m(t))$$

The frequency modulation index Δf, defines the relationship between the message amplitude and the bandwidth of the transmitted signal, and is given by

$$\beta_f = \frac{k_f A_m}{W} = \frac{\Delta}{W}$$

where Am is the peak value of the modulating signal, Δf is the peak frequency deviation of the transmitter and W is the maximum bandwidth of the modulating signal.

The phase modulation index Δp is given by

$$\beta_p = k_\theta A_m = \Delta\theta$$

where, Δθ is the peak phase deviation of the transmitter.

BFSK

In Binary Frequency Shift keying (BFSK),the frequency of constant amplitude carrier signal is switched between two values according to the two possible message states (called high and low tones) corresponding to a binary 1 or 0. Depending on how the frequency variations are imparted into the transmitted waveform,the FSK signal will have either a discontinuous phase or continuous phase between bits. In general, an FSK signal may be represented as

$$S(t) = \sqrt{(2E_b/T)} \cos(2\pi f_i t).$$

where T is the symbol duration and Eb is the energy per bit

$$S_i = \sqrt{(E_b)}\phi(t).$$

$$\phi(t) = \sqrt{(2/T)} \cos(2\pi f_i t).$$

There are two FSK signals to represent 1 and 0, i.e.,

$$S_1(t) = \sqrt{(2E_b/T)} \cos(2\pi f_1 t + \theta(0)) \qquad \to 1$$

$$S_2(t) = \sqrt{(2E_b/T)} \cos(2\pi f_2 t + \theta(0)) \qquad \to 0$$

where θ(0) sums the phase up to t = 0. Let us now consider a continuous phase FSK as

$$S(t) = \sqrt{(2E_b/T)} \cos(2\pi f_c t + \theta(t))$$

Expressing θ(t) in terms of θ(0) with a new unknown factor h, we get

$$\theta(t) = \theta(0) \pm \pi h t/T \qquad\qquad 0 \le t \le T$$

$$S(t) = \sqrt{\frac{2E_b}{T}} \cos(2\pi f_c t \pm \pi h t/T + \theta(0)) = \sqrt{\frac{2E_b}{T}} \cos(2\pi(f_c \pm h/2T)t + \theta(0))$$

It shows that we can choose two frequencies f1 and f2 such that

$$f_1 = f_c + h/2T$$

$$f_2 = f_c - h/2T$$

for which the expression of FSK conforms to that of CPFSK. On the other hand, fc and h can be expressed in terms of f1 and f2 as

$$f_c = [f_1 + f_2]/2$$

$$h = \frac{(f_1 - f_2)}{1/T}.$$

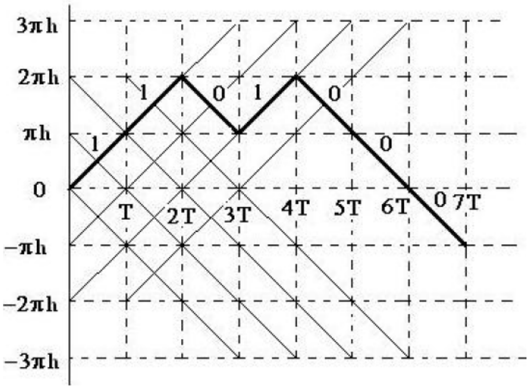
Therefore, the unknown factor h can be treated as the difference between f1 and f2, normalized with respect to bit rate 1=T . It is called the deviation ratio. We know that

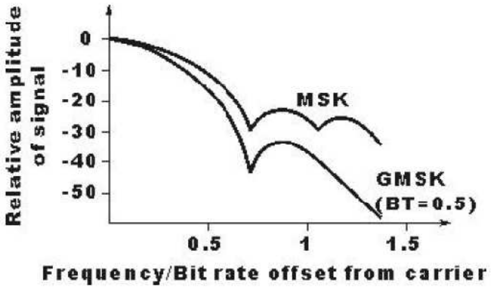
$\theta(t) - \theta(0) = \pm\pi ht/T,\ 0 \leq t \leq T$

If we substitute t = T, we have

$$\begin{aligned} \theta(T) - \theta(0) &= \pm\pi h && \text{where} \\ &= \pi h && \rightarrow 1 \\ &= -\pi h && \rightarrow 0 \end{aligned}$$

This type of CPFSK is advantageous since by looking only at the phase, the transmitted bit can be predicted. In Figure 6.8, we show a phase tree of such a CPFSK signal with the transmitted bit stream of 1101000. A special case of CPFSK is achieved with h = 0:5, and the resulting scheme is called Minimum Shift Keying (MSK) which is used in mobile communications. In this case, the phase differences reduce to only θ_=2 and the phase tree is called the phase trellis. An MSK signal can also be thought as a special case of OQPSK where the baseband rectangular pulses are replaced by half sinusoidal pulses. Spectral characteristics of an MSK signal is shown in Figure 6.9 from which it is clear that ACI is present in the spectrum. Hence a pulse shaping technique is required. In order to have a compact signal spectrum as well as maintaining the constant envelope property, we use a pulse shaping filter with



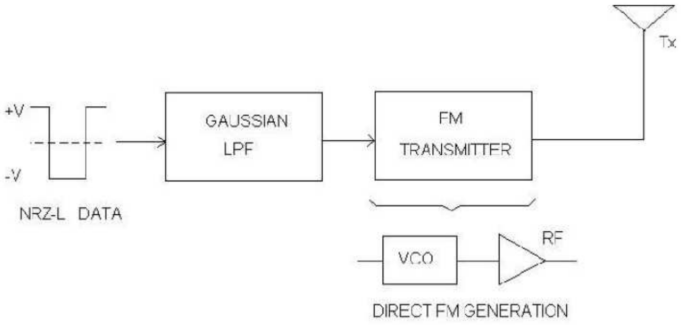


1. a narrow BW frequency and sharp cutoff characteristics (in order to suppress the high frequency component of the signal);
2. an impulse response with relatively low overshoot (to limit FM instant frequency deviation;

GMSK Scheme

GMSK is a simple modulation scheme that may be taken as a derivative of MSK. In GMSK, the sidelobe levels of the spectrum are further reduced by passing a non- return to zero (NRZ-L) data waveform through a premodulation Gaussian pulse shaping filter. Baseband

Gaussian pulse shaping smoothes the trajectory of the MSK signals and hence stabilizes instantaneous frequency variations over time. This has the effect of considerably reducing the sidelobes in the transmitted spectrum. A GMSK generation scheme with NRZ-L data is shown in Figure and a receiver of the same scheme with some MSI gates is shown in Figure



Module 3

Spread Spectrum Techniques :

Spread Spectrum techniques have some powerful properties which make them an excellent candidate for networking applications. To better understand why, we will take a closer look at this fascinating area, and its implications for networking.
The first major application of Spread Spectrum Techniques (SST) arose during the mid-sixties, when NASA employed the method to precisely measure the range to deep space probes. In the following years, the US military became enamoured of SST due to its ability to withstand jamming (ie intentional interference), and it ability to resist eavesdropping. Today this technology forms the basis for the ubiquitous NavStar Global Positioning System (GPS), the soon to become ubiquitous JTIDS (Joint Tactical Information Distribution System/Link-16) datalink (used between aircraft, ships and land vehicles), and last but not least, the virtually undetectable bombing and navigation radar on the bat-winged B-2 bomber. if you ever get asked what your mobile networked laptop shares in common with a stealth bomber (excluding astronomical cost), you can state without fear of contradiction that it uses the same class of modulation algorithm.How is this black magic achieved ? The starting point is Claude Shannon's information theory, a topic beloved by diehard communications engineers. Shannon's formula for channel capacity is a relationship between achievable bit rate, signal bandwidth and signal to noise ratio. Channel capacity is proportional to bandwidth and the logarithm to the base of two of one plus the signal to noise ratio, or:*Capacity = Bandwidth*log2 (1 + SNR).*What this means is that the more bandwidth and the better the signal to noise ratio, the more bits per second you can push through a channel. This is indeed common sense. However, let us consider a situation where

the signal is weaker than the noise which is trashing it. Under these conditions this relationship becomes much simpler, and can be approximated by a ratio of

Capacity/Bandwidth = 1.44 SNR.*

What this says is that we can trade signal to noise ratio for bandwidth, or vice versa. If we can find a way of encoding our data into a large signal bandwidth, then we can get error free transmission under conditions where the noise is much more powerful than the signal we are using. This very simple idea is the secret behind spread spectrum techniques. Consider the example of a 3 kHz voice signal which we wish to send through a channel with a noise level 100 times as powerful as the signal. Manipulating the preceding equation, we soon find that we require a bandwidth of 208 kHz, which is about 70 times greater than the voice signal we wish to carry. Readers with a knowledge of radio will note here that this idea of spreading is a central part of FM radio and the reason why it produces good sound quality compared to the simpler AM scheme. Other than punching through large levels of background noise, why would we otherwise consider using spread spectrum techniques ? There are a number of good practical reasons why spread spectrum modulation is technically superior to the intuitively more obvious techniques such as AM and FM, and all of the hybrids which lie in between.

- The Ability to Selectively Address. If we are clever about how we spread the signal, and use the proper encoding method, then the signal can only be decoded by a receiver which knows the transmitter's code. Therefore by setting the transmitter's code, we can target a specific receiver in a group, or vice versa. This is termed *Code Division Multiple Access*.
- Bandwidth Sharing. If we are clever about selecting our modulation codes, it is entirely feasible to have multiple pairs of receivers and transmitters occupying the same bandwidth. This would be equivalent to having say ten TV channels all operating at the same frequency. In a world where the radio spectrum is being busily carved up for commercial broadcast users, the ability to share bandwidth is a valuable capability.
- Security from Eavesdropping. If an eavesdropper does not know the modulation code of a spread spectrum transmission, all the eavesdropper will see is random electrical noise rather than something to eavesdrop. If done properly, this can provide almost perfect immunity to interception.
- Immunity to Interference. If an external radio signal interferes with a spread spectrum transmission, it will be rejected by the demodulation mechanism in a fashion similar to noise. Therefore we return to the starting point of this discussion, which is that spread spectrum methods can provide excellent error rates even with very faint signals.
- Difficulty in Detection. Because a spread spectrum link puts out much less power per bandwidth than a conventional radio link, having spread it over a wider bandwidth, and a knowledge of the link's code is required to demodulate it, spread spectrum signals are extremely difficult to detect. This means that they can coexist with other more conventional signals without causing catastrophic interference to narrowband links.

These characteristics endeared spread spectrum comes to the military community, who are understandably paranoid about being eavesdropped and jammed. However, the same properties are no less useful for local area networking over radio links. Indeed these are the reasons why the current IEEE draft specification for radio LANs is written around spread spectrum modulations. To better understand the inner workings of this fascinating area, we will now more closely examine the various choices we have for spread spectrum designs. The two basic methods are indeed both used in LAN equipment.

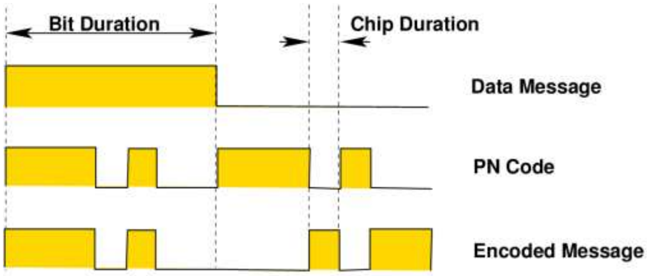


Fig 11: DIRECT SPREADING (DS)

Direct Sequence Systems : Direct Sequence (DS) methods are the most frequently used spread spectrum technique, and also the conceptually simplest to understand. DS modulation is achieved by modulating the carrier wave with a digital code sequence which has a bit rate much higher than that of the message to be sent. This code sequence is typically a pseudorandom binary code (often termed "pseudo-noise" or PN), specifically chosen for desirable statistical properties. In effect we are transmitting a wideband noise like signal which contains embedded message data. The time period of a single bit in the PN code is termed a *chip*, and the bit rate of the PN code is termed the *chip rate*. A wide range of pseudorandom codes exist which can be applied to this task. These codes should ideally be balanced, with an equal number of ones and zeroes over the length of the sequence (also termed the code run), as well a good code should be cryptographically secure. A spread spectrum system which uses a cryptographically insecure code will still possess the properties previously discussed, but if an eavesdropper can synchronise on to the signal they should be able to eventually crack it and extract the data. Using a secure code prevents this. The mechanics of generating pseudorandom codes is a fascinating area within itself. The most commonly used approach for producing a wide range of code types is the use of a tapped register with feedback, very simple to implement in hardware. A PN code generator of this type uses a register with taps between selected stages. These taps are logically ORed and then fed back in to the input stage of the register. The state machine produced in this fashion will periodically cycle through the same PN sequence as the clock is applied. Significantly, code sequence lengths of up to thousands of bits in length can be produced with about a dozen register stages. With modern VLSI techniques it is feasible to build generators with clock speeds up to hundreds of MHz on any die, moreover recent high speed Emitter Coupled Logic devices allow the creation of generators with clock speeds into the GHz region.

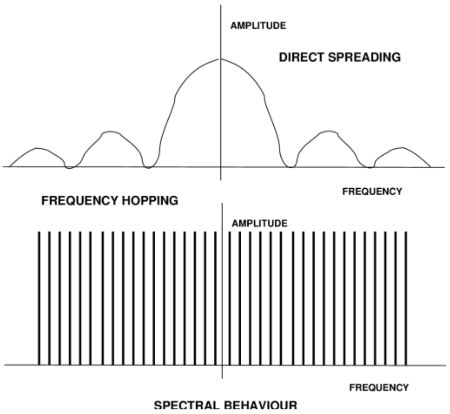


Fig 12:

Having produced a black box which generates a PN code with the required characteristics, the process of combining the PN modulation with the data to be transmitted, and modulating this upon a carrier is not technically difficult at all. The simplest technique, one of many, is to invert the PN code when a '0' bit of message data is to be sent, and to transmit the PN code unchanged when a '1' bit of message data is to be sent. This technique is termed *Bit Inversion Modulation*. The result is a PN code with an embedded data message. The simplest form of carrier modulation which can be used is AM, however in practice one or another form of *Phase Shift Keying (PSK)* is usually employed. PSK schemes are commonly used in modems, and involve the modulation of the carrier phase with the data signal. In a DS transmitter using Binary PSK, the carrier wave is phase shifted back and forth 180 degrees with each 1 or 0 in the PN code chip stream being sent. The process of modulating the carrier with the PN code is often termed *spreading*.

The internals of a DS receiver are somewhat more complex than those of the transmitter, but not vastly so. The central idea in all SST receivers is the use of the correlation operation. Correlation, a favourite method of our friends in the statistics community, is a mathematical operation which determines a measure of likeness or similarity between two sets of data or two time processes. In an SST receiver, the correlation operation is use to measure the similarity of a received PN code sequence to an internally generated PN code sequence. Ideally, if these PN sequences are the same, a high correlation will be detected, whereas if the codes are different, a low correlation is detected.

Mathematically the correlation operation, in its simplest form, is the integral of the product of two time varying functions. In a DS receiver of the simplest kind, the hardware maps directly onto the basic maths. The *correlator* is built by combining a multiplier with a low pass filter (ie integrator in a control engineer's language).

One of the two time varying functions is the received PN modulated signal, the other is the PN sequence produced by a PN generator internal to the receiver. In the simplest situation, the receiver's PN generator is a clone of the PN generator in the transmitter.

The multiplier can be one of many designs, importantly it multiplies in effect two single numbers and is therefore trivially simple. Classical textbooks cite the analogue doubly balanced mixer as the standard multiplier. The output from the multiplier is a time varying measure of the similarity between the two codes, blended with the remnants of uncorrelated (ie real) noise and interfering signals.

The integration operation disposes of the latter, and we are then left with the data which we intended to extract. This series of operations is often termed *despreading*. In practice, we often need to synchronise our receiver's PN generator to the incoming SST signal, therefore there is often much additional complexity required to produce an internal reference PN sequence in proper lockstep with the incoming message PN sequence. At this point it is worth reflecting upon what we have. We can generate either cryptographically secure or insecure codes. We can embed a digital data stream in one or another fashion into the code stream. All of this can be performed with pure digital logic. Once we have a combined data/code stream, we can use a very simple analogue modulation to put the message upon a carrier. The resulting radio signal looks like white noise to a third party who doesn't know out code. Our receiver shares similar hardware design with our transmitter. It uses a trivial demodulation scheme, and extracts digital data from the incoming PN data/code stream. Other radio signals occupying our bandwidth are largely ignored. Whilst an SST transmitter-receiver pair may be conceptually more complex to understand than most classical analogue schemes, it is well suited to implementation in digital logic because most of the smarts at either end of the link are purely digital. This means that such hardware can be made much more compact than many classical narrowband analogue schemes, which often require a lot of analogue hardware which may or may not be easy to squeeze into Silicon. Consider a narrowband 16 or 64 level QAM scheme, which is not only vulnerable to interference and noise, but also requires a digital signal processing chip to demodulate. For those readers with a bent toward radio engineering, the spectral envelope of a DS system is typically a sinc function, with suppressed outer sidebands beyond the first null, and often a suppressed carrier. A parameter which radio types will appreciate is *process gain*, a measure of signal to noise ratio improvement achieved by despreading the received signal. For a DS system it is typically about twice the ratio of RF bandwidth to message bandwidth. Therefore to improve your ability to reject interference by 20 dB, you need to increase your chip rate by a factor of 100.

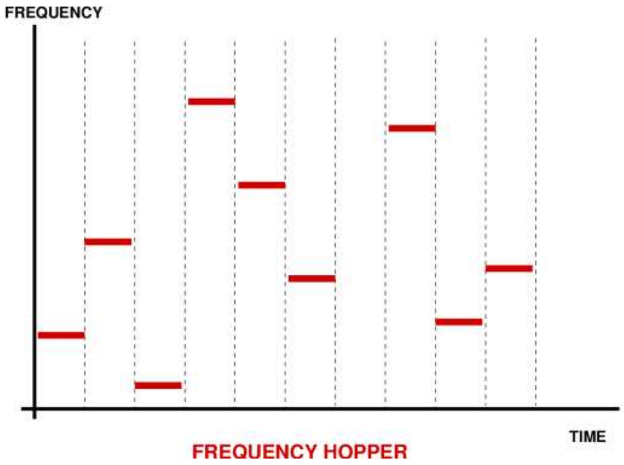


Fig 13

Frequency Hopping Systems : Frequency Hoppers (FH) are a more sophisticated and arguably better family of spread spectrum techniques than the simpler DS systems. However, performance comes with a price tag here, and FH systems are significantly more complex than DS systems. The central idea behind a FH system is to retune the transmitter RF carrier frequency to a pseudorandomly determined frequency value. In this fashion the carrier keeps popping up a different frequencies, in a pseudorandom pattern. The carrier itself amy be modulated directly with the data using one of many possible schemes. The available radio spectrum is thus split up into a discrete number of frequency channels, which are occupied by the RF carrier pseudorandomly in time. Unless you know the PN code used, you have no idea where the carrier wave is likely to pop up next, therefore eavesdropping will be quite difficult. Frequency hoppers are typically divided into fast and slow hoppers. A slow frequency hopper will change carrier frequency pseudorandomly at a frequency which is much slower than the data bit rate on the carrier. A fast frequency hopper will do so at a frequency which is faster than that of the data message.

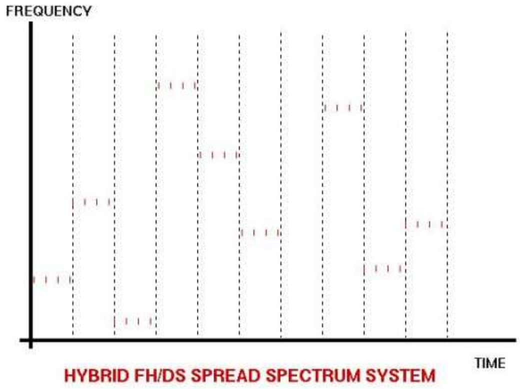


Fig 14

Hybrid(FH/DS)Systems

If we are *really* paranoid about being eavesdropped, we can take further steps to make our signal difficult to find. A commonly used example is that of a hybrid spread spectrum system using both FH and DS techniques. Such schemes will typically employ frequency hopping of the carrier wave, while concurrently using a DS modulation technique to modulate the data upon the carrier. In this fashion an essentially DS modulated message is hopped about the spectrum. To successfully intercept such a signal you must first crack the FH code, and then crack the DS code. If you want to be further secure, you encrypt your data stream with a very secure crypto code before you feed it into your DS modulator, and employ cryptographically secure PN codes for the DS and FH operations. Your eavesdropper then has to chew his way through three levels of encoding. Such a scheme is used in the military JTIDS/Link 16 datalink.

Multiple Access Techniques : In wireless communication systems it is often desirable to allow the subscriber to send simultaneously information to the base station while receiving information from the base station. A cellular system divides any given area into cells where a mobile unit in each cell communicates with a base station. The main aim in the cellular system design is to be able to increase the capacity of the channel i.e. to handle as many calls as possible in a given bandwidth with a sufficient level of quality of service. There are several different ways to allow access to the channel. These includes mainly the following:

1) Frequency division multiple-access (FDMA)
2) Time division multiple-access (TDMA)
3) Code division multiple-access (CDMA)
4) Space Division Multiple access (SDMA)

Frequency Division Multiple Access

This was the initial multiple-access technique for cellular systems in which each individual user is assigned a pair of frequencies while making or receiving a call as shown in Figure below. One frequency is used for downlink and one pair for uplink. This is called frequency division duplexing (FDD). That allocated frequency pair is not used in the same cell or adjacent cells during the call so as to reduce the co-channel interference. Even though the user may not be talking, the spectrum cannot be reassigned as long as a call is in place. Different users can use the same frequency in the same cell except that they must transmit at different times. The features of FDMA are as follows: The FDMA channel carries only one phone circuit at a time. If an FDMA channel is not in use, then it sits idle and it cannot be used by other users to increase share capacity. After the assignment of the voice channel the BS and the MS transmit simultaneously and continuously. The bandwidths of FDMA systems are generally narrow i.e. FDMA is usually 159 implemented in a narrow band system The symbol time is large compared to the average delay spread. The complexity of the FDMA mobile systems is lower than that of TDMA mobile systems. FDMA requires tight filtering to minimize the adjacent channel interference.

Time Division Multiple Access

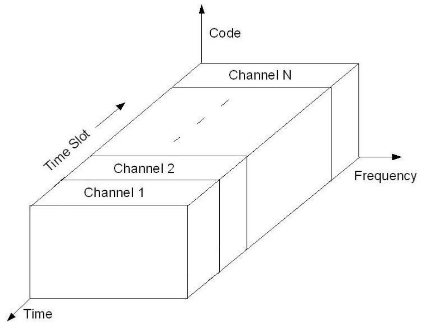


Fig 15: Time division Multiplexing

In digital systems, continuous transmission is not required because users do not use the allotted bandwidth all the time. In such cases, TDMA is a complimentary access technique to FDMA. Global Systems for Mobile communications (GSM) uses the TDMA technique. In TDMA, the entire bandwidth is available to the user but only for a finite period of time. In most cases the available bandwidth is divided into fewer channels compared to FDMA and the users are allotted time slots during which they have the entire channel bandwidth at their disposal, as shown in Figure above. TDMA requires careful time synchronization since users share the bandwidth in the frequency domain. The number of channels are less, inter channel interference is almost negligible. TDMA uses different time slots for transmission and reception.

This type of duplexing is referred to as Time division duplexing(TDD). The features of TDMA includes the following: TDMA shares a single carrier frequency with several users where each users makes use of non overlapping time slots. The number of time slots per frame depends on several factors such as modulation technique, available bandwidth etc. Data transmission in TDMA is not continuous but occurs in bursts. This results in low battery consumption since the subscriber transmitter can be turned OFF when not in use. Because of a discontinuous transmission in TDMA the handoff process is much simpler for a subscriber unit, since it is able to listen to other base stations during idle time slots. TDMA uses different time slots for transmission and reception thus duplexers are not required. TDMA has an advantage that is possible to allocate different numbers of time slots per frame to different users. Thus bandwidth can be supplied on demand to different users by concatenating or reassigning time slot based on priority.

Code Division Multiple Access: In CDMA, the same bandwidth is occupied by all the users, however they are all assigned separate codes, which differentiates them from each other (shown in Figure). CDMA utilize a spread spectrum technique in which a spreading signal (which is uncorrelated to the signal and has a large bandwidth) is used to spread the narrow band message signal. The most commonly used technology for CDMA is Direct Sequence Spread Spectrum (DS-SS) . In DS-SS, the message signal is multiplied by a Pseudo Random Noise Code. Each user is given his own codeword which is orthogonal to the codes of other users and in order to detect the user, the receiver must know the codeword used by the transmitter. There are, however, two problems in such systems which are discussed in the sequel.

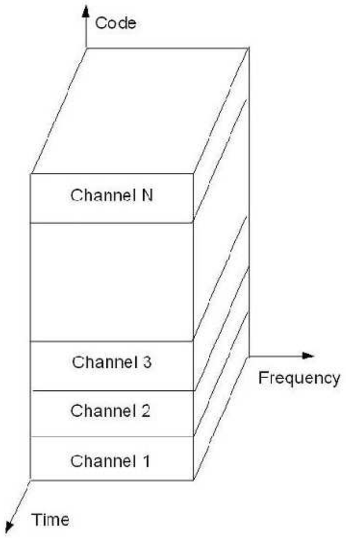


Fig 16 Code Division Multiple Access

Space Division Multiple Access

SDMA utilizes the spatial separation of the users in order to optimize the use of the frequency spectrum. A primitive form of SDMA is when the same frequency is reused in different cells in a cellular wireless network. The radiated power of each user is controlled by Space division multiple access. SDMA serves different users by using spot beam antenna. These areas may be served by the same frequency or different frequencies. However for limited co-channel interference it is required that the cells be sufficiently separated. This limits the number of cells a region can be divided into and hence limits the frequency re-use factor. A more advanced approach can further increase the capacity of the network. This technique would enable frequency re-use within the cell. In a practical cellular environment it is improbable to have just one transmitter fall within the receiver beam width. Therefore it becomes imperative to use other multiple access techniques in conjunction with SDMA. When different areas are covered by the antenna beam, frequency can be re-used, in which case TDMA or CDMA is employed, for different frequencies FDMA can be used.

Module 4

Various Generations of Wireless Networks:

At the initial phase, mobile communication was restricted to certain official users and the cellular concept was never even dreamt of being made commercially available. Moreover, even the growth in the cellular networks was very slow. However, with the development of newer and better technologies starting from the 1970s and with the mobile users now connected to the PSTN, there has been a remarkable growth in the cellular radio. However, the spread of mobile communication was very fast in the 1990s when the government throughout the world provided radio spectrum licenses for Personal Communication Service (PCS) in 1.8 - 2 GHz frequency band.

First Generation Networks

The first mobile phone system in the market was AMPS. It was the first U.S. cellular telephone system, deployed in Chicago in 1983. The main technology of this first generation mobile system was FDMA/FDD and analog FM.

Second Generation Networks

Digital modulation formats were introduced in this generation with the main technology as TDMA/FDD and CDMA/FDD. The 2G systems introduced three popular TDMA standards and one popular CDMA standard in the market. These are as follows:

TDMA/FDD Standards

(a) Global System for Mobile (GSM): The GSM standard, introduced by Groupe Special Mobile, was aimed at designing a uniform pan-European mobile system. It was the first fully digital system utilizing the 900 MHz frequency band. The initial GSM had 200 KHz radio channels, 8 full-rate or 16 half-rate TDMA channels per carrier, encryption of speech, low speed data services and support for SMS for which it gained quick popularity.

(b) Interim Standard 136 (IS-136): It was popularly known as North American Digital Cellular (NADC) system. In this system, there were 3 full-rate TDMA users over each 30 KHz channel. The need of this system was mainly to increase the capacity over the earlier analog (AMPS) system.

(c) Pacific Digital Cellular (PDC): This standard was developed as the counterpart of NADC in Japan. The main advantage of this standard was its low transmission bit rate which led to its better spectrum utilization.

CDMA/FDD Standard

Interim Standard 95 (IS-95): The IS-95 standard, also popularly known as CDMAOne, uses 64 orthogonally coded users and codewords are transmitted simultaneously on each of 1.25 MHz channels. Certain services that have been standardized as a part of IS-95 standard are: short messaging service, slotted paging, over-the-air activation (meaning the mobile can be activated by the service provider without any third party intervention), enhanced mobile station identities etc.

2.5G Mobile Networks

In an effort to retrofft the 2G standards for compatibility with increased throughput rates to support modern Internet application, the new data centric standards were developed to be overlaid on 2G standards and this is known as 2.5G standard. Here, the main upgradation techniques are:

ff supporting higher data rate transmission for web browsing

ff supporting e-mail traffic

ff enabling location-based mobile service

2.5G networks also brought into the market some popular application, a few of which are: Wireless Application Protocol (WAP), General Packet Radio Service (GPRS), High Speed Circuit Switched Dada (HSCSD), Enhanced Data rates for GSM Evolution (EDGE) etc.

3G: Third Generation Networks

3G is the third generation of mobile phone standards and technology, superseding 2.5G. It is based on the International Telecommunication Union (ITU) family of standards under the International Mobile Telecommunications-2000 (IMT-2000). ITU launched IMT-2000 program, which, together with the main industry and standardization bodies worldwide, targets to implement a global frequency band that would support a single, ubiquitous wireless communication standard for all countries, to provide the framework for the definition of the 3G mobile systems. Several radio access technologies have been accepted by ITU as part of the IMT-2000 framework. 3G networks enable network operators to offer users a wider range of more advanced services while achieving greater network capacity through improved spectral efficiency. Services include wide-area wireless voice telephony, video calls, and broadband wireless data, all in a mobile environment. Additional features also include HSPA data transmission capabilities able to deliver speeds up to 14.4Mbit/s on the

down link and 5.8Mbit/s on the uplink. 3G networks are wide area cellular telephone networks which evolved to incorporate high-speed internet access and video telephony. IMT-2000 defines a set of technical requirements for the realization of such targets, which can be summarized as follows:

_ high data rates: 144 kbps in all environments and 2 Mbps in low-mobility and
indoor environments

_ symmetrical and asymmetrical data transmission

_ circuit-switched and packet-switched-based services
_ speech quality comparable to wire-line quality
_ improved spectral efficiency
_ several simultaneous services to end users for multimedia services
_ seamless incorporation of second-generation cellular systems
_ global roaming
_ open architecture for the rapid introduction of new services and technology.

Beyond 3G networks, or 4G (Fourth Generation), represent the next complete evolution in wireless communications. A 4G system will be able to provide a comprehensive IP solution where voice, data and streamed multimedia can be given to users at higher data rates than previous generations. There is no formal definition for 4G ; however, there are certain objectives that are projected for 4G. It will be capable of providing between 100 Mbit/s and 1 Gbit/s speeds both indoors and outdoors, with premium quality and high security. It would also support systems like multicarrier communication, MIMO and UWB.

GSM Architecture

Figure below depicts the original GSM architecture that supported circuit switching services only. Mobile Stations (MS) (handheld phones) have a radio connection with a cell. One or a set of cells are supported by a Base Transceiver Station (BTS). A BTS may have an antenna high above the ground on top of a mast in order to increase the coverage of the cells. Cells cover the area where GSM users are reachable. Cells may be organized into several layers. Large cells are useful for fast moving Mobile Stations and small cells are needed to increase network capacity – the number of MSs that can be served simultaneously in a particular spot. A call may start in one cell, the MS may traverse through a number of cells while the call is on and the MS may be located in a cell under a different BTS or even a different MSC when the call ends. The action of changing a cell during a call is called a handover. Several BTSs are controlled by a Base Station Controller (BSC). There may be many BSCs under the control of one Mobile Switching Center (MSC). An MSC controlling BSCs is in the same position GSM as a Local Exchange in PSTN or ISDN. The difference is that an MSC does "own" its subscribers. Rather, all the MSs it is controlling are visitors. A Visitor Location Register (VLR) for storing information about the users and MSs visiting this MSC is separa tely specified but always resides in the MSC. Recall that also a wire line Local Exchange contains a subscriber database.

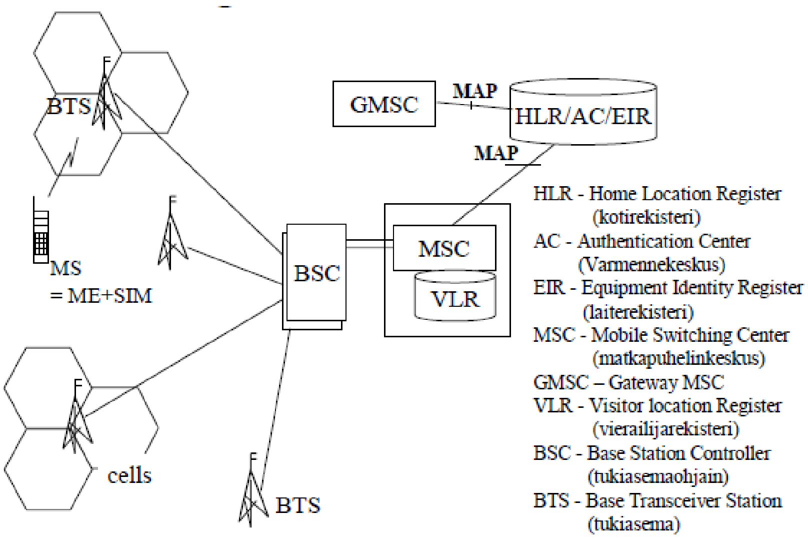

BTS
MS
= ME+SIM
cells
BTS
BSC
MSC
VLR
GMSC
MAP
HLR/AC/EIR
MAP
HLR - Home Location Register
(kotirekisteri)
AC - Authentication Center
(Varmennekeskus)
EIR - Equipment Identity Register
(laiterekisteri)
MSC - Mobile Switching Center
(matkapuhelinkeskus)
GMSC – Gateway MSC
VLR - Visitor location Register
(vierailijarekisteri)
BSC - Base Station Controller
(tukiasemaohjain)
BTS - Base Transceiver Station
(tukiasema)

Fig 17: GSM Architecture

Cells normally forming a continuous geographical area are organized into a Location Area. A location area is a logical concept and gives the accuracy with which the location of MSs is maintained in the HLR. Each MS has a MSISDN number. It is a logical or a directory number. An MSISDN number maps to a particular HLR based on its leading digits. Thus MSISDN is routable to the HLR that is supposed to know where the MS is located. The HLR also house the Authentication Center (AC) for storing authentication information about the users and the Equipment Identity Register (EIR). Each MS has an Equipment Identity on silicon and thus stolen equipment can be identified, traced and removed from the network. MSs are reachable because they regularly make location updates to the VLR. The VLR will authenticate the user and establish that the user can be billed for the use of the network resources by contacting the HLR. When the MS first establishes a connection with the network or changes the Location Area, VLR will update the location of the MS in the user's HLR. For call setup the GSM system uses ISUP between exchanges and BSSAP as the access signalling between BSS and the Circuit Core. Due to mobility additional signalling functionality is required in the Core network. This is mainly provided by the Mobile Application Part. Making a call from an MS (Mobile Originated call) is quite similar to the wire line case. Terminating a call to a Mobile (Mobile Terminated call) is different. When an exchange sees that it has a mobile number for the callee, it will route the call to a gateway MSC. The GMSC will send a request using the SS7 signalling network and the Mobile Application Part signaling to the HLR. The HLR will return the Mobile Station Routing Number (MSRN) to the GMSC. The MSRN like its name tells is routable in all switching systems i.e. based on leading digits only. It was dynamically assigned by the VLR for the call or for the duration of the visit and given to the HLR to be used for terminating calls. When the call arrives to the visited MSC, the MSC finds out the location of MS in the VLR with the accuracy of several cells. The MSC will page the MS in all those cells. Paging means that the MSC sends a call signal to the mobile using signaling channels over the air in several cells simultaneously. The MS will respond using a signaling channel and the cell that it sees best. There is one more significant difference in call establishment in GSM as compared to wire line networks. Radio resources are seen as very expensive and should be preserved as much as possible. Therefore, when an MS makes a call, no radio resources are allocated at the originating cell until it is known that the callee is not busy, out of coverage and is also willing to take the call. So, the reservation of timeslots of the air takes place later in the call establishment process.

From the very beginning GSM supported roaming. This means that subscribers of one operator can move into the area of another and make the use of the other operator's network. Usually, only international roaming is supported for business reasons. I.e. the two operators that allow their subscribers to use each other's networks cover different countries. International roaming has been a major benefit of GSM to the users and has helped to consolidate the GSM operators.

GSM frame structure

In GSM frequency band of 25 MHz is divided into 200 KHz of smaller bands, each carry one RF carrier, this gives 125 carriers. As one carrier is used as guard channel between GSM and other frequency bands 124 carriers are useful RF channels. This division of frequency pool is called FDMA. Now each RF carrier will have eight time slots. This division time wise is called TDMA. Here each RF carrier frequency is shared between 8 users hence in GSM system, the basic radio resource is a time slot with duration of about 577 microsec. As mentioned each time slot has 15/26 or 0.577ms of time duration. This time slot carries 156.25 bits which leads to bit rate of 270.833 kbps. This is explained below in TDMA gsm frame structure. For E-GSM number of ARFCNs are 174, for DCS1800 ARFNCs are 374. The GSM frame structure is designated as hyperframe, superframe, multiframe and frame. The minimum unit being frame (or TDMA frame) is made of 8 time slots. One GSM hyperframe composed of 2048 superframes. Each GSM superframe composed of multiframes (either 26 or 51 as described below). Each GSM multiframe composed of frames (either 51 or 26 based on multiframe type). Each frame composed of 8 time slots. Hence there will be total of 2715648 TDMA frames available in GSM and the same cycle continues. As shown in the figure 2 below, there are two varients to multiframe structure. 1. 26 frame multiframe - Called traffic multiframe, composed of 26 bursts in a duration of 120ms, out of these 24 are used for traffic, one for SACCH and one is not used. 2. 51 frame multiframe- Called control multiframe, composed of 51 bursts in a duration of 235.4 ms. This type of multiframe is divided into logical channels. These logical channels are time scheduled by BTS. Always occur at beacon frequency in time slot 0, it may also take up other time slots if required by system for example below

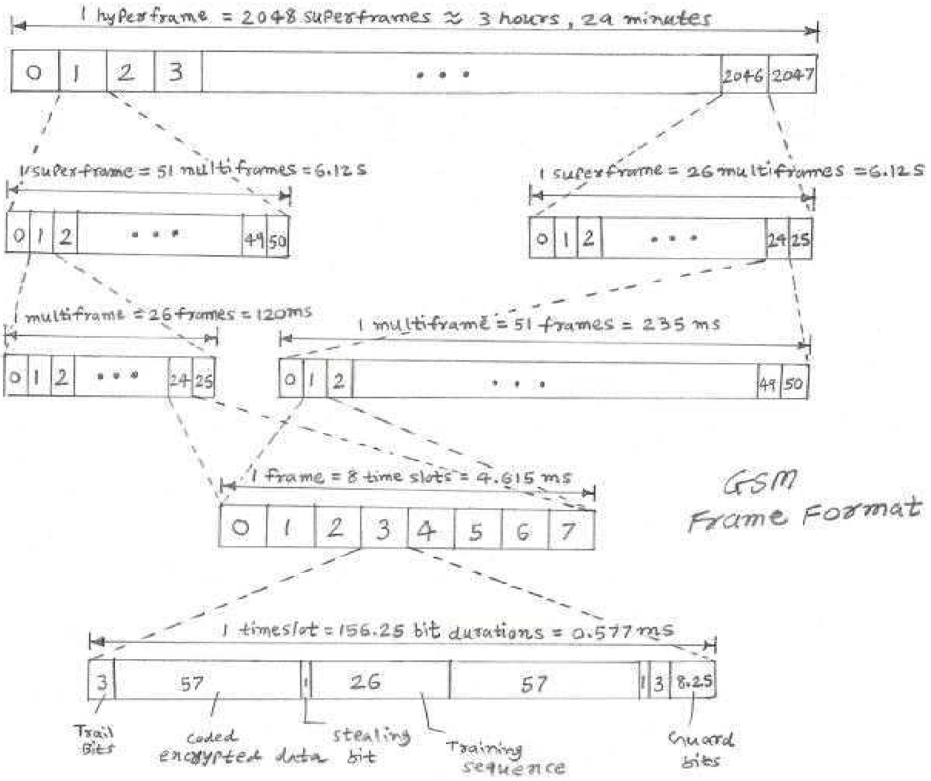

1 hyperframe = 2048 superframes ≈ 3 hours, 29 minutes
0
1
2
3
...
2046
2047
1 superframe = 51 multiframes = 6.12 s
0
1
2
...
49
50
1 superframe = 26 multiframes = 6.12 s
0
1
2
...
24
25
1 multiframe = 26 frames = 120 ms
0
1
2
...
24
25
1 multiframe = 51 frames = 235 ms
0
1
2
...
49
50
1 frame = 8 time slots = 4.615 ms
0
1
2
3
4
5
6
7
GSM Frame Format
1 timeslot = 156.25 bit durations = 0.577 ms
3
57
1
26
57
1
3
8.25
Trail bits
coded encrypted data
stealing bit
Training sequence
Guard bits

Fig 18. GSM Frame Structure

As shown in fig above each ARFCN or each channel in GSM will have 8 time slots TS0 to TS7. During network entry each GSM mobile phone is allocated one slot in downlink and one slot in uplink. Here in the figure GSM Mobile is allocated 890.2 MHz in the uplink and 935.2 MHz in the downlink. As mentioned TS0 is allocated which follows either 51 or 26 frame multiframe structure. Hence if at start 'F' is depicted which is FCCH after 4.615 ms (which is 7 time slot duration) S(SCH) will appear then after another 7 slots B(BCCH) will appear and so on till end of 51 frame Multiframe structure is completed and cycle continues as long as connection between Mobile and base station is active. similarly in the uplink, 26 frame multiframe structure follow, where T is TCH/FS (Traffic channel for full rate speech), and S is SACCH. The gsm frame structure can best be understood as depicted in the figure below with respect to downlink(BTS to MS) and uplink (MS to BTS) directions.

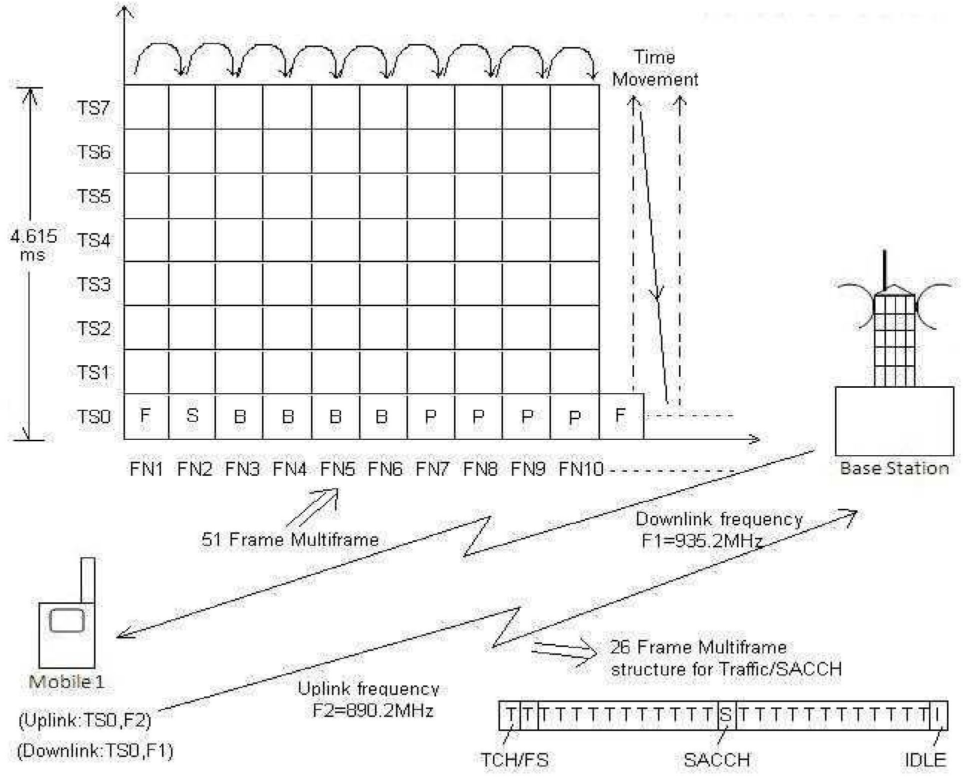

Time Movement
TS7
TS6
TS5
TS4
TS3
TS2
TS1
TS0
4.615 ms
F
S
B
B
B
B
P
P
P
P
F
FN1 FN2 FN3 FN4 FN5 FN6 FN7 FN8 FN9 FN10
Base Station
51 Frame Multiframe
Downlink frequency F1=935.2MHz
Mobile 1
(Uplink:TS0,F2)
(Downlink:TS0,F1)
Uplink frequency F2=890.2MHz
26 Frame Multiframe structure for Traffic/SACCH
T T T T T T T T T T T T T S T T T T T T T T T T T T I
TCH/FS
SACCH
IDLE

Fig 19. GSM Physical and logical channel concept

Frequencies in the uplink = 890.2 + 0.2 (N-1) MHz
Frequencies in the downlink = 935.2 + 0.2 (N-1) MHz
where, N is from 1 to 124 called ARFCN
As same antenna is used for transmit as well as receive, there is 3 time slots delay introduced between TS0 of uplink and TSO of downlink frequency. This helps avoid need of imultaneous transmission and reception by GSM mobile phone. The 3 slot time period is used by the Mobile subscriber to perform various functions e.g. processing data, measuring signal quality of neighbour cells etc. Engineers working in GSM should know GSM frame structure for both the downlink as well as uplink. They should also understand mapping of different channels to time slots in these GSM frame structures.

Traffic Routing in Wireless Networks

When there are many devices, it is necessary to develop suitable echanism for communication between any two devices. One a lternative is to establish point-to-point communication between each pair of devices using mesh topology. However, mesh topology is impractical for larg
e number of devices, because the number of links increases exponentially (n(n-1)/2, where n is the number of devices) with the number of devices. A better alternative is to use switching techniques leading to switched communication network. In the switched network methodology, the network consists of a set of interconnected nodes, among which information is transmitted from source to destination via different routes, which is controlled by the switching mechanism. A basic model of a
switched communication is shown in Fig. below. The end devices that wish to communicate with each other are called stations . The switching evices are called nodes . Some nodes

connect to other nodes and some are to connected to some stations . Key features of a switched communication network are given below:
Network Topology is not regular.
Uses FDM or TDM for node-to-node communication.
There exist multiple paths between a source-destination pair for better network reliability. The switching nodes are not concerned with the contents of data. Their purpose is to provide a switching facility that will move data from node to node until they reach the destination. The switching performed by different nodes can be categorized into the following three types:

Circuit Switching

Packet Switching

Message Switching

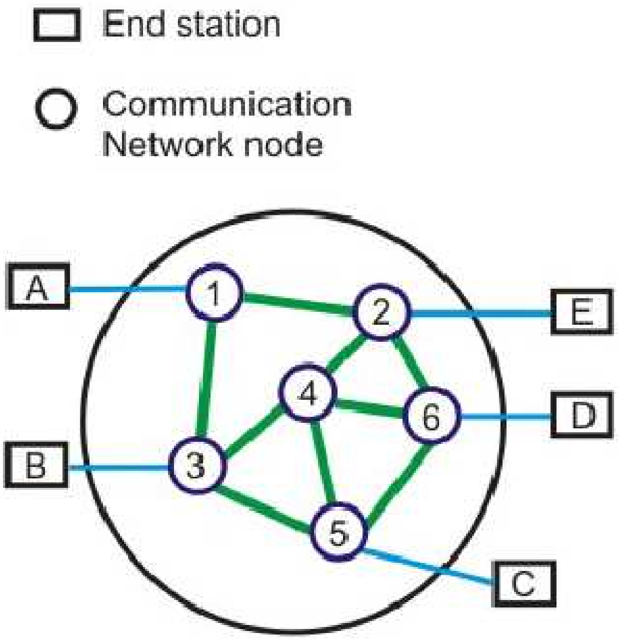


Figure20 : Basic model of a switched communication network

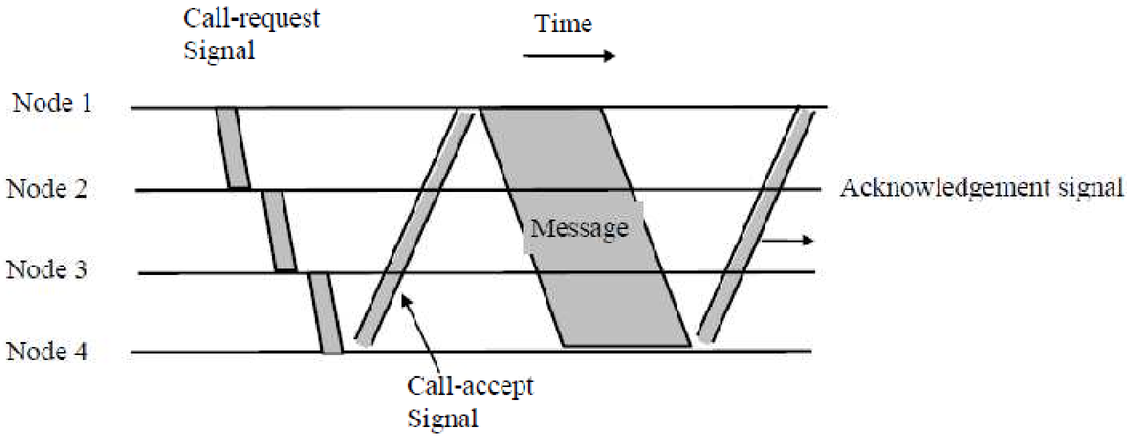
Circuit switching Technique

Communication via circuit switching implies that there is a dedicated communication path between the two stations. The path is a connected through a sequence of links between network nodes. On each physical link, a logical channel is dedicated to the connection. Circuit switching is commonly used technique in telephony, where the caller sends a special message with the address of the callee (i.e. by dialling a number) to state its destination. It involved the following three distinct steps, as shown in Fig

Circuit Establishment: To establish an end-to-end connection before any transfer of data.

Some segments of the circuit may be a dedicated link, while some other segments may be shared. *Data transfer:* Transfer data is from the source to the destination. The data may be analog or digital, depending on the nature of the network. The connection is generally full-duplex. Circuit disconnect: Terminate connection at the end of data transfer.

• Signals must be propagated to deallocate the dedicated resources.



Thus the actual physical electrical path or circuit between the source and destination host must be established before the message is transmitted. This connection, once established, remains exclusive and continuous for the complete duration of information exchange and the circuit becomes disconnected only when the source wants to do so.

Switching Node

Let us consider the operation of a single circuit switched node comprising a collection of stations attached to a central switching unit, which establishes a dedicated path between any two devices that wish to communicate.

Major elements of a single-node network are summarized below:

Digital switch: That provides a transparent (full-duplex) signal path between any pair of attached devices.

Network interface: That represents the functions and hardware needed to connect digital devices to the network (like telephones).

Control unit: That establishes, maintains, and tears down a connection.

The simplified schematic diagram of a switching node is shown in Fig. An important characteristic of a circuit-switch node is whether it is *blocking* or *non-blocking*. A blocking network is one, which may be unable to connect two stations because all possible paths between them are already in use. A non-blocking network permits all stations to be connected (in pairs) at once and grants all possible connection requests as long as the called party is free. For a network that supports only voice traffic, a blocking configuration may be acceptable, since most phone calls are of short duration. For data applications, where a connection may remain active for hours, non-blocking configuration is desirable.

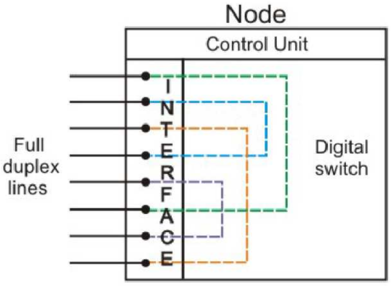
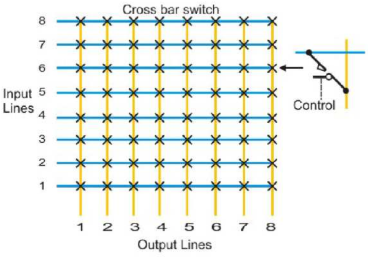


Fig 22:

Circuit switching uses any of the three technologies: Space-division switches, Time-division switches or a combination of both. In Space-division switching, the paths in the circuit are separated with each other spatially, i.e. different ongoing connections, at a same instant of time, uses different switching paths, which are separated spatially. This was originally developed for the analog environment, and has been carried over to the digital domain. Some of the space switches are crossbar switches, Multi-stage switches (e.g. Omega Switches). A crossbar switch is shown in Fig. 4.1.4. Basic building block of the switch is a metallic crosspoint or semiconductor gate that can be enabled or disabled by a control unit.



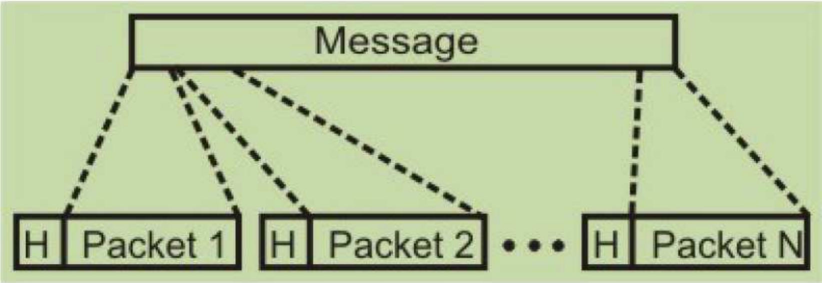
In circuit switching, network resources are dedicated to a particular connection. Although this satisfies the requirement of voice communication, it suffers from the following two shortcomings for data communication:

o In a typical user/host data connection, line utilization is very low.
o Provides facility for data transmission at a constant rate.

However, for information transmission applications, the circuit switching method is very slow, relatively expensive and inefficient. First of all, the need to establish a dedicated connection before sending the message itself inserts a delay time, which might become significant for the total message transfer time. Moreover, the total channel remains idle and unavailable to the other users once a connection is made. On the other hand once a connection is established, it is guaranteed and orderly delivery of message is ensured. Unfortunately, the data transmission pattern may not ensure this, because data transmission is bursty in nature. As a consequence, it limits the utility of the method To overcome the limitations of message switching, another switching technique, known as packet switching was invented

Packet Switching

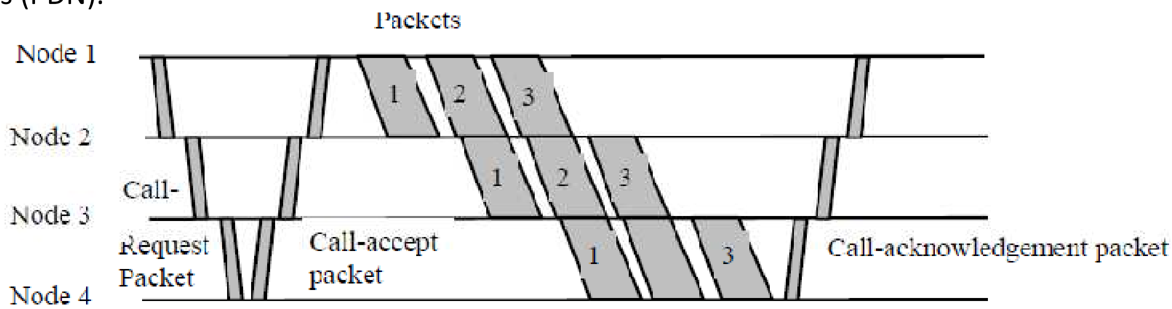
The basic approach is not much different from message switching. It is also based on the same ‘store-and-forward’ approach. However, to overcome the limitations of message switching, messages are divided into subsets of equal length called packets. This approach was developed for long-distance data communication (1970) and it has evolved over time. In packet switching approach, data are transmitted in short packets (few Kbytes). A long message is broken up into a series of packets as shown in Fig. 4.2.2. Every packet contains some control information in its header, which is required for routing and other purposes.



Main difference between Packet switching and Circuit Switching is that the communication lines are not dedicated to passing messages from the source to the destination. In Packet Switching, different messages (and even different packets) can pass through different routes, and when there is a "dead time" in the communication between the source and the destination, the lines can be used by other sources. There are two basic approaches commonly used to packet Switching: virtual-circuit packet switching and datagram packet switching. In virtual-circuit packet switching a virtual circuit is made before actual data is transmitted, but it is different from circuit switching in a sense that in circuit switching the call accept signal comes only from the final destination to the source while in case of virtual-packet switching this call accept signal is transmitted between each adjacent intermediate node as shown in Fig. Other features of virtual circuit packet switching are discussed in the following subsection.

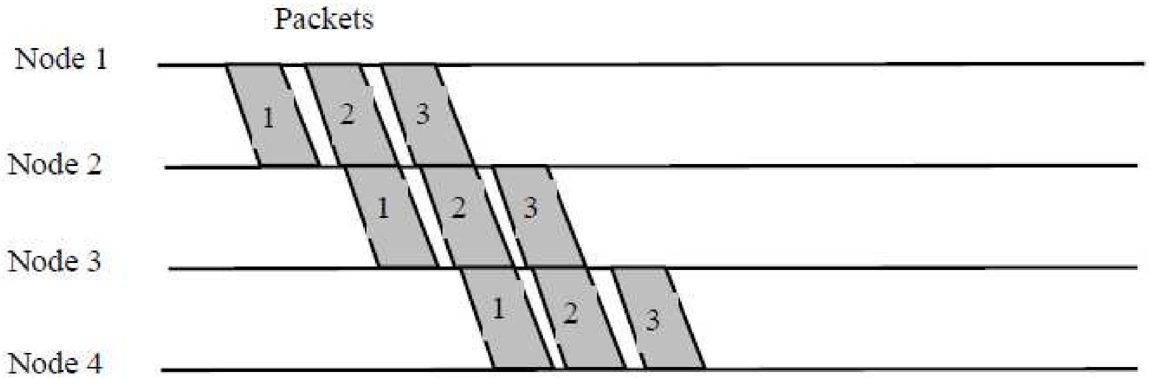
Virtual Circuit Packet Switching Networks

An initial setup phase is used to set up a route between the intermediate nodes for all the packets passed during the session between the two end nodes. In each intermediate node, an entry is registered in a table to indicate the route for the connection that has been set up. Thus, packets passed through this route, can have short headers, containing only a *virtual circuit identifier* (VCI), and not their destination. Each intermediate node passes the packets according to the information that was stored in it, in the setup phase. In this way, packets arrive at the destination in the correct sequence, and it is guaranteed that essentially there will not be errors. This approach is slower than Circuit Switching, since different virtual circuits may compete over the same resources, and an initial setup phase is needed to initiate the circuit. As in Circuit Switching, if an intermediate node fails, all virtual circuits that pass through it are lost. The most common forms of Virtual Circuit networks are X.25 and Frame Relay, which are commonly used for public data networks (PDN).



Datagram Packet Switching Networks

This approach uses a different, more dynamic scheme, to determine the route through the network links. Each packet is treated as an independent entity, and its header contains full information about the destination of the packet. The intermediate nodes examine the header of the packet, and decide to which node to send the packet so that it will reach its destination. In the decision two factors are taken into account: The shortest ways to pass the packet to its destination - protocols such as RIP/OSPF are used to determine the shortest path to the destination. Finding a free node to pass the packet to - in this way, bottlenecks are eliminated, since packets can reach the destination in alternate routes. Thus, in this method, the packets don't follow a pre-established route, and the intermediate nodes (the routers) don't have pre-defined knowledge of the routes that the packets should be passed through.



Packets can follow different routes to the destination, and delivery is not guaranteed (although packets usually do follow the same route, and are reliably sent). Due to the nature of this method, the packets can reach the destination in a different order than they were sent, thus they must be sorted at the destination to form the original message. This approach is time consuming since every router has to decide where to send each packet. The main implementation of Datagram Switching network is the Internet, which uses the IP network protocol.

The X .25 Protocol :

X.25 is an ITU-T standard protocol suite for packet switched wide area network (WAN) communication. An X.25 WAN consists of packet-switching exchange (PSE) nodes as the networking hardware, and leased lines, plain old telephone service connections or ISDN connections as physical links. X.25 is a family of protocols that was popular during the 1980s with telecommunications companies and in financial transaction systems such as automated teller machines. X.25 was originally defined by the International Telegraph and Telephone Consultative Committee (CCITT, now ITU-T) in a series of drafts and finalized in a publication known as The Orange Book in 1976. While X.25 has, to a large extent, been replaced by less complex protocols, especially the Internet protocol (IP), the service is still used and available in niche and legacy applications

Architecture

The general concept of X.25 was to create a universal and global packet-switched network. Much of the X.25 system is a description of the rigorous error correction needed to achieve this, as well as more efficient sharing of capital-intensive physical resources. The X.25 specification defines only the interface between a subscriber (DTE) and an X.25 network (DCE). X.75, a very similar protocol to X.25, defines the interface between two X.25 networks to allow connections to traverse two or more networks. X.25 does not specify how the network operates internally—many X.25 network implementations used something very similar to X.25 or X.75 internally, but others used quite different protocols internally. The ISO equivalent protocol to X.25, ISO 8208, is compatible with X.25, but additionally includes provision for two X.25 DTEs to be directly connected to each other with no network in between. By separating the Packet-Layer Protocol, ISO 8208 permits operation over additional networks such as ISO 8802 LLC2 (ISO LAN) and the OSI data link layer. X.25 originally defined three basic protocol levels or architectural layers. In the original specifications these were referred to as levels and also had a level number, whereas all ITU-T X.25 recommendations and ISO 8208 standards released after 1984 refer to them as layers. The layer numbers were dropped to avoid confusion with the OSI Model layers. Physical layer: This layer specifies the physical, electrical, functional and procedural characteristics to control the physical link between a DTE and a DCE. Common implementations use X.21, EIA-232, EIA-449 or other serial protocols.

- Data link layer: The data link layer consists of the link access procedure for data interchange on the link between a DTE and a DCE. In its implementation, the Link Access Procedure, Balanced (LAPB) is a data link protocol that manages a communication session and controls the packet framing. It is a bit-oriented protocol that provides error correction and orderly delivery.
- Packet layer: This layer defined a packet-layer protocol for exchanging control and user data packets to form a packet-switching network based on virtual calls, according to the Packet Layer Protocol.

The X.25 model was based on the traditional telephony concept of establishing reliable circuits through a shared network, but using software to create "virtual calls" through the network. These calls interconnect "data terminal equipment" (DTE) providing endpoints to users, which looked like point-to-point connections. Each endpoint can establish many separate virtual calls to different endpoints. For a brief period, the specification also included a connectionless datagram service, but this was dropped in the next revision. The "fast select with restricted response facility" is intermediate between full call establishment and connectionless communication. It is widely used in query-response transaction applications involving a single request and response limited to 128 bytes of data carried each way. The data is carried in an extended call request packet and the response is carried in an extended field of the call reject packet, with a connection never being fully established. Closely related to the X.25 protocol are the protocols to connect asynchronous devices (such as dumb terminals and printers) to an X.25 network: X.3, X.28 and X.29. This functionality was performed using a Packet Assembler/Disassembler or PAD (also known as a Triple-X device, referring to the three protocols used).

Error control

Error recovery procedures at the packet layer assume that the data link layer is responsible for retransmitting data received in error. Packet layer error handling focuses on resynchronizing the information flow in calls, as well as clearing calls that have gone into unrecoverable states: Level 3 Reset packets, which re-initializes the flow on a virtual call (but does not break the virtual call).Restart packet, which clears down all virtual calls on the data link and resets all permanent virtual circuits on the data link.

Addressing and virtual circuits

An X.25 Modem once used to connect to the German Datex-P network.

X.25 supports two types of virtual circuits, virtual calls (VC) and permanent virtual circuits (PVC). Virtual calls are established on an as-needed basis. For example, a VC is established when a call is placed and turn down after the call is complete. VCs are established through a call establishment and clearing procedure. On the other hand, permanent virtual circuits are preconfigured into the network.[20] PVCs are seldom turn down and thus provide a dedicated connection between end points. VC may be established using X.121 addresses. The X.121 address consists of a three-digit data country code (DCC) plus a network digit, together forming the four-digit data network identification code (DNIC), followed by the national terminal number (NTN) of at most ten digits. Note the use of a single network digit, seemingly allowing for only 10 network carriers per country, but some countries are assigned more than one DCC to avoid this limitation. Networks often used fewer than the full NTN digits for routing, and made the spare digits available to the subscriber (sometimes called the sub-address) where they could be used to identify applications or for further routing on the subscribers networks. NSAP addressing facility was added in the X.25(1984) revision of the specification, and this enabled X.25 to better meet the requirements of OSI Connection Oriented Network Service (CONS).[21] Public X.25 networks were not required to make use of NSAP addressing, but, to support OSI CONS, were required to carry the NSAP addresses and other ITU-T specified DTE facilities transparently from DTE to DTE. Later revisions allowed multiple addresses in addition to X.121 addresses to be carried on the same DTE-DCE interface: Telex addressing (F.69), PSTN addressing (E.163), ISDN addressing (E.164), Internet Protocol addresses (IANA ICP), and local IEEE 802.2 MAC addresses. VCs are permanently established in the network and therefore do not require the use of addresses for call setup. PVCs are identified at the subscriber interface by their logical channel identifier (see below). However, in practice not many of the national X.25 networks supported PVCs.One DTE-DCE interface to an X.25 network has a maximum of 4095 logical channels on which it is allowed to establish virtual calls and permanent virtual circuitsalthough networks are not expected to support a full 4095 virtual circuits. For identifying the channel to which a packet is associated, each packet contains a 12 bit logical channel identifier made up of an 8-bit logical channel number and a 4-bit logical channel group number. Logical channel identifiers remain assigned to a virtual circuit for the duration of the connection. Logical channel identifiers identify a specific logical channel between the DTE (subscriber appliance) and the DCE (network), and only has local significance on the link between the subscriber and the network. The other end of the connection at the remote DTE is likely to have assigned a different logical channel identifier. The range of possible logical channels is split into 4 groups: channels assigned to permanent virtual circuits, assigned to incoming virtual calls, two-way (incoming or outgoing) virtual calls, and outgoing virtual calls. (Directions refer to the direction of virtual call initiation as viewed by the DTE—they all carry data in both directions.). The ranges allowed a subscriber to be configured to handle significantly differing numbers of calls in each direction while reserving some channels for calls in one direction. All International networks are required to implement support for permanent virtual circuits, two-way logical channels and one-way logical channels outgoing; one-way logical channels incoming is an additional optional facility. DTE-DCE interfaces are not required to support more than one logical channel. Logical channel identifier zero will not be assigned to a permanent virtual circuit or virtual call. The logical channel identifier of zero is used for packets which don't relate to a specific virtual circuit (e.g. packet layer restart, registration, and diagnostic packets).lower price-per-segment than VCs, making them cheaper only where large volumes of data are passed.

X.25 packet types

Packet Type	DCE → DTE	DTE → DCE	Service VC	PVC
Call setup and Clearing	Incoming Call	Call Request	X	
	Call Connected	Call Accepted	X	
	Clear Indication	Clear Request	X	
	Clear Confirmation	Clear Confirmation	X	
Data and Interrupt	Data	Data	X	X
	Interrupt	Interrupt	X	X
	Interrupt Confirmation	Interrupt Confirmation	X	X
Flow Control and Reset	RR	RR	X	X

	RNR	RNR		X	X
	REJ	REJ		X	X
	Reset Indication	Reset Request		X	X
	Reset Confirmation	Reset Confirmation		X	X
Restart	Restart Indication	Restart Request	X		
	Restart Confirmation	Restart Confirmation	X		
Diagnostic	Diagnostic		X		
Registration	Registration Confirmation	Registration Request	X		

X.25 details

The network may allow the selection of the maximal length in range 16 to 4096 octets (2^n values only) per virtual circuit by negotiation as part of the call setup procedure. The maximal length may be different at the two ends of the virtual circuit.

- Data terminal equipment constructs control packets which are encapsulated into data packets. The packets are sent to the data circuit-terminating equipment, using LAPB Protocol.
- Data circuit-terminating equipment strips the layer-2 headers in order to encapsulate packets to the internal network protocol.

X.25 facilities

X.25 provides a set of user facilities defined and described in ITU-T Recommendation X.2.[35] The X.2 user facilities fall into five categories:

- essential facilities;
- additional facilities;
- conditional facilities;
- mandatory facilities; and,
- optional facilities.

X.25 also provides X.25 and ITU-T specified DTE optional user facilities defined and described in ITU-T Recommendation X.7.[36] The X.7 optional user facilities fall into four categories of user facilities that require:

- subscription only;
- subscription followed by dynamic invocation;
- subscription or dynamic invocation; and,
- dynamic invocation only.

GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

Data Communication and Computer Network

For 4th Semester

ELECTRONICS AND TELECOMMUNICATION

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Mr. Sarada Prasanna Singh

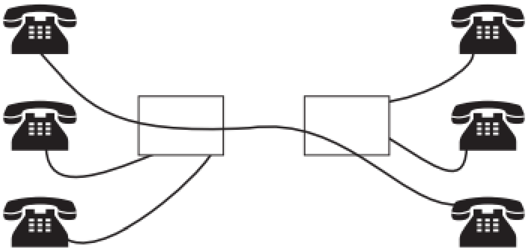
(Lecturer in Electronics & Telecommunication Engineering)

Module-I: OVERVIEW OF SWITCHING SYSTEMS

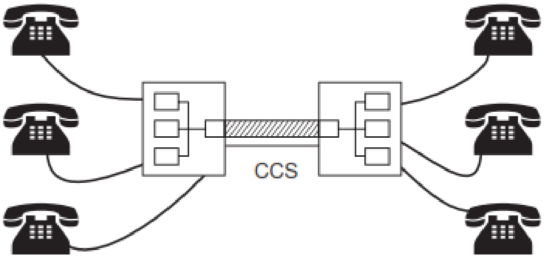
Electronic switching and stored program control systems, Centralized SPC, Availability, Distributed SPC, Enhanced services, Digital switching: time switching, space switching, time and space switches, Switching techniques: Circuit Switching, Message and Packet Switching

Telecommunication networks carry information signals among entities, which are geographically far apart. An entity may be a computer or human being, a facsimile machine, a teleprinter, a data terminal and so on. The entities are involved in the process of information transfer which may be in the form of a telephone conversation (telephony) or a file transfer between two computers or message transfer between two terminals etc. Today it is almost true to state that telecommunication systems are the symbol of our information age. With the rapidly growing traffic and untargeted growth of cyberspace, telecommunication becomes a fabric of our life. The future challenges are enormous as we anticipate rapid growth items of new services and number of users. What comes with the challenge is a genuine need for more advanced methodology supporting analysis and design of telecommunication architectures. Telecommunication has evaluated and growth at an explosive rate in recent years and will undoubtedly continue to do so. The communication switching system enables the universal connectivity. The universal connectivity is realized when any entity in one part of the world can communicate with any other entity in another part of the world. In many ways telecommunication will acts as a substitute for the increasingly expensive physical transportation. The telecommunication links and switching were mainly designed for voice communication. With the appropriate attachments/equipment, they can be used to transmit data. A modern society, therefore needs new facilities including very high bandwidth switched data networks, and large communication satellites with small, cheap earth antennas.

The use of computers to control the switching led to the designation ‘‘electronic’’ switching system (ESS) or Electronic automatic exchange (EAX). In 1970, first electronic switching system No. 1 ESS or No. 1 EAX was introduced. Digital electronic switching matrices were first introduced into the U.S. Public network in 1976 with AT & T’s No. 4 ESS digital toll switch. By the mid 1980’s the interoffice transmission environment has changed to almost exclusively digital. Fig. 1.1 shows the various telephone networks.



(a) Telephone network around 1890



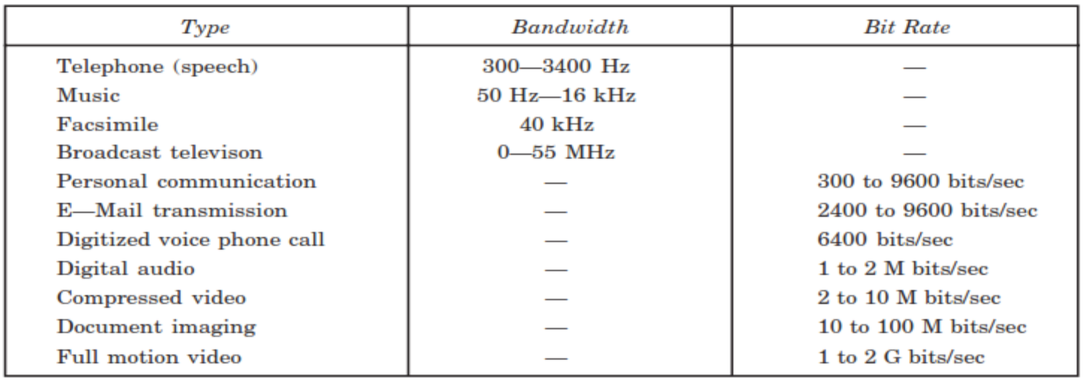
(b) Telephone network around 1988

Fig. 1.1. Various telephone networks

Telecommunication is mainly concerned with the transmission of messages between two distant points. The signal that contains the messages is usually converted into electrical waves before transmission. Our voice is an analog signal which has amplitude and frequency characteristic.

Voice frequencies. The range of frequencies used by a communication device determines the communication channel, communicating devices, bandwidth or information carrying capacity. The most commonly used parameter that characterizes an electrical signal is its bandwidth of analog signal or bit rate if it is a digital signal. In telephone system, the frequencies it passes are restricted to between 300 to 3400 Hz. Thus the network bandwidth is 3100 Hz. The bandwidth and bit rate for various types of system are shown in Table 1.1.

Table 1.1. Bandwidth requirements of various applications



Type	*Bandwidth*	*Bit Rate*
Telephone (speech)	300—3400 Hz	—
Music	50 Hz—16 kHz	—
Facsimile	40 kHz	—
Broadcast televison	0—55 MHz	—
Personal communication	—	300 to 9600 bits/sec
E—Mail transmission	—	2400 to 9600 bits/sec
Digitized voice phone call	—	6400 bits/sec
Digital audio	—	1 to 2 M bits/sec
Compressed video	—	2 to 10 M bits/sec
Document imaging	—	10 to 100 M bits/sec
Full motion video	—	1 to 2 G bits/sec

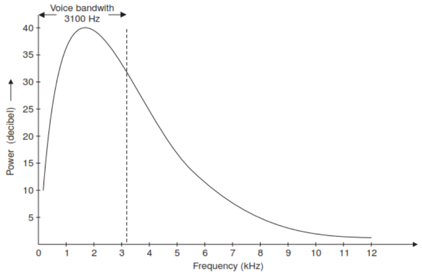


Fig. 1.2. Speech Spectrum.

Speech spectrum. The telephone channel over which we wish to send data are designed to transmit electrical oscillations (microphone converts sound into equivalent number of electrical oscillation) of voice. Fig. 1.2 is described as a speech spectrum diagram. It illustrates human speech strength variations at various frequencies. Most of the energy is concentrated between 300 Hz to 3400 Hz.

ELEMENTS OF COMMUNICATION SWITCHING SYSTEM

The purpose of a telecommunication switching system is to provide the means to pass information from any terminal device to any other terminal device selected by the originator.

Telecommunication system can be divided into four main parts. They are

1. End system or Instruments
2. Transmission system
3. Switching system
4. Signaling.

CRITERIA FOR THE DESIGN OF TELECOMMUNICATION SYSTEM

Traditionally, the design for telephone switching centre or equipment requirement in a telecommunication system are determined on the basis of the traffic intensity of the busy hour. The traffic intensity is defined as the product of the calling rate and the average holding time. The busy hour is defined as that continuous sixty-minute period during which the traffic intensity is highest. The calling rate is the average number of request for connection that are made per unit time. If the instant in time that a call request arises is a random variable, the calling rate maybe stated as the probability that a call request will occur in a certain short interval of time. The holding time is the mean time that calls last. Otherwise the average holding time is the average duration of occupancy of traffic path by a call.

Grade of Service. In telephone field, the so called busy hour traffic are used for planning purposes. Once the statistical properties of the traffic are known, the objective for the performance of a switching system should be stated. This is done by specifying a grade of service (GOS). GOS is a measure of congestion expressed as the probability that a call will be blocked or delayed. Thus when dealing with GOS in traffic engineering, the clear understanding of blocking criteria, delay criteria and congestion are essential.

Blocking criteria. If the design of a system is based on the fraction of calls blocked (the blocking probability), then the system is said to be engineered on a blocking basis or call loss basis. Blocking can occur if all devices are occupied when a demand of service is initiated. Blocking criteria are often used for the dimensioning of switching networks and interoffice trunk groups. For a system designed on a loss basis, a suitable GOS is the percentage of calls which are lost because no equipment is available at the instant of call request.

Delay criteria. If the design of a system is based on the fraction of calls delayed longer than a specified length of time (the delay probability), the system is said to be a waiting system or engineered on a delay basis. Delay criteria are used in telephone systems for the dimensioning of registers. In waiting system, a GOS objective could be either the percentage of calls which are delayed or the percentage which are delayed more than a certain length of time.

Congestion. It is the condition in a switching centre when a subscriber cannot obtain a connection to the wanted subscriber immediately. In a circuit switching system, there will be a period of congestion during which no new calls can be accepted. There are two ways of specifying congestion.

1. **Time congestion.** It is the probability that all servers are busy. It is also called the probability of blocking.

2. **Call congestion.** It is the proportion of calls arising that do not find a free server. Call congestion is a loss system and also known as the probability of loss while in a delay system it is referred to as the probability of waiting. If the number of sources is equal to the number of servers, the time congestion is finite, but the call congestion is zero. When the number of sources is large in comparison with servers, the probability of a new call arising is independent of the number already in progress and therefore, the call congestion is equal to the time congestion. In general, time and call congestions are different but in most practical cases, the discrepancies are small.

Measure of GOS. GOS is expressed as a probability. The GOS of 2% (0.02) mean that98% of the calls will reach a called instrument if it is free. Generally, GOS is quoted as P.02 or simply P02 to represent a network busy probability of 0.02. GOS is applied to a terminal-to terminal connection. For the system connection many switching centres, the system is generally broken into following components.(*i*) an internal call (calling subscriber to switching office)(*ii*) an outgoing call to the trunk network (switching office to trunk)(*iii*) The trunk network (trunk to trunk)(*iv*) A terminating call (switching office to called subscriber)The GOS of each component is called component GOS. The GOS for internal calls is 3 to 5%, for trunk calls 1-3%, for outgoing calls 2% and for terminating calls 2%. The overall GOS of a system is approximately the sum of the component grade of service. In practice, in order to ensure that the GOS does not deteriorate disastrously if the actual busy hour traffic exceeds the mean, GOS are specified 10% or 20% more of the mean.

A telephone network is composed of a variety of all processing equipment, interstate switching links and inters office trunks. Because of the random nature of the call request, the design of equipment switching links and trunks are quite difficult. Thus, the traffic analysis is the fundamental request for the design of cost effective, efficient and effective configuration of networks. The effectiveness of a network can be evaluated in terms of how much traffic it carries under normal or average loads and how often the traffic volume exceeds the capacity of the network. Fundamental problem in the design of telecommunication networks concerns the dimensioning of a route. To dimension the route, volume of traffic required grade of service and capacity (in bits per sec) must be known.

Traffic. In telecommunication system, traffic is defined as the occupancy of the server in the network. There are two types of traffic viz. voice traffic and data traffic. For voice traffic, the calling rate is defined as the number of calls per traffic path during the busy hour. In a day, the 60 minutes interval in which the traffic is highest is called busy hour (BH).

Average occupancy. If the average number of calls to and from a terminal during a period T second is '*n*' and the average holding time is '*h*' seconds, the average occupancy of the terminal is given by

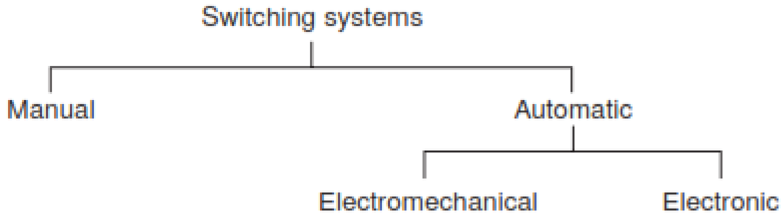
$$\mathrm{A} = \frac{nh}{\mathrm{T}}$$

The average occupancy is also referred as traffic flow of traffic intensity. The international unit of telephone traffic is the Erlang.

Telecommunication system is an important and integral part of modern society. In addition to public switched telephone network (PSTN), it plays vital role in radio and television networks, internet and Asynchronous transfer mode (ATM) networks. The switching system provides various services to the subscribers. The switching system is a collection of switching elements arranged and controlled in such a way as to setup a communication path between any two distant points. This chapter demonstrates the switching systems of manual exchanges to the electronic switching systems. The process of transferring message from one place to another (or line to line) is called switching related to outside the switching plant or systems. There are three types of switching namely a circuit switching, message switching and packet switching. In telecommunication switching, the circuit switching and message switching are used. The switching technique used in computer communication network or data transfer is packet switching. Telecommunication is the communication of voice or data over long distances using public switched telephone network (PSTN). PSTN consists of transmission component, switching components and facilities for maintaining equipment, billing system and other internal components. PSTN also referred to as plain old telephone system (POTS). The switching technique used in PSTN is circuit switching in general. To setup connection between subscribers, the PSTN consists of the transmission systems, switching system and signalling systems.

Classification of Switching System

In early days, the human exchange provided switching facilities. In manual exchanges, a human operator and the elements like switches, plugs and sacks were used to connect two subscribers. Around 1890's many electromechanical switching devices were introduced. Till 1940, different electromechanical switching system were invented, of which strowger switching system and cross bar switching system were still popular. The later invention of electronic switching system (ESS) which uses stored program control (SPC) and computer controlled switching systems are presently dominating the worldwide exchanges. Fig. shows the classification of switching system.



The electronic switching system (ESS) uses stored program control. The further classification of ESS are space division switching and time division switching. The time division switching is divided into digital and analog switching systems. The digital switching system is classified into space switch, time switch and combination switch.

Requirements of Switching System

All practical switching system should satisfy the following requirements for the economic use of the equipment of the system and to provide efficient service to the subscribers. Depends on the place (Rural or town, big town, city or big cities). The local exchange located, the service provided to the subscriber may vary. Some important requirements are discussed briefly.

High availability. The telephone system must be very reliable. System reliability can be expressed mathematically as the ratio of uptime to sum of the uptime and down time. The uptime is the total time that the system is operating satisfactorily and the down time is the total time that is not. In telephone switching networks, the availability or full accessibility is possible if all of the lines are equally accessible to all incoming calls. The full accessibility is also defined as the capacity or number of outlets of a switch to access a given route. If each incoming trunk has access to a sufficient number of trunks on each route to give the required grade of service is known as limited availability. The availability is defined as

$$A = \frac{\text{Uptime}}{\text{Uptime} + \text{down time}} \qquad A = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}$$

MTBF = Mean time between failure

MTTR = Mean time to repair.

The unavailability of the system is given by

$$U = 1 - A < \frac{\text{MTTR}}{\text{MTBF} + \text{MTTR}}$$

High speed. The switching speed should be high enough to make use of the switching system efficiently. The speed of switching depends on how quickly the control signals are transmitted. For instance, the seize signal from the calling terminal must be identified quickly by the system to realise the need of path setup by the subscriber. The common control should be used effectively to identify the called terminal or the free trunks to setup a path. Thus the switching system must have the facility of quick access of the switching equipment and networks.

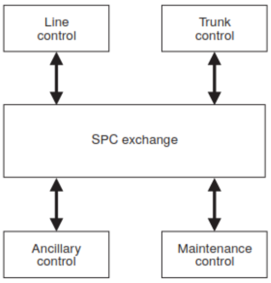
Low down time. The down time is the total time the switching system is not operating satisfactorily. The down time is low enough to have high availability. The unavailability of switching system may be due to failure of equipment, troubles in transmission media, and human errors in switching etc.

Good facilities. A switching system must have various facilities to serve the subscriber. For example wake up calls, address identification on phone number or phone number identification on address, recording facilities, quick service for the emergency numbers, good accessibility etc. Also it should have good servicing facilities in case of repair of equipment, skilled technicians, standby systems, etc. Good facilities is possible any switching system whether it is at rural or town or in cities, if that exchange is not overloaded.

High security. To ensure satisfied or correct operation (*i.e.* providing path and supervising the entire calls to pass necessary control signals) provision should be provided in the switching system.

Duplicated common control circuits, registers, processors and standby systems are used provide high security.

There are two classes of switching system based on the division of information in space, time. They are (*i*) Space division switch (*ii*) Time division switch. The space division provides fixed path for the entire duration of a call. Simply, unlimited bandwidth, cross talk limitations are the advantages of space division switches. But these space switches are slow to operate, bulky, and involves large amount of wiring. In time division switching all inlets and outlet one connected to a common switch mechanism. The switch is connected to the required inlet and outlet for short durations. Each input is sampled to change the connecting pattern. Thus switch is fast and compact. This technique may only be used where the signal is not affected by the sampling process. Time division switches of analog signals have limited applications. Thus time division switches have more practical value only when the signal is already in digital form.



In 1965, Bell system installed the first computer controlled switching system which uses a stored program digital computer for its control functions. The SPC concepts permits the features like abbreviated dialling, call forwarding, call waiting etc. The SPC provides significant advantages to end users. The SPC enables easier number changes, automated call tracing message unit accounting (for billing) etc.

In SPC, a programme or a set of instructions are stored in its memory and executed automatically one by one by the processor. Carrying out the exchange control functions through programs stored in the memory of a computer led to the name stored program control. A computer can be programmed to test the conditions of the inputs and last states and decide on new outputs and states. The decisions are expressed as programs which can be rewritten to modify or extend the functions of control system. All switching systems manufactured for use as public switching systems now use computers and software programming to control the switching of calls. Using SPC, 20 mA transmitter (old transmitter need 23 mA) with 52 V battery feed and longer subscriber loop can be achieved.

The SPC uses processors designed to meet the various requirements of the exchange. More than one processors are used for the reliability. Normally these processors are duplicated. Also the SPC system uses distributed software and hardware architectures. To carry over the maintenance functions of the switching system, a separate processor is used. Using the above setup, the SPC performs trunk routing to other control or tandem offices. Special features and functions are also enabled with sophisticated equipment’s and in compact form. There are two types in SPC exchanges, namely centralised SPC and distributed SPC.

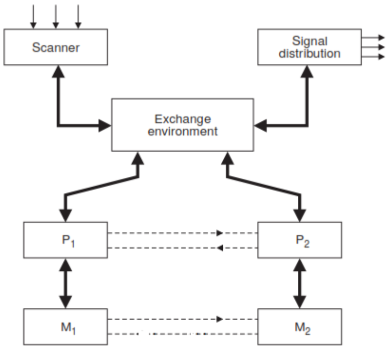


Fig. Centralized SPC.

Early electronic switching systems are centralised SPC exchanges and used a single processor to perform the exchange functions. Presently centralised exchanges uses dual processor for high reliability. All the control equipment are replaced by the processors. A dual processor architecture may be configured to operate in (*a*) standby mode (*b*) synchronous duplex mode and (*c*) Load sharing mode.

Standby mode. In this mode, any one of the processors will be active and the rest is standby. The standby processor is brought online only when the active processor fail. This mode of exchange uses a secondary storage common to both processors. The active processor copies the status of the system periodically and stores in axis secondary storage. In this mode the processors are not connected directly. In secondary storage, programs and instructions related to the control functions, routine programs and other required information are stored.

Synchronous duplex mode. In this mode, the processors *p*1 and *p2* are connected together to exchange instructions and controls. Instead of a secondary storage common to P1 and P2, separate memory M1 and M2 are used. These processors are coupled to exchange stored data. This mode of operation also uses a comparator in between *p2*. The comparator compares the result of the processors. During normal operation, both processors receives all the information from the exchange and receives related data from their memories. Although only one processor actually controls the exchange and remaining is in synchronism with first one. If a mismatch occurs, the fault is identified by the comparator, and the faulty processor is identified by operating both individually. After the rectification of fault, the processor is brought into service.

Load sharing mode. In this mode, the comparator is removed and alternatively an exclusion device (ED) is used. The processors calls for ED to share the resources, so that both the processors do not seek the same resource at the same time. In this mode, both the processor are active simultaneously and share the resources of exchange and the load dynamically. If one processor fails, with the help of ED, the other processor takes over the entire load of the exchange. Under normal operation, each processor handles one half of the calls on a statistical basis. However the exchange operator can vary the processor

Single processor.

Availability where MTBF = Mean time between failures, MTTR = Mean time to repair.

Unavailability = 1 – A $A = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}$

$$U = 1 - \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}} \; ; \; U = \frac{\text{MTTR}}{\text{MTBF} + \text{MTTR}}$$

If MTBF >> MTTR,

Dual Processor. A dual processor system is said to have failed only when both processor fails and the total system is unavailable. The MTBF of dual processor is given by

$$(\text{MTBF})_D = \frac{(\text{MTBF})^2}{2\text{MTTR}}$$

where $(\text{MTBF})_D$ = MTBF of dual processor, MTBF = MTBF single processor

Availability $$A_D = \frac{(\text{MTBF})_D}{\text{MTTR} + (\text{MTBF})_D}$$

Substituting $(\text{MTBF})_D$ in the above equation, we have

$$A_D = \frac{(\text{MTBF})^2 / 2\text{MTTR}}{\text{MTTR} + \dfrac{(\text{MTBF})^2}{2\text{MTTR}}}$$

$$A_D = \frac{(\text{MTBF})^2}{(\text{MTBF})^2 + 2(\text{MTTR})^2}$$

Unavailability $$U = 1 - A_D = 1 - \frac{(\text{MTBF})^2}{(\text{MTBF})^2 + 2(\text{MTTR})^2}$$

$$= \frac{2(\text{MTTR})^2}{(\text{MTBF})^2 + 2(\text{MTTR})^2}$$

If MTBF >> MTTR, $$U_D = \frac{2(\text{MTTR})^2}{(\text{MTBF})^2}$$

For example problems, please refer text books.

The introduction of **distributed SPC** enabled customers to be provided with a wider range of services than those available with centralised and electromechanical switching system. Instead of all processing being performed by a one or two processor in centralised switching, functions are delegated to separate small processors (referred as regional processors). But central processors is still required to direct the regional processors and to perform more complex tasks.

$$U = \frac{\text{MTTR}}{\text{MTBF}}$$

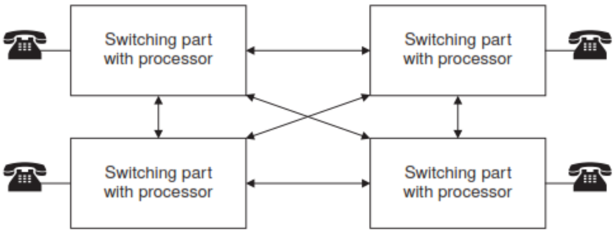

Switching part with processor
Switching part with processor
Switching part with processor
Switching part with processor

Fig. Distributed SPC

SWITCHING TECHNIQUES

This section describes various techniques used to establish connections between users' exchanges. Switches are hardware and/or software devices used to connect two or more users' temporarily. Message switching, circuit switching and packet switching are the most important switching methods. The terminals of the message switching systems are usually tele printers. In this switching, delays are incurred but no calls are lost as each messages are queued for each link. Thus, much higher link utilisation is achieved. The reason for the delay is that the system is designed to maximise the utilisation of transmission links by queueing message awaiting the use of a line. This switching is also called store and forward switching. The circuit switching sets up connection between the telephones, telex networks etc. which interchange information directly. If a subscriber or system to which connection to be made as engaged with other connection, path setup cannot be made. Thus circuit switching is also referred as lost call system. The modified form of message switching is called packet switching. Packet switching system carries data from a terminal or computer as a short packets of information to the required destination. This system is midway between message switching and circuit switching.

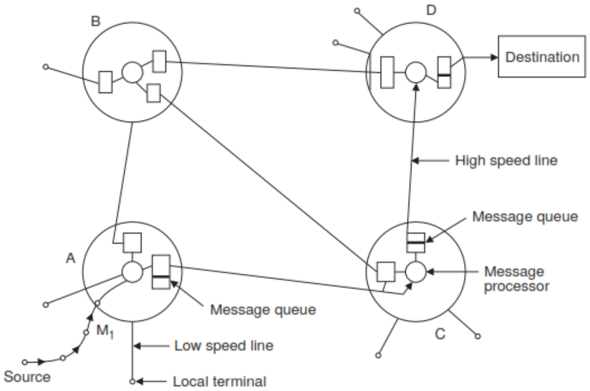
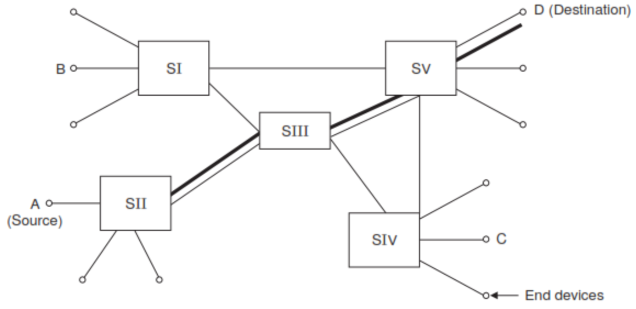
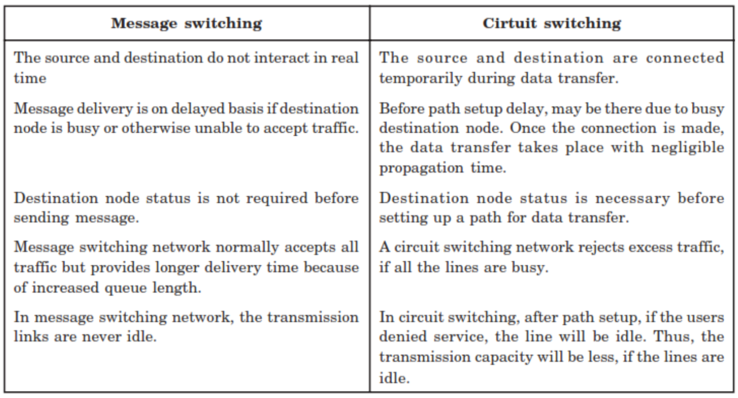


Fig. Message switching



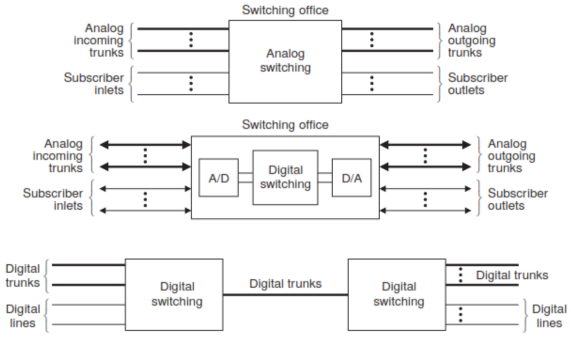
S1-SN → Switches, —Path between switches, ━Path setup between end device A&D,

Fig. Circuit switching



Message switching	Cirtuit switching
The source and destination do not interact in real time	The source and destination are connected temporarily during data transfer.
Message delivery is on delayed basis if destination node is busy or otherwise unable to accept traffic.	Before path setup delay, may be there due to busy destination node. Once the connection is made, the data transfer takes place with negligible propagation time.
Destination node status is not required before sending message.	Destination node status is necessary before setting up a path for data transfer.
Message switching network normally accepts all traffic but provides longer delivery time because of increased queue length.	A circuit switching network rejects excess traffic, if all the lines are busy.
In message switching network, the transmission links are never idle.	In circuit switching, after path setup, if the users denied service, the line will be idle. Thus, the transmission capacity will be less, if the lines are idle.

DIGITAL SWITCHING

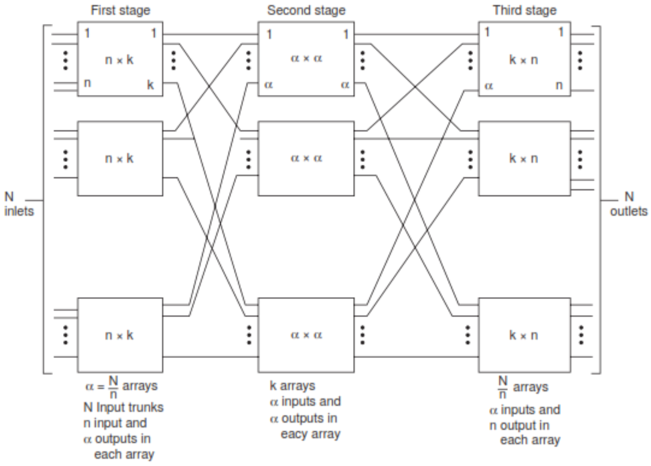


SPACE DIVISION SWITCHING

The fundamental operation of a switch is to setup and release connection between subscribers. It involves direct connection between subscriber loops at an end office or between station loops at a PBX. The switches are hardware and/or software devices capable of creating temporary connections between two or more subscribers. In space division switching, the paths in the circuit are separated from each other spatially. It was originally designed for analog networks, but is used currently in both digital and analog switching. A cross point switch is referred to as a space division switch because it moves a bit stream from one circuit/bus to another. For large group of outlets, considerable savings in total cross points can be achieved if each inlet can access only a limited number of outlets. Such situation is called limited availability. By overlapping the available outlet groups for various inlet groups, a technique called ‘‘grading’’ as established. Rectangular cross point array is an example of grading. For longer trunk groups, large cross points were expensive and not used now-a-days. The number of crosspoints required are M × N, where M is number of inlets and N is number of outlets.

Multistage Switching

It is inefficient to build complete exchanges in single stages. Single stage can only be used to interconnect one particular inlet outlet pair. Also the number of cross points grows as the square of the inputs for grading, N (N–1)/2 for a triangular array and N (N–1) for a square array. Also the large number of cross points on each inlet and outlet line imply a large amount of capacitive loading on the message paths. Therefore, it is usual to build exchanges in two or three stages to reduce the number of cross points and to provide alternative paths. The sharing of cross points for potential paths through the switch is accomplished by multiple stage switching. Fig. shows the three stage switching structure to accommodate 128 input and 128 output terminals with 16 first stage and last stage.



The structure shown in Fig. provides path for N inlets and N outlets. The N input lines are divided into N/*n* groups of *n* lines each. Each group of *n* inputs is accommodated by an *n*-input, *k* output matrix. The output matrices are identical to the input matrices except they are reversed. The intermediate stages are *k* in number and N/*n* inputs and N/*n* outputs. The interstage connections are often called junctors. Each of the *k* paths utilizes a separate center stage array. An arbitrary input can find *k* alternate output. Thus multistage structure provides alternate paths. Also the switching link is connected to a limited number of crosspoints. This enables the minimized capacitive loading.

The total number of crosspoints N_X for three stage is

$$N_X = 2NK + K\left(\frac{N}{n}\right)^2$$

where N = Number of inlets-outlets

n = size of each inlet-outlet group

k = number of second stage.

2Nk = number of cross points in 1st and 2nd stage

$\left(\frac{N}{n}\right)^2$ = number of cross points in each array of second stage

$k\left(\frac{N}{n}\right)^2$ = number of cross points in second stage.

Also $$N_X = K\left[2N + \left(\frac{N}{n}\right)^2\right]$$

The three stage switching matrix require that $k > 2n - 1$ to generate no blocking.

$k = 2n - 1$

Substituting we get

$$N_X = 2N\,(2n - 1) + (2n - 1)\left(\frac{N}{n}\right)^2$$

For large N, $n = \sqrt{\frac{N}{2}}$

Substitute $\sqrt{\frac{N}{2}}$ for n

Thus $$N_x = 2N\,(2\sqrt{n/2} - 1) + (2\sqrt{N/2} - 1)\left(\frac{N}{\sqrt{\frac{N}{2}}}\right)^2$$

$$= 2N\,(\sqrt{2n} - 1) + (\sqrt{2n} - 1)\;2N$$

$$N_{x(\min)} = 4N\,(\sqrt{2n} - 1)$$

Number of cross points for a single stage switching matrix to connect N inlets to N outlets is N*x* (SS)=N^2

$$\lambda \equiv \frac{N_x\,(\text{min, 3 stage})}{N_x\,(\text{SS})} = \frac{4\sqrt{2}\,N^{3/2}}{N^2}\ .$$

$$\lambda = \frac{4\sqrt{2}}{\sqrt{N}}$$

where N*x* (SS) = Number of cross points in single stage. In practice, this gives reasonable scaling.

Blocking Probability Evaluation Techniques

All the switching systems are designed to provide a certain maximum probability of blocking for the busiest hour of the day. It is one of the aspects of the grade of service of the telephone company. There are variety of techniques to evaluate the blocking probability of a switching matrix. Depends on the accuracy, required availability, geographical area, priority, complexity and applicability of different network structures, the techniques are varying. Here, two techniques are described.

1. **Lee graphs.** It was proposed by C.Y. Lee. It is a most versatile and straight forward approaches of calculating probabilities with the use of probability graphs.

2. **Jacobaeus method.** It was presented in 1950 by C. Jacobaeus. It is more accurate than Lee graph method.

Lee graphics. C.Y. Lee's approach of determining the blocking probabilities of various switching system is based on the use of utilization percentage or loadings of individual links. Let *p* be the probability that a link is busy. The probability that a link is idle is denoted by $q = 1 - p$. When any one of *n* parallel links can be used to complete a connection, the blocking probability B is the probability that all links are busy is given by $B = p^n$

when a series of *n* links are all needed to complete a connection, $B = 1 - q^n$For a probability graph of three stage network, shown in Fig. the probability of blocking is given by$B = (1 - q^2)^k$where q' = probability that an interstage link is idle = $1 - p'$ p' = probability that any particular interstage link is busy k = number of centre stage arrays.If *p* is known, the probability that an interstage link is busy is given by$p' = p^ß$ where ß = *k*/*n*ß is the factor by which the percentage of interstage links that are busy is reduced.

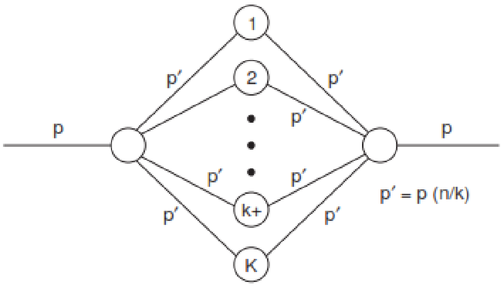


Fig. Probability graph of three stage network. Substituting we set $q' = 1 - p/ß$

Substituting again we get complete expression for the blocking probability of a three stage switch in terms *p* as

$$B = \left[1 - \left[1 - \frac{p}{\beta}\right]^2\right]^k$$

with inlets of 10% busy, the switch size of N with *n* = 8, *h* = 5, ß = 0.625 requires 2560 crosspoints. The merits of this method are(*i*) It provide accurate results(*ii*) Its formulas are directly relate to the network structures(*iii*) It provides insight of the network and thus provides ideas to change the structure for high performance.

Jacobaeus. The Lee's graph approach is not much accurate. Because the probability graphs entail several simplifying assumptions. The important one which gives erroneous values of blocking is the assumption that the individual probabilities are independent. In fact the probabilities not independent and highly dependent when significant amounts of expansion are not present. According to C. Jacobaeus the blocking probability of a three stage switch is

$$\mathrm{B} = \frac{(n\,!)^2}{k\,!\,(2n-k)\,!}\,p^k\,(2-p)^{2n-k}$$

where *n* = number of inlets (outlets) per first (third) stage array

k = number of second stage array

p = inlet utilization.

More accurate techniques can be used for systems with high concentrations and high blocking. As the high blocking probabilities not having much practical value, those techniques are not considered.

TIME DIVISION SWITCHING

In space division switching, crosspoints are used to establish a specific connection between two subscribers. The crosspoints of multistage space switches assigned to a particular connection is dedicated to that connection for its duration. Thus the crosspoints cannot be shared. Time division switching involves the sharing of crosspoints for shorter periods of time. This paves way for the reassign of crosspoints and its associated circuits for other needed connections. Therefore, in time division switching, greater savings in crosspoints can be achieved. Hence, by using a dynamic control mechanisms, a switching element can be assigned to many inlet-outlet pairs for few microseconds. This is the principle of time division switching. Time division switching uses time division multiplexing to achieve switching. Two popular methods that are used in time division multiplexing are (*a*) the time slot interchange (TSI) and (*b*) the TDM bus. In ordinary time division multiplexing, the data reaches the output in the same order as they sent. But TSI changes the ordering of slots based on the desired connections. The demultiplexer separates the stots and passes them to the proper outputs. The TDM uses a control unit. The control unit opens and closes the gates according to the switching need. The principle of time division switching can be equally applied to analog and digital signals. For interfacing sampled analog signals but not digitized, the analog time division switches are attractive. But for larger switches, there are some limitations due to noise, distortion and crosstalk which normally occurs in PAM signals. Thus analog switching is now used only in smaller switching systems.

Analog Time Division Switching

Fig. shows a simple analog time division switching structure. The speech is carried as PAM analog samples or PCM digital samples, occurring at 125 μs intervals. When PAM samples are switched in a time division manner, the switching is known as analog time division switching. If PCM binary samples are switched, then the switching is known as digital time division switching. A single switching bus supports a multiple number of connections by interleaving PAM samples from receive line interfaces to transmit line interfaces. There are two cyclic control stores. The first control store controls gating of inputs onto the bus one sample at a time. The second control store operates in synchronism with the first and selects the appropriate output line for each input sample.

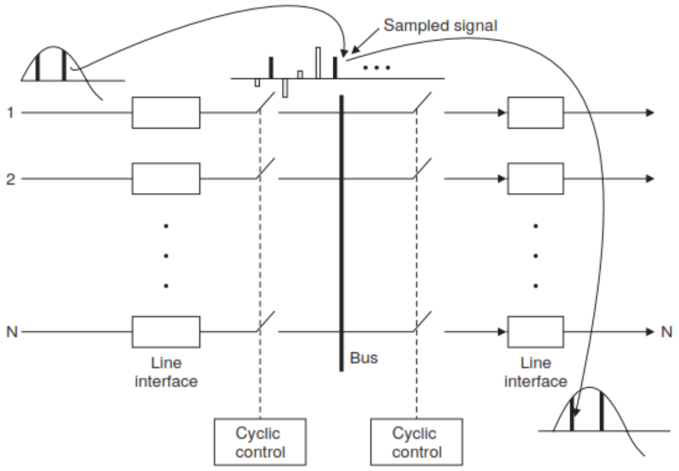
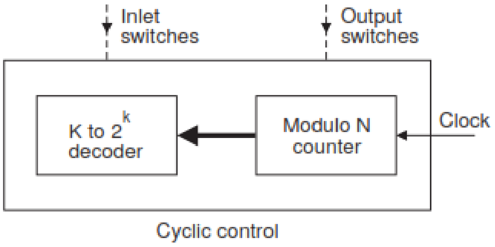


Fig. Analog Time Division Switching structure.

The selection of inlet/outlet is controlled by various ways. The (*a*) cyclic control and (*b*) memory based control are the important controls.



The cyclic control is organised by using Modulo-N counter and *k* to 2^k decoder. The *k* and N are related by

$\lceil \log_2 N \rceil = k$

where N = number of inlets/outlets

k = decoder size. It means *k* may be assumed lowest integer or more than that.

This kind of switching is non-blocking but lack of full availability as it is not possible to connect inlet to any outlet. The switching capacity or number of channel supported by cyclic controlled system is

$$C = \frac{125\,\mu\,\text{sec}}{t_s}$$

Memory based control. Full availability can be achieved if any one control is made memory based. If the input side is cyclically switched and the outlets are connected based on the addresses of the outlets stored in contiguous location is referred as input controlled or input driven. If the outlets are cyclically switched, the switch is referred as output controlled or output driven. As the physical connection is established between the inlet and the outlet through the common bus for the duration of one sample transfer, the switching technique is known as time division space multiplexing. For this system,

$$C = \frac{125\,\mu\,\text{sec}}{t_i + t_m + t_d + t_t}$$

t_m= time to read the control memory

t_d= time to decode address and select the inlet and outlet.

t_i= time to increment the modulo N counter.

t_t= time to transfer the sample.

The capacity equations are valid only for a 8 kHz sampling and non folded network (can be used for folded network with certain changes in network). The switching capacity in the memory controlled is equal to N. The use of cyclic control in input or output controlled switches restricts the number of subscribers on the system rather than the switching capacity since all the lines are scanned whether it is active or not. No restrictions on subscriber number and full availability of the switching system can be achieved by designing a switching configuration with control memory for controlling both inlets and outlets. This configuration referred to as memory controlled time division space switch is shown in Fig. As each word of the control memory has inlet address and an outlet address, the control memory width is $2^{\lceil \log_2 N \rceil}$. modulo counter is updated at the clock rate. For the path setup of addresses are entered in control memory and path is made. Then the location is marked busy. When conversation is terminated, the addresses are replaced by null values and location is marked free. Hence

$$C = \frac{125}{t_s}\,\mu\,\text{sec, where } t_s = t_i + t_m + t_d + t_t$$

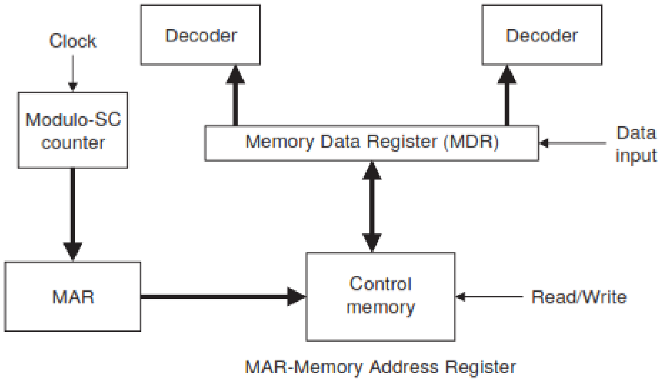


Fig. memory controlled for both input and output.

Digital Time Division Switching

The analog time division switching is useful for both analog and digital signals. The digital time division multiplexed signals usually requires switching between time slots as well as between physical lines. The switching between time slots are usually referred as time switching. Similar to analog time division switching the switching structure can be organised expect the use of memory block in place of the bus. This adds the serial to parallel and parallel to serial bit conversion circuitry's as the input to the memory block should be in parallel form. The time division switch can be controlled in any of the following three ways.

The basic requirement of time division switching is that the transfer of information arriving at in a time slot of one input link to other time slot of any one of output link. A complete set of pulses, arriving at each active input line is referred to as a frame. The frame rate is equal to the sample rate of each line. A time switch operates by writing data into and reading data out of a single memory. In the process the information in selected time slots is interchanged as shown in Fig.

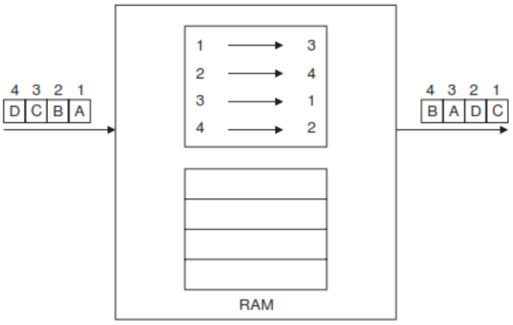


Fig. TSI operation

In TSI operation, inputs are sequentially controlled and outputs are selectively controlled. The RAM have several memory locations, each size is the same as of single time slot. Fig. shows the general arrangement of the time division time switching.

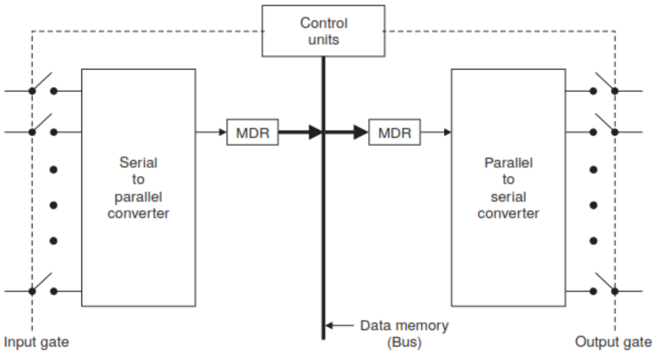
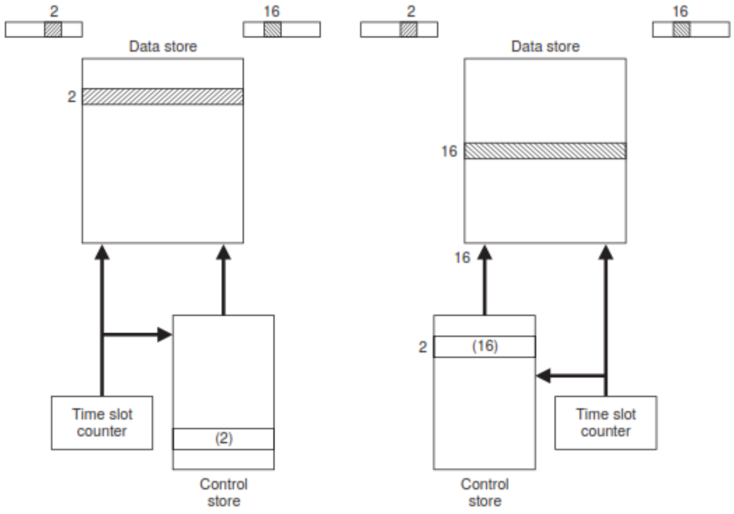


Fig. TDTS diagram

The serial to parallel and parallel to serial converter are used to write the data into the memory and read the data out of memory. For convenience, two MDR are shown, but MDR is a single register. Gating mechanism is used to connect the inlet/outlet to MDR. The input and output lines are connected to a high speed bus through input and output gates. Each input gate is closed during one of the four time slots. During the same time slot, only one output gate closed. This pair of gates allows a burst of data to be transferred from one input line to a specific output line through the bus. The control unit opens and closes the gates according to switching need. The time division time switch may be controlled by sequential write/random read or random write/ sequential read. Fig. depicts both modes of operation and indicates how the memories are accessed to translate information from time slot 2 to time slot 16. Both methods use a cyclic control. Fig. (*a*) implies that specific memory locations are dedicated to respective channels of the incoming TDM link. Data are stored in sequential locations in memory by incrementing modulo N counter with every time slot. Thus incoming data during time slot 2 is stored in the second location within the memory. On output, information retrieved from the control store specifies which address is to be accessed for that particular time slot. Thus sixteenth word of control store contains the number 2, implying that the contents of data store address 2 is transferred to the output link during outgoing slot 16. Random write/sequential read mode of operation is opposite to that of sequential write/random read. Incoming data are written into the memory locations as specified by the control store, but outgoing data are retrieved sequentially under control of an outgoing time slot counter. The data received during time slot 2 is written directly into data store address 16 and it is retrieved during outgoing TDM channel number 16.



(a) (b)

TWO DIMENSIONAL DIGITAL SWITCHING

Combination of the time and space switches leads to a configuration that achieved both timeslot interchange and sample switching across trunks. These structures also permit a large number of simultaneous connections to be supported for a given technology. Large digital switches require switching operations in both a space dimension and a time dimension. There are a large variety of network configurations that can be used to accomplish these requirements. The incoming and outgoing PCM highways are spatially separate. So the connection of one line of local exchange obviously requires space switching to connect to the channel of outgoing highways. Thus the switching network must be able to receive PCM samples from one time slot and retransmit them in a different time-slot. This is known as time slot interchange, or simply as time switching. Thus the switching network must perform both space and time switching. The space switching and time switching may be accomplished in many ways. A two stage combination switch may be organised with time switch as first stage and the space switch as the second stage or vice versa. The resulting configurations are referred as time space (TS) or space time (ST) switches respectively. Three stage time and space combinations of TST and STS configurations are more popular and flexible. Very large division switches includes many combinations of time and space switches. Typical configurations are TSST, TSSSST, and TSTSTSTS. These switches support 40000 lines or more economically. The general block diagram involving time and space switching is shown in Fig.

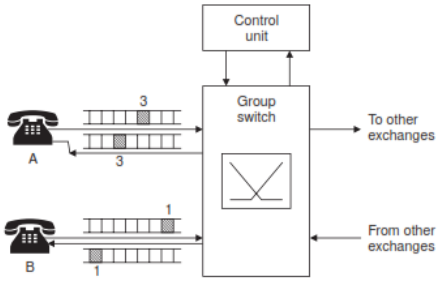


Fig. Combination switching

The main task of the switching part is to interconnect an incoming time slot and an outgoing time slot. The unit responsible for this function is group switch. There are two types of building block in the digital group switch. They are time switch and space switch. In Fig. the subscriber makes a local call to B. The control unit has assigned timeslot 3 to the call on its way into the group switch, and time slot 1 on its way out of the group switch (to B). This is maintained during the entire call. Similarly B to A also carried out. The fundamental design and structure of the two switches viz. time switch and space switch are described in the following sections.

Space and Time Switches

Space switch. Fig. shows a typical space switch. It uses a space array to provide switching generally the space switch consists of a matrix of M × N switching points where M is number of inlets and N is number of outlets. A connection between an inlet and an outlet is made by the simple logic gates (AND gates). As logic gates are unidirectional, two paths through switching matrix must be established to accommodate a two way conversation. The logic gate array can serve for concentration, expansion or distribution depending on M is larger, smaller or equal to N. Fig. shows only one voice direction. However, the corresponding components are available for the opposite direction too. A number of M, of X slot multiplexers, provide the inputs and the outlets are connected to N, X slot demultiplexers. The gate select memory has X locations. The word containing information about which cross point is to be enabled is decoded by the translator. During each internal time slot, one cross point is activated. In the shift to the next interval time slot, the control memory is incremented by one step, and a new crosspoint pattern is formed in the matrix.

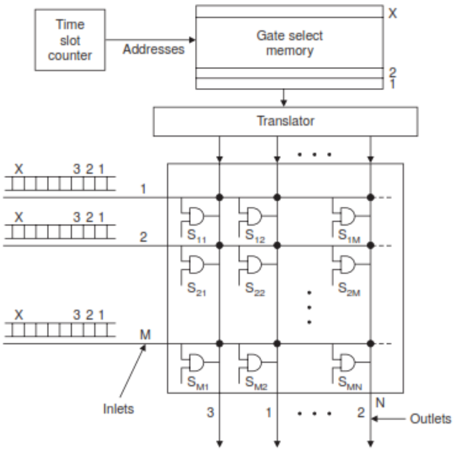


Fig. space switch

Time switch. The time-slot interchange (TSI) system is referred to as time switching (T-switching). Fig. shows the block diagram of time switch. Each incoming time slot is stored in sequence in a speech memory (SM). The control memory (CM) determines in which order the time slots are to be read from SM. This means that a voice sample may be moved from say incoming time slot 3 to outgoing time slot 1.

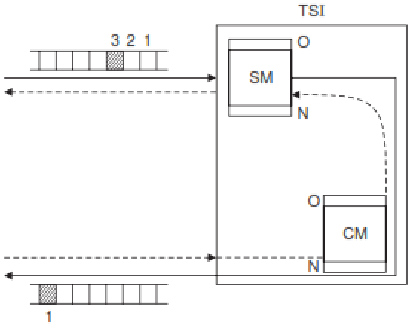


Fig. Time switch

Time-space (TS) Switching

This switch consists of only two stages. This structure contains a time stage T followed by a space stage S as shown in Fig. Thus this structure is referred to as time-space (TS) switch. The space array have N inlets and N outlets. For each inlet line, a time slot interchange with T slots is introduced. Each TSI is provided with a time slot memories (not shown). Similarly a gate select memory needs to be provided for the space array (not shown).

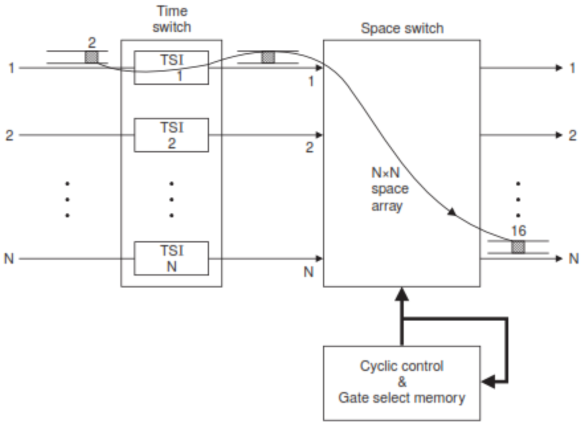
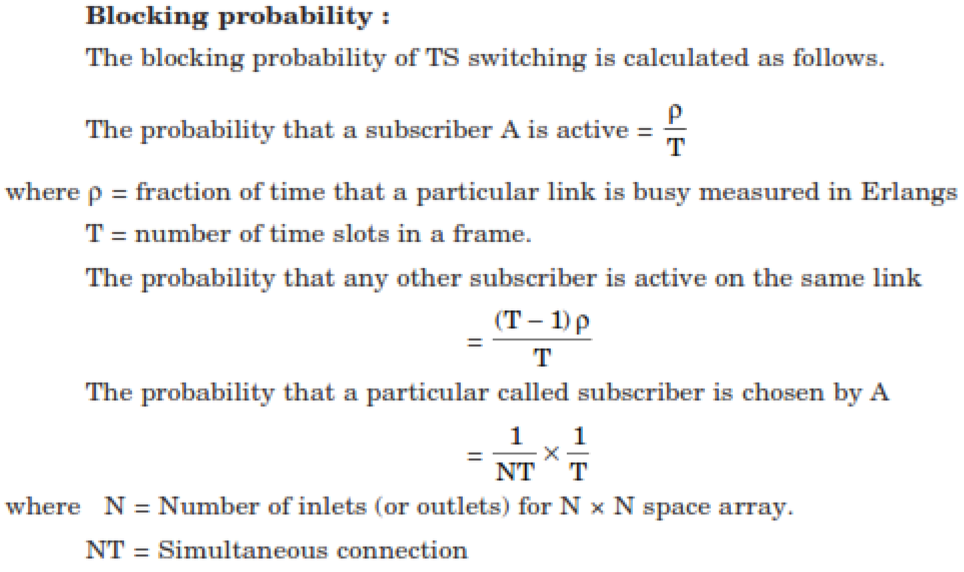


Fig. TS switch

The transmission of signals carried out from sender to receiver through multiplexer input and demultiplexer output. The reverse communication also similar. Thus a hybrid arrangement is needed to isolate the transmitted signal from the received signal. The basic function of the time switch is to delay information in arriving time slots until the desired output time slot occurs. Let the communication is to take place between subscriber A and B. Let A is assigned time slot 2 and line 7 and subscriber B is assigned time slot 16 and line 11. Then the signal moved from time slot 2 to time slot 16 by the time-slot exchanger and is transferred from line 7to line 11 in the space array. Similarly, the signal originated by B is moved from slot 16 to slot2 through line 11 to 8.The cyclic control and gate select memory contains the information needed to specify the space stage configuration for each individual time slot of a frame. The time stage have to provide decays ranging from one time slot to a full frame. During each outgoing time slot, control information is accessed that specifies interstage link number to output link. During other time slots, the space switch is completely reconfigured to support other connections.

Let each time slot interchanger have T slots. If the space array is a N × N, then the simultaneous connections possible is NT. If T = 128 and N = 16, 2048 connections can be supported. This structure is not free of blocking. The control store is a parallel end around shift resister. If space array is at the inlet side and time switch is at the output side, the structure is referred as space time (ST) switching. Both TS and ST arrangements are equally effective.



Blocking probability :

The blocking probability of TS switching is calculated as follows.

The probability that a subscriber A is active = $\frac{\rho}{T}$

where ρ = fraction of time that a particular link is busy measured in Erlangs

T = number of time slots in a frame.

The probability that any other subscriber is active on the same link

$$= \frac{(T-1)\rho}{T}$$

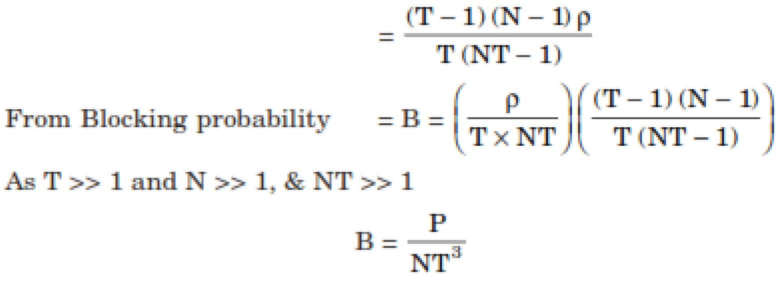
The probability that a particular called subscriber is chosen by A

$$= \frac{1}{NT} \times \frac{1}{T}$$

where N = Number of inlets (or outlets) for N × N space array.

NT = Simultaneous connection

The probability that the same time slot on a different outlet is chosen by the other subscribers on the same inlet



$$= \frac{(T-1)(N-1)\rho}{T(NT-1)}$$

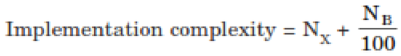
From Blocking probability $= B = \left(\frac{\rho}{T \times NT}\right)\left(\frac{(T-1)(N-1)}{T(NT-1)}\right)$

As T >> 1 and N >> 1, & NT >> 1

$$B = \frac{P}{NT^3}$$

The TS switch can be made non-blocking by using an expanding time switch (T to T slots) and a concentrating space switch (which is complex).

Implementation complexity. In general the complexity of the switching is represented in terms of number of cross points (N) and its associated cost. The number of cross points in space stage can be easily calculated which is based on the array size. The time stage uses significant amount of memory which adds the cost of the whole system. To take this into account the cost of memory bit is assumed one hundredth of the cost of cross point. Thus,



Implementation complexity = $N_X + \frac{N_B}{100}$

where N_X = Number of space stage cross points

N_B = Number of bits of memory.

The N_B not only includes the time stage memory arrays, but also the control memory (store) of the time stage and space stage. Thus,

$$N_B = N_{BX} + N_{BT}$$

where N_{BX} = Number of memory bits for the space stage control store

= N × (Number of control words) (number of bits per control word)

N_{BT} = Number of memory bits in the time stage equal to sum of time slot interchange and the control store bits.

= N × number of channels × number of bits per channel + N × number of control words × number of bits per control world.

For example problems please refer text books.

STS and TST Switching

The TS structure is of blocking nature. Let A and B are the subscribers using different timeslot on the same line want to connect to two subscribers C and D using same time slot on different lines. A and B can be moved to the same time slot but during that time slot, the inlet line can be connected to C's line or D's line but not both. This is the significant limitation of the structure. Moreover, time stage switching is generally less expensive than space stage switching as digital memory is much cheaper than digital cross points (AND gates).The multiple stages overcomes the limitations of the individual switches and cost savings can also be achieved. TST, STS, TSST, TSSSST and TSTSTSTSTSTSTS are the switching system configurations used in digital switching system. However, the TST structure is the most common.

STS Switching: In STS switching, the time stage is sandwiched between two space arrays. The digital switching system ITS 4/5 of USA (1976) uses the STS switching configuration. It handles 3000 trunks and accommodates 1500 Erlangs of traffic. Fig. shows the space-time-space (S-T-S) switching network for M incoming and outgoing PCM highways. Establishing a path through an STS switch requires finding a time switch array with unavailable units' access during the incoming time slot and an available read access during the desired outgoing time slot. The input side space stage as well as the output side space stage is free to utilise any free time switch modules. In the diagram shown in Fig. the time slot 2 is connected to the TSM 2 where the time slot allotted is 16 and passed to the (M – 1)th line of output space array. Thus the path is provided. This structure is of non-blocking nature.

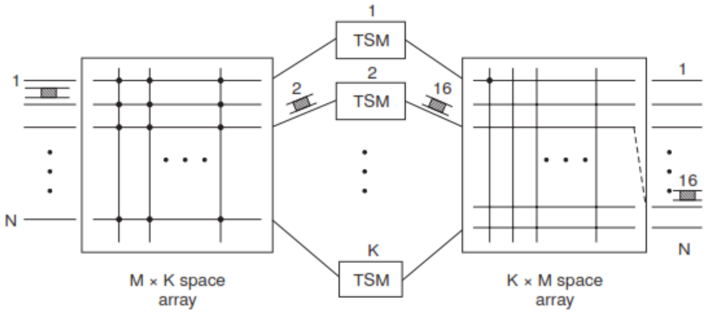
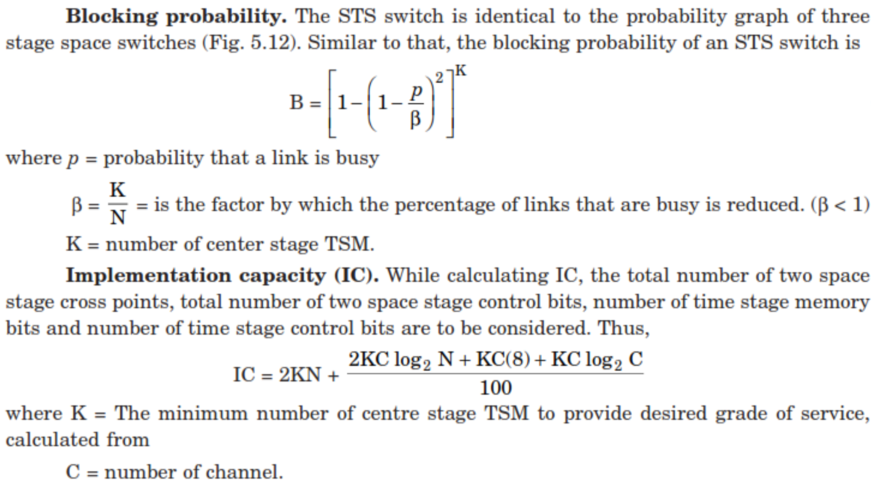


Fig. STS switching structure.



Blocking probability. The STS switch is identical to the probability graph of three stage space switches (Fig. 5.12). Similar to that, the blocking probability of an STS switch is

$$B = \left[1 - \left(1 - \frac{p}{\beta}\right)^2\right]^K$$

where p = probability that a link is busy

$\beta = \frac{K}{N}$ = is the factor by which the percentage of links that are busy is reduced. ($\beta < 1$)

K = number of center stage TSM.

Implementation capacity (IC). While calculating IC, the total number of two space stage cross points, total number of two space stage control bits, number of time stage memory bits and number of time stage control bits are to be considered. Thus,

$$IC = 2KN + \frac{2KC \log_2 N + KC(8) + KC \log_2 C}{100}$$

where K = The minimum number of centre stage TSM to provide desired grade of service, calculated from

C = number of channel.

TST Switching. In TST switching the space stage is sandwiched between two time stage switches. Of all the multistage switching, TST is a popular one. Some important features of TST switches are:

(*i*) **Low blocking probability.** An incoming channel time slot may be connected to an outgoing channel time slot using any possible space array time slot. Thus there are many alternative paths between two subscribers. This concept reduces the blocking probability of a three stage combination switch.

(*ii*) **Stage independency.** The space stage operates in a time-divided fashion, independently of the external TDM links. The number of space stage time slots L does not coincide with the number of external TDM time slots T.

(*iii*) **Implementation advantage.** The factors to be considered for switching design and implementation are traffic loads, modularity, testability, expandability and simple control requirements. For large switches with heavy traffic loads, the TST have good implementation advantage.

(*iv*) **More cost effective.** If the input channel loading is high, the time expansion of TST and space expansion of STS are required. Time expansion of TST can be achieved at less cost than space expansion of STS. In comparison with STS, the TST have certain limitations. For small switches, the STS architectures are less complex to implement than TST. The control requirements of STS is simpler than TST.

The principle of operation of TST switching is shown in Fig. In figure, two flows of time slots, one for each direction are connected together.

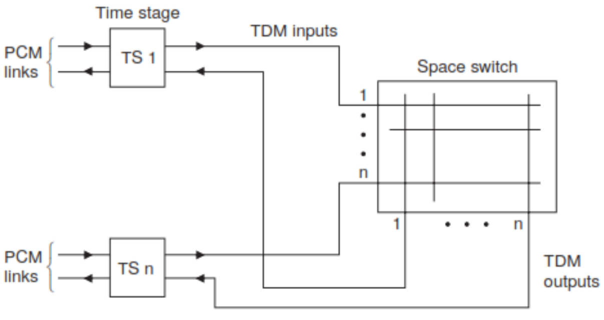


Fig. TST switching

The functional block diagram which explains the transfer of signals from inlet to outlet is shown in Fig. The information arriving at the incoming link of TDM channel is delayed in the· inlet times stage until an appropriate path through the space stage is available. Then the information is transferred through the space stage to the appropriate outlet time stage. Here the information is held until the desired outgoing time slot occurs. Any space stage time slot can be used to establish a connection. The space stage operates in a time divided fashion independently of the external TDM links. There are many alternative paths between a prescribed input and output unlike a two stage network which has only one fixed path.

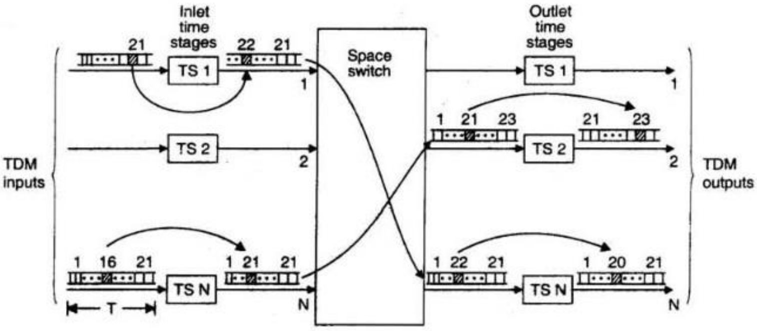
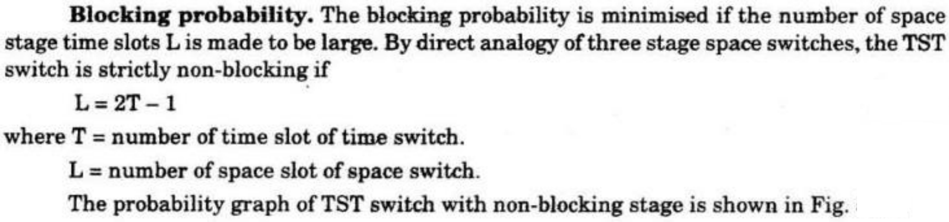


Fig. TST switching structure.


Blocking probability. The blocking probability is minimised if the number of space stage time slots L is made to be large. By direct analogy of three stage space switches, the TST switch is strictly non-blocking if

$$L = 2T - 1$$

where T = number of time slot of time switch.

L = number of space slot of space switch.

The probability graph of TST switch with non-blocking stage is shown in Fig.

Module-II: COMPUTER CONTROLLED SWITCHING SYSTEMS

Introduction, Call processing, signal exchange diagram, state transition diagram, hardware configuration, switching system software organization, software classification and interfacing, Maintenance software, call processing software, Administration software, Electronic Exchanges in India.

INTRODUCTION

Most digital switching systems have a quasi-distributed hardware architecture, since they maintain control of the switching functions through an intermediate processors. All digital switching systems employ multiprocessor subsystem for the best understanding of communication and control process. The architecture of a working digital switching system is very complex with many subsystems. All present day digital switching system includes minimum software which are necessary for implementation of call processing for all the levels of control structure. In modern digital switching systems, many call processing functions are performed by using interface controllers. Some of the call processing are call identification, call routing, path setup between subscribers, digital translation, call status, billing etc.

CALL PROCESSING

In this section, the basic steps involved in processing a call is discussed. Most digital system follow a similar scheme. For any switching system design, the range of signals that has to be interchanged between a terminal and system is considered. These signals described in signal exchange diagram. The sequence of operation between subscribers and system are shown in state transition diagram (s.t.d.).

Basic Steps to Process a Call

The sequence of processing between subscribers are described below:

1. **Idle state.** At this state, the subscriber handset is in ‘on-hook’ condition. The exchange is ready to detect the call request from the subscriber.

2. **Call request identification.** The exchange identifies a line requiring for a service. When the handset is lifted, current flows in the line called seize signal indicates the call request.

3. **Providing dial tone.** Once the seize signal is received, an exchange sends a dial tone to the calling subscriber to dial the numbers.

4. **Address analysis.** Once the first digit received, the exchange removes the dial tone and collect all numbers. Then the address is analysed for the validity of the number, local, STD or ISD etc. If the number is invalid, a recorded message may be sent to the calling subscriber and terminates call request.

5. **Called line identification.** The exchange determines the required outgoing line termination from the address that it has received.

6. **Status of called subscriber.** The called line may be busy or free or unavailable or even out of service. In the case of PBX, where the customer have a group of lines, the exchange tests each termination until either it finds a free one or all one found busy. For busy, number unobtainable or the handset off hook, a status signal or call progress signal is sent to the calling subscribers for line termination. Now the exchange resumes idle state.

7. **Ringing.** Once, the exchange finds the called subscriber is free, power ringing is provided to the called subscriber and audible ringing to the calling subscriber.

8. **Path setup.** When the called subscriber lifts his handset, the line is looped and ringing is removed. Once the conversation started, the exchange completes the connections between the subscribers.

9. **Supervision.** The exchange supervises the connection to detect the end of the call for charging.

10. **Clear signal.** Once the need for connection is over, either customer may replace his handset. It causes the line current seize and provides a clear signal to exchange. If the calling subscriber replaces his phone set, the clear signal sent to the exchange is called clear forward signal. If called subscriber do first, the clear signal is called clear backward signal.

Signal Exchange Diagram

There are two types of diagrams used to represent the sequence of events between the subscriber and exchanges. They are signal exchange diagram and state transition diagram. Both diagrams

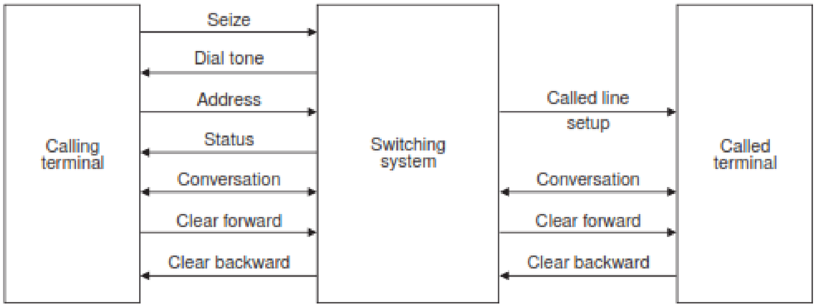


Fig. Signal exchange diagram

can be used to specify the behaviour of different control units in switching centre. For the local call, the steps involved in processing a call is shown in Fig. 6.1.Normally, once the conversation is over, the exchange will be at idle state. But in general, there are two types difficulties arises.

1. **Called subscriber held (CSH).** This condition arises when the called subscriber replaces the hand set but the caller does not. In this case, the caller does not originate a call or receiver a call.

2. **Permanent loop condition (PL).** This condition occurs when the caller replaces the phone but the called subscriber does not. Now, a loop present between called and exchange and it results in busy tone to another call to the same called subscriber. In strowger system, this condition is called permanent glow condition. In electromechanical system, the above conditions are removed by manual disconnection. In modern ESS systems, a time out process is used. If the call setup between two subscribers are made through many exchanges and trunks, the originating exchange where calling subscriber is connected sends the seize and then address to the terminating exchange where the called subscriber is connected. Remaining signalling are similar to the local call, but through the originating and terminating exchanges. In electromechanical system, the signalling between exchanges are sent through same interexchange circuits referred as channel associated signalling. In SPC controlled exchanges, interexchange signals are generated at originating exchange, but processed at terminating exchange. The signals are transferred over high speed data like instead of speech connections are referred as common channel signalling.

State Transition Diagram

The state transition diagram (s.t.d.) specifies the response of a control unit to any sequence of events. s.t.d. is a powerful design tool. It helps the designer to consider all possibilities of occurrence of events. Fig. shows the basic symbols used in a state transition diagram.

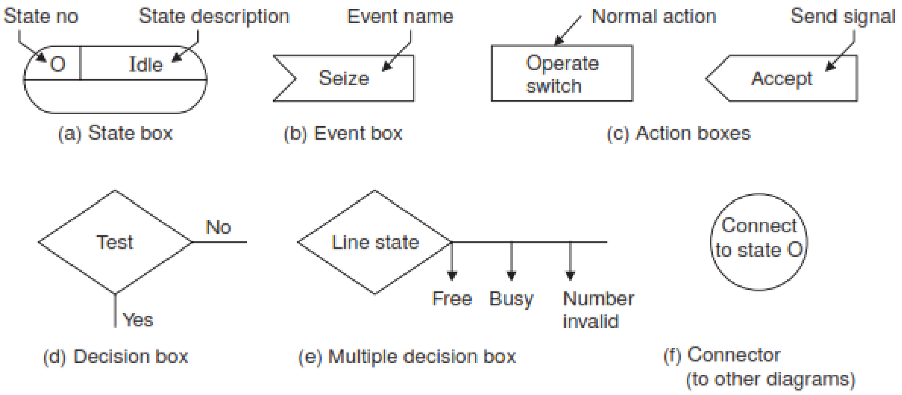


Fig. basic symbols of s.t.d.

The basic symbols are defined as follows:

State boxes. The state boxes are labelled with state number and state description. If necessary, additional information can also be included. The combination of the present state and a new event defines a task and performing this results in next state. Sometimes more than one state occurs, the choice depending on external information.

Event boxes. The intended arrow of the symbol indicate whether the event corresponds to the receipt of forward or backward signal. The forward signal and backward signal refers to the flow of signal from calling to called and called to calling subscriber through exchange respectively.

Action boxes. The rectangular box represents the action taken on the event. The protruding arrow indicates whether the signal is sent forward or backward.

Decision boxes. The diamond shaped box is used for the cases where two divisions are possible. For multiple decisions, another symbol shown in Fig. (*e*) is used.

Connectors. This symbols are used to connect one flow chart to another diagram.

Fig. shows the s.t.d. diagram for a typical local call. Let the calling subscriber is A and the called subscriber is B.

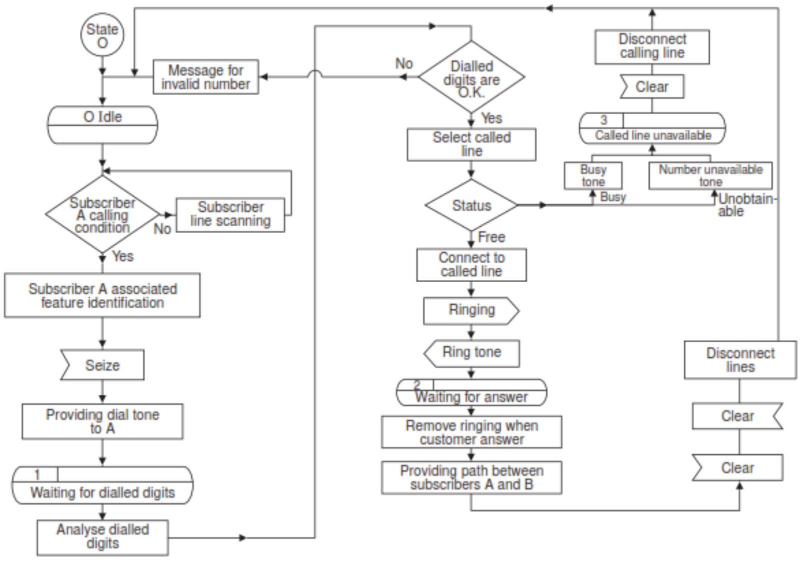


Fig. State transition diagram.

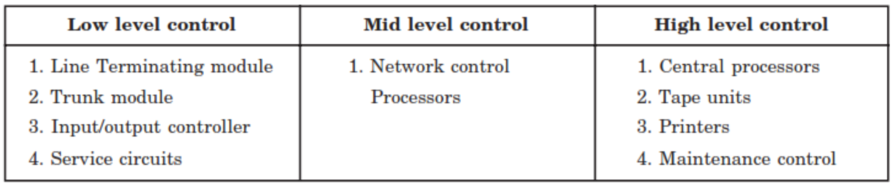
The computer controlled switching is in general referred as electronic switching system (ESS). ESS offers the greatest potential for both voice and data communications. An ESS consists of 1.computer

2. Memory or storage

3. Programming capability

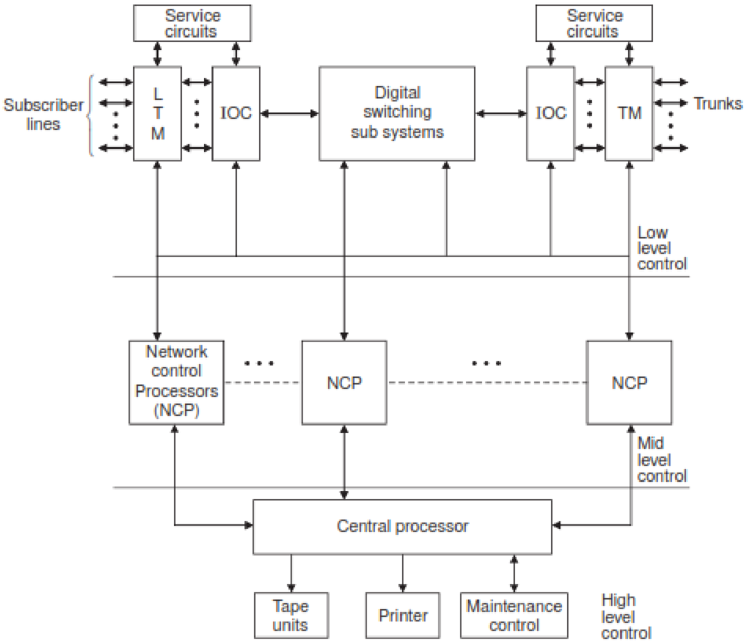
4. An extremely rapid switching component.

A computer based common control switching equipment implies two distinct type of units. They are 1. Control unit 2. Switching network. The common control receives, stores and interprets dial pulses and then selects an available path through the switching hardware to complete connection. Efficient high speed common control equipment can complete many calling connections during the time of an average phone call. Thus it saves a lot of time and money. The switching network can be used to connect many lines by one common group of control devices referred as control unit. Thus the control unit is the brain of a switching system, a control unit completes its function in a small fraction of a second for a single call. The hardware of digital switching system are broadly divided by their functions into many subsystems. The functions performed by the subsystem includes line and trunk access, line scanning, message interpretation, switching communications, path setup between subscribers, line supervision, line termination, billing providing advanced features and system maintenance. These subsystems are classified into various levels of control. Each level of control and its subsystems are tabulated in table.



Low level control	Mid level control	High level control
1. Line Terminating module 2. Trunk module 3. Input/output controller 4. Service circuits	1. Network control Processors	1. Central processors 2. Tape units 3. Printers 4. Maintenance control

A general hardware configuration is shown in Fig. However, various switching system may have different kind of arrangements of the subsystems. Most digital switching systems have a quasi-distributed hardware architecture, as the control of the switching functions are made through an intermediate processors. All digital switching systems employ multiprocessor subsystems as shown in Fig. A similar architecture is used by most of the digital telephone exchange systems. Some popular systems are AXE – 10 systems (Sweden), DMS – 10 (Canada),E – 10 system (France), No. 5 ESS system (USA) EWS D system (Germany) and the NEAX system (Japan). Fig. illustrates the hardware architecture of the digital switching system.



LTM : Line terminating modules, TM : Trunk modules,
IOC : Input/Output controller, NCP : Network control processors.

Low level control. This level associated with subscriber lines, trunks, selective circuits, Input/output controller and digital subsystems. The line terminating module and trunk modules are microprocessor based and communicate with subsystems through the input/output controllers. The input/output controllers interpret the incoming messages and takes necessary actions and communicate to the network control processors. All subscriber lines connected to digital switching system through the main distributing frame (MDF) are continuously scanned to detect the state of the subscriber.

When the customer lifts his handset, the line scanning program detects this state and reports to the input/output controller. The IOC is the primary peripheral controller and it controls all peripherals associated with call or trunk processing. At this level, all the requests of incoming and outgoing trunks are handled. Any advanced features to be incorporated in a digital switching system also handled at this level using IOC.

Mid-level control. This level is associated with network control processors and associated circuits. The IOC is controlled by the network control processors (NCP). Many NCP's are used depends on the size of the digital switching system. A dedicated bus system is usually required for the processors to communicate with one another. Specific messaging protocols are used to communicate between processors. For messaging between the peripherals and external systems, many digital switching systems utilize standard protocols such as signalling system 7(SS7); X.25 and X.75. Thus this is the most important level of control any digital switching system. Distributed processing are performed at this level.

High level control. This level associated with central processor which organizes the entire network control sub processors. In includes many subsystems like call accounting subsystems (CAS), call processing subsystems (CPS). Digital switching subsystems (DSS).Digital subscriber's switching subsystem (DSSS), Local administration (LA), maintenance control subsystems (MCS); management statistics subsystems (MSS), message transmission subsystems (MTS), signal interworking subsystems etc. This central processor is normally a main frame type computers. Thus all basic controls of a digital switching system are incorporated at this level. In real time operation, the processor determines the state of a call by reading data from memory. The store areas (not shown) include,

Line store. In this memory, the status of the line is stored. The status may be busy, free or disconnected.

Call record. All the call processing data's such as origin of a call, path of a call, and duration of a call and clearing of a call are stored.

Translation tables. Most switching system require a look-up table in order to decode routing digits into suitable routings. For electromechanical system, such tables are realized by distribution frame. Hundreds of translation tables are built for a switching system which stores data for equipment number (EN) to directory number (DN) and for DN–to–EN translation. Also it consists of, features related to a particular subscriber, data to route the call based on the first 3 digits dialled, area code translation, international call translators etc.

Map of the switching network. There are two techniques for selection junctors.

1. **Map-in-memory.** In this technique, the memory contain a bit for each link. If it is set to 1 the link is free and if this bit is set to zero, the line is busy.

2. **Map-in-network.** In this technique, the junctor itself contains a one bit memory element, which is read by the path setup program to check whether it is free. The map-in-network consumes more time, but more advantages when several processors controlling the system.

SWITCHING SYSTEM SOFTWARE ORGANIZATION

In last section, three levels of controls of hardware architectures were discussed for a general digital switching system. For effective processing of a call, to perform various functions of subsystems and to interface with the other subsystems, software plays a vital role. The software programs enables any digital switching system input data, to give outputs in a fraction of seconds, concurrent processing of many calls in real time and performs many features other than simple path set between subscribers for conversation. In this section, the need for software, the software classification, basic software architecture, the involvement of software in various levels of hardware architecture, interfacing between subsystems through software and software presently used in various digital switching system are described.

Need for Software

Other than call processing, any exchange is to serve the subscriber various facilities and many administrative tasks. Fig. shows various activities of a switching system. To carry out these activities efficiently and effectively, the use of software is unavoidable.

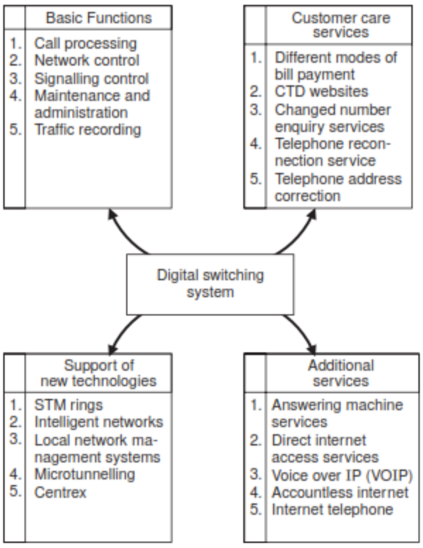


Fig. Activities of switching system.

To perform the above tasks, a large amount of software is required. However, the software for basic functions are must and remaining services are optional and requires software depends on the location of switching systems. Approximately 70% of the total software is used to perform basic functions. Only 0.1% of the total processing time is used by the 30% of the remaining service oriented software packages.

Software Classification and Interfacing

Classification. At various levels of hardware architecture, the software are used. Thus, many digital switching systems employ some system level software. Basic software systems are classified as:

1. Maintenance software

2. Call processing software

3. Database/Administration software

4. Feature software.

Above software packages are divided into program modules. Each module dealing with specific task. Several modules are grouped together to form functional units. Various factors are associated with the development of software product. These factors include the requirements of the business, the location of telephone exchanges, customer needs, internal requirements, and parameterised design. The parameterised design includes hardware parameter and software parameters. The hardware parameter are based on the hardware used in the central office or exchanges. They are number of network control processors, number of line controllers, number of subscribers to be serviced, number of trunks for which the exchange is engineered etc. Some examples of software parameters are the registers associated with number and size of automatic message accounting (AMA) registers, number and size of buffers for various telephony function and various features to be included for that particular exchanges. Thus, the parameterised design helps in designing software common to the similar types of exchanges.

Maintenance software

There are various activities and tests involved to maintain a switching system. Some of them are :

1. Supervision of the proper functioning of the exchange equipment, trunks and subscriber lines.

2. Monitoring the database of line and trunk assignments.

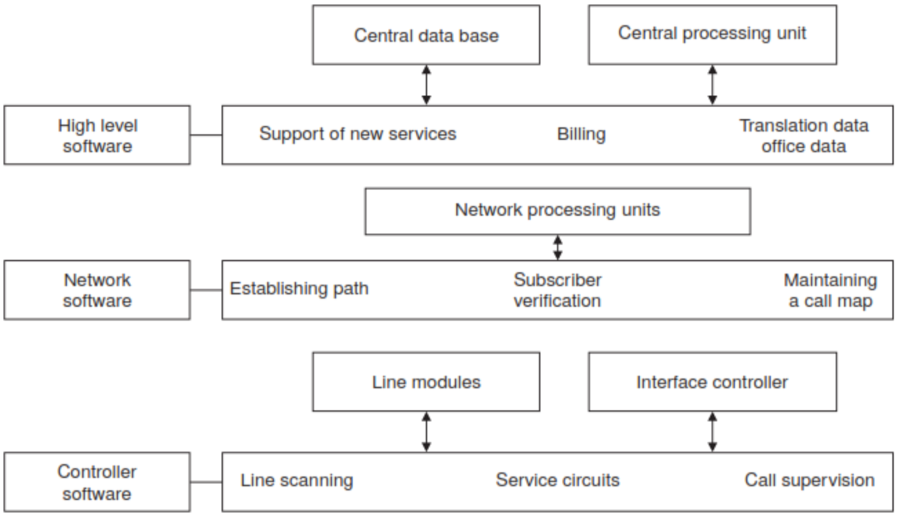
3. Efforts for the system recovery in case of failure.

4. Automatic line tests, which permits maintenance persons to attend several exchanges from one control location.

5. Effective diagnostic programs and maintenance strategies used to reduce the maintenance cost.

The root cause of the failure of any digital switching system is related to the software bugs which affects the memory and program loops, hardware failures, failure to identify the exact problem of failure and at least but not least the human error. Thus, the first step in software build is to select the appropriate program modules which is suitable for the switching system. The points to be considered are types of lines, location of switching system, signalling systems, availability of skilled person. Preventive maintenance programs are activated during the normal traffic. If a fault occurs, the OS activates the maintenance program to recover the system. Effective preventive and maintenance programs and strategies helps in proper maintenance of digital switching system with reduced maintenance cost.

Call processing software. The call processing functions are controlled by a central processor. Other functions carried out by the central processor are maintenance and administration, signalling, network control. Thus, the call processing programs are usually responsible for call processing and to interface with the translation data, office data, and automatic message accounting and maintenance programs. The translation data is the type of data generated by telephone companies related to subscriber. The office data is related to a particular digital switch. The call processing programs can be derived from state-transition diagrams in specification and description language (SDL). The SDL description in text form, is machine read and stored in memory in the form of data structures and linked lists and translation tables. An interpreter programs is written to access the lists and tables and to process the call by interpreting the data within them. Fig. shows three levels of call processing program. But it varies depends on the digital switching system.



Data base/Administration software

For administration and data base management, large amount of software required. But these tasks are performed infrequently, it uses less than 5% of the total processing time. The administration tasks includes

1. Alarm processing

2. Traffic recording

3. Change of numbers or area codes corresponding to the change in subscriber rate and Government policy.

4. Changing routing and routing codes. This decisions made on the traffic intensity of a particular exchange.

5. Generation of exchange management statistics.

Most digital switching system employ a data base system to:

1. Record office information

2. Billing information

3. Software and hardware parameters

4. System recovery parameters

5. System diagnostics.

Feature Software.

Most of the present day digital switching systems uses all packages.

Switching software. Software for digital switching systems are written in high level languages. Early electronic switching systems used assembly language programmes. In 1980, Plenary Assembly, CCITT approved the definition of a high level language as Recommendations–200. This language is known as CCITT high level language (CHILL). It has three major features as data structure, program structure and action statements. It is designed for the various SPC modules discussed earlier. Software codes for digital switching systems are also written in high level programming languages such as C, C ++, PASCAL.

Interfacing. The line control programs scan the status of lines and reports the status to the network status program. The network status programs works with network control programs. To provide dial tone, ringing, message to caller for invalid number, status of the subscriber and to receive dialled digits, and to clear signals from the subscriber, the line control programs interface with the network control programs. The call processing software which is responsible for call processing and in addition interfaces with accounting and maintenance programs for billing, recording and to identify the fault in lines. The call processing software also interfaces with feature programs to serve the customers need. The trunk modules interface different types of trunks to the digital switching system. Most digital switching systems employ special modules to connect ISDN and other digital services to the switch. Some specialized module interfaces are used to provide enhanced services such as **AIN** and packet switching.

ELECTRONIC EXCHANGES IN INDIA

Overview of Telecommunication Organizations

Department of Telecommunication (DOT) is the Government of India department under the ministry of communications. The main role of DOT in coordination with Telecom Regulatory Authority of India (TRAI) are Policy making, licencing and coordination relating to telegraphs, telephones, wireless, data, facsimile and telematics services. It also enforces wireless regulatory measures for wireless transmission by users in the country. The public sector companies under the ministry of communications which plays vital role in the telecommunications in India are

1. Bharat Sanchar Nigam Limited (BSNL)

2. Indian Telephone Industries Ltd (ITI)

3. Telecommunications consultants India (TCIL) Ltd

4. Mahanagar Telephone Nigam Limited (MTNL)

5. Videsh Sanchar Nigam Limited (VSNL)

6. Centre for development of telematics

The details of the BSNL which is the major telecom service provider and ITI, the leading telecom products manufacturer are given below in brief. For detailed information, the reader can refer the related websites. On October 1, 2000 the Department of Telecom operations, Govt. of India become a corporation and was christened Bharat Sanchar Nigam Limited (BSNL). Today, BSNL is the No. 1 telecommunication company and the largest public sector undertaking of India. It has a network of over 45 million lines covering more than 5000 towns and over 35 million telephone connections. The main functions of BSNL includes planning, engineering, installation, maintenance, management and operation of voice and non-voice telecommunication services all over the country. ITI established in 1948 is a Telecom company manufacturing the entire range of telecom equipment which includes telephones, large digital switches, and transmission systems like microwave, Fibre optic systems and satellite communication systems. Its highly satisfied customers in India include BSNL, MTNL, defence services, parliamentary, police and internal security organisations, railways etc. Many African and south Asian nations are its overseas customers. Related to digital switches, ITI in collaboration with ALCATEL, France manufactures large digital switches and with C-DOT India, manufactures small, medium and large digital switches. TCIL is a premier telecommunication consultancy and engineering company under the ministry of communications. TCIL with its number of joint venture Company’s manufactures computer hardware, copper and optical fibre cables, developing software packages and providing consultancy and engineering services to other computer, information technology, telecom and software companies.

Switching Systems in India

ITI has contributed to 73% of the installed base of Public switching lines and two thirds of the installed base of large switches in India. ITI provides similar service support for these products outside India also, which will be cost effective. The indigenously develop switching systems used in India are:

- CDOT 256P RAX • CDOT TAX-XL
- CDOT SBM. • CDOT AN-RAX
- CDOT SBM-XL • CDOT RLC
- CDOT MAX-L • CDOT CNMS
- CDOT MAX-XL

The digital switching systems in collaboration with other countries are:

- OCB-283 M/S ITI, M/S Alcatel
- 5 ESS M/S lucent
- EWSD M/S HTL, M/S Siemens
- AXE-10 M/S Ericsson
- FETEX-150 M/S Fujitsu
- NEAX-61E M/S NEC

Module-III: TRAFFIC ENGINEERING

Traffic pattern, Grade of Service and blocking probability, modeling of switching systems: Markov Process, Birth-Death Process.

Telephone network organization: Network management, Network services, various networking plans, types of networks, Routing plan, International numbering plan, National numbering plan, Numbering plan in India, Signaling: in channel signaling, common channel signaling.

The telecommunication system has to service the voice traffic and data traffic. The traffic is defined as the occupancy of the server. The basic purpose of the traffic engineering is to determine the conditions under which adequate service is provided to subscribers while making economical use of the resources providing the service. The functions performed by the telecommunication network depends on the applications it handles. Some major functions are switching, routing, flow control, security, failure monitoring, traffic monitoring, accountability internetworking and network management. To perform the above functions, a telephone network is composed of variety of common equipment such as digit receivers, call processors, inter stage switching links and interoffice links etc. Thus traffic engineering provides the basis for analysis and design of telecommunication networks or model. It provides means to determine the quantum of common equipment required to provide a particular level of service for a given traffic pattern and volume. The developed model is capable to provide best accessibility and greater utilization of their lines and trunks. Also the design is to provide cost effectiveness of various sizes and configuration of networks. The traffic engineering also determines the ability of a telecom network to carry a given traffic at a particular loss probability. Traffic theory and queuing theory are used to estimate the probability of the occurrence of call blocking. Earlier traffic analysis based purely on analytical approach that involved advanced mathematical concepts and complicated operations research techniques. Present day approaches combine the advent of powerful and affordable software tools that aim to implement traffic engineering concepts and automate network engineering tasks. In the study of tele traffic engineering, to model a system and to analyse the change in traffic after designing, the static characteristics of an exchange should be studied. The incoming traffic undergoes variations in many ways. Due to peak hours, business hours, seasons, weekends, festival, location of exchange, tourism area etc., and the traffic is unpredictable and random in nature. So, the traffic pattern/characteristics of an exchange should be analysed for the system design. The grade of service and the blocking probability are also important parameters for the traffic study.

The following statistical information provides answer for the requirement of trunk circuits for a given volume of offered traffic and grade of service to interconnect the end offices. The statistical descriptions of a traffic is important for the analysis and design of any switching network.

1. **Calling rate.** This is the average number of requests for connection that are made per unit time. If the instant in time that a call request arises is a random variable, the calling rate may be stated as the probability that a call request will occur in a certain short interval of time. If '*n*' is the average number of calls to and from a terminal during a period T seconds, the calling rate is defined as $\lambda = n/T$

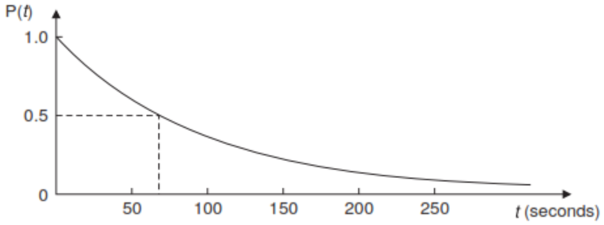
In telecommunication system, voice traffic and data traffic are the two types of traffic. The calling rate (λ) is also referred as average arrival rate. The average calling rate is measured in calls per hour.

2. **Holding time.** The average holding time or service time '*h*' is the average duration of occupancy of a traffic path by a call. For voice traffic, it is the average holding time per call in hours or 100 seconds and for data traffic, average transmission per message in seconds. The reciprocal of the average holding time referred to as service rate (μ) in calls per hour is given as $\mu = 1/h$

Sometimes, the statistical distribution of holding time is needed. The distribution leads to a convenient analytic equation. The most commonly used distribution is the negative exponential distribution. The probability of a call lasting at least *t* seconds is given by

$P(t) = \exp(-t/h)$

For a mean holding time of *h* = 100 seconds, the negative exponential distribution function is shown in Fig.



3. **Distribution of destinations.** Number of calls receiving at a exchange may be destined to its own exchange or remoted exchange or a foreign exchange. The destination distribution is

described as the probability of a call request being for particular destination. As the hierarchical structure of telecommunication network includes many intermediate exchanges, the knowledge of this parameter helps in determining the number of trunks needed between individual centres.

4. **User behaviour.** The statistical properties of the switching system are a function of the behaviour of users who encounter call blocking. The system behaves differently for different users. The user may abandon the request if his first attempt to make a call is failed. The user may makes repeated attempts to setup a call. Otherwise the user may wait some times to make next attempt to setup a call. These behaviour varies person to person and also depends on the situation.

5. **Average occupancy.** If the average number of calls to and from a terminal during a period T seconds is '*n*' and the average holding time is '*h*' seconds, the average occupancy of the terminal is given by

$A = nh/T = \lambda h = \lambda/\mu$

Thus, average occupancy is the ratio of average arrival rate to the average service rate. It is measured in Erlangs. Average occupancy is also referred as traffic flow or traffic intensity or carried traffic.

Traffic Pattern

An understanding of the nature of telephone traffic and its distribution with respect to time (traffic load) which is normally 24 hours is essential. It helps in determining the amount of lines required to serve the subscriber needs. According to the needs of telephone subscribers, the telephone traffic varies greatly. The variations are not uniform and varies season to season, month to month, day to day and hour to hour. But the degree of hourly variations is greater than that of any other period.

Various parameters related to traffic pattern are discussed below:

Busy hour. Traditionally, a telecommunication facility is engineered on the intensity of traffic during the busy hour in the busy session. The busy hour vary from exchange to exchange, month to month and day to day and even season to season. The busy hour can be defined in a variety of ways. In general, the busy hour is defined as the 60 minutes interval in a day, in which the traffic is the highest. Taking into account the fluctuations in traffic, CCITT in its recommendations E.600 defined the busy hour as follows.

1. **Busy hour.** Continuous 60 minutes interval for which the traffic volume or the number of call attempts is greatest.

2. **Peak busy hour.** It is the busy hour each day varies from day to day, over a number of days.

3. **Time consistent busy hour.** The 1 hour period starting at the same time each day for which the average traffic volume or the number of call attempts is greatest over the days under consideration. In order to simplify the traffic measurement, the busy hour always commences on the hour, half hour, or quarter hour and is the busiest of such hours. The busy hour can also be expressed as a percentage (usually between 10 and 15%) of the traffic occurring in a 24 hour period.

Call completion rate (CCR). Based on the status of the called subscriber or the design of switching system the call attempted may be successful or not. The call completion rate is defined as the ratio of the number of successful calls to the number of call attempts. A CCR value of 0.75 is considered excellent and 0.70 is usually expected.

Busy hour call attempts. It is an important parameter in deciding the processing capacity of an exchange. It is defined as the number of call attempts in a busy hour.

Busy hour calling rate. It is a useful parameter in designing a local office to handle the peak hour traffic. It is defined as the average number of calls originated by a subscriber during the busy hour.

Day-to-day hour traffic ratio. It is defined as the ratio of busy hour calling rate to the average calling rate for that day. It is normally 6 or 7 for rural areas and over 20 for city exchanges.

Units of Telephone Traffic

Traffic intensity is measured in two ways. They are (*a*) Erlangs and (*b*) Cent call seconds (CCS).

Erlangs. The international unit of traffic is the Erlangs. It is named after the Danish Mathematician, A.K. Erlang, who laid the foundation to traffic theory in the work he did for the Copenhagen telephone company starting 1908. A server is said to have 1 erlang of traffic if it is occupied for the entire period of observation. More simply, one erlang represents one circuit occupied for one hour.

The maximum capacity of a single server (or channel) is 1 erlang (server is always busy). Thus the maximum capacity in erlangs of a group of servers is merely equal to the number of servers.

For example problems on traffic engg. Please refer text books.

Cent call seconds (CCS). It is also referred as hundred call seconds. CCS as a measure of traffic intensity is valid only in telephone circuits. CCS represents a call time product. This is used as a measure of the amount of traffic expressed in units of 100 seconds. Sometimes call seconds (CS) and call minutes (CM) are also used as a measure of traffic intensity. The relation between erlang and CCS is given by

1E = 36 CCS = 3600 CS = 60 CM

Grade of Service (GOS)

For non-blocking service of an exchange, it is necessary to provide as many lines as there are subscribers. But it is not economical. So, some calls have to be rejected and retried when the lines are being used by other subscribers. The grade of service refers to the proportion of unsuccessful calls relative to the total number of calls. GOS is defined as the ratio of lost traffic to offered traffic.

$$\text{GOS} = \frac{\text{Blocked Busy Hour calls}}{\text{Offered Busy Hour calls}}$$

$$\text{GOS} = \frac{A - A_0}{A}$$

where A0 = carried traffic

A = offered traffic

A – A0 = lost traffic.

The smaller the value of grade of service, the better is the service. The recommended GOS is 0.002, *i.e.* 2 call per 1000 offered may lost. In a system, with equal no. of servers and subscribers, GOS is equal to zero. GOS is applied to a terminal to terminal connection. But usually a switching centre is broken into following components

(*a*) an internal call (subscriber to switching office)

(*b*) an outgoing call to the trunk network (switching office to trunk)

(*c*) the trunk network (trunk to trunk)

(*d*) a terminating call (switching office to subscriber).

The GOS calculated for each component is called component GOS. The overall GOS is in fact approximately the sum of the component grade of service.

There are two possibilities of call blocking. They are (*a*) Lost system and (*b*) Waiting system. In lost system, a suitable GOS is a percentage of calls which are lost because no equipment is available at the instant of call request. In waiting system, a GOS objective could be either the percentage of calls which are delayed or the percentage which are delayed more than a certain length of time.

Blocking Probability and Congestion

The value of the blocking probability is one aspect of the telephone company's grade of service. The basic difference between GOS and blocking probability is that GOS is a measure from subscriber point of view whereas the blocking probability is a measure from the network or switching point of view. Based on the number of rejected calls, GOS is calculated, whereas by observing the busy servers in the switching system, blocking probability will be calculated. The blocking probabilities can be evaluated by using various techniques. Lee graphs and Jacobaeus methods are popular and accurate methods. The blocking probability B is defined as the probability that all the servers in a system are busy. Congestion theory deals with the probability that the offered traffic load exceeds some value. Thus, during congestion, no new calls can be accepted. There are two ways of specifying congestion. They are time congestion and call congestion. Time congestion is the percentage of time that all servers in a group are busy. The call or demand congestion is the proportion of calls arising that do not find a free server. In general GOS is called call congestion or loss probability and the blocking probability is called time congestion. If the number of sources is equal to the number of servers, the time congestion is finite, but the call congestion is zero. When the number of sources is large, the probability of a new call arising is independent of the number already in progress and therefore the call congestion is equal to time congestion.

MODELLING OF TRAFFIC

To analyse the statistical characteristics of a switching system, traffic flow and service time, itis necessary to have a mathematical model of the traffic offered to telecommunication systems. The model is a mathematical expression of physical quantity to represents the behaviour of the quantity under consideration. Also the model provides an analytical solutions to a tele traffic problems. As the switching system may be represented in different ways, different models are possible. Depending on the particular system and particular circumstance, a suitable model can be selected. In practice, the facilities of the switching systems are shared by many users. This arrangement may introduce the possibility of call setup inability due to lack of available facilities. Also in data transfer, a system has to buffer message while waiting for transmission. Here size of the buffer depends on traffic flow. As serving the number of subscribers subject to fluctuation (due to random generation of subscriber calls, variations in holding time, location of the exchange, limitation in servers etc.), modelling of traffic is studied using the concepts and methods of the theory of probability. If a subscriber finds no available server for his call attempt, he will wait in a line (queue) or leave immediately. This phenomenon may be regarded as a queueing system. The mathematical description of the queueing system characteristics is called a queueing model.

The random process may be discrete or continuous. Similarly the time index of random variables can be discrete or continuous. Thus, there are four different types of process namely (*a*) continuous time continuous state (*b*) continuous time discrete state (*c*) discrete time continuous state and (*d*) discrete time discrete state. In telecommunication switching system, our interest is discrete random process and therefore for modelling a switching system, we use discrete state stochastic process. A discrete state stochastic process is often called a chain. A statistical properties of a random process may be obtained in two ways:

(*i*) Observing the behaviour of the system to be modelled over a period of time repeatedly. The data obtained is called a single sample. The average determined by measurements on a single sample function at successive times will yield a **time average**.

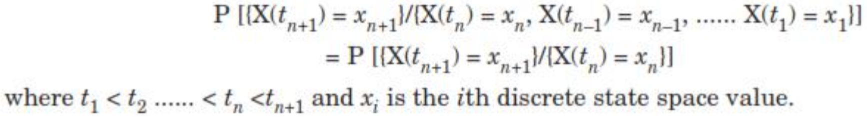
(*ii*) Simultaneous measurements of the output of a large number of statistically identical random sources. Such a collection of sources is called an **ensemble** and the individual noise waveforms is called the **sample function**. The statistical average made at some fixed time $t = t1$ on all the sample functions of the ensample is the **ensemble average.**

The above two ways are analogous to obtaining the statistics from tossing a die repeatedly (large number) or tossing one time the large number of dice. In general, time average and ensemble average are not the same due to various reasons. When the statistical characteristics of the sample functions do not change with time, the random process is described as being **stationary**. The random process which have identical time and ensemble average are known as **ergodic processes.** An ergodic process is stationary, but a stationary process is not necessarily ergodic.

Telephone traffic is nonstationary. But the traffic obtained during busy hour may be considered as stationary (which is important for modelling) as modelling non-stationary is difficult.

Pure Chance Traffic

Here, the call arrivals and call terminations are independent random events. If call arrivals are independent random events, their occurrence is not affected by previous calls. This traffic is therefore sometimes called **memory less traffic.** A.A. Markov in 1907, defined properties and proposed a simple and highly useful form of dependency. This class of processes is of great interest to our modelling of switching systems. A discrete time Markov chain *i.e.* discrete time discrete state Markov process is defined as one which has the following property.



$$P\,[\{X(t_{n+1}) = x_{n+1}\}/\{X(t_n) = x_n,\ X(t_{n-1}) = x_{n-1},\ \ldots\ldots\ X(t_1) = x_1\}]$$
$$= P\,[\{X(t_{n+1}) = x_{n+1}\}/\{X(t_n) = x_n\}]$$

where $t_1 < t_2 \ldots\ldots < t_n < t_{n+1}$ and x_i is the ith discrete state space value.

Equation states that the probability that the random variable X takes on the value x_{n+1}

at time step $n + 1$ is entirely determined by its state value in the previous time step n and is independent of its state values in earlier time steps ; $n – 1$, $n – 2$, $n – 3$ etc.

The Birth and Death Process

The birth and death process is a special case of the discrete state continuous time Markov process, which is often called a continuous-time Markov chain. The number of calls in progress is always between 0 and N. It thus has N + 1 states. If the Markov chain can occur only to adjacent states (*i.e.* probability change from each state to the one above and one below it) the process is known as birth-death (B–D) process. The basic feature of the method of Markov chains is the kolmogorov differential-difference equation, for the limiting case, can provide a solution to the state probability distribution for the Erlang systems and Engest systems.

Let N(t) be a random variable specifying the size of the population at time t. For a complete description of a birth and death process, we assume that N(t) is in state k at time t and has the following properties:

1. P(k) is the probability of state k and P(k + 1) is the probability of state k + 1.

2. The probability of transition from state k to state $k + 1$ in short duration Δt is $\lambda\ \Delta t$, where λ is called the birth rate in state k.

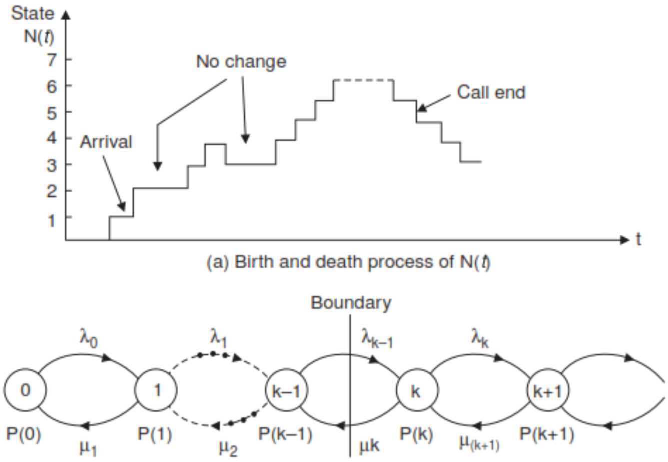
3. The probability of transition from state *k* to state *k* – 1 in the time interval Δ*t* is μ*k* Δ*t*, where μ is called the death rate in state *k*.

4. The probability of no change of state in the time interval Δ*t* is equal to 1 – (λ*k*) Δ*t*.

5. The probability in Δ*t*, from state *k* to a state other than *k* + 1 or *k* – 1 is zero. Based on the above properties, birth and death process of N(*t*) and state transition rate diagram are shown in Fig. At statistical equilibrium (*i.e.* stationary), let P_{jk} is the conditional probability, that is the probability of state increases from *j* to *k*. Similarly P_{kj} is the probability of state decrease from *k* to *j*.

The probabilities P(0), P(1), ... P(N) are called the **state probabilities** and the conditional probabilities P_{jk}, P_{kj} are called **transition probabilities.** The transition probabilities satisfy the following condition:

$$P_{jk}(t) \geq 0, \sum_{k=0}^{\infty} P_{jk}(t) = 1$$



(a) Birth and death process of N(*t*)

(b) State transition diagram

Numbered circles denote the state of the process at *t*

Markov theorem states that for any Markov process characterized by the transition probability P*jk*, the limit

$$\lim_{t \to \infty} P_{jk} = P(k)$$

exist and does not depend on *j* and the probability P(*k*).

According to Markov's,

$$\frac{d}{dt}\,\mathrm{P}(k) = -\,(\lambda_k + \mu_k)\,\mathrm{P}(k) + \lambda_{k-1}\,\mathrm{P}(k-1) + \mu_{k+1}\,\mathrm{P}(k+1)$$

$$k = 0, 1, 2, \ldots, \text{ with } \lambda_{-1} = \mu_0 = \mathrm{P}_{-1} = 0$$

This set of differential-difference equations represents the dynamic behaviour of the birth and death process. As *t* →∞,

$$-\,(\lambda_k + \mu_k)\,\mathrm{P}(k) + \lambda_{k-1}\,\mathrm{P}(k-1) + \mu_{k+1}\,\mathrm{P}(k+1) = 0$$

with $\lambda - 1 = \mu_0 = \mathrm{P}_{-1} = 0.$

This set of equations together with the normalization condition uniquely determines the required

$$\sum_{k=0}^{\infty} \mathrm{P}(k) = 1$$

and state probabilities P(*k*) as $\mathrm{P}(k) = \frac{\lambda_{k-1}}{\mu_k}\,\mathrm{P}(k-1)$

The probability P(0) can be determined by the equation

$$\mathrm{P}(k) = \frac{\lambda_0\,\lambda_1 \ldots\ldots \lambda_{k-1}}{\mu_1\,\mu_2 \ldots\ldots \mu_k}\,\mathrm{P}(0),\ k = 1, 2, 3, \ldots\ldots \qquad \mathrm{P}(0) = \left[1 + \sum_{k=1}^{\infty} \frac{\lambda_0\,\lambda_1 \ldots\ldots \lambda_{k-1}}{\mu_1\,\mu_2 \ldots\ldots \mu_k}\right]^{-1}$$

$$\mathrm{P}(k) = \frac{\lambda_{k-1}\,\mathrm{P}(k-1)}{\mu_k}$$

LOSS SYSTEMS

The service of incoming calls depends on the number of lines. If number of lines equal to the number of subscribers, there is no question of traffic analysis. But it is not only uneconomical but not possible also. So, if the incoming calls finds all available lines busy, the call is said to be **blocked.** The blocked calls can be handled in two ways. The type of system by which a blocked call is simply refused and is lost is called **loss system.** Most notably, traditional analog telephone systems simply block calls from entering the system, if no line available. Modern telephone networks can statistically multiplex calls or even packetize for lower blocking at the cost of delay. In the case of data networks, if dedicated buffer and lines are not available, they block calls from entering the system. In the second type of system, a blocked call remains in the system and waits for a free line. This type of system is known as **delay system.** These two types differs in network, way of obtaining solution for the problem and GOS.

For loss system, the GOS is probability of blocking. For delay system, GOS is the probability of waiting.

Erlang determined the GOS of loss systems having N trunks, with offered traffic A, with the following assumptions. (*a*) Pure chance traffic (*b*) Statistical equilibrium (*c*) Full availability and (*d*) Calls which encounter congestion are lost. The first two are explained in previous section. A system with a collection of lines is said to be a fully-accessible system, if all the lines are equally accessible to all in arriving calls. For example, the trunk lines for interoffice calls are fully accessible lines. The lost call assumption implies that any attempted call which encounters congestion is immediately cleared from the system. In such a case, the user may try again and it may cause more traffic during busy hour. The Erlang loss system may be defined by the following specifications.

1. The arrival process of calls is assumed to be Poisson with a rate of λ calls per hour.

2. The holding times are assumed to be mutually independent and identically distributed random variables following an exponential distribution with 1/μ seconds.

3. Calls are served in the order of arrival.

There are three models of loss systems. They are:

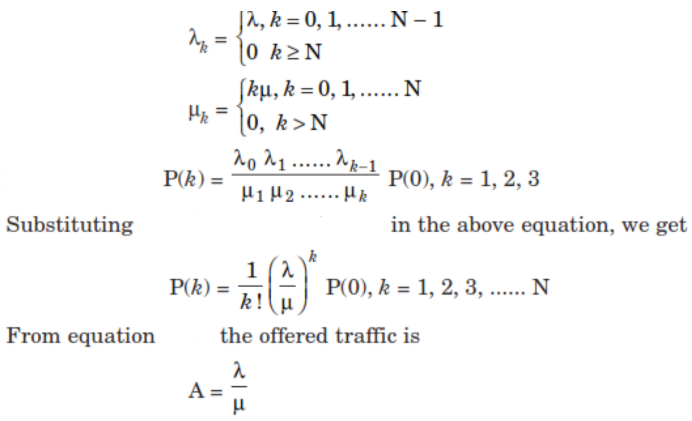
1. Lost calls cleared (LCC)

2. Lost calls returned (LCR)

3. Lost calls held (LCH)

Lost Calls Cleared (LCC) System

The LCC model assumes that, the subscriber who does not avail the service, hangs up the call, and tries later. The next attempt is assumed as a new call. Hence, the call is said to be cleared. This also referred as blocked calls lost assumption. The first person to account fully and accurately for the effect of cleared calls in the calculation of blocking probabilities was A.K.Erlang in 1917. Consider the Erlang loss system with N fully accessible lines and exponential holding times. The Erlang loss system can be modelled by birth and death process with birth and death rate as follows.



$$\lambda_k = \begin{cases} \lambda, k = 0, 1, \ldots\ldots N-1 \\ 0 \quad k \geq N \end{cases}$$

$$\mu_k = \begin{cases} k\mu, k = 0, 1, \ldots\ldots N \\ 0, \quad k > N \end{cases}$$

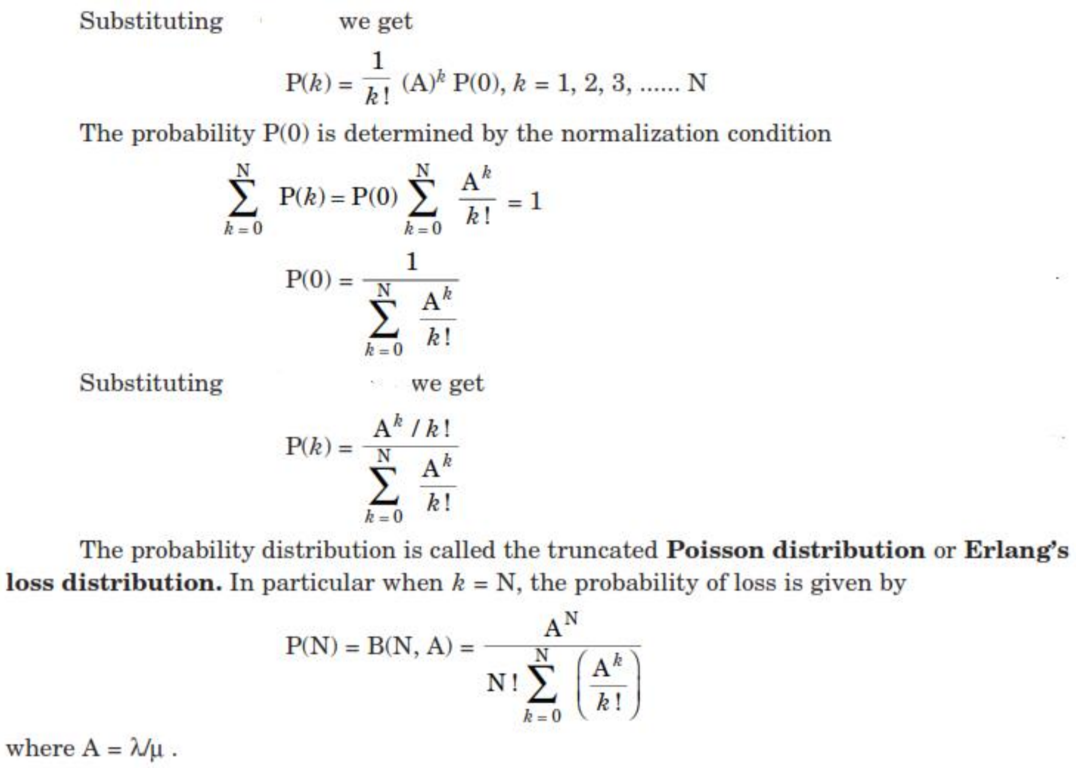
$$P(k) = \frac{\lambda_0\, \lambda_1 \ldots\ldots \lambda_{k-1}}{\mu_1\, \mu_2 \ldots\ldots \mu_k}\, P(0),\ k = 1, 2, 3$$

Substituting in the above equation, we get

$$P(k) = \frac{1}{k\,!}\left(\frac{\lambda}{\mu}\right)^k P(0),\ k = 1, 2, 3, \ldots\ldots N$$

From equation the offered traffic is

$$A = \frac{\lambda}{\mu}$$



Substituting we get

$$P(k) = \frac{1}{k\,!}\,(A)^k\, P(0),\ k = 1, 2, 3, \ldots\ldots N$$

The probability P(0) is determined by the normalization condition

$$\sum_{k=0}^{N} P(k) = P(0) \sum_{k=0}^{N} \frac{A^k}{k\,!} = 1$$

$$P(0) = \frac{1}{\displaystyle\sum_{k=0}^{N} \frac{A^k}{k\,!}}$$

Substituting we get

$$P(k) = \frac{A^k / k\,!}{\displaystyle\sum_{k=0}^{N} \frac{A^k}{k\,!}}$$

The probability distribution is called the truncated **Poisson distribution** or **Erlang's loss distribution.** In particular when k = N, the probability of loss is given by

$$P(N) = B(N, A) = \frac{A^N}{N\,! \displaystyle\sum_{k=0}^{N} \left(\frac{A^k}{k\,!}\right)}$$

where A = λ/μ .

This result is variously referred to as **Erlang's formula of the first kind, the Erlang's-B formula or Erlang's loss formula.**

Equation specifies the probability of blocking for a system with random arrivals from an infinite source and arbitrary holding time distributions. The Erlang B formula gives the time congestion of the system and relates the probability of blocking to the offered traffic and the number of trunk lines. Values from B(N, A) obtained from equation have been plotted against the offered traffic 'A' Erlang's for different values of the number of N lines in Fig. In design problems, it is necessary to find the number of trunk lines needed for a given offered traffic and a specified grade of service.

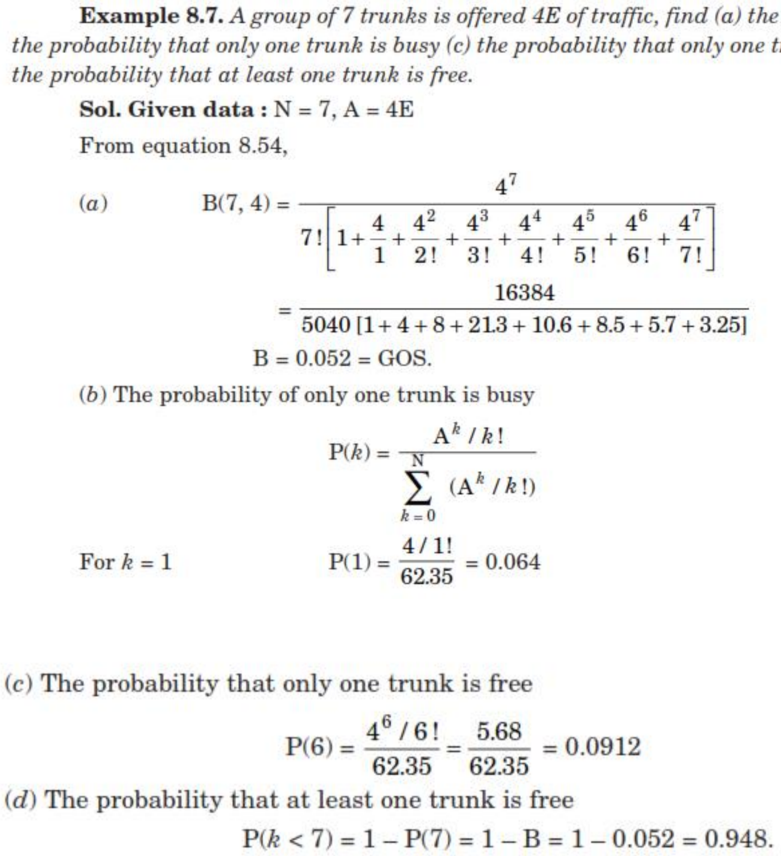
For example problems please refer text books.

A' = A [1 – B (N, A)]

Thus, the carried load is the position of the offered load that is not lost from the system. The carried load per line is known as the trunk occupancy.

$$\rho = \frac{A'}{N} = \frac{A(1-B)}{N}$$

The trunk occupancy ρ is a measure of the degree of utilization of a group of lines and is sometimes called the utilization factor.



Example 8.7. *A group of 7 trunks is offered 4E of traffic, find (a) the grade of service (b) the probability that only one trunk is busy (c) the probability that only one trunk is free and (d) the probability that at least one trunk is free.*

Sol. Given data : N = 7, A = 4E

From equation 8.54,

(*a*)

$$B(7, 4) = \frac{4^7}{7!\left[1+\frac{4}{1}+\frac{4^2}{2!}+\frac{4^3}{3!}+\frac{4^4}{4!}+\frac{4^5}{5!}+\frac{4^6}{6!}+\frac{4^7}{7!}\right]}$$

$$= \frac{16384}{5040\,[1+4+8+21.3+10.6+8.5+5.7+3.25]}$$

$$B = 0.052 = \text{GOS}.$$

(*b*) The probability of only one trunk is busy

$$P(k) = \frac{A^k/k!}{\sum_{k=0}^{N}(A^k/k!)}$$

For $k = 1$ $$P(1) = \frac{4/1!}{62.35} = 0.064$$

(*c*) The probability that only one trunk is free

$$P(6) = \frac{4^6/6!}{62.35} = \frac{5.68}{62.35} = 0.0912$$

(*d*) The probability that at least one trunk is free

$$P(k < 7) = 1 - P(7) = 1 - B = 1 - 0.052 = 0.948.$$

Lost Calls Returned (LCR) System

In LCC system, it is assumed that unserviceable requests leave the system and never return. This assumption is appropriate where traffic overflow occurs and the other routes are in other calls service. If the repeated calls not exist, LCC system is used. But in many cases, blocked calls return to the system in the form of retries. Some examples are subscriber concentrator systems, corporate tie lines and PBX trunks, calls to busy telephone numbers and access to WATS lines. Including the retried calls, the offered traffic now comprise two components *viz.*, new traffic and retry traffic. The model used for this analysis is known as lost calls returned (LCR) model. The following assumptions are made to analyse the CLR model.

1. All blocked calls return to the system and eventually get serviced, even if multiple retries are required.

2. Time between call blocking and regeneration is random statistically independent of each other. This assumption avoid complications arising when retries are correlated to each other and tend to cause recurring traffic peaks at a particular waiting time interval.

3. Time between call blocking and retry is somewhat longer than average holding time of a connection. If retries are immediate, congestion may occur or the network operation becomes delay system.

Consider a system with first attempt call arrival ratio of λ (say 100). If a percentage B (say 8%) of the calls blocked, B time's λ retries (*i.e.* 8 calls retries). Of these retries, however a percentage B will be blocked again.

Hence by infinite series, total arrival rate λ' is given as

$$\lambda' = \lambda + B\lambda + B^2\lambda + B^3\lambda + \ldots\ldots$$

$$\lambda' = \frac{\lambda}{1 - B}$$

where B is the blocking probability from a lost calls cleared (LCC) analysis.

The effect of returning traffic is insignificant when operating at low blocking probabilities. At high blocking probabilities, it is necessary to incorporate the effects of the returning traffic into analysis.

Lost Calls Held (LCH) System

In a lost calls held system, blocked calls are held by the system and serviced when the necessary facilities become available. The total time spend by a call is the sum of waiting time and the service time. Each arrival requires service for a continuous period of time and terminates its request independently of its being serviced or not. If number of calls blocked, a portion of it is lost until a server becomes free to service a call. An example of LCH system is the time assigned speech interpolation (TASI) system.

LCH systems generally arise in real time applications in which the sources are continuously in need of service, whether or not the facilities are available. Normally, telephone network does not operate in a lost call held manner. The LCH analysis produces a conservative design that helps account for retries and day to day variations in the busy horn calling intensities. A TASI system concentrates some number of voice sources onto a smaller number of transmission channels. A source receives service only when it is active. If a source becomes active when all channels are busy, it is blocked and speech clipping occurs. Each speech segment starts and stops independently of whether it is served or not. Digital circuit multiplication (DCM) systems in contrast with original TASI, can delay speech for a small amount of time, when necessary to minimize the clipping. LCH are easily analysed to determine the probability of the total number of calls in the system at any one time. The number of active calls in the system at any time is identical to the number of active sources in a system capable of carrying all traffic as it arises. Thus the distribution of the number in the system is the Poisson distribution. The Poisson distribution is given as

$$P(x) = \frac{\mu^x}{x!} e^{-\mu}.$$

The probability that k sources requesting service are being blocked is simply the probability that $k + N$ sources are active when N is the number of servers.

DELAY SYSTEMS

The delay system places the call or message arrivals in a queue if it finds all N servers (or lines) occupied. This system delays non-serviceable requests until the necessary facilities become available. These systems are variously referred to as delay system, waiting-call systems and queueing systems. The delay systems are analysed using queueing theory which is sometimes known as waiting line theory. This delay system have wide applications outside the telecommunications. Some of the more common applications are data processing, supermarket checkout counters, aircraft landings, inventory control and various forms of services. Consider that there are k calls (in service and waiting) in the system and N lines to serve the calls. If $k = N$, k lines are occupied and no calls are waiting. If $k > N$, all N lines are occupied and $k – N$ calls waiting. Hence a delay operation allows for greater utilization of servers than does a loss system. Even though arrivals to the system are random, the servers see a somewhat regular arrival pattern. A queueing model for the Erlang delay system is shown in Fig.

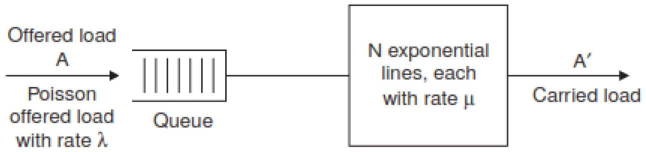


Fig. Queueing model

The basic purpose of the investigation of delay system is to determine the probability distribution of waiting times. From this, the average waiting time W as random variable can be easily determined. The waiting times are dependent on the following factors:

1. Number of sources

2. Number of servers

3. Intensity and probabilistic nature of the offered traffic

4. Distribution of service times

5. Service discipline of the queue.

In a delay system, there may be a finite number of sources in a physical sense but an infinite number of sources in an operational sense because each source may have an arbitrary number of requests outstanding. If the offered traffic intensity is less than the servers, no statistical limit exists on the arrival of calls in a short period of time. In practice, only finite queue can be realised. There are two service time distributions. They are constant service times and exponential service times. With constant service times, the service time is deterministic and with exponential, it is random. The service discipline of the que involves two important factors.

1. Waiting calls are selected on of first-come, first served (FCFS) or first-in-first-out (FIFO) service.

2. The second aspect of the service discipline is the length of the queue. Under heavy loads, blocking occurs. The blocking probability or delay probability in the system is based on the queue size in comparison with number of effective sources. We can model the Erlang delay system by the birth and death process with the following birth and death rates respectively.

$\lambda_k = \lambda$, $k = 0, 1, \ldots\ldots$ and

$$\mu_k = \begin{cases} k\mu \text{ , } k = 0, 1, \ldots\ldots \text{N} - 1 \\ \text{N}\mu \text{ } k \geq \text{S} \end{cases}$$

Under equilibrium conditions, the state probability distribution P(k) can be obtained by substituting these birth rates into the following equation

$$P(k) = \frac{\lambda_0\, \lambda_1 \ldots\ldots \lambda_{k-1}}{\mu_1\, \mu_2 \ldots\ldots \mu_k}\, P(0)\; k = 1, 2, \ldots\ldots$$

we set

$$P(k) = \begin{cases} \dfrac{1}{k\,!}\left(\dfrac{\lambda}{\mu}\right)^k P(0) & 0 \le k \le S \\ \dfrac{\left(\dfrac{\lambda}{\mu}\right)^k}{N\,!\,N^{k-N}}\, P(0) & k \ge N. \end{cases}$$

As $A = \dfrac{\lambda}{\mu}$, we get

$$p(k) = \begin{cases} \dfrac{A^k}{K\,!} P(0) & 0 \le k \le S \\ \dfrac{A^k}{N\,!\,N^{k-N}} P(0); & k > N \end{cases}$$

Under normalised condition,

$$\sum_{k=0}^{\infty} P(k) = 1 \qquad \text{or} \qquad \sum_{k=0}^{N-1} \frac{A^k}{k\,!} P(0) + \sum_{k=N}^{\infty} \frac{A^k}{N\,!\,N^{k-N}} P(0) = 1$$

$$\frac{1}{P(0)} = \sum_{k=0}^{N-1} \frac{A^k}{k\,!} + \frac{N^N}{N\,!} \sum_{k=N}^{\infty} \left(\frac{A}{N}\right)^k$$

$$= \sum_{k=0}^{N-1} \frac{A^k}{k\,!} + \frac{N^N}{N\,!}\left[\left(\frac{A}{N}\right)^N + \left(\frac{A}{N}\right)^{N+1} + \left(\frac{A}{N}\right)^{N+2} + \ldots\ldots\right]$$

$$= \sum_{k=0}^{N-1} \frac{A^k}{k\,!} + \frac{N^N}{N\,!}\left(\frac{A}{N}\right)^N \left[1 + \frac{A}{N} + \left(\frac{A}{N}\right)^2 + \ldots\ldots\right]$$

$$= \sum_{k=0}^{N-1} \frac{A^k}{k\,!} + \frac{A^N}{N\,!}\left[\frac{1}{1 - A/N}\right] = \sum_{k=0}^{N-1} \frac{A^k}{k\,!} + \left[\frac{A^N}{N\,!} + \frac{A^N}{N\,!}\left(\frac{A}{N-A}\right)\right]$$

$$\frac{1}{P(0)} = \sum_{k=0}^{N} \frac{A^k}{k\,!} + \frac{A^N}{N\,!}\left(\frac{A}{N-A}\right)$$

$$\frac{1}{P(0)} = \sum_{k=0}^{N} \frac{A^k}{k\,!} + \frac{A^N}{N\,!}\left(\frac{A}{N-A}\right)$$

$$P(0) = \frac{1}{\displaystyle\sum_{k=0}^{N} \frac{A^k}{k\,!} + \frac{A^N}{N\,!}\left(\frac{A}{N-A}\right)}$$

We know

$$P(k) = \frac{A^k}{k\,!}\, P(0),\; k = 1, 2, \ldots\ldots, N$$

$$\mathrm{C(N, A)} = \frac{\mathrm{A^N / N!}}{\sum_{k=0}^{\mathrm{N}} \frac{\mathrm{A}^k}{k!} + \frac{\mathrm{A^N}}{\mathrm{N!}}\left(\frac{\mathrm{A}}{\mathrm{N-A}}\right)}$$

$$\frac{1}{\mathrm{C(N, A)}} = \frac{\sum_{k=0}^{\mathrm{N}} \frac{\mathrm{A}^k}{k!}}{\frac{\mathrm{A^N}}{\mathrm{N!}}} + \frac{\frac{\mathrm{A^N}}{\mathrm{N!}}\left(\frac{\mathrm{A}}{\mathrm{N-A}}\right)}{\frac{\mathrm{A^N}}{\mathrm{N!}}!}$$

$$\frac{1}{\mathrm{C(N, A)}} = \frac{1}{\mathrm{B}} + \frac{\mathrm{A}}{\mathrm{N-A}} .$$

$$\text{Prob. (delay)} = \mathrm{P(> 0)\ C(N, A)} = \frac{\mathrm{BN}}{\mathrm{N - A\,(1 - B)}}$$

where B = Blocking probability for a LCC system

N = Number of servers

A = Offered load (Erlangs)

Equation above are referred as **Erlang’s second formula, Erlang’s delay formula or Erlang’s C formula.**

For single server systems (N = 1), the probability of delay reduces to ρ, which is simply the output utilization or traffic carried by the server. Thus the probability of delay for a single server system is also.

TELEPHONE NETWORK ORGANIZATION

A telecommunication network contains a large number of links joining different locations, which are known as the nodes of the network. These nodes may be end instruments (subscriber nodes), switching centres (switching nodes), networks providing just a link between nodes (transmission nodes) or service nodes (which provides service on demand such a voice mail boxes, stock market price announcement, sports results etc.). To provide efficient communication, a telephone network should include various transmission system (for example, terestial, microwave, optical satellite communications), switching system (to identity and connect calling and called subscriber) and to exchange information between subscriber and switching systems or between interexchange, a good signalling system required. The calling and called subscriber should be connected almost instantly. So, as an identification, a numbering system is introduced and it varies region to region and country to country. Telephone networks require certain form of procedure to route a particular call to the destination for effective and cost effective communication. So, the telephone network should be implemented with a good routing plan. An establishment of an exchange includes heavy expenses on switching equipment, establishing trunks and links, buildings, infrastructure, human resources to handle the exchange etc. These capital lost and the day-to-day expenses must be met by the exchanges through its subscribers. So, the billing and charging the subscriber calls or data transfer is a vital part of the network. Also, introducing a new exchange, extension of the existing exchanges, upgradation of the facilities and speed up the switching, changing the sales strategy based on the competition, addition of new services, management of maintenance, providing employment to the skilled peoples etc., are based on the government policies or the telephone company's business strategies. Thus, the functions of telecommunication networks is limit less and network management is an important part of any telecommunication network organization.

NETWORK MANAGEMENT

The basic goal of the network management is to maintain efficient operations during equipment failures and traffic overloads. Also controlling the flow of call requests during network overload is a vital function of network management. For the effective network management the study of various services provided by the network, offered load of the network, classification of the network based on services offered, interconnection of different types of networks and network planning is important. Based on the data available for the above factors, the network management has become updated.

Network Planning

The planning of telecommunication system comprising a network of switching centres includes various plans. From initiating the network to the extension of the network based on the increased load, network planning plays a vital role.

Network services: The capabilities often collectively referred to "as intelligence" within the network are listed below. Depending upon the applications the network is to handle the interconnectivity with other networks. The following functions and its essential parts are included in the network. Various services are:

1. **Switching.** The process of interconnecting incoming calls or data to the appropriate outgoing channel called destination is referred switching.

2. **Routing.** The ability of the network to select a path to connect calling and called subscriber for telephone conversations or providing path for data transfer between source and destination is referred as routing. The network generally choses a path and sometimes user may specify it.

3. **Flow control.** It is the ability of a network to reject traffic. Managing the rate at which traffic enters a network is referred to as flow control. A network without effective flow control procedures becomes very inefficient.

4. **Security.** There are two ways of providing security of the network. First, to increase the security of operation in presence of faults. To provide adequate security, the complete network may be duplicated or triplicated. Second, preventing unauthorized access to the network and the data it carries. This may be achieved by pass words, data encryption and providing limiting factors in accessing the network.

5. **Signalling.** A signalling system link the variety of switching system, transmission system and subscriber equipment in a telecommunication network to enable the network to function as a whole.

6. **Traffic management.** The ability of the network to keep track of traffic levels is referred as traffic management. Traffic management is useful both in short term and long term bases. On a short term basis, it can be used to support dynamic routing and flow control. Over a long term it can be used in network design to identify parts of the network where capacity may be productively increased or decreased.

7. **Accountability.** This includes charging, billing, accounting and inventory control. This is the ability of the network to track the users of the network.

8. **Administration.** It is related to the ability of the network to identify the load of a network and providing corresponding upgradation of parts, extension of networks facility. It also identities the sales strategy, investment planning etc.

9. **Inter-networking.** It is the ability of the network to perform the functions needed to communicate with and across other networks. This includes providing routes for traffic crossing through, into and out of the network, and allocating resources such as buffers and link capacity to traffic originating other networks.

Network levels: All the above functions and some other service related functions are usually classified or grouped into different levels. This grouping ease the network and the concerned network engineers to carry out the functions efficiently. The grouping differs from network to network or divisions to divisions or between telephone companies. The ability of the network to provide these functions has a profound effect on the type of network. In particular, the nodes of the network become more complex and more expensive as their functionality increases. Fig. shows the different levels of the network management.

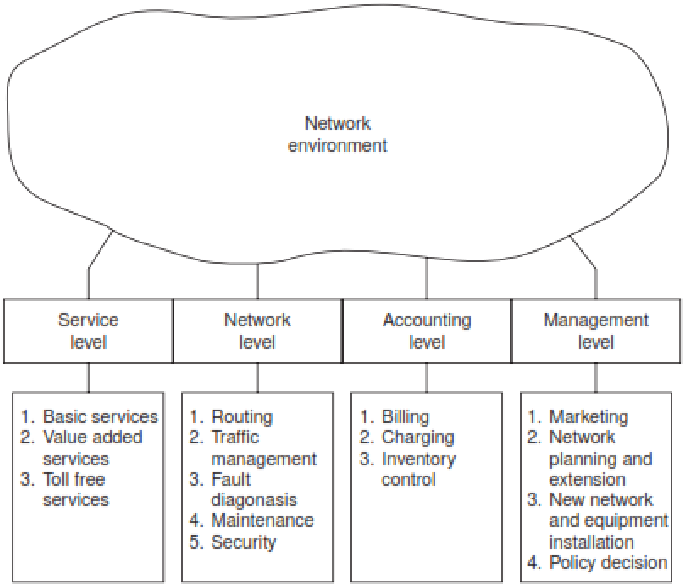


Fig. Different levels of network management.

Various networking plans: A national telecommunication network is large and complex. Therefore certain plans are needed to govern the design of network. The plans are independent and are affected by the predicted (or planned) growth rate of the telecommunication system. More specific network planning are :

1. Routing plans

2. Numbering plans

3. Charging plan

4. Transmission plan

5. Signalling plan

6. Grade of service

7. Network control and network administration.

The choice of a plan for a telecommunication system generally involves comparison of the economics of various possible plans. It also involves comparison of the economy of various possible plans and involves a certain amount of human judgement.

Types of Networks

There are in general three networks which can be used for any services (voice or data transfer).

They are:

(*a*) **Public switched network.** It allows access to the end office, connects through the long-distance network, and delivers to the end point. There are many hierarchies depends on the wish of network provider. But, the goal of the network hierarchies is to complete the call in the least amount of time and the shortest route possible.

(*b*) **Private networks.** Many companies, depending on their size and need, create or build their own networks. If their networks are underutilised, they may give their network for hire or lease. These networks employ mixture of technologies.

(*c*) **Hybrid networks.** To provide a service, if an organisation uses both private and public networks, the network is referred as hybrid network. Normally, the high-end usage services are connected via private facilities, the lower volume locations use the switched network.

This usually works out better financially for the organisation because the costs can be fully justified on a location by location basis. As the private line networks are designed to suit integration of voice, data, video graphics and facsimile transmission, pressure is more on it.

The customers are generally in need of variety of services. Even though certain networks are used to perform many services, they are more effective for a particular one or two services.

Hence, based on the services, the networks are classified as

1. The Public Switched telephone Network (PSTN)—for telephony.

2. The public switched telegraph network—for telex

3. Data networks—for voice and data

4. Cellular radio network—for mobile communication

5. Special service networks—to meet specialised demands.

As the above networks may be used to perform mixture of services, many authors categorised the networks into only three classes. They are generally considered as major telecommunication networks.

1. PSTN or POTS

2. Data networks

3. ISDN.

ROUTING PLAN

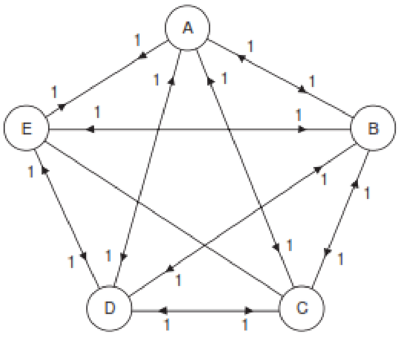
Routing planning refers to the procedures that determine which path in a network are assigned to particular connections. The switching centres may use fixed routes to each destination. Adaptive routing may be employed in which each exchange may use different routes for the same destination, depending upon traffic conditions. For effective routing of a call, some form of interconnection of switching exchanges are required. In the following paragraph various forms of networking and its related routing are discussed.

9.3.1. Basic Topologies

Three basic topologies are adopted for interconnecting exchanges. Exchanges are interconnected by group of trunk lines referred as trunk groups. Two trunk groups are required between any two exchanges. Mesh, star and mixed or hierarchical are the three basic topologies.

Determination at the total number of trunk circuits in any network is necessarily a function of the amount of traffic between each pair of stations or exchanges.

Mesh-connected network. This is also called fully connected topology. The advantage of mesh network is that each station has a dedicated connection to other stations. Therefore, this topology offers the highest reliability and security. If one link in the mesh topology breaks, the network remains active. A major or disadvantage of this topology is that it uses too many connections and therefore requires great deal of wiring, especially when the number of station increases. The mesh topology requires N (N – 1)/2 connections. For 100 stations, 4950 links required. Fig. depicts a full connected mesh structure.

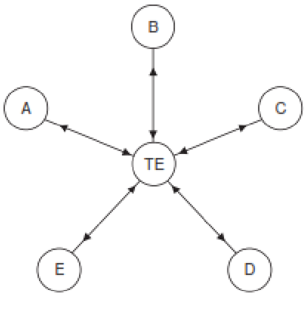


Mesh connected network.

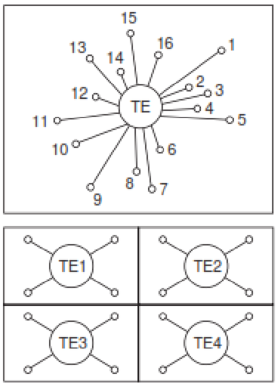
For high traffic networks, the cost of network increases. As a compromise, generally, the circuits are divided into three groups. Two groups are used for one way trunks and third group is for both way trunks. An exchange first finds its outgoing trunk before trying one in common group.

This arrangement is practicable if number of exchanges in the network are limited and placed nearby (*i.e.* the lines are short). As the telephone network users are increasing at rapid rate, the concept of mesh network is uneconomical and the technological growth of transmission of signal by multiplexing made this technique to become completely obsolete.

Star topology. It is an alternative to the mesh arrangement. The network configuration shown in Fig. (*a*) is called star network. In star network, the number of lines is equal to the number of stations. As shown, a star connection utilises an intermediate exchange called a tandom exchange. Through the tandom exchange (TE) all other exchanges communicate.



(a) Star network



(b) Area with single TE and multi TE

If the number of stations served by a TE increases, they are divided into smaller network, each served by its TE. Fig. (*b*) shows the star network with splitted setup. This configuration reduces the line cost but increases the exchange costs. With the star arrangement the outer centres require fewer trunk terminations, but the trunk centre has greatly increased number of terminations. So, the star arrangement reduces the design requirements on all but one of the switching centres. It needs larger, and more powerful trunk centre. As only one larger centre is required, star arrangement preferable. Note that the exchange area indicates that all the calls in that area are considered to be local calls.

Hierarchical networks. Many star networks may be inter connected by using an additional tandom exchange, leading to two level star network. An orderly construction of multilevel star networks leads to hierarchical networks. Fig. shows the two types of hierarchical structures of AT & T and ITU–T. Hierarchical networks are capable of handling heavy traffic with minimal number of trunk groups. The hierarchical network requires more switching nodes, but achieves significant savings in the number of trunks. Determination of the total number of trunk circuits in entire network is necessarily a function of the amount of traffic between each pair of switching nodes. The efficiency of circuit utilization is the basic motivation for hierarchical switching structures. In Fig. it is shown that, if there is a high traffic intensity between any pair of exchanges, direct trunk groups may be established between any pair of exchanges (dotted lines or trunks of AT & T hierarchical network). These direct routes are known as high usage routes or trunks. In a strictly hierarchical network, traffic from subscriber A to B and vice versa flows through the highest level of hierarchy. A traffic route via the highest level of hierarchy is known as the final route. Whenever high usage route exists, route is primarily routed through them. The overflow traffic is routed through hierarchical network. Traffic is always routed through the lowest available level of the network. In addition to the high usage trunks, the tandom switches which is employed at the lowest level (not part of toll network) is augmented. The term tandom refers specifically to intermediate switching within the exchange area. The exchange area is an area within which all calls are considered to be local calls. The basic function of a tandom office is to interconnect those central offices (or class 5 or local exchanges) within an exchange area having insufficient interoffice traffic volumes to justify direct trunks. Tandom exchanges also provide alternate routes for exchange area call get blocked on direct routes between end offices.

NUMBERING PLAN

The numbering plan is used to identify the subscribers connected in a telecommunication network. The main objective of numbering plan by any nation is to standardise the number length wherever practical according to CCITT recommendations. Other objectives includes (*a*) to meet the challenges of the changing telecom environment (*b*) to meet subscriber needs for a meaningful and user friendly scheme (*c*) to reserve numbering capacity to meet the undefined future needs.

9.4.1. ITU Recommendations in Numbering

Some important recommendations of ITU are described below:

Recommendation E.164: It provides the number structure and functionality for three categories of numbers used for international public telecommunication. The three categories

of numbers are:

1. **National telephone services.** An international public telecommunication number (for geographic areas) is also referred to as the national significant number (NSN). NSN consists of the country code (CC), national destination code (NDC) and the subscriber number (SN).

2. **Global telephone services.** An international public telecommunication number for global telephone service consists of a three digit country code and global subscriber number.

The country code is always in the 8XX or 9XX range.

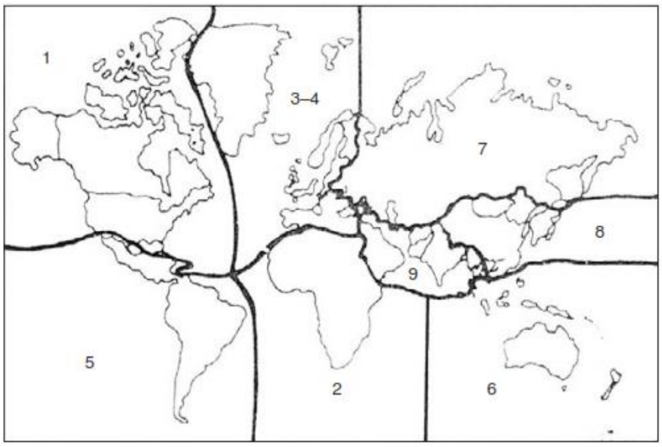
3. **International networks.** An international public telecommunication number for international networks consists of three digit country code, a network identification code and a subscriber number. The country code is always in the 8XX range. The identification code is one to four digits.

Recommendation E.123. This defines a standard way to write telephone numbers, email addresses and web addresses. It recommends how to use hyphen (-), space ( ), or period (•). ( ) are used to indicate digits that are sometimes not dialled, / is used to indicate alternate numbers and • is used in web addresses.

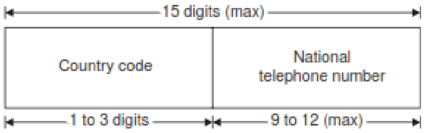
Recommendation E.162. This recommendation describes that the originating country must analyse a maximum of seven digits of the E.164 international number. When a number is being analysed, it will be done according to this recommendations. Also, the international numbering plan or world numbering plan has been defined in recommendations E.160; E.161 and E.162.

International Numbering Plan

This plan has to be implemented irrespective of a country’s national numbering plane and implemented in accordance to the recommendations of ITU. With some standard international framework, subscribers from different countries can call each other. This plan makes it possible to access all countries with the same country code anywhere in the world. For the international numbering plan, the world has been divided into nine geographical area. The general rule is that within each global region each country code starts with the same digit. Fig. 9.5 shows the geographical map of world numbering zones.

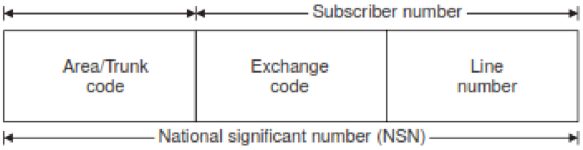


An international telephone number starts with one to three digit country code followed by 9 to 12 subscriber number. The dialling procedure is that the international prefix ‘00’ should be dialled first followed by the telephone number.



National Numbering Plane

Each country decides for itself what kind of numbering plan it can have. A numbering plan may be open, semi open or closed. Each country decides what rules to follow when issuing telephone numbers. Such a numbering plan is called national numbering plan. An open numbering plane or non-uniform numbering scheme allows variations in the number of digits to be used to identify the subscriber. This plan is used in countries equipped extensively with non-director strowger switching system. This scheme is almost extinct. A closed numbering plan or uniform numbering plan refers to a numbering plan which only allows telephone numbers of a predetermined length. Special services (toll free, premium rate, etc.) are usually excluded from this rule. A semi-open plan permits number lengths to differ by almost one or two digits. Today, this scheme is the most common and is used in many countries including India, Sweden, Switzerland and U.K. The dialling procedure for national numbering plan are also comes in two categories. A closed numbering plan refers to a numbering plan which requires users to dial all numbers at all times. This means that local-local calling also requires the area code to be dialled, as well as the trunk prefix. In open dialling plan local calls can be placed without the trunk prefix and area code.



Numbering Plan in India

DOT India has released its national numbering plan dated April 2003. It was last reviewed during 1993. This existing numbering plan was formulated at a time when there was no competition in the basic telecom services were not available in the country. Further, the existing numbering plan was meant to address monopolistic environment in national and international long distance dialling. The new numbering plan has been formulated for a projected forecast of 50% tele density by the year 2030 and thus making numbering space available for 75 crore telephone connections in the country comprising of 30 crore basic and 45 crore cellular mobile connections. The new national numbering plan will be able to meet the challenges of multi operator, multi service environment and will be flexible enough to allow for scalability for next 30 years without any change in basic structure. This plan is aimed at PSTN services, cellular mobile services and paging services.

National Numbering Scheme. There are ten levels of numbering schemes starting from 0 to 9. In each level, there are many sublevels as a classification of services. All the levels are defined briefly and some sublevels are explained in the following paragraph.

Level 0. Prefix codes. There are various sublevels. Some are described. Sublevel ‘000’ as a prefix shall be used for home country direct service (Bilateral) and international toll free service (Bilateral). The format is 000 + country code + operator code ‘000800’ is used for bilateral international toll free service. Sublevel ‘00’ as a prefix shall be used for international dialling. The format as per E.164 recommendation is 00 + country code + NSN.

Sublevel ‘0’ as a prefix shall be used for national long distance calls. The format is 0 + SDCA code + subscriber number the sublevel ‘09’ is used for cellular mobile services, satellite based services and Intelligent Network (IN) Services.

Level 1. Special services. This level is used for accessing special services like emergency services, supplementary services, inquiry and operator assisted services. The format contains 3 to *N* digit depending on service.

Level 2 to 6. PSTN subscriber number. This format contains the telephone exchange code and subscriber number.

Level 7 and 8. These two levels not being allocated and the same are reserved for new services.

Level 9. Services. The range of numbers in level ‘9’ except ‘90’, ‘95’ and ‘96’ (‘96’ is used in paging services) are reserved for cellular mobile services. Starting from 90 to 99, there are 9 sublevels. ‘90’ is used as spare not allocated for any service.

SIGNALLING TECHNIQUES

A subscriber can be able to talk with or send data to someone in any part of the world almost instantly and an exchange is able to set path and clear it after the conversation instantly by an effective signalling system. A signalling system link the variety of switching system, transmission systems and subscriber equipment in a telecommunication network to enable the network to function as a whole. The signalling are classified according to the internal signalling of an exchange, signalling between exchanges and signalling between an exchange and subscriber. Thus a signalling system must be obviously be compatible with the switching systems which itself partitioned into subsystems in a network. Traditional exchanges sent signals over the same circuit in the network. The introduction of SPC in exchanges enhances the services and introduced new services to the subscriber. These services require more signals to be transmitted and hence needs a separate data channel. The former method of signalling is referred as channel associated signalling and the latter is common channel signalling.

SIGNALLING CLASSIFICATION

Communication networks generally connect equipment such as telephones and fax machines via several line sections, switches and transmission media for exchange of speech, text and data. To achieve this, control information has to be transferred between exchanges for call control. Call control is the process of establishing and releasing a call. This is referred as signalling. In general, signalling is defines as follows. ‘‘Signalling is the process of generating and exchanging information among components of a telecommunication system to establish, monitor or release connections and to control related network and system operations’’.

(*i*) **Supervisory signals or line signals.** These are the signals necessary to initiate a call setup and to supervise it, once it has been established. It is also referred as subscriber loop signalling. Line signals can be transmitted by the use of a single control channel in each direction line.

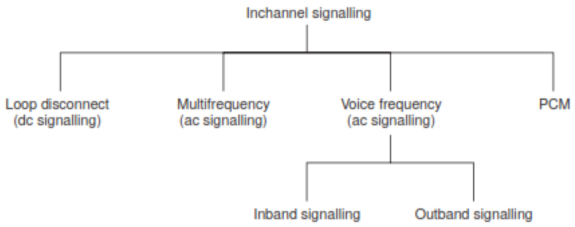
(*ii*) **Routing signals or register signals.** Information transfer related to call setup is usually referred to as register signals. The basic information is the dialled code which indicates to the subsequent switching centres the required routing. In addition to the basic information, signals such as route information, terminal information, register control signals, acknowledgement signals, status of called terminal etc. are also involved.

(*iii*) **Management signals or inter register signalling.** These signals are used to convey information or control between exchanges. This signalling also referred as inter exchange signalling. This signalling involves remote switching of private circuits, routing plans, modification of routing plans, and traffic over load, priority of the call, class of service etc. The signalling may be performed by link-by-link basis, which passes signals exchange to exchange or end-to-end signalling which is between originating and terminating exchange referred as line signalling.

Classification

Traditional signalling uses the same channel to carry voice/data and control signals to carry out the path setup for speech or data transfer. This signalling is referred as in channel signalling or per trunk signalling (PTS). An alternative to in channel signalling is called as common channel signalling (CCS). CCS, uses a separate common channel for passing control signals. It couples the signals for a large number of calls together and send them on a separate signalling channel. Hence basically, signalling technique is classified into 1. In channel signalling/per trunk signalling (PTS) 2. Common channel signalling (CCS)

The inchannel signalling is classified further into four categories as shown in Fig.


Inchannel signalling
Loop disconnect (dc signalling)
Multifrequency (ac signalling)
Voice frequency (ac signalling)
PCM
Inband signalling
Outband signalling

The common channel signalling (CCS) is classified according to the transmission of signals between exchangers. They are:

1. associated signalling

2. Non-associated signalling

3. Quasi associated signalling

INCHANNEL SIGNALLING

For data transfer and path setup for conversation, various forms if signalling were developed in automatic switching system. Various classifications are mentioned in last section. In this section, the inchannel signalling which uses same path for control signals and data/speech transfer are discussed.

Loop Disconnect - dc Signalling

The earliest and still the most common telephone set is the rotary dial telephone. The mechanism to transmit the identity of the called subscriber to exchange is pulse dialling. The basic idea is to interrupt the d.c. path of the subscriber's loop for a specified number of short periods to indicate the number dialled. This is called loop-disconnect (or rotary) signalling. The clear signal is produced when the subscriber replaces the handset, by breaking the d.c. path. Any break of more than 100 m sec is assumed as a clear signal. This signalling is relatively cheap but slow. For long distance lines due to the characteristics of lines, performance variance of equipment, the pulse shape may be degraded. This cause more errors in dc signalling.

Multi frequency (mf) – ac Signalling

The slow dialling and hence slow call processing of the rotary dial telephone reduces the productivity of the switching system. The touch tone phone created by AT & T's Western Electric Division sends out frequency based tones instead of electrical impulses. This is called a dial tone multi frequency (DTMF). When a number is pressed, two separate tones are generated and sent to the local exchange simultaneously. As each number has distinct set of dial tones, the switching system recognizes the number dialled. By rotary dial, the pulses are sent at a rate of 10 per second (*i.e.* 10 pulses per sec), whereas tone dialling generated at 23 pulses per sec (PPS). Thus tone dialling are must faster and hence quicker caller processing and increased productivity. The extra keys available in touch tone key pad enables additional/new features. The detection of the digit is carried out by using frequency filters. The frequencies are within the voice band and care must be taken to reduce the risk of speech imitations. The path setup for a rotary call is around 43 seconds, whereas with tone dialling, the call can be setup in approximately 6 to 10 seconds. An *mf* key phone has a little inter-digit pause because the numbers can be keyed very quickly.

Voice Frequency (vf) – ac Signalling

The base band of the telephone channel is 0-4000 Hz. Normally, the speech band occupies the bandwidth of 300-3400 Hz. If the signalling frequencies are chosen within the range of base band of telephone channel, then signalling is referred as voice frequency signalling. The choice of frequency for the control signal depends on various parameters. Based on the selection of frequency the voice frequency signalling are classified into two classes. They are

1. In band signalling

2. Out band signalling

This signalling is also referred as frequency division multiplex (fdm) system.

In band signalling. If the control signal frequencies are within the speech band (300–3400 Hz), the signalling is called inband signalling. Typical in the range of 2280 or 2600 Hz. As it uses the same frequency band as the voice (300–3400 Hz), it must be protected against imitations of speech. For example, if a tone is around 2600 Hz and lasts more than 50 m sec, the switching equipment assume it as a line disconnect signal. Thus, the control signal frequencies must be selected carefully to avoid this limitation. The equipment must be able to distinguish speech and signal. Various parameters are available to distinguish signal from speech. Some of them are

(*i*) Selection of frequency for signal.

(*ii*) Signal recognition time

(*iii*) Voice characteristics.

Advantages of Inband signalling:

1. Inband signalling can be used on any transmission medium.

2. The control signals can be sent to every part where a speech signal can reach.

3. Owing to the flexibility of operation, it is the most widely used signalling system for long distance telephone networks.

4. It is operations are simpler.

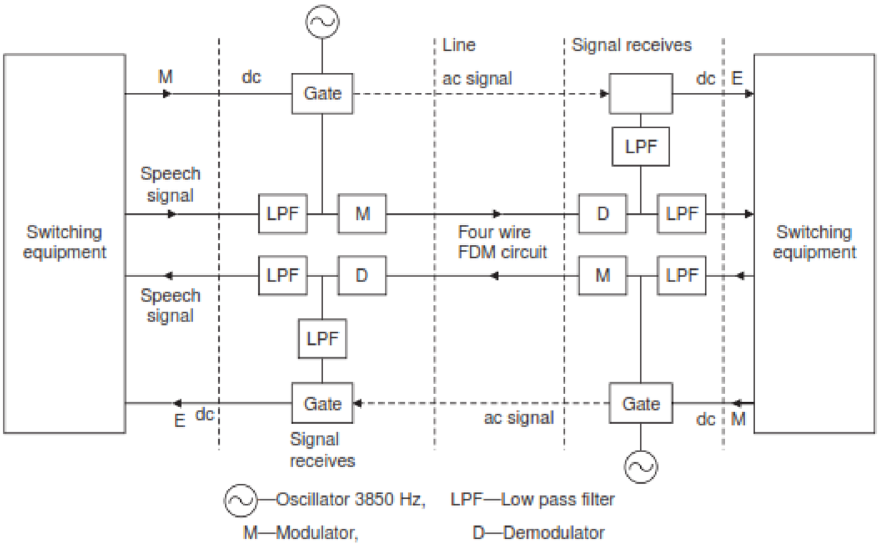
Disadvantages of Inband signalling:

1. More possibility of speech signals imitating control signals. This problem can be reduced using suitable guard circuit.

2. The inband signal may 'spill-over' from one link to another and causes error in that signalling system. This limitation occurs when several transmission links one connected end-to-end. The spill over problem can be eliminated by operating a line split to disconnect link whenever a signal is detected. The line split is designed generally to operate within 35 ms.

Outband signalling

This signalling has frequencies above the voice band but below the upper limit of 4 kHz. The CCITT recommended frequency for outband signalling is 3825 Hz, but 3700 Hz and 3850 Hz are also used. The general layout of outband signalling is shown in Fig.



Advantages:

1. The requirement of line splits are not necessary to avoid signal limitation.

2. Signals and speech can be transmitted simultaneously without disturbing the conversation.

3. Simple and consequently cheap.

Disadvantages:

1. Very narrow bandwidth is available for signalling.

2. Filtering circuits are needed to handle the signalling bands.

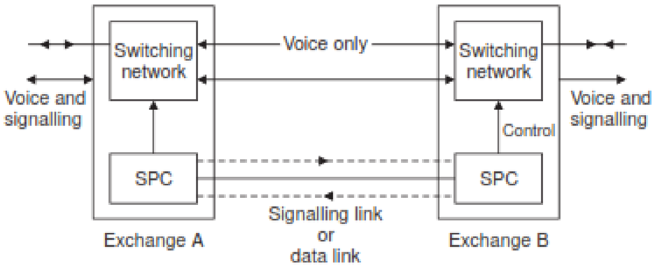
3. More dependent on the transmission system.

PCM Signalling

In PCM systems, signalling and speech are sampled, coded and transmitted within the frame of PCM channels. Thus, with PCM, a convenient way of transmission is possible. The signalling information and speech information carried in the same time slot is referred as in slot signalling. The signalling information carried in a separate time slot is referred as out slot signalling. The timeslot of the inslot signalling is fixed at eight bits. As one bit is used for signalling, the speech bit rate is reduced to 56 kbps from 64 kbps and the bandwidth is reduced. Telephone channels are combined by time division multiplexing to form an assembly of 24 or 30 channels. This is known as primary multiplex group. Two frame structures are widely used in practice. They are DS1 24 channel system and European 30 channel system. DS1 24 channel system is popular in North America and Japan. Originally, DS1 is called as T1 system. T1 (Telecommunication standards entity number 1) is a standards committee designated as ANSI T1-nnn-date. T1 uses some very specific conventions to transmit information between both ends. One of these is a framing sequences that formats the samples of voice or data transmission. Framing undergone several evolutions.

COMMON CHANNEL SIGNALLING

Introduction of SPC digital switching systems with high speed processors in the telecom network has necessitated modernisation of signalling. Also, in order to meet the transfer of varieties of information for call management and network management and to satisfy the subscribers’ requirement on various features, the uninterrupted, high speed signalling has become inevitable. The rapid development of digital systems paved way for the new signalling system called common channel signalling. Instead of using the same link for signalling information and message as in In channel signalling, the common channel signalling (CCS) uses a dedicated line for the signalling information between the stored program control elements of switching systems. Fig. shows the basic schematic of CCS.



The data link sends messages that identify specific trunks and events. Two signalling channels, one for each direction are used in a dedicated manner to carry signalling information. Hence, they are capable of carrying information for a group of circuits. At the bit rate of 2.6 kbps, CCS can carry signals for 1500–2000 speech circuits. CCS network is basically a store and forward network. In CCS network, the signalling information travels in a link-by-link basis along the route. The information arrived at a node is sorted, processed and forwarded to the next node in the route. The CCS technique is also called the transparent mode for signalling.

Advantages and Disadvantages of CCS

Advantages. The major advantages of CCS are listed below:

1. For each associated trunk group, only one set of signalling facility is required. The channel used for CCS need not be associated with any particular group. Fig. shows a CCS network that is disassociated from the message network structure.

2. The introduction of SPC switching machines and CCS provides efficient routing procedure.

3. CCS allows for signalling at any time in the entire duration of a call, not only at the beginning.

4. CCS removes most of the signalling costs associated with inter office trunks.

5. Information can be exchanged between processor at high speed.

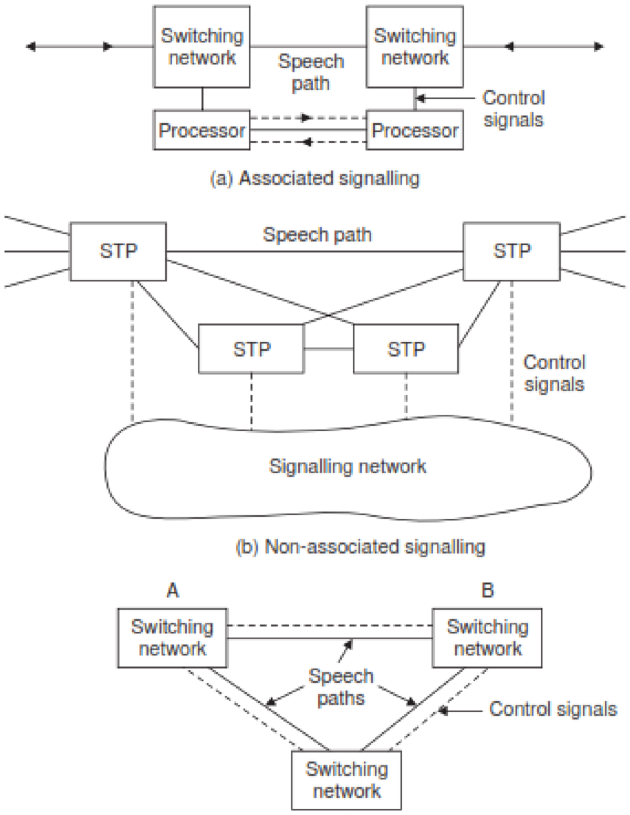
6. The CCS provides acceptable quality for network related signalling tones such as DTMF, MF and SF achieved.

7. As there is no need of line signalling equipment on every function, considerable cost savings can be achieved.

8. The D1 channel bank use 1 bit per time slot for signalling and 7 bits for voice which provides signalling rate 8 kbps. D2 channel provides higher data rate which use 1 bit in every sixth frame for signalling. This is referred as ‘‘robbed bit signalling.’’ When CCS is utilized, the associated T carrier system no longer need to carry signalling information on a per channel basis and a full 8 bits of voice can be transmitted in every time slot of every frame.

9. As separate channels are used for voice and control. There is no chance of mutual interference and the error rate is very low.

10. CCS enables more services to the subscribers. A signalling link operating at 64 kbit/s normally provides signalling up to 1000 or 1500 speech circuits.



(a) Associated signalling

(b) Non-associated signalling

(c) Quasi-associated signalling

Disadvantages of CCS:

Some disadvantages of CCS are listed below:

1. The CCS network is basically a store and forward network. So, in a established circuit, the signalling information are stored, processed and then forwarded to next node. This causes additional overhead and disconnect to the continuity.

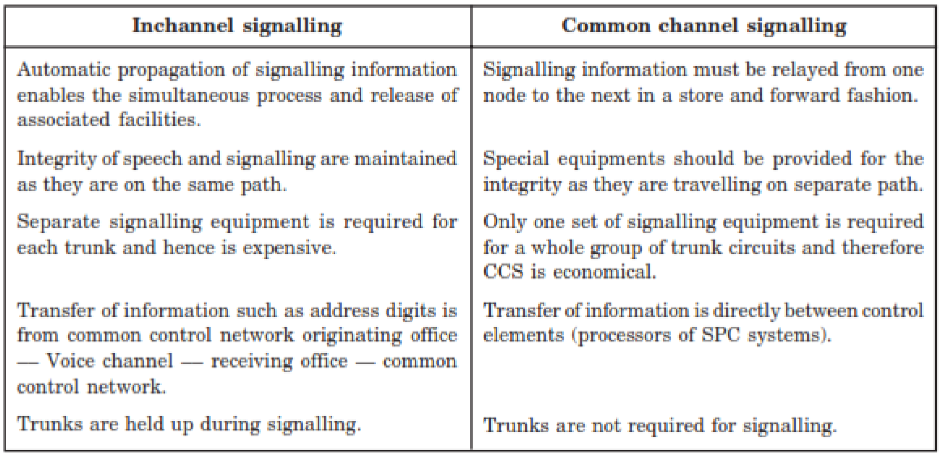
2. If one node fails to transmit properly, the facilities downstream from the disconnect will not be released. Thus, a high degree of reliability is required for the common channel.

3. Proper interfacing facility is necessary, as most of the present day telephone networks are equipment with inchannel signalling systems.

4. As the signalling information is not actually sent over speech paths in CCS, the integrity of speech path is not assured. As a remedy, routing testing of idle paths and the continuity test of an established path become necessary in CCS.

5. Different trunks in a group may terminate at different switch, say local exchange, other foreign exchange circuits etc. With CCS, all trunks are first terminated to the local central office and then forward to the different destination.

In channel and common channel signalling comparison is tabulated in Table.



Inchannel signalling	**Common channel signalling**
Automatic propagation of signalling information enables the simultaneous process and release of associated facilities.	Signalling information must be relayed from one node to the next in a store and forward fashion.
Integrity of speech and signalling are maintained as they are on the same path.	Special equipments should be provided for the integrity as they are travelling on separate path.
Separate signalling equipment is required for each trunk and hence is expensive.	Only one set of signalling equipment is required for a whole group of trunk circuits and therefore CCS is economical.
Transfer of information such as address digits is from common control network originating office — Voice channel — receiving office — common control network.	Transfer of information is directly between control elements (processors of SPC systems).
Trunks are held up during signalling.	Trunks are not required for signalling.

Module-IV: OVERVIEW OF

ISDN

Integrated Services Digital Network is a telephone system network. It is a wide area network becoming widely available. Prior to the ISDN, the phone system was viewed as a way to transport voice, with some special services available for data. The key feature of the ISDN is that it integrates speech and data on the same lines, adding features that were not available in the classic telephone system.

ISDN is a circuit -switched telephone network system, that also provides access to packet switched networks, designed to allow digital transmission of voice and data over ordinary telephone copper wires, resulting in better voice quality than· an analog phone. It offers circuit-switched connections (for either voice or data), and packet-switched connections (for data), in increments of 64 Kbit/s.

Another major market application is Internet access, where ISDN typically provides a maximum of 128 Kbit/s in both upstream and downstream directions (which can be considered to be broadband speed, since it exceeds the narrowband speeds of standard analog 56k telephone lines). ISDN B-channels can be bonded to achieve a greater data rate; typically 3 or 4 BRIs (6 to 8 64 Kbit/s channels) are bonded.

ISDN should not be mistaken for its use with a specific protocol, such as Q.931 whereby ISDN is employed as the network, data-link and physical layers in the context of the OSI model. In a broad sense ISDN can be considered a suite of digital services existing on layers 1, 2 and 3 of the OSI model. ISDN is designed to provide access to voice and data services simultaneously.

However, common use has reduced ISDN to be limited to Q.931 and related protocols, which are a set of protocols for establishing and breaking circuit switched connections, and for advanced call features for the user. They were introduced in 1986. In a videoconference, ISDN provides simultaneous voice, video, and text transmission between individual desktop videoconferencing systems and group (room) videoconferencing systems.

The first generation of ISDN is called as a narrowband ISDN and it is based on the use of 64 kbps channel as the basic unit of switching and has a circuit switching orientation. The main device in the narrowband ISDN is the frame relay. The second generation of ISDN is referred to as the broadband ISDN (B-ISDN).

It supports very high data rates (typically hundreds of Mbps). It has a packet switching orientation. The main important technical contribution of B-ISDN is the asynchronous transfer mode (ATM), which is also called as cell relay.

ISDN Interfaces

There are several kinds of access interfaces to the ISDN:

Basic Rate Interface (BRI)

Basic Rate Interface service consists of two data-bearing channels ('B' channels) and one signalling channel ('D' channel) to initiate connections. The B channels operate at 64 Kbps maximum; however, (in the U.S. it can be limited to 56 Kbps.

The D channel operates at a maximum of 16 Kbps. The two channels can operate independently. For example, one channel can be used to send a fax to a remote location, while the other channel is used as a TCP/IP connection to a different location. ISDN service on the iSeries supports basic rate interface (BRl).

The basic rate interface (BRl) specifies a digital pipe consisting of two B channels and 16 Kbps D channel. Two B channels of 64 Kbps each, plus one D channel of 16 Kbps, equal 144 Kbps. In addition, the BRl service itself requires 48 Kbps of operating overhead. BRl therefore requires a digital pipe of 192 Kbps. Conceptually, the BRl service is like a large pipe that contains three smaller pipes, two for the B channels and one for the D channel.

The remainder of the space inside the large pipe carries the overhead bits required for its operation.

Primary Rate Interface (PRI)

Primary Rate Interface service consists of a D channel and either 23 (depending on the country you are in). PRI is not supported on the iSeries. Or 30 B channels

The usual Primary Rate Interface (PRI) specifies a digital pipe with 23 *B* channels and one 64 Kbps D channel. Twenty-three B channels of 64 Kbps each, plus one D channel of 64 Kbps equals 1.536 Mbps. In addition, the PRI service itself uses 8 Kbps of overhead.

PRI therefore requires a digital pipe of 1.544 Mbps. Conceptually; the PRI service is like a *large* pipe containing 24 smaller pipes, 23 for the B channels and 1 for the D channel. The rest of the pipe carries the overhead bits required for its operation.

Broadband-ISDN (B-ISDN)

Narrowband ISDN has been designed to operate over the current communications infrastructure, which is heavily dependent on the copper cable. B-ISDN however, relies mainly on the evolution of fibre optics. According to CCITT B-ISDN is best described as 'a service requiring transmission channels capable of supporting rates greater than the primary rate.

Principle of ISDN

The ISDN works based on the standards defined by ITU-T (formerly CCITT). (The Telecommunication Standardization Sector (ITU-T) coordinates standards for telecommunications on behalf of the International Telecommunication Union (ITU) and is based in Geneva, Switzerland. The standardization work of ITU dates back to 1865, with the birth of the International Telegraph Union.

It became a United Nations specialized agency in 1947, and the International Telegraph and Telephone Consultative Committee (CCITT), (from the French name "Comite Consultatif International Telephonique et Telegraphique") was created in 1956. It was renamed ITU-T in 1993.

Principle of ISDN according to ITU –T

The ISDN is supported by a wide range of voice and non-voice applications of the same network. It provides a range of services· using a limited set of connections and multipurpose user-network interface arrangements.

ISDN supports a variety of applications that include both switched and non-switched connections. The switched connections. Include both circuit and packet switched connections.

As far as possible, new services introduced into an ISDN should be arranged to be compatible with the 64 Kbps switched digital connections.

A layered protocol structure should be used for the specification of access to an ISDN.

This is the same as the OSI reference model. The standards which have already been developed for OSI applications such as X.25 can be used for ISDN.

ISDNs may be implemented in a variety of configurations.

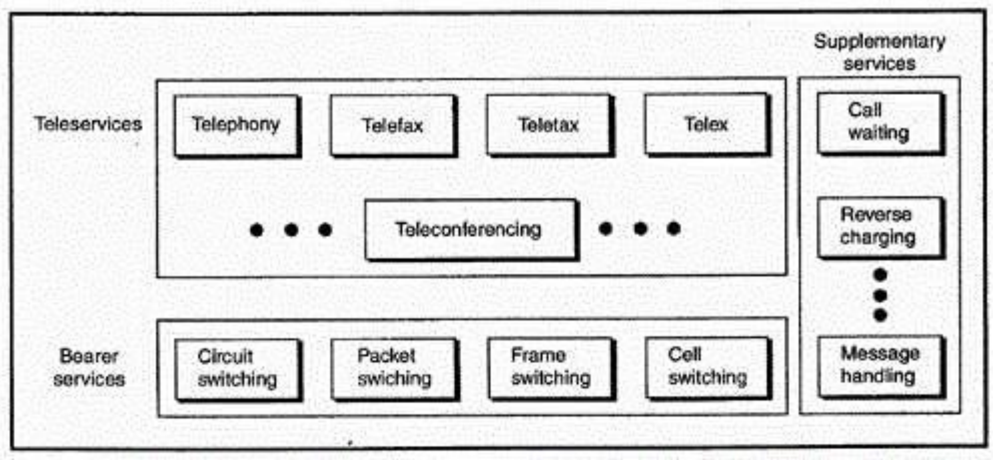
ISDN Services

The purpose of the ISDN is to provide fully integrated digital services to users. These services fall into categories- better services, teleservices and supplementary services.

1. Bearer Services: Bearer services provide the means to transfer information (voice, data and video) between users without the network manipulating the content of that information. The network does not need to process the information and therefore does not change the content.

Bearer services belong to the *first* three layers of the OSI model and are well defined in the ISDN standard. They can be provided using circuit-switched, packet-switched, frame-switched, or cell-switched networks.

2. Teleservices: In teleservices, the network may change or process the contents of the data. These services correspond to layers 4-7 of the OSI model. Teleservices relay on the facilities of the bearer services and are designed to accommodate complex user needs, without the user having to be aware of the details of the process. Teleservices include telephony, teletex, telefax, videotex, telex and teleconferencing. Although the ISDN defines these services by name, they have not yet become standards.



3. Supplementary Service: Supplementary services are those services that provide additional functionality to the bearer services and teleservices. Examples of these services are reverse charging, call waiting, and message handling, all familiar from today's telephone company services.

Principles of ISDN

The various principles of ISDN as per ITU-T recommendation are:

I. To support voice and non-voice applications

The main feature of the ISDN concept is the support of a wide range of voice (for *e.g.* Telephone calls) & non-voice (for *e.g.* digital data exchange) applications in the same network.

2. To support switched and non-switched applications

ISDN supports both circuit switching and packet switching. In addition ISDN supports non-switched services in the form of dedicated lines.

3. Reliance on 64-kbps connections

ISDN provides circuit switched and packet switched connections at 64 kbps. This is the fundamental building block of ISDN. This rate was chosen because at the time, it was standard rate for digitized voice.

4. Intelligence in the network

An ISDN is expected to provide sophisticated services beyond the simple setup of circuit switched calls. These services include maintenance and network management functions.

5. Layered protocol architecture

A layered protocol structure should be used for the specification of the access to an ISDN. Such a structure can be mapped into OSI model.

6. Variety of configurations

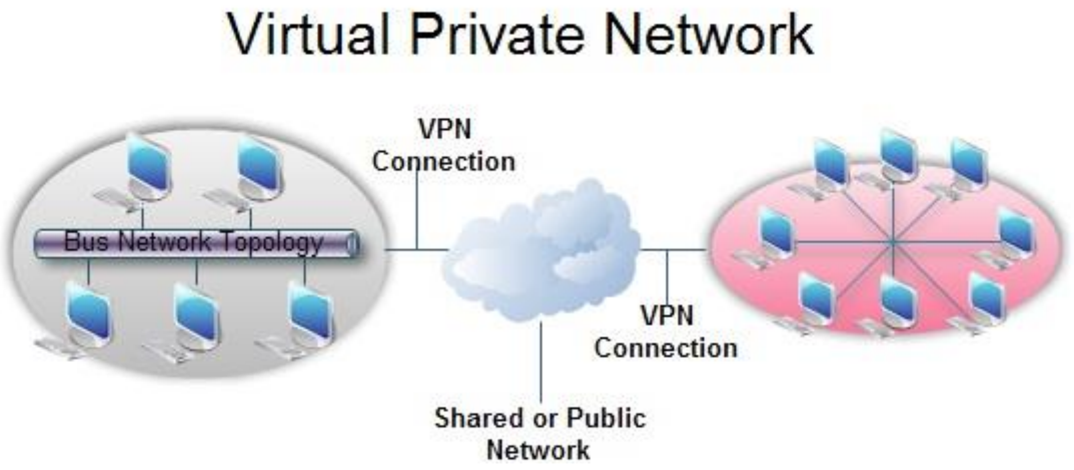
Several configurations are possible for implementing ISDN. This allows for differences in national policy, in state of technology and in the needs and existing equipment of the customer base.

VPN (*Virtual Private Network*) *Definition*: VPN meaning that it is a private point-to-point connection between two machines or networks over a shared or public network such as the internet. VPN (Virtual Private Network) technology, can be use in organization to extend its safe encrypted connection over less secure internet to connect remote users, branch offices, and partner private, internal network. VPN turn the Internet into a simulated private WAN.

A VPN client uses TCP/IP protocol, that is called tunnelling protocols, to make a virtual call to VPN server.

What is VPN

VPN allows users working at home or office to connect in a secure fashion to a remote corporate server using the routing infrastructure provided by a public inter-network (such as the Internet). From the user's perspective, the VPN is a point-to-point connection between the user's computer and a corporate server. The nature of the intermediate inter-network is irrelevant to the user because it appears as if the data is being sent over a dedicated private link.



Main Network Protocols

There are three network protocols are used within VPN tunnels. That are:

IPSec

IPsec (*Internet Protocol Security*) is a framework for uses cryptographic security services developed by the IETF to protect secure exchange communications over Internet Protocol (IP).It supported encryption modes are transport and tunnel.

PPTP

PPTP (*Point-to-Point Tunneling Protocol*) is a network protocol that extending the organization private networks over the public Internet via "tunnels.

L2TP

L2TP (*Layer Two Tunneling Protocol*) is an extension of the Point-to-Point Tunneling Protocol (PPTP) used by Internet service providers (ISPs) to operate Virtual Private Networks (VPNs).

Privacy, Security and Encryption

Data sent across the public Internet is generally not protected from curious eyes, but you can make your Internet communications secure and extend your private network with a virtual private network (VPN) connection. VPN uses a technique known as tunneling to transfer data securely on the Internet to a remote access.

The Internet connection over the VPN is encrypted and secure. New authentication and encryption protocols are enforced by the remote access server. Sensitive data is hidden from the public, but it is securely accessible to appropriate users through a VPN.

How to Setup a VPN

There are following two ways to create a VPN connection:

By dialling an Internet service provider (ISP)

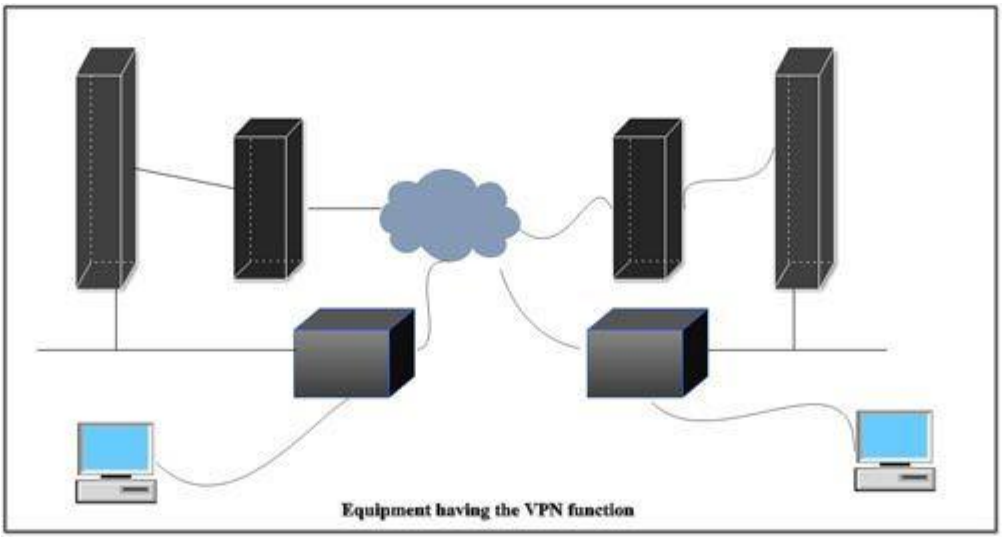
If you dial-in to an ISP, your ISP then makes another call to the private network's remote access server to establish the PPTP or L2TP tunnel after authentication, you can access the private network.

By connecting directly to the Internet

If you are already connected to an Internet, on a local area network, a cable modem, or a digital subscriber line (DSL), you can make a tunnel through the Internet and connects directly to the remote access server. After authentication, you can access the corporate network.

Equipment of VPN

Equipment having the VPN function includes routers and firewalls. Basically, communication is made via VPN equipment. Information is encrypted by the transmission VPN equipment before transmission and decoded by the receiving VPN equipment after receipt of information. The key for encrypt the data is set in VPN equipment in advance. The VPN equipment at receiving side decodes encrypted data before sending it to the receiving computer.



Equipment having the VPN function

The advantages of encryption by way of cryptography may be looked into other services, such as

1. Assuring integrity check: This ensures that undesirable person has not tampered data delivered to the destination during transmission.
2. Providing authentication: Authentication authorizes the sender identity.

Features of a Typical VPN solution

When the remote offices connect each other to share vital resources and secret information, the VPN solution must ensure the privacy and integrity of the data as it traverses the Internet. Therefore, a VPN solution must provide at least all of the following:

Keep data confidential (encryption)

- Data carried on the public network must be rendered unreadable to unauthorized clients on the network.

Ensure the identities of two parties communicating (authentication)

- The solution must verify the user's identity and restrict VPN access to authorized users only. It must also provide audit and accounting records to show who accessed what information and when.
- Safeguard the identities of communicating parties (tunnelling)
- Guard against packets being sent over and over (replay prevention)
- Ensure data is accurate and in its original form (non-repudiation)

Address Management. The solution must assign a client's address on the private net and ensure that private addresses are kept private.

Key Management. The solution must generate and refresh encryption keys for the client and the server.

Multiprotocol Support. The solution must handle common protocols used in the public network. These include IP, Internet Packet Exchange (IPX), and so on.

An Internet VPN solution based on the Point-to-Point Tunneling Protocol (PPTP) or Layer 2 Tunneling Protocol (L2TP) meets all of these basic requirements and takes advantage of the broad availability of the Internet. Other solutions, including the new IP Security Protocol (IPSec), meet only some of these requirements, but remain useful for specific situations.

Benefits of VPN

The main benefit of a VPN is the potential for significant cost savings compared to traditional leased lines or dial up networking. These savings come with a certain (in amount of risk, however, particularly when using the public Internet as the delivery mechanism for VPN data.

The performance of a VPN will be more unpredictable and generally slower than dedicated lines due to public Net traffic. Likewise, many more points of failure can affect a Net-based VPN than in a closed private system. Utilizing any public network for communications naturally raises new security concerns not present when using more controlled environments like point-to-point leased lines.

Voice over Internet Protocol, also called Voice over IP or just VoIP technology is having a major impact on the telecommunications industry. VoIP technology provides advantages for both the user and also the provider, allowing calls to be made more cheaply, as well as enabling data and voice to be carried over the same network efficiently. In view of the way VoIP technology is being adopted, telecommunications providers are having to adopt the new technology. Already it has caused some impact on major businesses, and there will be more to come.

Until recently voice traffic was carried using a circuit switched approach. Here a dedicated circuit was switched to provide a call for a user. Now with new data and Internet style technology used for VoIP, packet data and Internet Protocol (IP) is used to enable a much more efficient use of the available capacity.

What is VoIP?

- *VoIP definition:* VoIP (voice over IP) is the transmission of voice and multimedia content over Internet Protocol (IP) networks. VoIP is enabled by a group of technologies and methodologies used to deliver voice communications over the internet, enterprise local area networks or wide area networks.

The concept of Voice over Interent Protocol, Voice over IP, or VoIP, is quite straightforward. A VoIP system basically consists of a number of endpoints which may be VoIP phones, mobile phones, VoIP enabled browsers on computers, etc and also an IP network over which the packet data is carried.

In a VoIP system, the phone or computer acting as an endpoint consists of a few blocks. It includes a vocoder (voice encoder / decoder) which converts the audio to and from the analogue format into a digital format. It also compresses the encoded audio, and in the reverse direction it decompresses the reconstituted audio. The data generated is split into packets in the required format by the network interface card which sends them with the relevant protocol into the outside world. Signalling and call control is also applied through this card so that calls may be set up, pulled down, and other actions may be undertaken.

The IP network accepts the packets and provides the medium over which they can be forwarded, routing them to their final destination. As complete circuits are not dedicated to a given user, at times when no data needs to be sent, for example during quiet periods in speech, etc., the capacity can be used by other users. This makes a significant difference to the efficiency of a system, and allows significant savings to be made.

Traditionally the term VoIP referred to systems where IP was used to connect private branch exchanges, PBXs, but now the term is more widely used and encompasses IP telephony.

VoIP Protocols

In order to be able to communicate using a VoIP system, there are a number of protocols that may be used.

- *H323:* The signalling protocol is used to control and manage the call. It includes elements such as call set up, clear down, call forwarding and the like. The first protocol to be widely used for VoIP was H323. However this is not a particularly rigorous definition and as a result other variants have been developed.
- *Skinny:* One other signalling protocol that was used was known as "Skinny" and is a Cisco Proprietary protocol and is from Nortel and another is called Unistem. In view of this there are often interfacing problems.
- *SIP:* SIP, Session Initiation Protocol, is now being widely adopted as the main standard is a far more rigorous protocol for signalling and is the one that is most widely used now.

- *RTP:* RTP, Real Time Protocol, is a data exchange protocol and this can handle both audio and video. RTP handles the data exchange, but in addition to this a codec is required. Where voice is used a vocoder is used (a codec can be used for any form of data including audio, video, etc).
- *G711:* G711 is possibly the most widely used VoIP vocoder and it is the standard for transmitting uncompressed packets. G.729 is the standard for compressed packets. Many equipment vendors also use their own proprietary codecs. Voice quality may suffer when compression is used, but compression reduces bandwidth requirements. There are many other vocoders/ codecs that are used with varying data rates and providing different levels of voice quality.

Service quality

Quality of Service, QoS, for the data link has a major impact on VoIP perceived sound quality. The data exchange must take place in real time and any delays in the system cause significant disruption to the traffic. Delayed packets may mean that packets arrive out of order, or with varying gaps between them, resulting in garbled speech, Packets may even disappear resulting in lost information.

For any packet passing through an IP network it is possible to define the class of service required. It is important that packets that need to be transferred in real time are given a higher quality of service than those that can be transferred as the network permits. This is particularly important for services like VoIP that are termed delay sensitive applications.

Advantages

Voice over IP, VoIP technology provides a number of significant advantages to operators and to users. For the user one of the main advantages is the flexibility. Phones are software based, sometimes being attached to computers. As a result a considerable degree of flexibility is afforded to the user. It is possible to move the phone around and by enabling the system to recognise the individual phone it is possible to route the data to it automatically. In addition to this ideas such as mobile IP could enable the user to be located away from the home network and still receive calls.

A further advantage is that the wireless network technologies such as 802.11 can carry the calls as voice is simply another form of application. This gives further flexibility as the phone does not have to be physically wired to a network. Again Quality of Service is a major factor and this is being addressed under 802.11e

For the operator some of the advantages are different. One of the major drivers towards the use of VoIP is cost. Previously digital traffic was handled using time division techniques. This had the disadvantage that when a particular time slot allocated to a user was dormant, it could not be used. Using IP techniques much higher levels of efficiency can be attained. Although the system required to carry packet data is more complicated, the returns far outweigh the additional costs.

As with all technologies there are disadvantages. The main one with VoIP is voice quality. This results from the use of a vocoder to digitise and compress the audio. Quality is comparable with that from a mobile phone, but for the future with rapidly improving standards of vocoders there are likely to be significant improvements in this area.

In the long term VoIP is the way the market is moving, and now with increasing speed. Offering not only great improvements in flexibility, but also major cost savings, but with the requirement for large levels of investment, this is the way that the telecommunications market is moving. However to remain competitive it will be necessary to adopt the new VoIP technology.

VoIP protocols overview

Although working together, there are a number of different organizations and bodies that are mentioned when referring to VoIP protocols:

- IETF This is the Internet Engineering Task Force. It is a community of engineers that defines some of the prominent standards used on the Internet (including VoIP protocols) and seeks to spread understanding of how they work.
- ITU the International Telecommunication Union. This is an international organization within the United Nations System used by where governments and private sector companies to coordinate and standardize telecommunications networks, services and standards on a global basis.

In addition to the organizations involved, there is also a variety of different VoIP protocols and standards.

- H.248 H.248 is an ITU Recommendation that defines "Gateway Control Protocol" and it is also referred to as IETF RFC 2885 (Megaco). It defines a centralized architecture for creating multimedia applications and it extends MGCP. H.248 is the result of a joint collaboration between the ITU and the IETF and it is another VoIP protocol.
- H.323 This is ITU Recommendation that defines "packet-based multimedia communications systems." H.323 defines a distributed architecture for multimedia applications, and it is thus a VoIP protocol.
- Megaco This is also known as IETF RFC 2885 and ITU Recommendation H.248. H.248 defines a centralized architecture for creating multimedia applications.
- Media Gateway Control Protocol (MGCP) This is also known as IETF RFC 2705. It defines a centralized architecture for creating multimedia applications, and it is therefore a VoIP protocol.
- Real-Time Transport Protocol (RTP) This VoIP protocol is defined under IETF RFC 1889 and it details a transport protocol for real-time applications. RTP provides the transport mechanism to carry the audio/media portion of VoIP communication and is used for all VoIP communications.
- Session Initiation Protocol (SIP) This is also known as IETF RFC 2543 and it defines a distributed architecture for creating multimedia applications.

Centralised and distributed architectures

One of the advantages of VoIP is that it does not legislate for the architecture of the network that carries the data. Early telecommunications networks used a centralised structure where all the intelligence was contained at the switching station or exchange. With the advent of packet technology, the routing and intelligence can be distributed to where it is most convenient to locate it. This may be by having a distributed architecture, or a centralised one. While both architectures can be employed with VoIP, the type of architecture does have an impact on the optimum VoIP protocols to use. This is one of the reasons why a number of VoIP protocols are used, and will remain to be used.

P networks carrying VoIP traffic are very complicated. They carry both voice and data traffic and this results in a variety of traffic with different requirements being carried and this presents many challenges. In order to ensure that all the requirements are met and the network operates to its maximum efficiency can present many challenges. Obviously the design must be correct, but once implemented testing of the network is needed to ensure that it is able to operate correctly when installed, and then maintained correctly ensuring that its performance is maintained or optimised to provide the performance meets the needs of the network provider and the user. For VoIP, testing is an essential element of any network. However specialised VoIP testing techniques are required.

VoIP network architecture

The structure of a VoIP network comprises many entities and this means that VoIP testing is essential to ensure that the network is operating satisfactorily. A typical VoIP network will include many different entities:

- Signalling gateways
- Media gateways
- Gatekeepers
- Class 5 switches
- SS7 network
- Network management system
- Billing system

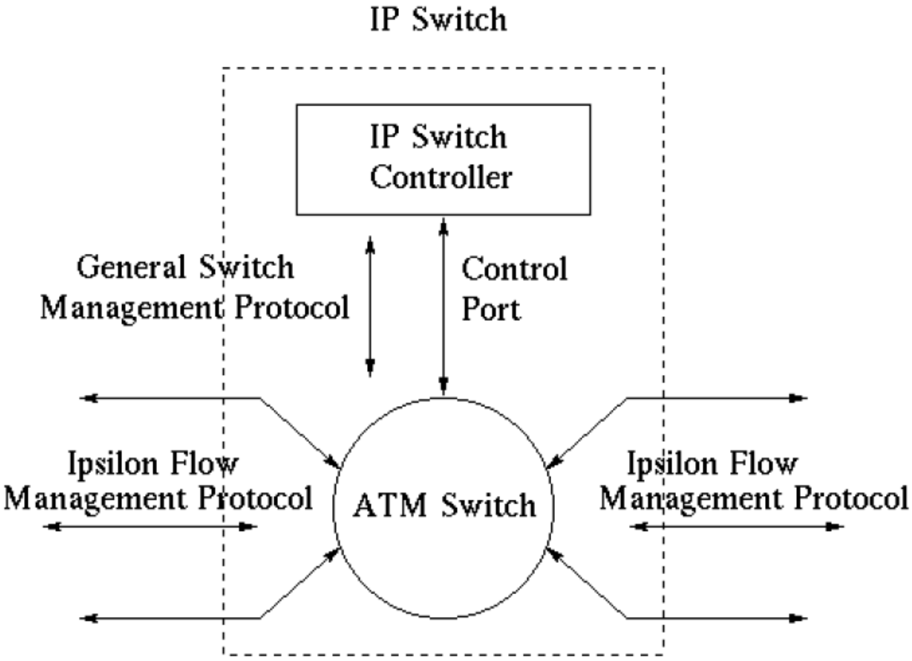
This variety of different entities within the VoIP network all communicate with each other using a variety of protocols. To perform correctly it is necessary to ensure that they communicate efficiently and that no bottlenecks are created. Analysing the performance of a VoIP network is not always easy. However it can be achieved and significant improvements in performance can be achieved if the VoIP testing scenarios are carefully chosen and planned, and the data analysed to reveal any problems.

IP Switching

This technology [Flow Labelled IP] has been developed by Ipsilon Networks Inc. The goal of Ipsilon is to make IP faster and offer the quality of service support. The "IP over ATM" approach tries to hide the underlying network topology from IP layer by treating the datalink layer as large, opaque network cloud. However, this leads to inefficiency, complexity and duplication of functionality in the resulting network [ATM under IP]. Ipsilon's approach is to discard the connection-oriented ATM software and implement the connectionless IP routing directly on the top of ATM hardware. This approach takes the advantage of robustness and scalability of connectionless IP and speed, capacity and scalability of ATM switches.

IP Switch

IP switch is basically an IP router with attached switching hardware that has the ability to cache routing decisions in switching hardware. To construct an IP switch, a standard ATM switch is taken, the hardware is left untouched, but all the control software above AAL-5 is removed. It is replaced by standard IP routing software, a flow classifier to decide whether to switch a flow or not and a driver to control the switch hardware. At system start up a default virtual channel is established between the control software of the IP switch and its neighbours, which is then used for default hop-by-hop forwarding of IP datagrams. To gain the benefits of switching, a mechanism has been defined to associate IP flows with the ATM labels.


IP Switch
IP Switch Controller
General Switch Management Protocol
Control Port
Ipsilon Flow Management Protocol
ATM Switch
Ipsilon Flow Management Protocol

Structure of an IP switch

Flow Classification

A flow is a sequence of packets from a source to a destination which get the same forwarding treatment at a router. An IP flow is characterized by the fields in IP/TCP/UDP header such as source address, destination address, port number, etc. Currently two types of flows have been defined by Ipsilon. Host pair flow type is defined for traffic between same source and destination IP address. A Port pair flow type is defined for traffic between same source and destination port between same source and destination IP address. When a packet is assembled and submitted to controller in IP switch, apart from forwarding it to the next hop it also does flow classification. Flow classification decision is a policy decision local to a IP switch. Depending on the classification, the switch decides to forward or switch the next packets of that flow. Usually the controller would decide to forward short term flows like database queries, DNS messages and switch long term flows like ftp or telnet data.

IP Switching Features

Following are some important features of IP Switching:

- Point-to-Point IP Switching advocates point-to-point network model for ATM rather than a logical shared medium model as proposed by some competing approaches.

- Multicast In case of multicast, a flow coming to an IP Switch will branch out to multiple destinations. Each of these branches can be individually redirected. If the whole flow is labelled, then multicast capabilities of ATM can be used.

- Quality of Service, An IP Switch can make QoS decision according to its local policy. The QoS information can be included in the flow classification process. Individual QoS requests for each flow can be supported using resource reservation protocols like RSVP. An IP Switch can be configured to utilize traffic management capabilities of ATM to guarantee the desired QoS.

- Latency The latency in setting up of a virtual channel from source to destination can be low for connection oriented protocols like TCP. Due to 3-way handshake in TCP setup procedure, a virtual channel can be established even before the first data packet is sent across. Also the tolerance in case of failures is good because the IP Switches can always go back to connectionless packet forwarding via a different route if any link fails.

GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

Digital Electronics & Microprocessor Engineering

For 4th Semester

ELECTRICAL

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Mr. Sarada Prasanna Singh

(Lecturer in Electronics & Telecommunication Engineering)

NUMBER SYSTEM AND CODES

INTRODUCTION:-

- The term digital refers to a process that is achieved by using discrete unit.
- In number system there are different symbols and each symbol has an absolute value and also has place value.

RADIX OR BASE:-

The radix or base of a number system is defined as the number of different digits which can occur in each position in the number system.

RADIX POINT :-

The generalized form of a decimal point is known as radix point. In any positional number system the radix point divides the integer and fractional part.

N_r = [Integer part **.** Fractional part]

↑

Radix point

NUMBER SYSTEM:-

In general a number in a system having base or radix ‘ r ’ can be written as

$$\mathbf{a_n \;\; a_{n-1} \;\; a_{n-2} \; \ldots\ldots\ldots\ldots\ldots \; a_0 \; . \; a_{-1} \;\; a_{-2} \ldots\ldots\ldots\ldots a_{-m}}$$

This will be interpreted as

$$\mathbf{Y = a_n \times r^n + a_{n-1} \times r^{n-1} + a_{n-2} \times r^{n-2} + \ldots\ldots\ldots + a_0 \times r^0 + a_{-1} \times r^{-1} + a_{-2} \times r^{-2} + \ldots\ldots\ldots + a_{-m} \times r^{-m}}$$

where Y = value of the entire number

a_n = the value of the nth digit

r = radix

TYPES OF NUMBER SYSTEM:-

There are four types of number systems. They are

1. Decimal number system
2. Binary number system
3. Octal number system
4. Hexadecimal number system

DECIMAL NUMBER SYSTEM:-

- The decimal number system contain ten unique symbols 0,1,2,3,4,5,6,7,8 and 9.
- In decimal system 10 symbols are involved, so the base or radix is 10.
- It is a positional weighted system.
- The value attached to the symbol depends on its location with respect to the decimal point.

In general,

$$d_n \quad d_{n-1} \quad d_{n-2} \ \ldots\ldots\ldots\ldots\ldots \ d_0 \ . \ d_{-1} \ d_{-2} \ldots\ldots\ldots\ldots d_{-m}$$

is given by

$$(d_n \times 10^n) + (d_{n-1} \times 10^{n-1}) + (d_{n-2} \times 10^{n-2}) + \ldots + (d_0 \times 10^0) + (d_{-1} \times 10^{-1}) + (d_{-2} \times 10^{-2}) + \ldots + (d_{-m} \times 10^{-m})$$

For example:-

9256.26 = 9 x 1000 + 2 x 100 + 5 x 10 + 6 x 1 + 2 x (1/10) + 6 x (1/100)

$= 9 \times 10^3 + 2 \times 10^2 + 5 \times 10^1 + 6 \times 10^0 + 2 \times 10^{-1} + 6 \times 10^{-2}$

BINARY NUMBER SYSTEM:-

- The binary number system is a positional weighted system.
- The base or radix of this number system is 2.
- It has two independent symbols.
- The symbols used are 0 and 1.
- A binary digit is called a bit.
- The binary point separates the integer and fraction parts.

In general,

$$d_n \quad d_{n-1} \quad d_{n-2} \ \ldots\ldots\ldots\ldots\ldots \ d_0 \ . \ d_{-1} \ d_{-2} \ldots\ldots\ldots\ldots d_{-k}$$

is given by

$$(d_n \times 2^n) + (d_{n-1} \times 2^{n-1}) + (d_{n-2} \times 2^{n-2}) + \ldots. + (d_0 \times 2^0) + (d_{-1} \times 2^{-1}) + (d_{-2} \times 2^{-2}) + \ldots. + (d_{-k} \times 2^{-k})$$

OCTAL NUMBER SYSTEM:-

- It is also a positional weighted system.
- Its base or radix is 8.
- It has 8 independent symbols 0,1,2,3,4,5,6 and 7.
- Its base 8 = 2^3 , every 3- bit group of binary can be represented by an octal digit.

HEXADECIMAL NUMBER SYSTEM:-

- The hexadecimal number system is a positional weighted system.
- The base or radix of this number system is 16.
- The symbols used are 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E and F
- The base 16 = 24 , every 4 – bit group of binary can be represented by an hexadecimal digit.

CONVERSION FROM ONE NUMBER SYSTEM TO ANOTHER :-

1. BINARY NUMBER SYSTEM:-

(a) Binary to decimal conversion:-

In this method, each binary digit of the number is multiplied by its positional weight and the product terms are added to obtain decimal number.

For example:

(i) Convert $(10101)_2$ to decimal.

Solution :

(Positional weight) $2^4\ 2^3\ 2^2\ 2^1\ 2^0$
Binary number 10101
$= (1 \times 2^4) + (0 \times 2^3) + (1\times 2^2) + (0 \times 2^1) + (1 \times 2^0)$
$= 16 + 0+ 4+ 0+ 1$
$= (21)_{10}$

(ii) Convert $(111.101)_2$ to decimal.

Solution:

$(111.101)_2 = (1 \times 2^2) + (1 \times 2^1) + (1\times 2^0) + (1 \times 2^{-1}) + (0 \times 2^{-2}) + (1 \times 2^{-3})$
$= 4+ 2+ 1 + 0.5 + 0 + 0.125$
$= (7.625)_{10}$

(b) Binary to Octal conversion:-

For conversion binary to octal the binary numbers are divided into groups of 3 bits each, starting at the binary point and proceeding towards left and right.

Octal	Binary	Octal	Binary
0	000	4	100
1	001	5	101
2	010	6	110
3	011	7	111

For example:

(i) Convert $(101111010110.110110011)_2$ into octal.

Solution :

Group of 3 bits are 101 111 010 110 . 110 110 011

Convert each group into octal = 5 7 2 6 . 6 6 3

The result is **$(5726.663)_8$**

(ii) Convert $(10101111001.0111)_2$ into octal.

Solution :

Binary number 10 101 111 001 . 011 1

Group of 3 bits are = 010 101 111 001 . 011 100

Convert each group into octal = 2 5 7 1 . 3 4

The result is **$(2571.34)_8$**

(c) Binary to Hexadecimal conversion:-

For conversion binary to hexadecimal number the binary numbers starting from the binary point, groups are made of 4 bits each, on either side of the binary point.

Hexadecimal	Binary	Hexadecimal	Binary
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

For example:
(i) Convert $(1011011011)_2$ into hexadecimal.

Solution:

Given Binary number 10 1101 1011

Group of 4 bits are 0010 1101 1011

Convert each group into hex = 2 D B

The result is **$(2DB)_{16}$**

(ii) Convert $(01011111011.011111)_2$ into hexadecimal.

Solution:

Given Binary number 010 1111 1011 **.** 0111 11

Group of 3 bits are = 0010 1111 1011 **.** 0111 1100

Convert each group into octal = 2 F B **.** 7 C

The result is **$(2FB.7C)_{16}$**

2. DECIMAL NUMBER SYSTEM:-

(a)Decimal to binary conversion:-

In the conversion the integer number are converted to the desired base using successive division by the base or radix.

For example:
(i) Convert $(52)_{10}$ into binary.

Solution:

Divide the given decimal number successively by 2 read the integer part remainder upwards to get equivalent binary number. Multiply the fraction part by 2. Keep the integer in the product as it is and multiply the new fraction in the product by 2. The process is continued and the integer are read in the products from top to bottom.

```
2 | 52
2 | 26    — 0
2 | 13    — 0
2 | 6     — 1
2 | 3     — 0
2 | 1     — 1
    0     — 1
```

Result of $(52)_{10}$ is **$(110100)_2$**

(ii) Convert $(105.15)_{10}$ into binary.

Solution:

	Integer part		**Fraction part**
2	105		0.15 x 2 = 0.30
2	52	— 1	0.30 x 2 = 0.60
2	26	— 0	0.60 x 2 = 1.20
2	13	— 0	0.20 x 2 = 0.40
2	6	— 1	0.40 x 2 = 0.80
2	3	— 0	0.80 x 2 = 1.60
2	1	— 1	
	0	— 1	

Result of $(105.15)_{10}$ is **$(1101001.001001)_2$**

(b)Decimal to octal conversion:-

To convert the given decimal integer number to octal, successively divide the given number by 8 till the quotient is 0. To convert the given decimal fractions to octal successively multiply the decimal fraction and the subsequent decimal fractions by 8 till the product is 0 or till the required accuracy is obtained.

For example:
(i) Convert $(378.93)_{10}$ into octal.

Solution:

8	378		0.93 x 8 = 7.44
8	47	— 2	0.44 x 8 = 3.52
8	5	— 7	0.52 x 8 = 4.16
	0	— 5	0.16 x 8 = 1.28

Result of $(378.93)_{10}$ is **$(572.7341)_8$**

(c)Decimal to hexadecimal conversion:-

The decimal to hexadecimal conversion is same as octal.

For example:
(i) Convert $(2598.675)_{10}$ into hexadecimal.

Solution:

		Remainder			
		Decimal	**Hex**		**Hex**
16	2598			0.675 x 16 = 10.8	A
16	162	— 6	6	0.800 x 16 = 12.8	C
16	10	— 2	2	0.800 x 16 = 12.8	C
	0	— 10	A	0.800 x 16 = 12.8	C

Result of $(2598.675)_{10}$ is **$(A26.ACCC)_{16}$**

3. OCTAL NUMBER SYSTEM:-

(a)Octal to binary conversion:-

To convert a given a octal number to binary, replace each octal digit by its 3- bit binary equivalent.

For example:

Convert $(367.52)_8$ into binary.

Solution:

Given Octal number is	3	6	7	.	5	2
Convert each group octal to binary	= 011	110	111	.	101	010

Result of $(367.52)_8$ is **$(011110111.101010)_2$**

(b)Octal to decimal conversion:-

For conversion octal to decimal number, multiply each digit in the octal number by the weight of its position and add all the product terms

For example: -

Convert $(4057.06)_8$ to decimal

Solution:

$$
\begin{aligned}
(4057.06)_8 &= 4 \times 8^3 + 0 \times 8^2 + 5 \times 8^1 + 7 \times 8^0 + 0 \times 8^{-1} + 6 \times 8^{-2} \\
&= 2048 + 0 + 40 + 7 + 0 + 0.0937 \\
&= (2095.\ 0937)_{10}
\end{aligned}
$$

Result is **$(2095.0937)_{10}$**

(c) Octal to hexadecimal conversion:-

For conversion of octal to Hexadecimal, first convert the given octal number to binary and then binary number to hexadecimal.

For example :-

Convert $(756.603)_8$ to hexadecimal.

Solution :-

Given octal no.		7	5	6	.	6	0	3
Convert each octal digit to binary	=	111	101	110	.	110	000	011
Group of 4bits are	=	0001	1110	1110	.	1100	0001	1000
Convert 4 bits group to hex.	=	1	E	E	.	C	1	8

Result is **$(1EE.C18)_{16}$**

(4) HEXADECIMAL NUMBER SYSTEM :-

(a)Hexadecimal to binary conversion:-

For conversion of hexadecimal to binary, replace hexadecimal digit by its 4 bit binary group.

For example:

Convert $(3A9E.B0D)_{16}$ into binary.

Solution:

Given Hexadecimal number is	3	A	9	E	.	B	0	D
Convert each hexadecimal digit to 4 bit binary	= 0011	1010	1001	1110	.	1011	0000	1101

Result of $(3A9E.B0D)_8$ is **$(0011101010011110.101100001101)_2$**

(b)Hexadecimal to decimal conversion:-

For conversion of hexadecimal to decimal, multiply each digit in the hexadecimal number by its position weight and add all those product terms.

For example: -
Convert $(A0F9.0EB)_{16}$ to decimal

Solution:

$$
\begin{aligned}
(A0F9.0EB)_{16} &= (10 \times 16^{3})+(0 \times 16^{2})+(15 \times 16^{1}) +(9 \times 16^{0}) +(0 \times 16^{-1}) +(14 \times 16^{-2}) +(11 \times 16^{-3}) \\
&= 40960 + 0 + 240 + 9 + 0 +0.0546 + 0.0026 \\
&= (41209.0572)_{10}
\end{aligned}
$$

Result is $\mathbf{(41209.0572)_{10}}$

(c) Hexadecimal to Octal conversion:-

For conversion of hexadecimal to octal, first convert the given hexadecimal number to binary and then binary number to octal.

For example :-
Convert $(B9F.AE)_{16}$ to octal.

Solution :-

Given hexadecimal no.is		B	9	F		.	A	E		
Convert each hex. digit to binary	=	1011	1001	1111		.	1010	1110		
Group of 3 bits are	=	101	110	011	111	.	101	011	100	
Convert 3 bits group to octal.	=	5	6	3	7	.	5	3	4	

Result is $\mathbf{(5637.534)_{8}}$

BINARY ARITHEMATIC OPERATION :-

1. BINARY ADDITION:-

The binary addition rules are as follows

0 + 0 = 0 ; 0 + 1 = 1 ; 1 + 0 = 1 ; 1 + 1 = 10 , i.e 0 with a carry of 1

For example :-

Add $(100101)_2$ and $(1101111)_2$.
Solution :-

```
      1 0 0 1 0 1
+   1 1 0 1 1 1 1
  ---------------
  1 0 0 1 0 1 0 0
```

Result is $\mathbf{(10010100)_{2}}$

2. BINARY SUBTRACTION:-

The binary subtraction rules are as follows

0 - 0 = 0 ; 1 - 1 = 0 ; 1 - 0 = 1 ; 0 - 1 = 1 , with a borrow of 1

For example :-

Substract (111.111)$_2$ from (1010.01)$_2$.

Solution :-

```
        1 0 1 0 . 0 1 0
  -       1 1 1 . 1 1 1
        0 0 1 0 . 0 1 1
```

Result is **(0010.011)$_2$**

3. BINARY MULTIPLICATION:-

The binary multiplication rules are as follows

0 x 0 = 0 ; 1 x 1 = 1 ; 1 x 0 = 0 ; 0 x 1 = 0

For example :-

Multiply (1101)$_2$ by (110)$_2$.

Solution :-

```
          1 1 0 1
    x       1 1 0
            0 0 0 0
          1 1 0 1
  +     1 1 0 1
      1 0 0 1 1 1 0
```

Result is **(1001110)$_2$**

4. BINARY DIVISION:-

The binary division is very simple and similar to decimal number system. The division by ‘0’ is meaningless. So we have only 2 rules

0 ÷ 1 = 0

1 ÷ 1 = 1

For example :-

Divide (10110)$_2$ by (110)$_2$.

Solution :-

```
110 ) 101101 ( 111.1
   -  110
       1010
        110
         1001
          110
          110
          110
          000
```

Result is **(111.1)$_2$**

1's COMPLEMENT REPRESENTATION :-

The 1's complement of a binary number is obtained by changing each 0 to 1 and each 1 to 0.

For example :-

Find $(1100)_2$ 1's complement.

Solution :-

Given	1	1	0	0
1's complement is	0	0	1	1

Result is **$(0011)_2$**

2's COMPLEMENT REPRESENTATION :-

The 2's complement of a binary number is a binary number which is obtained by adding 1 to the 1's complement of a number i.e.

2's complement = 1's complement + 1

For example :-

Find $(1010)_2$ 2's complement.

Solution :-

Given	1	0	1	0
1's complement is	0	1	0	1
+				1
2's complement	0	1	1	0

Result is **$(0110)_2$**

SIGNED NUMBER :-

In sign – magnitude form, additional bit called the sign bit is placed in front of the number. If the sign bit is 0, the number is positive. If it is a 1, the number is negative.

For example:-

0 1 0 1 0 0 1 = +41

↑

Sign bit

1 1 0 1 0 0 1 = -41

↑

Sign bit

SUBSTRACTION USING COMPLEMENT METHOD :-

1's COMPLEMENT:-

In 1's complement subtraction, add the 1's complement of subtrahend to the minuend. If there is a carry out, then the carry is added to the LSB. This is called end around carry. If the MSB is 0, the result is positive. If the MSB is 1, the result is negative and is in its 1's complement form. Then take its 1's complement to get the magnitude in binary.

For example:-

Subtract $(10000)_2$ from $(11010)_2$ using 1's complement.

Solution:-

```
   1 1 0 1 0                 1 1 0 1 0                       =   26
-  1 0 0 0 0      =>       + 0 1 1 1 1  (1's complement)     = - 16
                Carry  →   1 0 1 0 0 1                         + 10
                           +         1
                             0 1 0 1 0   = +10
```

Result is **+10**

2's COMPLEMENT:-

In 2's complement subtraction, add the 2's complement of subtrahend to the minuend. If there is a carry out, ignore it. If the MSB is 0, the result is positive. If the MSB is 1, the result is negative and is in its 2's complement form. Then take its 2's complement to get the magnitude in binary.

For example:-

Subtract $(1010100)_2$ from $(1010100)_2$ using 2's complement.

Solution:-

```
   1 0 1 0 1 0 0                1 0 1 0 1 0 0                        =    84
-  1 0 1 0 1 0 0      =>      + 0 1 0 1 1 0 0  (2's complement)      = -  84
                              1 0 0 0 0 0 0 0 ( Ignore the carry)          0
                         =                  0 (result = 0)
```

Hence MSB is 0. The answer is positive. So it is +0000000 = **0**

DIGITAL CODES:-

In practice the digital electronics requires to handle data which may be numeric, alphabets and special characters. This requires the conversion of the incoming data into binary format before it can be processed. There is various possible ways of doing this and this process is called encoding. To achieve the reverse of it, we use decoders.

WEIGHTED AND NON-WEIGHTED CODES:-

There are two types of binary codes

1) Weighted binary codes
2) Non- weighted binary codes

In weighted codes, for each position (or bit) ,there is specific weight attached.

For example, in binary number, each bit is assigned particular weight 2n where 'n' is the bit number for n = 0,1,2,3,4 the weights are 1,2,4,8,16 respectively.

Example :- BCD

Non-weighted codes are codes which are not assigned with any weight to each digit position, i.e., each digit position within the number is not assigned fixed value.

Example:- Excess – 3 (XS -3) code and Gray codes

BINARY CODED DECIMAL (BCD):-

BCD is a weighted code. In weighted codes, each successive digit from right to left represents weights equal to some specified value and to get the equivalent decimal number add the products of the weights by the corresponding binary digit. 8421 is the most common because 8421 BCD is the most natural amongst the other possible codes.

For example:-

$(567)_{10}$ is encoded in various 4 bit codes.

Solution:-

Decimal	→	5	6	7
8421 code	→	0101	0110	0111
6311 code	→	0111	1000	1001
5421 code	→	1000	0100	1010

BCD ADDITION:-

Addition of BCD (8421) is performed by adding two digits of binary, starting from least significant digit. In case if the result is an illegal code (greater than 9) or if there is a carry out of one then add 0110(6) and add the resulting carry to the next most significant.

For example:-

Add 679.6 from 536.8 using BCD addition.

Solution:-

```
   6 7 9 . 6          0110  0111  1001 . 0110     ( 679.6 in BCD)
+  5 3 6 . 8     =>+ 0101  0011  0110 . 1000      (536.8 in BCD)
 1 2 1 6 . 4          1011  1010  1111 . 1110     ( All are illegal codes)
                    + 0110 +0110 +0110 .+0110     ( Add 0110 to each)
              0001  0010  0001  0110 . 0100
                1     2     1     6  .   4        ( corrected sum = 1216.4)
```

Result is **1216.4**

BCD SUBTRACTION:-

The BCD subtraction is performed by subtracting the digits of each 4 – bit group of the subtrahend from corresponding 4 – bit group of the minuend in the binary starting from the LSD. If there is no borrow from the next higher group[then no correction is required. If there is a borrow from the next group, then 6_{10} (0110) is subtracted from the difference term of this group.

For example:-

Subtract 147.8 from 206.7 using 8421 BCD code.

Solution:-

```
   2 0 6 . 7          0010  0000  0110 . 0111     ( 206.7 in BCD)
-  1 4 7 . 8     =>- 0001  0100  0111 . 1000      (147.8 in BCD)
     5 8 . 9          0000  1011  1110 . 1111     ( Borrows are present)
                          - 0110 -0110 .- 0110
                            0101  1000 . 1001
                              5     8  .   9      ( corrected difference = 58.9)
```

Result is **$(58.9)_{10}$**

EXCESS THREE(XS-3) CODE:-

The Excess-3 code, also called XS-3, is a non- weighted BCD code. This derives it name from the fact that each binary code word is the corresponding 8421 code word plus 0011(3). It is a sequential code. It is a self complementing code.

XS-3 ADDITION:-

In XS-3 addition, add the XS-3 numbers by adding the 4 bit groups in each column starting from the LSD. If there is no carry out from the addition of any of the 4 bit groups, subtract 0011 from the sum term of those groups. If there is a carry out, add 0011 to the sum term of those groups

For example:-

Add 37 and 28 using XS-3 code.

Solution:-

3 7   0110 1010   (37 in XS-3)

+ 2 8   =>   + 0101 1011   (28 in XS-3)

6 5   1011 **1**1010   (Carry is generated)

+ 1   (Propagate carry)

1100 0101   (Add 0110 to correct 0101 and

- 0011 +0011   subtract 0011 to correct 1100)

1001 1000   (Corrected sum in XS-3 = 65_{10})

XS-3 SUBTRACTION:-

To subtract in XS-3 number by subtracting each 4-bit group of the subtrahend from the corresponding 4-bit group of the minuend starting from the LSD. If there is no borrow from the next 4-bit group. add 0011 to the difference term of such groups. If there is a borrow, subtract 0011 from the difference term.

For example :-

. **Subtract 175 from 267 using XS-3 code**.

Solution :-`

267   0101 1010 1010   (267 in XS-3)

-175   =>   - 0100 1010 1000   (175 in XS-3)

092   0000 1111 0010   (Correct 0010 and 0000 by adding 0011 and

+0011 -0011 +0011   correct 1111 by subtracting 0011)

0011 1100 0101   (Corrected difference in XS-3 = 92_{10})

ASCII CODE:-

The American Standard Code for Information Interchange (ASCII) pronounced as 'ASKEE' is widely used alphanumeric code. This is basically a 7 bit code. The number of different bit patterns that can be created with 7 bits is 27 = 128 , the ASCII can be used to encode both the uppercase and lowercase characters of the alphabet (52 symbols) and some special symbols in addition to the 10 decimal digits. It is used extensively for printers and terminals that interface with small computer systems. The table shown below shows the ASCII groups.

The ASCII code

LSBs	**MSBs**							
	000	001	010	011	100	101	110	111
0000	NUL	DEL	Space	0	@	P	P	
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	“	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s

0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	‘	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DLE

EBCDIC CODE:-

The Extended Binary Coded Decimal Interchange Code (EBCDIC) pronounced as ‘eb – si- dik’ is an 8 bit alphanumeric code. Since 28 = 256 bit patterns can be formed with 8 bits. It is used by most large computers to communicate in alphanumeric data. The table shown below shows the EBCDIC code.

The EBCDIC code

LSD (Hex)	**MSD(Hex)**															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DLE	DS		SP	&							[	]	\	0
1	SOH	DC1	SOS				/		a	j	~		A	J		1
2	STX	DC2	FS	SYN					b	k	s		B	K	S	2
3	ETX	DC3							c	l	t		C	L	T	3
4	PF	RES	BYP	PN					d	m	u		D	M	U	4
5	HT	NL	LF	RS					e	n	v		E	N	V	5
6	LC	BS	EOB	YC					f	o	w		F	O	W	6
7	DEL	IL	PRE	EOT					g	p	x		G	P	X	7
8		CAN							h	q	y		H	Q	Y	8
9		EM							i	r	z		I	R	Z	9
A	SMM	CC	SM		Ø	!	I	:								
B	VT				.	$	,	#								
C	FF	IFS		DC4	<	*	%	@								
D	CR	IGS	ENQ	NAK	(	)	_	‘								
E	SO	IRS	ACK		+	;	>	=								
F	SI	IUS	BEL	SUB	I	‘	?	‘								

GRAY CODE:-

The gray code is a non-weighted code. It is not a BCD code. It is cyclic code because successive words in this differ in one bit position only i.e it is a unit distance code.

Gray code is used in instrumentation and data acquisition systems where linear or angular displacement is measured. They are also used in shaft encoders, I/O devices, A/D converters and other peripheral equipment.

BINARY- TO – GRAY CONVERSION:-

If an n-bit binary number is represented by B_n B_{n-1} ----- B_1 and its gray code equivalent by G_n G_{n-1} ----- G_1, where B_n and G_n are the MSBs , then gray code bits are obtained from the binary code as follows

$G_n = B_n$

$G_{n-1} = B_n \oplus B_{n-1}$

.

.

.

.

$G_1 = B_2 \oplus B_1$

Where the symbol $\oplus$ stands for Exclusive OR (X-OR)

For example :-

Convert the binary 1001 to the Gray code.

Solution :-`

$$\begin{tikzcd}
\text{Binary} \arrow[r] & \mathbf{1} \arrow[r, dash] \arrow[d] & \oplus \arrow[r] & \mathbf{0} \arrow[r, dash] \arrow[d] & \oplus \arrow[r] & \mathbf{0} \arrow[r, dash] \arrow[d] & \oplus \arrow[r] & \mathbf{1} \arrow[d] \\
\text{Gray} \arrow[r] & \mathbf{1} & & \mathbf{1} & & \mathbf{0} & & \mathbf{1}
\end{tikzcd}$$

The gray code is **1101**

GRAY- TO - BINARY CONVERSION:-

If an n-bit gray number is represented by G_n G_{n-1} ------- G_1 and its binary equivalent by B_n B_{n-1} ----- B_1, then binary bits are obtained from Gray bits as follows :

$B_n = G_n$

$B_{n-1} = B_n \oplus G_{n-1}$

.

.

.

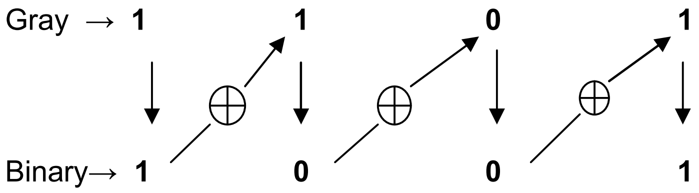
.

$B_1 = B_2 \oplus G_1$

For example :-

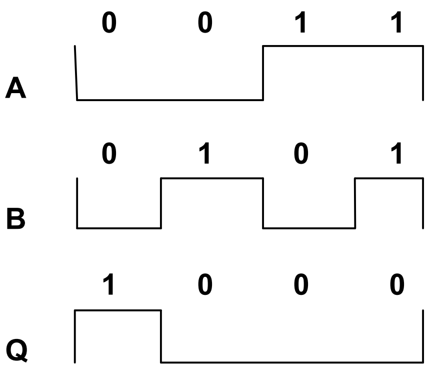
Convert the Gray code 1101 to the binary.

Solution :-



The binary code is **1001**

Timing Diagram

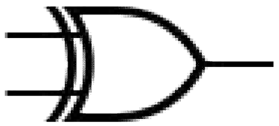


EXCLUSIVE – OR (X-OR) GATE:-

- An X-OR gate is a two input, one output logic circuit.
- The output is logic 1 when one and only one of its two inputs is logic 1. When both the inputs is logic 0 or when both the inputs is logic 1, the output is logic 0.

IC No.:- 7486

Logic Symbol



Truth Table

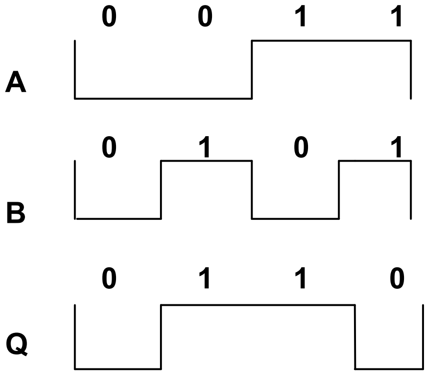
INPUT		OUTPUT
A	**B**	**Q = A $\oplus$ B**
0	0	0
0	1	1
1	0	1
1	1	0

INPUTS are **A** and **B**

OUTPUT is **Q** = A $\oplus$ B

= $A\bar{B} + A\bar{B}$

Timing Diagram

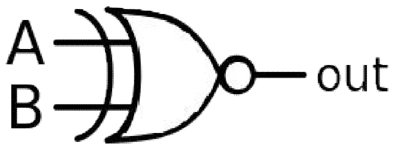


EXCLUSIVE – NOR (X-NOR) GATE:-

- An X-NOR gate is the combination of an X-OR gate and a NOT gate.
- An X-NOR gate is a two input, one output logic circuit.
- The output is logic 1 only when both the inputs are logic 0 or when both the inputs is 1.
- The output is logic 0 when one of the inputs is logic 0 and other is 1.

IC No.:- 74266

Logic Symbol

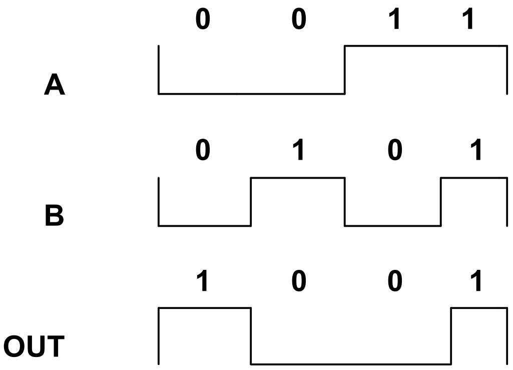


INPUT		OUTPUT
A	B	OUT =A XNOR B
0	0	1
0	1	0
1	0	0
1	1	1

OUT =A B + $\overline{A}\ \overline{B}$

= A XNOR B

Timing Diagram

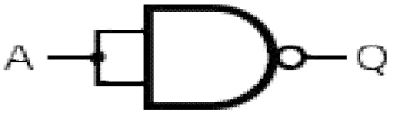


UNIVERSAL GATES:-

There are 3 basic gates AND, OR and NOT, there are two universal gates NAND and NOR, each of which can realize logic circuits single handedly. The NAND and NOR gates are called universal building blocks. Both NAND and NOR gates can perform all logic functions i.e. AND, OR, NOT, EXOR and EXNOR.

NAND GATE:-

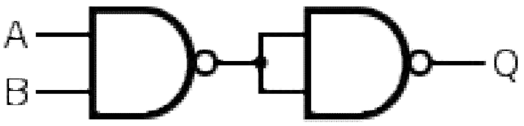
a) **Inverter from NAND gate**



Input = **A**
Output **Q =** $\overline{A}$

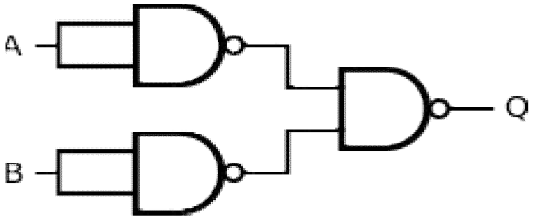
b) **AND gate from NAND gate**

Input s are **A** and **B**
Output **Q = A.B**



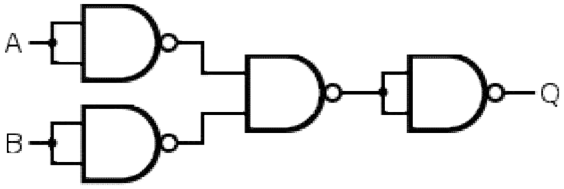
c) **OR gate from NAND gate**

Inputs are **A** and **B**
Output **Q = A+B**



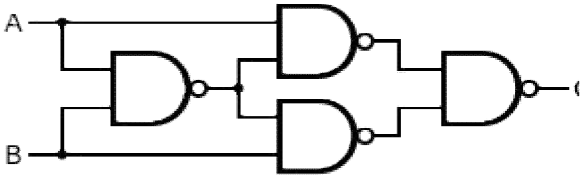
d) **NOR gate from NAND gate**

Inputs are **A** and **B**
Output **Q = $\overline{A+B}$**



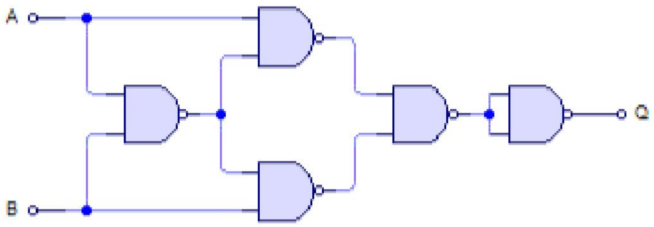
e) **EX-OR gate from NAND gate**

Inputs are **A** and **B**
Output **Q = $A\overline{B} + \overline{A}B$**



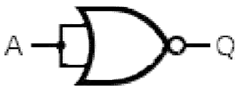
f) **EX-NOR gate From NAND gate**

Inputs are **A** and **B**
Output **Q = $AB + \overline{A}\,\overline{B}$**

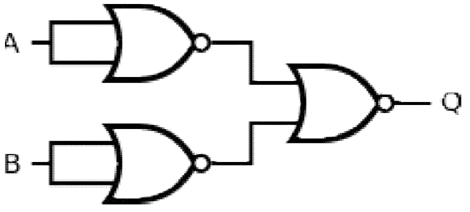


NOR GATE:-

a) **Inverter from NOR gate**
Input = **A**
Output **Q = $\overline{A}$**

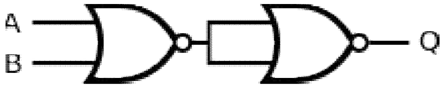


b) **AND gate from NOR gate**
Input s are **A** and **B**
Output **Q = A.B**



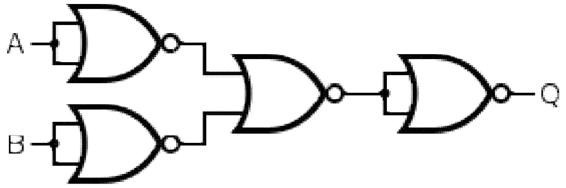
c) OR gate from NOR gate

Inputs are **A** and **B**
Output **Q = A+B**



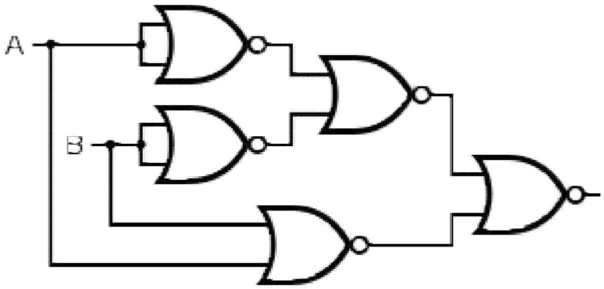
d) NAND gate from NOR gate

Inputs are **A** and **B**
Output **Q =** $\overline{\mathbf{A.B}}$



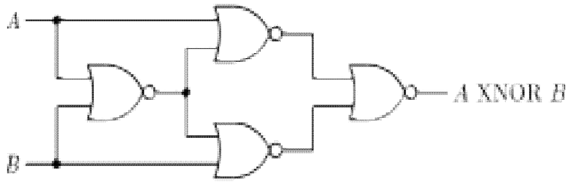
e) EX-OR gate from NOR gate

Inputs are **A** and **B**
Output **Q =** $\mathbf{A\,\overline{B} + \overline{A}B}$



f) EX-NOR gate From NOR gate

Inputs are **A** and **B**
Output **Q =** $\mathbf{A\,B + \overline{A}\,\overline{B}}$



THRESHOLD LOGIC:-

INTRODUCTION:-

- The threshold element, also called the threshold gate (T-gate) is a much more powerful device than any of the conventional logic gates such as NAND, NOR and others.
- Complex, large Boolean functions can be realized using much fewer threshold gates.
- Frequently a single threshold gate can realize a very complex function which otherwise might require a large number of conventional gates.
- T-gate offers incomparably economical realization; it has not found extensive use with the digital system designers mainly because of the following limitations.
  1. It is very sensitive to parameter variations.
  2. It is difficult to fabricate it in IC form.

3. The speed of switching of threshold elements in much lower than that of conventional gates.

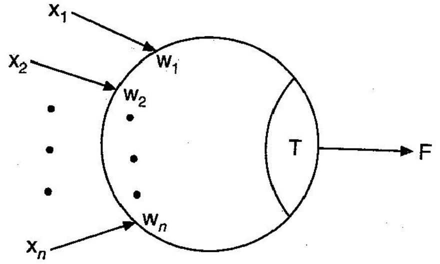
THE THRESHOLD ELEMENTS:-

- A threshold element or gate has 'n' binary inputs $x_1, x_2, \ldots.., x_n$; and a single binary output F. But in addition to those, it has two more parameters.
- Its parameters are a threshold T and weights $w_1, w_2, \ldots.,w_n$. The weights $w_1, w_2, \ldots, w_n$ are associated with the input variables $x_1, x_2, \ldots, x_n$.
- The value of the threshold (T) and weights may be real, positive or negative number.
- The symbol of the threshold element is shown in fig.(a).
- It is represented by a circle partitioned into two parts, one part represents the weights and other represents T.
- It is defined as

  $$F(x_1, x_2, \ldots\ldots, x_n) = 1 \text{ if and only if } \sum_{i=1}^{n} w_i x_i \geq T$$

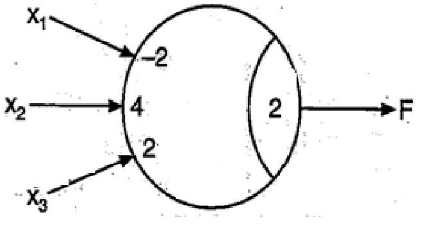
  otherwise
  $F(x_1, x_2, \ldots\ldots, x_n) = 0$

- The sum and product operation are normal arithmetic operations and the sum $\sum_{i=1}^{n} w_i x_i \geq T$ is called the weighted sum of the element or gate.



Example:-

Obtain the minimal Boolean expression from the threshold gate shown in figure.



Solution:-

The threshold gate with three inputs x_1, x_2, x_3 with weights -2(w_1) , 4(w_2) and 2(w_3) respectively. The value of threshold is 2(T). The table shown is the weighted sums and outputs for all input combinations. For this threshold gate, the weighted sum is

$$w = w_1x_1 + w_2x_2 + w_3x_3$$

$$= (-2)x_1 + (4)x_2 + (2)x_3$$

$$= -2x_1 + 4x_2 + 2x_3$$

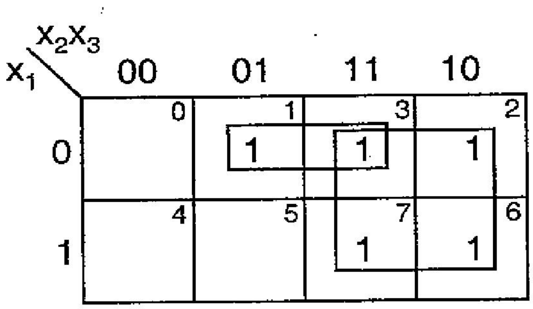
The output F is logic 1 for w≥2 and it is logic 0 for w<2

Input Variables			**Weighted Sum**	**Output**
x_1	**x_2**	**x_3**	**$w = -2x_1 + 4x_2 + 2x_3$**	**F**
0	0	0	0	0
0	0	1	2	1
0	1	0	4	1
0	1	1	6	1
1	0	0	-2	0
1	0	1	0	0
1	1	0	2	1
1	1	1	4	1

From the input – output relation is given in the table, the Boolean expression for the output is

$$F=\sum m\ (1, 2, 3, 6, 7)$$

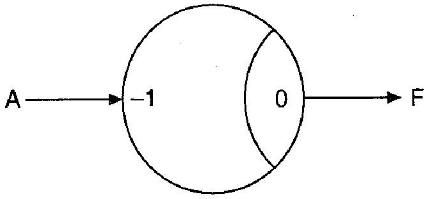
The K-map for F is



$$F_{min} = \bar{x}_1x_3 + x_2$$

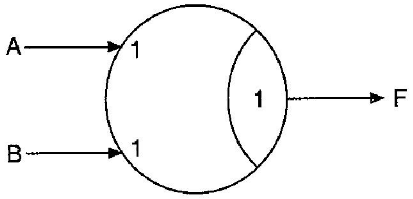
UNIVERSALITY OF A T-GATE:-

- A single T-gate can realize a large number of functions by merely changing either the weights or the threshold or both, which can be done by altering the value of the corresponding resistors.
- Since a threshold gate can realize universal gates, i.e., NAND gates and NOR gates, a threshold gate is also a universal gate.
- Single threshold gate cannot realize by a single T-gate
- Realization of logic gates using T-gates is shown in the below figure.



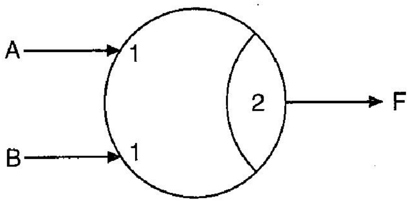
(a) NOT gate $F = \overline{A}$

Input	Weighted sum	Output
A	W = −A	F
0	0	1
1	−1	0



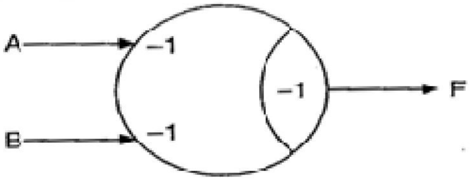
(b) OR gate F = A + B

Inputs		Weighted sum	Output
A	B	w = A + B	F
0	0	0	0
0	1	1	1
1	0	1	1
1	1	2	1



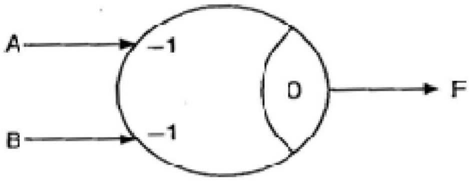
(c) AND gate F = AB

Inputs		Weighted sum	Output
A	B	w = A + B	F
0	0	0	0
0	1	1	0
1	0	1	0
1	1	2	1



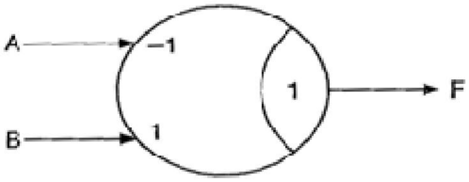
NAND gate
(d) $F = \overline{A} + \overline{B} = \overline{AB}$

Inputs		Weighted sum	Output
A	B	w = −A − B	F
0	0	0	1
0	1	−1	1
1	0	−1	1
1	1	−2	0



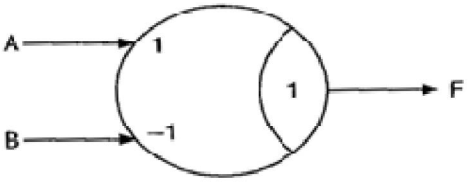
NOR gate
(e) $F = \overline{A} \cdot \overline{B} = \overline{A + B}$

Inputs		Weighted sum	Output
A	B	w = −A − B	F
0	0	0	1
0	1	−1	0
1	0	−1	0
1	1	−2	0



(f) $F = \overline{A}B$

Inputs		Weighted sum	Output
A	B	w = −A + B	F
0	0	0	0
0	1	1	1
1	0	−1	0
1	1	0	0



(g) $F = A\overline{B}$

Inputs		Weighted sum	Output
A	B	w = A − B	F
0	0	0	0
0	1	−1	0
1	0	1	1
1	1	0	0

BOOLEAN ALGEBRA

INTRODUCTION:-

- Switching circuits are also called logic circuits, gates circuits and digital circuits.
- Switching algebra is also called Boolean algebra.
- Boolean algebra is a system of mathematical logic. It is an algebraic system consisting of the set of elements (0,1), two binary operators called OR and AND and unary operator called NOT.
- It is the basic mathematical tool in the analysis and synthesis of switching circuits.
- It is a way to express logic functions algebraically.
- Any complex logic can be expressed by a Boolean function.
- The Boolean algebra is governed by certain well developed rules and laws.

AXIOMS AND LAWS OF BOOLEAN ALGEBRA:-

Axioms or postulates of Boolean algebra are set of logical expressions that are accepted without proof and upon which we can build a set of useful theorems. Actually, axioms are nothing more than the definitions of the three basic logic operations AND, OR and INVERTER. Each axiom can be interpreted as the outcome of an operation performed by a logic gate.

AND operation	OR operation	NOT operation
Axiom 1: $0 . 0 = 0$	Axiom 5: $0 + 0 = 0$	Axiom 9: $\overline{1} = 0$
Axiom 2: $0 . 1 = 0$	Axiom 6: $0 + 1 = 1$	Axiom 10:$\overline{0} = 1$
Axiom 3: $1 . 0 = 0$	Axiom 7: $1 + 0 = 1$	
Axiom 2: $1 . 1 = 1$	Axiom 8: $1 + 1 = 1$	

1. Complementation Laws:-

The term complement simply means to invert, i.e. to changes 0s to 1s and 1s to 0s. The five laws of complementation are as follows:

Law 1: $\overline{0} = 1$

Law 2: $\overline{1} = 0$

Law 3: if A = 0, then $\overline{A} = 1$

Law 4: if A = 1,then$\overline{A} = 0$

Law 5: $\overline{\overline{A}} = 0$ (double complementation law)

2. OR Laws:-

The four OR laws are as follows

Law 1: A + 0 = 0(Null law)

Law 2: A + 1 = 1(Identity law)

Law 3: A + A = A

Law 4: A +$\overline{A}$ = 1

3. AND Laws:-

The four AND laws are as follows

Law 1: A . 0 = 0(Null law)

Law 2: A . 1 = 1(Identity law)

Law 3: A . A = A

Law 4: A .$\overline{A}$ = 0

4. Commutative Laws:-

Commutative laws allow change in position of AND or OR variables. There are two commutative laws.

Law 1: A + B = B + A

Proof

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

=

B	A	B+ A
0	0	0
0	1	1
1	0	1
1	1	1

Law 2: A . B = B . A

Proof

A	B	A . B
0	0	0
0	1	0
1	0	0
1	1	1

=

B	A	B. A
0	0	0
0	1	0
1	0	0
1	1	1

This law can be extended to any number of variables. For example
A.B. C = B. C. A = C. A. B = B. A. C

5. Associative Laws:-

The associative laws allow grouping of variables. There are 2 associative laws.

Law 1: (A + B) + C = A + (B + C)

Proof

A	B	C	A+B	(A+B)+C
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

=

A	B	C	B+C	A+(B+C)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Law 2: (A .B) C = A (B .C)

Proof

A	**B**	**C**	**AB**	**(AB)C**
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	1

=

A	**B**	**C**	**B.C**	**A(B.C)**
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

This law can be extended to any number of variables. For example

A(BCD) = (ABC)D = (AB) (CD)

6. Distributive Laws:-

The distributive laws allow factoring or multiplying out of expressions. There are two distributive laws.

Law 1: A (B + C) = AB + AC

Proof

A	**B**	**C**	**B+C**	**A(B+C)**
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

=

A	**B**	**C**	**AB**	**AC**	**A+(B+C)**
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

Law 2: A + BC = (A+B) (A+C)

Proof **RHS** = (A+B) (A+C)

= AA + AC + BA + BC

= A + AC + AB + BC

= A (1+ C + B) + BC

= A. 1 + BC (1 +C + B = 1 + B = 1)

= A + BC

= **LHS**

7. Redundant Literal Rule (RLR):-

Law 1: A + $\bar{A}$B = A + B

Proof

$$\begin{aligned} A + \overline{A}B &= (A + \overline{A})\,(A + B) \\ &= 1.\,(A + B) \\ &= A + B \end{aligned}$$

Law 2: $A(\overline{A} + B) = AB$

Proof

$$\begin{aligned} A(\overline{A} + B) &= A\overline{A} + AB \\ &= 0 + AB \\ &= AB \end{aligned}$$

8. Idempotence Laws:-

Idempotence means same value.

Law 1: A**.** A = A

Proof

If A = 0, then A**.** A = 0**.** 0 =0 = A

If A = 1, then A**.** A = 1**.** 1 = 1 = A

This law states that AND of a variable with itself is equal to that variable only.

Law 2: A + A = A

Proof

If A = 0, then A + A = 0 + 0 = 0 = A

If A = 1, then A + A = 1 + 1 = 1 = A

This law states that OR of a variable with itself is equal to that variable only.

9. Absorption Laws:-

There are two laws:

Law 1: A + A · B = A

Proof

A + A · B = A (1 + B) = A · 1 = A

A	**B**	**AB**	**A+AB**
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Law 2: A (A + B) = A

Proof

A (A + B) = A · A + A · B = A + AB = A(1 + B) = A · 1 = A

A	**B**	**A+B**	**A(A+B)**
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

10. Consensus Theorem (Included Factor Theorem):-

Theorem 1:

$AB + \bar{A}C + BC = AB + \bar{A}C$

Proof

LHS = $AB + \bar{A}C + BC$

$= AB + \bar{A}C + BC\,(A+\bar{A})$

$= AB + \bar{A}C + BCA + BC\bar{A}$

$= AB\,(1 + C) + \bar{A}C\,(1+ B)$

$=AB\,(1) + \bar{A}C\,(1)$

$= AB + \bar{A}C$

= **RHS**

Theorem 2:

$(A + B)(\bar{A} + C)(B + C) =(A + B)(\bar{A} + C)$

Proof

LHS = $(A + B)\,(\bar{A} + C)\,(B + C)$

$= (A\bar{A} + AC + B\bar{A} + BC)\,(B + C)$

$= (AC + BC + \bar{A}B)\,(B + C)$

$= ABC + BC + \bar{A}B + AC + BC+\bar{A}BC$

$= AC + BC + \bar{A}B$

RHS= $(A + B)\,(\bar{A}+C)$

$= A\bar{A} + AC + BC + \bar{A}B$

$= AC + BC + \bar{A}B$

= LHS

11. Transposition Theorem:-

Theorem:

$AB + \bar{A}C = (A + C)(\bar{A} + B)$

Proof

RHS= $(A + C)\,(\bar{A} + B)$

$= A\bar{A} + C\bar{A} + AB + CB$

$= 0 + \bar{A}C + AB + BC$

$= \bar{A}C + AB + BC\,(A+\bar{A})$

$= AB + ABC + \bar{A}C + \bar{A}BC$

$= AB + \bar{A}C$

= LHS

12. De Morgan's Theorem:-

De Morgan's theorem represents two laws in Boolean algebra.

Law 1: $\overline{A + B} = \bar{A}\cdot \bar{B}$

Proof

A	B	A + B	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

=

A	B	$\bar{A}$	$\bar{B}$	$\bar{A}\ \bar{B}$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

This law states that the complement of a sum of variables is equal to the product of their individual complements.

Law 2: $\overline{A\cdot B} = \overline{A} + \overline{B}$

Proof

A	B	A . B	$\overline{A . B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

=

A	B	$\overline{A}$	$\overline{B}$	$\overline{A} + \overline{B}$
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

This law states that the complement of a product of variables is equal to the sum of their individual complements.

DUALITY:-

The implication of the duality concept is that once a theorem or statement is proved, the dual also thus stand proved. This is called the principle of duality.

$$[f\ (A, B, C,\ldots\ldots,0, 1, +, \cdot)]_d = f(A, B, C, \ldots., 1, 0, \cdot, +)$$

Relations between complement and dual

$$f_c\ (A, B, C, \ldots\ldots) = \overline{f\ (A, B, C, \ldots\ldots)} = f_d\,(\overline{A}, \overline{B}, \overline{C},\ldots)$$

$$f_d\ (A, B, C, \ldots\ldots) = \overline{f\ (\overline{A}, \overline{B}, \overline{C},\ldots)} = f_c\,(\overline{A}, \overline{B}, \overline{C}, \ldots\ldots)$$

The first relation states that the complement of a function f(A, B, C, …) can be obtained by complementing all the variables in the dual function f_d (A, B, C, …..).

The second relation states that the dual can be obtained by complementing all the literals in $\overline{f\ (A, B, C, \ldots.)}$.

DUALS:-

	Given expression	***Dual***
1.	$\overline{0} = 1$	$\overline{1} = 0$
2.	$0 \cdot 1 = 0$	$1 + 0 = 1$
3.	$0 \cdot 0 = 0$	$1 + 1 = 1$
4.	$1 \cdot 1 = 1$	$0 + 0 = 0$
5.	$A \cdot 0 = 0$	$A + 1 = 1$
6.	$A \cdot 1 = A$	$A + 0 = A$
7.	$A \cdot A = A$	$A + A = A$
8.	$A \cdot \overline{A} = 0$	$A + \overline{A} = 1$
9.	$A \cdot B = B \cdot A$	$A + B = B + A$
10.	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	$A + (B + C) = (A + B) + C$
11.	$A \cdot (B + C) = AB + AC$	$A + BC = (A + B)(A + C)$
12.	$A(A + B) = A$	$A + AB = A$
13.	$A \cdot (A \cdot B) = A \cdot B$	$A + A + B = A + B$
14.	$\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}\,\overline{B}$
15.	$(A + B)(\overline{A} + C)(B + C) = (A + B)(\overline{A} + C)$	$AB + \overline{A}C + BC = AB + \overline{A}C$
16.	$A + \overline{B}C = (A + \overline{B})(A + C)$	$A(\overline{B} + C) = A\overline{B} + AC$
17.	$(A+C)(\overline{A}+B) = AB+\overline{A}C$	$AC+\overline{A}B=(A+B)(\overline{A}+C)$
18.	$(A+B)(C+D) = AC + AD + BC + BD$	$(AB+CD) = (A+C)(A+D)(B+C)(B+D)$
19.	$A + B = AB + \overline{A}B + A\overline{B}$	$AB = (A+B)(\overline{A}+B)(A+\overline{B})$
20.	$\overline{AB} + \overline{A} + AB = 0$	$\overline{A + B} \cdot \overline{A} \cdot (A + B) = 1$

SUM - OF - PRODUCTS FORM:-

- This is also called disjunctive Canonical Form (DCF) or Expanded Sum of Products Form or Canonical Sum of Products Form.
- In this form, the function is the sum of a number of products terms where each product term contains all variables of the function either in complemented or uncomplemented form.
- This can also be derived from the truth table by finding the sum of all the terms that corresponds to those combinations for which ‘f ’ assumes the value 1.

  For example

  $f(A, B, C) = \overline{A}B + \overline{B}C$

  $= \overline{A}B\ (C + \overline{C}) + \overline{B}C\ (A + \overline{A})$

  $= \overline{A}\ \overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}C$
- The product term which contains all the variables of the functions either in complemented or uncomplemented form is called a minterm.
- The minterm is denoted as mo, m1, m2 … .
- An ‘n’ variable function can have 2n minterms.
- Another way of representing the function in canonical SOP form is the showing the sum of minterms for which the function equals to 1.

  For example

  $f\ (A, B, C) = m_1 + m_2 + m_3 + m_5$

  or

  $f\ (A, B, C) = \sum m\ (1, 2, 3, 5)$

  where ∑m represents the sum of all the minterms whose decimal codes are given the parenthesis.

PRODUCT- OF - SUMS FORM:-

- This form is also called as Conjunctive Canonical Form (CCF) or Expanded Product - of – Sums Form or Canonical Product Of Sums Form.
- This is by considering the combinations for which f = 0
- Each term is a sum of all the variables.
- The function $f\ (A, B, C) = (A + B + C\cdot C) + (A + B + C\cdot\overline{C})$

  $= (\overline{A} + \overline{B} + C)\ (\overline{A} + \overline{B} + \overline{C})\ (A + B + C)\ (A + B + \overline{C})$
- The sum term which contains each of the ‘n’ variables in either complemented or uncomplemented form is called a maxterm.
- Maxterm is represented as M_0, M_1, M_2, …….

  Thus CCF of ‘f’ may be written as

  $f(A, B, C)= M_0 \cdot M_4 \cdot M_6 \cdot M_7$

  or

  $f(A, B, C) = (0, 4, 6, 7)$

  Where represented the product of all maxterms.

CONVERSION BETWEEN CANONICAL FORM:-

The complement of a function expressed as the sum of minterms equals the sum of minterms missing from the original function.

Example:-

$f(A, B, C) = \sum m(0,2,4,6,7)$

This has a complement that can be expressed as

$f\ (\overline{A, B, C}) = \sum m(1, 3, 5) = m_1 + m_3 + m_5$

If we complement f by De- Morgan’s theorem we obtain ‘f’ in a form.

$f =(\overline{m_1+ m_3 + m_5}) = \overline{m}_1.\ \overline{m}_3.\ \overline{m}_5$

$= M_1 \, M_3 \, M_5 = \prod M(1, 3, 5)$

Example:-

Expand $A(\bar{A} + B)(\bar{A} + B + \bar{C})$ to maxterms and minterms.

<u>Solution</u>:-

In POS form

$A(\bar{A} + B)(\bar{A} + B + \bar{C})$

$A = A + B\bar{B} + C\bar{C}$

$= (A + B)(A + \bar{B}) + C\cdot\bar{C}$

$= (A + B + C\bar{C})(A + B + C\bar{C})$

$= (A + B + C)(A + B + \bar{C})(A + \bar{B} + C)(A + \bar{B} + \bar{C})$

$A + B = A + B + C\cdot\bar{C}$

$= (\bar{A} + B + C)(\bar{A} + B + \bar{C})$

Therefore

$A(\bar{A} + B)(\bar{A} + B + \bar{C})$

$= (A + B + C)(A + B + \bar{C})(A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + B + C)(\bar{A} + B + \bar{C})$

$= (000)(001)(010)(011)(100)(101)$

$= M_0 \cdot M_1 \cdot M_2 \cdot M_3 \cdot M_4 \cdot M_5$

$= \prod M(0, 1, 2, 3, 4, 5)$

The maxterms M_6 and M_7 are missing in the POS form.

So, the SOP form will contain the minterms 6 and 7

<u>KARNAUGH MAP</u> <u>OR</u> <u>K- MAP</u>:-

- The K- map is a chart or a graph, composed of an arrangement of adjacent cells, each representing a particular combination of variables in sum or product form.
- The K- map is systematic method of simplifying the Boolean expression.

<u>TWO VARIABLE K- MAP</u>:-

A two variable expression can have $2^2 = 4$ possible combinations of the input variables A and B.

Mapping of SOP Expression:-

- The 2 variable K-map has $2^2 = 4$ squares. These squares are called cells.
- A '1' is placed in any square indicates that corresponding minterm is included in the output expression, and a 0 or no entry in any square indicates that the corresponding minterm does not appear in the expression for output.

		B	
		0	1
A	0	$\bar{A}\,\bar{B}$	$\bar{A}B$
	1	$A\bar{B}$	AB

Example:-

Map expression f= $\bar{A}B + A\bar{B}$

Solution:-

The expression minterms is

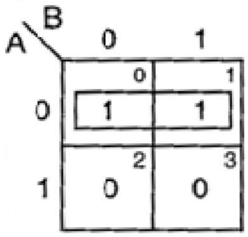
$F = m_1 + m_2 = m(1, 2)$

		B	
		0	1
A	0	0 0	1 1
	1	2 1	3 0

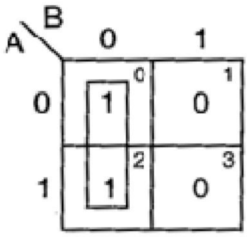
Minimization of SOP Expression:-

To minimize a Boolean expression given in the SOP form by using K- map, the adjacent squares having 1s, that is minterms adjacent to each other are combined to form larger squares to eliminate some variables.

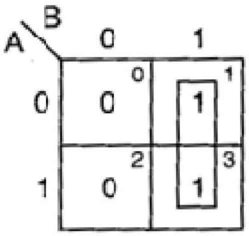
The possible minterm grouping in a two variable K- map are shown below



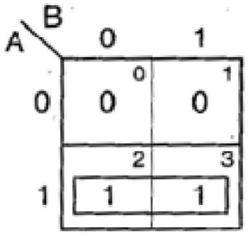
$f_1 = \bar{A}$



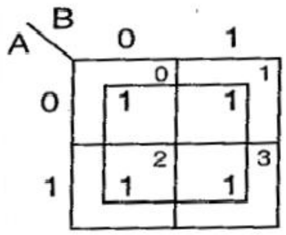
$f_2 = \bar{B}$



$f_3 = B$



$f_4 = A$



$f_5 = 1$

- Two minterms, which are adjacent to each other, can be combined to form a bigger square called 2 – square or a pair. This eliminates one variable that is not common to both the minterms.
- Two 2-squares adjacent to each other can be combined to form a 4- square. A 4- square eliminates 2 variables. A 4-square is called a quad.
- Consider only those variables which remain constant throughout the square, and ignore the variables which are varying. The non-complemented variable is the variable remaining constant as 1.The complemented variable is the variable remaining constant as a 0 and the variables are written as a product term.

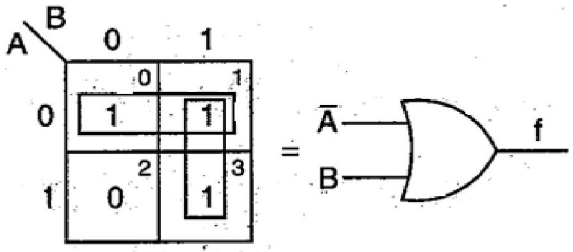
Example:-

Reduce the expression f= $\bar{A}\bar{B} + \bar{A}\,B + AB$ using mapping.

Solution:-

Expressed in terms of minterms, the given expression is

$$f = m_0 + m_1 + m_3 = \sum m\ (0, 1, 3)$$

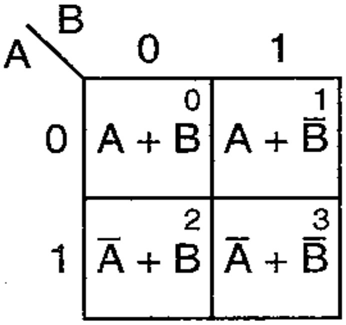


$f = \bar{A} + B$

F = $\bar{A}$ + B

Mapping of POS Expression:-

Each sum term in the standard POS expression is called a Maxterm. A function in two variables (A,B) has 4 possible maxterms, A + B, A + $\overline{B}$, $\overline{A}$ + B and $\overline{A}$ + $\overline{B}$. They are represented as M_0, M_1, M_2 and M_3 respectively.



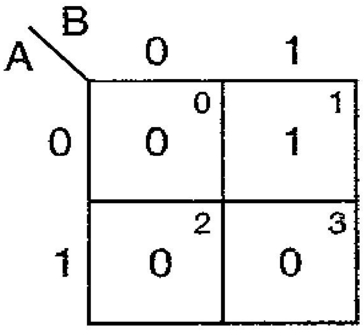
The maxterm of a two variable K-map

Example:-

Plot the expression f= (A + B)($\overline{A}$ + B)($\overline{A}$ + $\overline{B}$)

Solution:-

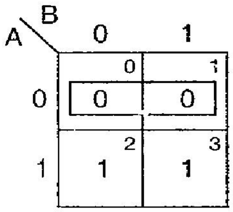
Expression interms of maxterms is f = ΠM (0, 2, 3)



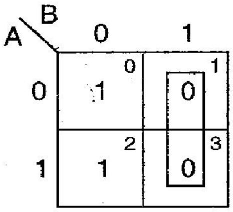
Minimization of POS Expressions:-

In POS form the adjacent 0s are combined into large square as possible. If the squares having complemented variable then the value remain constant as a 1 and the non-complemented variable if its value remains constant as a 0 along the entire square and then their sum term is written.

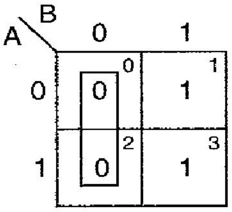
The possible maxterms grouping in a two variable K-map are shown below



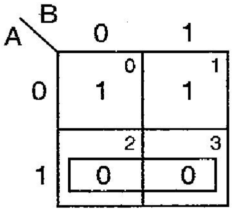
f_1 = A



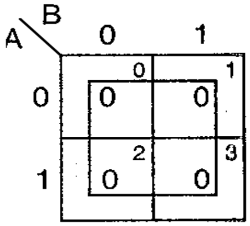
f_2 = $\overline{B}$



f_3 = B



f_4 = $\overline{A}$



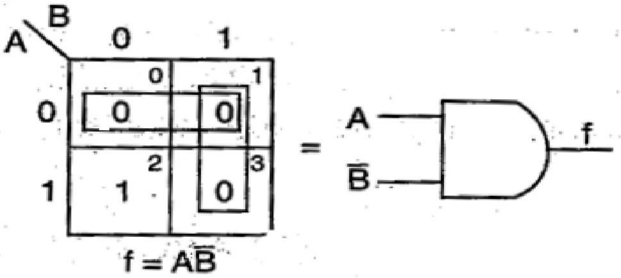
f_5 = 0

Example:-

Reduce the expression $f = (A + \bar{B})(\bar{A} + \bar{B})(A + B)$ using mapping

Solution:-

The given expression in terms of maxterms is f = ΠM (0, 1, 3)

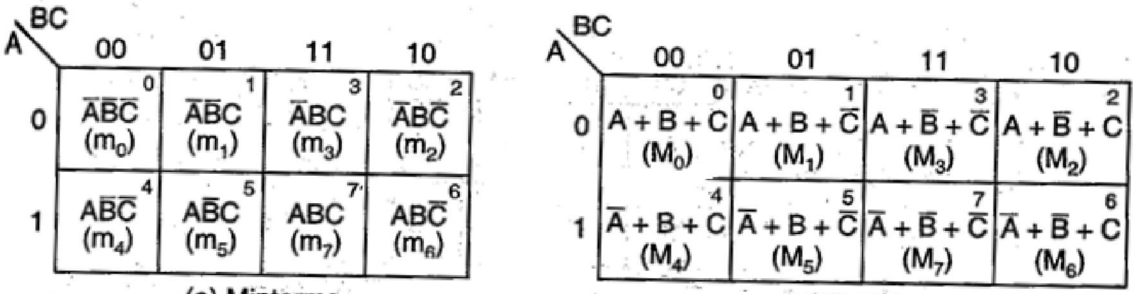


A \ B	0	1
0	0 (0)	0 (1)
1	1 (2)	0 (3)

$f = A\bar{B}$

THREE VARIABLE K- MAP:-

A function in three variables (A, B, C) can be expressed in SOP and POS form having eight possible combination. A three variable K- map have 8 squares or cells and each square on the map represents a minterm or maxterm is shown in the figure below.



A \ BC	00	01	11	10
0	$\bar{A}\bar{B}\bar{C}$ (m_0) [0]	$\bar{A}\bar{B}C$ (m_1) [1]	$\bar{A}BC$ (m_3) [3]	$\bar{A}B\bar{C}$ (m_2) [2]
1	$A\bar{B}\bar{C}$ (m_4) [4]	$A\bar{B}C$ (m_5) [5]	ABC (m_7) [7]	$AB\bar{C}$ (m_6) [6]

(a) Minterms

A \ BC	00	01	11	10
0	$A+B+C$ (M_0) [0]	$A+B+\bar{C}$ (M_1) [1]	$A+\bar{B}+\bar{C}$ (M_3) [3]	$A+\bar{B}+C$ (M_2) [2]
1	$\bar{A}+B+C$ (M_4) [4]	$\bar{A}+B+\bar{C}$ (M_5) [5]	$\bar{A}+\bar{B}+\bar{C}$ (M_7) [7]	$\bar{A}+\bar{B}+C$ (M_6) [6]

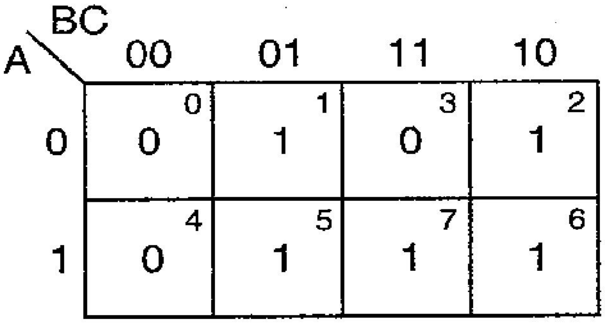
(b) Maxterms

Example:-

Map the expression $f = \bar{A}\bar{B}C + A\bar{B}C + \bar{A}B\bar{C} + AB\bar{C} + ABC$

Solution:-

So in the SOP form the expression is f = ∑ m (1, 5, 2, 6, 7)



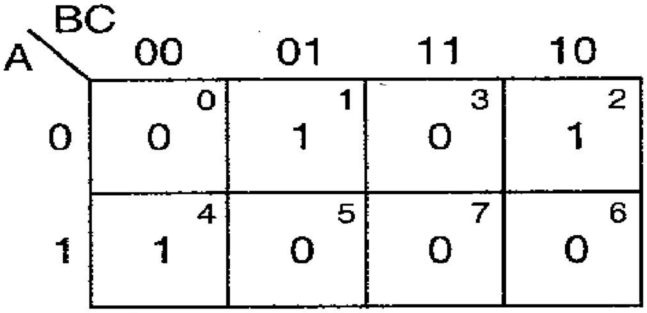
A \ BC	00	01	11	10
0	0 (0)	1 (1)	0 (3)	1 (2)
1	0 (4)	1 (5)	1 (7)	1 (6)

Example:-

Map the expression $f = (A + B + C)(\bar{A} + B + \bar{C})(\bar{A} + \bar{B} + \bar{C})(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + C)$

Solution:-

So in the POS form the expression is f = π M (0, 5, 7, 3, 6)



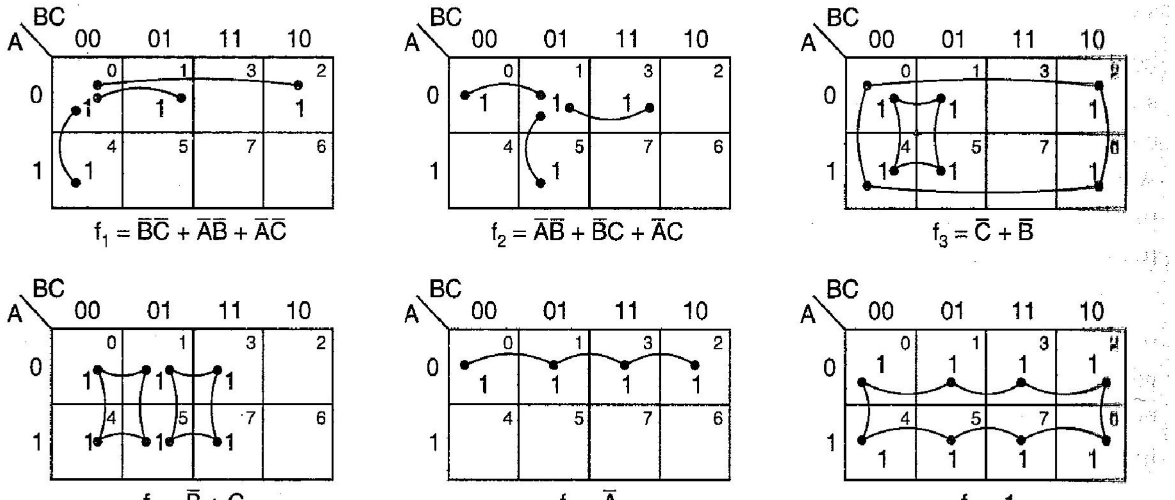
A \ BC	00	01	11	10
0	0 0	1 1	3 0	2 1
1	4 1	5 0	7 0	6 0

Minimization of SOP and POS Expressions:-

For reducing the Boolean expressions in SOP (POS) form the following steps are given below

- Draw the K-map and place 1s (0s) corresponding to the minterms (maxterms) of the SOP (POS) expression.
- In the map 1s (0s) which are not adjacent to any other 1(0) are the isolated minterms (maxterms). They are to be read as they are because they cannot be combined even into a 2-square.
- For those 1s (0s) which are adjacent to only one other 1(0) make them pairs (2 squares).
- For quads (4- squares) and octet (8 squares) of adjacent 1s (0s) even if they contain some 1s (0s) which have already been combined. They must geometrically form a square or a rectangle.
- For any 1s (0s) that have not been combined yet then combine them into bigger squares if possible.
- Form the minimal expression by summing (multiplying) the product (sum) terms of all the groups.

Some of the possible combinations of minterms in SOP form



$f_1 = \overline{B}\overline{C} + \overline{A}\overline{B} + \overline{A}\overline{C}$

$f_2 = \overline{A}\overline{B} + \overline{B}C + \overline{A}C$

$f_3 = \overline{C} + \overline{B}$

$f_4 = \overline{B} + C$

$f_5 = \overline{A}$

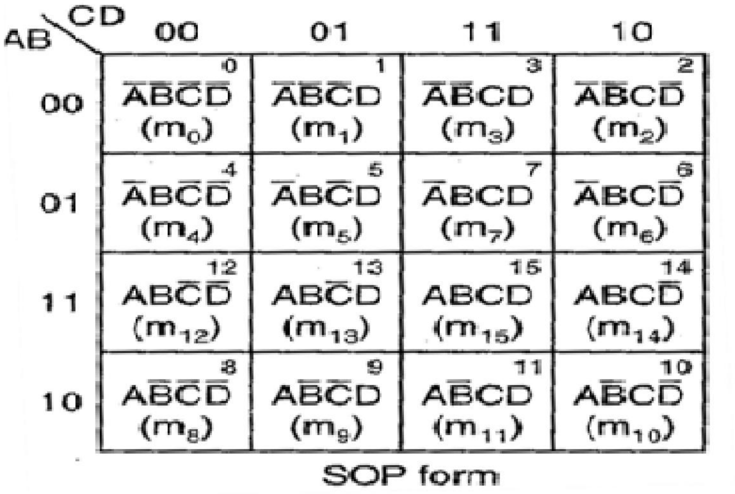
$f_6 = 1$

These possible combinations are also for POS but 1s are replaced by 0s.

FOUR VARIABLE K-MAP:-

A four variable (A, B, C, D) expression can have 2^4 = 16 possible combinations of input variables. A four variable K-map has 2^4 = 16 squares or cells and each square on the map represents either a minterm or a maxterm as shown in the figure below. The binary number designations of the rows and columns are in the gray code. The binary numbers along the top of the map indicate the conditions of C and D along any column and binary numbers along left side indicate the conditions of A and B along any row. The numbers in the top right corners of the squares indicate the minterm or maxterm desginations.

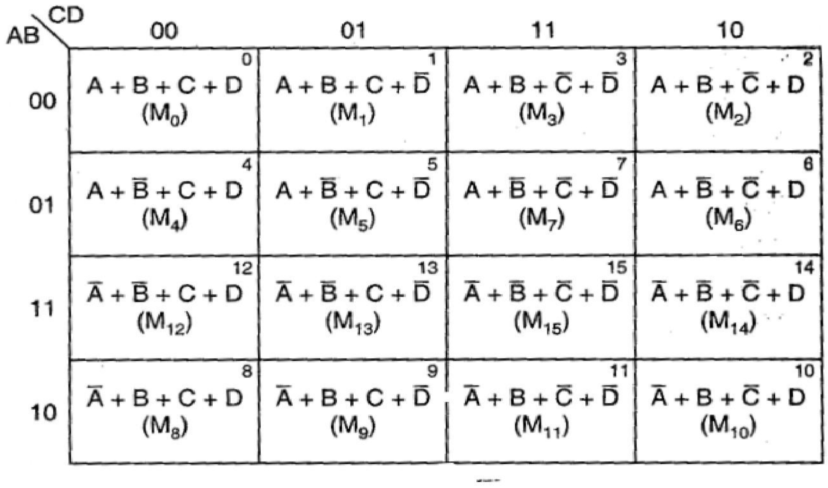
SOP FORM



AB \ CD	00	01	11	10
00	0 $\overline{A}\,\overline{B}\,\overline{C}\,\overline{D}$ (m_0)	1 $\overline{A}\,\overline{B}\,\overline{C}D$ (m_1)	3 $\overline{A}\,\overline{B}CD$ (m_3)	2 $\overline{A}\,\overline{B}C\overline{D}$ (m_2)
01	4 $\overline{A}B\overline{C}\,\overline{D}$ (m_4)	5 $\overline{A}B\overline{C}D$ (m_5)	7 $\overline{A}BCD$ (m_7)	6 $\overline{A}BC\overline{D}$ (m_6)
11	12 $AB\overline{C}\,\overline{D}$ (m_{12})	13 $AB\overline{C}D$ (m_{13})	15 $ABCD$ (m_{15})	14 $ABC\overline{D}$ (m_{14})
10	8 $A\overline{B}\,\overline{C}\,\overline{D}$ (m_8)	9 $A\overline{B}\,\overline{C}D$ (m_9)	11 $A\overline{B}CD$ (m_{11})	10 $A\overline{B}C\overline{D}$ (m_{10})

SOP form

POS FORM



AB \ CD	00	01	11	10
00	0 $A+B+C+D$ (M_0)	1 $A+B+C+\overline{D}$ (M_1)	3 $A+B+\overline{C}+\overline{D}$ (M_3)	2 $A+B+\overline{C}+D$ (M_2)
01	4 $A+\overline{B}+C+D$ (M_4)	5 $A+\overline{B}+C+\overline{D}$ (M_5)	7 $A+\overline{B}+\overline{C}+\overline{D}$ (M_7)	6 $A+\overline{B}+\overline{C}+D$ (M_6)
11	12 $\overline{A}+\overline{B}+C+D$ (M_{12})	13 $\overline{A}+\overline{B}+C+\overline{D}$ (M_{13})	15 $\overline{A}+\overline{B}+\overline{C}+\overline{D}$ (M_{15})	14 $\overline{A}+\overline{B}+\overline{C}+D$ (M_{14})
10	8 $\overline{A}+B+C+D$ (M_8)	9 $\overline{A}+B+C+\overline{D}$ (M_9)	11 $\overline{A}+B+\overline{C}+\overline{D}$ (M_{11})	10 $\overline{A}+B+\overline{C}+D$ (M_{10})

Minimization of SOP and POS Expressions:-

For reducing the Boolean expressions in SOP (POS) form the following steps are given below

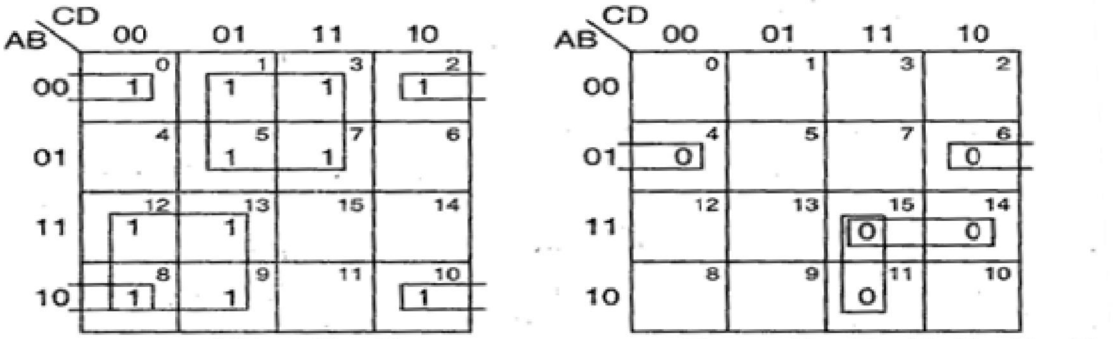
- Draw the K-map and place 1s (0s) corresponding to the minterms (maxterms) of the SOP (POS) expression.
- In the map 1s (0s) which are not adjacent to any other 1(0) are the isolated minterms (maxterms). They are to be read as they are because they cannot be combined even into a 2-square.
- For those 1s (0s) which are adjacent to only one other 1(0) make them pairs (2 squares).
- For quads (4- squares) and octet (8 squares) of adjacent 1s (0s) even if they contain some 1s (0s) which have already been combined. They must geometrically form a square or a rectangle.
- For any 1s (0s) that have not been combined yet then combine them into bigger squares if possible.
- Form the minimal expression by summing (multiplying) the product (sum) terms of all the groups.

Example:-

Reduce using mapping the expression f = ∑ m (0, 1, 2, 3, 5, 7, 8, 9, 10, 12, 13)

Solution:-

The given expression in POS form is f = ℿ M (4, 6, 11, 14, 15) and in SOP form f = ∑ m (0, 1, 2, 3, 5, 7, 8, 9, 10, 12, 13)



AB \ CD	00	01	11	10
00	1	1	1	1
01		1	1	
11	1	1		
10	1	1		1

$f_{min} = \overline{B}\,\overline{D} + A\overline{C} + \overline{A}D$
(a) SOP K-map

AB \ CD	00	01	11	10
00				
01	0			0
11			0	0
10			0	

$f_{min} = (A + \overline{B} + D)(\overline{A} + \overline{C} + \overline{D})(\overline{A} + \overline{B} + \overline{C})$
(b) POS K-map

The minimal SOP expression is $f_{min} = \overline{B}\,\overline{D} + A\overline{C} + \overline{A}D$

The minimal POS expression is $f_{min} =(A +\overline{B} + D) (\overline{A} + \overline{C} + D) (A + B + C)$

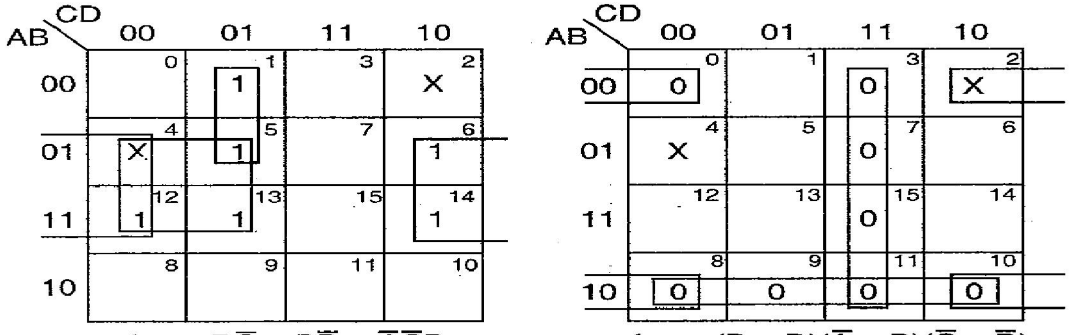
DON'T CARE COMBINATIONS:-

The combinations for which the values of the expression are not specified are called don't care combinations or optional combinations and such expression stand incompletely specified. The output is a don't care for these invalid combinations. The don't care terms are denoted by d or X. During the process of designing using SOP maps, each don't care is treated as 1 to reduce the map otherwise it is treated as 0 and left alone. During the process of designing using POS maps, each don't care is treated as 0 to reduce the map otherwise it is treated as 1 and left alone.

A standard SOP expression with don't cares can be converted into standard POS form by keeping the don't cares as they are, and the missing minterms of the SOP form are written as the maxterms of the POS form. Similarly, to convert a standard POS expression with don't cares can be converted into standard SOP form by keeping the don't cares as they are, and the missing maxterms of the POS form are written as the minterms of the SOP form.

Example:-
Reduce the expression f = ∑ m(1, 5, 6, 12, 13, 14) + d(2, 4) using K- map.
Solution:-
The given expression in SOP form is f = ∑ m (1, 5, 6, 12, 13, 14) + d(2, 4)

The given expression in POS form is f = π M (0, 3, 7, 8, 9, 10, 11,15) + d(2, 4)



AB \ CD	00	01	11	10
00		1		X
01	X	1		1
11	1	1		1
10				

$f_{min} = B\overline{C} + B\overline{D} + \overline{AC}D$
(a) SOP K-map

AB \ CD	00	01	11	10
00	0		0	X
01	X		0	
11			0	
10	0	0	0	0

$f_{min} = (B + D)(\overline{A} + B)(\overline{C} + \overline{D})$
(b) POS K-map

The minimal of SOP expression is $f_{min} = B\overline{C} + B\overline{D} +\overline{A}\,\overline{C}D$

The minimal of POS expression is $f_{min} = (B + D)(\overline{A} + B) (\overline{C} + \overline{D})$

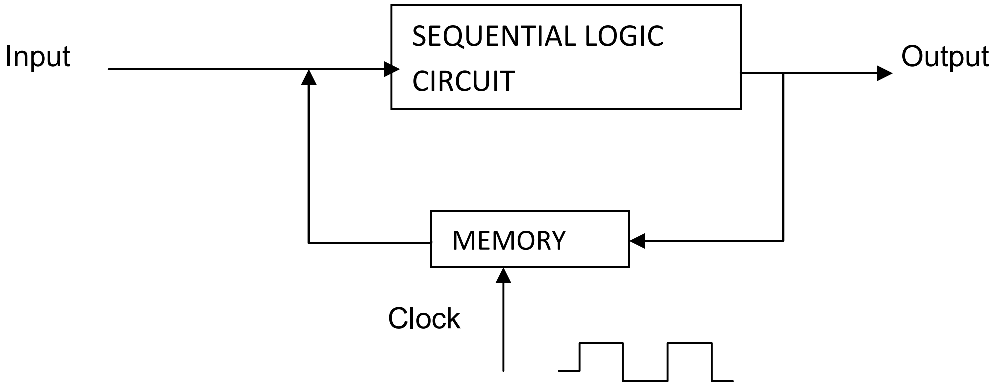
SEQUENTIAL LOGIC CIRCUIT

SEQUENTIAL CIRCUIT:-

- It is a circuit whose output depends upon the present input, previous output and the sequence in which the inputs are applied.

HOW THE SEQUENTIAL CIRCUIT IS DIFFERENT FROM COMBINATIONAL CIRCUIT? :-

- In combinational circuit output depends upon present input at any instant of time and do not use memory. Hence previous input does not have any effect on the circuit. But sequential circuit has memory and depends upon present input and previous output.
- Sequential circuits are slower than combinational circuits and these sequential circuits are harder to design.



[Block diagram of Sequential Logic Circuit]

- The data stored by the memory element at any given instant of time is called the present state of sequential circuit.

TYPES:-

Sequential logic circuits (SLC) are classified as

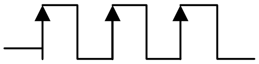
(i) Synchronous SLC
(ii) Asynchronous SLC

- The SLC that are controlled by clock are called synchronous SLC and those which are not controlled by a clock are asynchronous SLC.
- Clock:- A recurring pulse is called a clock.

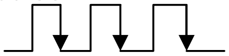
FLIP-FLOP AND LATCH:-

- A flip-flop or latch is a circuit that has two stable states and can be used to store information.
- A flip-flop is a binary storage device capable of storing one bit of information. In a stable state, the output of a flip-flop is either 0 or 1.
- Latch is a non-clocked flip-flop and it is the building block for the flip-flop.
- A storage element in digital circuit can maintain a binary state indefinitely until directed by an input signal to switch state.
- Storage element that operate with signal level are called latches and those operate with clock transition are called as flip-flops.

- The circuit can be made to change state by signals applied to one or more control inputs and will have one or two outputs.
- A flip-flop is called so because its output either flips or flops meaning to switch back and forth.
- A flip-flop is also called a bi-stable multi-vibrator as it has two stable states. The input signals which command the flip-flop to change state are called excitations.
- Flip-flops are storage devices and can store 1 or 0.
- Flip-flops using the clock signal are called clocked flip-flops. Control signals are effective only if they are applied in synchronization with the clock signal.
- Clock-signals may be positive-edge triggered or negative-edge triggered.
- Positive-edge triggered flip-flops are those in which state transitions take place only at positive- going edge of the clock pulse.



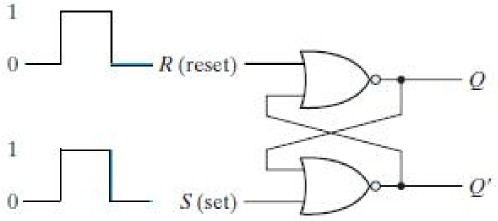
- Negative-edge triggered flip-flops are those in which state transition take place only at negative- going edge of the clock pulse.



- Some common type of flip-flops include
  a) SR (set-reset) F-F
  b) D (data or delay) F-F
  c) T (toggle) F-F and
  d) JK F-F

SR latch:-

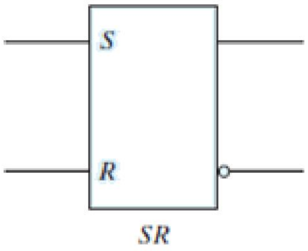
- The SR latch is a circuit with two cross-coupled NOR gates or two cross-coupled NAND gates.
- It has two outputs labeled Q and Q'. Two inputs are there labeled S for set and R foe reset.
- The latch has two useful states. When Q=0 and Q'=1 the condition is called reset state and when Q=1 and Q'=0 the condition is called set state.
- Normally Q and Q' are complement of each other.
- The figure represents a SR latch with two cross-coupled NOR gates. The circuit has NOR gates and as we know if any one of the input for a NOR gate is HIGH then its output will be LOW and if both the inputs are LOW then only the output will be HIGH.



- Under normal conditions, both inputs of the latch remain at 0 unless the state has to be changed. The application of a momentary 1 to the S input causes the latch to go to the set state. The S input must go back to 0 before any other changes take place, in order to avoid the occurrence of an undefined next state that results from the forbidden input condition.
- The first condition (S = 1, R = 0) is the action that must be taken by input S to bring the circuit to the set state. Removing the active input from S leaves the circuit in the same state. After both inputs return to 0, it is then possible to shift to the reset state by momentary applying a 1 to the R input. The 1 can then be removed from R, whereupon the circuit remains in the reset state. When both inputs S and R are equal to 0, the latch can be in either the set or the reset state, depending on which input was most recently a 1.

- If a 1 is applied to both the S and R inputs of the latch, both outputs go to 0. This action produces an undefined next state, because the state that results from the input transitions depends on the order in which they return to 0. It also violates the requirement that outputs be the complement of each other. In normal operation, this condition is avoided by making sure that 1's are not applied to both inputs simultaneously.
- Truth table for SR latch designed with NOR gates is shown below.

Input		Output				Comment
S	R	Q	Q'	Q_{Next}	Q'_{Next}	
0	0	0	1	0	1	No change
0	0	1	0	1	0	
0	1	0	1	0	1	Reset
0	1	1	0	0	1	
1	0	0	1	1	0	Set
1	0	1	0	1	0	
1	1	0	1	X	X	Prohibited state
1	1	1	0	X	X	



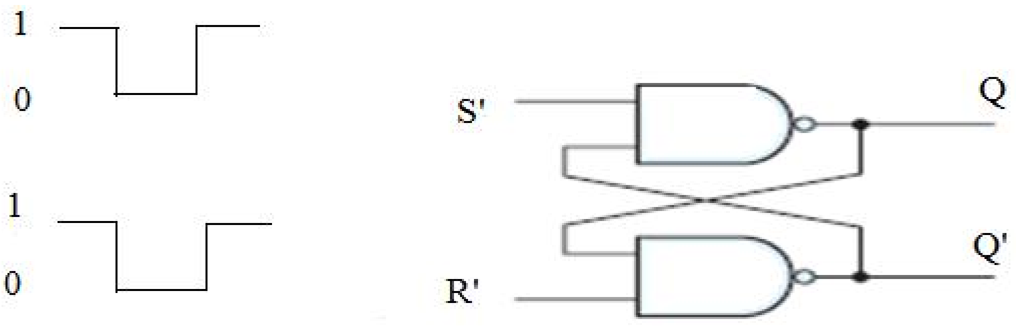
Symbol for SR NOR Latch

Racing Condition:-

In case of a SR latch when S=R=1 input is given both the output will try to become 0. This is called Racing condition.

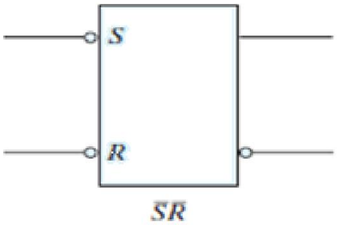
SR latch using NAND gate:-

- The below figure represents a SR latch with two cross-coupled NAND gates. The circuit has NAND gates and as we know if any one of the input for a NAND gate is LOW then its output will be HIGH and if both the inputs are HIGH then only the output will be LOW.
- It operates with both inputs normally at 1, unless the state of the latch has to be changed. The application of 0 to the S input causes output Q to go to 1, putting the latch in the set state. When the S input goes back to 1, the circuit remains in the set state. After both inputs go back to 1, we are allowed to change the state of the latch by placing a 0 in the R input. This action causes the circuit to go to the reset state and stay there even after both inputs return to 1.



- The condition that is forbidden for the NAND latch is both inputs being equal to 0 at the same time, an input combination that should be avoided.

In comparing the NAND with the NOR latch, note that the input signals for the NAND require the complement of those values used for the NOR latch. Because the NAND latch requires a 0 signal to change its state, it is sometimes referred to as an S'R' latch. The primes (or, sometimes, bars over the letters) designate the fact that the inputs must be in their complement form to activate the circuit.



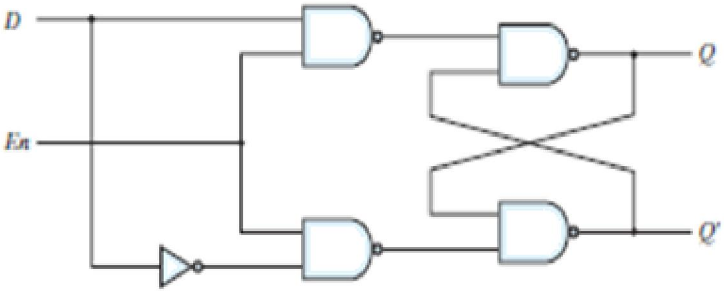
The above represents the symbol for inverted SR latch or SR latch using NAND gate.

Truth table for SR latch using NAND gate or Inverted SR latch

S	R	Q_{next}	Q'_{next}
0	0	Race	Race
0	1	0	1 (Reset)
1	0	1	0 (Set)
1	1	Q (No change)	Q' (No change)

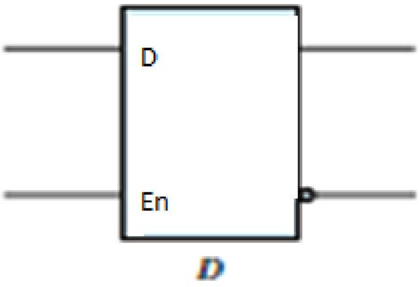
D LATCH:-

- One way to eliminate the undesirable condition of the indeterminate state in the SR latch is to ensure that inputs S and R are never equal to 1 at the same time.



(a) Logic diagram

- This is done in the D latch. This latch has only two inputs: D (data) and En (enable).
- The D input goes directly to the S input, and its complement is applied to the R input.



(Symbol for D-Latch)

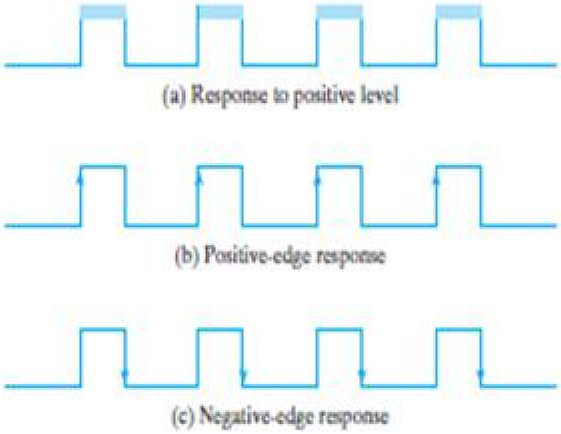
- As long as the enable input is at 0, the cross-coupled SR latch has both inputs at the 1 level and the circuit can't change state regardless of the value of D.
- The below represents the truth table for the D-latch.

En	D	Next State of Q
0	X	No change
1	0	Q=0;Reset State
1	1	Q=1;Set State

- The D input is sampled when En = 1. If D = 1, the Q output goes to 1, placing the circuit in the set state. If D = 0, output Q goes to 0, placing the circuit in the reset state. This situation provides a path from input D to the output, and for this reason, the circuit is often called a TRANSPARENT latch.

TRIGGERING METHODS:-

- The state of a latch or flip-flop is switched by a change in the control input. This momentary change is called a trigger, and the transition it causes is said to trigger the flip-flop.
- Flip-flop circuits are constructed in such a way as to make them operate properly when they are part of a sequential circuit that employs a common clock.
- The problem with the latch is that it responds to a change in the level of a clock pulse. For proper operation of a flip-flop it should be triggered only during a signal transition.
- This can be accomplished by eliminating the feedback path that is inherent in the operation of the sequential circuit using latches. A clock pulse goes through two transitions: from 0 to 1 and the return from 1 to 0.
- A ways that a latch can be modified to form a flip-flop is to produce a flip-flop that triggers only during a signal transition (from 0 to 1 or from 1 to 0) of the synchronizing signal (clock) and is disabled during the rest of the clock pulse.



(a) Response to positive level

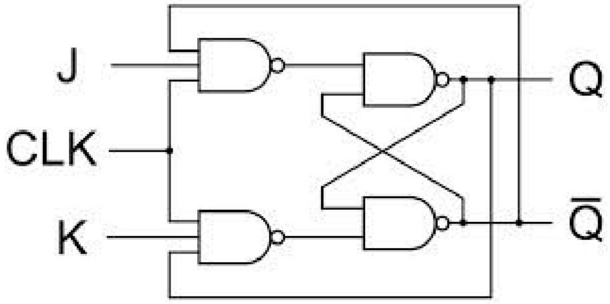
(b) Positive-edge response

(c) Negative-edge response

JK FLIP-FLOP:-

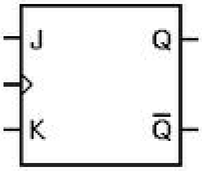
- The JK flip-flop can be constructed by using basic SR latch and a clock. In this case the outputs Q and Q' are returned back and connected to the inputs of NAND gates.
- This simple JK flip Flop is the most widely used of all the flip-flop designs and is considered to be a universal flip-flop circuit.
- The sequential operation of the JK flip flop is exactly the same as for the previous SR flip-flop with the same "Set" and "Reset" inputs.
- The difference this time is that the "JK flip flop" has no invalid or forbidden input states of the SR Latch even when S and R are both at logic "1".

(The below diagram shows the circuit diagram of a JK flip-flop)



- The JK flip flop is basically a gated SR Flip-flop with the addition of a clock input circuitry that prevents the illegal or invalid output condition that can occur when both inputs S and R are equal to logic level "1".
- Due to this additional clocked input, a JK flip-flop has four possible input combinations, "logic 1", "logic 0", "no change" and "toggle".

- The symbol for a JK flip flop is similar to that of an SR bistable latch except the clock input.



(The above diagram shows the symbol of a JK flip-flop.)

- Both the S and the R inputs of the SR bi-stable have now been replaced by two inputs called the J and K inputs, respectively after its inventor Jack and Kilby. Then this equates to: J = S and K = R.
- The two 2-input NAND gates of the gated SR bi-stable have now been replaced by two 3-input NAND gates with the third input of each gate connected to the outputs at Q and Q'.
- This cross coupling of the SR flip-flop allows the previously invalid condition of S = "1" and R = "1" state to be used to produce a "toggle action" as the two inputs are now interlocked.
- If the circuit is now "SET" the J input is inhibited by the "0" status of Q' through the lower NAND gate. If the circuit is "RESET" the K input is inhibited by the "0" status of Q through the upper NAND gate. As Q and Q' are always different we can use them to control the input.

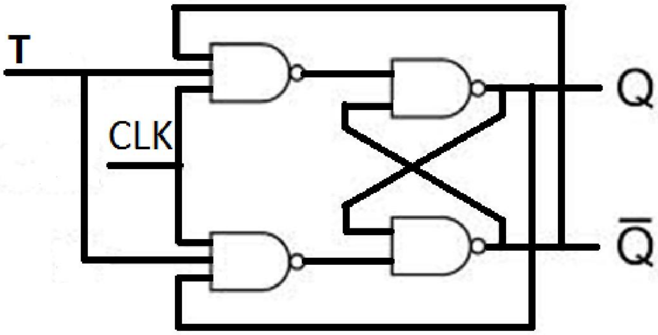
(Truth table for JK flip-flop)

Input		Output		Comment
J	K	Q	Q_{next}	
0	0	0	0	No change
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

- When both inputs J and K are equal to logic "1", the JK flip flop toggles.

T FLIP-FLOP:-

- Toggle flip-flop or commonly known as T flip-flop.
- This flip-flop has the similar operation as that of the JK flip-flop with both the inputs J and K are shorted i.e. both are given the common input.



- Hence its truth table is same as that of JK flip-flop when J=K= 0 and J=K=1.So its truth table is as follows.

T	Q	Q_{next}	Comment
0	0	0	No change
	1	1	
1	0	1	Toggles
	1	0	

CHARACTERISTIC TABLE:-

- A characteristic table defines the logical properties of a flip-flop by describing its operation in tabular form.
- The next state is defined as a function of the inputs and the present state.
- Q (t) refers to the present state and Q (t + 1) is the next.
- Thus, Q (t) denotes the state of the flip-flop immediately before the clock edge, and Q(t + 1) denotes the state that results from the clock transition.
- The characteristic table for the JK flip-flop shows that the next state is equal to the present state when inputs J and K are both equal to 0. This condition can be expressed as Q (t + 1) = Q (t), indicating that the clock produces no change of state.

Characteristic Table Of JK Flip-Flop

J	K	Q(t+1)
0	0	Q(t) No change
0	1	0 Reset
1	0	1 Set
1	1	Q'(t) Complement

- When K = 1 and J = 0, the clock resets the flip-flop and Q(t + 1) = 0. With J = 1 and K = 0, the flip-flop sets and Q(t + 1) = 1. When both J and K are equal to 1, the next state changes to the complement of the present state, a transition that can be expressed as Q(t + 1) = Q'(t).
- The characteristic equation for JK flip-flop is represented as

$$Q(t+1)= JQ' + K'Q$$

Characteristic Table of D Flip-Flop

D	Q(t+1)
0	0
1	1

- The next state of a D flip-flop is dependent only on the D input and is independent of the present state.
- This can be expressed as Q (t + 1) = D. It means that the next-state value is equal to the value of D. Note that the D flip-flop does not have a "no-change" condition and its characteristic equation is written as Q(t+1)=D.

Characteristic Table of T Flip-Flop

T	Q(t+1)
0	Q(t) No change
1	Q'(t) Complement

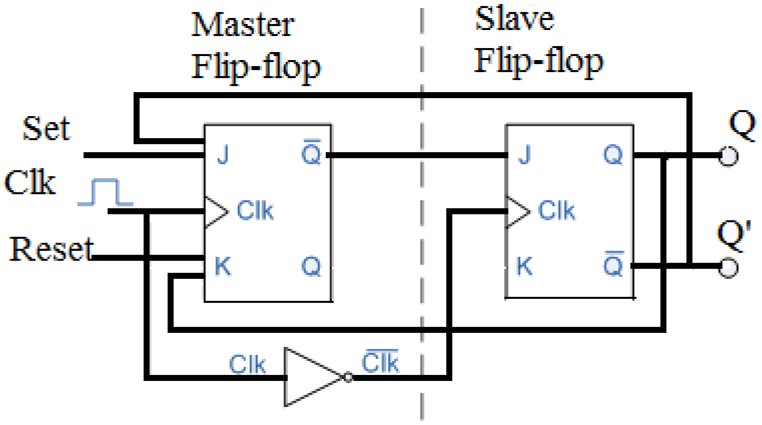
- The characteristic table of T flip-flop has only two conditions: When T = 0, the clock edge does not change the state; when T = 1, the clock edge complements the state of the flip-flop and the characteristic equation is

$$Q(t+1)= T \oplus Q = T'Q + TQ'$$

MASTER-SLAVE JK FLIP-FLOP:-

- The Master-Slave Flip-Flop is basically two gated SR flip-flops connected together in a series configuration with the slave having an inverted clock pulse.
- The outputs from Q and Q' from the "Slave" flip-flop are fed back to the inputs of the "Master" with the outputs of the "Master" flip flop being connected to the two inputs of the "Slave" flip flop.
- This feedback configuration from the slave's output to the master's input gives the characteristic toggle of the JK flip flop as shown below.

The Master-Slave JK Flip Flop



- The input signals J and K are connected to the gated "master" SR flip flop which "locks" the input condition while the clock (Clk) input is "HIGH" at logic level "1".
- As the clock input of the "slave" flip flop is the inverse (complement) of the "master" clock input, the "slave" SR flip flop does not toggle.
- The outputs from the "master" flip flop are only "seen" by the gated "slave" flip flop when the clock input goes "LOW" to logic level "0".
- When the clock is "LOW", the outputs from the "master" flip flop are latched and any additional changes to its inputs are ignored.
- The gated "slave" flip flop now responds to the state of its inputs passed over by the "master" section.
- Then on the "Low-to-High" transition of the clock pulse the inputs of the "master" flip flop are fed through to the gated inputs of the "slave" flip flop and on the "High-to-Low" transition the same inputs are reflected on the output of the "slave" making this type of flip flop edge or pulse-triggered.
- Then, the circuit accepts input data when the clock signal is "HIGH", and passes the data to the output on the falling-edge of the clock signal.
- In other words, the Master-Slave JK Flip flop is a "Synchronous" device as it only passes data with the timing of the clock signal.

FLIP-FLOP CONVERSIONS:-

SR Flip Flop to JK Flip Flop

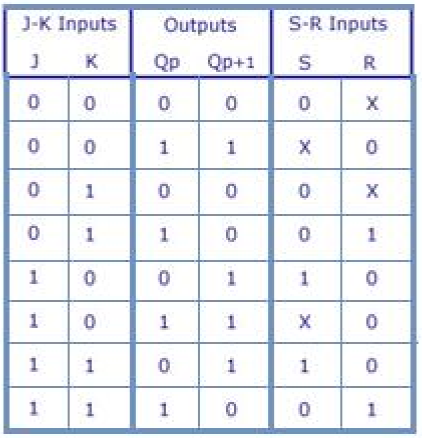
For this J and K will be given as external inputs to S and R. As shown in the logic diagram below, S and R will be the outputs of the combinational circuit.

The truth tables for the flip flop conversion are given below. The present state is represented by Qp and Qp+1 is the next state to be obtained when the J and K inputs are applied.

For two inputs J and K, there will be eight possible combinations. For each combination of J, K and Qp, the corresponding Qp+1 states are found. Qp+1 simply suggests the future values to be obtained by the JK flip flop after the value of Qp. The table is then completed by writing the values of S and R required to get each Qp+1 from the corresponding Qp. That is, the values of S and R that are required to change the state of the flip flop from Qp to Qp+1 are written.

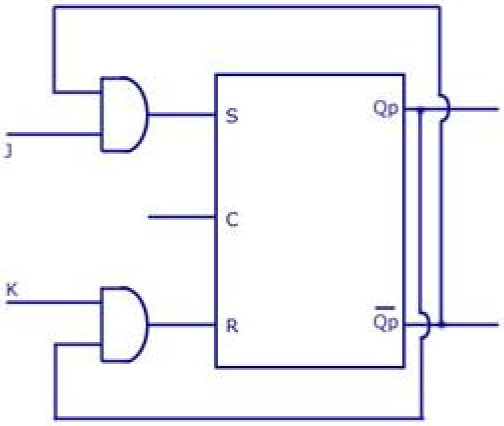
S-R Flip Flop to J-K Flip Flop

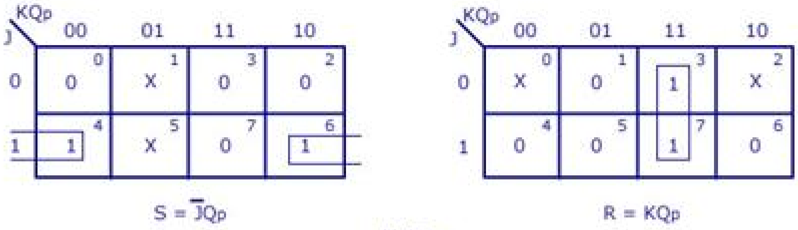
Conversion Table



J-K Inputs		Outputs		S-R Inputs	
J	K	Qp	Qp+1	S	R
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	X	0
1	1	0	1	1	0
1	1	1	0	0	1

Logic Diagram





J \ KQp	00	01	11	10
0	0	X	0	0
1	1	X	0	1

$S = \overline{J}Qp$

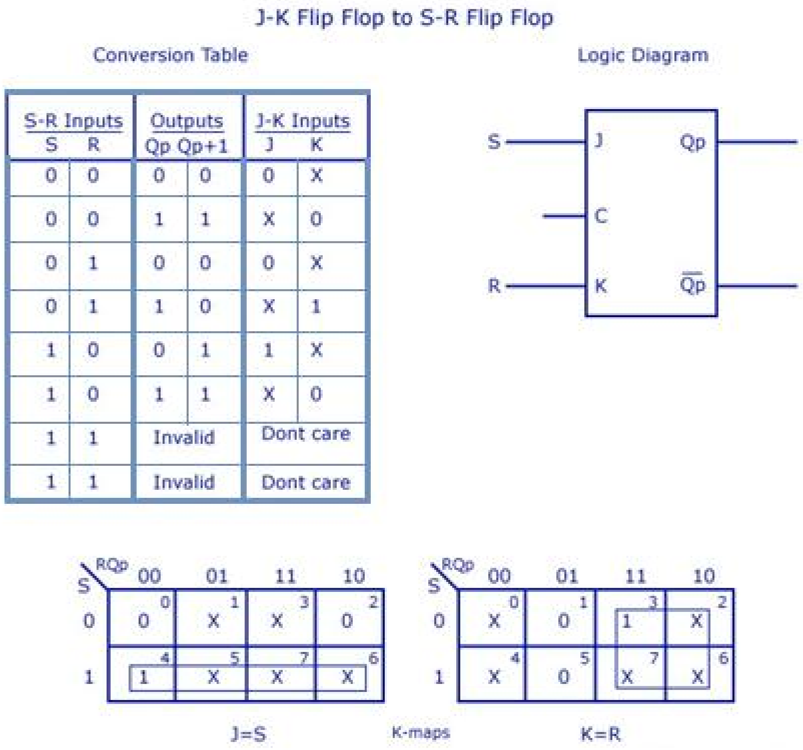
J \ KQp	00	01	11	10
0	X	0	1	X
1	0	0	1	0

$R = KQp$

K-Map

JK Flip Flop to SR Flip Flop

- This will be the reverse process of the above explained conversion. S and R will be the external inputs to J and K. J and K will be the outputs of the combinational circuit. Thus, the values of J and K have to be obtained in terms of S, R and Qp.
- A conversion table is to be written using S, R, Qp, Qp+1, J and K.
- For two inputs, S and R, eight combinations are made. For each combination, the corresponding Qp+1 outputs are found out.
- The outputs for the combinations of S=1 and R=1 are not permitted for an SR flip flop. Thus the outputs are considered invalid and the J and K values are taken as “don’t cares”.



Conversion Table

S-R Inputs		Outputs		J-K Inputs	
S	R	Qp	Qp+1	J	K
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	X	1
1	0	0	1	1	X
1	0	1	1	X	0
1	1	Invalid		Dont care	
1	1	Invalid		Dont care	

S \ RQp	00	01	11	10
0	0	X	X	0
1	1	X	X	X

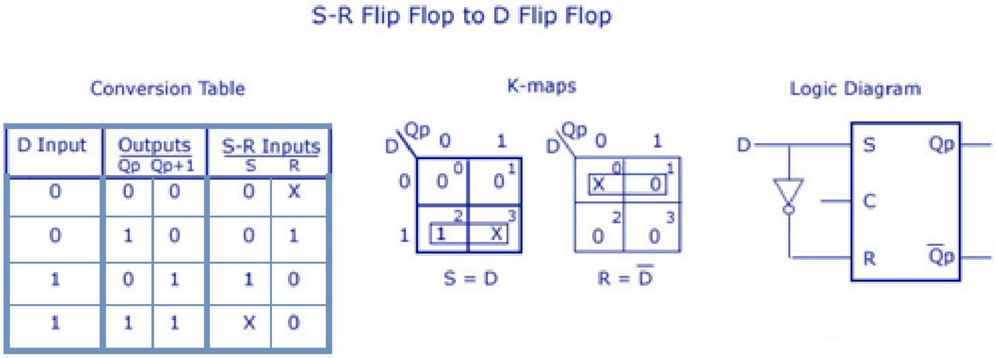
J=S

S \ RQp	00	01	11	10
0	X	0	1	X
1	X	0	X	X

K=R

SR Flip Flop to D Flip Flop

- S and R are the actual inputs of the flip flop and D is the external input of the flip flop.
- The four combinations, the logic diagram, conversion table, and the K-map for S and R in terms of D and Qp are shown below.



Conversion Table

D Input	Outputs		S-R Inputs	
	Qp	Qp+1	S	R
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0

D \ Qp	0	1
0	0	0
1	1	X

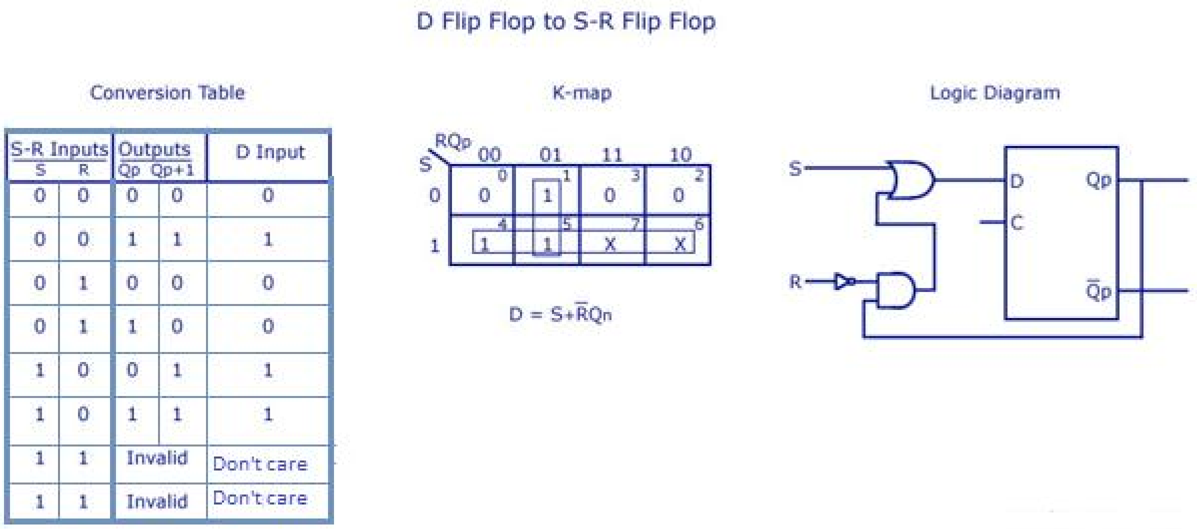
S = D

D \ Qp	0	1
0	X	0
1	0	0

R = $\overline{D}$

D Flip Flop to SR Flip Flop

- D is the actual input of the flip flop and S and R are the external inputs. Eight possible combinations are achieved from the external inputs S, R and Qp.
- But, since the combination of S=1 and R=1 are invalid, the values of Qp+1 and D are considered as “don’t cares”.
- The logic diagram showing the conversion from D to SR, and the K-map for D in terms of S, R and Qp are shown below.



D Flip Flop to S-R Flip Flop

Conversion Table

S-R Inputs		Outputs		D Input
S	R	Qp	Qp+1	
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	Invalid		Don't care
1	1	Invalid		Don't care

K-map

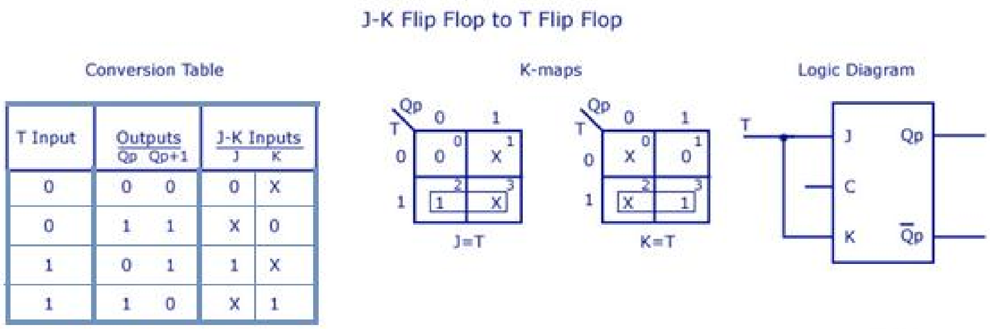
S \ RQp	00	01	11	10
0	0	1	0	0
1	1	1	X	X

D = S+$\bar{R}$Qn

Logic Diagram

JK Flip Flop to T Flip Flop:-

- J and K are the actual inputs of the flip flop and T is taken as the external input for conversion
- Four combinations are produced with T and Qp. J and K are expressed in terms of T and Qp.
  - The conversion table, K-maps, and the logic diagram are given below.



J-K Flip Flop to T Flip Flop

Conversion Table

T Input	Outputs		J-K Inputs	
	Qp	Qp+1	J	K
0	0	0	0	X
0	1	1	X	0
1	0	1	1	X
1	1	0	X	1

K-maps

T \ Qp	0	1
0	0	X
1	1	X

J=T

T \ Qp	0	1
0	X	0
1	X	1

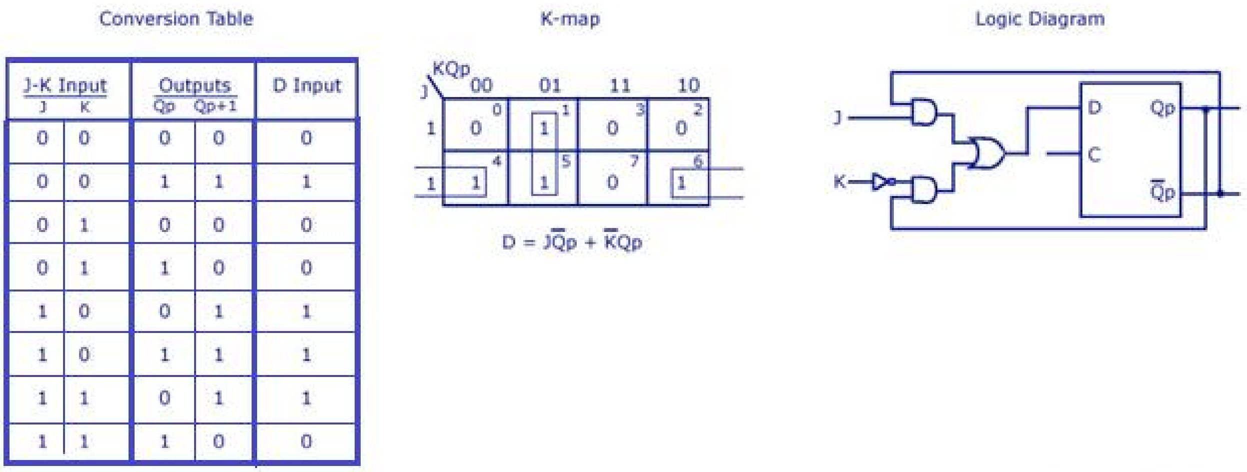
K=T

Logic Diagram

D Flip Flop to JK Flip Flop:-

- In this conversion, D is the actual input to the flip flop and J and K are the external inputs.
- J, K and Qp make eight possible combinations, as shown in the conversion table below. D is expressed in terms of J, K and Qp.
- The conversion table, the K-map for D in terms of J, K and Qp and the logic diagram showing the conversion from D to JK are given in the figure below.

D Flip Flop to J-K Flip Flop



Conversion Table

J-K Input		Outputs		D Input
J	K	Qp	Qp+1	
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

K-map

J \ KQp	00	01	11	10
1	0	1	0	0
1	1	1	0	1

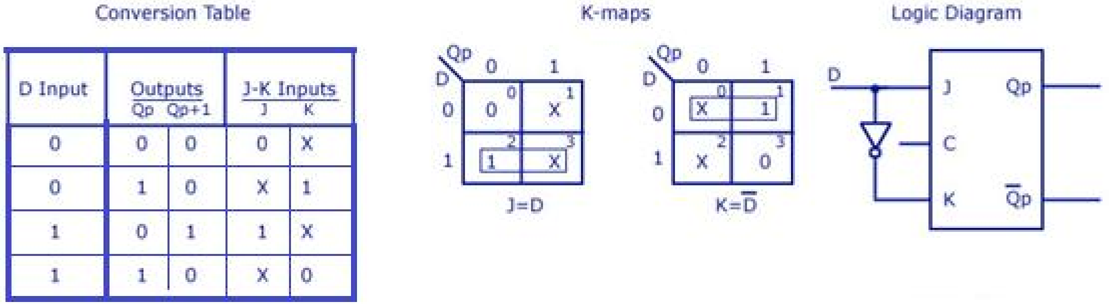
$D = J\overline{Qp} + \overline{K}Qp$

Logic Diagram

JK Flip Flop to D Flip Flop:-

- D is the external input and J and K are the actual inputs of the flip flop. D and Qp make four combinations. J and K are expressed in terms of D and Qp.
- The four combination conversion table, the K-maps for J and K in terms of D and Qp.

J-K Flip Flop to D Flip Flop



Conversion Table

D Input	Outputs		J-K Inputs	
	Qp	Qp+1	J	K
0	0	0	0	X
0	1	0	X	1
1	0	1	1	X
1	1	0	X	0

K-maps

D \ Qp	0	1
0	0	X
1	1	X

J=D

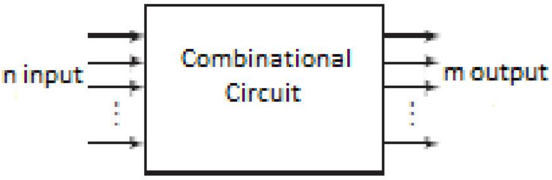
D \ Qp	0	1
0	X	1
1	X	0

$K=\overline{D}$

Logic Diagram

COMBINATIONAL LOGIC CIRCUIT

- A combinational circuit consists of logic gates whose outputs at any time are determined from only the present combination of inputs.
- A combinational circuit performs an operation that can be specified logically by a set of Boolean functions.
- It consists of an interconnection of logic gates. Combinational logic gates react to the values of the signals at their inputs and produce the value of the output signal, transforming binary information from the given input data to a required output data.
- A block diagram of a combinational circuit is shown in the below figure.
- The n input binary variables come from an external source; the m output variables are produced by the internal combinational logic circuit and go to an external destination.
- Each input and output variable exists physically as an analog signal whose values are interpreted to be a binary signal that represents logic 1and logic 0.

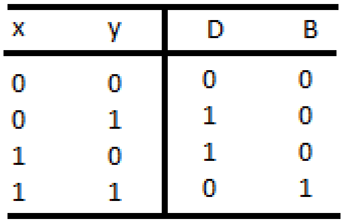


BINARY ADDER–SUBTRACTOR:-

- Digital computers perform a variety of information-processing tasks. Among the functions encountered are the various arithmetic operations.
- The most basic arithmetic operation is the addition of two binary digits. This simple addition consists of four possible elementary operations: 0 + 0 = 0, 0 + 1 = 1, 1 + 0 = 1, and 1 + 1 = 10.
- The first three operations produce a sum of one digit, but when both augend and addend bits are equal to 1; the binary sum consists of two digits. The higher significant bit of this result is called a carry.
- When the augend and addend numbers contain more significant digits, the carry obtained from the addition of two bits is added to the next higher order pair of significant bits.
- A combinational circuit that performs the addition of two bits is called a half adder.
- One that performs the addition of three bits (two significant bits and a previous carry) is a full adder. The names of the circuits stem from the fact that two half adders can be employed to implement a full adder.

HALF ADDER:-

- This circuit needs two binary inputs and two binary outputs.
- The input variables designate the augend and addend bits; the output variables produce the sum and carry. Symbols x and y are assigned to the two inputs and S (for sum) and C (for carry) to the outputs.
- The truth table for the half adder is listed in the below table.
- The C output is 1 only when both inputs are 1. The S output represents the least significant bit of the sum.
- The simplified Boolean functions for the two outputs can be obtained directly from the truth table.



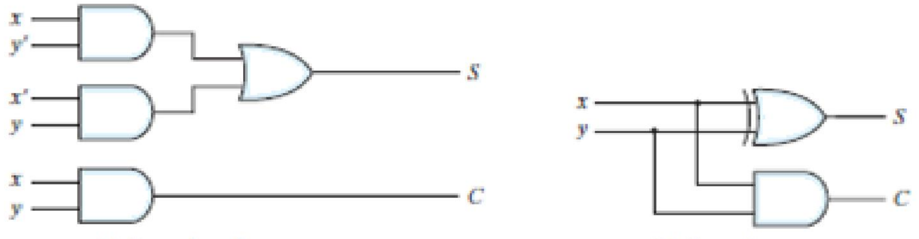
x	y	D	B
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Truth Table

- The simplified sum-of-products expressions are

$$S = x'y + xy'$$
$$C = xy$$

- The logic diagram of the half adder implemented in sum of products is shown in the below figure. It can be also implemented with an exclusive-OR and an AND gate.



(a) $S = xy' + x'y$
$C = xy$

(b) $S = x \oplus y$
$C = xy$

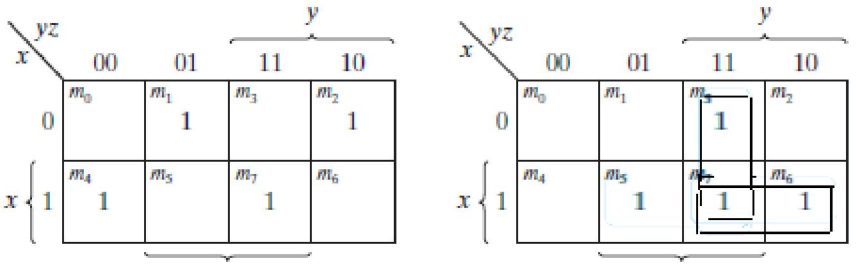
FULL ADDER:-

- A full adder is a combinational circuit that forms the arithmetic sum of three bits.
- It consists of three inputs and two outputs. Two of the input variables, denoted by x and y , represent the two significant bits to be added. The third input, z , represents the carry from the previous lower significant position.

x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Truth Table

- Two outputs are necessary because the arithmetic sum of three binary digits ranges in value from 0 to 3, and binary representation of 2 or 3 needs two bits. The two outputs are designated by the symbols S for sum and C for carry.



(a) $S = x'y'z + x'yz' + xy'z' + xyz$

(b) $C = xy + xz + yz$

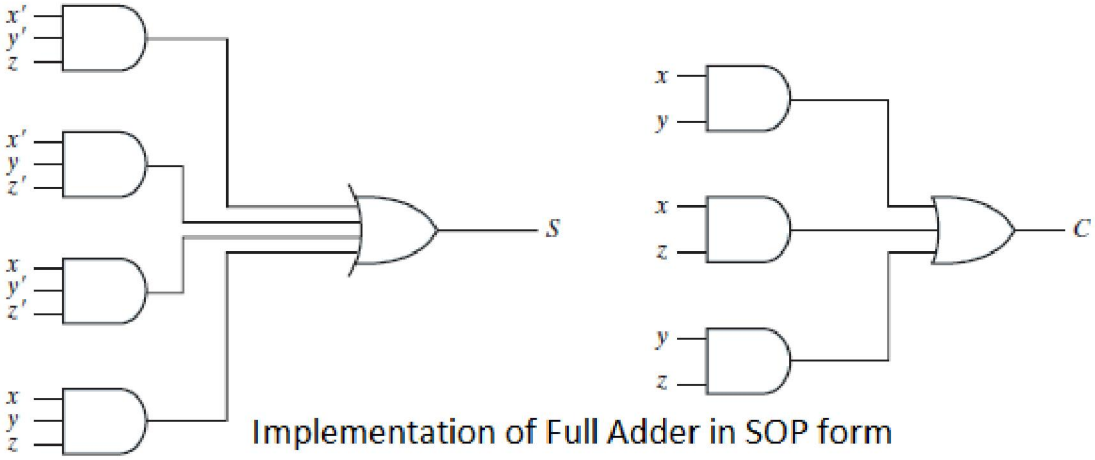
K-Map for full adder

- The binary variable S gives the value of the least significant bit of the sum. The binary variable C gives the output carry formed by adding the input carry and the bits of the words.
- The eight rows under the input variables designate all possible combinations of the three variables. The output variables are determined from the arithmetic sum of the input bits. When all input bits are 0, the output is 0.
- The S output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1. The C output has a carry of 1 if two or three inputs are equal to 1.
- The simplified expressions are

$$S = x'y'z + x'yz' + xy'z' + xyz$$

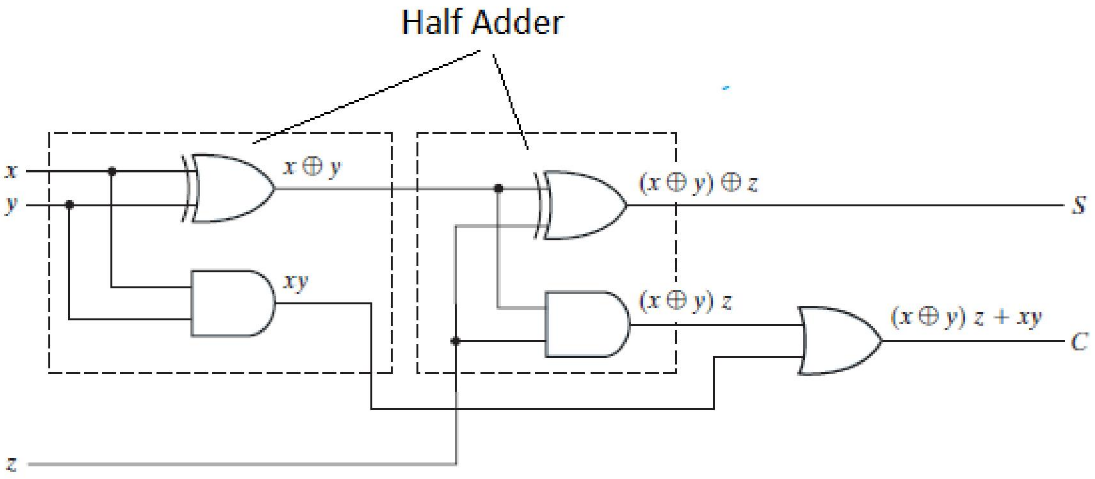
$$C = xy + xz + yz$$

- The logic diagram for the full adder implemented in sum-of-products form is shown in figure.



Implementation of Full Adder in SOP form

- It can also be implemented with two half adders and one OR gate as shown in the figure.



Implementation of Full Adder using Two Half Adders and an OR gate

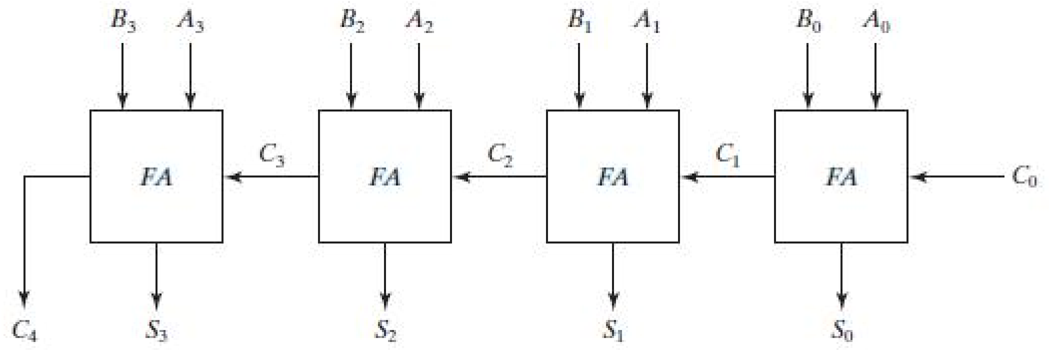
- A full adder is a combinational circuit that forms the arithmetic sum of three bits.

BINARY ADDER:-

- A binary adder is a digital circuit that produces the arithmetic sum of two binary numbers.
- It can be constructed with full adders connected in cascade, with the output carry from each full adder connected to the input carry of the next full adder in the chain.
- Addition of n-bit numbers requires a chain of n full adders or a chain of one-half adder and n-1 full adders. In the former case, the input carry to the least significant position is fixed at 0.
- The interconnection of four full-adder (FA) circuits to provide a four-bit binary ripple carry adder is shown in the figure.
- The augend bits of A and the addend bits of B are designated by subscript numbers from right to left, with subscript 0 denoting the least significant bit.
- The carries are connected in a chain through the full adders. The input carry to the adder is C0, and it ripples through the full adders to the output carry C4. The S outputs generate the required sum bits.
- An n -bit adder requires n full adders, with each output carry connected to the input carry of the next higher order full adder.
- Consider the two binary numbers A = 1011 and B = 0011. Their sum S = 1110 is formed with the four-bit adder as follows:

Subscript i:	3	2	1	0	
Input carry	0	1	1	0	C_i
Augend	1	0	1	1	A_i
Addend	0	0	1	1	B_i
Sum	1	1	1	0	S_i
Output carry	0	0	1	1	C_{i+1}

- The bits are added with full adders, starting from the least significant position (subscript 0), to form the sum bit and carry bit. The input carry C_0 in the least significant position must be 0.
- The value of C_{i+1} in a given significant position is the output carry of the full adder. This value is transferred into the input carry of the full adder that adds the bits one higher significant position to the left.
- The sum bits are thus generated starting from the rightmost position and are available as soon as the corresponding previous carry bit is generated. All the carries must be generated for the correct sum bits to appear at the outputs.



Four Bit Binary Adder

<u>HALF SUBTRACTOR:-</u>

- This circuit needs two binary inputs and two binary outputs.
- Symbols x and y are assigned to the two inputs and D (for difference) and B (for borrow) to the outputs.
- The truth table for the half subtractor is listed in the below table.

x	y	D	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

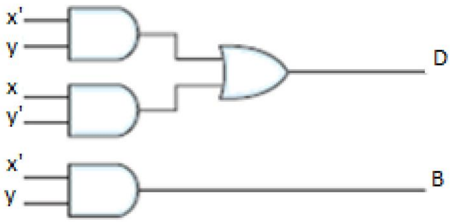
Truth Table

- The B output is 1 only when the inputs are 0 and 1. The D output represents the least significant bit of the subtraction.
- The subtraction operation is done by using the following rules as

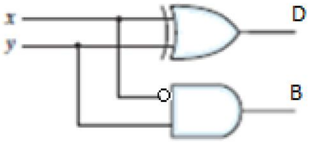
  0-0=0;
  0-1=1 with borrow 1;
  1-0=1;
  1-1=0.

- The simplified Boolean functions for the two outputs can be obtained directly from the truth table. The simplified sum-of-products expressions are

  D = x’y + xy’ and B =x’y



D = x'y+xy'
B = x'y



D = x ⊕ y
B=x'y

- The logic diagram of the half adder implemented in sum of products is shown in the figure. It can be also implemented with an exclusive-OR and an AND gate with one inverted input.

FULL SUBTRACTOR:-

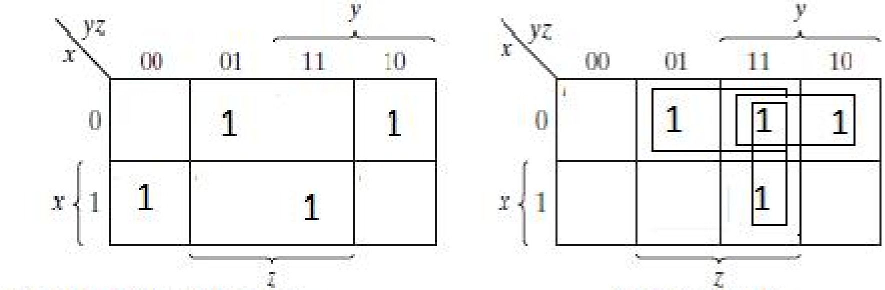
- A full subtractor is a combinational circuit that forms the arithmetic subtraction operation of three bits.
- It consists of three inputs and two outputs. Two of the input variables, denoted by x and y , represent the two significant bits to be subtracted. The third input, z , is subtracted from the result 0f the first subtraction.

x	y	z	D	B
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Truth Table

- Two outputs are necessary because the arithmetic subtraction of three binary digits ranges in value from 0 to 3, and binary representation of 2 or 3 needs two bits. The two outputs are designated by the symbols D for difference and B for borrow.

- The binary variable D gives the value of the least significant bit of the difference. The binary variable B gives the output borrow formed during the subtraction process.



D = x'y'z+x'yz'+xy'z'+xyz

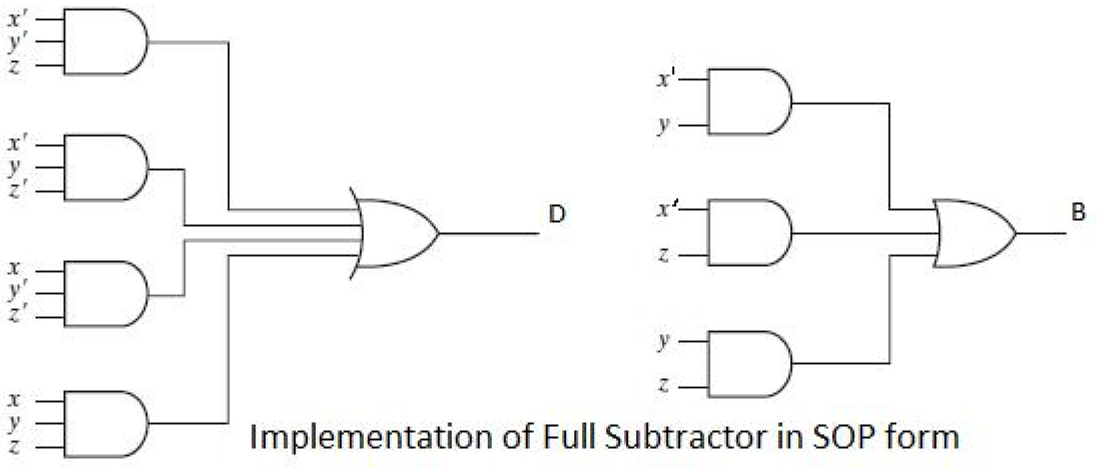
B = x'z+x'y+yz

K-Map for full Subtractor

- The eight rows under the input variables designate all possible combinations of the three variables. The output variables are determined from the arithmetic subtraction of the input bits.
- The difference D becomes 1 when any one of the input is 1or all three inputs are equal to1 and the borrow B is 1 when the input combination is (0 0 1) or (0 1 0) or (0 1 1) or (1 1 1).
- The simplified expressions are

D = x'y'z + x'yz' + xy'z' + xyz
B = x'z + x'y + yz

- The logic diagram for the full adder implemented in sum-of-products form is shown in figure.



Implementation of Full Subtractor in SOP form

MAGNITUDE COMPARATOR:-

- A magnitude comparator is a combinational circuit that compares two numbers A and B and determines their relative magnitudes.
- The following description is about a 2-bit magnitude comparator circuit.
- The outcome of the comparison is specified by three binary variables that indicate whether A < B, A = B, or A > B.
- Consider two numbers, A and B, with two digits each. Now writing the coefficients of the numbers in descending order of significance:

$$A = A_1\ A_0$$
$$B = B_1\ B_0$$

- The two numbers are equal if all pairs of significant digits are equal i.e. if and only if A1 = B1, and A0 = B0.
- When the numbers are binary, the digits are either 1 or 0, and the equality of each pair of bits can be expressed logically with an exclusive-NOR function as

$x1=A_1B_1+A_1'B_1'$

And $x0=A_0B_0+A_0'B_0'$

- The equality of the two numbers A and B is displayed in a combinational circuit by an output binary variable that we designate by the symbol (A = B).
- This binary variable is equal to 1 if the input numbers, A and B , are equal, and is equal to 0 otherwise.
- For equality to exist, all xi variables must be equal to 1, a condition that dictates an AND operation of all variables:

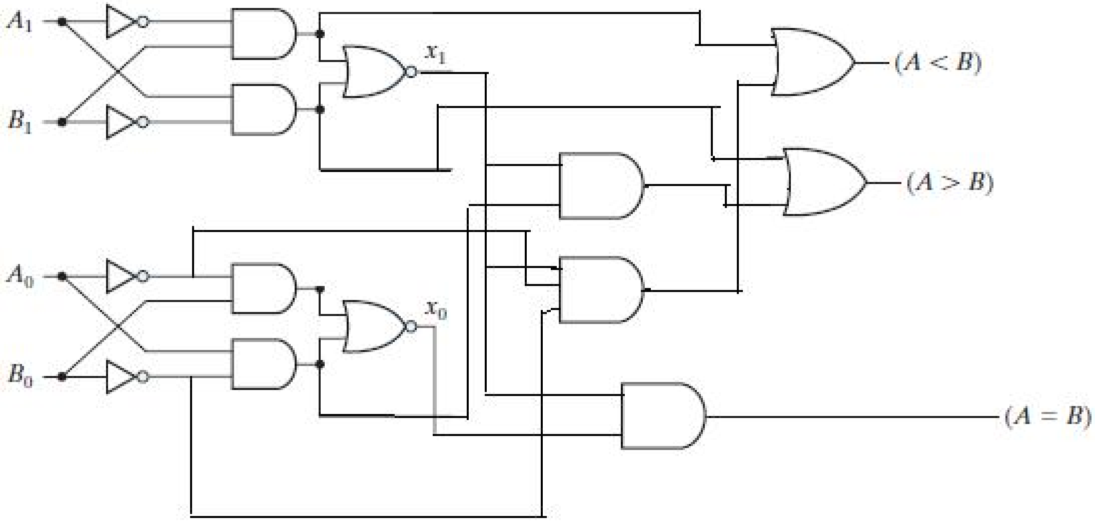
$$(A = B) = x_1x_0$$

- The binary variable (A = B) is equal to 1 only if all pairs of digits of the two numbers are equal.
- To determine whether A is greater or less than B, we inspect the relative magnitudes of pairs of significant digits, starting from the most significant position. If the two digits of a pair are equal, we compare the next lower significant pair of digits. If the corresponding digit of A is 1 and that of B is 0, we conclude that A > B. If the corresponding digit of A is 0 and that of B is 1, we have A < B. The sequential comparison can be expressed logically by the two Boolean functions

$$(A > B) = A_1B_1'+x_1A_0B'_0$$
$$(A < B) = A_1'\ B_1\ +x_1A_0'B_0'$$

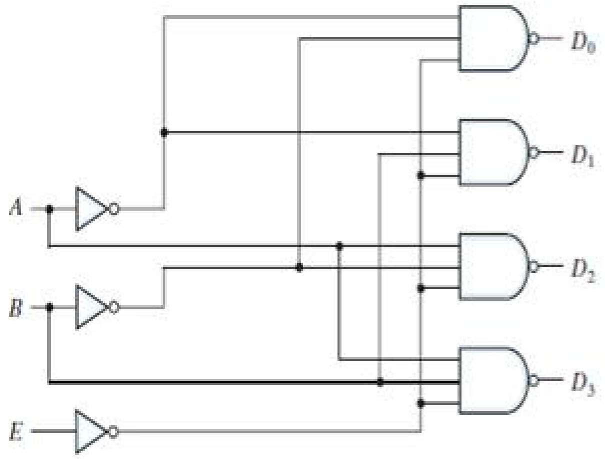
A_1	A_0	B_1	B_0	A>B	A<B	A=B
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	0	0	1
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	0	1

Truth Table



Logic Diagram of 2-bit Magnitude Comparator

DECODER:-

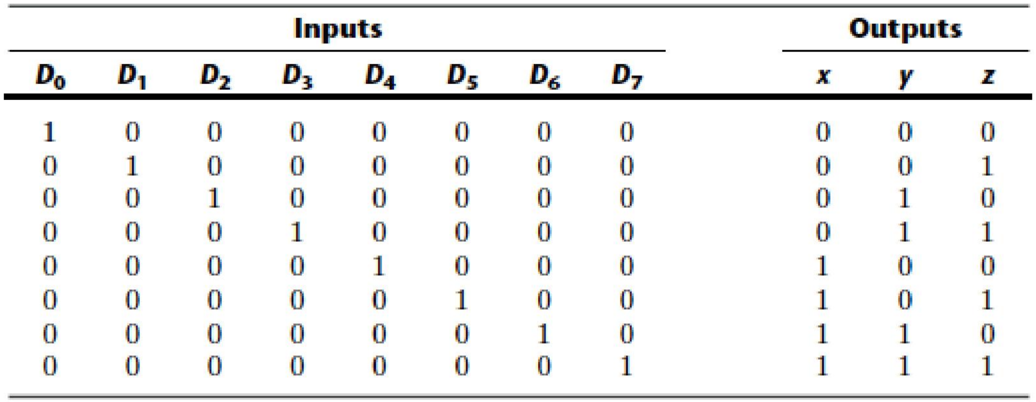


E	A	B	D_0	D_1	D_2	D_3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	1

- A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2^n unique output lines.
- If the n -bit coded information has unused combinations, the decoder may have fewer than 2n outputs.
- The decoders presented here are called n -to- m -line decoders, where m … 2n.
- Their purpose is to generate the 2n (or fewer) minterms of n input variables.
- Each combination of inputs will assert a unique output. The name decoder is also used in conjunction with other code converters, such as a BCD-to-seven-segment decoder.
- Consider the three-to-eight-line decoder circuit of three inputs are decoded into eight outputs, each representing one of the minterms of the three input variables.
- The three inverters provide the complement of the inputs, and each one of the eight AND gates generates one of the minterms.
- The input variables represent a binary number, and the outputs represent the eight digits of a number in the octal number system.
- However, a three-to-eight-line decoder can be used for decoding any three-bit code to provide eight outputs, one for each element of the code.
- A two-to-four-line decoder with an enable input constructed with NAND gates is shown in Fig.
- The circuit operates with complemented outputs and a complement enable input. The decoder is enabled when E is equal to 0 (i.e., active-low enable). As indicated by the truth table, only one output can be equal to 0 at any given time; all other outputs are equal to 1.
- The output whose value is equal to 0 represents the minterm selected by inputs A and B.
- The circuit is disabled when E is equal to 1, regardless of the values of the other two inputs.
- When the circuit is disabled, none of the outputs are equal to 0 and none of the minterms are selected.
- In general, a decoder may operate with complemented or un-complemented outputs.
- The enable input may be activated with a 0 or with a 1 signal.
- Some decoders have two or more enable inputs that must satisfy a given logic condition in order to enable the circuit.
- A decoder with enable input can function as a demultiplexer— a circuit that receives information from a single line and directs it to one of 2n possible output lines.
- The selection of a specific output is controlled by the bit combination of n selection lines.
- The decoder of Fig. can function as a one-to-four-line demultiplexer when E is taken as a data input line and A and B are taken as the selection inputs.
- The single input variable E has a path to all four outputs, but the input information is directed to only one of the output lines, as specified by the binary combination of the two selection lines A and B .
- This feature can be verified from the truth table of the circuit.
- For example, if the selection lines AB = 10, output D_2 will be the same as the input value E, while all other outputs are maintained at 1.
- Since decoder and demultiplexer operations are obtained from the same circuit, a decoder with an enable input is referred to as a decoder – demultiplexer.
- A application of this decoder is binary-to-octal conversion.

ENCODER:-

- An encoder is a digital circuit that performs the inverse operation of a decoder.
- An encoder has 2n (or fewer) input lines and n output lines.
- The output lines, as an aggregate, generate the binary code corresponding to the input value.



Inputs								Outputs		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

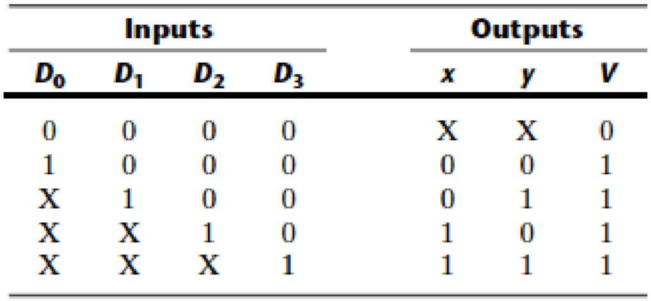
- The above Encoder has eight inputs (one for each of the octal digits) and three outputs that generate the corresponding binary number.
- It is assumed that only one input has a value of 1 at any given time.
- The encoder can be implemented with OR gates whose inputs are determined directly from the truth table.
- Output z is equal to 1 when the input octal digit is 1, 3, 5, or 7.
- Output y is 1 for octal digits 2, 3, 6, or 7, and output x is 1 for digits 4, 5, 6, or 7.
- These conditions can be expressed by the following Boolean output functions:

$$z = D_1 + D_3 + D_5 + D_7$$
$$y = D_2 + D_3 + D_6 + D_7$$
$$x = D_4 + D_5 + D_6 + D_7$$

- The encoder can be implemented with three OR gates.
- The encoder defined above has the limitation that only one input can be active at any given time.
- If two inputs are active simultaneously, the output produces an undefined combination.
- To resolve this ambiguity, encoder circuits must establish an input priority to ensure that only one input is encoded which is done in the Priority Encoder .

PRIORITY ENCODER:-

- A priority encoder is an encoder circuit that includes the priority function.
- The operation of the priority encoder is such that if two or more inputs are equal to 1 at the same time, the input having the highest priority will take precedence.



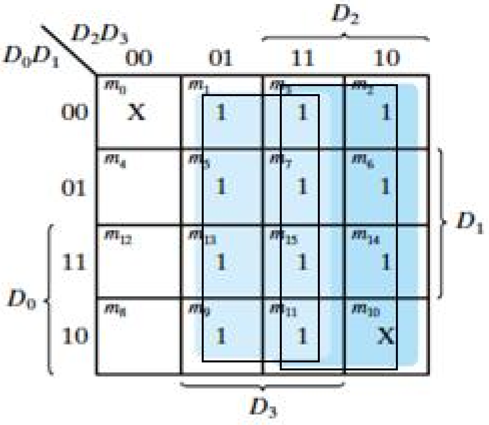
Inputs				Outputs		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

- In addition to the two outputs x and y , the circuit has a third output designated by V ; this is a valid bit indicator that is set to 1 when one or more inputs are equal to 1.

- If all inputs are 0, there is no valid input and V is equal to 0.
- The other two outputs are not inspected when V equals 0 and are specified as don't-care conditions.
- Here X 's in output columns represent don't-care conditions, the X 's in the input columns are useful for representing a truth table in condensed form.

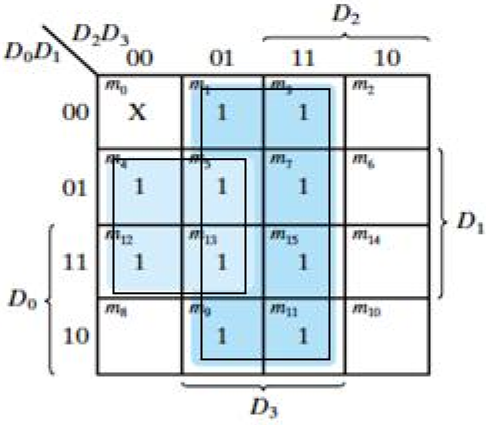
Inputs				Outputs		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

- Higher the subscript number, the higher the priority of the input.
- Input D3 has the highest priority, so, regardless of the values of the other inputs, when this input is 1, the output for xy is 11 (binary 3).
- If D2 = 1, provided that D3 = 0, regardless of the values of the other two lower priority inputs the output is 10.
- The output for D1 is generated only if higher priority inputs are 0, and so on down the priority levels.



D_0D_1 \ D_2D_3	00	01	11	10
00	m_0 X	m_1 1	m_3 1	m_2 1
01	m_4	m_5 1	m_7 1	m_6 1
11	m_{12}	m_{13} 1	m_{15} 1	m_{14} 1
10	m_8	m_9 1	m_{11} 1	m_{10} X

$x = D_2 + D_3$

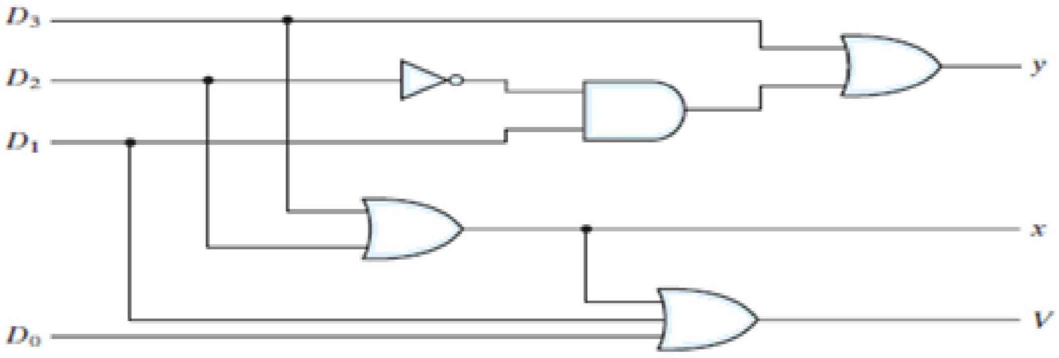


D_0D_1 \ D_2D_3	00	01	11	10
00	m_0 X	m_1 1	m_3 1	m_2
01	m_4 1	m_5 1	m_7 1	m_6
11	m_{12} 1	m_{13} 1	m_{15} 1	m_{14}
10	m_8	m_9 1	m_{11} 1	m_{10}

$y = D_3 + D_1D'_2$

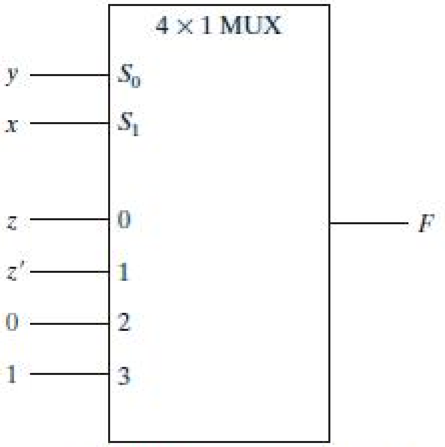
- The maps for simplifying outputs x and y are shown in above Fig.
- The minterms for the two functions are derived from its truth table.
- Although the table has only five rows, when each X in a row is replaced first by 0 and then by 1, we obtain all 16 possible input combinations.
- For example, the fourth row in the table, with inputs XX10, represents the four minterms 0010, 0110, 1010, and 1110. The simplified Boolean expressions for the priority encoder are obtained from the maps.
- The condition for output V is an OR function of all the input variables.
- The priority encoder is implemented according to the following Boolean functions:

$$x = D_2 + D_3$$
$$y = D_3 + D_1 D'_2$$
$$V = D_0 + D_1 + D_2 + D_3$$

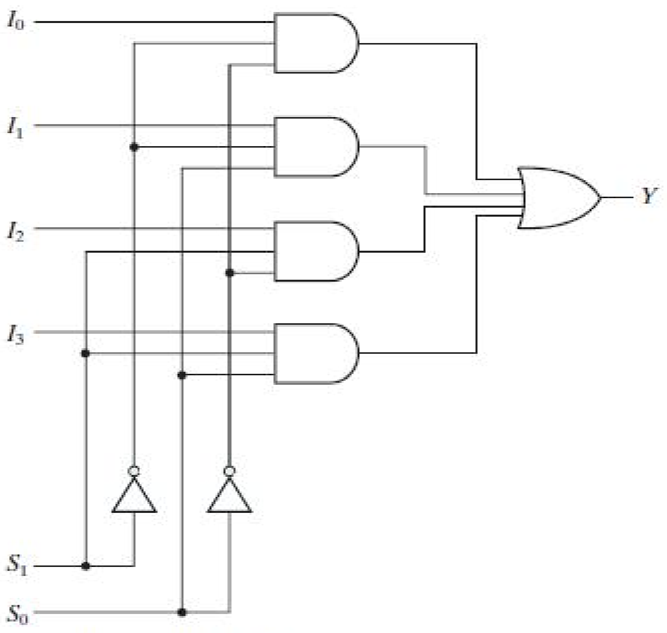


MULTIPLEXER:-

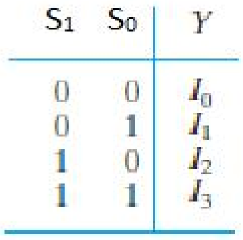
- A multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line.
- The selection of a particular input line is controlled by a set of selection lines.
- Normally, there are 2^n input lines and n selection lines whose bit combinations determine which input is selected.
- A four-to-one-line multiplexer is shown in the below figure. Each of the four inputs, I_0 through I_3, is applied to one input of an AND gate.
- Selection lines S_1 and S_0 are decoded to select a particular AND gate. The outputs of the AND gates are applied to a single OR gate that provides the one-line output.
- The function table lists the input that is passed to the output for each combination of the binary selection values.
- To demonstrate the operation of the circuit, consider the case when
  S_1S_0= 10.
- The AND gate associated with input I_2 has two of its inputs equal to 1 and the third input connected to I_2.
- The other three AND gates have at least one input equal to 0, which makes their outputs equal to 0. The output of the OR gate is now equal to the value of I_2, providing a path from the selected input to the output.
- A multiplexer is also called a data selector, since it selects one of many inputs and steers the binary information to the output line.



(b) Multiplexer implementation



Logic diagram



S1	S0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

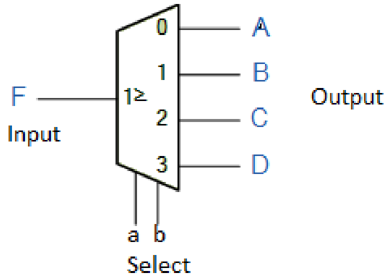
Truth table

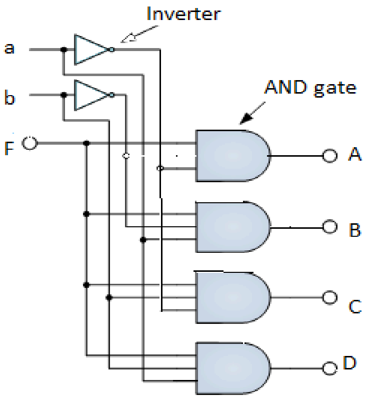
DEMULTIPLEXER:-

- The data distributor, known more commonly as a Demultiplexer or “Demux” for short, is the exact opposite of the Multiplexer.
- The demultiplexer takes one single input data line and then switches it to any one of a number of individual output lines one at a time. The demultiplexer converts a serial data signal at the input to a parallel data at its output lines as shown below.
- The Boolean expression for this 1-to-4 demultiplexer above with outputs A to D and data select lines a, b is given as:

$$F = (ab)'A + a'bB + ab'C + abD$$

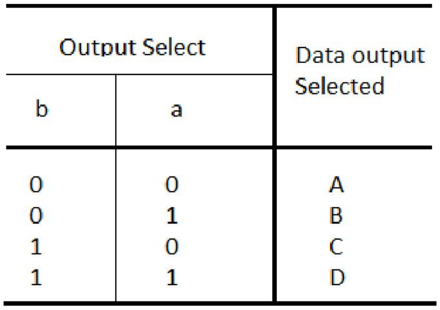
- The function of the demultiplexer is to switch one common data input line to any one of the 4 output data lines A to D in our example above. As with the multiplexer the individual solid state switches are selected by the binary input address code on the output select pins “a” and “b” as shown.





Logic Diagram

- Unlike multiplexers which convert data from a single data line to multiple lines and demultiplexers which convert multiple lines to a single data line, there are devices available which convert data to and from multiple lines and in the next tutorial about combinational logic devices.
- Standard demultiplexer IC packages available are the TTL 74LS138 1 to 8-output demultiplexer, the TTL 74LS139 Dual 1-to-4 output demultiplexer or the CMOS CD4514 1-to-16 output demultiplexer.



Output Select		Data output Selected
b	a	
0	0	A
0	1	B
1	0	C
1	1	D

Truth Table

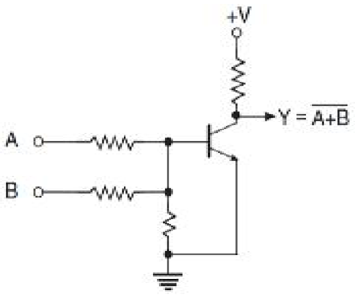
LOGIC FAMILIES

- A circuit configuration or approach used to produce a type of digital integrated circuit is called Logic Family.
- By using logic families we can generate different logic functions, when fabricated in the form of an IC with the same approach, or in other words belonging to the same logic family, will have identical electrical characteristics.
- The set of digital ICs belonging to the same logic family are electrically compatible with each other.
- Some common Characteristics of the Same Logic Family include Supply voltage range, speed of response, power dissipation, input and output logic levels, current sourcing and sinking capability, fan-out, noise margin, etc.
- Choosing digital ICs from the same logic family guarantees that these ICs are compatible with respect to each other and that the system as a whole performs the intended logic function.

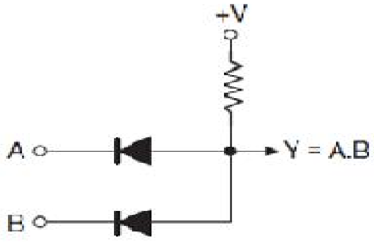
TYPES OF LOGIC FAMILY:-

- The entire range of digital ICs is fabricated using either bipolar devices or MOS devices or a combination of the two.
- Bipolar families include:-

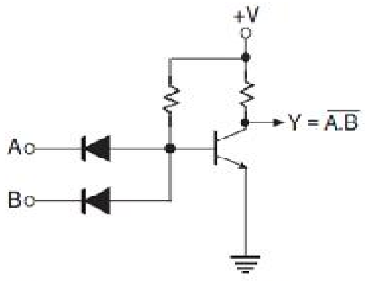
  Diode logic (DL)
  Resistor-Transistor logic (RTL)
  Diode-transistor logic (DTL)
  Transistor- Transistor logic (TTL)
  Emitter Coupled Logic (ECL),
  (also known as Current Mode Logic(CML))
  Integrated Injection logic (I2L)
- The Bi-MOS logic family uses both bipolar and MOS devices.



Resistor -Transistor logic (RTL)



Diode Logic (DL)



Diode-Transistor logic (DTL)

- Above are some example of DL, RTL and DTL.
- MOS families include:-

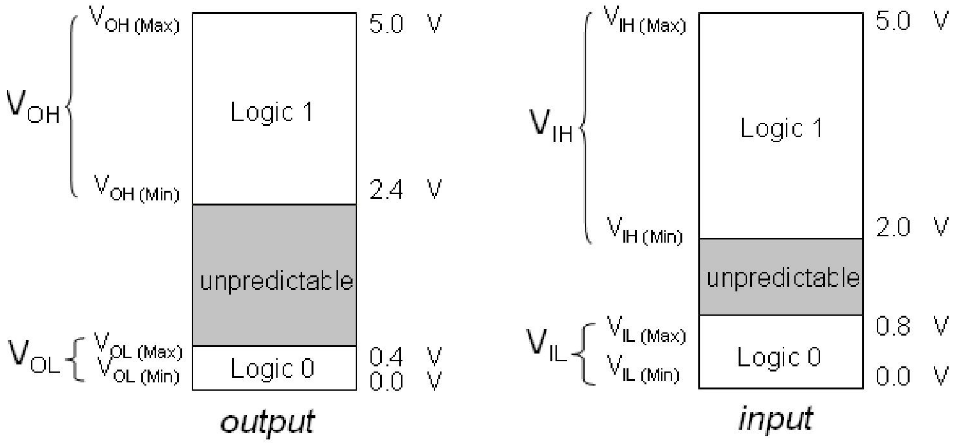
  ThePMOS family (using P-channel MOSFETs)
  The NMOS family (using N-channel MOSFETs)
  The CMOS family (using both N- and P-channel devices)

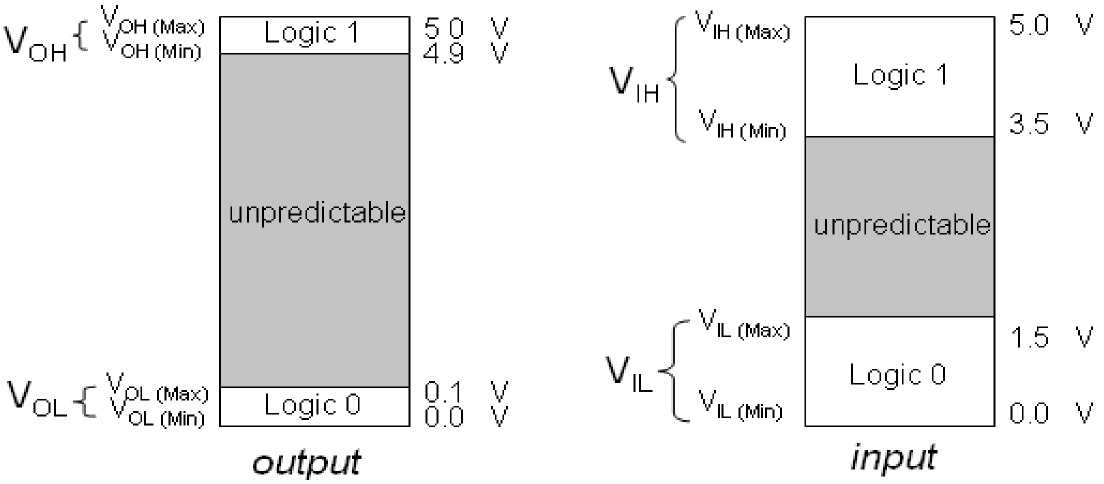
SOME OPERATIONAL PROPERTIES OF LOGIC FAMILY:-

DC Supply Voltage:-

- The nominal value of the dc supply voltage for TTL (transisitor-transistor logic) and CMOS (complementary metal-oxide semiconductor) devices is +5V. Although ommitted from logic diagrams for simplicity, this voltage is connected to Vcc or VDD pin of an IC package and ground is connected to the GND pin.

TTL Logic Levels





CMOS Logic Levels

Noise Immunity:-

- Noise is the unwanted voltage that is induced in electrical circuits and can present a threat to the poor operation of the circuit. In order not to be adversely effected by noise, a logic circuit must have a certain amount of 'noise immunity'.
- This is the ability to tolerate a certain amount of unwanted voltage fluctuation on its inputs without changing its output state is called Noise Immunity.

Noise Margin:-

- A measure of a circuit's noise immunity is called 'noise margin' which is expressed in volts.
- There are two values of noise margin specified for a given logic circuit: the HIGH (V_{NH}) and LOW (V_{NL}) noise margins.

These are defined by following equations :

$V_{NH} = V_{OH}$ (Min) - V_{IH} (Min) $V_{NL} = V_{IL}$ (Max) - V_{OL} (Max)

Power Dissipation:-

- A logic gate draws ICCH current from the supply when the gate is in the HIGH output state, draws ICCL current from the supply in the LOW output state.
- Average power is

  PD = VCC ICC where ICC = (ICCH + ICCL) / 2

Propagation Delay time:-

- When a signal passes (propagates) through a logic circuit, it always experiences a time delay as shown below. A change in the output level always occurs a short time, called 'propagation delay time' , later than the change in the input level that caused it.

Fan Out of Gates:-

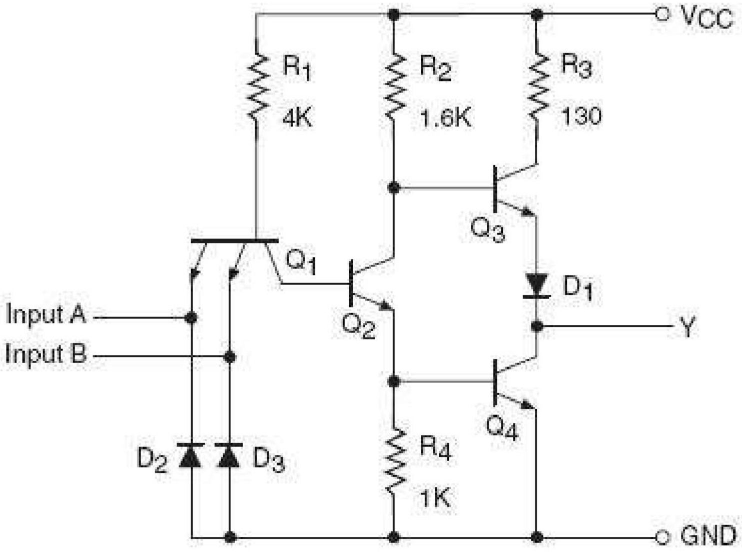
- When the output of a logic gate is connected to one or more inputs of other gates, a load on the driving gate is created. There is a limit to the number of load gates that a given gate can drive. This limit is called the 'Fan-Out' of the gate.

TRANSISTOR-TRANSISTOR LOGIC:-

- In Transistor-Transistor logic or just TTL, logic gates are built only around transistors.
- TTL was developed in 1965. Through the years basic TTL has been improved to meet performance requirements. There are many versions or families of TTL.
- For example
  - Standard TTL
  - High Speed TTL (twice as fast, twice as much power)
  - Low Power TTL (1/10 the speed, 1/10 the power of “standard" TTL)
  - Schhottky TTL etc. (for high-frequency uses)
- All TTL logic families have three configurations for outputs
  1. Totem pole output
  2. Open collector output
  3. Tristate output

Totem pole output:-

- Addition of an active pull up circuit in the output of a gate is called totem pole.
- To increase the switching speed of the gate which is limited due to the parasitic capacitance at the output totem pole is used.
- The circuit of a totem-pole NAND gate is shown below, which has got three stages
  1. Input Stage
  2. Phase Splitter Stage
  3. Output Stage



- Transistor Q1 is a two-emitter NPN transistor, which is equivalent two NPN transistors with their base and emitter terminals tied together.
- The two emitters are the two inputs of the NAND gate

  In TTL technology multiple emitter transistors are used for the input devices

  Diodes D2 and D3 are protection diodes used to limit negative input voltages.
- When there is large negative voltage at input, the diode conducts and shorting it to the ground Q2 provides complementary voltages for the output transistors Q3 and Q4.
- The combination of Q3 and Q4 forms the output circuit often referred to as a totem pole arrangement (Q4 is stacked on top of Q3). In such an arrangement, either Q3 or Q4 conducts at a time depending upon the logic status of the inputs

  Diode D1 ensures that Q4 will turn off when Q2 is on (HIGH input)

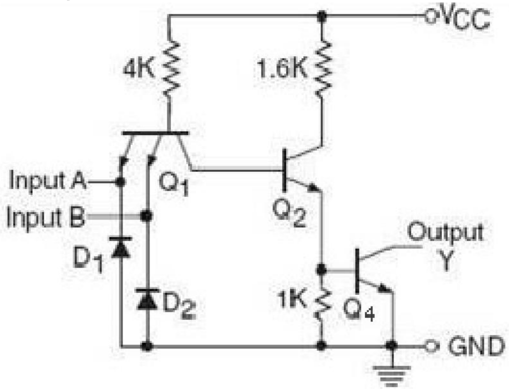
  The output Y is taken from the top of Q3

Advantages of Totem Pole Output:-

- The features of this arrangement are
  1. Low power consumption
  2. Fast switching
  3. Low output impedance

OPEN COLLECTOR OUTPUT:-

- Figure below shows the circuit of a typical TTL gate with open-collector output Observe here that the circuit elements associated with Q3 in the totem-pole circuit are missing and the collector of Q4 is left open-circuited, hence the name open-collector.



- An open-collector output can present a logic LOW output. Since there is no internal path from the output Y to the supply voltage V_{CC} , the circuit cannot present a logic HIGH on its own.

Advantages of Open Collector Outputs:-

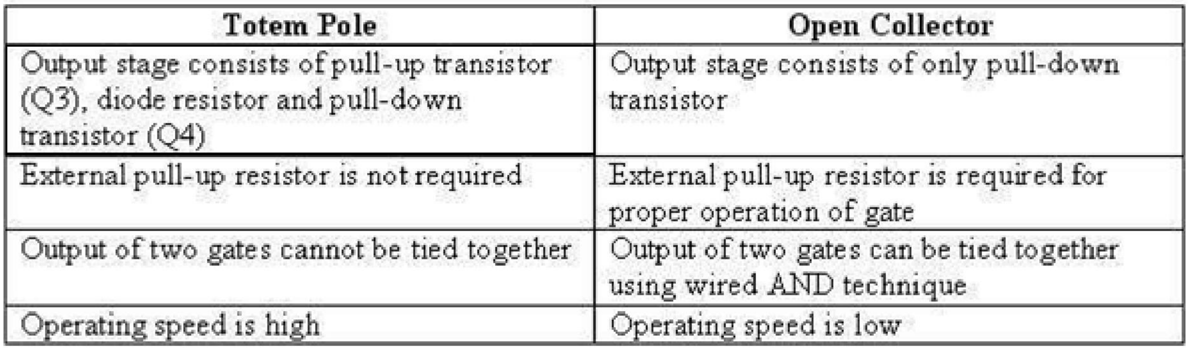
- Open-collector outputs can be tied directly together which results in the logical ANDing of the outputs. Thus the equivalent of an AND gate can be formed by simply connecting the outputs.
- Increased current levels - Standard TTL gates with totem-pole outputs can only provide a HIGH current output of 0.4 mA and a LOW current of 1.6 mA. Many open-collector gates have increased current ratings.
- Different voltage levels - A wide variety of output HIGH voltages can be achieved using open-collector gates. This is useful in interfacing different logic families that have different voltage and current level requirements.

Disadvantage of open-collector gates:-

- They have slow switching speed. This is because the value of pull-up resistor is in kW, which results in a relatively long time Constants

Comparison of Totem Pole and Open Collector Output:-

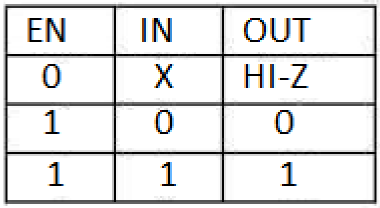
- The major advantage of using a totem-pole connection is that it offers low-output impedance in both the HIGH and LOW output states



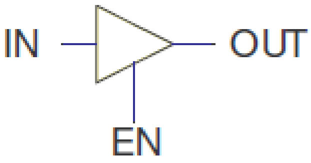
Totem Pole	Open Collector
Output stage consists of pull-up transistor (Q3), diode resistor and pull-down transistor (Q4)	Output stage consists of only pull-down transistor
External pull-up resistor is not required	External pull-up resistor is required for proper operation of gate
Output of two gates cannot be tied together	Output of two gates can be tied together using wired AND technique
Operating speed is high	Operating speed is low

TRISTATE (THREE-STATE) LOGIC OUPUT:-

- Tristate output combines the advantages of the totem-pole and open collector circuits.
- Three output states are HIGH, LOW, and high impedance (Hi-Z).

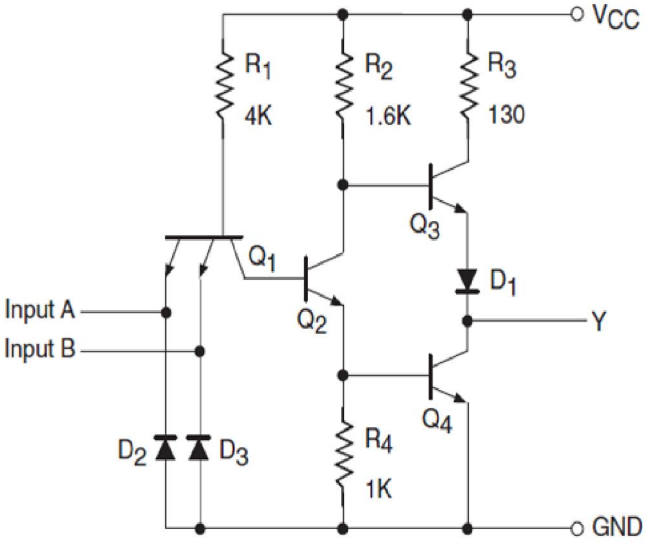


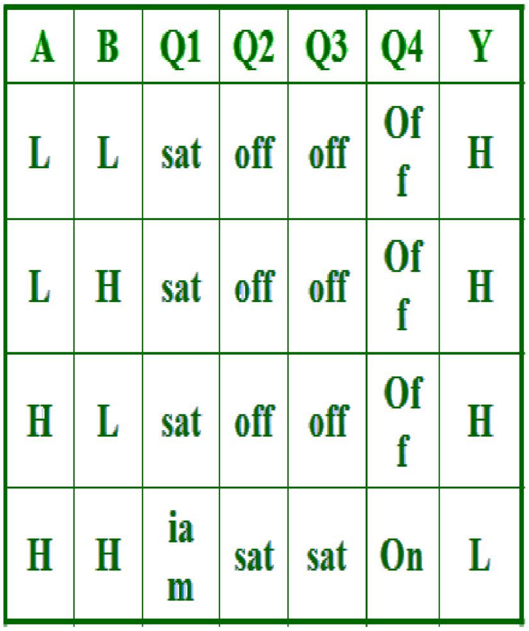
EN	IN	OUT
0	X	HI-Z
1	0	0
1	1	1



- For the symbol and truth table, IN is the data input, and EN, the additional enable input for control. For EN = 0, regardless of the value on IN(denoted by X), the output value is Hi-Z. For EN = 1, the output value follows the input value.
- Data input, IN, can be inverted. Control input, EN, can be inverted by addition of "bubbles" to signals IN OUT EN.
- This requires two inputs: input and enable EN is to make output Hi-Z or follow input.

STANDARD TTL NAND GATE:





A	B	Q1	Q2	Q3	Q4	Y
L	L	sat	off	off	Of f	H
L	H	sat	off	off	Of f	H
H	L	sat	off	off	Of f	H
H	H	ia m	sat	sat	On	L

CMOS TECHNOLOGY:-

- MOS stands for Metal Oxide Semiconductor and this technology uses FETs.
- MOS can be classified into three sub-families:

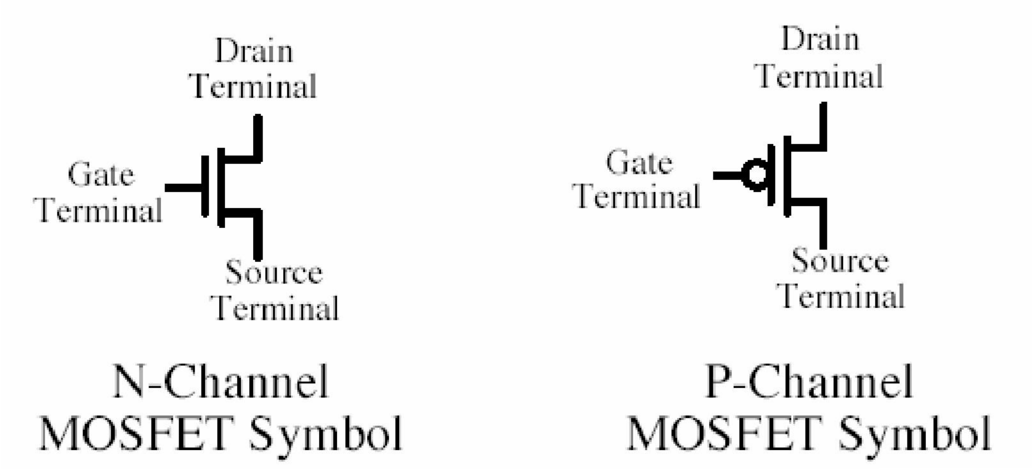
  PMOS (P-channel)
  NMOS (N-channel)
  CMOS (Complementary MOS, most common)
- The following simplified symbols are used to represent MOSFET transistors in most CMOS. The gate of a MOS transistor controls the flow of the current between the drain and the source. The MOS transistor can be viewed as a simple ON/OFF switch.

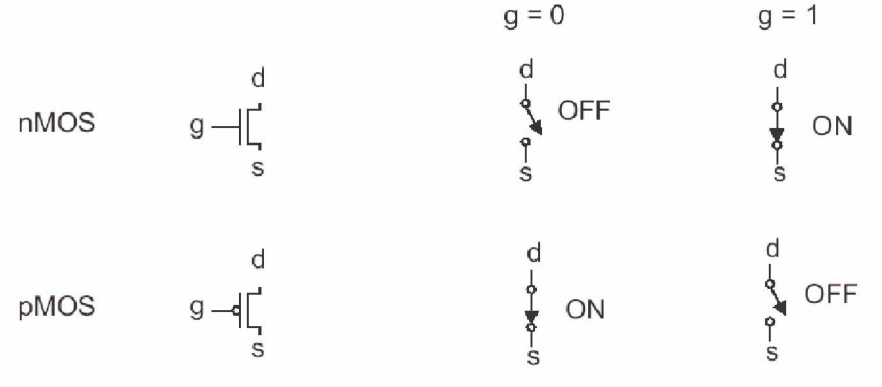
Advantages of MOS Digital ICs:-

- They are simple and inexpensive to fabricate.
- Can be used for Higher integration and consume little power.

Disadvantages of MOS Digital ICs:-

- There is possibility for Static-electricity damage.
- They are slower than TTL.



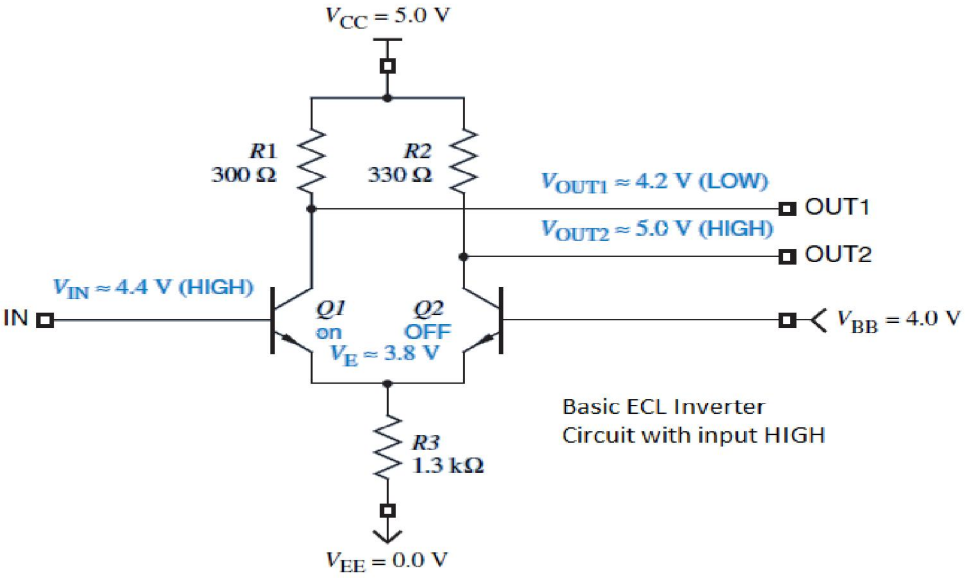


ECL: EMITTER-COUPLED LOGIC:-

- The key to reduce propagation delay in a bipolar logic family is to prevent a gate’s transistors from saturating. It is possible to prevent saturation by using a radically different circuit structure, called current-mode logic (CML) or emitter-coupled logic (ECL).
- Unlike the other logic families in this chapter, ECL does not produce a large voltage swing between the LOW and HIGH levels but it has a small voltage swing, less than a volt, and it internally switches current between two possible paths, depending on the output state.

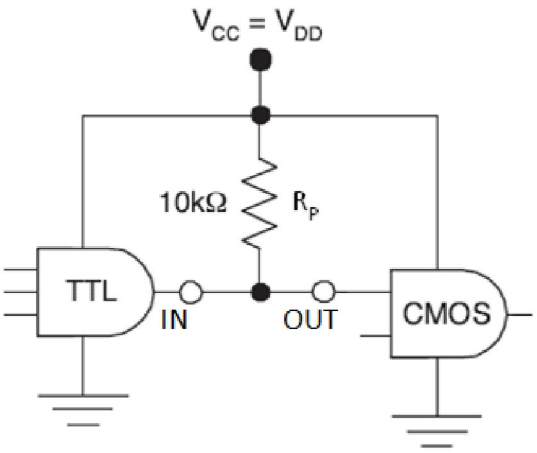
Basic ECL Circuit

- The basic idea of current-mode logic is illustrated by the inverter/buffer circuit in the figure. This circuit has both an inverting output (OUT1) and a non-inverting output (OUT2).
- Two transistors are connected as a differential amplifier with a common emitter resistor.
- The supply voltages for this example are VCC = 5.0, VBB = 4.0, and VEE = 0 V, and the input LOW and HIGH levels are defined to be 3.6 and 4.4 V. This circuit actually produces output LOW and HIGH levels that are 0.6 V higher (4.2 and 5.0 V).

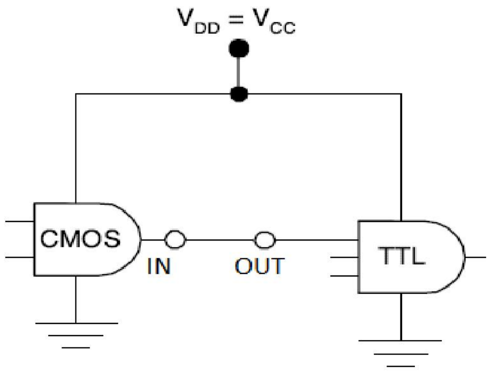


Basic ECL Inverter Circuit with input HIGH

INTERFACING OF TTL TO CMOS



INTERFACING OF CMOS TO TTL



TTL vs. CMOS:-

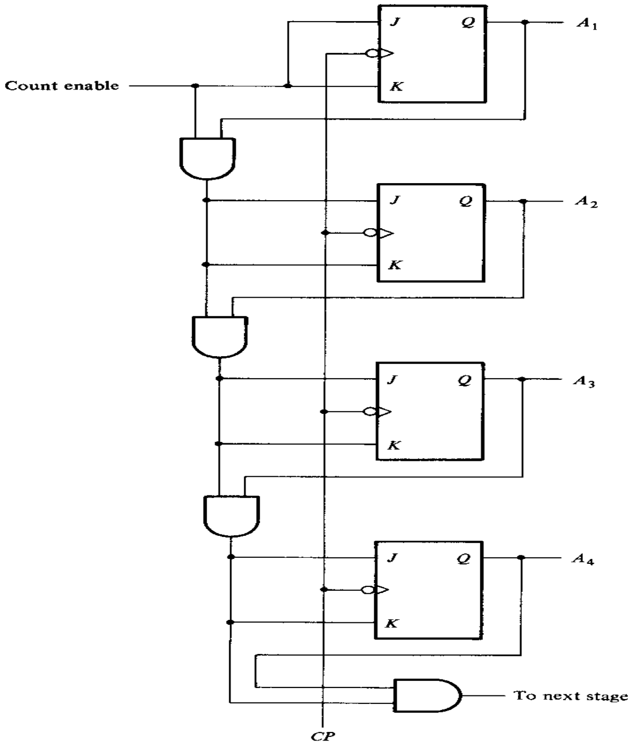
- TTL has less propagation delay than CMOS i.e. TTL is good where high speed is needed.
- And CMOS 4000 is good for Battery equipment and where speed is not so important.
- CMOS requires less power than TTL i.e. power dissipation and hence power consumption is less for CMOS.

COUNTER

- A counter is a device which stores (and sometimes displays) the number of times a particular event or process has occurred. In electronics, counters can be implemented quite easily using register-type circuits.
- There are different types of counters, viz.
  - Asynchronous (ripple) counter
  - Synchronous counter
  - Decade counter
  - Up/down counter
  - Ring counter
  - Johnson counter
  - Cascaded counter
  - Modulus counter.

Synchronous counter

- A 4-bit synchronous counter using JK flip-flops is shown in the figure.
- In synchronous counters, the clock inputs of all the flip-flops are connected together and are triggered by the input pulses. Thus, all the flip-flops change state simultaneously (in parallel).

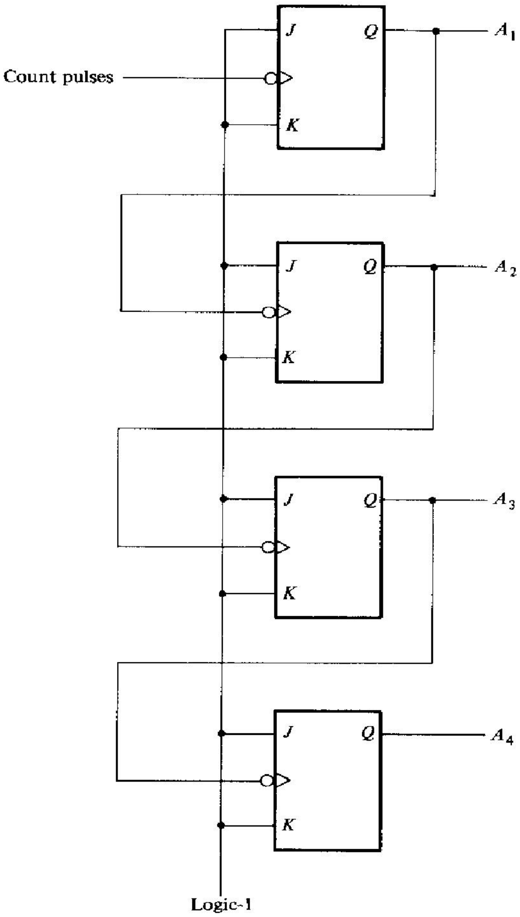


- The circuit below is a 4-bit synchronous counter.
- The J and K inputs of FF0 are connected to HIGH. FF1 has its J and K inputs connected to the output of FF0, and the J and K inputs of FF2 are connected to the output of an AND gate that is fed by the outputs of FF0 and FF1.
- A simple way of implementing the logic for each bit of an ascending counter (which is what is depicted in the image to the right) is for each bit to toggle when all of the less significant bits are at a logic high state.
- For example, bit 1 toggles when bit 0 is logic high; bit 2 toggles when both bit 1 and bit 0 are logic high; bit 3 toggles when bit 2, bit 1 and bit 0 are all high; and so on.

- Synchronous counters can also be implemented with hardware finite state machines, which are more complex but allow for smoother, more stable transitions.

Asynchronous Counter

- An asynchronous (ripple) counter is a single d-type flip-flop, with its J (data) input fed from its own inverted output.
- This circuit can store one bit, and hence can count from zero to one before it overflows (starts over from 0).



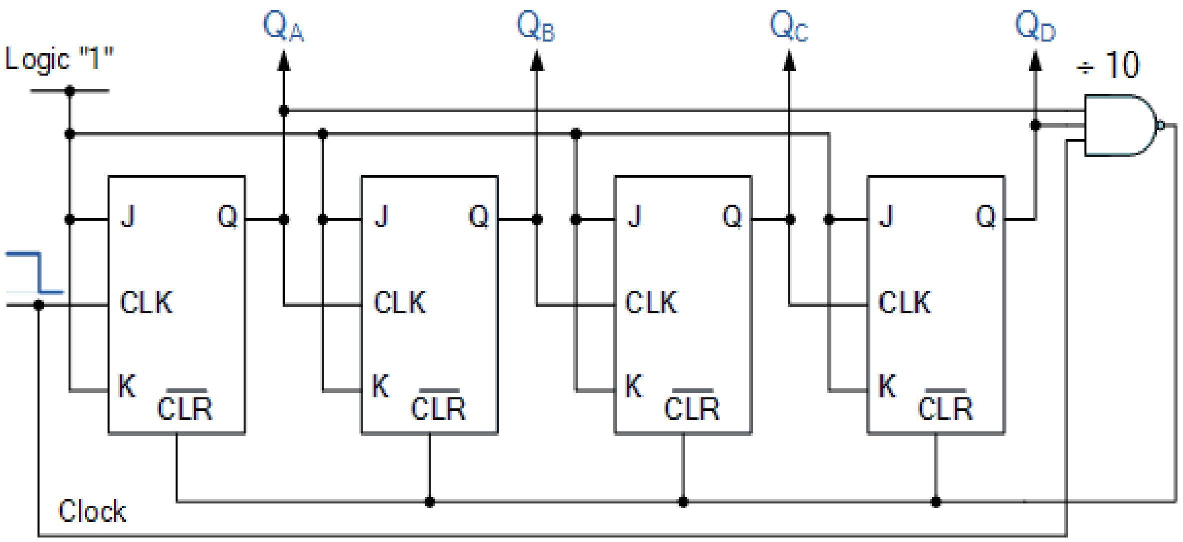
- This counter will increment once for every clock cycle and takes two clock cycles to overflow, so every cycle it will alternate between a transition from 0 to 1 and a transition from 1 to 0.
- This creates a new clock with a 50% duty cycle at exactly half the frequency of the input clock.
- If this output is then used as the clock signal for a similarly arranged D flip-flop, remembering to invert the output to the input, one will get another 1 bit counter that counts half as fast. These together yield a two-bit counter.
- Additional flip-flops can be added, by always inverting the output to its own input, and using the output from the previous flip-flop as the clock signal. The result is called a ripple counter, which can count to 2^n – 1, where n is the number of bits (flip-flop stages) in the counter.
- Ripple counters suffer from unstable outputs as the overflows "ripple" from stage to stage, but they find application as dividers for clock signals.

Modulus Counter

- A modulus counter is that which produces an output pulse after a certain number of input pulses is applied.
- In modulus counter the total count possible is based on the number of stages, i.e., digit positions.

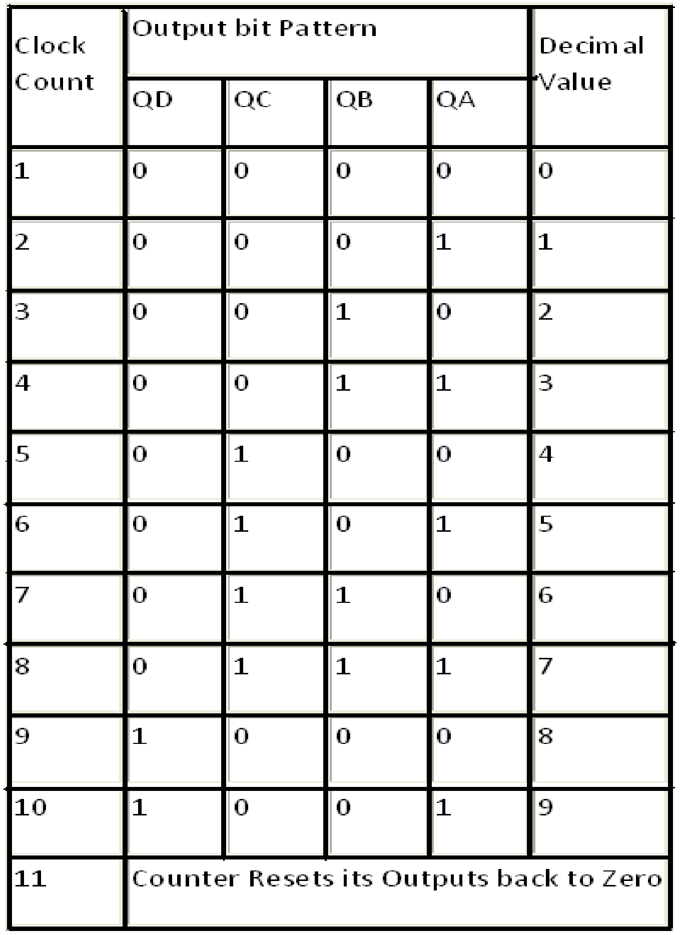
- Modulus counters are used in digital computers.
- A binary modulo-8 counter with three flip-flops, i.e., three stages, will produce an output pulse, i.e., display an output one-digit, after eight input pulses have been counted, i.e., entered or applied. This assumes that the counter started in the zero-condition.

Asynchronous Decade Counter



- A decade counter can count from BCD “0” to BCD “9”.
- A decade counter requires resetting to zero when the output count reaches the decimal value of 10, ie. when DCBA = 1010 and this condition is fed back to the reset input.
- A counter with a count sequence from binary “0000” (BCD = “0”) through to “1001” (BCD = “9”) is generally referred to as a BCD binary-coded-decimal counter because its ten state sequence is that of a BCD code but binary decade counters are more common.
- This type of asynchronous counter counts upwards on each leading edge of the input clock signal starting from 0000 until it reaches an output 1001 (decimal 9).
- Both outputs Q_A and Q_D are now equal to logic “1” and the output from the NAND gate changes state from logic “1” to a logic “0” level and whose output is also connected to the CLEAR (CLR) inputs of all the J-K Flip-flops.
- This signal causes all of the Q outputs to be reset back to binary 0000 on the count of 10. Once QA and QD are both equal to logic “0” the output of the NAND gate returns back to a logic level “1” and the counter restarts again from 0000. We now have a decade or Modulo-10 counter.

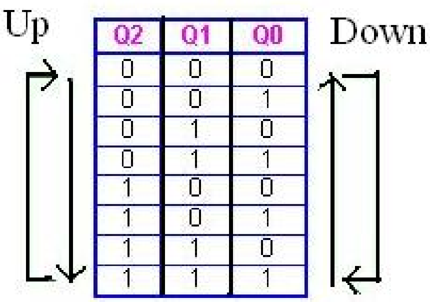
Decade Counter Truth Table



Clock Count	Output bit Pattern				Decimal Value
	QD	QC	QB	QA	
1	0	0	0	0	0
2	0	0	0	1	1
3	0	0	1	0	2
4	0	0	1	1	3
5	0	1	0	0	4
6	0	1	0	1	5
7	0	1	1	0	6
8	0	1	1	1	7
9	1	0	0	0	8
10	1	0	0	1	9
11	Counter Resets its Outputs back to Zero				

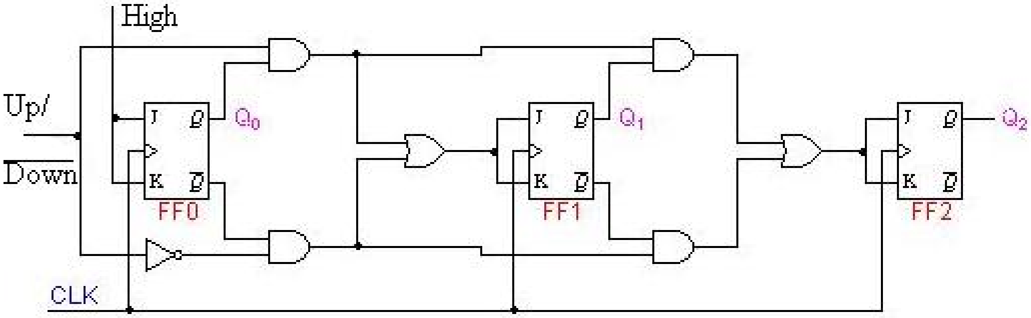
Up/Down Counter

- In a synchronous up-down binary counter the flip-flop in the lowest-order position is complemented with every pulse.
- A flip-flop in any other position is complemented with a pulse, provided all the lower-order pulse equal to 0.
- Up/Down counter is used to control the direction of the counter through a certain sequence.



Q2	Q1	Q0
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

- From the sequence table we can observe that:
  - For both the UP and DOWN sequences, Q_0 toggles on each clock pulse.
  - For the UP sequence, Q_1 changes state on the next clock pulse when $Q_0=1$.
  - For the DOWN sequence, Q_1 changes state on the next clock pulse when $Q_0=0$.
  - For the UP sequence, Q_2 changes state on the next clock pulse when $Q_0=Q_1=1$.
  - For the DOWN sequence, Q_2 changes state on the next clock pulse when $Q_0=Q_1=0$.



- These characteristics are implemented with the AND, OR & NOT logic connected as shown in the logic diagram above.

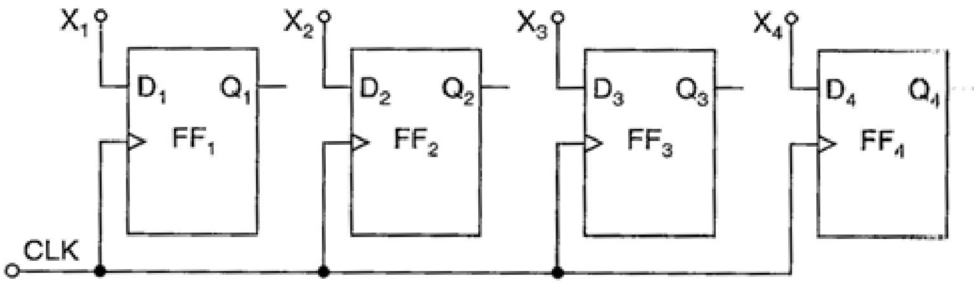
REGISTERS

INTRODUCTION:-

- The sequential circuits known as register are very important logical block in most of the digital systems.
- Registers are used for storage and transfer of binary information in a digital system.
- A register is mostly used for the purpose of storing and shifting binary data entered into it from an external source and has no characteristics internal sequence of states.
- The storage capacity of a register is defined as the number of bits of digital data, it can store or retain.
- These registers are normally used for temporary storage of data.

BUFFER REGISTER:-

- These are the simplest registers and are used for simply storing a binary word.
- These may be controlled by Controlled Buffer Register.
- D flip – flops are used for constructing a buffer register or other flip- flop can be used.
- The figure shown below is a 4- bit buffer register.



Logic diagram of a 4-bit buffer register.

- The binary word to be stored is applied to the data terminals.
- When the clock pulse is applied, the output word becomes the same as the word applied at the input terminals, i.e. the input word is loaded into the register by the application of clock pulse.
- When the positive clock edge arrives, the stored word becomes:

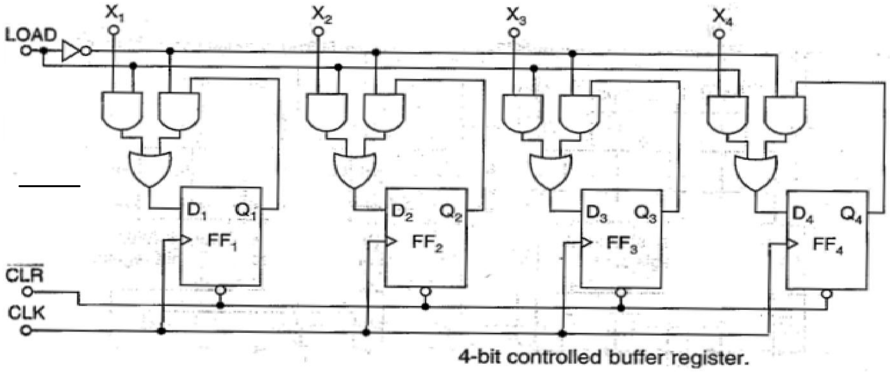
  Q4 Q3 Q2 Q1= X4 X3 X2 X1

  or Q = X .

  This circuit is too primitive to be of any use.

CONTROLLED BUFFER REGISTER:-

- The figure shows a controlled buffer register.



4-bit controlled buffer register.

- If $\overline{\text{CLR}}$ goes LOW, all the flip-flops are RESET and the output becomes, Q = 0000.
- When $\overline{\text{CLR}}$ is HIGH, the register is ready for action

- LOAD is control input.
- When LOAD is HIGH, the data bits X can reach the D inputs of FFs.
- At the positive going edge of the next clock pulse, the register is loaded, i.e.

  Q4 Q3 Q2 Q1= X4 X3 X2 X1
  or Q = X .

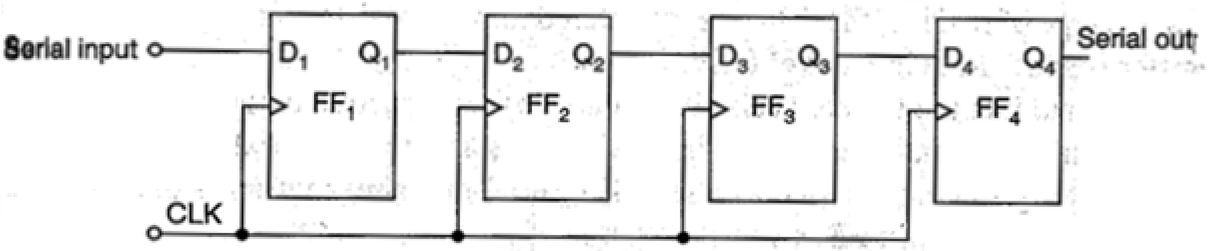
- When LOAD is LOW, the X bits cannot reach the FFs. At the same time the inverted signal $\overline{\text{LOAD}}$ is HIGH. This forces each flip-flop output to feedback to its data input.
- Therefore data is circulated or retained as each clock pulse arrives.
- In other words the content register remains unchanged in spite of the clock pulses.
- Longer buffer registers can built by adding more FFs.

CONTROLLED BUFFER REGISTER:-

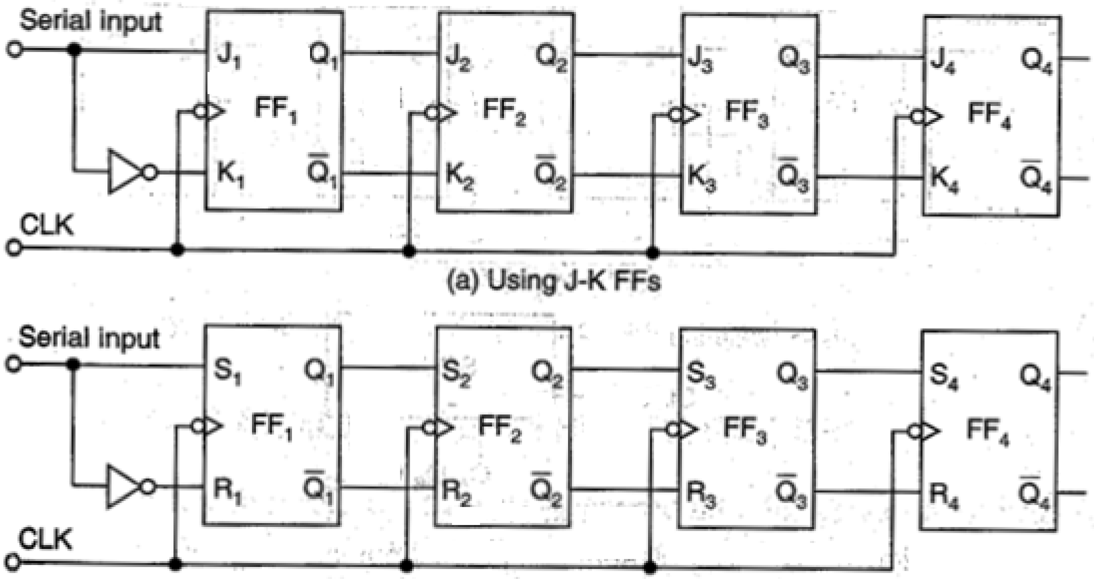
- A number of FFs connected together such that data may be shifted into and shifted out of them is called a shift register.
- Data may be shifted into or out of the register either in serial form or in parallel form.
- There are four basic types of shift registers
  1. Serial in, serial out
  2. Serial in, parallel out
  3. Parallel in, serial out
  4. Parallel in , parallel out

SERIAL IN, SERIAL OUT SHIFT REGISTER:-

- This type of shift register accepts data serially, i.e., one bit at a time and also outputs data serially.
- The logic diagram of a four bit serial in, serial out shift register is shown in below figure:
- In 4 stages i.e. with 4 FFs, the register can store upto 4 bits of data.
- Serial data is applied at the D input of the first FF. The Q output of the first FF is connected to the D input of the second FF, the output of the second FF is connected to the D input of the third FF and the Q output of the third FF is connected to the D input of the fourth FF. The data is outputted from the Q terminal of the last FF.
- When a serial data is transferred to a register, each new bit is clocked into the first FF at the positive going edge of each clock pulse.
- The bit that is previously stored by the first FF is transferred to the second FF.
- The bit that is stored by the second FF is transferred to the third FF, and so on.
- The bit that was stored by the last FF is shifted out.
- A shift register can also be constructed using J-K FFs or S-R FFs as shown in the figure below.



4-bit serial-in, serial-out, shift-right, shift register.



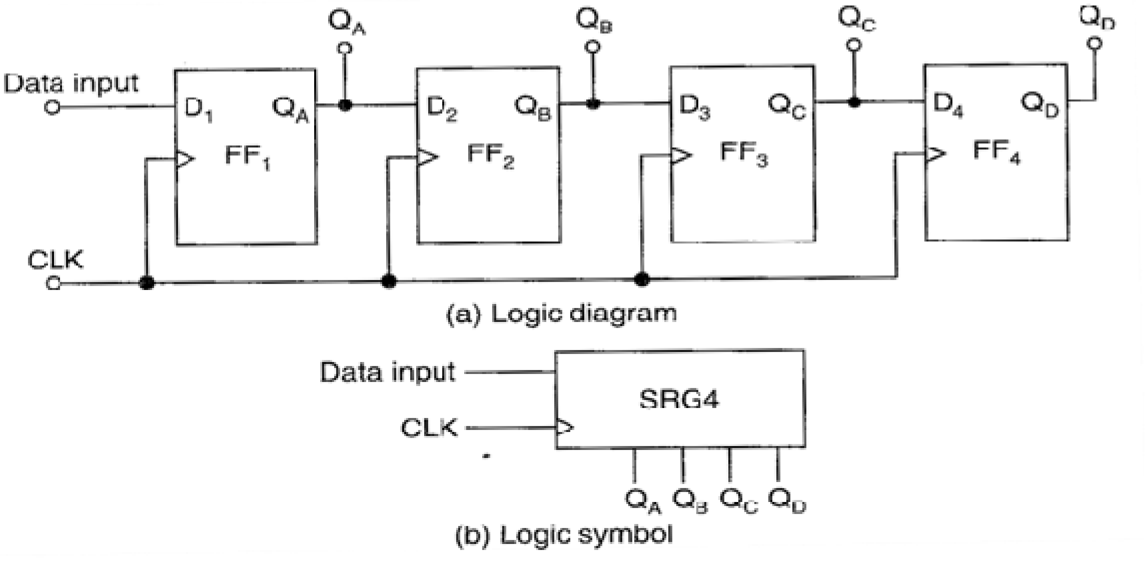
(a) Using J-K FFs

(b) Using S-R FFs

A 4-bit serial-in, serial-out, shift register.

SERIAL IN, PARALLEL OUT SHIFT REGISTER:-

- In this type of register, the data bits are entered into the register serially, but the data stored in the register serially, but the stored in the register is shifted out in the parallel form.
- When the data bits are stored once, each bits appears on its respective output line and all bits are available simultaneously, rather than bit – by – bit basis as in the serial output.
- The serial in, parallel out shift register can be used as a serial in, serial out shift register if the output is taken from the Q terminal of the last FF.
- The logic diagram and logic symbol of a 4 bit serial in, parallel out shift register is given below.



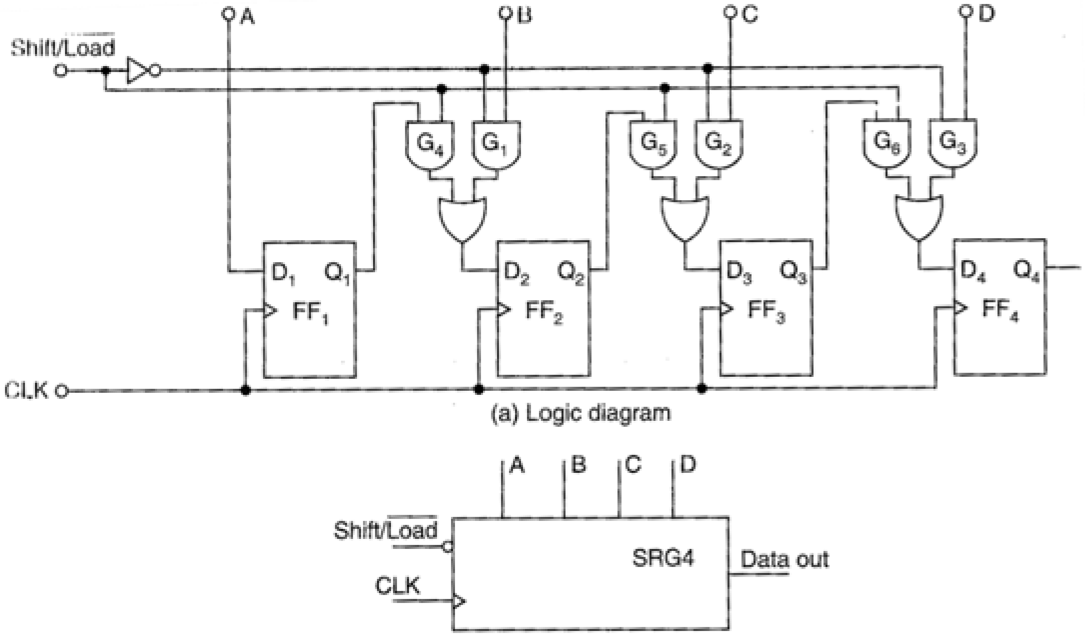
(a) Logic diagram

(b) Logic symbol

A 4- bit serial in, parallel out shift register

PARALLEL IN, SERIAL OUT SHIFT REGISTER:-

- For parallel in, serial out shift register the data bits are entered simultaneously into their respective stages on parallel lines, rather than on bit by bit basis on one line as with serial data inputs, but the data bits are transferred out of the register serially, i.e., on a bit by bit basis over a single line.
- The logic diagram and logic symbol of 4 bit parallel in, serial out shift register using D FFs is shown below.
- There are four data lines A, B, C and D through which the data is entered into the register in parallel form.
- The signal Shift /$\overline{\text{LOAD}}$ allows
  1. The data to be entered in parallel form into the register and
  2. The data to be shifted out serially from terminal Q_4.
- When Shift /$\overline{\text{LOAD}}$ line is HIGH, gates G1, G2, and G3 are disabled, but gates G4, G5 and G6 are enabled allowing the data bits to shift right from one stage to next.
- When Shift /$\overline{\text{LOAD}}$ line is LOW, gates G4, G5 and G6 are disabled, whereas gates G1, G2 and G3 are enabled allowing the data input to appear at the D inputs of the respective FFs.
- When clock pulse is applied, these data bits are shifted to the Q output terminals of the FFs and therefore the data is inputted in one step.
- The OR gate allows either the normal shifting operation or the parallel data entry depending on which AND gates are enabled by the level on the Shift /$\overline{\text{LOAD}}$ input.



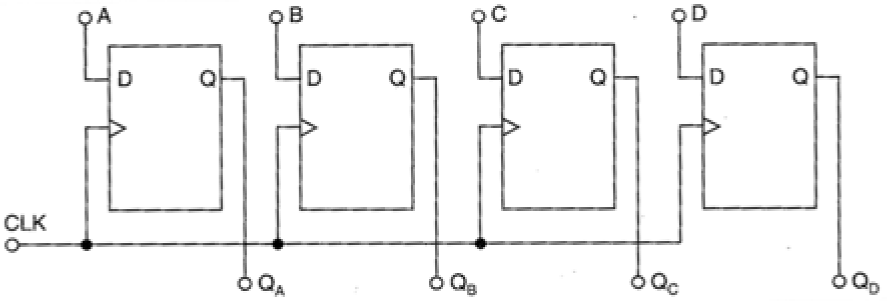
(a) Logic diagram

(b) Logic symbol

A 4- bit parallel in, serial out shift register

PARALLEL IN, PARALLEL OUT SHIFT REGISTER:-

- In a parallel in, parallel out shift register, the data entered into the register in parallel form and also the data taken out of the register in parallel form. Immediately following the simultaneous entry of all data bits appear on the parallel outputs.
- The figure shown below is a 4 bit parallel in parallel out shift register using D FFs.
- Data applied to the D input terminals of the FFs.
- When a clock pulse is applied at the positive edge of that pulse, the D inputs are shifted into the Q outputs of the FFs.
- The register now stores the data.
- The stored data is available instantaneously for shifting out in parallel form.

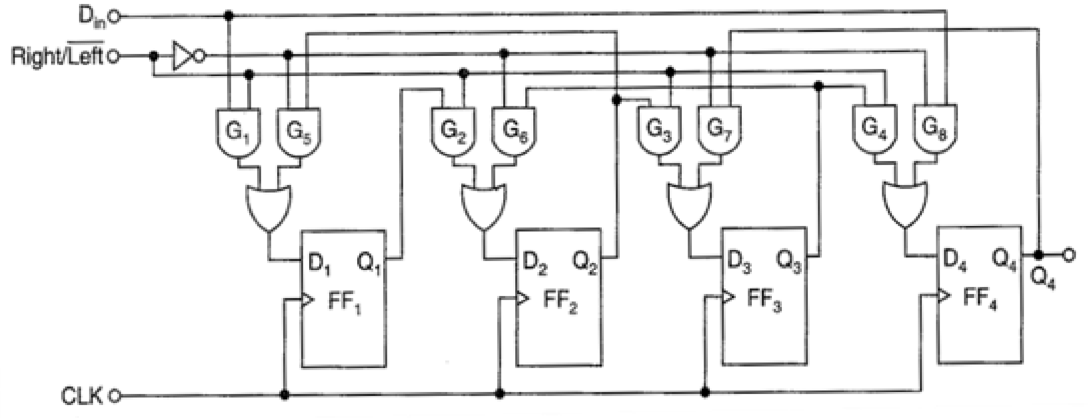


Logic diagram of a 4 – bit parallel in, parallel out shift register

BIDIRECTIONAL SHIFT REGISTER:-

- In bidirectional shift register is one in which the data bits can be shifted from left to right or from right to left.
- The figure shown below the logic diagram of a 4 bit serial in, serial out, bidirectional (shift-left, shift-right) shift register.
- Right /$\overline{\text{Left}}$ is the mode signal. When Right /$\overline{\text{Left}}$ is a 1, the logic circuit works as a shift right shift register. When Right /$\overline{\text{Left}}$ is a 0, the logic circuit works as a shift right shift register.

- The bidirectional is achieved by using the mode signal and two AND gates and one OR gate for each stage.
- A HIGH on the Right/$\overline{\text{Left}}$ control input enables the AND gates G_1, G_2, G_3 and G_4 and disables the AND gates G_5, G_6, G_7 and G_8 and the state of Q output of each FF is passed through the gate to the D input of the following FF. When clock pulse occurs, the data bits are effectively shifted one place to the right.
- A LOW Right/$\overline{\text{Left}}$ control input enables the AND gates G_5, G_6, G_7 and G_8 and disables the AND gates G_1, G_2, G_3 and G_4 and the Q output of each FF is passed to the D input of the preceding FF. When clock pulse occurs the data bits are then effectively shifted one place to the left.
- So, the circuit works as a bidirectional shift register.



Logic diagram of 4- bit bidirectional shift register

UNIVERSAL SHIFT REGISTERS:-

- The register which has both shifts and parallel load capabilities, it is referred as a universal shift register. So, universal shift register is a bidirectional register, whose input can be either in serial form or in parallel form and whose output also can be either in serial form or parallel form.
- The universal shift register can be realized using multiplexers.
- The figure shows the logic diagram of a 4 bit universal shift register that has all the capabilities of a general shift register.

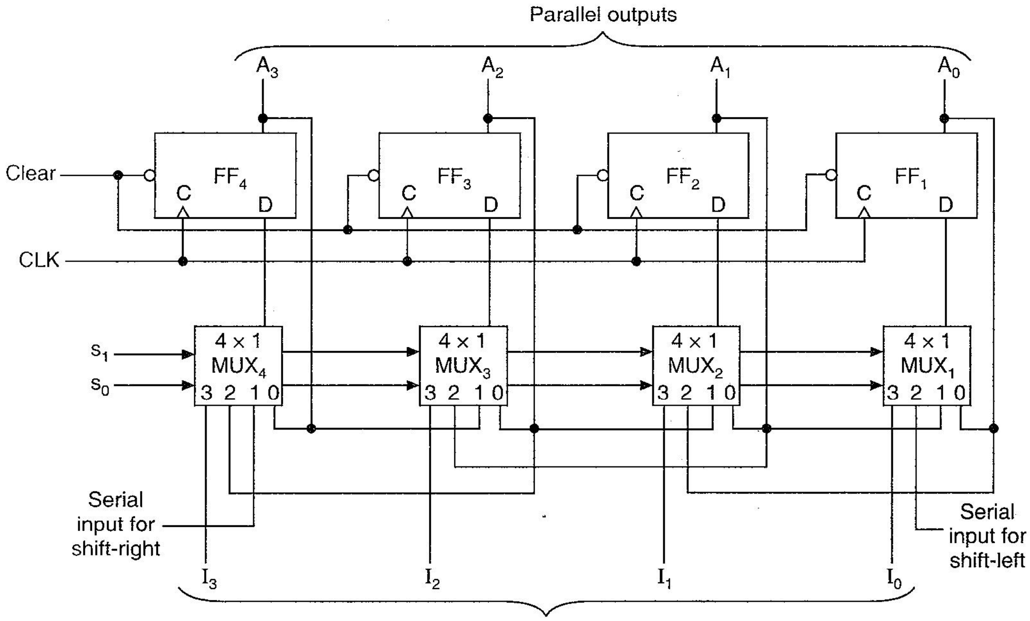
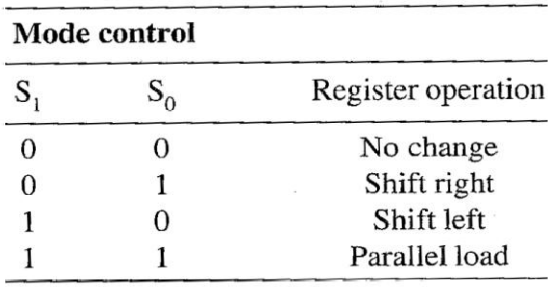


Fig- (a) 4 bit universal shift register

- It consists of four D flip- flops and four multiplexers.
- The four multiplexers have two common selection inputs S_1 and S_0.
- Input 0 in each multiplexer is selected when S_1S_0 = 00, input 1 is selected when S_1S_0 = 01, and input 2 is selected when S_1S_0 = 10 and input 3 is selected when S_1S_0= 11.
- The selection inputs control the mode of operation of the register is according to the function entries shown in the table.
- When S_1S_0 = 00 the present value of the register is applied to the D inputs of flip-flops. This condition forms a path from the output of each FF into the input of the same FF.
- The next clock edge transfers into each FF the binary value it held previously, and no change of state occurs.
- When S_1S_0 = 01, terminal 1 of the multiplexer inputs have a path of the D inputs of the flip- flops. This causes a shift right operation, with serial input transferred into FF_4.
- When S_1S_0 = 10 a shift left operation results with the other serial input going into the FF_1.
- Finally when S_1S_0 = 11, the binary information on the parallel input lines is transferred into the register simultaneously during the next clock edge.

Functional table for the register of fig – a:



Mode control		
S_1	S_0	Register operation
0	0	No change
0	1	Shift right
1	0	Shift left
1	1	Parallel load

APPLICATIONS OF SHIFT REGISTERS:-

1. **Time delays:**
   - In digital systems, it is necessary to delay the transfer of data until the operation of the other data have been completed, or to synchronize the arrival of data at a subsystem where it is processed with other data.
   - A shift register can be used to delay the arrival of serial data by a specific number of clock pulses, since the number of stages corresponds to the number of clock pulses required to shift each bit completely through the register.
   - The total time delay can be controlled by adjusting the clock frequency and by the number of stages in the register.
   - In practice, the clock frequency is fixed and the total delay can be adjusted only by controlling the number of stages through which the data is passed.
2. **Serial / Parallel data conversion:**
   - Transfer of data in parallel form is much faster than that in serial form.
   - Similarly the processing of data is much faster when all the data bits are available simultaneously. Thus in digital systems in which speed is important so to operate on data parallel form is used.
   - When large data is to be transmitted over long distances, transmitting data on parallel lines is costly and impracticable.
   - It is convenient and economical to transmit data in serial form, since serial data transmission requires only one line.

- Shift registers are used for converting serial data to parallel form, so that a serial input can be processed by a parallel system and for converting parallel data to serial form, so that parallel data can be transmitted serially.
- A serial in, parallel out shift register can be used to perform serial-to parallel conversion, and a parallel in, serial out shift register can be used to perform parallel- to –serial conversion.
- A universal shift register can be used to perform both the serial- to – parallel and parallel-to-serial data conversion.
- A bidirectional shift register can be used to reverse the order of data.

RING AND JOHNSON COUNTER:-

- Ring counters are constructed by modifying the serial-in, serial-out, shift register.
- There are two types of ring counters
  i) Basic ring counter
  ii) Johnson counter
- The basic ring counter can be obtained from a serial-in serial- out shift register by connecting the Q output of the last FF to the D input of the first FF.
- The Johnson counter can be obtained from serial-in, serial- out, shift register by connecting the Q output of the last FF to the D input of the first FF.
- Ring counter outputs can be used as a sequence of synchronizing pulses.
- The ring counter is a decimal counter.

D/A and A/D Converter

Weighted Register Network

The most significant bit (MSB) resistance is one-eighth of the least significant bit (LSB) resistance. R_L is much larger than 8R. The voltages V_A, V_B, V_C and V_D can be either equal to V (for logic 1) or 0 (for logical 0). Thus there are 2^4 = 16 input combinations from 0000 to 1111. The output voltage V_0, given by Millman's theorem is

V_0 =

When input is 0001, V_A = V_B= V_C= 0 and V_D = V and output is V/15. If input is 0010, V_A = V_B=V_D =0 and V_C=V giving an output of 2V/15. If input is 0011, V_A = V_B=0 and V_C=V_D = V giving an output of 3v/15. Thus, the output voltage varies from 0 to V in steps of V/15.

Binary Ladder Network

The weighted resistor network requires a range of resistor values. The binary ladder network requires only two resistance values. From node 1, the resistance to the digital source is 2R and resistance to ground is also 2R. From node 2, the resistance to digital source is 2R and resistance to ground =

R + (2R) (2R) / (2R+2R) = 2R

Thus, from each of the nodes 1,2,3,4, the resistance to source and ground is 2R each. A digital input 0001 means that D is connected to V and A, B, C are grounded. The output voltage V_0 is V/16. Thus as input varies from 0000 to 1111, the output varies from V/16 to V in steps of V/16.

A complete digital-to-analog converter circuit consists of a number of ladder networks (to deal with more bits of data), operational amplifier, gates etc.

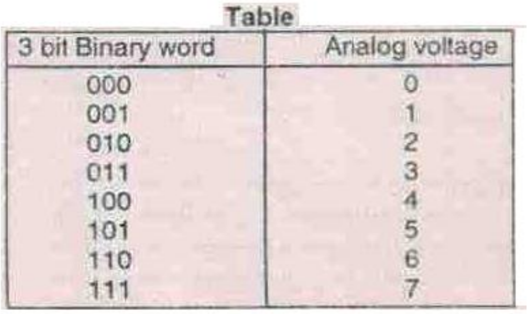
Performance Characteristics of D/A converters

The performance characteristics of D/A converters are resolution, accuracy, linear errors, monotonicity, setting time and temperature sensitivity.

(a) **Resolution:** It is the reciprocal of the number of discrete steps in the D/A output. Evidently resolution depends on the number of bits. The percentage resolution is [1/ (2^N-1)] * 100 where N is the number of bits. The percentage resolution for different values of N is given in table.

(b) **Accuracy:** It is a measure of the difference between actual output and expected output. It is expressed as a percentage of the maximum output voltage. If the maximum output voltage (or full scale deflection) is 5 V and accuracy is ±0.1%, then the maximum error is $\frac{0.1}{100}$ * 5 = 0.005 V or 5 mV. Ideally the accuracy should be better than ±0.5 of LSB. In an 8 bit converter, LSB is 1/256 or 0.39% of full scale. The accuracy should be better than 0.2%.

(c) **Setting Time:** When the input signal changes, it is desirable that analog output signal should immediately show the new output value. However in actual practice, the D/A converter takes some time to settle at the new position of the output voltage. Setting time is defined as the time taken by the D/A converter to settle with ±1/2 LSB of its final value when a change in input digital signal occurs. The final time taken to settle down to new value is due to the transients and oscillations in the output voltage. Figure shows the definition of setting time.



Table

3 bit Binary word	Analog voltage
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Fig 1

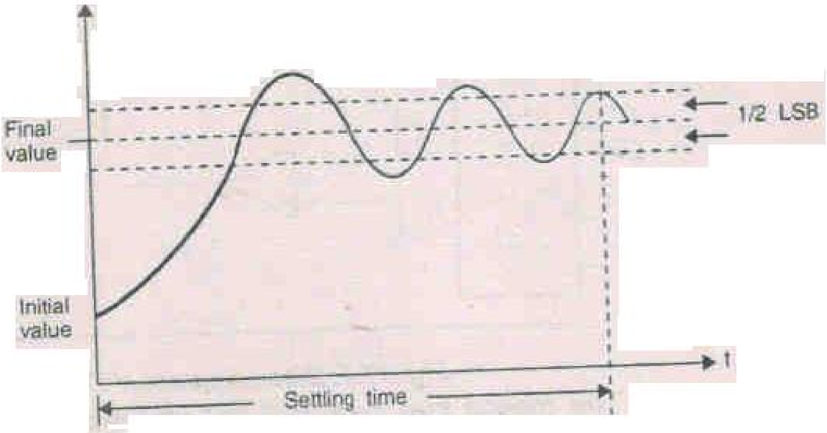
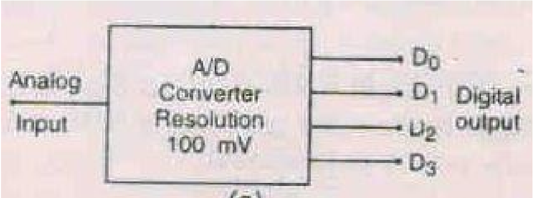


Fig. Settling time of D/A converter

Quantization error:

An analog to digital converter changes analog signal into digital signal. It is important to note that in D/A converter the number of input is fixed. In 4 bit D/a converter there are 16 possible inputs and in 6 bit D/A converter there are 64 possible inputs. However, in A/D converter the analog input voltage can have any value in the specified range but the digital output can have only 2^N discrete levels (for N bit converter). This means that there is a certain range of input voltage which correspond to every discrete output level.

Consider a 4 bit A/D converter having a resolution of 1 count per 100 mV. Fig (b) shows the analog input and digital output. It is seen that for input voltage range of 50 mV to 150 mV, the output is same i.e. 0001, for input voltage range of 150 mV to 250 mV, the output is the same, i.e. 0010. Thus we have one digital output for each 100 mV input range. If the digital signal of 0010 is fed to a D/A converter, it will show an output of 200 V whereas the original input voltage was between 150 V and 250 v. This error is called quntisation error and in this case this quntisation error can be ±50 mV and is equal to ±1/2 LSB.



(a)

Fig (a) A/D Converter

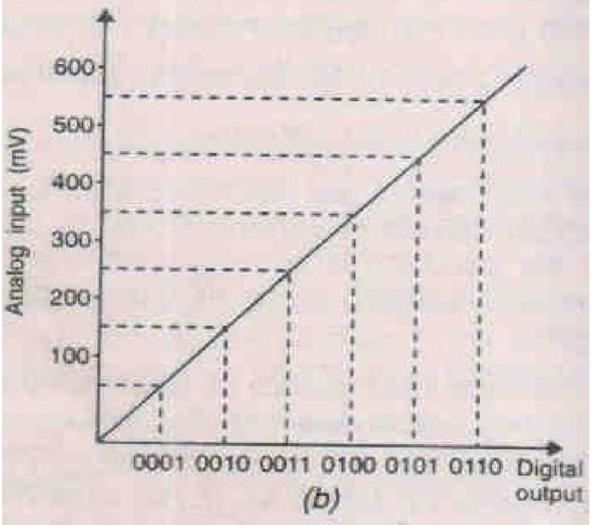


Fig (b) Quantisation error

Stair Step A/D Converter / Ramp A/D converter:

This converter is also called digital ramp or the counter type A/D converter. Figure shows the configuration for 8 bit converter. As seen in figure it uses a D/A converter and a binary counter to produce the digital number corresponding to analog input. The main components are comparator, AND gate, D/A converter, divide by 256 counter and latches. The analog input is given to non-inverting terminal of comparator. The D/A converter provides stair step reference voltage.

Let he counter be in reset state and output of D/A converter be zero. An analog input is given to non-inverting terminal of comparator. Since the reference input is 0, the comparator gives High output and enables the AND gate. The clock pulses cause advancing of counter through its binary states and stair step reference voltage is produced from D/A converter. As the counter keeps advancing, successively higher stair step output voltage is produced. When this stair step voltage reaches the level of analog input voltage, the comparator output goes Low and disables the AND gate. The clock pulses are cut off and counter stops. The state of counter at this point is equal to the number of steps in reference voltage at which comparison occurs. The binary number corresponding to this number of steps is the value of the analog input voltage. The control logic causes this binary number to be loaded into the latches and counter is reset.

This converter is rather slow in action because the counter has to pass through the maximum number of states before a conversion takes place. For 8 bit device this means 256 counter states.

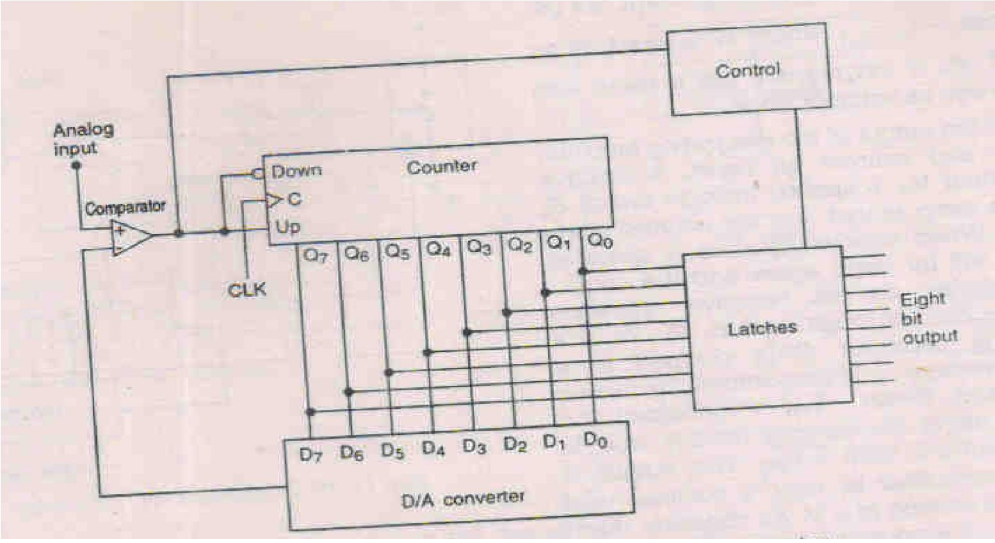


Fig (a) 8 bit up-down counter type A/D converter

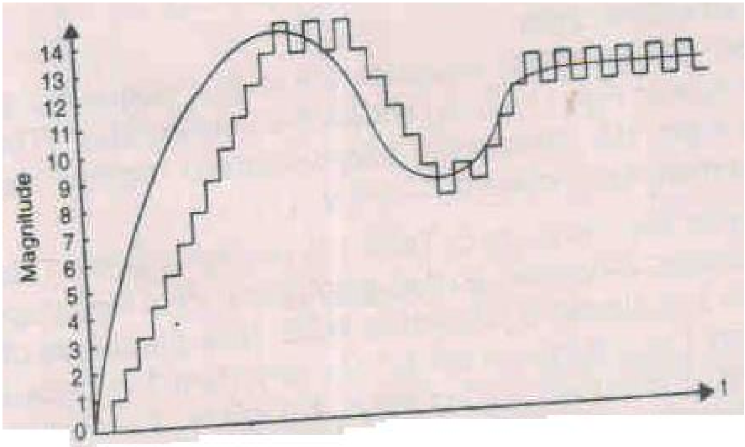


Fig (b) Tracking action of updown counter type
A/D Converter

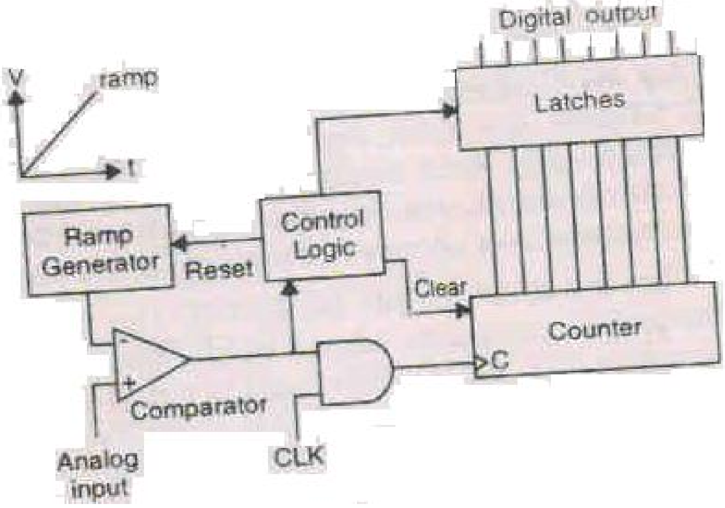


Fig (c) Single slope A/D converter

Dual slope A/D converter:

The single slope A/D converter is suscetible to noise. The dual slope converter is free from this problem. It uses an op-amp used as integreting amplifier for ramp generator. It is dual slope device because it uses a fixed slope ramp as well as variable slope ramp. Fig. Shows the configuration.

It is seen that the integreting op-amp uses a capacitor in the feedback path.

Output voltage of integreting op-amp = $-\frac{1}{C}\int i\,dt$ = $-\frac{1}{RC}\int V_{in}$ dt

Thus the output voltage is integral of analog input voltage. If V_{in} is constant, we get an output $-V_{in}\frac{t}{RC}$ which is a fixed slope ramp. If V_{in} is varing we get a ramp with fixed as well as variable slope.

Let the output of the integreting amplifier be zero and counter be reset. A positive analog input V_{in} is applied through switch S, we get a ramp output and the counter starts working. When counter reaches a specified count, it will be reset again and the control logic switches on the negative reference voltage $-V_{ref}$ (through switch S). At this instant the capacitor C is charged to a negative voltage - V proportional to analog input voltage. When $-V_{ref}$ is connected the capacitor starts discharging linearly due to constant current from $-V_{ref}$. The output of integreting amplifier is now a positive fixed slope ramp starting at – V. As capacitor discharges, the counter advances from the reset state. When the output of integretor becomes zero, the comparator output becomes Low and disables the clock signal to the AND gate. The counter is therefore stopped and the binary counter is latched. This completes one conversion cycle. The binary count is propor tional to analog input V_{in}.

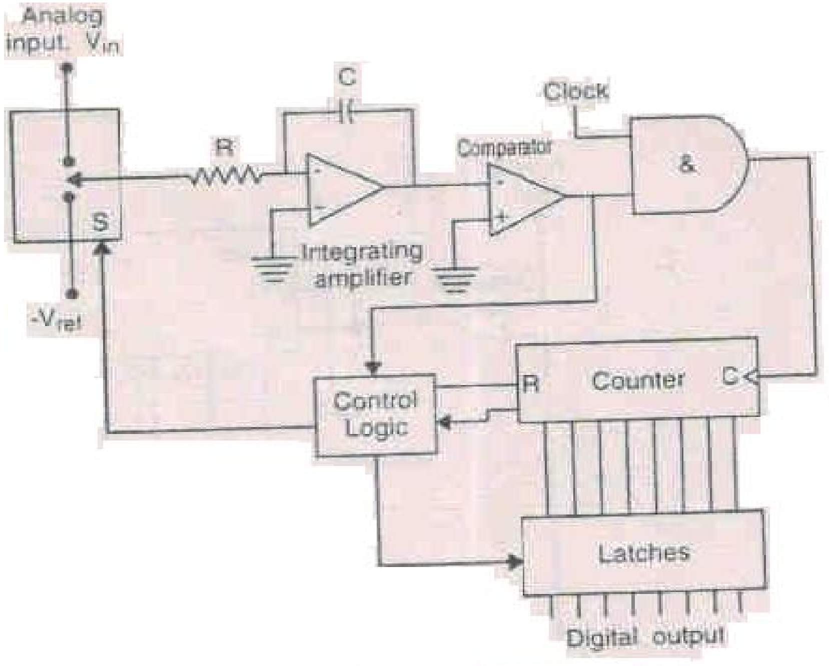


Fig. Dual slope A/D converter

Successive Approximation A/D Converter:

This is the most widely used A/D converter. As the name suggests the digital output tends towards analog input through successive approximations. Fig. Shows the configuration. The main components are op-amp comparator, control logic, SA (successive approximation) register and D/A converter. As shown it is a six bit device using a maximum reference of 64 V.

Let the analog input be 26.1 v. The SA register is first set to zero. Then 1 is placed in MSB. This is fed to D/A converter whose output goes to comparator. Since the analog input (26.1 V) is less than D/A output (i.e. 32 V) the MSB is set to zero. Then 1 is placed in bit next to MSB. Now the output of D/A is 16 V. Since analog input is more than 16 V, this 1 is retained in this bit position. Next 1 is placed in third bit position. Now the D/A output is 24 V which is less than analog input. Therefore this 1 bit is retained and 1 is placed in the next bit. Now the D/A output is 28 V, which is more than analog input. Therefore this 1 bit is set to zero and 1 is placed in 5th bit position producing a D/A output of 26 V. It is less than analog input. Therefore this 1 bit is retained. Now 1 is placed in LSB producing a D/A output of 27 V which is more than analog input. Therefore LSB is set to zero and the converter gives an output of 26 V.

The successive approximation method of A/D converter is very fast and takes only about 250 ns/ bit.

Performance Characteristics of A/D converters:

The performance characteristics of A/D converters are resolution, accuracy, A/D gain and drift and A/D speed.

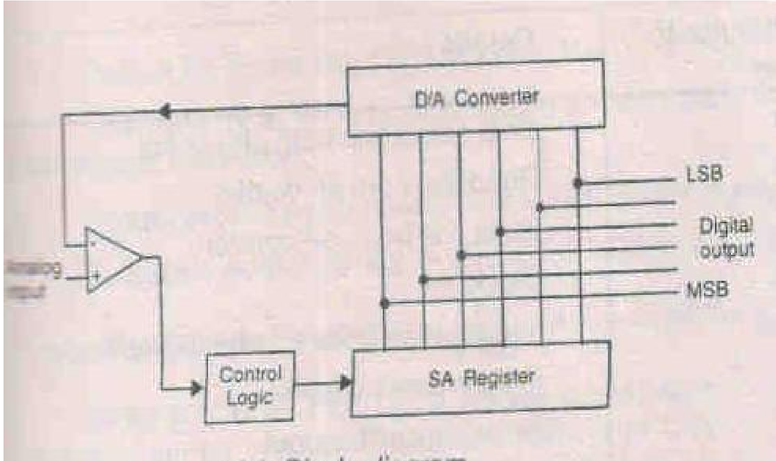
(a) Resolution: A/D rsolution is the change in voltage input necessary for a one bit change in output. It can also be expressed as percent.

(b) A/D Accuracy: The accuracy of A/D conversion is limited by the ±1/2 LSB due to quantisation error and the other errors of the system. It is defined as the maximum deviation of digital output from the ideal linear reference line. Ideally it aproaches ±1/2 LSB.

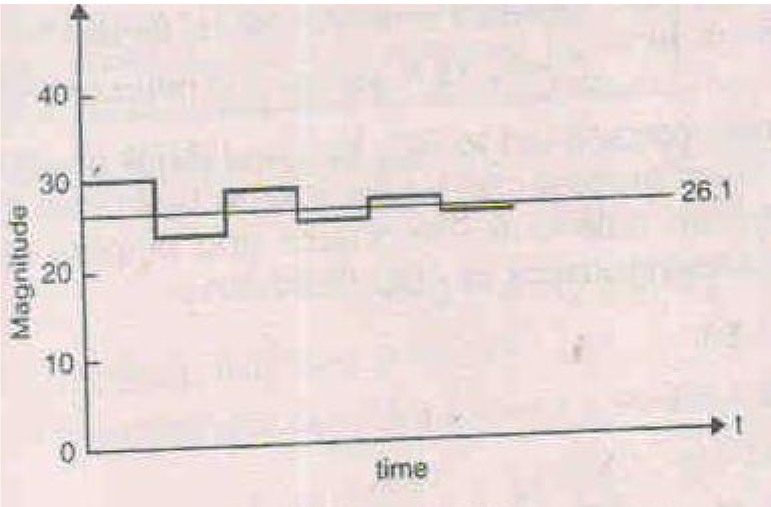
(c) A/D gain and Drift: A/D gain is the voltage output is devided by the voltage input at the linearity reference line. It can usually be zeroed out.

Drift means change in circuit parameters with time. Drift errors of upto ±1/2 LSB will cause a maximum errors of one LSB between the first and the last transition. Very low drift is quite difficult to achieve and increases cost of the device.

(d) A/D speed: It can be defined in two ways, i.e. either the time necessary to do one conversion or the line between successive conversion at the highest rate possible. Speed depends on the settling time of components and the speed of the logic.



(a) Block diagram



(b) Waveforms

MODULE: 1

1. INTRODUCTION TO MICROPROCESSOR AND MICROCOMPUTER ARCHITECTURE:

A *microprocessor* is a programmable electronics chip that has computing and decision making capabilities similar to central processing unit of a computer. Any microprocessor-based systems having limited number of resources are called *microcomputers*. Nowadays, microprocessor can be seen in almost all types of electronics devices like mobile phones, printers, washing machines etc. Microprocessors are also used in advanced applications like radars, satellites and flights. Due to the rapid advancements in electronic industry and large scale integration of devices results in a significant cost reduction and increase application of microprocessors and their derivatives.

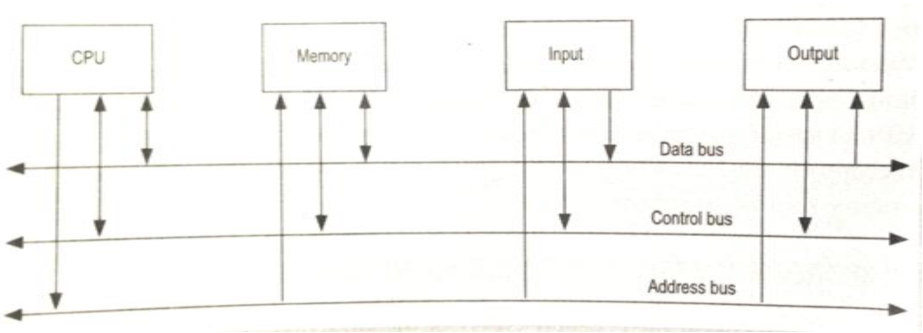


Fig.1 Microprocessor-based system

- **Bit**: A bit is a single binary digit.
- **Word**: A word refers to the basic data size or bit size that can be processed by the arithmetic and logic unit of the processor. A 16-bit binary number is called a word in a 16-bit processor.
- **Bus**: A bus is a group of wires/lines that carry similar information.
- **System Bus**: The system bus is a group of wires/lines used for communication between the microprocessor and peripherals.
- **Memory Word**: The number of bits that can be stored in a register or memory element is called a memory word.
- **Address Bus**: It carries the address, which is a unique binary pattern used to identify a memory location or an I/O port. For example, an eight bit address bus has eight lines and thus it can address 2^8 = 256 different locations. The locations in hexadecimal format can be written as 00H – FFH.
- **Data Bus**: The data bus is used to transfer data between memory and processor or between I/O device and processor. For example, an 8-bit processor will generally have an 8-bit data bus and a 16-bit processor will have 16-bit data bus.
- **Control Bus**: The control bus carry control signals, which consists of signals for selection of memory or I/O device from the given address, direction of data transfer and synchronization of data transfer in case of slow devices.

A typical microprocessor consists of arithmetic and logic unit (ALU) in association with control unit to process the instruction execution. Almost all the microprocessors are based on the principle of store-program concept. In *store-program concept*, programs or instructions are sequentially stored in the memory locations that are to be executed. To do any task using a microprocessor, it is to be programmed by the user. So the programmer must have idea about its internal resources, features and supported instructions. Each microprocessor has a set of instructions, a list which is provided by the microprocessor manufacturer. The instruction set of a microprocessor is provided in two forms: *binary machine code and mnemonics*.

Microprocessor communicates and operates in binary numbers 0 and 1. The set of instructions in the form of binary patterns is called a *machine language* and it is difficult for us to understand. Therefore, the binary patterns are given abbreviated names, called mnemonics, which forms the *assembly language*. The conversion of assembly-level language into binary machine-level language is done by using an application called *assembler.*

Technology Used:

The semiconductor manufacturing technologies used for chips are:

- Transistor-Transistor Logic (TTL)
- Emitter Coupled Logic (ECL)
- Complementary Metal-Oxide Semiconductor (CMOS)

Classification of Microprocessors:

Based on their specification, application and architecture microprocessors are classified.

Based on size of data bus:

- 4-bit microprocessor
- 8-bit microprocessor
- 16-bit microprocessor
- 32-bit microprocessor

Based on application:

- General-purpose microprocessor- used in general computer system and can be used by programmer for any application. Examples, 8085 to Intel Pentium.
- Microcontroller- microprocessor with built-in memory and ports and can be programmed for any generic control application. Example, 8051.
- Special-purpose processors- designed to handle special functions required for an application. Examples, digital signal processors and application-specific integrated circuit (ASIC) chips.

Based on architecture:

- Reduced Instruction Set Computer (RISC) processors
- Complex Instruction Set Computer (CISC) processors

2. 8085 MICROPROCESSOR ARCHITECTURE

The 8085 microprocessor is an 8-bit processor available as a 40-pin IC package and uses +5 V for power. It can run at a maximum frequency of 3 MHz. Its data bus width is 8-bit and address bus width is 16-bit, thus it can address 2^{16} = 64 KB of memory. The internal architecture of 8085 is shown is Fig. 2.

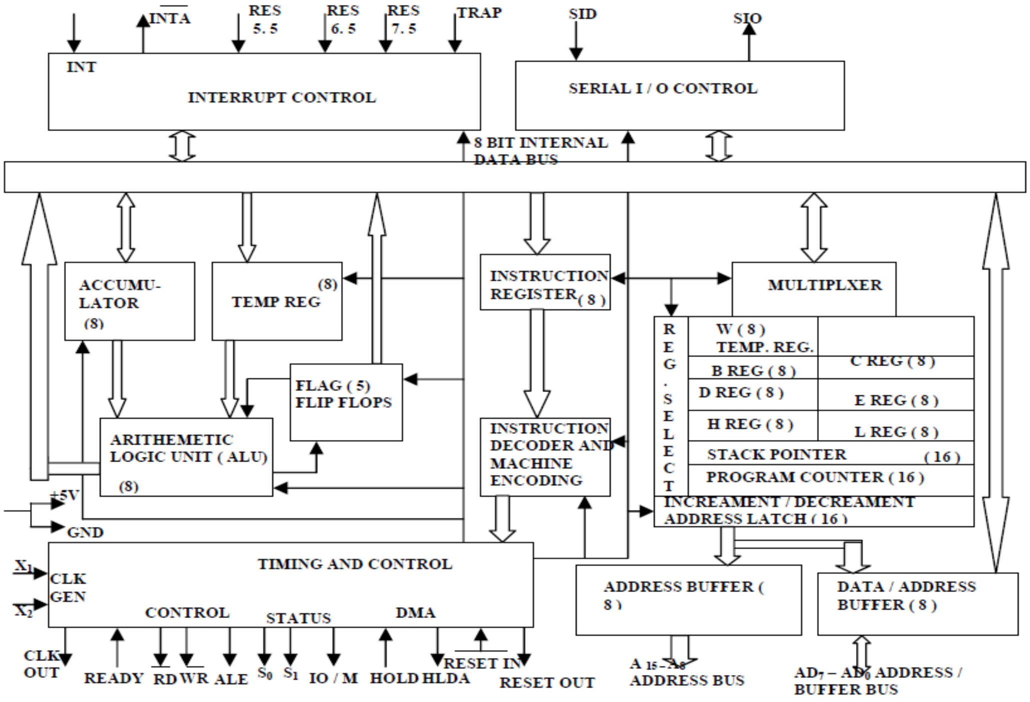


Fig. 2 Internal Architecture of 8085

Arithmetic and Logic Unit

The ALU performs the actual numerical and logical operations such as Addition (ADD), Subtraction (SUB), AND, OR etc. It uses data from memory and from Accumulator to perform operations. The results of the arithmetic and logical operations are stored in the accumulator.

Registers

The 8085 includes six registers, one accumulator and one flag register, as shown in Fig. 3. In addition, it has two 16-bit registers: stack pointer and program counter. They are briefly described as follows.

The 8085 has six general-purpose registers to store 8-bit data; these are identified as B, C, D, E, H and L. they can be combined as register pairs - BC, DE and HL to perform some

16-bit operations. The programmer can use these registers to store or copy data into the register by using data copy instructions.

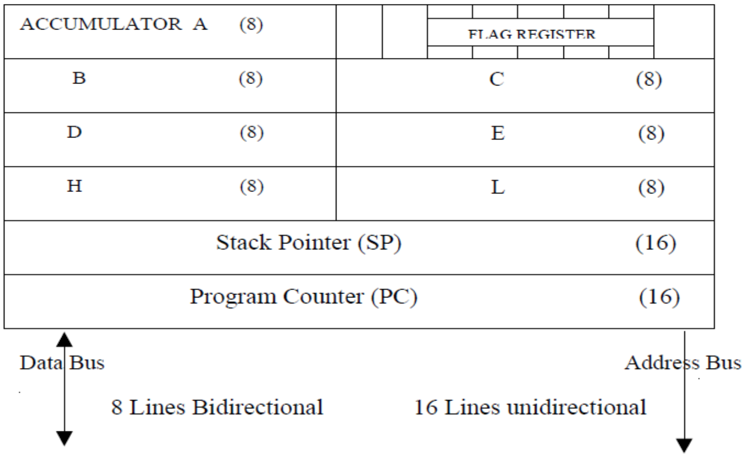


Fig. 3 Register organisation

Accumulator

The accumulator is an 8-bit register that is a part of ALU. This register is used to store 8-bit data and to perform arithmetic and logical operations. The result of an operation is stored in the accumulator. The accumulator is also identified as register A.

Flag register

The ALU includes five flip-flops, which are set or reset after an operation according to data condition of the result in the accumulator and other registers. They are called Zero (Z), Carry (CY), Sign (S), Parity (P) and Auxiliary Carry (AC) flags. Their bit positions in the flag register are shown in Fig. 4. The microprocessor uses these flags to test data conditions.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
S	**Z**		**AC**		**P**		**CY**

Fig. 4 Flag register

For example, after an addition of two numbers, if the result in the accumulator is larger than 8-bit, the flip-flop uses to indicate a carry by setting CY flag to 1. When an arithmetic operation results in zero, Z flag is set to 1. The S flag is just a copy of the bit D7 of the accumulator. A negative number has a 1 in bit D7 and a positive number has a 0 in 2’s complement representation. The AC flag is set to 1, when a carry result from bit D3 and passes to bit D4. The P flag is set to 1, when the result in accumulator contains even number of 1s.

Program Counter (PC)

This 16-bit register deals with sequencing the execution of instructions. This register is a memory pointer. The microprocessor uses this register to sequence the execution of the instructions. The function of the program counter is to point to the memory address from which the next byte is to be fetched. When a byte is being fetched, the program counter is automatically incremented by one to point to the next memory location.

Stack Pointer (SP)

The stack pointer is also a 16-bit register, used as a memory pointer. It points to a memory location in R/W memory, called stack. The beginning of the stack is defined by loading 16-bit address in the stack pointer.

Instruction Register/Decoder

It is an 8-bit register that temporarily stores the current instruction of a program. Latest instruction sent here from memory prior to execution. Decoder then takes instruction and decodes or interprets the instruction. Decoded instruction then passed to next stage.

Control Unit

Generates signals on data bus, address bus and control bus within microprocessor to carry out the instruction, which has been decoded. Typical buses and their timing are described as follows:

- *Data Bus*: Data bus carries data in binary form between microprocessor and other external units such as memory. It is used to transmit data i.e. information, results of arithmetic etc between memory and the microprocessor. Data bus is bidirectional in nature. The data bus width of 8085 microprocessor is 8-bit i.e. 2^8 combination of binary digits and are typically identified as D0 – D7. Thus size of the data bus determines what arithmetic can be done. If only 8-bit wide then largest number is 11111111 (255 in decimal). Therefore, larger numbers have to be broken down into chunks of 255. This slows microprocessor.
- *Address Bus*: The address bus carries addresses and is one way bus from microprocessor to the memory or other devices. 8085 microprocessor contain 16-bit address bus and are generally identified as A0 - A15. The higher order address lines (A8 – A15) are unidirectional and the lower order lines (A0 – A7) are multiplexed (time-shared) with the eight data bits (D0 – D7) and hence, they are bidirectional.
- *Control Bus*: Control bus are various lines which have specific functions for coordinating and controlling microprocessor operations. The control bus carries control signals partly unidirectional and partly bidirectional. The following control and status signals are used by 8085 processor:
    I. ALE (output): Address Latch Enable is a pulse that is provided when an address appears on the AD0 – AD7 lines, after which it becomes 0.

II. $\overline{RD}$ (active low output): The Read signal indicates that data are being read from the selected I/O or memory device and that they are available on the data bus.

III. $\overline{WR}$ (active low output): The Write signal indicates that data on the data bus are to be written into a selected memory or I/O location.

IV. $IO/\overline{M}$ (output): It is a signal that distinguished between a memory operation and an I/O operation. When $IO/\overline{M} = 0$ it is a memory operation and $IO/\overline{M} = 1$ it is an I/O operation.

V. S1 and S0 (output): These are status signals used to specify the type of operation being performed; they are listed in Table 1.

Table 1 Status signals and associated operations

S1	S0	States
0	0	Halt
0	1	Write
1	0	Read
1	1	Fetch

The schematic representation of the 8085 bus structure is as shown in Fig. 5. The microprocessor performs primarily four operations:

I. Memory Read: Reads data (or instruction) from memory.
II. Memory Write: Writes data (or instruction) into memory.
III. I/O Read: Accepts data from input device.
IV. I/O Write: Sends data to output device.

The 8085 processor performs these functions using address bus, data bus and control bus as shown in Fig. 5.

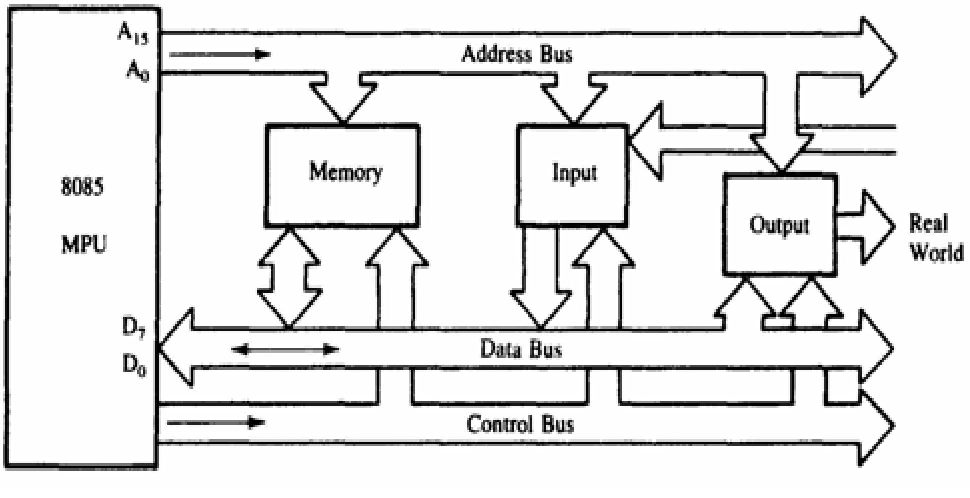


Fig. 5 The 8085 bus structure

3. 8085 PIN DESCRIPTION

Properties:

- It is a 8-bit microprocessor
- Manufactured with N-MOS technology
- 40 pin IC package
- It has 16-bit address bus and thus has 2^{16} = 64 KB addressing capability.
- Operate with 3 MHz single-phase clock
- +5 V single power supply

The logic pin layout and signal groups of the 8085nmicroprocessor are shown in Fig. 6. All the signals are classified into six groups:

- Address bus
- Data bus
- Control & status signals
- Power supply and frequency signals
- Externally initiated signals
- Serial I/O signals

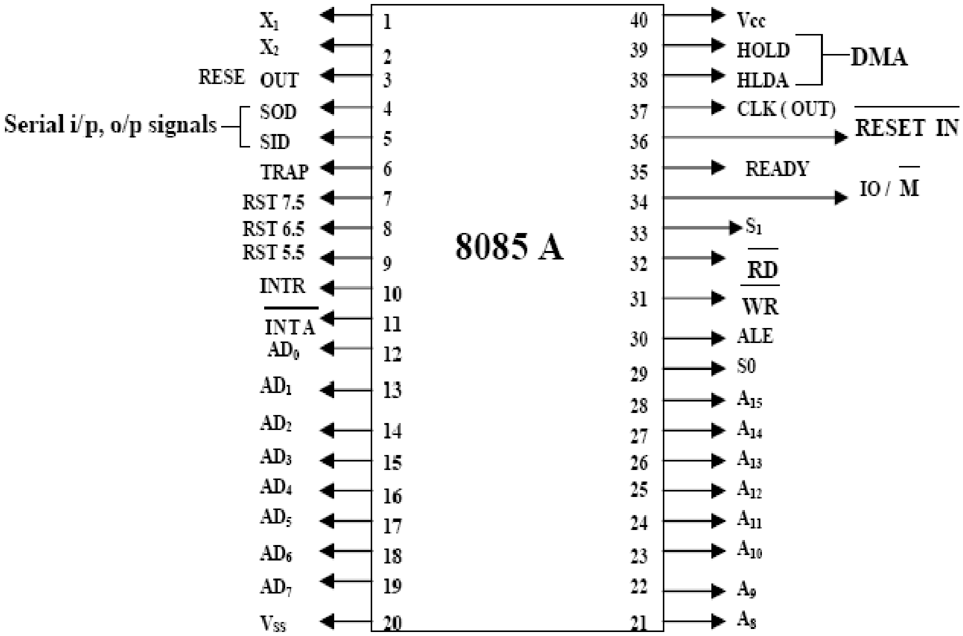


Fig. 6 8085 microprocessor pin layout and signal groups

Address and Data Buses:

- A8 – A15 (output, 3-state): Most significant eight bits of memory addresses and the eight bits of the I/O addresses. These lines enter into tri-state high impedance state during HOLD and HALT modes.
- AD0 – AD7 (input/output, 3-state): Lower significant bits of memory addresses and the eight bits of the I/O addresses during first clock cycle. Behaves as data bus

during third and fourth clock cycle. These lines enter into tri-state high impedance state during HOLD and HALT modes.

Control & Status Signals:

- ALE: Address latch enable
- $\overline{\text{RD}}$: Read control signal.
- $\overline{\text{WR}}$: Write control signal.
- IO/$\overline{\text{M}}$, S1 and S0 : Status signals.

Power Supply & Clock Frequency:

- Vcc: +5 V power supply
- Vss: Ground reference
- X1, X2: A crystal having frequency of 6 MHz is connected at these two pins
- CLK: Clock output

Externally Initiated and Interrupt Signals:

- $\overline{\text{RESET IN}}$: When the signal on this pin is low, the PC is set to 0, the buses are tri-stated and the processor is reset.
- RESET OUT: This signal indicates that the processor is being reset. The signal can be used to reset other devices.
- READY: When this signal is low, the processor waits for an integral number of clock cycles until it goes high.
- HOLD: This signal indicates that a peripheral like DMA (direct memory access) controller is requesting the use of address and data bus.
- HLDA: This signal acknowledges the HOLD request.
- INTR: Interrupt request is a general-purpose interrupt.
- $\overline{\text{INTA}}$: This is used to acknowledge an interrupt.
- RST 7.5, RST 6.5, RST 5,5 – restart interrupt: These are vectored interrupts and have highest priority than INTR interrupt.
- TRAP: This is a non-maskable interrupt and has the highest priority.

Serial I/O Signals:

- SID: Serial input signal. Bit on this line is loaded to D7 bit of register A using RIM instruction.
- SOD: Serial output signal. Output SOD is set or reset by using SIM instruction.
-

4. INSTRUCTION SET AND EXECUTION IN 8085

Based on the design of the ALU and decoding unit, the microprocessor manufacturer provides instruction set for every microprocessor. The instruction set consists of both machine code and mnemonics.

An instruction is a binary pattern designed inside a microprocessor to perform a specific function. The entire group of instructions that a microprocessor supports is called instruction set. Microprocessor instructions can be classified based on the parameters such functionality, length and operand addressing.

Classification based on functionality:

I. Data transfer operations: This group of instructions copies data from source to destination. The content of the source is not altered.

II. Arithmetic operations: Instructions of this group perform operations like addition, subtraction, increment & decrement. One of the data used in arithmetic operation is stored in accumulator and the result is also stored in accumulator.

III. Logical operations: Logical operations include AND, OR, EXOR, NOT. The operations like AND, OR and EXOR uses two operands, one is stored in accumulator and other can be any register or memory location. The result is stored in accumulator. NOT operation requires single operand, which is stored in accumulator.

IV. Branching operations: Instructions in this group can be used to transfer program sequence from one memory location to another either conditionally or unconditionally.

V. Machine control operations: Instruction in this group control execution of other instructions and control operations like interrupt, halt etc.

Classification based on length:

I. One-byte instructions: Instruction having one byte in machine code. Examples are depicted in Table 2.

I. Two-byte instructions: Instruction having two byte in machine code. Examples are depicted in Table 3

II. Three-byte instructions: Instruction having three byte in machine code. Examples are depicted in Table 4.

Table 2 Examples of one byte instructions

Opcode	Operand	Machine code/Hex code
MOV	A, B	78
ADD	M	86

Table 3 Examples of two byte instructions

Opcode	Operand	Machine code/Hex code	Byte description
MVI	A, 7FH	3E	First byte
		7F	Second byte
ADI	0FH	C6	First byte
		0F	Second byte

Table 4 Examples of three byte instructions

Opcode	Operand	Machine code/Hex code	Byte description
JMP	9050H	C3	First byte
		50	Second byte
		90	Third byte
LDA	8850H	3A	First byte
		50	Second byte
		88	Third byte

Addressing Modes in Instructions:

The process of specifying the data to be operated on by the instruction is called addressing. The various formats for specifying operands are called addressing modes. The 8085 has the following five types of addressing:

I. Immediate addressing
II. Memory direct addressing
III. Register direct addressing
IV. Indirect addressing
V. Implicit addressing

Immediate Addressing:

In this mode, the operand given in the instruction - a byte or word – transfers to the destination register or memory location.

Ex: MVI A, 9AH

- The operand is a part of the instruction.
- The operand is stored in the register mentioned in the instruction.

Memory Direct Addressing:

Memory direct addressing moves a byte or word between a memory location and register. The memory location address is given in the instruction.

Ex: LDA 850FH

This instruction is used to load the content of memory address 850FH in the accumulator.

Register Direct Addressing:

Register direct addressing transfer a copy of a byte or word from source register to destination register.

Ex: MOV B, C

It copies the content of register C to register B.

Indirect Addressing:

Indirect addressing transfers a byte or word between a register and a memory location.

Ex: MOV A, M

Here the data is in the memory location pointed to by the contents of HL pair. The data is moved to the accumulator.

Implicit Addressing

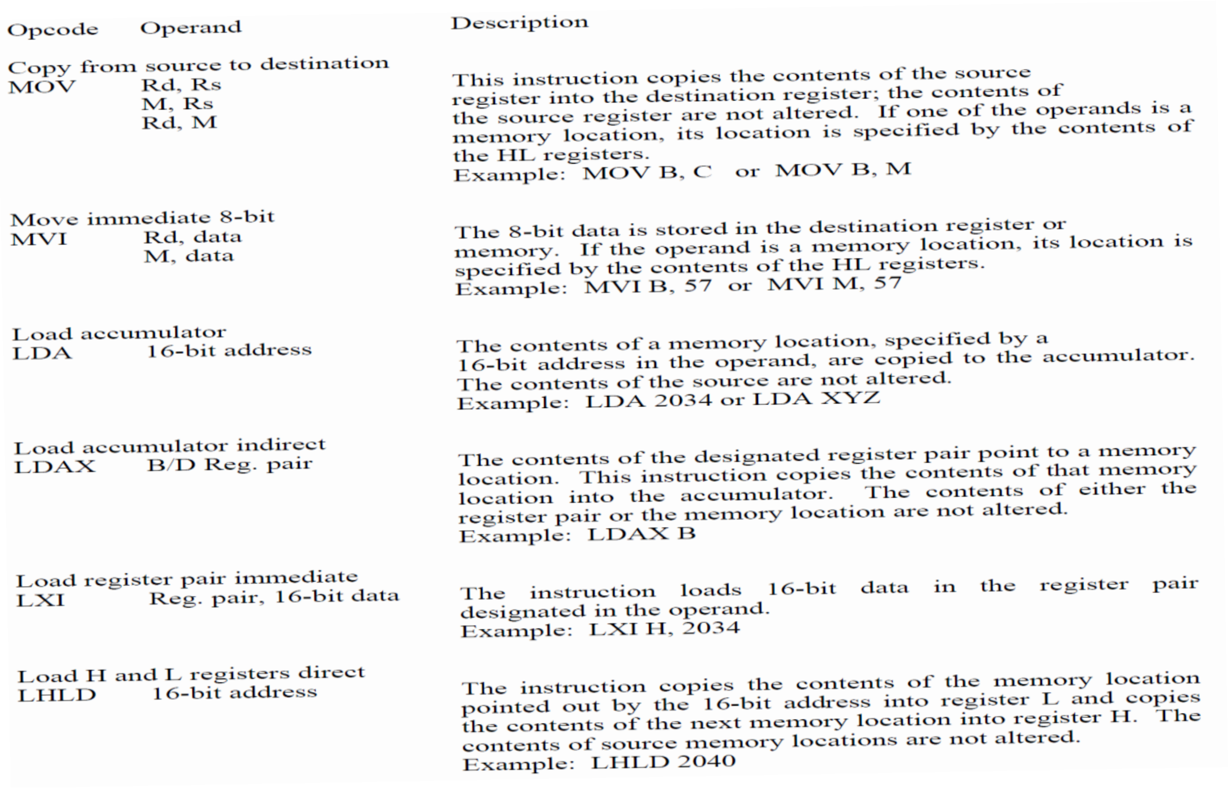
In this addressing mode the data itself specifies the data to be operated upon.

Ex: CMA

The instruction complements the content of the accumulator. No specific data or operand is mentioned in the instruction.

5. **INSTRUCTION SET OF 8085**

Data Transfer Instructions:



Opcode	Operand	Description
Copy from source to destination MOV	 Rd, Rs M, Rs Rd, M	This instruction copies the contents of the source register into the destination register; the contents of the source register are not altered. If one of the operands is a memory location, its location is specified by the contents of the HL registers. Example: MOV B, C or MOV B, M
Move immediate 8-bit MVI	 Rd, data M, data	The 8-bit data is stored in the destination register or memory. If the operand is a memory location, its location is specified by the contents of the HL registers. Example: MVI B, 57 or MVI M, 57
Load accumulator LDA	 16-bit address	The contents of a memory location, specified by a 16-bit address in the operand, are copied to the accumulator. The contents of the source are not altered. Example: LDA 2034 or LDA XYZ
Load accumulator indirect LDAX	 B/D Reg. pair	The contents of the designated register pair point to a memory location. This instruction copies the contents of that memory location into the accumulator. The contents of either the register pair or the memory location are not altered. Example: LDAX B
Load register pair immediate LXI	 Reg. pair, 16-bit data	The instruction loads 16-bit data in the register pair designated in the operand. Example: LXI H, 2034
Load H and L registers direct LHLD	 16-bit address	The instruction copies the contents of the memory location pointed out by the 16-bit address into register L and copies the contents of the next memory location into register H. The contents of source memory locations are not altered. Example: LHLD 2040

Store accumulator direct
STA 16-bit address

The contents of the accumulator are copied into the memory location specified by the operand. This is a 3-byte instruction, the second byte specifies the low-order address and the third byte specifies the high-order address.
Example: STA 4350 or STA XYZ

Store accumulator indirect
STAX Reg. pair

The contents of the accumulator are copied into the memory location specified by the contents of the operand (register pair). The contents of the accumulator are not altered.
Example: STAX B

Store H and L registers direct
SHLD 16-bit address

The contents of register L are stored into the memory location specified by the 16-bit address in the operand and the contents of H register are stored into the next memory location by incrementing the operand. The contents of registers HL are not altered. This is a 3-byte instruction, the second byte specifies the low-order address and the third byte specifies the high-order address.
Example: SHLD 2470

Exchange H and L with D and E
XCHG none

The contents of register H are exchanged with the contents of register D, and the contents of register L are exchanged with the contents of register E.
Example: XCHG

Copy H and L registers to the stack pointer
SPHL none

The instruction loads the contents of the H and L registers into the stack pointer register, the contents of the H register provide the high-order address and the contents of the L register provide the low-order address. The contents of the H and L registers are not altered.
Example: SPHL

Exchange H and L with top of stack
XTHL none

The contents of the L register are exchanged with the stack location pointed out by the contents of the stack pointer register. The contents of the H register are exchanged with the next stack location (SP+1); however, the contents of the stack pointer register are not altered.
Example: XTHL

Push register pair onto stack		
PUSH	Reg. pair	The contents of the register pair designated in the operand are copied onto the stack in the following sequence. The stack pointer register is decremented and the contents of the high-order register (B, D, H, A) are copied into that location. The stack pointer register is decremented again and the contents of the low-order register (C, E, L, flags) are copied to that location. Example: PUSH B or PUSH A
Pop off stack to register pair		
POP	Reg. pair	The contents of the memory location pointed out by the stack pointer register are copied to the low-order register (C, E, L, status flags) of the operand. The stack pointer is incremented by 1 and the contents of that memory location are copied to the high-order register (B, D, H, A) of the operand. The stack pointer register is again incremented by 1. Example: POP H or POP A
Output data from accumulator to a port with 8-bit address		
OUT	8-bit port address	The contents of the accumulator are copied into the I/O port specified by the operand. Example: OUT 87
Input data to accumulator from a port with 8-bit address		
IN	8-bit port address	The contents of the input port designated in the operand are read and loaded into the accumulator. Example: IN 82

Arithmetic Instructions:

Opcode	Operand	Description
Add register or memory to accumulator		
ADD	R M	The contents of the operand (register or memory) are added to the contents of the accumulator and the result is stored in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the addition. Example: ADD B or ADD M
Add register to accumulator with carry		
ADC	R M	The contents of the operand (register or memory) and the Carry flag are added to the contents of the accumulator and the result is stored in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the addition. Example: ADC B or ADC M
Add immediate to accumulator		
ADI	8-bit data	The 8-bit data (operand) is added to the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the addition. Example: ADI 45
Add immediate to accumulator with carry		
ACI	8-bit data	The 8-bit data (operand) and the Carry flag are added to the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the addition. Example: ACI 45
Add register pair to H and L registers		
DAD	Reg. pair	The 16-bit contents of the specified register pair are added to the contents of the HL register and the sum is stored in the HL register. The contents of the source register pair are not altered. If the result is larger than 16 bits, the CY flag is set. No other flags are affected. Example: DAD H

Subtract register or memory from accumulator

SUB R
M

The contents of the operand (register or memory) are subtracted from the contents of the accumulator, and the result is stored in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the subtraction.
Example: SUB B or SUB M

Subtract source and borrow from accumulator

SBB R
M

The contents of the operand (register or memory) and the Borrow flag are subtracted from the contents of the accumulator and the result is placed in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the subtraction.
Example: SBB B or SBB M

Subtract immediate from accumulator

SUI 8-bit data

The 8-bit data (operand) is subtracted from the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the subtraction.
Example: SUI 45

Subtract immediate from accumulator with borrow

SBI 8-bit data

The 8-bit data (operand) and the Borrow flag are subtracted from the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the subtracion.
Example: SBI 45

Increment register or memory by 1

INR R
M

The contents of the designated register or memory) are incremented by 1 and the result is stored in the same place. If the operand is a memory location, its location is specified by the contents of the HL registers.
Example: INR B or INR M

Increment register pair by 1

INX R

The contents of the designated register pair are incremented by 1 and the result is stored in the same place.
Example: INX H

Decrement register or memory by 1

DCR	R M	The contents of the designated register or memory are decremented by 1 and the result is stored in the same place. If the operand is a memory location, its location is specified by the contents of the HL registers. Example: DCR B or DCR M

Decrement register pair by 1

DCX	R	The contents of the designated register pair are decremented by 1 and the result is stored in the same place. Example: DCX H

Decimal adjust accumulator

DAA	none	The contents of the accumulator are changed from a binary value to two 4-bit binary coded decimal (BCD) digits. This is the only instruction that uses the auxiliary flag to perform the binary to BCD conversion, and the conversion procedure is described below. S, Z, AC, P, CY flags are altered to reflect the results of the operation. If the value of the low-order 4-bits in the accumulator is greater than 9 or if AC flag is set, the instruction adds 6 to the low-order four bits. If the value of the high-order 4-bits in the accumulator is greater than 9 or if the Carry flag is set, the instruction adds 6 to the high-order four bits. Example: DAA

BRANCHING INSTRUCTIONS

Opcode	Operand	Description
Jump unconditionally		
JMP	16-bit address	The program sequence is transferred to the memory location specified by the 16-bit address given in the operand. Example: JMP 2034 or JMP XYZ

Jump conditionally

Operand: 16-bit address

The program sequence is transferred to the memory location specified by the 16-bit address given in the operand based on the specified flag of the PSW as described below.
Example: JZ 2034 or JZ XYZ

Opcode	Description	Flag Status
JC	Jump on Carry	CY = 1
JNC	Jump on no Carry	CY = 0
JP	Jump on positive	S = 0
JM	Jump on minus	S = 1
JZ	Jump on zero	Z = 1
JNZ	Jump on no zero	Z = 0
JPE	Jump on parity even	P = 1
JPO	Jump on parity odd	P = 0

Unconditional subroutine call

CALL 16-bit address

The program sequence is transferred to the memory location specified by the 16-bit address given in the operand. Before the transfer, the address of the next instruction after CALL (the contents of the program counter) is pushed onto the stack.
Example: CALL 2034 or CALL XYZ

Call conditionally

Operand: 16-bit address

The program sequence is transferred to the memory location specified by the 16-bit address given in the operand based on the specified flag of the PSW as described below. Before the transfer, the address of the next instruction after the call (the contents of the program counter) is pushed onto the stack.
Example: CZ 2034 or CZ XYZ

Opcode	Description	Flag Status
CC	Call on Carry	CY = 1
CNC	Call on no Carry	CY = 0
CP	Call on positive	S = 0
CM	Call on minus	S = 1
CZ	Call on zero	Z = 1
CNZ	Call on no zero	Z = 0
CPE	Call on parity even	P = 1
CPO	Call on parity odd	P = 0

Return from subroutine unconditionally

RET none

The program sequence is transferred from the subroutine to the calling program. The two bytes from the top of the stack are copied into the program counter, and program execution begins at the new address.
Example: RET

Return from subroutine conditionally

Operand: none

The program sequence is transferred from the subroutine to the calling program based on the specified flag of the PSW as described below. The two bytes from the top of the stack are copied into the program counter, and program execution begins at the new address.
Example: RZ

Opcode	Description	Flag Status
RC	Return on Carry	CY = 1
RNC	Return on no Carry	CY = 0
RP	Return on positive	S = 0
RM	Return on minus	S = 1
RZ	Return on zero	Z = 1
RNZ	Return on no zero	Z = 0
RPE	Return on parity even	P = 1
RPO	Return on parity odd	P = 0

Load program counter with HL contents

PCHL none

The contents of registers H and L are copied into the program counter. The contents of H are placed as the high-order byte and the contents of L as the low-order byte.
Example: PCHL

Restart

RST 0-7

The RST instruction is equivalent to a 1-byte call instruction to one of eight memory locations depending upon the number. The instructions are generally used in conjunction with interrupts and inserted using external hardware. However these can be used as software instructions in a program to transfer program execution to one of the eight locations. The addresses are:

Instruction	Restart Address
RST 0	0000H
RST 1	0008H
RST 2	0010H
RST 3	0018H
RST 4	0020H
RST 5	0028H
RST 6	0030H
RST 7	0038H

The 8085 has four additional interrupts and these interrupts generate RST instructions internally and thus do not require any external hardware. These instructions and their Restart addresses are:

Interrupt	Restart Address
TRAP	0024H
RST 5.5	002CH
RST 6.5	0034H
RST 7.5	003CH

LOGICAL INSTRUCTIONS

Opcode Operand Description

Compare register or memory with accumulator

CMP R
M

The contents of the operand (register or memory) are compared with the contents of the accumulator. Both contents are preserved . The result of the comparison is shown by setting the flags of the PSW as follows:
if (A) < (reg/mem): carry flag is set, s=1
if (A) = (reg/mem): zero flag is set, s=0
if (A) > (reg/mem): carry and zero flags are reset, s=0
Example: CMP B or CMP M

Compare immediate with accumulator

CPI 8-bit data

The second byte (8-bit data) is compared with the contents of the accumulator. The values being compared remain unchanged. The result of the comparison is shown by setting the flags of the PSW as follows:
if (A) < data: carry flag is set, s=1
if (A) = data: zero flag is set, s=0
if (A) > data: carry and zero flags are reset, s=0
Example: CPI 89

Logical AND register or memory with accumulator

ANA R
M

The contents of the accumulator are logically ANDed with the contents of the operand (register or memory), and the result is placed in the accumulator. If the operand is a memory location, its address is specified by the contents of HL registers. S, Z, P are modified to reflect the result of the operation. CY is reset. AC is set.
Example: ANA B or ANA M

Logical AND immediate with accumulator

ANI 8-bit data

The contents of the accumulator are logically ANDed with the 8-bit data (operand) and the result is placed in the accumulator. S, Z, P are modified to reflect the result of the operation. CY is reset. AC is set.
Example: ANI 86

Exclusive OR register or memory with accumulator

XRA R
M

The contents of the accumulator are Exclusive ORed with the contents of the operand (register or memory), and the result is placed in the accumulator. If the operand is a memory location, its address is specified by the contents of HL registers. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.

Example: XRA B or XRA M

Exclusive OR immediate with accumulator

XRI 8-bit data

The contents of the accumulator are Exclusive ORed with the 8-bit data (operand) and the result is placed in the accumulator. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.

Example: XRI 86

Logical OR register or memory with accumulaotr

ORA R
M

The contents of the accumulator are logically ORed with the contents of the operand (register or memory), and the result is placed in the accumulator. If the operand is a memory location, its address is specified by the contents of HL registers. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.

Example: ORA B or ORA M

Logical OR immediate with accumulator

ORI 8-bit data

The contents of the accumulator are logically ORed with the 8-bit data (operand) and the result is placed in the accumulator. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.

Example: ORI 86

Rotate accumulator left

RLC none

Each binary bit of the accumulator is rotated left by one position. Bit D_7 is placed in the position of D_0 as well as in the Carry flag. CY is modified according to bit D_7. S, Z, P, AC are not affected.

Example: RLC

Rotate accumulator right

RRC none

Each binary bit of the accumulator is rotated right by one position. Bit D_0 is placed in the position of D_7 as well as in the Carry flag. CY is modified according to bit D_0. S, Z, P, AC are not affected.

Example: RRC

Rotate accumulator left through carry

RAL none

Each binary bit of the accumulator is rotated left by one position through the Carry flag. Bit D7 is placed in the Carry flag, and the Carry flag is placed in the least significant position D0. CY is modified according to bit D7. S, Z, P, AC are not affected.
Example: RAL

Rotate accumulator right through carry

RAR none

Each binary bit of the accumulator is rotated right by one position through the Carry flag. Bit D0 is placed in the Carry flag, and the Carry flag is placed in the most significant position D7. CY is modified according to bit D0. S, Z, P, AC are not affected.
Example: RAR

Complement accumulator

CMA none

The contents of the accumulator are complemented. No flags are affected.
Example: CMA

Complement carry

CMC none

The Carry flag is complemented. No other flags are affected.
Example: CMC

Set Carry

STC none

The Carry flag is set to 1. No other flags are affected.
Example: STC

CONTROL INSTRUCTIONS

Opcode Operand Description

No operation

NOP none

No operation is performed. The instruction is fetched and decoded. However no operation is executed.
Example: NOP

Halt and enter wait state

HLT none

The CPU finishes executing the current instruction and halts any further execution. An interrupt or reset is necessary to exit from the halt state.
Example: HLT

Disable interrupts

DI none

The interrupt enable flip-flop is reset and all the interrupts except the TRAP are disabled. No flags are affected.
Example: DI

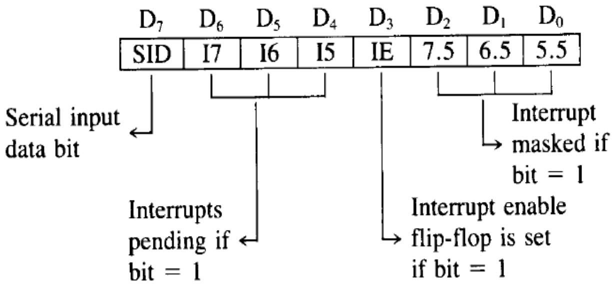
Enable interrupts

EI none

The interrupt enable flip-flop is set and all interrupts are enabled. No flags are affected. After a system reset or the acknowledgement of an interrupt, the interrupt enable flip-flop is reset, thus disabling the interrupts. This instruction is necessary to reenable the interrupts (except TRAP).
Example: EI

Read interrupt mask
RIM none

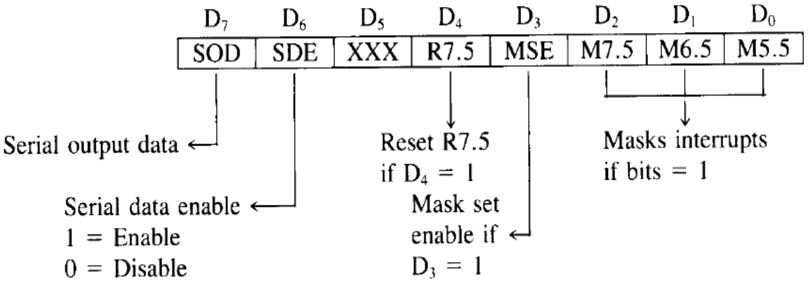
This is a multipurpose instruction used to read the status of interrupts 7.5, 6.5, 5.5 and read serial data input bit. The instruction loads eight bits in the accumulator with the following interpretations.
Example: RIM



D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0
SID
I7
I6
I5
IE
7.5
6.5
5.5
Serial input data bit
Interrupt masked if bit = 1
Interrupts pending if bit = 1
Interrupt enable flip-flop is set if bit = 1

Set interrupt mask
SIM none

This is a multipurpose instruction and used to implement the 8085 interrupts 7.5, 6.5, 5.5, and serial data output. The instruction interprets the accumulator contents as follows.
Example: SIM



D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0
SOD
SDE
XXX
R7.5
MSE
M7.5
M6.5
M5.5
Serial output data
Serial data enable 1 = Enable 0 = Disable
Reset R7.5 if D_4 = 1
Mask set enable if D_3 = 1
Masks interrupts if bits = 1

□ SOD—Serial Output Data: Bit D_7 of the accumulator is latched into the SOD output line and made available to a serial peripheral if bit D_6 = 1.

□ SDE—Serial Data Enable: If this bit = 1, it enables the serial output. To implement serial output, this bit needs to be enabled.

□ XXX—Don't Care

□ R7.5—Reset RST 7.5: If this bit = 1, RST 7.5 flip-flop is reset. This is an additional control to reset RST 7.5.

□ MSE—Mask Set Enable: If this bit is high, it enables the functions of bits D_2, D_1, D_0. This is a master control over all the interrupt masking bits. If this bit is low, bits D_2, D_1, and D_0 do not have any effect on the masks.

□ M7.5—D_2 = 0, RST 7.5 is enabled.
= 1, RST 7.5 is masked or disabled.

□ M6.5—D_1 = 0, RST 6.5 is enabled.
= 1, RST 6.5 is masked or disabled.

□ M5.5—D_0 = 0, RST 5.5 is enabled.
= 1, RST 5.5 is masked or disabled.

6. **INSTRUCTION EXECUTION AND TIMING DIAGRAM:**

Each instruction in 8085 microprocessor consists of two part- operation code (opcode) and operand. The opcode is a command such as ADD and the operand is an object to be operated on, such as a byte or the content of a register.

Instruction Cycle: The time taken by the processor to complete the execution of an instruction. An instruction cycle consists of one to six machine cycles.

Machine Cycle: The time required to complete one operation; accessing either the memory or I/O device. A machine cycle consists of three to six T-states.

T-State: Time corresponding to one clock period. It is the basic unit to calculate execution of instructions or programs in a processor.

To execute a program, 8085 performs various operations as:

- Opcode fetch
- Operand fetch
- Memory read/write
- I/O read/write

External communication functions are:

- Memory read/write
- I/O read/write
- Interrupt request acknowledge

Opcode Fetch Machine Cycle:

It is the first step in the execution of any instruction. The timing diagram of this cycle is given in Fig. 7.

The following points explain the various operations that take place and the signals that are changed during the execution of opcode fetch machine cycle:

T1 clock cycle

i. The content of PC is placed in the address bus; AD0 - AD7 lines contains lower bit address and A8 – A15 contains higher bit address.

ii. IO/$\overline{M}$ signal is low indicating that a memory location is being accessed. S1 and S0 also changed to the levels as indicated in Table 1.

iii. ALE is high, indicates that multiplexed AD0 – AD7 act as lower order bus.

T2 clock cycle

i. Multiplexed address bus is now changed to data bus.

ii. The $\overline{RD}$ signal is made low by the processor. This signal makes the memory device load the data bus with the contents of the location addressed by the processor.

T3 clock cycle

i. The opcode available on the data bus is read by the processor and moved to the instruction register.
ii. The $\overline{RD}$ signal is deactivated by making it logic 1.

T4 clock cycle

i. The processor decode the instruction in the instruction register and generate the necessary control signals to execute the instruction. Based on the instruction further operations such as fetching, writing into memory etc takes place.

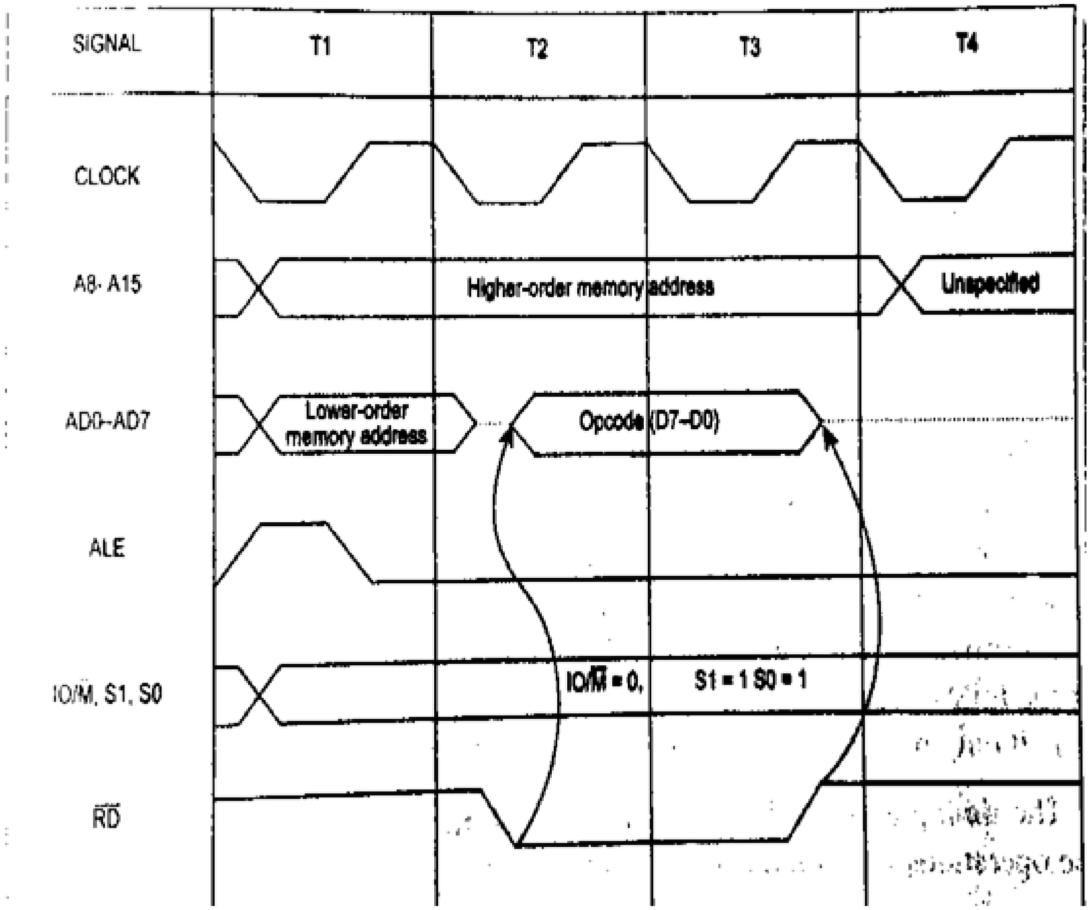


Fig. 7 Timing diagram for opcode fetch cycle

Memory Read Machine Cycle:

The memory read cycle is executed by the processor to read a data byte from memory. The machine cycle is exactly same to opcode fetch except: a) It has three T-states b) The S0 signal is set to 0. The timing diagram of this cycle is given in Fig. 8.

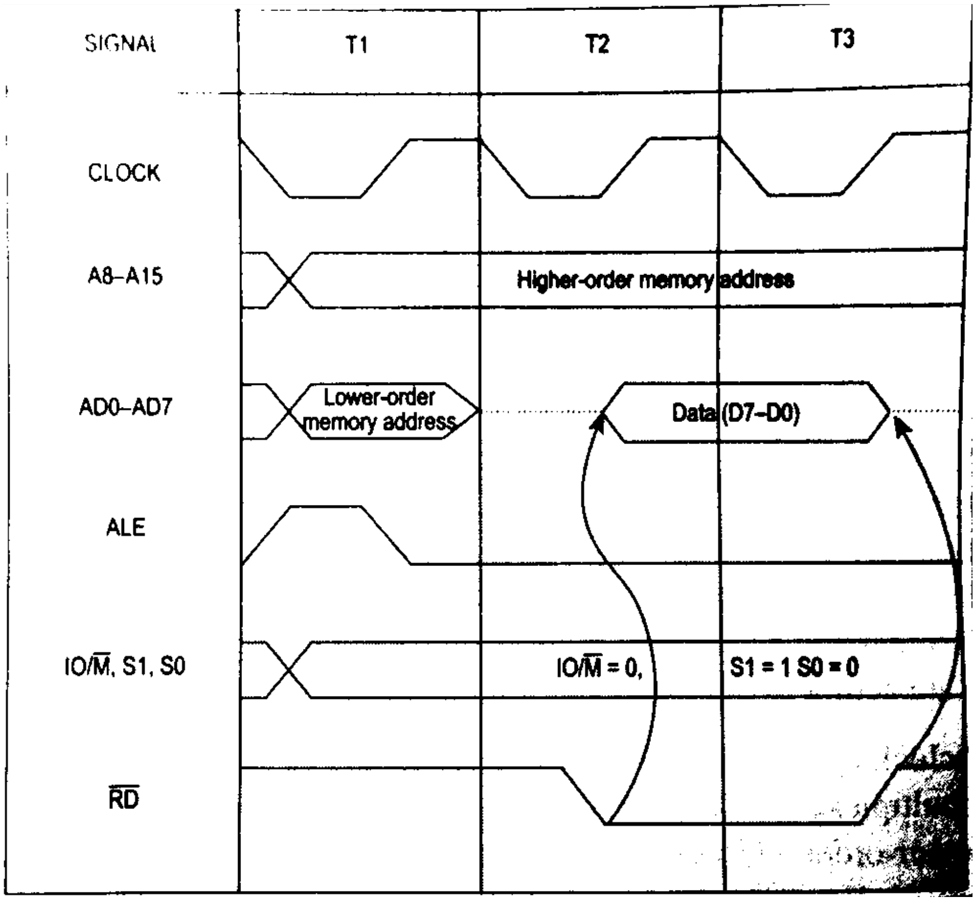


Fig. 8 Timing diagram for memory read machine cycle

Memory Write Machine Cycle:

The memory write cycle is executed by the processor to write a data byte in a memory location. The processor takes three T-states and $\overline{\text{WR}}$ signal is made low. The timing diagram of this cycle is given in Fig. 9.

I/O Read Cycle:

The I/O read cycle is executed by the processor to read a data byte from I/O port or from peripheral, which is I/O mapped in the system. The 8-bit port address is placed both in the lower and higher order address bus. The processor takes three T-states to execute this machine cycle. The timing diagram of this cycle is given in Fig. 10.

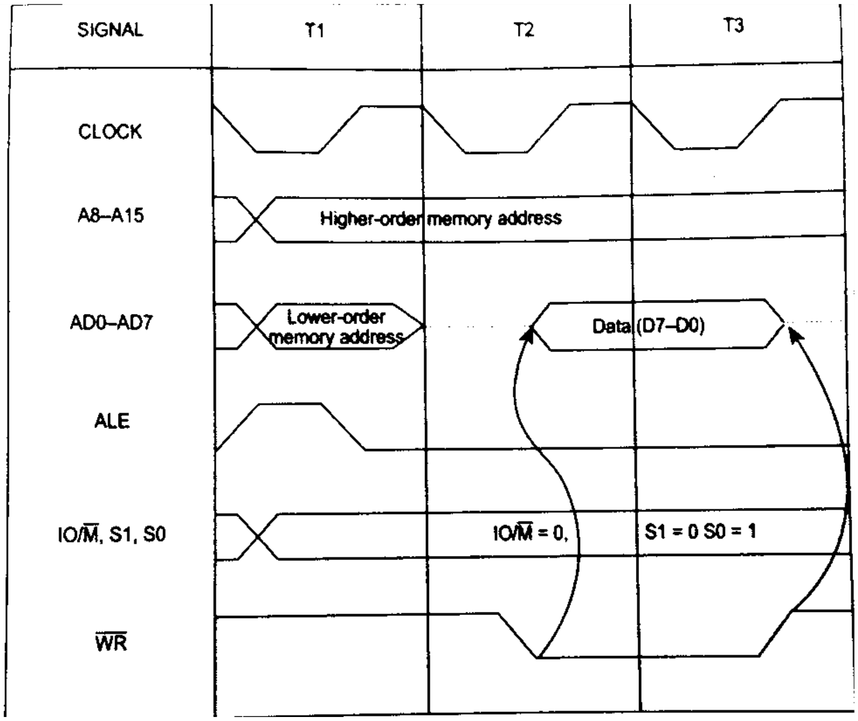


Fig. 9 Timing diagram for memory write machine cycle

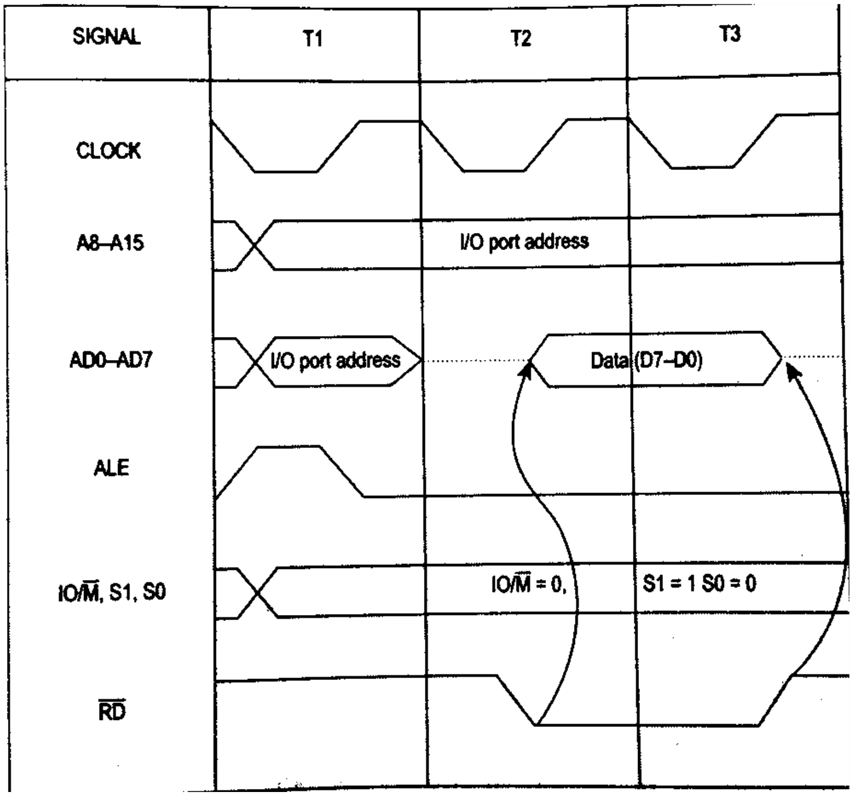


Fig. 10 Timing diagram I/O read machine cycle

I/O Write Cycle:

The I/O write cycle is executed by the processor to write a data byte to I/O port or to a peripheral, which is I/O mapped in the system. The processor takes three T-states to execute this machine cycle. The timing diagram of this cycle is given in Fig. 11.

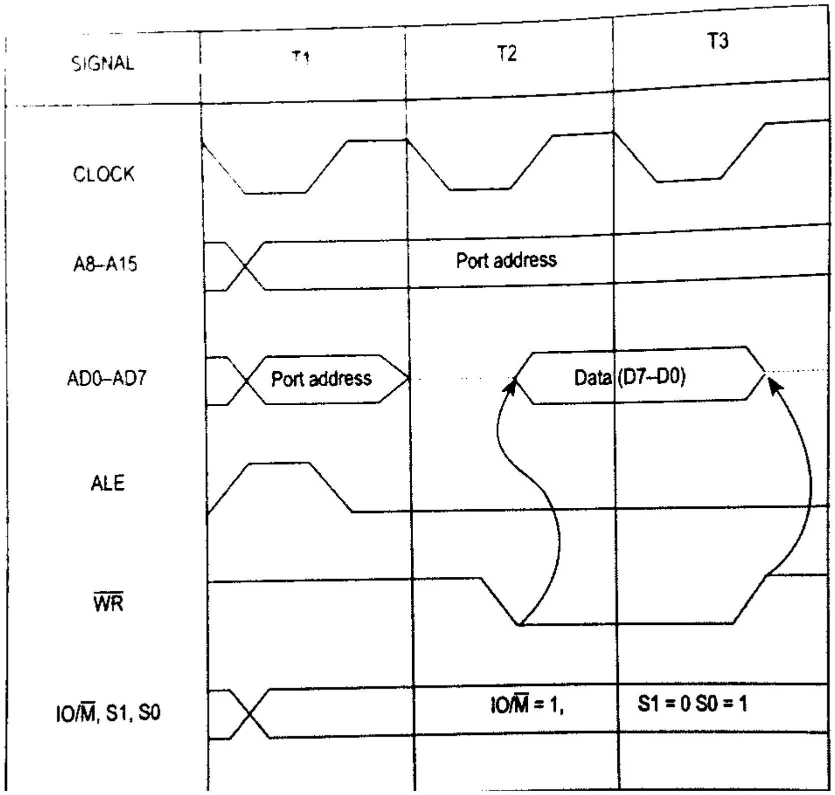


Fig. 11 Timing diagram I/O write machine cycle

Ex: Timing diagram for IN 80H.

The instruction and the corresponding codes and memory locations are given in Table 5.

Table 5 IN instruction

Address	Mnemonics	Opcode
800F	IN 80H	DB
8010		80

i. During the first machine cycle, the opcode DB is fetched from the memory, placed in the instruction register and decoded.
ii. During second machine cycle, the port address 80H is read from the next memory location.
iii. During the third machine cycle, the address 80H is placed in the address bus and the data read from that port address is placed in the accumulator.

The timing diagram is shown in Fig. 12.

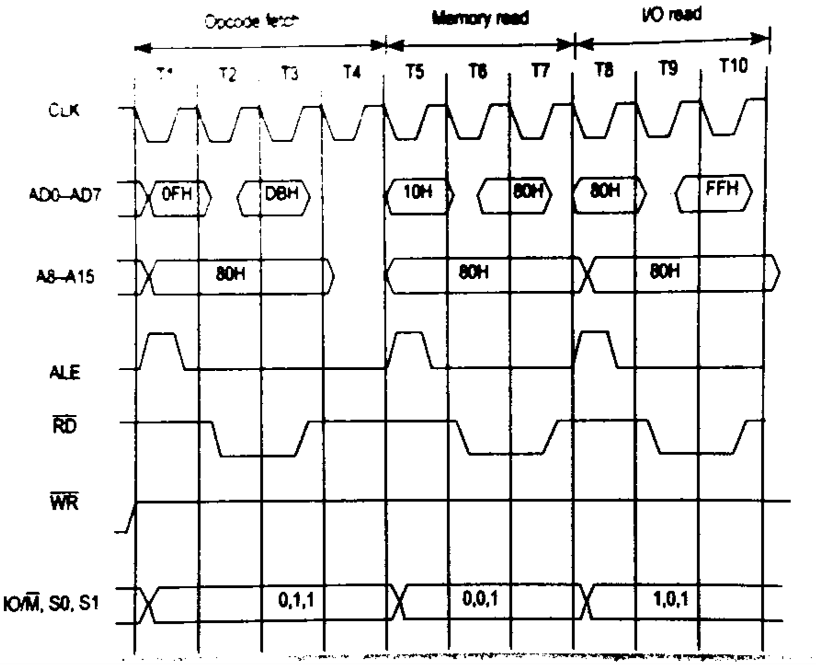


Fig. 12 Timing diagram for the IN instruction

7. 8085 INTERRUPTS

Interrupt Structure:

Interrupt is the mechanism by which the processor is made to transfer control from its current program execution to another program having higher priority. The interrupt signal may be given to the processor by any external peripheral device.

The program or the routine that is executed upon interrupt is called interrupt service routine (ISR). After execution of ISR, the processor must return to the interrupted program. Key features in the interrupt structure of any microprocessor are as follows:

i. Number and types of interrupt signals available.
ii. The address of the memory where the ISR is located for a particular interrupt signal. This address is called interrupt vector address (IVA).
iii. Masking and unmasking feature of the interrupt signals.
iv. Priority among the interrupts.
v. Timing of the interrupt signals.
vi. Handling and storing of information about the interrupt program (status information).

Types of Interrupts:

Interrupts are classified based on their maskability, IVA and source. They are classified as:

i. Vectored and Non-Vectored Interrupts
   - Vectored interrupts require the IVA to be supplied by the external device that gives the interrupt signal. This technique is vectoring, is implemented in number of ways.
   - Non-vectored interrupts have fixed IVA for ISRs of different interrupt signals.

ii. Maskable and Non-Maskable Interrupts
   - Maskable interrupts are interrupts that can be blocked. Masking can be done by software or hardware means.
   - Non-maskable interrupts are interrupts that are always recognized; the corresponding ISRs are executed.

iii. Software and Hardware Interrupts
   - Software interrupts are special instructions, after execution transfer the control to predefined ISR.
   - Hardware interrupts are signals given to the processor, for recognition as an interrupt and execution of the corresponding ISR.

Interrupt Handling Procedure:

The following sequence of operations takes place when an interrupt signal is recognized:

i. Save the PC content and information about current state (flags, registers etc) in the stack.
ii. Load PC with the beginning address of an ISR and start to execute it.
iii. Finish ISR when the return instruction is executed.
iv. Return to the point in the interrupted program where execution was interrupted.

Interrupt Sources and Vector Addresses in 8085:

Software Interrupts:

8085 instruction set includes eight software interrupt instructions called Restart (RST) instructions. These are one byte instructions that make the processor execute a subroutine at predefined locations. Instructions and their vector addresses are given in Table 6.

Table 6 Software interrupts and their vector addresses

Instruction	Machine hex code	Interrupt Vector Address
RST 0	C7	0000H
RST 1	CF	0008H
RST 2	D7	0010H
RST 3	DF	0018H
RST 4	E7	0020H
RST 5	EF	0028H
RST 6	F7	0030H
RST 7	FF	0032H

The software interrupts can be treated as CALL instructions with default call locations. The concept of priority does not apply to software interrupts as they are inserted into the program as instructions by the programmer and executed by the processor when the respective program lines are read.

Hardware Interrupts and Priorities:

8085 have five hardware interrupts – INTR, RST 5.5, RST 6.5, RST 7.5 and TRAP. Their IVA and priorities are given in Table 7.

Table 7 Hardware interrupts of 8085

Interrupt	Interrupt vector address	Maskable or non-maskable	Edge or level triggered	priority
TRAP	0024H	Non-makable	Level	1
RST 7.5	003CH	Maskable	Rising edge	2
RST 6.5	0034H	Maskable	Level	3
RST 5.5	002CH	Maskable	Level	4
INTR	Decided by hardware	Maskable	Level	5

Masking of Interrupts:

Masking can be done for four hardware interrupts INTR, RST 5.5, RST 6.5, and RST 7.5. The masking of 8085 interrupts is done at different levels. Fig. 13 shows the organization of hardware interrupts in the 8085.

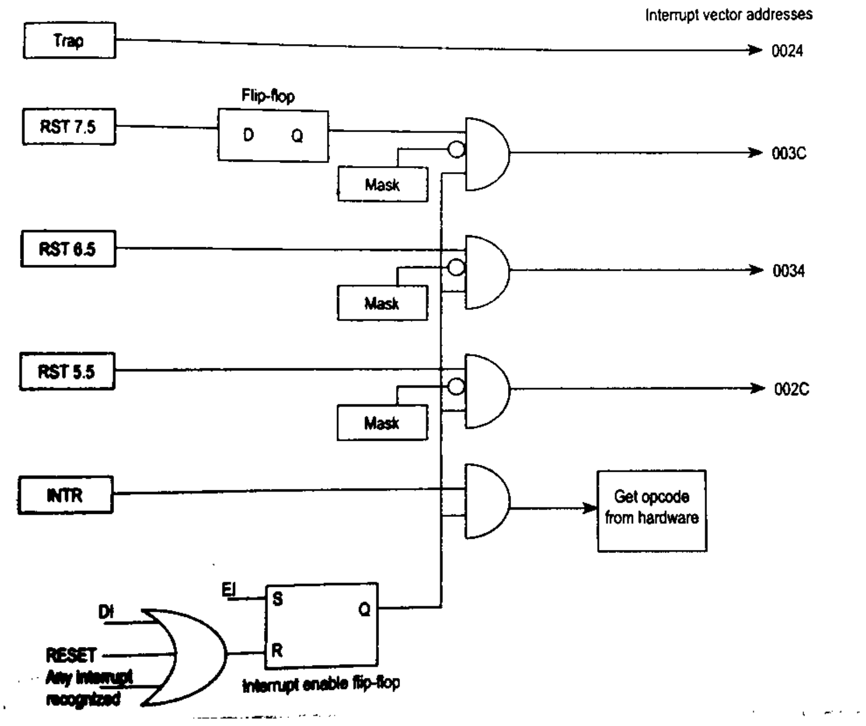


Fig. 13 Interrupt structure of 8085

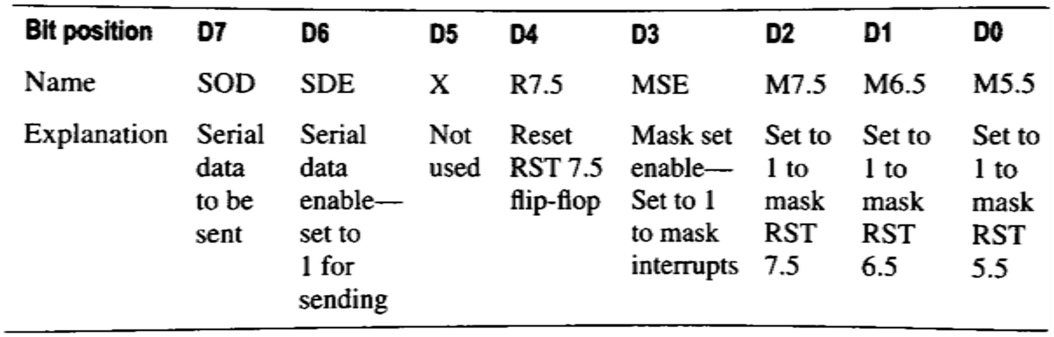
The Fig. 13 is explained by the following five points:

i. The maskable interrupts are by default masked by the Reset signal. So no interrupt is recognized by the hardware reset.
ii. The interrupts can be enabled by the EI instruction.
iii. The three RST interrupts can be selectively masked by loading the appropriate word in the accumulator and executing SIM instruction. This is called software masking.
iv. All maskable interrupts are disabled whenever an interrupt is recognized.
v. All maskable interrupts can be disabled by executing the DI instruction.

RST 7.5 alone has a flip-flop to recognize edge transition. The DI instruction reset interrupt enable flip-flop in the processor and the interrupts are disabled. To enable interrupts, EI instruction has to be executed.

SIM Instruction:

The SIM instruction is used to mask or unmask RST hardware interrupts. When executed, the SIM instruction reads the content of accumulator and accordingly mask or unmask the interrupts. The format of control word to be stored in the accumulator before executing SIM instruction is as shown in Fig. 14.



Bit position	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Name	SOD	SDE	X	R7.5	MSE	M7.5	M6.5	M5.5
Explanation	Serial data to be sent	Serial data enable—set to 1 for sending	Not used	Reset RST 7.5 flip-flop	Mask set enable—Set to 1 to mask interrupts	Set to 1 to mask RST 7.5	Set to 1 to mask RST 6.5	Set to 1 to mask RST 5.5

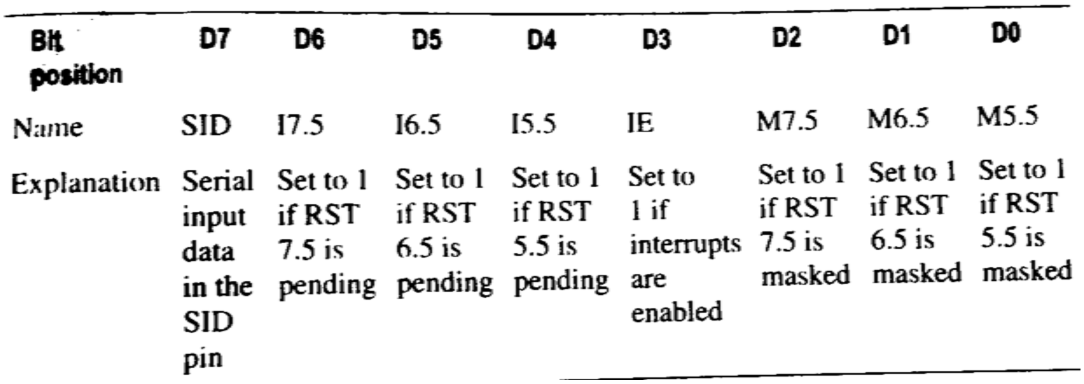
Fig. 14 Accumulator bit pattern for SIM instruction

In addition to masking interrupts, SIM instruction can be used to send serial data on the SOD line of the processor. The data to be send is placed in the MSB bit of the accumulator and the serial data output is enabled by making D6 bit to 1.

RIM Instruction:

RIM instruction is used to read the status of the interrupt mask bits. When RIM instruction is executed, the accumulator is loaded with the current status of the interrupt masks and the pending interrupts. The format and the meaning of the data stored in the accumulator after execution of RIM instruction is shown in Fig. 15.

In addition RIM instruction is also used to read the serial data on the SID pin of the processor. The data on the SID pin is stored in the MSB of the accumulator after the execution of the RIM instruction.



Bit position	D7	D6	D5	D4	D3	D2	D1	D0
Name	SID	I7.5	I6.5	I5.5	IE	M7.5	M6.5	M5.5
Explanation	Serial input data in the SID pin	Set to 1 if RST 7.5 is pending	Set to 1 if RST 6.5 is pending	Set to 1 if RST 5.5 is pending	Set to 1 if interrupts are enabled	Set to 1 if RST 7.5 is masked	Set to 1 if RST 6.5 is masked	Set to 1 if RST 5.5 is masked

Fig. 15 Accumulator bit pattern after execution of RIM instruction

Ex: Write an assembly language program to enables all the interrupts in 8085 after reset.

```
EI            : Enable interrupts

MVI A, 08H    : Unmask the interrupts

SIM           : Set the mask and unmask using SIM instruction
```

Timing of Interrupts:

The interrupts are sensed by the processor one cycle before the end of execution of each instruction. An interrupts signal must be applied long enough for it to be recognized. The longest instruction of the 8085 takes 18 clock periods. So, the interrupt signal must be applied for at least 17.5 clock periods. This decides the minimum pulse width for the interrupt signal.

The maximum pulse width for the interrupt signal is decided by the condition that the interrupt signal must not be recognized once again. This is under the control of the programmer.

QUESTIONS:

1. What is the function of a microprocessor in a system?
2. Why is the data bus in 8085 bidirectional?
3. How does microprocessor differentiate between data and instruction?
4. How long would the processor take to execute the instruction LDA 1753H if the T-state duration is 2µs?
5. Draw the timing diagram of the instruction LDAX B.
6. Sketch and explain the various pins of the 8085.
7. Explain direct addressing mode of 8085 with an example?
8. Draw and explain the timing diagram of the instruction IN 82H.
9. What is meant by 'priority of the interrupts'? Explain the operation of the interrupts structure of the 8085, with the help of a circuit diagram.
10. Explain the bit pattern for SIM instruction. Write the assembly language program lines to enable all the interrupts in the 8085 after reset.
11. Write the logical instructions which affect and which does not affect flags in 8085.
12. Write an ALP in 8085 MPU to reject all the negative readings and add all the positive reading from a set of ten reading stored in memory locations starting at XX60H. When the sum exceeds eight bits produce output FFH to PORT1 to indicate overload otherwise display the sum.
13. Write an ALP in 8085 to eliminate the blanks (bytes with zero value) from a string of eight data bytes. Use two memory pointers: one to get a byte and the other to store the byte.
14. Design an up-down counter to count from 0 to 9 and 9 to 0 continuously with a 1.5 second delay between each count, and display the count at one of the output ports.

MODULE: 2

1. **INTERFACING MEMORY AND I/O DEVICES WITH 8085**

The programs and data that are executed by the microprocessor have to be stored in ROM/EPROM and RAM, which are basically semiconductor memory chips. The programs and data that are stored in ROM/EPROM are not erased even when power supply to the chip is removed. Hence, they are called non-volatile memory. They can be used to store permanent programs.

In a RAM, stored programs and data are erased when the power supply to the chip is removed. Hence, RAM is called volatile memory. RAM can be used to store programs and data that include, programs written during software development for a microprocessor based system, program written when one is learning assembly language programming and data enter while testing these programs.

Input and output devices, which are interfaced with 8085, are essential in any microprocessor based system. They can be interfaced using two schemes: I/O mapped I/O and memory-mapped I/O. In the I/O mapped I/O scheme, the I/O devices are treated differently from memory. In the memory-mapped I/O scheme, each I/O device is assumed to be a memory location.

2. INTERFACING MEMORY CHIPS WITH 8085

8085 has 16 address lines (A0 - A15), hence a maximum of 64 KB (= 2^{16} bytes) of memory locations can be interfaced with it. The memory address space of the 8085 takes values from 0000H to FFFFH.

The 8085 initiates set of signals such as $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ when it wants to read from and write into memory. Similarly, each memory chip has signals such as $\overline{CE}$ or $\overline{CS}$ (chip enable or chip select), $\overline{OE}$ or $\overline{RD}$ (output enable or read) and $\overline{WE}$ or $\overline{WR}$ (write enable or write) associated with it.

Generation of Control Signals for Memory:

When the 8085 wants to read from and write into memory, it activates $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals as shown in Table 8.

Table 8 Status of $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals during memory read and write operations

$IO/\overline{M}$	$\overline{RD}$	$\overline{WR}$	Operation
0	0	1	8085 reads data from memory
0	1	0	8085 writes data into memory

Using $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals, two control signals $\overline{MEMR}$ (memory read) and $\overline{MEMW}$ (memory write) are generated. Fig. 16 shows the circuit used to generate these signals.

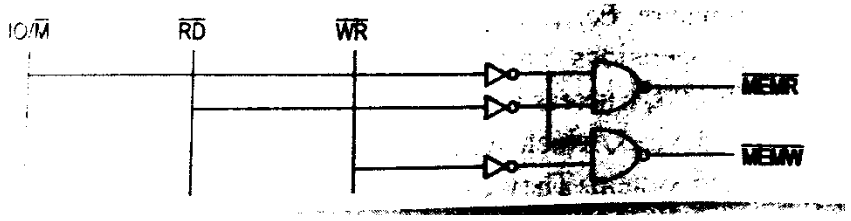


Fig. 16 Circuit used to generate $\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$ signals

When is $\text{IO}/\overline{\text{M}}$ high, both memory control signals are deactivated irrespective of the status of $\overline{\text{RD}}$ and $\overline{\text{WR}}$ signals.

Ex: Interface an IC 2764 with 8085 using NAND gate address decoder such that the address range allocated to the chip is 0000H – 1FFFH.

Specification of IC 2764:

- 8 KB (8 x 2^{10} byte) EPROM chip
- 13 address lines (2^{13} bytes = 8 KB)

Interfacing:

- 13 address lines of IC are connected to the corresponding address lines of 8085.
- Remaining address lines of 8085 are connected to address decoder formed using logic gates, the output of which is connected to the $\overline{\text{CE}}$ pin of IC.
- Address range allocated to the chip is shown in Table 9.
- Chip is enabled whenever the 8085 places an address allocated to EPROM chip in the address bus. This is shown in Fig. 17.

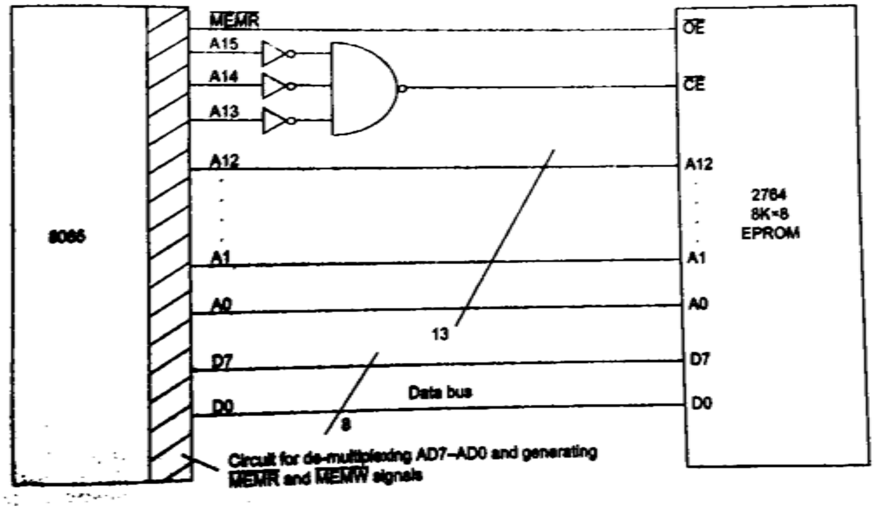
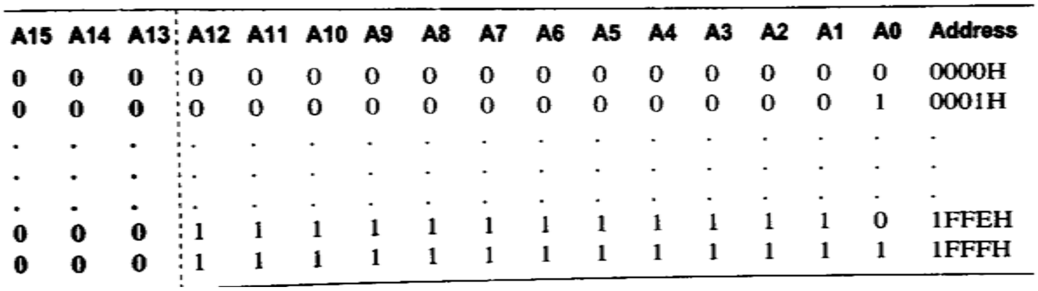


Fig. 17 Interfacing IC 2764 with the 8085

Table 9 Address allocated to IC 2764



A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Address
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0001H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1FFEH
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFFH

Ex: Interface a 6264 IC (8K x 8 RAM) with the 8085 using NAND gate decoder such that the starting address assigned to the chip is 4000H.

Specification of IC 6264:

- 8K x 8 RAM
- 8 KB = 2^{13} bytes
- 13 address lines

The ending address of the chip is 5FFFH (since 4000H + 1FFFH = 5FFFH). When the address 4000H to 5FFFH are written in binary form, the values in the lines A15, A14, A13 are 0, 1 and 0 respectively. The NAND gate is designed such that when the lines A15 and A13 carry 0 and A14 carries 1, the output of the NAND gate is 0. The NAND gate output is in turn connected to the $\overline{\text{CE1}}$ pin of the RAM chip. A NAND output of 0 selects the RAM chip for read or write operation, since CE2 is already 1 because of its connection to +5V. Fig. 18 shows the interfacing of IC 6264 with the 8085.

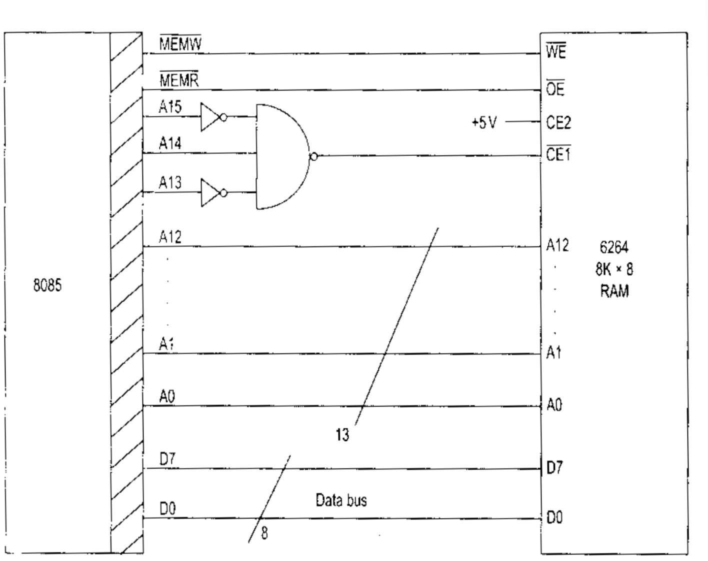


Fig. 18 Interfacing 6264 IC with the 8085

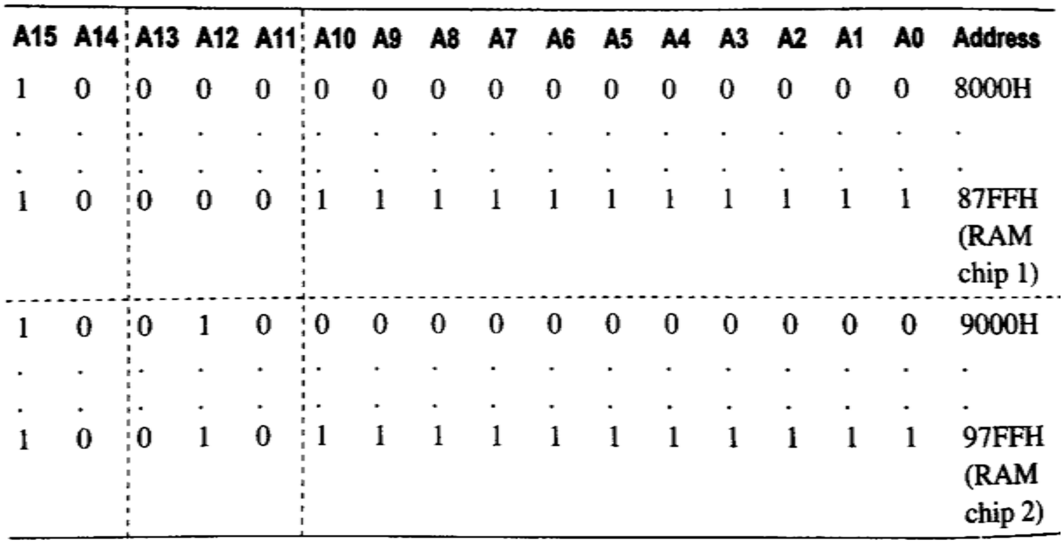
Ex: Interface two 6116 ICs with the 8085 using 74LS138 decoder such that the starting addresses assigned to them are 8000H and 9000H, respectively.

Specification of IC 6116:

- 2 K x 8 RAM
- 2 KB = 2^{11} bytes
- 11 address lines

6116 has 11 address lines and since 2 KB, therefore ending addresses of 6116 chip 1 is and chip 2 are 87FFH and 97FFH, respectively. Table 10 shows the address range of the two chips.

Table 10 Address range for IC 6116



A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Address
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	87FFH (RAM chip 1)
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	9000H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	97FFH (RAM chip 2)

Interfacing:

- Fig. 19 shows the interfacing.
- A0 – A10 lines of 8085 are connected to 11 address lines of the RAM chips.
- Three address lines of 8085 having specific value for a particular RAM are connected to the three select inputs (C, B and A) of 74LS138 decoder.
- Table 10 shows that A13=A12=A11=0 for the address assigned to RAM 1 and A13=0, A12=1 and A11=0 for the address assigned to RAM 2.
- Remaining lines of 8085 which are constant for the address range assigned to the two RAM are connected to the enable inputs of decoder.
- When 8085 places any address between 8000H and 87FFH in the address bus, the select inputs C, B and A of the decoder are all 0. The Y0 output of the decoder is also 0, selecting RAM 1.
- When 8085 places any address between 9000H and 97FFH in the address bus, the select inputs C, B and A of the decoder are 0, 1 and 0. The Y2 output of the decoder is also 0, selecting RAM 2.

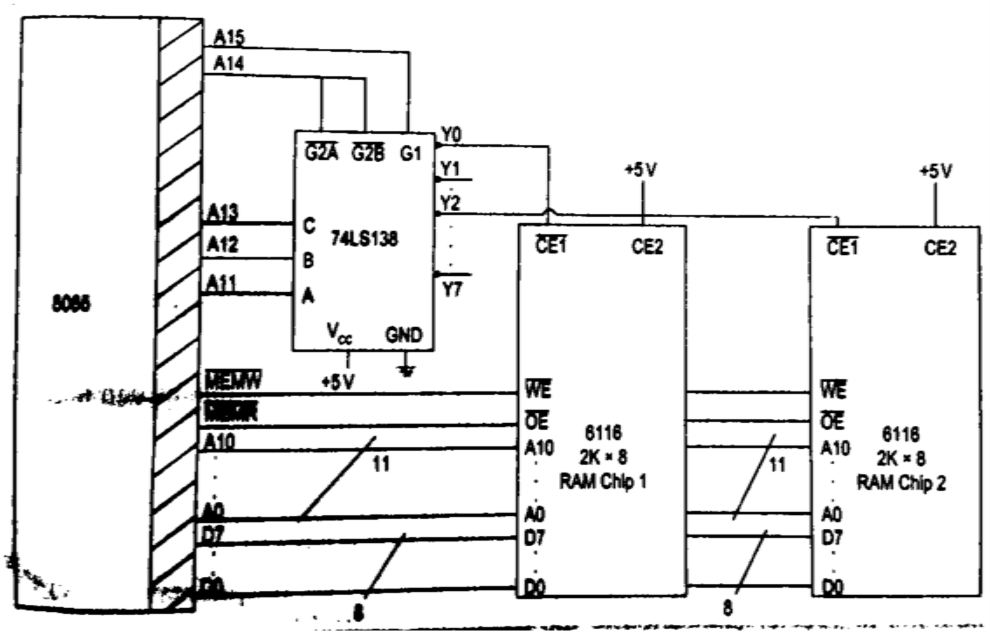


Fig. 19 Interfacing two 6116 RAM chips using 74LS138 decoder

3. PERIPHERAL MAPPED I/O INTERFACING

In this method, the I/O devices are treated differently from memory chips. The control signals I/O read ($\overline{IOR}$) and I/O write ($\overline{IOW}$), which are derived from the $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals of the 8085, are used to activate input and output devices, respectively. Generation of these control signals is shown in Fig. 20. Table 11 shows the status of $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals during I/O read and I/O write operation.

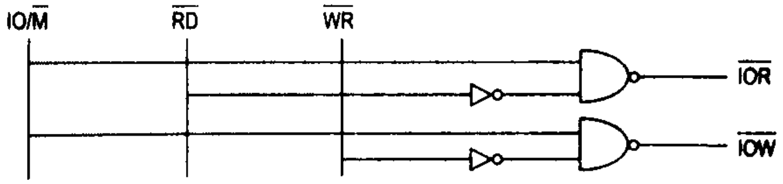


Fig. 20 Generation of $\overline{IOR}$ and $\overline{IOW}$ signals

IN instruction is used to access input device and OUT instruction is used to access output device. Each I/O device is identified by a unique 8-bit address assigned to it. Since the control signals used to access input and output devices are different, and all I/O device use 8-bit address, a maximum of 256 (2^8) input devices and 256 output devices can be interfaced with 8085.

Table 11 Status of $\overline{IOR}$ and $\overline{IOW}$ signals in 8085.

$IO/\overline{M}$	$\overline{RD}$	$\overline{WR}$	$\overline{IOR}$	$\overline{IOW}$	Operation
1	0	1	0	1	I/O read operation
1	1	0	1	0	I/O write operation
0	X	X	1	1	Memory read or write operation

Ex: Interface an 8-bit DIP switch with the 8085 such that the address assigned to the DIP switch if F0H.

IN instruction is used to get data from DIP switch and store it in accumulator. Steps involved in the execution of this instruction are:

i. Address F0H is placed in the lines A0 – A7 and a copy of it in lines A8 – A15.
ii. The $\overline{IOR}$ signal is activated ($\overline{IOR}$ = 0), which makes the selected input device to place its data in the data bus.
iii. The data in the data bus is read and store in the accumulator.

Fig. 21 shows the interfacing of DIP switch.

A7	A6	A5	A4	A3	A2	A1	A0	
1	1	1	1	0	0	0	0	= F0H

A0 – A7 lines are connected to a NAND gate decoder such that the output of NAND gate is 0. The output of NAND gate is ORed with the $\overline{IOR}$ signal and the output of OR gate is connected to $\overline{1G}$ and $\overline{2G}$ of the 74LS244. When 74LS244 is enabled, data from the DIP switch is placed on the data bus of the 8085. The 8085 read data and store in the accumulator. Thus data from DIP switch is transferred to the accumulator.

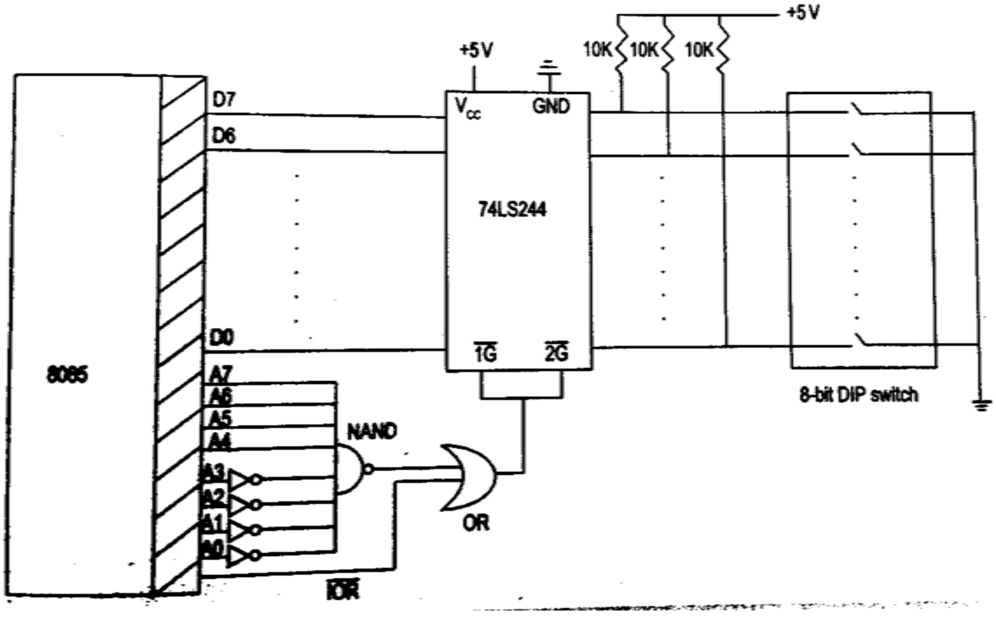


Fig. 21 interfacing of 8-bit DIP switch with 8085

4. MEMORY MAPPED I/O INTERFACING

In memory-mapped I/O, each input or output device is treated as if it is a memory location. The $\overline{MEMR}$ and $\overline{MEMW}$ control signals are used to activate the devices. Each input or output device is identified by unique 16-bit address, similar to 16-bit address assigned to memory location. All memory related instruction like LDA 2000H, LDAX B, MOV A, M can be used.

Since the I/O devices use some of the memory address space of 8085, the maximum memory capacity is lesser than 64 KB in this method.

Ex: Interface an 8-bit DIP switch with the 8085 using logic gates such that the address assigned to it is F0F0H.

Since a 16-bit address has to be assigned to a DIP switch, the memory-mapped I/O technique must be used. Using LDA F0F0H instruction, the data from the 8-bit DIP switch can be transferred to the accumulator. The steps involved are:

i. The address F0F0H is placed in the address bus A0 – A15.
ii. The $\overline{\text{MEMR}}$ signal is made low for some time.
iii. The data in the data bus is read and stored in the accumulator.

Fig. 22 shows the interfacing diagram.

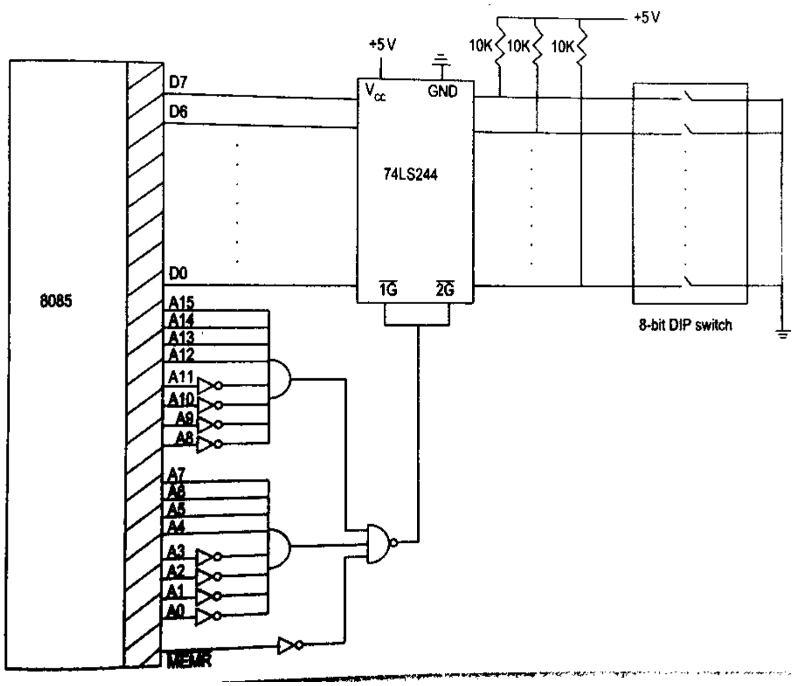


Fig. 22 Interfacing 8-bit DIP switch with 8085

When 8085 executes the instruction LDA F0F0H, it places the address F0F0H in the address lines A0 – A15 as:

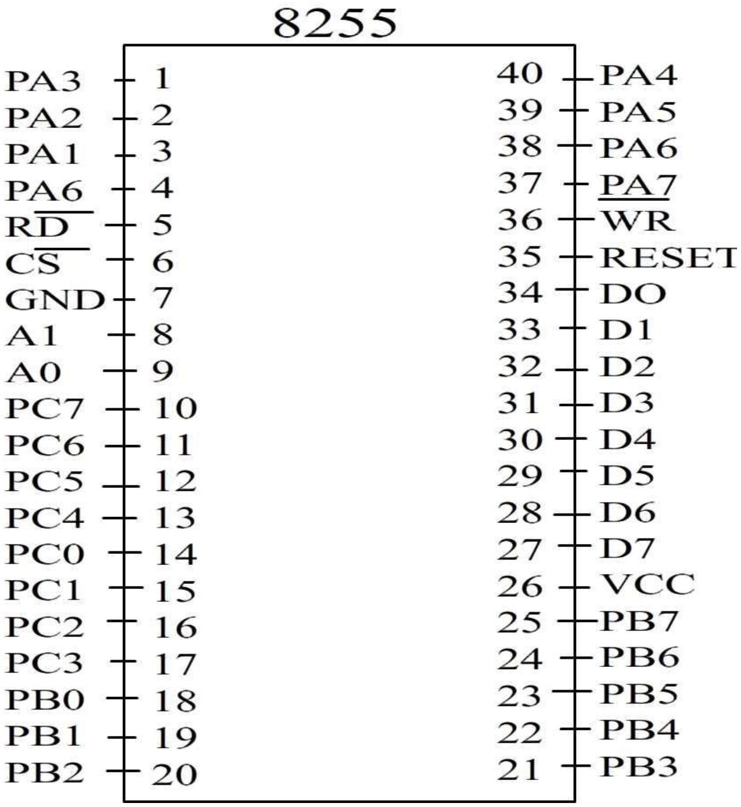
A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	= F0F0H

The address lines are connected to AND gates. The output of these gates along with $\overline{\text{MEMR}}$ signal are connected to a NAND gate, so that when the address F0F0H is placed in the address bus and $\overline{\text{MEMR}}$ = 0 its output becomes 0, thereby enabling the buffer 74LS244. The data from the DIP switch is placed in the 8085 data bus. The 8085 reads the data from the data bus and stores it in the accumulator.

INTEL 8255: (Programmable Peripheral Interface)

The 8255A is a general purpose programmable I/O device designed for use with Intel microprocessors. It consists of three 8-bit bidirectional I/O ports (24I/O lines) that can be configured to meet different system I/O needs. The three ports are PORT A, PORT B & PORT C. Port A contains one 8-bit output latch/buffer and one 8-bit input buffer. Port B is same as PORT A or PORT B. However, PORT C can be split into two parts PORT C lower (PC_0-PC_3) and PORT C upper (PC_7-PC_4) by the control word. The three ports are divided in two groups Group A (PORT A and upper PORT C) Group B (PORT B and lower PORT C). The two groups can be programmed in three different modes. In the first mode (mode 0), each group may be programmed in either input mode or output mode (PORT A, PORT B, PORT C lower, PORT C upper). In mode 1, the second's mode, each group may be programmed to have 8-lines of input or output (PORT A or PORT B) of the remaining 4-lines (PORT C lower or PORT C upper) 3-lines are used for hand shaking and interrupt control signals. The third mode of operation (mode 2) is a bidirectional bus mode which uses 8-line (PORT A only for a bidirectional bus and five lines (PORT C upper 4 lines and borrowing one from other group) for handshaking.

The 8255 is contained in a 40-pin package, whose pin out is shown below:



PIN Names

RESET – Reset input

$\overline{CS}$ - Chip selected

$\overline{RD}$ - Read input

$\overline{WR}$ - Write input

A_0 A_1 – Port Address

PA_7 – PA_0 – PORT A

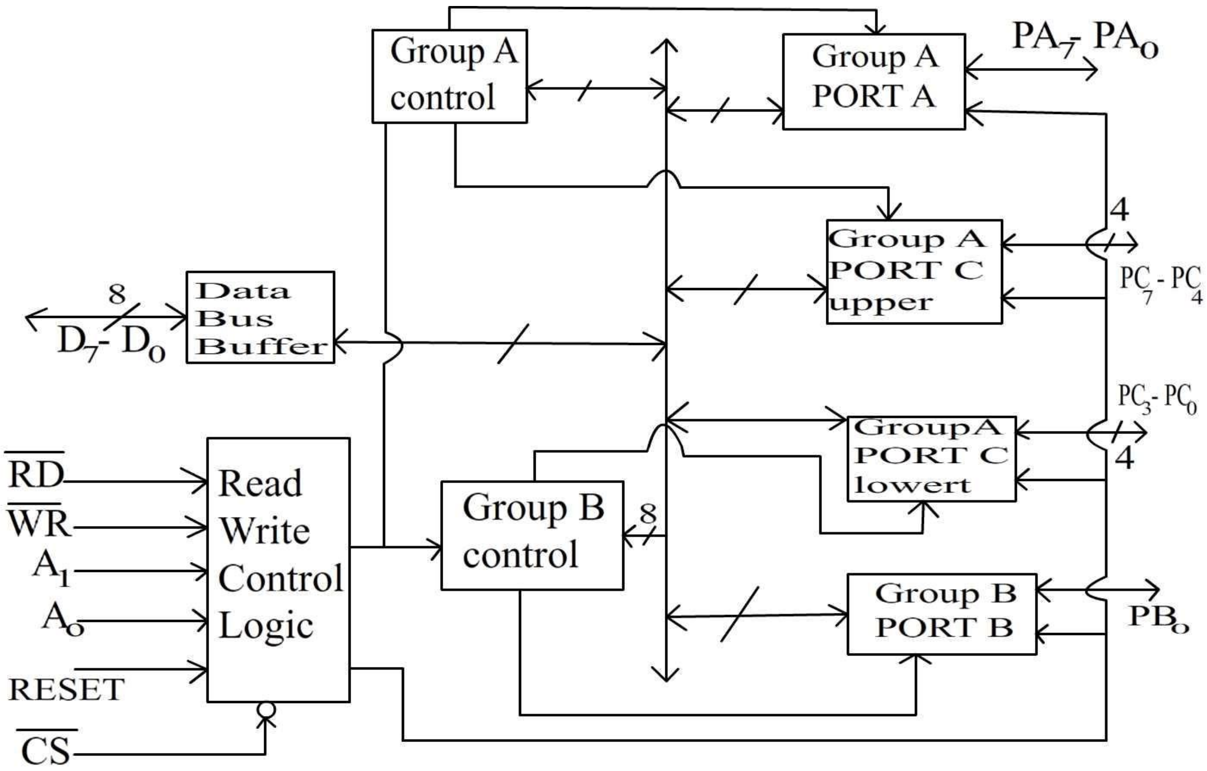
PB_7 – PB_0 – PORT B

PC_7 – PC_0 – PORT C

VCC - +5v

GND - Ground

The block diagram is shown below:



Functional Description:

This support chip is a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. It is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer:

It is a tri-state 8-bit buffer used to interface the chip to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer. The data lines are connected to BDB of μp.

Read/Write and logic control:

The function of this block is to control the internal operation of the device and to control the transfer of data and control or status words. It accepts inputs from the CPU address and control buses and in turn issues command to both the control groups.

$\overline{CS}$ Chip Select:

A low on this input selects the chip and enables the communication between the 8255 A & the CPU. It is connected to the output of address decode circuitry to select the device when it $\overline{RD}$ (Read). A low on this input enables the 8255 to send the data or status information to the CPU on the data bus.

$\overline{WR}$ (Write):

A low on this input pin enables the CPU to write data or control words into the 8255 A.

A_1, A_0 port select:

These input signals, in conjunction with the $\overline{RD}$ and $\overline{WR}$ inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus (A_0 and A_1).

Following Table gives the basic operation,

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Input operation
0	0	0	1	0	PORT A → Data bus
0	1	0	1	0	PORT B → Data bus
1	0	0	1	0	PORT C → Data bus
0	0	1	0	0	Output operation Data bus → PORT A
0	1	1	0	0	Data bus → PORT B
1	0	1	0	0	Data bus → PORT C
1	1	1	0	0	Data bus → control

All other states put data bus into tri-state/illegal condition.

RESET:

A high on this input pin clears the control register and all ports (A, B & C) are initialized to input mode. This is connected to RESET OUT of 8255. This is done to prevent destruction of circuitry connected to port lines. If port lines are initialized as output after a power up or reset, the port might try to output into the output of a device connected to same inputs might destroy one or both of them.

PORTs A, B and C:

The 8255A contains three 8-bit ports (A, B and C). All can be configured in a variety of functional characteristic by the system software.

PORTA:

One 8-bit data output latch/buffer and one 8-bit data input latch.

PORT B:

One 8-bit data output latch/buffer and one 8-bit data input buffer.

PORT C:

One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signals inputs in conjunction with ports A and B.

Group A & Group B control:

The functional configuration of each port is programmed by the system software. The control words outputted by the CPU configure the associated ports of the each of the two groups. Each control block accepts command from Read/Write content logic receives control words from the internal data bus and issues proper commands to its associated ports.

Control Group A – Port A & Port C upper

Control Group B – Port B & Port C lower

The control word register can only be written into No read operation if the control word register is allowed.

Operation Description:

Mode selection:

There are three basic modes of operation that can be selected by the system software.

Mode 0: Basic Input/output

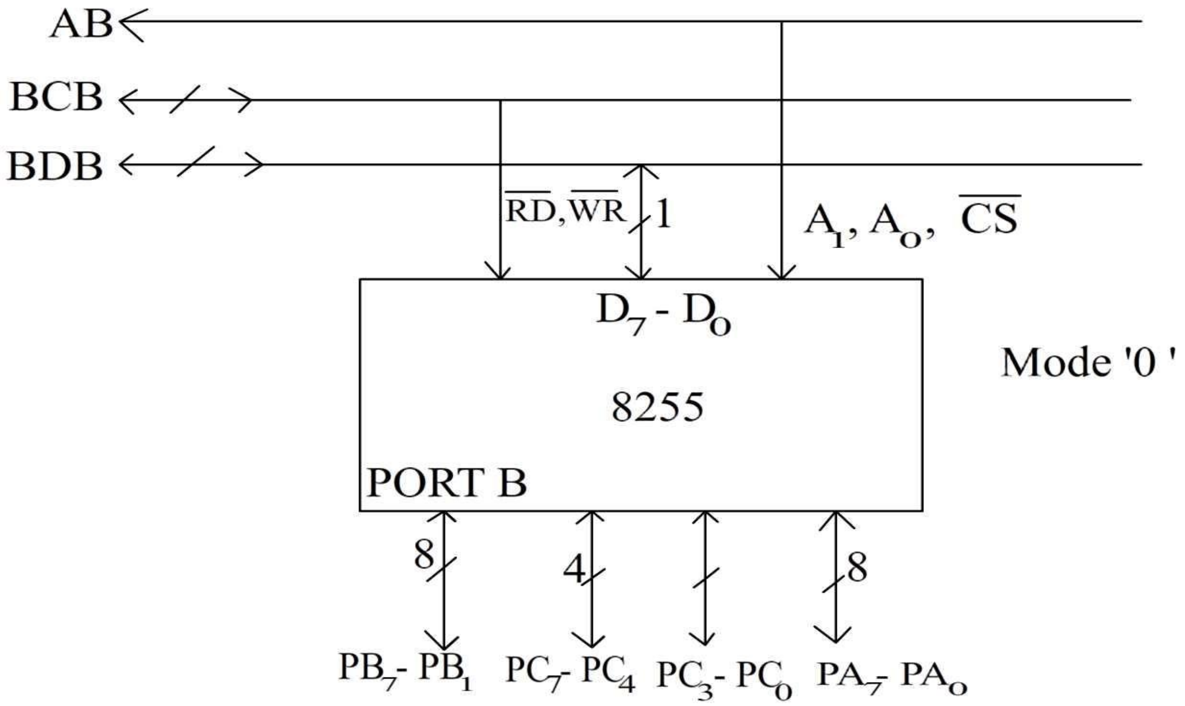
Mode 1: Strobes Input/output

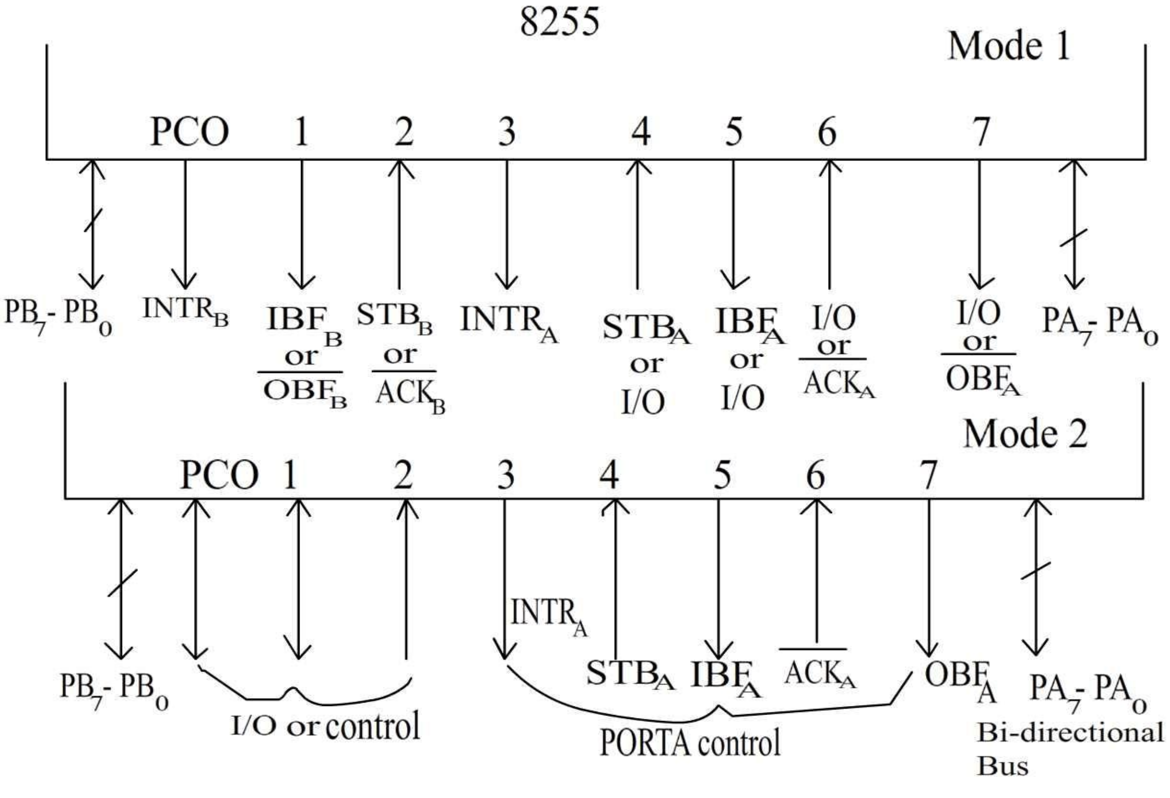
Mode 2: Bi-direction bus.

When the reset input goes HIGH all poets are set to mode’0’ as input which means all 24 lines are in high impedance state and can be used as normal input. After the reset is removed the 8255A remains in the input mode with no additional initialization. During the execution of the program any of the other modes may be selected using a single output instruction.

The modes for PORT A & PORT B can be separately defined, while PORT C is divided into two portions as required by the PORT A and PORT B definitions. The ports are thus divided into two groups Group A & Group B. All the output register, including the status flip-flop will be reset whenever the mode is changed. Modes of the two group may be combined for any desired I/O operation e.g. Group A in mode ‘1’ and group B in mode ‘0’.

The basic mode definitions with bus interface and the mode definition format are given in fig (a) & (b),


AB
BCB
BDB
RD, WR
1
A1, A0, CS
D7 - D0
8255
PORT B
Mode '0 '
8
4
8
PB7 - PB1
PC7 - PC4
PC3 - PC0
PA7 - PA0

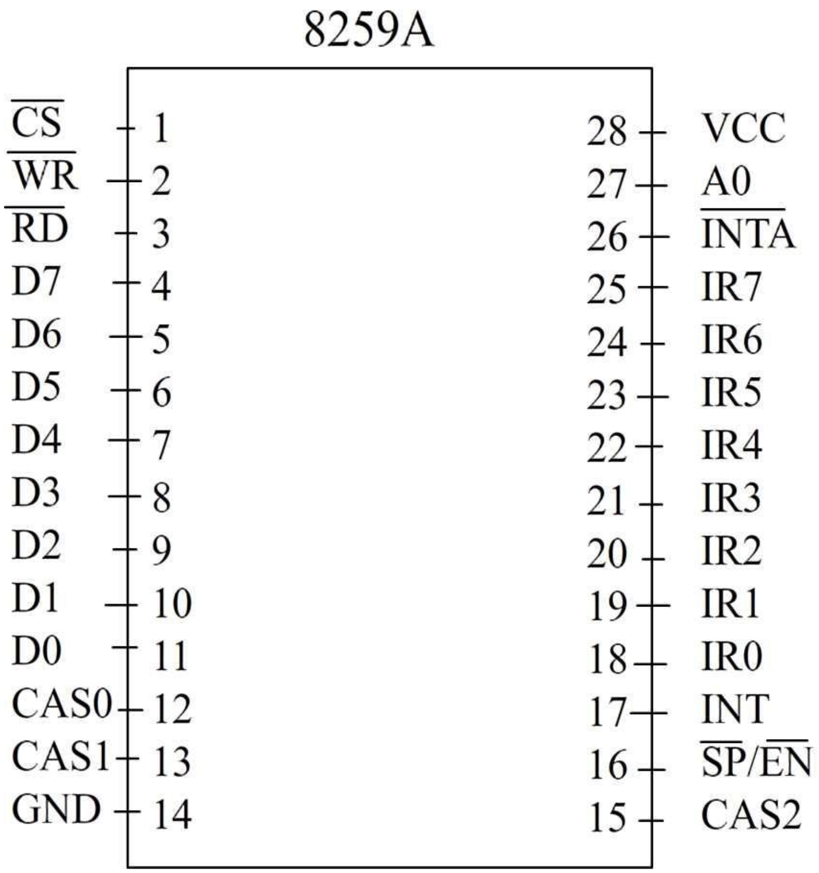

8255
Mode 1
PCO
1
2
3
4
5
6
7
PB7 - PB0
INTRB
IBFB or OBFB
STBB or ACKB
INTRA
STBA or I/O
IBFA or I/O
I/O or ACKA
I/O or OBFA
PA7 - PA0
Mode 2
PCO
1
2
3
4
5
6
7
INTRA
PB7 - PB0
I/O or control
STBA
IBFA
ACKA
OBFA
PORTA control
PA7 - PA0
Bi-directional Bus

INTEL 8259A Programmable Interrupt Controller

The 8259A is a programmable interrupt controller designed to work with Intel microprocessor 8080 A, 8085, 8086, 8088. The 8259 A interrupt controller can

1) Handle eight interrupt inputs. This is equivalent to providing eight interrupt pins on the processor in place of one INTR/INT pin.
2) Vector an interrupt request anywhere in the memory map. However, all the eight interrupt are spaced at the interval of either four or eight location. This eliminates the major drawback, 8085 interrupt, in which all interrupts are vectored to memory location on page 00_H.
3) Resolve eight levels of interrupt priorities in a variety of modes.
4) Mask each interrupt request individually.
5) Read the status of pending interrupts, in service interrupts, and masked interrupts.
6) Be set up to accept either the level triggered or edge triggered interrupt request.
7) Mine 8259 as can be cascade in a master slave configuration to handle 64 interrupt inputs.

The 8259 A is contained in a 28-element in line package that requires only a compatible with 8259. The main difference between the two is that the 8259 A can be used with Intel 8086/8088 processor. It also induces additional features such as level triggered mode, buffered mode and automatic end of interrupt mode. The pin diagram and interval block diagram is shown below:



The pins are defined as follows:

$\overline{CS}$: Chip select

To access this chip, $\overline{CS}$ is made low. A LOW on this pin enables $\overline{RD}$ & $\overline{WR}$ communication between the CPU and the 8259A. This pin is connected to address bus through the decoder logic circuits. INTA functions are independent of $\overline{CS}$

$\overline{WR}$:

A low on this pin. When $\overline{CS}$ is low enables the 8259 A to accept command words from CPU.

$\overline{RD}$:

A low on this pin when $\overline{CS}$ is low enables these 8259 A to release status on to the data bus for the CPU. The status in dudes the contents of IMR, ISR or TRR register or a priority level.

D_7-D_0:

Bidirectional data bus control status and interrupt in a this bus. This bus is connected to BDB of 8085.

CAS_0-CAS_2:

Cascade lines: The CAS lines form a private 8259A bus to control a multiple 8259A structure ie to identify a particular slave device. These pins are outputs of a master 8259A and inputs for a slave 8259A.

$\overline{SP}$/ $\overline{EN}$: Salve program/enable buffer:

This is a dual function pin. It is used as an input to determine whether the 8259A is to a master ($\overline{SP}$/ $\overline{EN}$ = 1) or as a slave ($\overline{SP}$/ $\overline{EN}$ = 0). It is also used as an output to disable the data bus transceivers when data are being transferred from the 8259A to the CPU. When in buffered mode, it can be used as an output and when not in the buffered mode it is used as an input.

INT:

This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the CPU, thus it is connected to the CPU’s interrupt pin (INTR).

$\overline{INTA}$:

Interrupt: Acknowledge. This pin is used to enable 8259A interrupt vector data on the data bus by a sequence of interrupt request pulses issued by the CPU.

IR_0-IR_7:

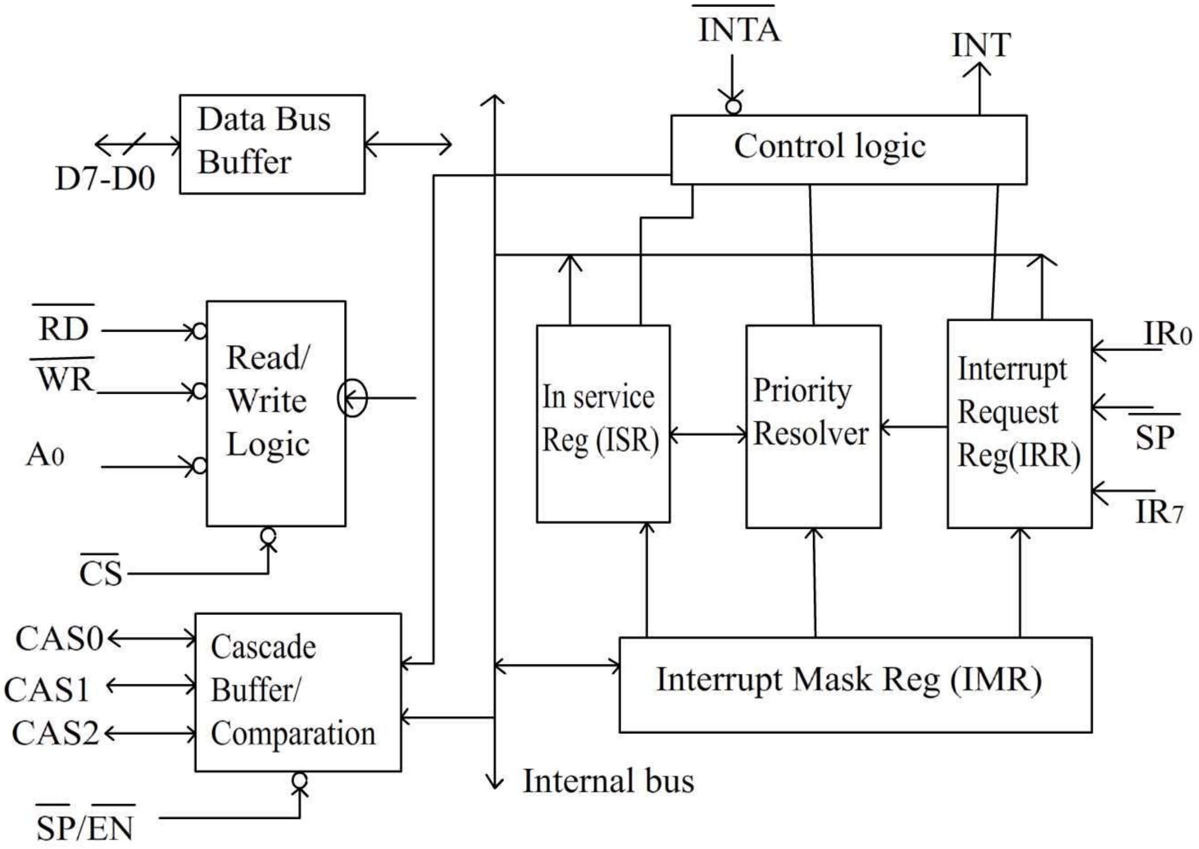
Interrupt Requests: Asynchronous interrupt inputs. An interrupt request is executed by raising an IR input (low to high), and holding it high until it is acknowledged. (Edge triggered mode).or just by a high level on an IR input (levels triggered mode).

A_0:

A_0 address line: This pin acts in conjunction with the $\overline{RD}$, $\overline{WR}$ & $\overline{CS}$ pins. It is used by the 8259A to send various command words from the CPU and to read the status. If is connected to the CPU A_0 address line. Two addresses must be reserved in the I/O address space for each 8259 in the system.

Functional Description:

The 8259 A has eight interrupt request inputs, TR2 IR0. The 8259 A uses its INT output to interrupt the 8085A via INTR pin. The 8259A receives interrupt acknowledge pulses from the μp at its $\overline{INTA}$ input. Vector address used by the 8085 A to transfer control to the service subroutine of the interrupting device, is provided by the 8259 A on the data bus. The 8259A is a programmable device that must be initialized by command words sent by the. After initialization the 8259 A mode of operation can be changed by operation command words from the.



The descriptions of various blocks are,

Data bus buffer:

This 3- state, bidirectional 8-bit buffer is used to interface the 8259A to the system data bus. Control words and status information are transferred through the data bus buffer.

Read/Write & control logic:

The function of this block is to accept OUTPUT commands from the CPU. It contains the initialization command word (ICW) register and operation command word (OCW) register which store the various control formats for device operation. This function block also allows the status of 8159A to be transferred to the data bus.

Interrupt request register (IRR):

IRR stores all the interrupt inputs that are requesting service. Basically, it keeps track of which interrupt inputs are asking for service. If an interrupt input is unmasked, and has an interrupt signal on it, then the corresponding bit in the IRR will be set.

Interrupt mask register (IMR):

The IMR is used to disable (Mask) or enable (Unmask) individual interrupt inputs. Each bit in this register corresponds to the interrupt input with the same number. The IMR operation on the IRR. Masking of higher priority input will not affect the interrupt request lines of lower priority. To unmask any interrupt the corresponding bit is set ‘0’.

In service register (ISR):

The in service registers keeps tracks of which interrupt inputs are currently being serviced. For each input that is currently being serviced the corresponding bit will be set in the in service register. Each of these 3-reg can be read as status reg.

Priority Resolver:

This logic block determines the priorities of the set in the IRR. The highest priority is selected and strobed into the corresponding bit of the ISR during $\overline{INTA}$ pulse.

Cascade buffer/comparator:

This function blocks stores and compare the IDS of all 8259A’s in the reg. The associated 3-I/O pins (CAS0-CAS2) are outputs when

8259A is used a master. Master and are inputs when 8259A is used as a slave. As a master, the 8259A sends the ID of the interrupting slave device onto the cas2-cas0. The slave thus selected will send its pre-programmed subroutine address on to the data bus during the next one or two successive $\overline{INTA}$ pulses.

8257: Direct Memory Access Controller

The *Direct Memory Access* or DMA mode of data transfer is the fastest amongst all the modes of data transfer. In this mode, the device may transfer data directly to/from memory without any interference from the CPU. The device requests the CPU (through a DMA controller) to hold its data, address and control bus, so that the device may transfer data directly to/from memory.

The DMA data transfer is initiated only after receiving HLDA signal from the CPU. Intel's 8257 is a four channel DMA controller designed to be interfaced with their family of microprocessors. The 8257, on behalf of the devices, requests the CPU for bus access using local bus request input i.e. HOLD in minimum mode. In maximum mode of the microprocessor RQ/GT pin is used as bus request input.

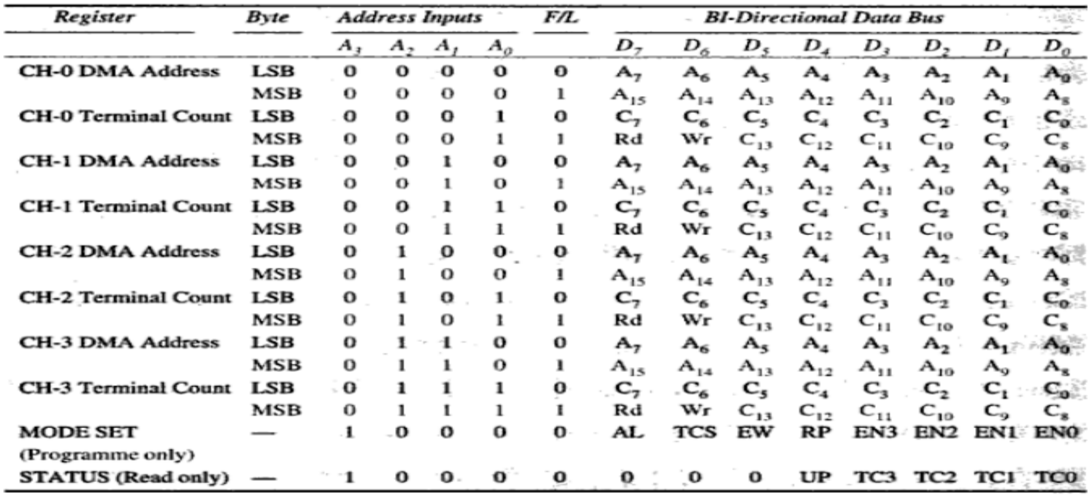
On receiving the HLDA signal (in minimum mode) or RQ/GT signal (in maximum mode) from the CPU, the requesting devices gets the access of the bus, and it completes the required number of DMA cycles for the data transfer and then hands over the control of the bus back to the CPU.

Internal Architecture of 8257

The internal architecture of 8257 is shown in figure. The chip support four DMA channels, i.e. four peripheral devices can independently request for DMA data transfer through these channels at a time. The DMA controller has 8-bit internal data buffer, a read/write unit, a control unit, a priority resolving unit along with a set of registers.

Register Organization of 8257

Table *8257 Register Selection*



Register	*Byte*	*Address Inputs*				*F/L*	*BI-Directional Data Bus*							
		A_3	A_2	A_1	A_0		D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
CH-0 DMA Address	LSB	0	0	0	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	0	0	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-0 Terminal Count	LSB	0	0	0	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	0	0	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
CH-1 DMA Address	LSB	0	0	1	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	0	1	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-1 Terminal Count	LSB	0	0	1	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	0	1	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
CH-2 DMA Address	LSB	0	1	0	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	1	0	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-2 Terminal Count	LSB	0	1	0	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	1	0	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
CH-3 DMA Address	LSB	0	1	1	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	1	1	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-3 Terminal Count	LSB	0	1	1	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	1	1	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
MODE SET (Programme only)	—	1	0	0	0	0	AL	TCS	EW	RP	EN3	EN2	EN1	EN0
STATUS (Read only)	—	1	0	0	0	0	0	0	0	UP	TC3	TC2	TC1	TC0

The 8257 performs the DMA operation over four independent DMA channels. Each of four channels of 8257 has a pair of two 16-bit registers, viz. DMA *address* *register* and *terminal count register*.

There are two common registers for all the channels, namely, *mode set register* and *status register*. Thus there are a total of ten registers. The CPU selects one of these ten registers using address lines Ao-A3. Table shows how the Ao-A3 bits may be used for selecting one of these registers.

DMA Address Register

Each DMA channel has one DMA address register. The function of this register is to store the address of the starting memory location, which will be accessed by the DMA channel. Thus the starting address of the memory block which will be accessed by the device is first loaded in the DMA address register of the channel.

The device that wants to transfer data over a DMA channel, will access the block of the memory with the starting address stored in the DMA Address Register.

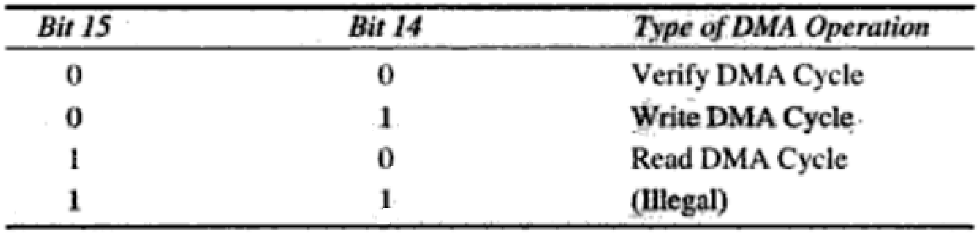
Terminal Count Register

Each of the four DMA channels of 8257 has one terminal count register (TC). This 16-bit register isused for ascertaining that the data transfer through a DMA channel ceases or stops after the required number of DMA cycles. The low order 14-bits of the terminal count register are initialised with the binary equivalent of the number of required DMA cycles minus one.

After each DMA cycle, the terminal count register content will be decremented by one and finally it becomes zero after the required number of DMA cycles are over. The bits 14 and 15 of this register indicate the type of the DMA operation (transfer). If the device wants to write data into the memory, the DMA operation is called DMA write operation. Bit 14 of the register in this case will be set to one and bit 15 will be set to zero.

Table gives detail of DMA operation selection and corresponding bit configuration of bits 14 and 15 of the TC register.

Table *DMA Operation Selection Using A_{15}/RD and A_{14}/WR*



Bit 15	*Bit 14*	*Type of DMA Operation*
0	0	Verify DMA Cycle
0	1	Write DMA Cycle
1	0	Read DMA Cycle
1	1	(Illegal)

Mode Set Register

The mode set register is used for programming the 8257 as per the requirements of the system. The function of the mode set register is to enable the DMA channels individually and also to set the various modes of operation.

The DMA channel should not be enabled till the DMA address register and the terminal count register contain valid information, otherwise, an unwanted DMA request may initiate a DMA cycle, probably destroying the valid memory data. The bits Do-D3 enable one of the four DMA channels of 8257. for example, if Do is ‘1’, channel 0 is enabled. If

bit 4 is set, rotating priority is enabled, otherwise, the normal, i.e. fixed priority is enabled.

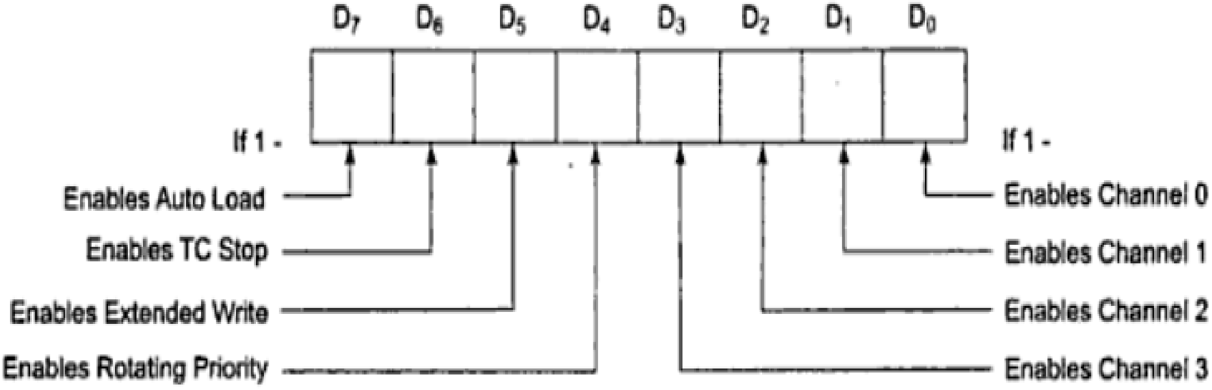


Fig. *Bit Definitions of the Mode Set Register*

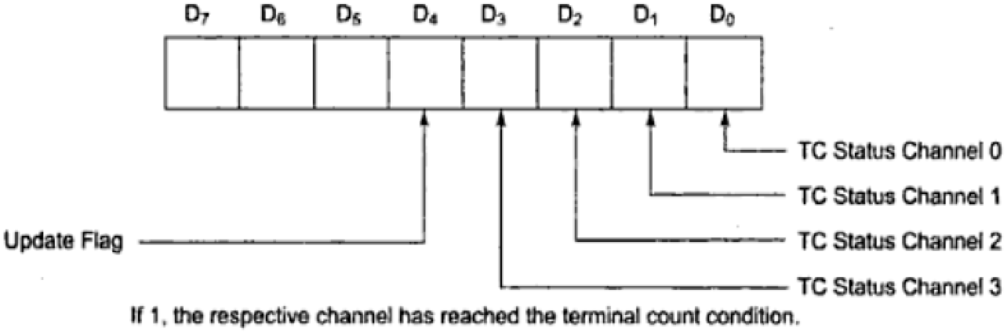
If the TC STOP bit is set, the selected channel is disabled after the *terminal count* condition is reached, and it further prevents any DMA cycle on the channel. To enable the channel again, this bit must be reprogrammed. If the TC STOP bit is programmed to be zero, the channel is not disabled, even after the count reaches zero and further request are allowed on the same channel.

The auto load bit, if set, enables channel 2 for the repeat block chaining operations, without immediate software intervention between the two successive blocks. The channel 2 registers are used as usual, while the channel 3 registers are used to store the block reinitialisation parameters, i.e. the DMA starting address and terminal count. After the first block is transferred using DMA, the channel 2 registers are reloaded with the corresponding channel 3 registers for the next block transfer, if the *update* flag is set. The extended write bit, if set to '1', extends the duration of MEMW and IOW signals by activating them earlier, this is useful in interfacing the peripherals with different access times.

If the peripheral is not accessed within the stipulated time, it is expected to give the 'NOT READY' indication to 8257, to request it to add one or more wait states in the DMA CYCLE. The mode set register can only be written into.

Status Register

The status register of 8257 is shown in figure. The lower order 4-bits of this register contain the terminal count status for the four individual channels. If any of these bits is set, it indicates that the specific channel has reached the terminal count condition.



If 1, the respective channel has reached the terminal count condition.

These bits remain set till either the status is read by the CPU or the 8257 is reset. The update flag is not affected by the read operation. This flag can only be cleared by resetting 8257 or by resetting the auto load bit of the mode set register. If the update flag is set, the contents of the channel 3 registers are reloaded to the corresponding registers of channel 2 whenever the channel 2 reaches a terminal count condition, after transferring one block and the next block is to be transferred using the autoload feature of 8257.

The update flag is set every time, the channel 2 registers are loaded with contents of the channel 3 registers. It is cleared by the completion of the first DMA cycle of the new block. This register can only read.

Data Bus Buffer, Read/Write Logic, Control Unit and Priority Resolver

The 8-bit. Tristate, bidirectional buffer interfaces the internal bus of 8257 with the external system bus under the control of various control signals.

In the slave mode, the read/write logic accepts the I/O Read or I/O Write signals, decodes the Ao-A3 lines and either writes the contents of the data bus to the addressed internal register or reads the contents of the selected register depending upon whether IOW or IOR signal is activated.

In master mode, the read/write logic generates the IOR and IOW signals to control the data flow to or from the selected peripheral. The control logic controls the sequences of operations and generates the required control signals like AEN, ADSTB, MEMR, MEMW, TC and MARK along with the address lines A4-A7, in master mode. The priority resolver resolves the priority of the four DMA channels depending upon whether normal priority or rotating priority is programmed.

Signal Description of 8257

DRQo-DRQ3 :

These are the four individual channel DMA request inputs, used by the peripheral devices for requesting the DMA services. The DRQo has the highest priority while DRQ**3** has the lowest one, if the fixed priority mode is selected.

DACKo-DACK3 :

These are the active-low DMA acknowledge output lines which inform the requesting peripheral that the request has been honoured and the bus is relinquished by the CPU. These lines may act as strobe lines for the requesting devices.

Do-D7:

These are bidirectional, data lines used to interface the system bus with the internal data bus of 8257. These lines carry command words to 8257 and status word from 8257, in slave mode, i.e. under the control of CPU.

The data over these lines may be transferred in both the directions. When the 8257 is the bus master (master mode, i.e. not under CPU control), it uses Do-D7 lines to send higher byte of the generated address to the latch. This address is further latched using ADSTB signal. the address is transferred over Do-D7 during the first clock cycle of the DMA cycle. During the rest of the period, data is available on the data bus.

IOR:

This is an active-low bidirectional tristate input line that acts as an input in the slave mode. In slave mode, this input signal is used by the CPU to read internal registers of 8257.this line acts output in master mode. In master mode, this signal is used to read data from a peripheral during a memory write cycle.

IOW :

This is an active low bidirection tristate line that acts as input in slave mode to load the contents of the data bus to the 8-bit mode register or upper/lower byte of a 16-bit DMA address register or terminal count register. In the master mode, it is a control output that loads the data to a peripheral during DMA memory read cycle (write to peripheral).

CLK:

This is a clock frequency input required to derive basic system timings for the internal operation of 8257.

RESET :

This active-high asynchronous input disables all the DMA channels by clearing the mode register and tristates all the control lines.

Ao-A3:

These are the four least significant address lines. In slave mode, they act as input which select one of the registers to be read or written. In the master mode, they are the four least significant memory address output lines generated by 8257.

CS:

This is an active-low chip select line that enables the read/write operations from/to 8257, in slave mode. In the master mode, it is automatically disabled to prevent the chip from getting selected (by CPU) while performing the DMA operation.

A4-A7 :

This is the higher nibble of the lower byte address generated by 8257 during the master mode of DMA operation.

READY:

This is an active-high asynchronous input used to stretch memory read and write cycles of 8257 by inserting wait states. This is used while interfacing slower peripherals..

HRQ:

The hold request output requests the access of the system bus. In the non-cascaded 8257 systems, this is connected with HOLD pin of CPU. In the cascade mode, this pin of a slave is connected with a DRQ input line of the master 8257, while that of the master is connected with HOLD input of the CPU.

HLDA :

The CPU drives this input to the DMA controller high, while granting the bus to the device. This pin is connected to the HLDA output of the CPU. This input, if high, indicates to the DMA controller that the bus has been granted to the requesting peripheral by the CPU.

MEMR:

This active –low memory read output is used to read data from the addressed memory locations during DMA read cycles.

MEMW :

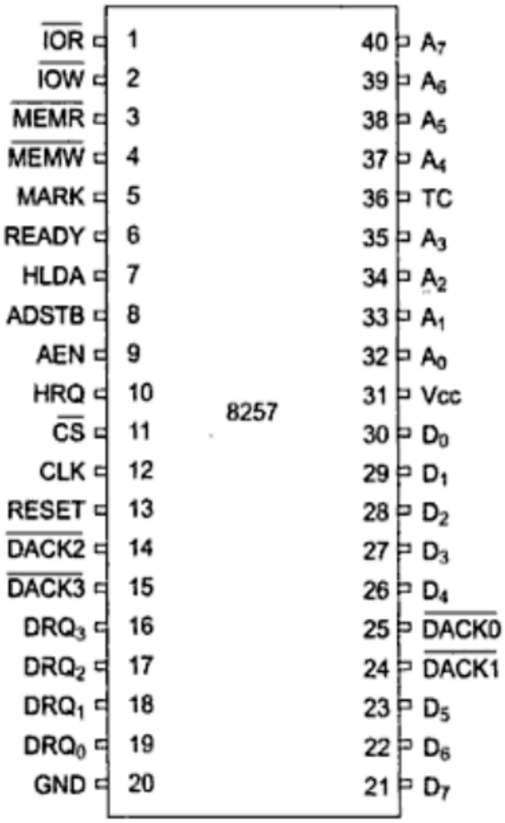
This active-low three state output is used to write data to the addressed memory location during DMA write operation.

ADST :

This output from 8257 strobes the higher byte of the memory address generated by the DMA controller into the latches.

AEN:

This output is used to disable the system data bus and the control the bus driven by the CPU, this may be used to disable the system address and data bus by using the enable input of the bus drivers to inhibit the non-DMA devices from responding during DMA operations. If the 8257 is I/O mapped, this should be used to disable the other I/O devices, when the DMA controller addresses is on the address bus.



Pin diagram of 8257

TC:

Terminal count output indicates to the currently selected peripherals that the present DMA cycle is the last for the previously programmed data block. If the TC STOP bit in the mode set register is set, the selected channel will be disabled at the end of the DMA cycle.

The TC pin is activated when the 14-bit content of the terminal count register of the selected channel becomes equal to zero. The lower order 14 bits of the terminal count register are to be programmed with a 14-bit equivalent of (n-1), if n is the desired number of DMA cycles.

MARK :

The modulo 128 mark output indicates to the selected peripheral that the current DMA cycle is the 128th cycle since the previous MARK output. The mark will be activated after each 128 cycles or integral multiples of it from the beginning if the data block (the first DMA cycle), if the total number of the required DMA cycles (n) is completely divisible by 128.

Vcc :

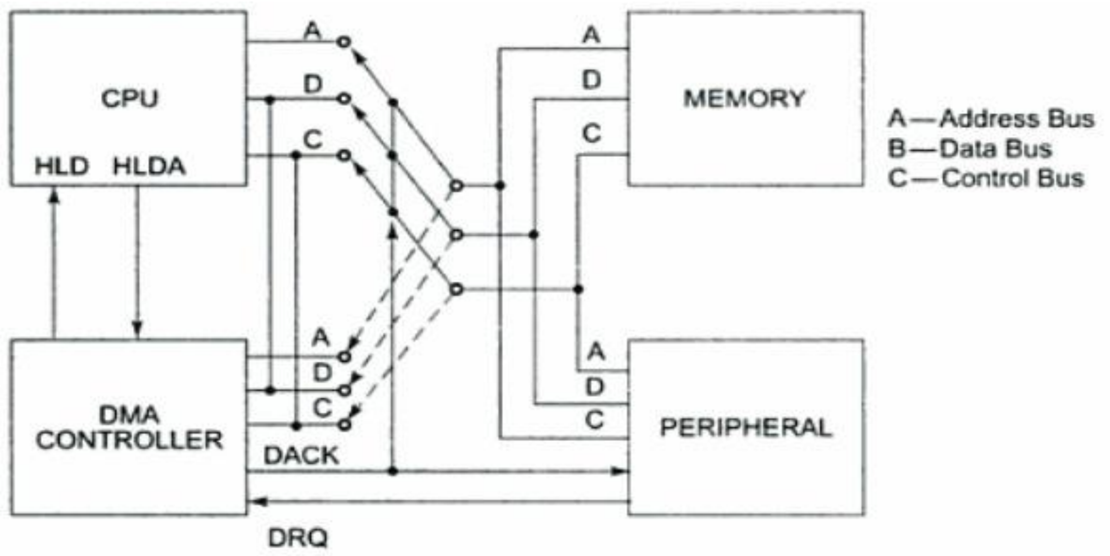
This is a +5v supply pin required for operation of the circuit.

GND :

This is a return line for the supply (ground pin of the IC).

Interfacing 8257 with 8086

Once a DMA controller is initialised by a CPU property, it is ready to take control of the system bus on a DMA request, either from a peripheral or itself (in case of memory-to-memory transfer). The DMA controller sends a HOLD request to the CPU and waits for the CPU to assert the HLDA signal. The CPU relinquishes the control of the bus before asserting the HLDA signal.



A conceptual implementation of the system is shown in Figure

Once the HLDA signal goes high, the DMA controller activates the DACK signal to the requesting peripheral and gains the control of the system bus. The DMA controller is the sole master of the bus, till the DMA operation is over. The CPU remains in the HOLD status (all of its signals are tristate except HOLD and HLDA), till the DMA controller is the master of the bus.

In other words, the DMA controller interfacing circuit implements a switching arrangement for the address, data and control busses of the memory and peripheral subsystem from/to the CPU to/from the DMA controller.

GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

Digital Electronics and Circuit

For 3rd Semester

ELECTRONICS AND TELECOMMUNICATION

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Mr. Sarada Prasanna Singh

(Lecturer in Electronics & Telecommunication Engineering)

NUMBER SYSTEM AND CODES

INTRODUCTION:-

- The term digital refers to a process that is achieved by using discrete unit.
- In number system there are different symbols and each symbol has an absolute value and also has place value.

RADIX OR BASE:-

The radix or base of a number system is defined as the number of different digits which can occur in each position in the number system.

RADIX POINT :-

The generalized form of a decimal point is known as radix point. In any positional number system the radix point divides the integer and fractional part.

N_r = [Integer part ▪ Fractional part]

↑

Radix point

NUMBER SYSTEM:-

In general a number in a system having base or radix ‘ r ’ can be written as

$$a_n \quad a_{n-1} \; a_{n-2} \; \ldots\ldots\ldots\ldots\ldots \; a_0 \; . \; a_{-1} \; a_{-2} \ldots\ldots\ldots\ldots a_{-m}$$

This will be interpreted as

$$Y = a_n \times r^n + a_{n-1} \times r^{n-1} + a_{n-2} \times r^{n-2} + \ldots\ldots\ldots + a_0 \times r^0 + a_{-1} \times r^{-1} + a_{-2} \times r^{-2} + \ldots\ldots\ldots + a_{-m} \times r^{-m}$$

where Y = value of the entire number

a_n = the value of the nth digit

r = radix

TYPES OF NUMBER SYSTEM:-

There are four types of number systems. They are

1. Decimal number system
2. Binary number system
3. Octal number system
4. Hexadecimal number system

DECIMAL NUMBER SYSTEM:-

- The decimal number system contain ten unique symbols 0,1,2,3,4,5,6,7,8 and 9.
- In decimal system 10 symbols are involved, so the base or radix is 10.
- It is a positional weighted system.
- The value attached to the symbol depends on its location with respect to the decimal point.

In general,

$$d_n \quad d_{n-1} \quad d_{n-2} \;\ldots\ldots\ldots\ldots\ldots\; d_0 \;.\; d_{-1} \quad d_{-2} \ldots\ldots\ldots\ldots d_{-m}$$

is given by

$$(d_n \times 10^n) + (d_{n-1} \times 10^{n-1}) + (d_{n-2} \times 10^{n-2}) + \ldots + (d_0 \times 10^0) + (d_{-1} \times 10^{-1}) + (d_{-2} \times 10^{-2}) + \ldots + (d_{-m} \times 10^{-m})$$

For example:-

9256.26 = 9 x 1000 + 2 x 100 + 5 x 10 + 6 x 1 + 2 x (1/10) + 6 x (1/100)

$= 9 \times 10^3 + 2 \times 10^2 + 5 \times 10^1 + 6 \times 10^0 + 2 \times 10^{-1} + 6 \times 10^{-2}$

BINARY NUMBER SYSTEM:-

- The binary number system is a positional weighted system.
- The base or radix of this number system is 2.
- It has two independent symbols.
- The symbols used are 0 and 1.
- A binary digit is called a bit.
- The binary point separates the integer and fraction parts.

In general,

$$d_n \quad d_{n-1} \quad d_{n-2} \;\ldots\ldots\ldots\ldots\ldots\; d_0 \;.\; d_{-1} \quad d_{-2} \ldots\ldots\ldots\ldots d_{-k}$$

is given by

$$(d_n \times 2^n) + (d_{n-1} \times 2^{n-1}) + (d_{n-2} \times 2^{n-2}) + \ldots.+ (d_0 \times 2^0) + (d_{-1} \times 2^{-1}) + (d_{-2} \times 2^{-2}) + \ldots.+(d_{-k} \times 2^{-k})$$

OCTAL NUMBER SYSTEM:-

- It is also a positional weighted system.
- Its base or radix is 8.
- It has 8 independent symbols 0,1,2,3,4,5,6 and 7.
- Its base 8 = 2^3 , every 3- bit group of binary can be represented by an octal digit.

HEXADECIMAL NUMBER SYSTEM:-

- The hexadecimal number system is a positional weighted system.
- The base or radix of this number system is 16.
- The symbols used are 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E and F
- The base 16 = 24 , every 4 – bit group of binary can be represented by an hexadecimal digit.

CONVERSION FROM ONE NUMBER SYSTEM TO ANOTHER :-

1. BINARY NUMBER SYSTEM:-

(a) Binary to decimal conversion:-

In this method, each binary digit of the number is multiplied by its positional weight and the product terms are added to obtain decimal number.

For example:

(i) Convert $(10101)_2$ to decimal.

Solution :

(Positional weight) $2^4\ 2^3\ 2^2\ 2^1\ 2^0$

Binary number 10101

$= (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0)$

$= 16 + 0 + 4 + 0 + 1$

$= (21)_{10}$

(ii) Convert $(111.101)_2$ to decimal.

Solution:

$(111.101)_2 = (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) + (1 \times 2^{-1}) + (0 \times 2^{-2}) + (1 \times 2^{-3})$

$= 4 + 2 + 1 + 0.5 + 0 + 0.125$

$= (7.625)_{10}$

(b) Binary to Octal conversion:-

For conversion binary to octal the binary numbers are divided into groups of 3 bits each, starting at the binary point and proceeding towards left and right.

Octal	Binary	Octal	Binary
0	000	4	100
1	001	5	101
2	010	6	110
3	011	7	111

For example:

(i) Convert $(101111010110.110110011)_2$ into octal.

Solution :

Group of 3 bits are 101 111 010 110 . 110 110 011

Convert each group into octal = 5 7 2 6 . 6 6 3

The result is **$(5726.663)_8$**

(ii) Convert $(10101111001.0111)_2$ into octal.

Solution :

Binary number 10 101 111 001 . 011 1

Group of 3 bits are = 010 101 111 001 . 011 100

Convert each group into octal = 2 5 7 1 . 3 4

The result is **$(2571.34)_8$**

(c) Binary to Hexadecimal conversion:-

For conversion binary to hexadecimal number the binary numbers starting from the binary point, groups are made of 4 bits each, on either side of the binary point.

Hexadecimal	Binary	Hexadecimal	Binary
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

For example:
(i) Convert $(1011011011)_2$ into hexadecimal.

Solution:

Given Binary number 10 1101 1011

Group of 4 bits are 0010 1101 1011

Convert each group into hex = 2 D B

The result is **$(2DB)_{16}$**

(ii) Convert $(01011111011.011111)_2$ into hexadecimal.

Solution:

Given Binary number 010 1111 1011 **.** 0111 11

Group of 3 bits are = 0010 1111 1011 **.** 0111 1100

Convert each group into octal = 2 F B **.** 7 C

The result is **$(2FB.7C)_{16}$**

2. DECIMAL NUMBER SYSTEM:-

(a)Decimal to binary conversion:-

In the conversion the integer number are converted to the desired base using successive division by the base or radix.

For example:
(i) Convert $(52)_{10}$ into binary.

Solution:

Divide the given decimal number successively by 2 read the integer part remainder upwards to get equivalent binary number. Multiply the fraction part by 2. Keep the integer in the product as it is and multiply the new fraction in the product by 2. The process is continued and the integer are read in the products from top to bottom.

2 | 52
2 | 26 — 0
2 | 13 — 0
2 | 6 — 1
2 | 3 — 0
2 | 1 — 1
0 — 1

Result of $(52)_{10}$ is **$(110100)_2$**

(ii) Convert $(105.15)_{10}$ into binary.

Solution:

Integer part		**Fraction part**
2 | 105		0.15 x 2 = 0.30
2 | 52	— 1	0.30 x 2 = 0.60
2 | 26	— 0	0.60 x 2 = 1.20
2 | 13	— 0	0.20 x 2 = 0.40
2 | 6	— 1	0.40 x 2 = 0.80
2 | 3	— 0	0.80 x 2 = 1.60
2 | 1	— 1	
0	— 1	

Result of $(105.15)_{10}$ is **$(1101001.001001)_2$**

(b)Decimal to octal conversion:-

To convert the given decimal integer number to octal, successively divide the given number by 8 till the quotient is 0. To convert the given decimal fractions to octal successively multiply the decimal fraction and the subsequent decimal fractions by 8 till the product is 0 or till the required accuracy is obtained.

For example:
(i) Convert $(378.93)_{10}$ into octal.

Solution:

8 | 378		0.93 x 8 = 7.44
8 | 47	— 2	0.44 x 8 = 3.52
8 | 5	— 7	0.52 x 8 = 4.16
0	— 5	0.16 x 8 = 1.28

Result of $(378.93)_{10}$ is **$(572.7341)_8$**

(c)Decimal to hexadecimal conversion:-

The decimal to hexadecimal conversion is same as octal.

For example:
(i) Convert $(2598.675)_{10}$ into hexadecimal.

Solution:

	Remainder			
	Decimal	**Hex**		**Hex**
16 | 2598			0.675 x 16 = 10.8	A
16 | 162	— 6	6	0.800 x 16 = 12.8	C
16 | 10	— 2	2	0.800 x 16 = 12.8	C
0	— 10	A	0.800 x 16 = 12.8	C

Result of $(2598.675)_{10}$ is **$(A26.ACCC)_{16}$**

3. OCTAL NUMBER SYSTEM:-

(a)Octal to binary conversion:-

To convert a given a octal number to binary, replace each octal digit by its 3- bit binary equivalent.

For example:

Convert $(367.52)_8$ into binary.

Solution:

Given Octal number is		3	6	7	.	5	2
Convert each group octal to binary	=	011	110	111	.	101	010

Result of $(367.52)_8$ is **$(011110111.101010)_2$**

(b)Octal to decimal conversion:-

For conversion octal to decimal number, multiply each digit in the octal number by the weight of its position and add all the product terms

For example: -

Convert $(4057.06)_8$ to decimal

Solution:

$$\begin{aligned}(4057.06)_8 &= 4 \times 8^3 + 0 \times 8^2 + 5 \times 8^1 + 7 \times 8^0 + 0 \times 8^{-1} + 6 \times 8^{-2}\\ &= 2048 + 0 + 40 + 7 + 0 + 0.0937\\ &= (2095.\ 0937)_{10}\end{aligned}$$

Result is **$(2095.0937)_{10}$**

(c) Octal to hexadecimal conversion:-

For conversion of octal to Hexadecimal, first convert the given octal number to binary and then binary number to hexadecimal.

For example :-

Convert $(756.603)_8$ to hexadecimal.

Solution :-

Given octal no.		7	5	6	.	6	0	3
Convert each octal digit to binary	=	111	101	110	.	110	000	011
Group of 4bits are	=	0001	1110	1110	.	1100	0001	1000
Convert 4 bits group to hex.	=	1	E	E	.	C	1	8

Result is (**1EE.C18)$_{16}$**

(4) HEXADECIMAL NUMBER SYSTEM :-

(a)Hexadecimal to binary conversion:-

For conversion of hexadecimal to binary, replace hexadecimal digit by its 4 bit binary group.

For example:

Convert $(3A9E.B0D)_{16}$ into binary.

Solution:

Given Hexadecimal number is		3	A	9	E	.	B	0	D
Convert each hexadecimal digit to 4 bit binary	=	0011	1010	1001	1110	.	1011	0000	1101

Result of $(3A9E.B0D)_8$ is **$(0011101010011110.101100001101)_2$**

(b)<u>Hexadecimal to decimal conversion</u>:-

For conversion of hexadecimal to decimal, multiply each digit in the hexadecimal number by its position weight and add all those product terms.

For example: -
Convert $(A0F9.0EB)_{16}$ to decimal

Solution:

$$\begin{aligned}(A0F9.0EB)_{16} &= (10 \times 16^{3})+(0 \times 16^{2})+(15 \times 16^{1}) +(9 \times 16^{0}) +(0 \times 16^{-1}) +(14 \times 16^{-2}) +(11 \times 16^{-3}) \\ &= 40960 + 0 + 240 + 9 + 0 +0.0546 + 0.0026 \\ &= (41209.0572)_{10}\end{aligned}$$

Result is **$(41209.0572)_{10}$**

(c) <u>Hexadecimal to Octal conversion</u>:-

For conversion of hexadecimal to octal, first convert the given hexadecimal number to binary and then binary number to octal.

For example :-
Convert $(B9F.AE)_{16}$ to octal.

Solution :-

Given hexadecimal no.is		B		9		F	**.**	A		E
Convert each hex. digit to binary	=	1011		1001		1111	**.**	1010		1110
Group of 3 bits are	=	101	110		011	111	**.**	101	011	100
Convert 3 bits group to octal.	=	5	6		3	7	**.**	5	3	4

Result is (**$5637.534)_{8}$**

<u>BINARY ARITHEMATIC OPERATION</u> :-

1. <u>BINARY ADDITION</u>:-

The binary addition rules are as follows
0 + 0 = 0 ; 0 + 1 = 1 ; 1 + 0 = 1 ; 1 + 1 = 10 , i.e 0 with a carry of 1

For example :-

Add $(100101)_2$ and $(1101111)_2$.
Solution :-

$$\begin{array}{rr} & 1\,0\,0\,1\,0\,1 \\ + & \underline{1\,1\,0\,1\,1\,1\,1} \\ & \underline{1\,0\,0\,1\,0\,1\,0\,0} \end{array}$$

Result is **$(10010100)_2$**

2. <u>BINARY SUBTRACTION</u>:-

The binary subtraction rules are as follows
0 - 0 = 0 ; 1 - 1 = 0 ; 1 - 0 = 1 ; 0 - 1 = 1 , with a borrow of 1

For example :-

Substract $(111.111)_2$ from $(1010.01)_2$.

Solution :-

$$\begin{array}{rr} & 1010.010 \\ - & \underline{111.111} \\ & \underline{0010.011} \end{array}$$

Result is **$(0010.011)_2$**

3. BINARY MULTIPLICATION:-

The binary multiplication rules are as follows

0 x 0 = 0 ; 1 x 1 = 1 ; 1 x 0 = 0 ; 0 x 1 = 0

For example :-

Multiply $(1101)_2$ by $(110)_2$.

Solution :-

$$\begin{array}{rr} & 1101 \\ \times & \underline{110} \\ & 0000 \\ & 1101 \\ + & \underline{1101} \\ & \underline{1001110} \end{array}$$

Result is **$(1001110)_2$**

4. BINARY DIVISION:-

The binary division is very simple and similar to decimal number system. The division by ‘0’ is meaningless. So we have only 2 rules

0 ÷ 1 = 0

1 ÷ 1 = 1

For example :-

Divide $(10110)_2$ by $(110)_2$.

Solution :-

$$\begin{array}{l} 110\ )\ 101101\ (\ 111.1 \\ \quad -\ \underline{110} \\ \qquad 1010 \\ \qquad \underline{110} \\ \qquad\ 1001 \\ \qquad\ \underline{110} \\ \qquad\ 110 \\ \qquad\ \underline{110} \\ \qquad\ \underline{000} \end{array}$$

Result is **$(111.1)_2$**

1's COMPLEMENT REPRESENTATION :-

The 1's complement of a binary number is obtained by changing each 0 to 1 and each 1 to 0.

For example :-

Find $(1100)_2$ 1's complement.

Solution :-

Given	1	1	0	0
1's complement is	0	0	1	1

Result is **$(0011)_2$**

2's COMPLEMENT REPRESENTATION :-

The 2's complement of a binary number is a binary number which is obtained by adding 1 to the 1's complement of a number i.e.

2's complement = 1's complement + 1

For example :-

Find $(1010)_2$ 2's complement.

Solution :-

Given		1	0	1	0
1's complement is		0	1	0	1
	+				1
2's complement		0	1	1	0

Result is **$(0110)_2$**

SIGNED NUMBER :-

In sign – magnitude form, additional bit called the sign bit is placed in front of the number. If the sign bit is 0, the number is positive. If it is a 1, the number is negative.

For example:-

0 1 0 1 0 0 1 = +41
↑
Sign bit

1 1 0 1 0 0 1 = -41
↑
Sign bit

SUBSTRACTION USING COMPLEMENT METHOD :-

1's COMPLEMENT:-

In 1's complement subtraction, add the 1's complement of subtrahend to the minuend. If there is a carry out, then the carry is added to the LSB. This is called end around carry. If the MSB is 0, the result is positive. If the MSB is 1, the result is negative and is in its 1's complement form. Then take its 1's complement to get the magnitude in binary.

For example:-

Subtract $(10000)_2$ from $(11010)_2$ using 1's complement.

Solution:-

```
   1 1 0 1 0                  1 1 0 1 0                       =   26
-  1 0 0 0 0      =>       +  0 1 1 1 1   (1's complement)    = - 16
                 Carry  →  1  0 1 0 0 1                         + 10
                          +           1
                              0 1 0 1 0   = +10
```

Result is **+10**

2's COMPLEMENT:-

In 2's complement subtraction, add the 2's complement of subtrahend to the minuend. If there is a carry out, ignore it. If the MSB is 0, the result is positive. If the MSB is 1, the result is negative and is in its 2's complement form. Then take its 2's complement to get the magnitude in binary.

For example:-

Subtract $(1010100)_2$ from $(1010100)_2$ using 2's complement.

Solution:-

```
   1 0 1 0 1 0 0                  1 0 1 0 1 0 0                          =     84
-  1 0 1 0 1 0 0       =>      +  0 1 0 1 1 0 0  (2's complement)        = -   84
                               1  0 0 0 0 0 0 0 ( Ignore the carry)            0
                          =                   0 (result = 0)
```

Hence MSB is 0. The answer is positive. So it is +0000000 = **0**

DIGITAL CODES:-

In practice the digital electronics requires to handle data which may be numeric, alphabets and special characters. This requires the conversion of the incoming data into binary format before it can be processed. There is various possible ways of doing this and this process is called encoding. To achieve the reverse of it, we use decoders.

WEIGHTED AND NON-WEIGHTED CODES:-

There are two types of binary codes

1) Weighted binary codes
2) Non- weighted binary codes

In weighted codes, for each position (or bit) ,there is specific weight attached.

For example, in binary number, each bit is assigned particular weight 2n where 'n' is the bit number for n = 0,1,2,3,4 the weights are 1,2,4,8,16 respectively.

Example :- BCD

Non-weighted codes are codes which are not assigned with any weight to each digit position, i.e., each digit position within the number is not assigned fixed value.

Example:- Excess – 3 (XS -3) code and Gray codes

BINARY CODED DECIMAL (BCD):-

BCD is a weighted code. In weighted codes, each successive digit from right to left represents weights equal to some specified value and to get the equivalent decimal number add the products of the weights by the corresponding binary digit. 8421 is the most common because 8421 BCD is the most natural amongst the other possible codes.

For example:-

$(567)_{10}$ is encoded in various 4 bit codes.

Solution:-

Decimal	→	5	6	7
8421 code	→	0101	0110	0111
6311 code	→	0111	1000	1001
5421 code	→	1000	0100	1010

BCD ADDITION:-

Addition of BCD (8421) is performed by adding two digits of binary, starting from least significant digit. In case if the result is an illegal code (greater than 9) or if there is a carry out of one then add 0110(6) and add the resulting carry to the next most significant.

For example:-

Add 679.6 from 536.8 using BCD addition.

Solution:-

```
   6 7 9 . 6          0110  0111  1001 . 0110     ( 679.6 in BCD)
+  5 3 6 . 8     =>+  0101  0011  0110 . 1000     (536.8 in BCD)
 1 2 1 6 . 4          1011  1010  1111 . 1110     ( All are illegal codes)
                    + 0110 +0110 +0110 .+0110     ( Add 0110 to each)
                0001  0010  0001  0110 . 0100
                  1     2     1     6  .   4      ( corrected sum = 1216.4)
```

Result is **1216.4**

BCD SUBTRACTION:-

The BCD subtraction is performed by subtracting the digits of each 4 – bit group of the subtrahend from corresponding 4 – bit group of the minuend in the binary starting from the LSD. If there is no borrow from the next higher group[then no correction is required. If there is a borrow from the next group, then 6_{10} (0110) is subtracted from the difference term of this group.

For example:-

Subtract 147.8 from 206.7 using 8421 BCD code.

Solution:-

```
   2 0 6 . 7          0010  0000  0110 . 0111     ( 206.7 in BCD)
-  1 4 7 . 8     =>-  0001  0100  0111 . 1000     (147.8 in BCD)
     5 8 . 9          0000  1011  1110 . 1111     ( Borrows are present)
                          - 0110 -0110 .- 0110
                            0101  1000 .  1001
                              5     8  .   9      ( corrected difference = 58.9)
```

Result is **$(58.9)_{10}$**

EXCESS THREE(XS-3) CODE:-

The Excess-3 code, also called XS-3, is a non- weighted BCD code. This derives it name from the fact that each binary code word is the corresponding 8421 code word plus 0011(3). It is a sequential code. It is a self complementing code.

XS-3 ADDITION:-

In XS-3 addition, add the XS-3 numbers by adding the 4 bit groups in each column starting from the LSD. If there is no carry out from the addition of any of the 4 bit groups, subtract 0011 from the sum term of those groups. If there is a carry out, add 0011 to the sum term of those groups

For example:-

Add 37 and 28 using XS-3 code.

Solution:-

```
   3 7                 0110   1010      ( 37 in XS-3)
+  2 8        =>     + 0101   1011      ( 28 in XS-3)
    6 5                1011 11010       ( Carry is generated)
                      +    1            ( Propagate carry)
                        1100   0101     ( Add 0110 to correct 0101 and
                       - 0011 +0011      subtract 0011 to correct 1100)
                        1001    1000    ( Corrected sum  in XS-3 = $65_{10}$)
```

XS-3 SUBTRACTION:-

To subtract in XS-3 number by subtracting each 4-bit group of the subtrahend from the corresponding 4-bit group of the minuend starting from the LSD. If there is no borrow from the next 4-bit group. add 0011 to the difference term of such groups. If there is a borrow, subtract 0011 from the difference term.

For example :-

. **Subtract 175 from 267 using XS-3 code**.

Solution :-`

```
   267                 0101   1010   1010    ( 267 in XS-3)
  -175        =>    -  0100   1010    1000   ( 175 in XS-3)
    092                0000   1111   0010     (Correct 0010 and 0000 by adding 0011 and
                      +0011  -0011 +0011         correct 1111 by subtracting 0011)
                       0011     1100   0101   (Corrected  difference in XS-3 = $92_{10}$ )
```

ASCII CODE:-

The American Standard Code for Information Interchange (ASCII) pronounced as ‘ASKEE’ is widely used alphanumeric code. This is basically a 7 bit code. The number of different bit patterns that can be created with 7 bits is 27 = 128 , the ASCII can be used to encode both the uppercase and lowercase characters of the alphabet (52 symbols) and some special symbols in addition to the 10 decimal digits.   It is used extensively for printers and terminals that interface with small computer systems. The table shown below shows the ASCII groups.

The ASCII code

LSBs	**MSBs**							
	000	001	010	011	100	101	110	111
0000	NUL	DEL	Space	0	@	P	P	
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	“	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s

0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	‘	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DLE

EBCDIC CODE:-

The Extended Binary Coded Decimal Interchange Code (EBCDIC) pronounced as ‘eb – si- dik’ is an 8 bit alphanumeric code. Since 28 = 256 bit patterns can be formed with 8 bits. It is used by most large computers to communicate in alphanumeric data. The table shown below shows the EBCDIC code.

The EBCDIC code

LSD (Hex)	**MSD(Hex)**															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DLE	DS		SP	&							[	]	\	0
1	SOH	DC1	SOS				/		a	j	~		A	J		1
2	STX	DC2	FS	SYN					b	k	s		B	K	S	2
3	ETX	DC3							c	l	t		C	L	T	3
4	PF	RES	BYP	PN					d	m	u		D	M	U	4
5	HT	NL	LF	RS					e	n	v		E	N	V	5
6	LC	BS	EOB	YC					f	o	w		F	O	W	6
7	DEL	IL	PRE	EOT					g	p	x		G	P	X	7
8		CAN							h	q	y		H	Q	Y	8
9		EM							i	r	z		I	R	Z	9
A	SMM	CC	SM		Ø	!	|	:								
B	VT				.	$	,	#								
C	FF	IFS		DC4	<	*	%	@								
D	CR	IGS	ENQ	NAK	(	)	_	‘								
E	SO	IRS	ACK		+	;	>	=								
F	SI	IUS	BEL	SUB	|	‘	?	‘								

GRAY CODE:-

The gray code is a non-weighted code. It is not a BCD code. It is cyclic code because successive words in this differ in one bit position only i.e it is a unit distance code.

Gray code is used in instrumentation and data acquisition systems where linear or angular displacement is measured. They are also used in shaft encoders, I/O devices, A/D converters and other peripheral equipment.

BINARY- TO – GRAY CONVERSION:-

If an n-bit binary number is represented by $B_n\ B_{n-1}$ - - - - - B_1 and its gray code equivalent by $G_n\ G_{n-1}$ - - - - - G_1, where B_n and G_n are the MSBs , then gray code bits are obtained from the binary code as follows

$$G_n = B_n$$

$$G_{n-1} = B_n \oplus B_{n-1}$$

.

.

.

.

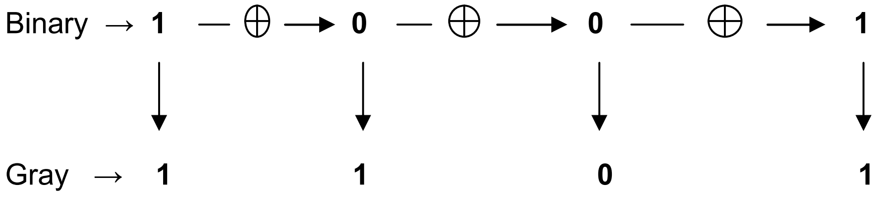
$$G_1 = B_2 \oplus B_1$$

Where the symbol $\oplus$ stands for Exclusive OR (X-OR)

For example :-

Convert the binary 1001 to the Gray code.

Solution :-`



The gray code is **1101**

GRAY- TO - BINARY CONVERSION:-

If an n-bit gray number is represented by $G_n\ G_{n-1}$ ------- G_1 and its binary equivalent by $B_n\ B_{n-1}$ - - - - - B_1, then binary bits are obtained from Gray bits as follows :

$$B_n = G_n$$

$$B_{n-1} = B_n \oplus G_{n-1}$$

.

.

.

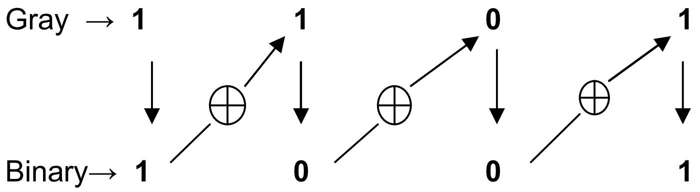
.

$$B_1 = B_2 \oplus G_1$$

For example :-

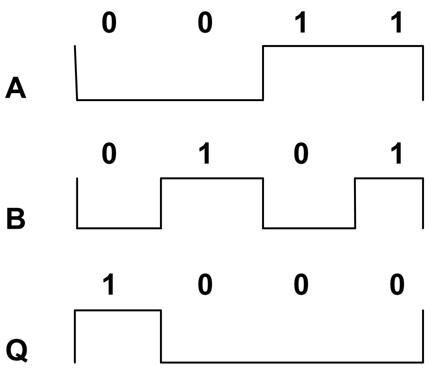
Convert the Gray code 1101 to the binary.

Solution :-



The binary code is **1001**

Timing Diagram

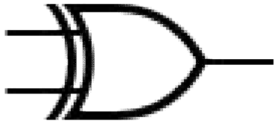


EXCLUSIVE – OR (X-OR) GATE:-

- An X-OR gate is a two input, one output logic circuit.
- The output is logic 1 when one and only one of its two inputs is logic 1. When both the inputs is logic 0 or when both the inputs is logic 1, the output is logic 0.

IC No.:- 7486

Logic Symbol



INPUTS are **A** and **B**

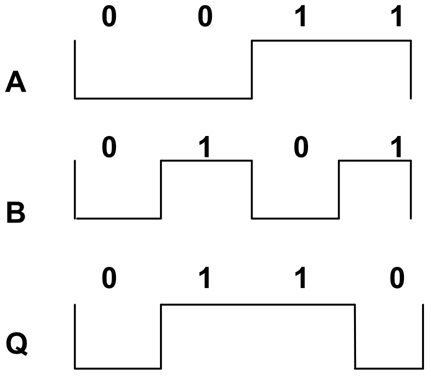
OUTPUT is **Q** = $A \oplus B$

= $A\,\overline{B} + A\,\overline{B}$

Truth Table

INPUT		OUTPUT
A	B	Q = A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

Timing Diagram

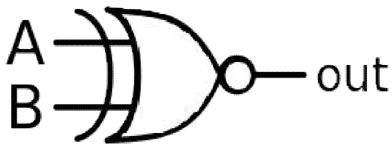


EXCLUSIVE – NOR (X-NOR) GATE:-

- An X-NOR gate is the combination of an X-OR gate and a NOT gate.
- An X-NOR gate is a two input, one output logic circuit.
- The output is logic 1 only when both the inputs are logic 0 or when both the inputs is 1.
- The output is logic 0 when one of the inputs is logic 0 and other is 1.

IC No.:- 74266

Logic Symbol

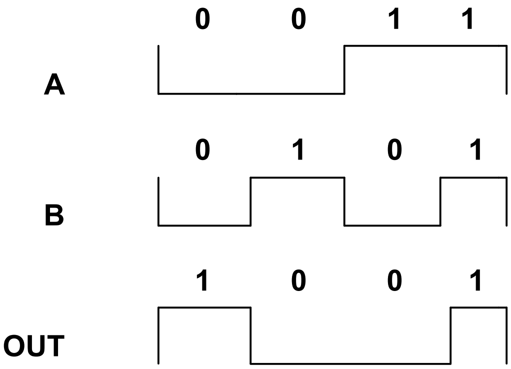


INPUT		OUTPUT
A	B	OUT =A XNOR B
0	0	1
0	1	0
1	0	0
1	1	1

OUT =A B + $\overline{A}$ $\overline{B}$

= A XNOR B

Timing Diagram

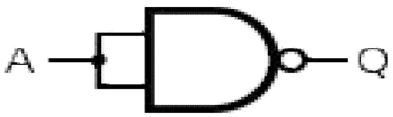


UNIVERSAL GATES:-

There are 3 basic gates AND, OR and NOT, there are two universal gates NAND and NOR, each of which can realize logic circuits single handedly. The NAND and NOR gates are called universal building blocks. Both NAND and NOR gates can perform all logic functions i.e. AND, OR, NOT, EXOR and EXNOR.

NAND GATE:-

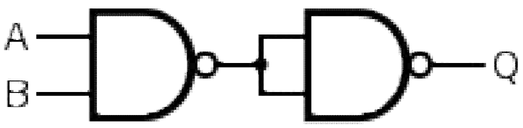
a) **Inverter from NAND gate**



Input **= A**
Output **Q = $\overline{A}$**

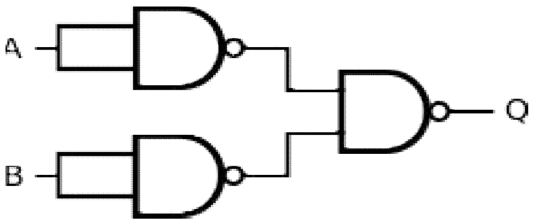
b) **AND gate from NAND gate**

Input s are **A** and **B**
Output **Q = A.B**



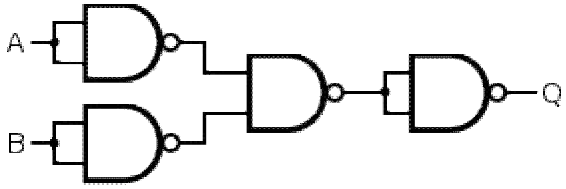
c) **OR gate from NAND gate**

Inputs are **A** and **B**
Output **Q = A+B**



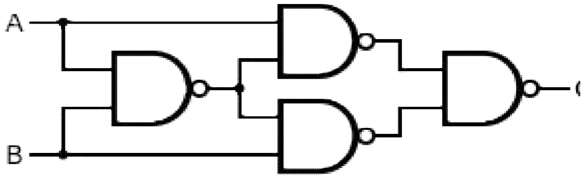
d) **NOR gate from NAND gate**

Inputs are **A** and **B**
Output **Q =** $\overline{\mathbf{A+B}}$



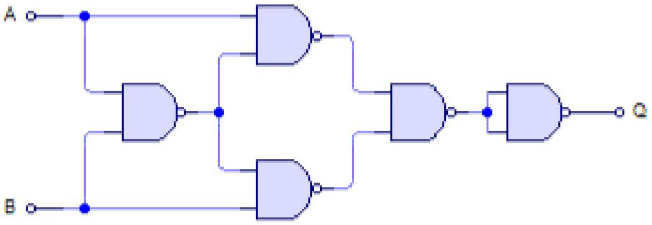
e) **EX-OR gate from NAND gate**

Inputs are **A** and **B**
Output **Q =** $\mathbf{A\,\overline{B} + \overline{A}B}$



f) **EX-NOR gate From NAND gate**

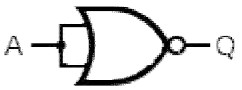
Inputs are **A** and **B**
Output **Q =** $\mathbf{A\,B + \overline{A}\,\overline{B}}$



NOR GATE:-

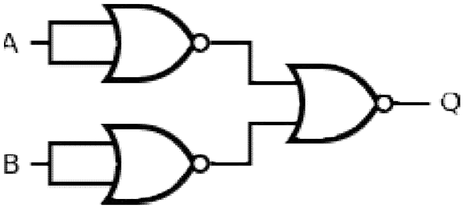
a) **Inverter from NOR gate**

Input = **A**
Output **Q =** $\overline{\mathbf{A}}$



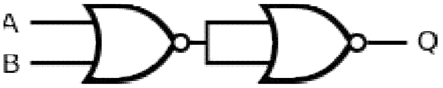
b) **AND gate from NOR gate**

Input s are **A** and **B**
Output **Q = A.B**



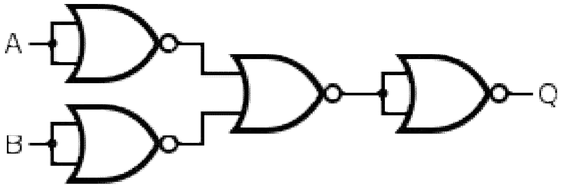
c) OR gate from NOR gate

Inputs are **A** and **B**
Output **$Q = A+B$**



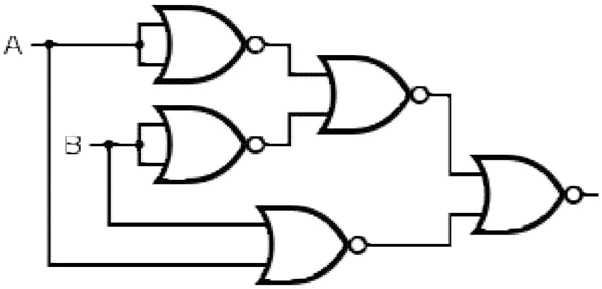
d) NAND gate from NOR gate

Inputs are **A** and **B**
Output **$Q = \overline{A.B}$**



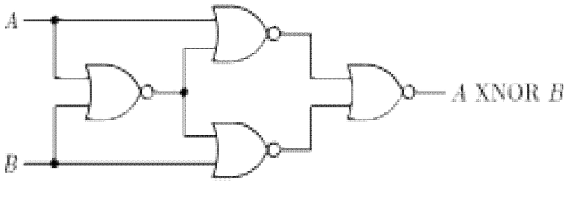
e) EX-OR gate from NOR gate

Inputs are **A** and **B**
Output **$Q = A\,\overline{B} + \overline{A}B$**



f) EX-NOR gate From NOR gate

Inputs are **A** and **B**
Output **$Q = A\,B + \overline{A}\,\overline{B}$**



THRESHOLD LOGIC:-

INTRODUCTION:-

- The threshold element, also called the threshold gate (T-gate) is a much more powerful device than any of the conventional logic gates such as NAND, NOR and others.
- Complex, large Boolean functions can be realized using much fewer threshold gates.
- Frequently a single threshold gate can realize a very complex function which otherwise might require a large number of conventional gates.
- T-gate offers incomparably economical realization; it has not found extensive use with the digital system designers mainly because of the following limitations.
  1. It is very sensitive to parameter variations.
  2. It is difficult to fabricate it in IC form.

3. The speed of switching of threshold elements in much lower than that of conventional gates.

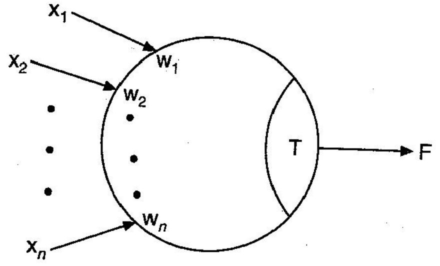
THE THRESHOLD ELEMENTS:-

- A threshold element or gate has 'n' binary inputs x_1, x_2, ..., x_n; and a single binary output F. But in addition to those, it has two more parameters.
- Its parameters are a threshold T and weights w_1, w_2, ...,w_n. The weights w_1, w_2, ..., w_n are associated with the input variables x_1, x_2, ..., x_n.
- The value of the threshold (T) and weights may be real, positive or negative number.
- The symbol of the threshold element is shown in fig.(a).
- It is represented by a circle partitioned into two parts, one part represents the weights and other represents T.
- It is defined as

$$F(x_1, x_2, \ldots\ldots, x_n) = 1 \text{ if and only if } \sum_{i=1}^{n} w_i x_i \geq T$$

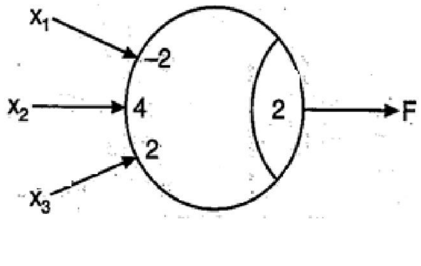
otherwise
$F(x_1, x_2, \ldots\ldots, x_n) = 0$

- The sum and product operation are normal arithmetic operations and the sum $\sum_{i=1}^{n} w_i x_i \geq T$ is called the weighted sum of the element or gate.



Example:-

Obtain the minimal Boolean expression from the threshold gate shown in figure.



Solution:-

The threshold gate with three inputs x_1, x_2, x_3 with weights -2(w_1) , 4(w_2) and 2(w_3) respectively. The value of threshold is 2(T). The table shown is the weighted sums and outputs for all input combinations. For this threshold gate, the weighted sum is

$$w = w_1x_1 + w_2x_2 + w_3x_3$$

$$= (-2)x_1 + (4)x_2 + (2)x_3$$

$$= -2x_1 + 4x_2 + 2x_3$$

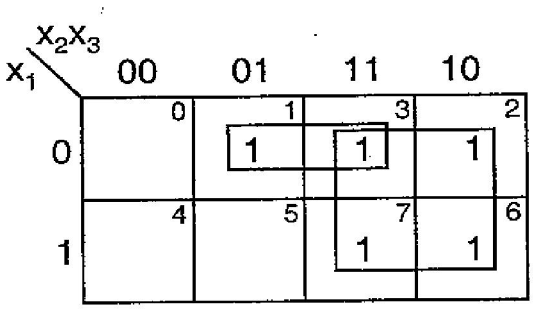
The output F is logic 1 for w≥2 and it is logic 0 for w<2

Input Variables			**Weighted Sum**	**Output**
x_1	x_2	x_3	$w = -2x_1 + 4x_2 + 2x_3$	**F**
0	0	0	0	0
0	0	1	2	1
0	1	0	4	1
0	1	1	6	1
1	0	0	-2	0
1	0	1	0	0
1	1	0	2	1
1	1	1	4	1

From the input – output relation is given in the table, the Boolean expression for the output is

$$F=\sum m\ (1, 2, 3, 6, 7)$$

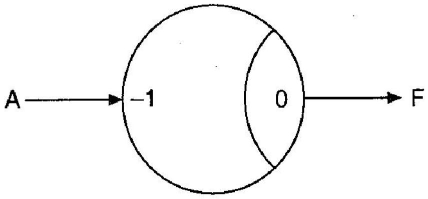
The K-map for F is



$$F_{min} = \bar{x}_1x_3 + x_2$$

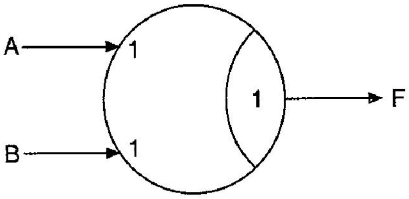
UNIVERSALITY OF A T-GATE:-

- A single T-gate can realize a large number of functions by merely changing either the weights or the threshold or both, which can be done by altering the value of the corresponding resistors.
- Since a threshold gate can realize universal gates, i.e., NAND gates and NOR gates, a threshold gate is also a universal gate.
- Single threshold gate cannot realize by a single T-gate
- Realization of logic gates using T-gates is shown in the below figure.



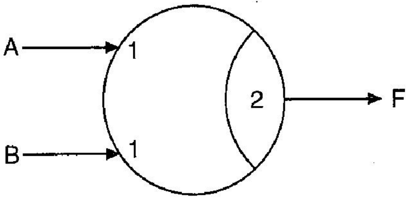
(a) NOT gate F = $\overline{A}$

Input	Weighted sum	Output
A	W = –A	F
0	0	1
1	–1	0



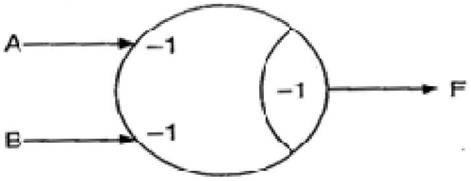
(b) OR gate F = A + B

Inputs		Weighted sum	Output
A	B	w = A + B	F
0	0	0	0
0	1	1	1
1	0	1	1
1	1	2	1



(c) AND gate F = AB

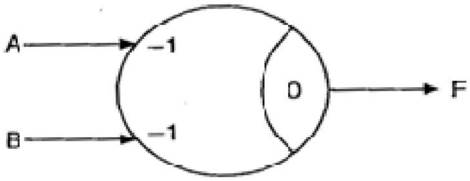
Inputs		Weighted sum	Output
A	B	w = A + B	F
0	0	0	0
0	1	1	0
1	0	1	0
1	1	2	1



NAND gate

(d) F = $\overline{A} + \overline{B} = \overline{AB}$

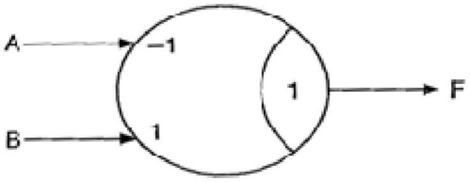
Inputs		Weighted sum	Output
A	B	w = –A – B	F
0	0	0	1
0	1	–1	1
1	0	–1	1
1	1	–2	0



NOR gate

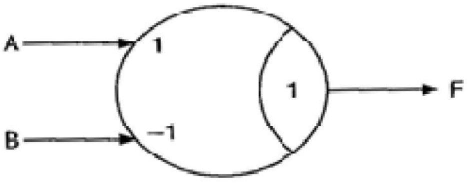
(e) F = $\overline{A} \cdot \overline{B} = \overline{A + B}$

Inputs		Weighted sum	Output
A	B	w = –A – B	F
0	0	0	1
0	1	–1	0
1	0	–1	0
1	1	–2	0



(f) F = $\overline{A}B$

Inputs		Weighted sum	Output
A	B	w = –A + B	F
0	0	0	0
0	1	1	1
1	0	–1	0
1	1	0	0



(g) F = $A\overline{B}$

Inputs		Weighted sum	Output
A	B	w = A – B	F
0	0	0	0
0	1	–1	0
1	0	1	1
1	1	0	0

BOOLEAN ALGEBRA

INTRODUCTION:-

- Switching circuits are also called logic circuits, gates circuits and digital circuits.
- Switching algebra is also called Boolean algebra.
- Boolean algebra is a system of mathematical logic. It is an algebraic system consisting of the set of elements (0,1), two binary operators called OR and AND and unary operator called NOT.
- It is the basic mathematical tool in the analysis and synthesis of switching circuits.
- It is a way to express logic functions algebraically.
- Any complex logic can be expressed by a Boolean function.
- The Boolean algebra is governed by certain well developed rules and laws.

AXIOMS AND LAWS OF BOOLEAN ALGEBRA:-

Axioms or postulates of Boolean algebra are set of logical expressions that are accepted without proof and upon which we can build a set of useful theorems. Actually, axioms are nothing more than the definitions of the three basic logic operations AND, OR and INVERTER. Each axiom can be interpreted as the outcome of an operation performed by a logic gate.

AND operation	**OR operation**	**NOT operation**
Axiom 1: 0 **.** 0 = 0	Axiom 5: 0 + 0 = 0	Axiom 9: $\overline{1}$ = 0
Axiom 2: 0 **.** 1 = 0	Axiom 6: 0 + 1 = 1	Axiom 10:$\overline{0}$ = 1
Axiom 3: 1 **.** 0 = 0	Axiom 7: 1 + 0 = 1	
Axiom 2: 1 **.** 1 = 1	Axiom 8: 1 + 1 = 1	

1. Complementation Laws:-

The term complement simply means to invert, i.e. to changes 0s to 1s and 1s to 0s. The five laws of complementation are as follows:

Law 1: $\overline{0}$ = 1
Law 2: $\overline{1}$ = 0
Law 3: if A = 0, then $\overline{A}$ = 1
Law 4: if A = 1,then$\overline{A}$ = 0
Law 5: $\overline{\overline{A}}$ = 0 (double complementation law)

2. OR Laws:-

The four OR laws are as follows

Law 1: A + 0 = 0(Null law)
Law 2: A + 1 = 1(Identity law)
Law 3: A + A = A
Law 4: A +$\overline{A}$ = 1

3. AND Laws:-

The four AND laws are as follows

Law 1: A . 0 = 0(Null law)
Law 2: A . 1 = 1(Identity law)
Law 3: A . A = A
Law 4: A .$\overline{A}$ = 0

4. Commutative Laws:-

Commutative laws allow change in position of AND or OR variables. There are two commutative laws.

Law 1: A + B = B + A

Proof

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

=

B	A	B+ A
0	0	0
0	1	1
1	0	1
1	1	1

Law 2: A . B = B . A

Proof

A	B	A . B
0	0	0
0	1	0
1	0	0
1	1	1

=

B	A	B. A
0	0	0
0	1	0
1	0	0
1	1	1

This law can be extended to any number of variables. For example

A.B. C = B. C. A = C. A. B = B. A. C

5. Associative Laws:-

The associative laws allow grouping of variables. There are 2 associative laws.

Law 1: (A + B) + C = A + (B + C)

Proof

A	B	C	A+B	(A+B)+C
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

=

A	B	C	B+C	A+(B+C)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Law 2: (A .B) C = A (B .C)

Proof

A	**B**	**C**	**AB**	**(AB)C**
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	1

=

A	**B**	**C**	**B.C**	**A(B.C)**
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

This law can be extended to any number of variables. For example

A(BCD) = (ABC)D = (AB) (CD)

6. Distributive Laws:-

The distributive laws allow factoring or multiplying out of expressions. There are two distributive laws.

Law 1: A (B + C) = AB + AC

Proof

A	**B**	**C**	**B+C**	**A(B+C)**
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

=

A	**B**	**C**	**AB**	**AC**	**A+(B+C)**
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

Law 2: A + BC = (A+B) (A+C)

Proof **RHS** = (A+B) (A+C)
= AA + AC + BA + BC
= A + AC + AB + BC
= A (1+ C + B) + BC
= A. 1 + BC (1 +C + B = 1 + B = 1)
= A + BC
= **LHS**

7. Redundant Literal Rule (RLR):-

Law 1: A + $\bar{A}$B = A + B

Proof

$A + \overline{A}B = (A + \overline{A})\,(A + B)$

$= 1.\,(A + B)$

$= A + B$

Law 2: $A(\overline{A} + B) = AB$

Proof

$A(\overline{A} + B) = A\overline{A} + AB$

$= 0 + AB$

$= AB$

8. Idempotence Laws:-

Idempotence means same value.

Law 1: A**.** A = A

Proof

If A = 0, then A**.** A = 0**.** 0 =0 = A

If A = 1, then A**.** A = 1**.** 1 = 1 = A

This law states that AND of a variable with itself is equal to that variable only.

Law 2: A + A = A

Proof

If A = 0, then A + A = 0 + 0 = 0 = A

If A = 1, then A + A = 1 + 1 = 1 = A

This law states that OR of a variable with itself is equal to that variable only.

9. Absorption Laws:-

There are two laws:

Law 1: A + A · B = A

Proof

A + A · B = A (1 + B) = A · 1 = A

A	**B**	**AB**	**A+AB**
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Law 2: A (A + B) = A

Proof

A (A + B) = A · A + A · B = A + AB = A(1 + B) = A · 1 = A

A	**B**	**A+B**	**A(A+B)**
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

10. Consensus Theorem (Included Factor Theorem):-

Theorem 1:

$AB + \bar{A}C + BC = AB + \bar{A}C$

Proof

LHS = $AB + \bar{A}C + BC$
$= AB + \bar{A}C + BC\,(A + \bar{A})$
$= AB + \bar{A}C + BCA + BC\bar{A}$
$= AB\,(1 + C) + \bar{A}C\,(1 + B)$
$= AB\,(1) + \bar{A}C\,(1)$
$= AB + \bar{A}C$
= **RHS**

Theorem 2:

$(A + B)(\bar{A} + C)(B + C) = (A + B)(\bar{A} + C)$

Proof

LHS = $(A + B)\,(\bar{A} + C)\,(B + C)$
$= (A\bar{A} + AC + B\bar{A} + BC)\,(B + C)$
$= (AC + BC + \bar{A}B)\,(B + C)$
$= ABC + BC + \bar{A}B + AC + BC + \bar{A}BC$
$= AC + BC + \bar{A}B$

RHS= $(A + B)\,(\bar{A} + C)$
$= A\bar{A} + AC + BC + \bar{A}B$
$= AC + BC + \bar{A}B$
= LHS

11. Transposition Theorem:-

Theorem:

$AB + \bar{A}C = (A + C)(\bar{A} + B)$

Proof

RHS= $(A + C)\,(\bar{A} + B)$
$= A\bar{A} + C\bar{A} + AB + CB$
$= 0 + \bar{A}C + AB + BC$
$= \bar{A}C + AB + BC\,(A + \bar{A})$
$= AB + ABC + \bar{A}C + \bar{A}BC$
$= AB + \bar{A}C$
= LHS

12. De Morgan’s Theorem:-

De Morgan’s theorem represents two laws in Boolean algebra.

Law 1: $\overline{A + B} = \bar{A} \cdot \bar{B}$

Proof

A	B	A + B	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

=

A	B	$\bar{A}$	$\bar{B}$	$\bar{A}\ \bar{B}$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

This law states that the complement of a sum of variables is equal to the product of their individual complements.

Law 2: $\overline{A \cdot B} = \overline{A} + \overline{B}$

Proof

A	B	A . B	$\overline{A . B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

=

A	B	$\overline{A}$	$\overline{B}$	$\overline{A} + \overline{B}$
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

This law states that the complement of a product of variables is equal to the sum of their individual complements.

DUALITY:-

The implication of the duality concept is that once a theorem or statement is proved, the dual also thus stand proved. This is called the principle of duality.

$$[f\ (A, B, C, \ldots, 0, 1, +, \cdot)]_d = f(A, B, C, \ldots, 1, 0, \cdot, +)$$

Relations between complement and dual

$$f_c\ (A, B, C, \ldots) = \overline{f\ (A, B, C, \ldots)} = f_d\ (\overline{A}, \overline{B}, \overline{C}, \ldots)$$

$$f_d\ (A, B, C, \ldots) = \overline{f\ (\overline{A}, \overline{B}, \overline{C}, \ldots)} = f_c\ (\overline{A}, \overline{B}, \overline{C}, \ldots)$$

The first relation states that the complement of a function f(A, B, C, …) can be obtained by complementing all the variables in the dual function f_d (A, B, C, …..).

The second relation states that the dual can be obtained by complementing all the literals in $\overline{f\ (A, B, C, \ldots)}$.

DUALS:-

Given expression	***Dual***
1. $\overline{0} = 1$	$\overline{1} = 0$
2. $0 \cdot 1 = 0$	$1 + 0 = 1$
3. $0 \cdot 0 = 0$	$1 + 1 = 1$
4. $1 \cdot 1 = 1$	$0 + 0 = 0$
5. $A \cdot 0 = 0$	$A + 1 = 1$
6. $A \cdot 1 = A$	$A + 0 = A$
7. $A \cdot A = A$	$A + A = A$
8. $A \cdot \overline{A} = 0$	$A + \overline{A} = 1$
9. $A \cdot B = B \cdot A$	$A + B = B + A$
10. $A \cdot (B \cdot C) = (A \cdot B) \cdot C$	$A + (B + C) = (A + B) + C$
11. $A \cdot (B + C) = AB + AC$	$A + BC = (A + B)(A + C)$
12. $A(A + B) = A$	$A + AB = A$
13. $A \cdot (A \cdot B) = A \cdot B$	$A + A + B = A + B$
14. $\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}\ \overline{B}$
15. $(A + B)(\overline{A} + C)(B + C) = (A + B)(\overline{A} + C)$	$AB + \overline{A}C + BC = AB + \overline{A}C$
16. $A + \overline{B}C = (A + \overline{B})(A + C)$	$A(\overline{B} + C) = A\overline{B} + AC$
17. $(A+C)(\overline{A}+B) = AB+\overline{A}C$	$AC+\overline{A}B=(A+B)(\overline{A}+C)$
18. $(A+B)(C+D) = AC + AD + BC + BD$	$(AB+CD) = (A+C)(A+D)(B+C)(B+D)$
19. $\underline{A + B = AB + }\overline{A}B + A\overline{B}$	$A\underline{B = (A+B)\ (\overline{A}+B)}\ (A+\overline{B})$
20. $\overline{AB} + \overline{A} + AB = 0$	$\overline{A + B} \cdot \overline{A} \cdot (A + B) = 1$

SUM - OF - PRODUCTS FORM:-

- This is also called disjunctive Canonical Form (DCF) or Expanded Sum of Products Form or Canonical Sum of Products Form.
- In this form, the function is the sum of a number of products terms where each product term contains all variables of the function either in complemented or uncomplemented form.
- This can also be derived from the truth table by finding the sum of all the terms that corresponds to those combinations for which ‘f ’ assumes the value 1.

  For example

  $$\begin{aligned} f(A, B, C) &= \bar{A}B + \bar{B}C \\ &= \bar{A}B(C + \bar{C}) + \bar{B}C(A + \bar{A}) \\ &= \bar{A}\,\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC + A\bar{B}C \end{aligned}$$
- The product term which contains all the variables of the functions either in complemented or uncomplemented form is called a minterm.
- The minterm is denoted as mo, m1, m2 … .
- An ‘n’ variable function can have 2n minterms.
- Another way of representing the function in canonical SOP form is the showing the sum of minterms for which the function equals to 1.

  For example

  $f(A, B, C) = m_1 + m_2 + m_3 + m_5$

  or

  $f(A, B, C) = \sum m(1, 2, 3, 5)$

  where ∑m represents the sum of all the minterms whose decimal codes are given the parenthesis.

PRODUCT- OF - SUMS FORM:-

- This form is also called as Conjunctive Canonical Form (CCF) or Expanded Product - of – Sums Form or Canonical Product Of Sums Form.
- This is by considering the combinations for which f = 0
- Each term is a sum of all the variables.
- The function $f(A, B, C) = (A + B + C\cdot C) + (A + B + C\cdot\bar{C})$

  $= (\bar{A} + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C})(A + B + C)(A + B + \bar{C})$
- The sum term which contains each of the ‘n’ variables in either complemented or uncomplemented form is called a maxterm.
- Maxterm is represented as M_0, M_1, M_2, …….

  Thus CCF of ‘f’ may be written as

  $f(A, B, C) = M_0 \cdot M_4 \cdot M_6 \cdot M_7$

  or

  $f(A, B, C) = (0, 4, 6, 7)$

  Where represented the product of all maxterms.

CONVERSION BETWEEN CANONICAL FORM:-

The complement of a function expressed as the sum of minterms equals the sum of minterms missing from the original function.

Example:-

$f(A, B, C) = \sum m(0,2,4,6,7)$

This has a complement that can be expressed as

$f\,\overline{(A, B, C)} = \sum m(1, 3, 5) = m_1 + m_3 + m_5$

If we complement f by De- Morgan’s theorem we obtain ‘f’ in a form.

$f = (\overline{m_1 + m_3 + m_5}) = \overline{m}_1 . \overline{m}_3 . \overline{m}_5$

$= M_1 M_3 M_5 = \prod M(1, 3, 5)$

Example:-

Expand $A(\bar{A} + B)(\bar{A} + B + \bar{C})$ to maxterms and minterms.

Solution:-

In POS form

$A(\bar{A} + B)(\bar{A} + B + \bar{C})$

$A = A + B\bar{B} + C\bar{C}$

$= (A + B)(A + \bar{B}) + C \cdot \bar{C}$

$= (A + B + C\bar{C})(A + B + C\bar{C})$

$= (A + B + C)(A + B + \bar{C})(A + \bar{B} + C)(A + \bar{B} + \bar{C})$

$A + B = A + B + C \cdot C$

$= (\bar{A} + B + C)(\bar{A} + B + \bar{C})$

Therefore

$A(\bar{A} + B)(\bar{A} + B + \bar{C})$

$= (A + B + C)(A + B + \bar{C})(A + \bar{B} + C)(A + \bar{B} + \bar{C})(\bar{A} + B + C)(\bar{A} + B + \bar{C})$

$= (000)(001)(010)(011)(100)(101)$

$= M_0 \cdot M_1 \cdot M_2 \cdot M_3 \cdot M_4 \cdot M_5$

$= \prod M(0, 1, 2, 3, 4, 5)$

The maxterms M_6 and M_7 are missing in the POS form.

So, the SOP form will contain the minterms 6 and 7

KARNAUGH MAP OR K- MAP:-

- The K- map is a chart or a graph, composed of an arrangement of adjacent cells, each representing a particular combination of variables in sum or product form.
- The K- map is systematic method of simplifying the Boolean expression.

TWO VARIABLE K- MAP:-

A two variable expression can have $2^2 = 4$ possible combinations of the input variables A and B.

Mapping of SOP Expression:-

- The 2 variable K-map has $2^2 = 4$ squares. These squares are called cells.
- A '1' is placed in any square indicates that corresponding minterm is included in the output expression, and a 0 or no entry in any square indicates that the corresponding minterm does not appear in the expression for output.

		B	
		0	1
A	0	$\bar{A}\bar{B}$	$\bar{A}B$
	1	$A\bar{B}$	AB

Example:-

Map expression $f = \bar{A}B + A\bar{B}$

Solution:-

The expression minterms is

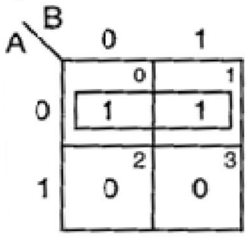
$F = m_1 + m_2 = m(1, 2)$

		B	
		0	1
A	0	0 0	1 1
	1	2 1	3 0

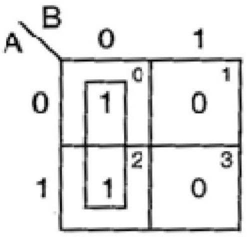
Minimization of SOP Expression:-

To minimize a Boolean expression given in the SOP form by using K- map, the adjacent squares having 1s, that is minterms adjacent to each other are combined to form larger squares to eliminate some variables.

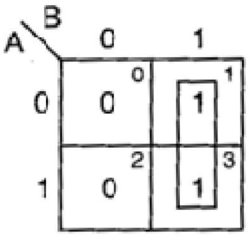
The possible minterm grouping in a two variable K- map are shown below



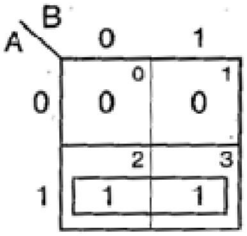
$f_1 = \bar{A}$



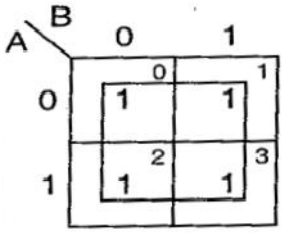
$f_2 = \bar{B}$



$f_3 = B$



$f_4 = A$



$f_5 = 1$

- Two minterms, which are adjacent to each other, can be combined to form a bigger square called 2 – square or a pair. This eliminates one variable that is not common to both the minterms.
- Two 2-squares adjacent to each other can be combined to form a 4- square. A 4- square eliminates 2 variables. A 4-square is called a quad.
- Consider only those variables which remain constant throughout the square, and ignore the variables which are varying. The non-complemented variable is the variable remaining constant as 1.The complemented variable is the variable remaining constant as a 0 and the variables are written as a product term.

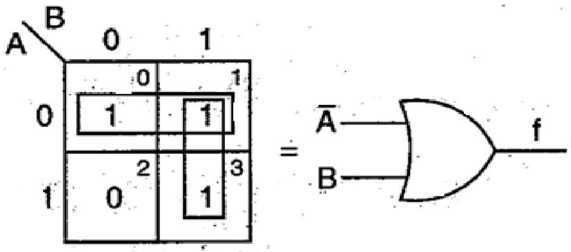
Example:-

Reduce the expression f= $\bar{A}\bar{B} + \bar{A}\ B + AB$ using mapping.

Solution:-

Expressed in terms of minterms, the given expression is

$$f = m_0 + m_1 + m_3 = \sum m\ (0, 1, 3)$$

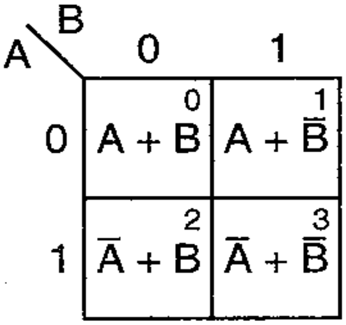


$f = \bar{A} + B$

F = $\bar{A}$ + B

Mapping of POS Expression:-

Each sum term in the standard POS expression is called a Maxterm. A function in two variables (A,B) has 4 possible maxterms, $A + B$, $A + \overline{B}$, $\overline{A} + B$ and $\overline{A} + \overline{B}$. They are represented as M_0, M_1, M_2 and M_3 respectively.



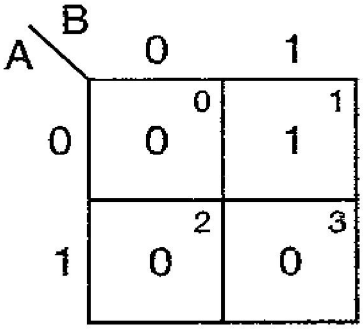
The maxterm of a two variable K-map

Example:-

Plot the expression f= $(A + B)(\overline{A} + B)(\overline{A} + \overline{B})$

Solution:-

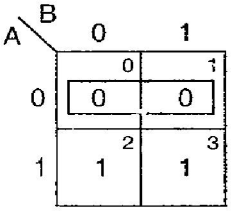
Expression interms of maxterms is f = ΠM (0, 2, 3)



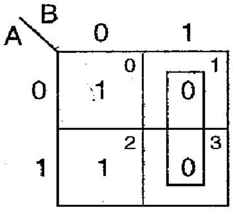
Minimization of POS Expressions:-

In POS form the adjacent 0s are combined into large square as possible. If the squares having complemented variable then the value remain constant as a 1 and the non-complemented variable if its value remains constant as a 0 along the entire square and then their sum term is written.

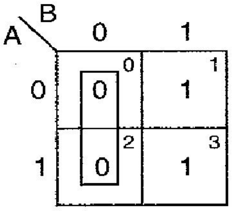
The possible maxterms grouping in a two variable K-map are shown below



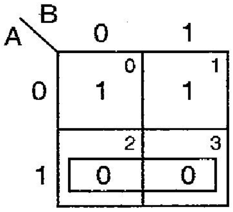
$f_1 = A$



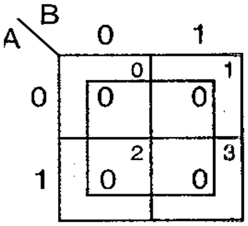
$f_2 = \overline{B}$



$f_3 = B$



$f_4 = \overline{A}$



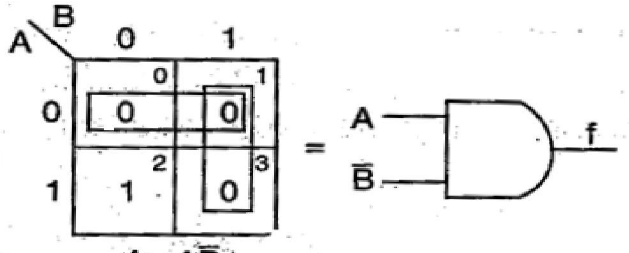
$f_5 = 0$

Example:-

Reduce the expression $f = (A + \bar{B})(\bar{A} + \bar{B})(A + B)$ using mapping

Solution:-

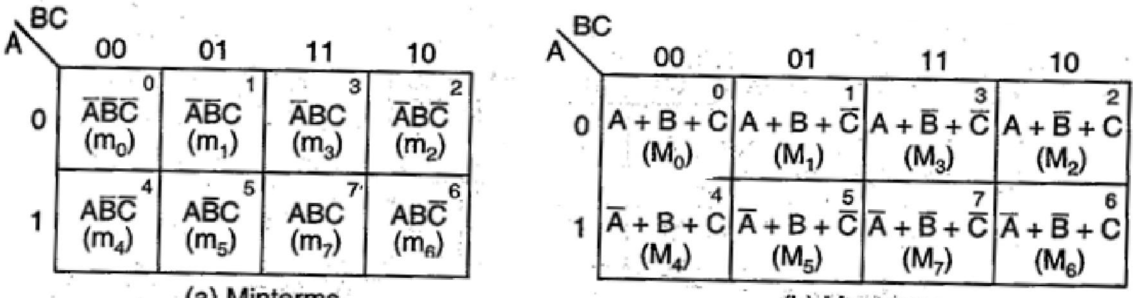
The given expression in terms of maxterms is $f = \prod M\,(0, 1, 3)$



$f = A\bar{B}$

THREE VARIABLE K- MAP:-

A function in three variables (A, B, C) can be expressed in SOP and POS form having eight possible combination. A three variable K- map have 8 squares or cells and each square on the map represents a minterm or maxterm is shown in the figure below.



A \ BC	00	01	11	10
0	0 $\bar{A}\bar{B}\bar{C}$ (m_0)	1 $\bar{A}\bar{B}C$ (m_1)	3 $\bar{A}BC$ (m_3)	2 $\bar{A}B\bar{C}$ (m_2)
1	4 $A\bar{B}\bar{C}$ (m_4)	5 $A\bar{B}C$ (m_5)	7 ABC (m_7)	6 $AB\bar{C}$ (m_6)

(a) Minterms

A \ BC	00	01	11	10
0	0 $A+B+C$ (M_0)	1 $A+B+\bar{C}$ (M_1)	3 $A+\bar{B}+\bar{C}$ (M_3)	2 $A+\bar{B}+C$ (M_2)
1	4 $\bar{A}+B+C$ (M_4)	5 $\bar{A}+B+\bar{C}$ (M_5)	7 $\bar{A}+\bar{B}+\bar{C}$ (M_7)	6 $\bar{A}+\bar{B}+C$ (M_6)

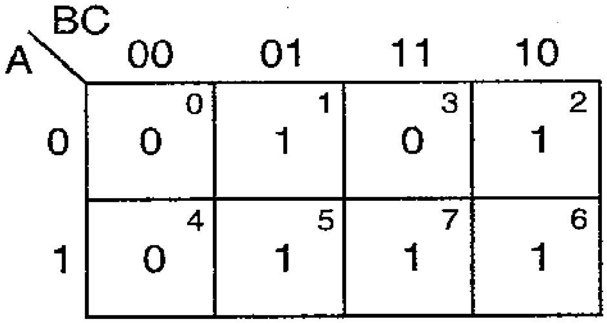
(b) Maxterms

Example:-

Map the expression $f = \bar{A}\bar{B}C + A\bar{B}C + \bar{A}B\bar{C} + AB\bar{C} + ABC$

Solution:-

So in the SOP form the expression is $f = \sum m\,(1, 5, 2, 6, 7)$



A \ BC	00	01	11	10
0	0 0	1 1	3 0	2 1
1	4 0	5 1	7 1	6 1

Example:-

Map the expression $f = (A + B + C)(\bar{A} + B + \bar{C})(\bar{A} + \bar{B} + \bar{C})(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + C)$

Solution:-

So in the POS form the expression is $f = \prod M\,(0, 5, 7, 3, 6)$

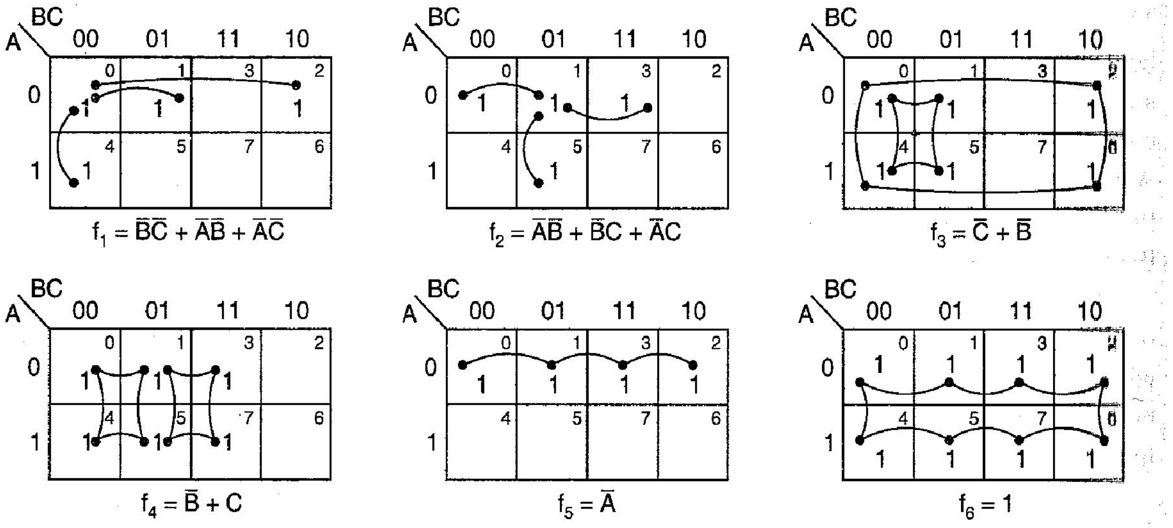
A \ BC	00	01	11	10
0	0 0	1 1	3 0	2 1
1	4 1	5 0	7 0	6 0

Minimization of SOP and POS Expressions:-

For reducing the Boolean expressions in SOP (POS) form the following steps are given below

- Draw the K-map and place 1s (0s) corresponding to the minterms (maxterms) of the SOP (POS) expression.
- In the map 1s (0s) which are not adjacent to any other 1(0) are the isolated minterms (maxterms). They are to be read as they are because they cannot be combined even into a 2-square.
- For those 1s (0s) which are adjacent to only one other 1(0) make them pairs (2 squares).
- For quads (4- squares) and octet (8 squares) of adjacent 1s (0s) even if they contain some 1s (0s) which have already been combined. They must geometrically form a square or a rectangle.
- For any 1s (0s) that have not been combined yet then combine them into bigger squares if possible.
- Form the minimal expression by summing (multiplying) the product (sum) terms of all the groups.

Some of the possible combinations of minterms in SOP form



$f_1 = \bar{B}\bar{C} + \bar{A}\bar{B} + \bar{A}\bar{C}$

$f_2 = \bar{A}\bar{B} + \bar{B}C + \bar{A}C$

$f_3 = \bar{C} + \bar{B}$

$f_4 = \bar{B} + C$

$f_5 = \bar{A}$

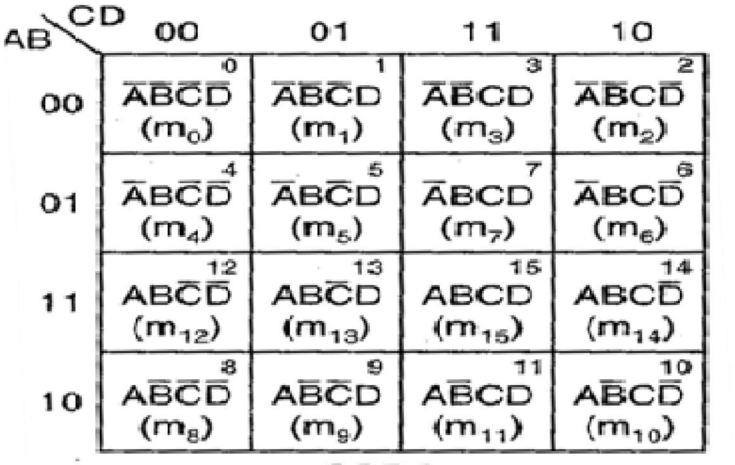
$f_6 = 1$

These possible combinations are also for POS but 1s are replaced by 0s.

FOUR VARIABLE K-MAP:-

A four variable (A, B, C, D) expression can have 2^4 = 16 possible combinations of input variables. A four variable K-map has 2^4 = 16 squares or cells and each square on the map represents either a minterm or a maxterm as shown in the figure below. The binary number designations of the rows and columns are in the gray code. The binary numbers along the top of the map indicate the conditions of C and D along any column and binary numbers along left side indicate the conditions of A and B along any row. The numbers in the top right corners of the squares indicate the minterm or maxterm desginations.

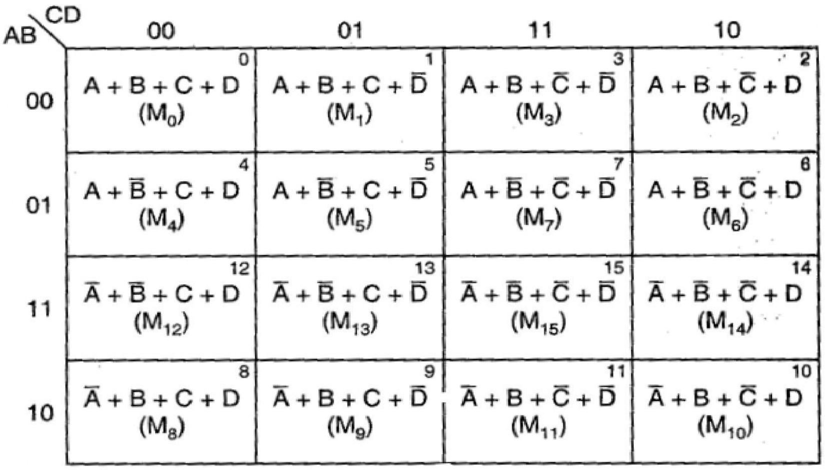
SOP FORM



AB \ CD	00	01	11	10
00	0 $\overline{A}\overline{B}\overline{C}\overline{D}$ (m_0)	1 $\overline{A}\overline{B}\overline{C}D$ (m_1)	3 $\overline{A}\overline{B}CD$ (m_3)	2 $\overline{A}\overline{B}C\overline{D}$ (m_2)
01	4 $\overline{A}B\overline{C}\overline{D}$ (m_4)	5 $\overline{A}B\overline{C}D$ (m_5)	7 $\overline{A}BCD$ (m_7)	6 $\overline{A}BC\overline{D}$ (m_6)
11	12 $AB\overline{C}\overline{D}$ (m_{12})	13 $AB\overline{C}D$ (m_{13})	15 $ABCD$ (m_{15})	14 $ABC\overline{D}$ (m_{14})
10	8 $A\overline{B}\overline{C}\overline{D}$ (m_8)	9 $A\overline{B}\overline{C}D$ (m_9)	11 $A\overline{B}CD$ (m_{11})	10 $A\overline{B}C\overline{D}$ (m_{10})

SOP form

POS FORM



AB \ CD	00	01	11	10
00	0 $A+B+C+D$ (M_0)	1 $A+B+C+\overline{D}$ (M_1)	3 $A+B+\overline{C}+\overline{D}$ (M_3)	2 $A+B+\overline{C}+D$ (M_2)
01	4 $A+\overline{B}+C+D$ (M_4)	5 $A+\overline{B}+C+\overline{D}$ (M_5)	7 $A+\overline{B}+\overline{C}+\overline{D}$ (M_7)	6 $A+\overline{B}+\overline{C}+D$ (M_6)
11	12 $\overline{A}+\overline{B}+C+D$ (M_{12})	13 $\overline{A}+\overline{B}+C+\overline{D}$ (M_{13})	15 $\overline{A}+\overline{B}+\overline{C}+\overline{D}$ (M_{15})	14 $\overline{A}+\overline{B}+\overline{C}+D$ (M_{14})
10	8 $\overline{A}+B+C+D$ (M_8)	9 $\overline{A}+B+C+\overline{D}$ (M_9)	11 $\overline{A}+B+\overline{C}+\overline{D}$ (M_{11})	10 $\overline{A}+B+\overline{C}+D$ (M_{10})

Minimization of SOP and POS Expressions:-

For reducing the Boolean expressions in SOP (POS) form the following steps are given below

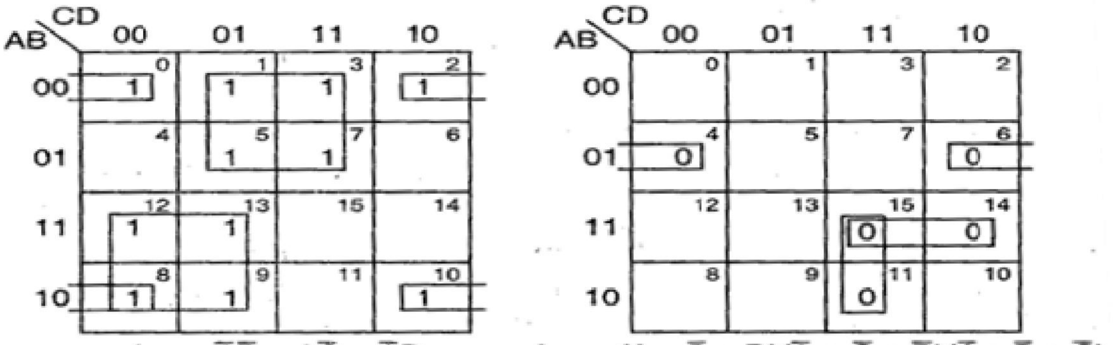
- Draw the K-map and place 1s (0s) corresponding to the minterms (maxterms) of the SOP (POS) expression.
- In the map 1s (0s) which are not adjacent to any other 1(0) are the isolated minterms (maxterms). They are to be read as they are because they cannot be combined even into a 2-square.
- For those 1s (0s) which are adjacent to only one other 1(0) make them pairs (2 squares).
- For quads (4- squares) and octet (8 squares) of adjacent 1s (0s) even if they contain some 1s (0s) which have already been combined. They must geometrically form a square or a rectangle.
- For any 1s (0s) that have not been combined yet then combine them into bigger squares if possible.
- Form the minimal expression by summing (multiplying) the product (sum) terms of all the groups.

Example:-

Reduce using mapping the expression f = ∑ m (0, 1, 2, 3, 5, 7, 8, 9, 10, 12, 13)

Solution:-

The given expression in POS form is f = ∏ M (4, 6, 11, 14, 15) and in SOP form f = ∑ m (0, 1, 2, 3, 5, 7, 8, 9, 10, 12, 13)



$f_{min} = \overline{B}\overline{D} + A\overline{C} + \overline{A}D$
(a) SOP K-map

$f_{min} = (A + \overline{B} + D)(\overline{A} + \overline{C} + \overline{D})(\overline{A} + \overline{B} + \overline{C})$
(b) POS K-map

The minimal SOP expression is $f_{min}= \overline{B}\overline{D} + A\overline{C} + \overline{A}D$

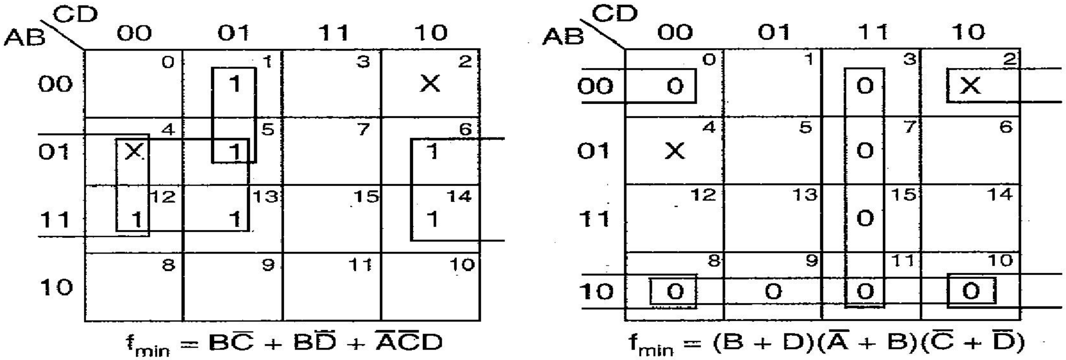
The minimal POS expression is $f_{min} =(A +\overline{B} + D) (\overline{A} + \overline{C} + D) (A + B + C)$

DON'T CARE COMBINATIONS:-

The combinations for which the values of the expression are not specified are called don't care combinations or optional combinations and such expression stand incompletely specified. The output is a don't care for these invalid combinations. The don't care terms are denoted by d or X. During the process of designing using SOP maps, each don't care is treated as 1 to reduce the map otherwise it is treated as 0 and left alone. During the process of designing using POS maps, each don't care is treated as 0 to reduce the map otherwise it is treated as 1 and left alone.

A standard SOP expression with don't cares can be converted into standard POS form by keeping the don't cares as they are, and the missing minterms of the SOP form are written as the maxterms of the POS form. Similarly, to convert a standard POS expression with don't cares can be converted into standard SOP form by keeping the don't cares as they are, and the missing maxterms of the POS form are written as the minterms of the SOP form.

Example:-
Reduce the expression f = ∑ m(1, 5, 6, 12, 13, 14) + d(2, 4) using K- map.
Solution:-
The given expression in SOP form is f = ∑ m (1, 5, 6, 12, 13, 14) + d(2, 4)
The given expression in POS form is f = π M (0, 3, 7, 8, 9, 10, 11,15) + d(2, 4)



$f_{min} = B\overline{C} + B\overline{D} + \overline{A}\overline{C}D$
(a) SOP K-map

$f_{min} = (B + D)(\overline{A} + B)(\overline{C} + \overline{D})$
(b) POS K-map

The minimal of SOP expression is $f_{min} = B\overline{C} + B\overline{D} +\overline{A}\overline{C}D$

The minimal of POS expression is $f_{min} = (B + D)(\overline{A} + B) (\overline{C} + \overline{D})$

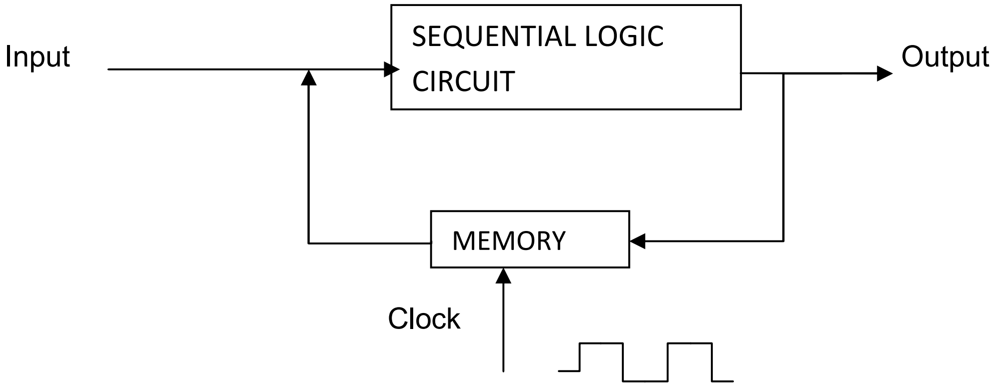
SEQUENTIAL LOGIC CIRCUIT

SEQUENTIAL CIRCUIT:-

- It is a circuit whose output depends upon the present input, previous output and the sequence in which the inputs are applied.

HOW THE SEQUENTIAL CIRCUIT IS DIFFERENT FROM COMBINATIONAL CIRCUIT? :-

- In combinational circuit output depends upon present input at any instant of time and do not use memory. Hence previous input does not have any effect on the circuit. But sequential circuit has memory and depends upon present input and previous output.
- Sequential circuits are slower than combinational circuits and these sequential circuits are harder to design.



[Block diagram of Sequential Logic Circuit]

- The data stored by the memory element at any given instant of time is called the present state of sequential circuit.

TYPES:-

Sequential logic circuits (SLC) are classified as

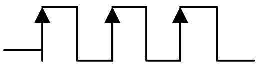
(i) Synchronous SLC
(ii) Asynchronous SLC

- The SLC that are controlled by clock are called synchronous SLC and those which are not controlled by a clock are asynchronous SLC.
- Clock:- A recurring pulse is called a clock.

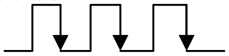
FLIP-FLOP AND LATCH:-

- A flip-flop or latch is a circuit that has two stable states and can be used to store information.
- A flip-flop is a binary storage device capable of storing one bit of information. In a stable state, the output of a flip-flop is either 0 or 1.
- Latch is a non-clocked flip-flop and it is the building block for the flip-flop.
- A storage element in digital circuit can maintain a binary state indefinitely until directed by an input signal to switch state.
- Storage element that operate with signal level are called latches and those operate with clock transition are called as flip-flops.

- The circuit can be made to change state by signals applied to one or more control inputs and will have one or two outputs.
- A flip-flop is called so because its output either flips or flops meaning to switch back and forth.
- A flip-flop is also called a bi-stable multi-vibrator as it has two stable states. The input signals which command the flip-flop to change state are called excitations.
- Flip-flops are storage devices and can store 1 or 0.
- Flip-flops using the clock signal are called clocked flip-flops. Control signals are effective only if they are applied in synchronization with the clock signal.
- Clock-signals may be positive-edge triggered or negative-edge triggered.
- Positive-edge triggered flip-flops are those in which state transitions take place only at positive- going edge of the clock pulse.



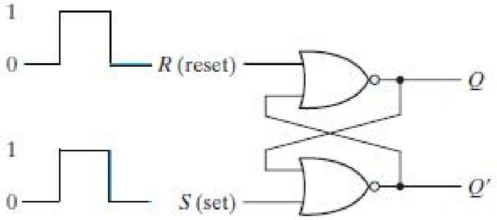
- Negative-edge triggered flip-flops are those in which state transition take place only at negative- going edge of the clock pulse.



- Some common type of flip-flops include
  a) SR (set-reset) F-F
  b) D (data or delay) F-F
  c) T (toggle) F-F and
  d) JK F-F

SR latch:-

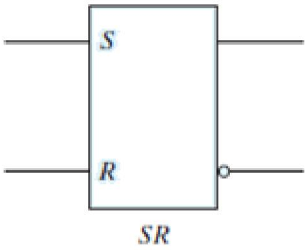
- The SR latch is a circuit with two cross-coupled NOR gates or two cross-coupled NAND gates.
- It has two outputs labeled Q and Q'. Two inputs are there labeled S for set and R foe reset.
- The latch has two useful states. When Q=0 and Q'=1 the condition is called reset state and when Q=1 and Q'=0 the condition is called set state.
- Normally Q and Q' are complement of each other.
- The figure represents a SR latch with two cross-coupled NOR gates. The circuit has NOR gates and as we know if any one of the input for a NOR gate is HIGH then its output will be LOW and if both the inputs are LOW then only the output will be HIGH.



- Under normal conditions, both inputs of the latch remain at 0 unless the state has to be changed. The application of a momentary 1 to the S input causes the latch to go to the set state. The S input must go back to 0 before any other changes take place, in order to avoid the occurrence of an undefined next state that results from the forbidden input condition.
- The first condition (S = 1, R = 0) is the action that must be taken by input S to bring the circuit to the set state. Removing the active input from S leaves the circuit in the same state. After both inputs return to 0, it is then possible to shift to the reset state by momentary applying a 1 to the R input. The 1 can then be removed from R, whereupon the circuit remains in the reset state. When both inputs S and R are equal to 0, the latch can be in either the set or the reset state, depending on which input was most recently a 1.

- If a 1 is applied to both the S and R inputs of the latch, both outputs go to 0. This action produces an undefined next state, because the state that results from the input transitions depends on the order in which they return to 0. It also violates the requirement that outputs be the complement of each other. In normal operation, this condition is avoided by making sure that 1's are not applied to both inputs simultaneously.
- Truth table for SR latch designed with NOR gates is shown below.

Input		Output				Comment
S	R	Q	Q'	Q_{Next}	Q'_{Next}	
0	0	0	1	0	1	No change
0	0	1	0	1	0	
0	1	0	1	0	1	Reset
0	1	1	0	0	1	
1	0	0	1	1	0	Set
1	0	1	0	1	0	
1	1	0	1	X	X	Prohibited state
1	1	1	0	X	X	



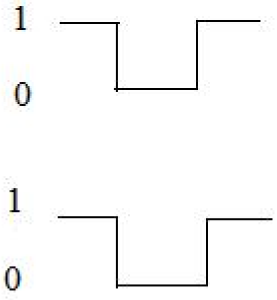
Symbol for SR NOR Latch

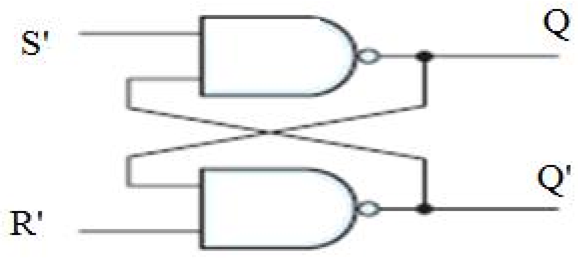
Racing Condition:-

In case of a SR latch when S=R=1 input is given both the output will try to become 0. This is called Racing condition.

SR latch using NAND gate:-

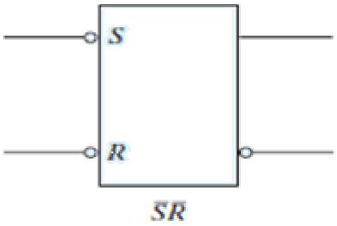
- The below figure represents a SR latch with two cross-coupled NAND gates. The circuit has NAND gates and as we know if any one of the input for a NAND gate is LOW then its output will be HIGH and if both the inputs are HIGH then only the output will be LOW.
- It operates with both inputs normally at 1, unless the state of the latch has to be changed. The application of 0 to the S input causes output Q to go to 1, putting the latch in the set state. When the S input goes back to 1, the circuit remains in the set state. After both inputs go back to 1, we are allowed to change the state of the latch by placing a 0 in the R input. This action causes the circuit to go to the reset state and stay there even after both inputs return to 1.





- The condition that is forbidden for the NAND latch is both inputs being equal to 0 at the same time, an input combination that should be avoided.

In comparing the NAND with the NOR latch, note that the input signals for the NAND require the complement of those values used for the NOR latch. Because the NAND latch requires a 0 signal to change its state, it is sometimes referred to as an S'R' latch. The primes (or, sometimes, bars over the letters) designate the fact that the inputs must be in their complement form to activate the circuit.



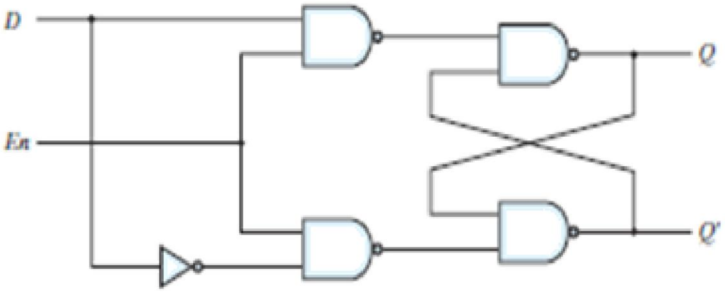
The above represents the symbol for inverted SR latch or SR latch using NAND gate.

Truth table for SR latch using NAND gate or Inverted SR latch

S	R	Q_{next}	Q'_{next}
0	0	Race	Race
0	1	0	1 (Reset)
1	0	1	0 (Set)
1	1	Q (No change)	Q' (No change)

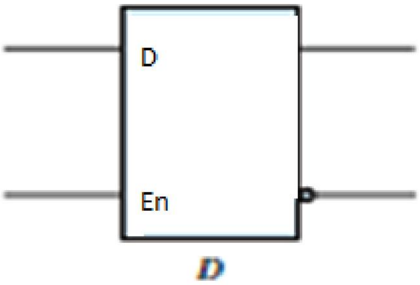
D LATCH:-

- One way to eliminate the undesirable condition of the indeterminate state in the SR latch is to ensure that inputs S and R are never equal to 1 at the same time.



(a) Logic diagram

- This is done in the D latch. This latch has only two inputs: D (data) and En (enable).
- The D input goes directly to the S input, and its complement is applied to the R input.



(Symbol for D-Latch)

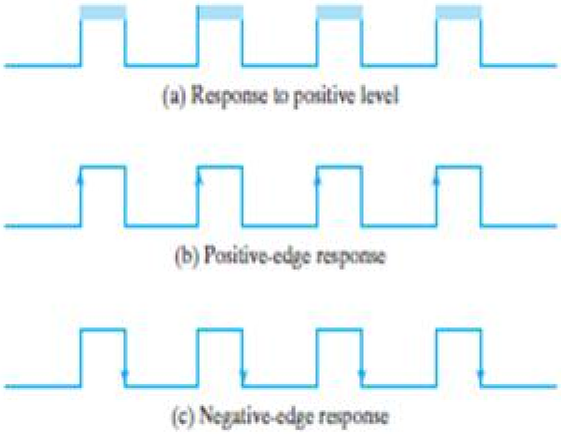
- As long as the enable input is at 0, the cross-coupled SR latch has both inputs at the 1 level and the circuit can't change state regardless of the value of D.
- The below represents the truth table for the D-latch.

En	D	Next State of Q
0	X	No change
1	0	Q=0;Reset State
1	1	Q=1;Set State

- The D input is sampled when En = 1. If D = 1, the Q output goes to 1, placing the circuit in the set state. If D = 0, output Q goes to 0, placing the circuit in the reset state. This situation provides a path from input D to the output, and for this reason, the circuit is often called a TRANSPARENT latch.

TRIGGERING METHODS:-

- The state of a latch or flip-flop is switched by a change in the control input. This momentary change is called a trigger, and the transition it causes is said to trigger the flip-flop.
- Flip-flop circuits are constructed in such a way as to make them operate properly when they are part of a sequential circuit that employs a common clock.
- The problem with the latch is that it responds to a change in the level of a clock pulse. For proper operation of a flip-flop it should be triggered only during a signal transition.
- This can be accomplished by eliminating the feedback path that is inherent in the operation of the sequential circuit using latches. A clock pulse goes through two transitions: from 0 to 1 and the return from 1 to 0.
- A ways that a latch can be modified to form a flip-flop is to produce a flip-flop that triggers only during a signal transition (from 0 to 1 or from 1 to 0) of the synchronizing signal (clock) and is disabled during the rest of the clock pulse.



(a) Response to positive level

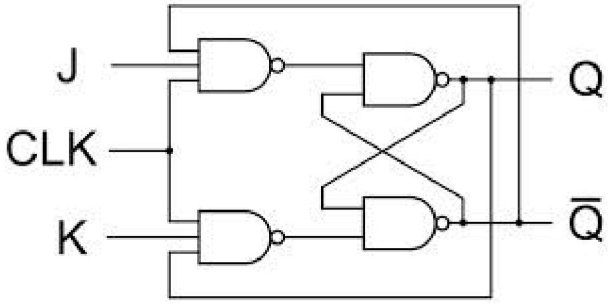
(b) Positive-edge response

(c) Negative-edge response

JK FLIP-FLOP:-

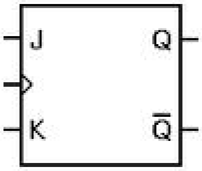
- The JK flip-flop can be constructed by using basic SR latch and a clock. In this case the outputs Q and Q' are returned back and connected to the inputs of NAND gates.
- This simple JK flip Flop is the most widely used of all the flip-flop designs and is considered to be a universal flip-flop circuit.
- The sequential operation of the JK flip flop is exactly the same as for the previous SR flip-flop with the same "Set" and "Reset" inputs.
- The difference this time is that the "JK flip flop" has no invalid or forbidden input states of the SR Latch even when S and R are both at logic "1".

(The below diagram shows the circuit diagram of a JK flip-flop)



- The JK flip flop is basically a gated SR Flip-flop with the addition of a clock input circuitry that prevents the illegal or invalid output condition that can occur when both inputs S and R are equal to logic level "1".
- Due to this additional clocked input, a JK flip-flop has four possible input combinations, "logic 1", "logic 0", "no change" and "toggle".

- The symbol for a JK flip flop is similar to that of an SR bistable latch except the clock input.



(The above diagram shows the symbol of a JK flip-flop.)

- Both the S and the R inputs of the SR bi-stable have now been replaced by two inputs called the J and K inputs, respectively after its inventor Jack and Kilby. Then this equates to: J = S and K = R.
- The two 2-input NAND gates of the gated SR bi-stable have now been replaced by two 3-input NAND gates with the third input of each gate connected to the outputs at Q and Q'.
- This cross coupling of the SR flip-flop allows the previously invalid condition of S = "1" and R = "1" state to be used to produce a "toggle action" as the two inputs are now interlocked.
- If the circuit is now "SET" the J input is inhibited by the "0" status of Q' through the lower NAND gate. If the circuit is "RESET" the K input is inhibited by the "0" status of Q through the upper NAND gate. As Q and Q' are always different we can use them to control the input.

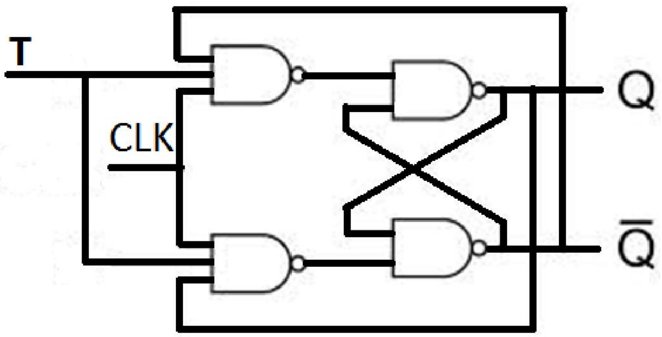
(Truth table for JK flip-flop)

Input		Output		Comment
J	K	Q	Q_{next}	
0	0	0	0	No change
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

- When both inputs J and K are equal to logic "1", the JK flip flop toggles.

<u>T FLIP-FLOP:-</u>

- Toggle flip-flop or commonly known as T flip-flop.
- This flip-flop has the similar operation as that of the JK flip-flop with both the inputs J and K are shorted i.e. both are given the common input.



- Hence its truth table is same as that of JK flip-flop when J=K= 0 and J=K=1.So its truth table is as follows.

T	Q	Q_{next}	Comment
0	0	0	No change
	1	1	
1	0	1	Toggles
	1	0	

CHARACTERISTIC TABLE:-

- A characteristic table defines the logical properties of a flip-flop by describing its operation in tabular form.
- The next state is defined as a function of the inputs and the present state.
- Q (t) refers to the present state and Q (t + 1) is the next.
- Thus, Q (t) denotes the state of the flip-flop immediately before the clock edge, and Q(t + 1) denotes the state that results from the clock transition.
- The characteristic table for the JK flip-flop shows that the next state is equal to the present state when inputs J and K are both equal to 0. This condition can be expressed as Q (t + 1) = Q (t), indicating that the clock produces no change of state.

Characteristic Table Of JK Flip-Flop

J	K	Q(t+1)
0	0	Q(t) No change
0	1	0 Reset
1	0	1 Set
1	1	Q'(t) Complement

- When K = 1 and J = 0, the clock resets the flip-flop and Q(t + 1) = 0. With J = 1 and K = 0, the flip-flop sets and Q(t + 1) = 1. When both J and K are equal to 1, the next state changes to the complement of the present state, a transition that can be expressed as Q(t + 1) = Q'(t).
- The characteristic equation for JK flip-flop is represented as

$$Q(t+1)= JQ' + K'Q$$

Characteristic Table of D Flip-Flop

D	Q(t+1)
0	0
1	1

- The next state of a D flip-flop is dependent only on the D input and is independent of the present state.
- This can be expressed as Q (t + 1) = D. It means that the next-state value is equal to the value of D. Note that the D flip-flop does not have a “no-change” condition and its characteristic equation is written as Q(t+1)=D.

Characteristic Table of T Flip-Flop

T	Q(t+1)
0	Q(t) No change
1	Q'(t) Complement

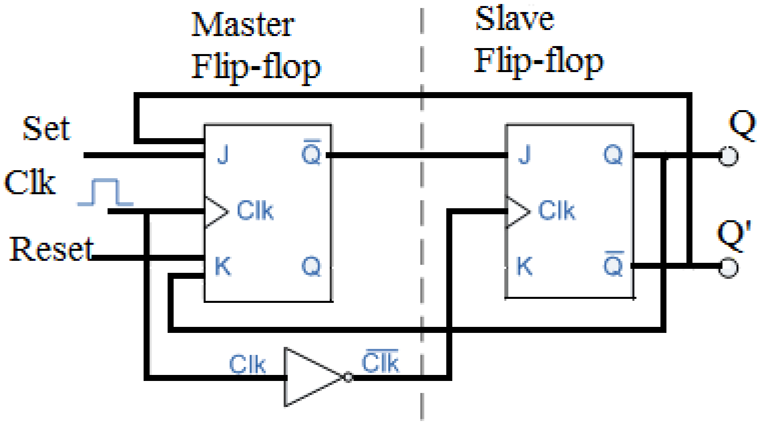
- The characteristic table of T flip-flop has only two conditions: When T = 0, the clock edge does not change the state; when T = 1, the clock edge complements the state of the flip-flop and the characteristic equation is

$$Q(t+1)= T \oplus Q = T'Q +TQ'$$

MASTER-SLAVE JK FLIP-FLOP:-

- The Master-Slave Flip-Flop is basically two gated SR flip-flops connected together in a series configuration with the slave having an inverted clock pulse.
- The outputs from Q and Q' from the "Slave" flip-flop are fed back to the inputs of the "Master" with the outputs of the "Master" flip flop being connected to the two inputs of the "Slave" flip flop.
- This feedback configuration from the slave's output to the master's input gives the characteristic toggle of the JK flip flop as shown below.

The Master-Slave JK Flip Flop



- The input signals J and K are connected to the gated "master" SR flip flop which "locks" the input condition while the clock (Clk) input is "HIGH" at logic level "1".
- As the clock input of the "slave" flip flop is the inverse (complement) of the "master" clock input, the "slave" SR flip flop does not toggle.
- The outputs from the "master" flip flop are only "seen" by the gated "slave" flip flop when the clock input goes "LOW" to logic level "0".
- When the clock is "LOW", the outputs from the "master" flip flop are latched and any additional changes to its inputs are ignored.
- The gated "slave" flip flop now responds to the state of its inputs passed over by the "master" section.
- Then on the "Low-to-High" transition of the clock pulse the inputs of the "master" flip flop are fed through to the gated inputs of the "slave" flip flop and on the "High-to-Low" transition the same inputs are reflected on the output of the "slave" making this type of flip flop edge or pulse-triggered.
- Then, the circuit accepts input data when the clock signal is "HIGH", and passes the data to the output on the falling-edge of the clock signal.
- In other words, the Master-Slave JK Flip flop is a "Synchronous" device as it only passes data with the timing of the clock signal.

FLIP-FLOP CONVERSIONS:-

SR Flip Flop to JK Flip Flop

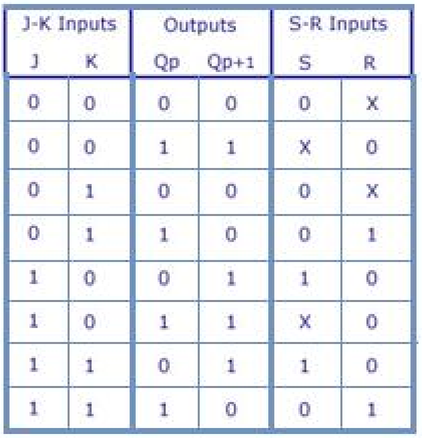
For this J and K will be given as external inputs to S and R. As shown in the logic diagram below, S and R will be the outputs of the combinational circuit.

The truth tables for the flip flop conversion are given below. The present state is represented by Qp and Qp+1 is the next state to be obtained when the J and K inputs are applied.

For two inputs J and K, there will be eight possible combinations. For each combination of J, K and Qp, the corresponding Qp+1 states are found. Qp+1 simply suggests the future values to be obtained by the JK flip flop after the value of Qp. The table is then completed by writing the values of S and R required to get each Qp+1 from the corresponding Qp. That is, the values of S and R that are required to change the state of the flip flop from Qp to Qp+1 are written.

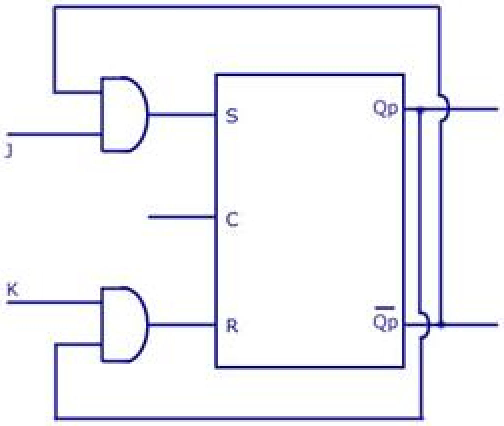
S-R Flip Flop to J-K Flip Flop

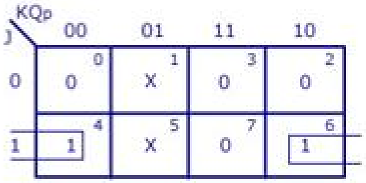
Conversion Table



J-K Inputs		Outputs		S-R Inputs	
J	K	Qp	Qp+1	S	R
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	X	0
1	1	0	1	1	0
1	1	1	0	0	1

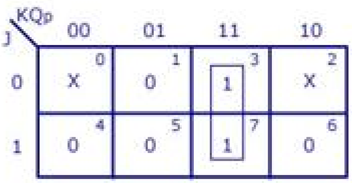
Logic Diagram





J \ KQp	00	01	11	10
0	0 0	1 X	3 0	2 0
1	4 1	5 X	7 0	6 1

$S = \overline{J}Qp$



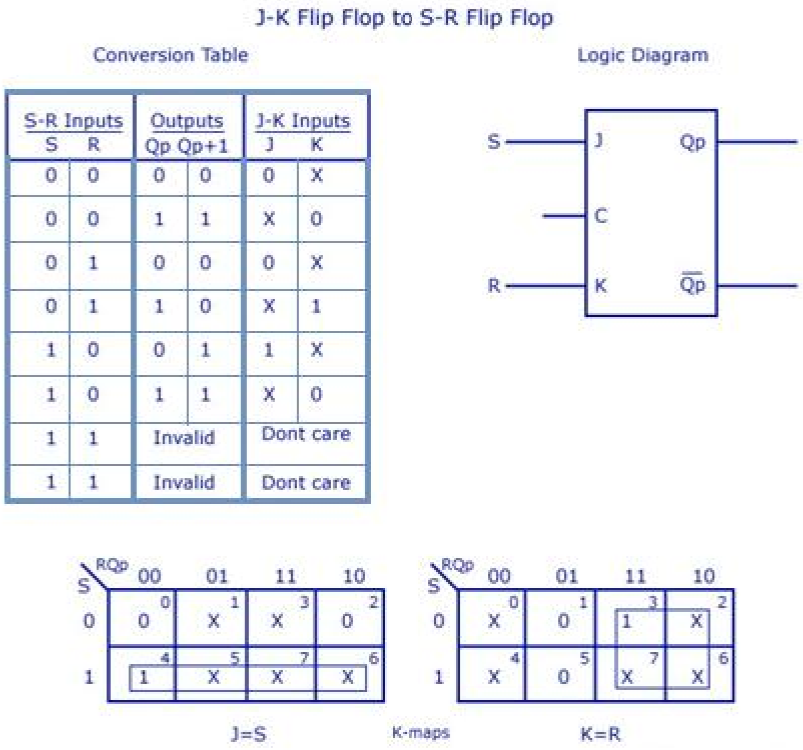
J \ KQp	00	01	11	10
0	0 X	1 0	3 1	2 X
1	4 0	5 0	7 1	6 0

$R = KQp$

K-Map

<u>JK Flip Flop to SR Flip Flop</u>

- This will be the reverse process of the above explained conversion. S and R will be the external inputs to J and K. J and K will be the outputs of the combinational circuit. Thus, the values of J and K have to be obtained in terms of S, R and Qp.
- A conversion table is to be written using S, R, Qp, Qp+1, J and K.
- For two inputs, S and R, eight combinations are made. For each combination, the corresponding Qp+1 outputs are found out.
- The outputs for the combinations of S=1 and R=1 are not permitted for an SR flip flop. Thus the outputs are considered invalid and the J and K values are taken as “don’t cares”.



J-K Flip Flop to S-R Flip Flop

Conversion Table

S-R Inputs		Outputs		J-K Inputs	
S	R	Qp	Qp+1	J	K
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	X	1
1	0	0	1	1	X
1	0	1	1	X	0
1	1	Invalid		Dont care	
1	1	Invalid		Dont care	

Logic Diagram

K-maps

S \ RQp	00	01	11	10
0	0	X	X	0
1	1	X	X	X

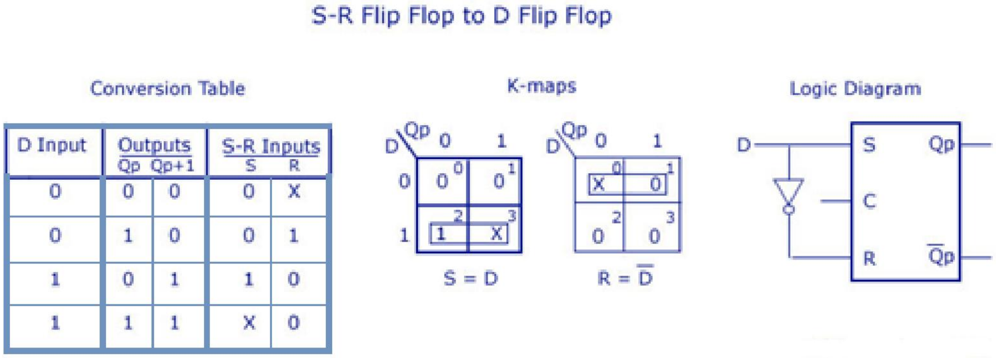
J=S

S \ RQp	00	01	11	10
0	X	0	1	X
1	X	0	X	X

K=R

SR Flip Flop to D Flip Flop

- S and R are the actual inputs of the flip flop and D is the external input of the flip flop.
- The four combinations, the logic diagram, conversion table, and the K-map for S and R in terms of D and Qp are shown below.



S-R Flip Flop to D Flip Flop

Conversion Table

D Input	Outputs		S-R Inputs	
	Qp	Qp+1	S	R
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0

K-maps

D \ Qp	0	1
0	0	0
1	1	X

S = D

D \ Qp	0	1
0	X	0
1	0	0

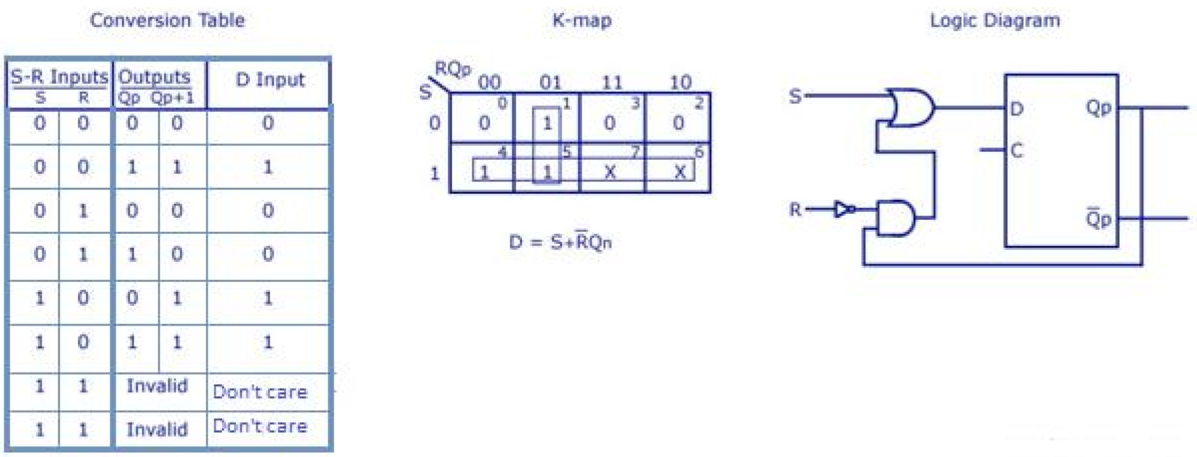
R = $\overline{D}$

Logic Diagram

D Flip Flop to SR Flip Flop

- D is the actual input of the flip flop and S and R are the external inputs. Eight possible combinations are achieved from the external inputs S, R and Qp.
- But, since the combination of S=1 and R=1 are invalid, the values of Qp+1 and D are considered as “don’t cares”.
- The logic diagram showing the conversion from D to SR, and the K-map for D in terms of S, R and Qp are shown below.

D Flip Flop to S-R Flip Flop



Conversion Table

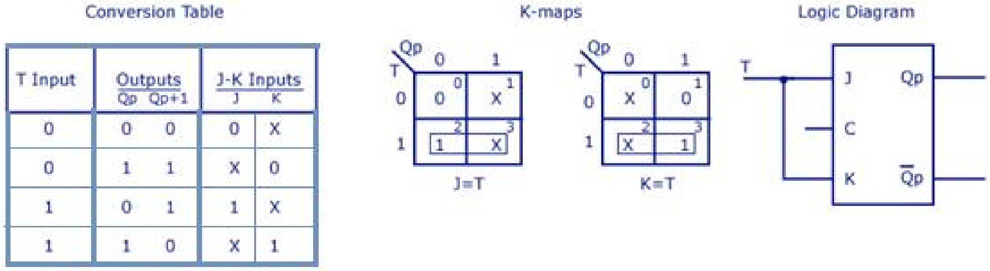
S-R Inputs		Outputs		D Input
S	R	Qp	Qp+1	
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	Invalid		Don't care
1	1	Invalid		Don't care

$D = S+\bar{R}Q_n$

JK Flip Flop to T Flip Flop:-

- J and K are the actual inputs of the flip flop and T is taken as the external input for conversion
- Four combinations are produced with T and Qp. J and K are expressed in terms of T and Qp.
- The conversion table, K-maps, and the logic diagram are given below.

J-K Flip Flop to T Flip Flop



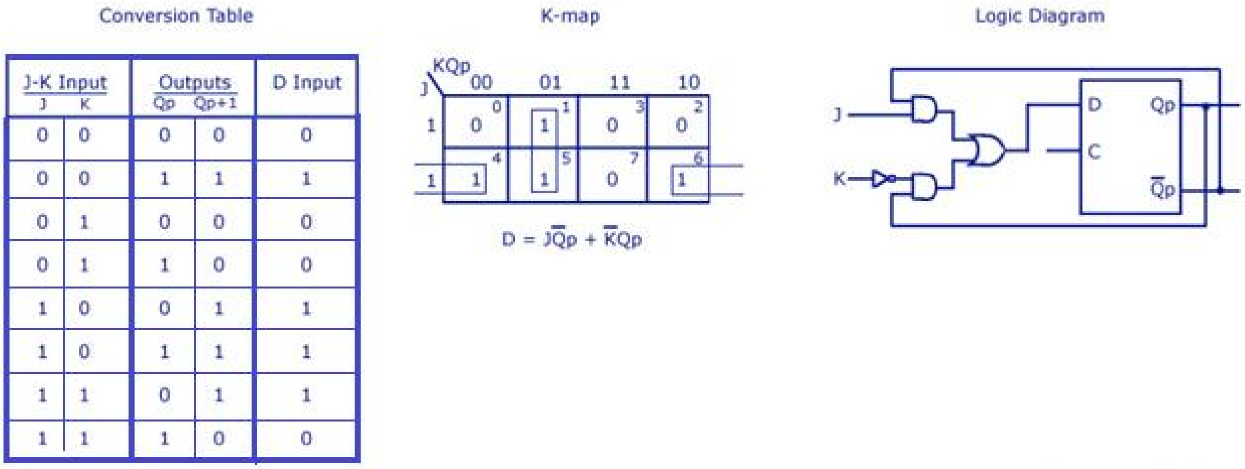
Conversion Table

T Input	Outputs		J-K Inputs	
	Qp	Qp+1	J	K
0	0	0	0	X
0	1	1	X	0
1	0	1	1	X
1	1	0	X	1

D Flip Flop to JK Flip Flop:-

- In this conversion, D is the actual input to the flip flop and J and K are the external inputs.
- J, K and Qp make eight possible combinations, as shown in the conversion table below. D is expressed in terms of J, K and Qp.
- The conversion table, the K-map for D in terms of J, K and Qp and the logic diagram showing the conversion from D to JK are given in the figure below.

D Flip Flop to J-K Flip Flop



Conversion Table

J-K Input		Outputs		D Input
J	K	Qp	Qp+1	
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

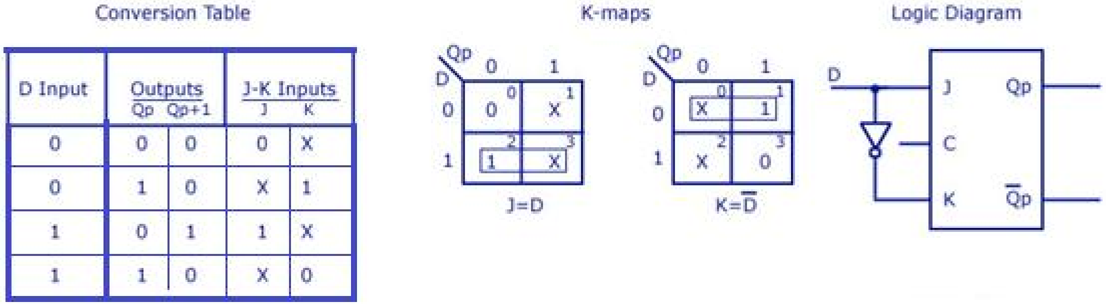
K-map

Logic Diagram

JK Flip Flop to D Flip Flop:-

- D is the external input and J and K are the actual inputs of the flip flop. D and Qp make four combinations. J and K are expressed in terms of D and Qp.
- The four combination conversion table, the K-maps for J and K in terms of D and Qp.

J-K Flip Flop to D Flip Flop



Conversion Table

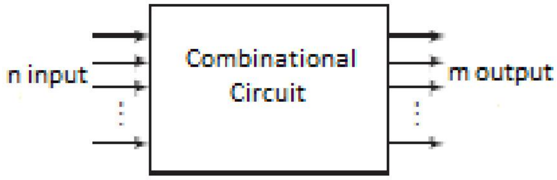
D Input	Outputs		J-K Inputs	
	Qp	Qp+1	J	K
0	0	0	0	X
0	1	0	X	1
1	0	1	1	X
1	1	0	X	0

K-maps

Logic Diagram

COMBINATIONAL LOGIC CIRCUIT

- A combinational circuit consists of logic gates whose outputs at any time are determined from only the present combination of inputs.
- A combinational circuit performs an operation that can be specified logically by a set of Boolean functions.
- It consists of an interconnection of logic gates. Combinational logic gates react to the values of the signals at their inputs and produce the value of the output signal, transforming binary information from the given input data to a required output data.
- A block diagram of a combinational circuit is shown in the below figure.
- The n input binary variables come from an external source; the m output variables are produced by the internal combinational logic circuit and go to an external destination.
- Each input and output variable exists physically as an analog signal whose values are interpreted to be a binary signal that represents logic 1and logic 0.

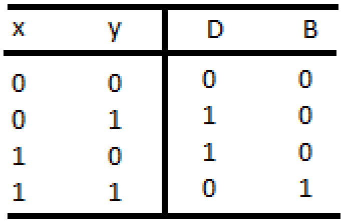


BINARY ADDER–SUBTRACTOR:-

- Digital computers perform a variety of information-processing tasks. Among the functions encountered are the various arithmetic operations.
- The most basic arithmetic operation is the addition of two binary digits. This simple addition consists of four possible elementary operations: 0 + 0 = 0, 0 + 1 = 1, 1 + 0 = 1, and 1 + 1 = 10.
- The first three operations produce a sum of one digit, but when both augend and addend bits are equal to 1; the binary sum consists of two digits. The higher significant bit of this result is called a carry.
- When the augend and addend numbers contain more significant digits, the carry obtained from the addition of two bits is added to the next higher order pair of significant bits.
- A combinational circuit that performs the addition of two bits is called a half adder.
- One that performs the addition of three bits (two significant bits and a previous carry) is a full adder. The names of the circuits stem from the fact that two half adders can be employed to implement a full adder.

HALF ADDER:-

- This circuit needs two binary inputs and two binary outputs.
- The input variables designate the augend and addend bits; the output variables produce the sum and carry. Symbols x and y are assigned to the two inputs and S (for sum) and C (for carry) to the outputs.
- The truth table for the half adder is listed in the below table.
- The C output is 1 only when both inputs are 1. The S output represents the least significant bit of the sum.
- The simplified Boolean functions for the two outputs can be obtained directly from the truth table.



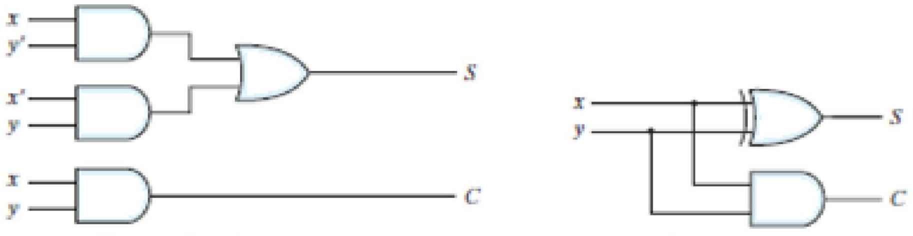
x	y	D	B
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Truth Table

- The simplified sum-of-products expressions are

$$S = x'y + xy'$$
$$C = xy$$

- The logic diagram of the half adder implemented in sum of products is shown in the below figure. It can be also implemented with an exclusive-OR and an AND gate.



(a) $S = xy' + x'y$
$C = xy$

(b) $S = x \oplus y$
$C = xy$

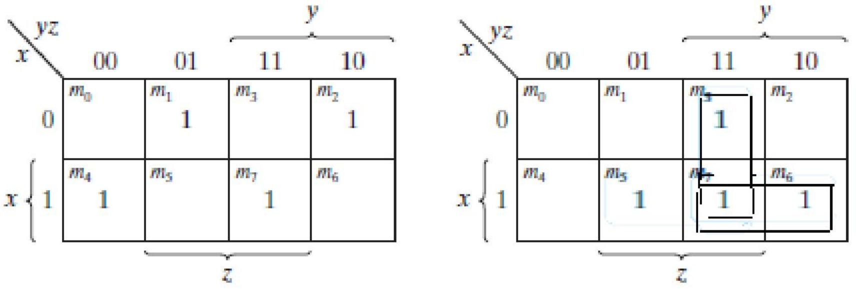
FULL ADDER:-

- A full adder is a combinational circuit that forms the arithmetic sum of three bits.
- It consists of three inputs and two outputs. Two of the input variables, denoted by x and y , represent the two significant bits to be added. The third input, z , represents the carry from the previous lower significant position.

x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Truth Table

- Two outputs are necessary because the arithmetic sum of three binary digits ranges in value from 0 to 3, and binary representation of 2 or 3 needs two bits. The two outputs are designated by the symbols S for sum and C for carry.



(a) $S = x'y'z + x'yz' + xy'z' + xyz$

(b) $C = xy + xz + yz$

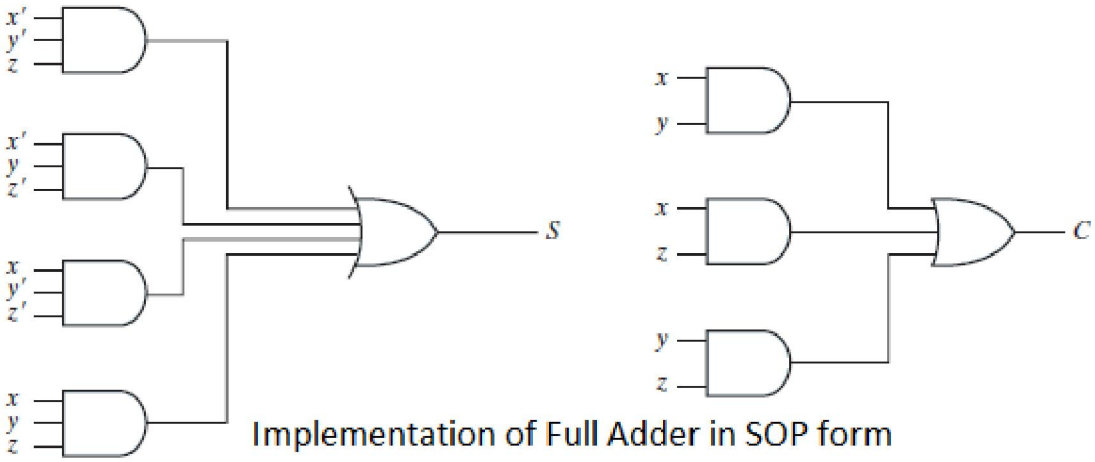
K-Map for full adder

- The binary variable S gives the value of the least significant bit of the sum. The binary variable C gives the output carry formed by adding the input carry and the bits of the words.
- The eight rows under the input variables designate all possible combinations of the three variables. The output variables are determined from the arithmetic sum of the input bits. When all input bits are 0, the output is 0.
- The S output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1. The C output has a carry of 1 if two or three inputs are equal to 1.
- The simplified expressions are

$$S = x'y'z + x'yz' + xy'z' + xyz$$

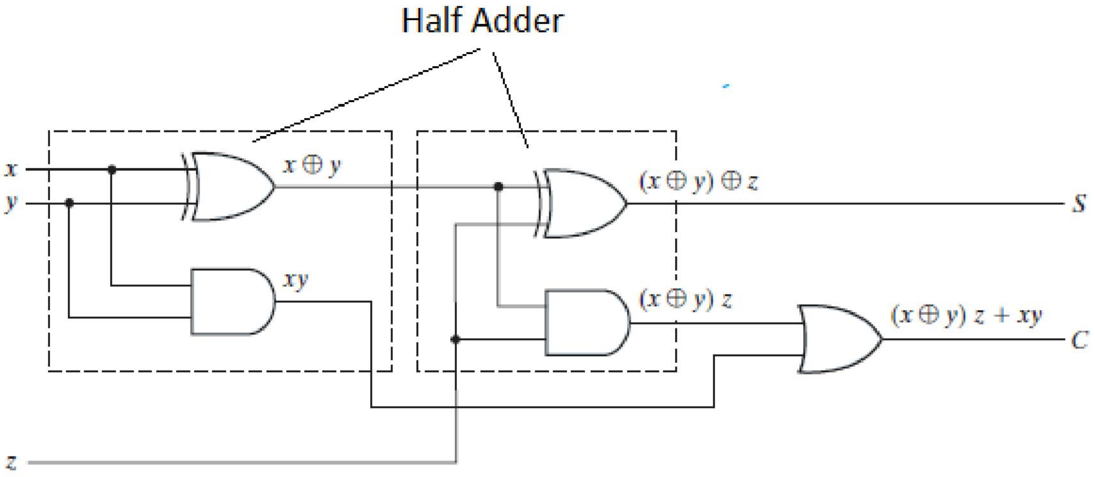
$$C = xy + xz + yz$$

- The logic diagram for the full adder implemented in sum-of-products form is shown in figure.



Implementation of Full Adder in SOP form

- It can also be implemented with two half adders and one OR gate as shown in the figure.

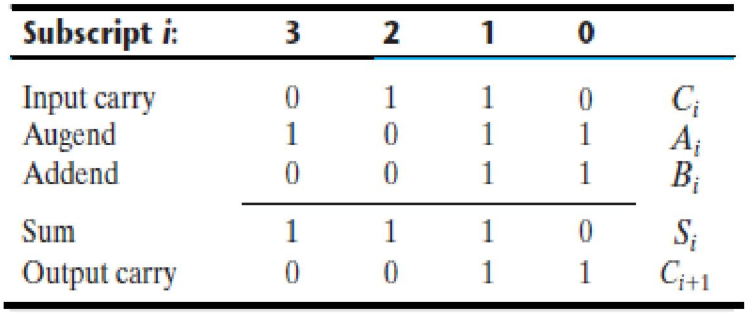


Implementation of Full Adder using Two Half Adders and an OR gate

- A full adder is a combinational circuit that forms the arithmetic sum of three bits.

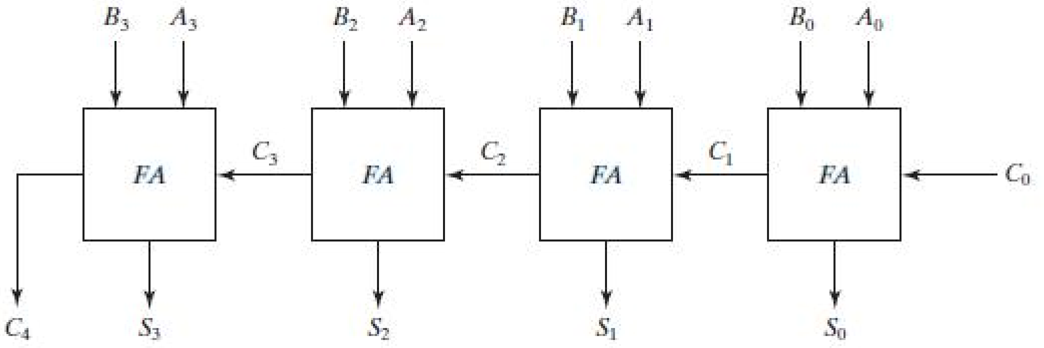
<u>BINARY ADDER:-</u>

- A binary adder is a digital circuit that produces the arithmetic sum of two binary numbers.
- It can be constructed with full adders connected in cascade, with the output carry from each full adder connected to the input carry of the next full adder in the chain.
- Addition of n-bit numbers requires a chain of n full adders or a chain of one-half adder and n-1 full adders. In the former case, the input carry to the least significant position is fixed at 0.
- The interconnection of four full-adder (FA) circuits to provide a four-bit binary ripple carry adder is shown in the figure.
- The augend bits of A and the addend bits of B are designated by subscript numbers from right to left, with subscript 0 denoting the least significant bit.
- The carries are connected in a chain through the full adders. The input carry to the adder is C0, and it ripples through the full adders to the output carry C4. The S outputs generate the required sum bits.
- An n -bit adder requires n full adders, with each output carry connected to the input carry of the next higher order full adder.
- Consider the two binary numbers A = 1011 and B = 0011. Their sum S = 1110 is formed with the four-bit adder as follows:



Subscript *i*:	3	2	1	0	
Input carry	0	1	1	0	C_i
Augend	1	0	1	1	A_i
Addend	0	0	1	1	B_i
Sum	1	1	1	0	S_i
Output carry	0	0	1	1	C_{i+1}

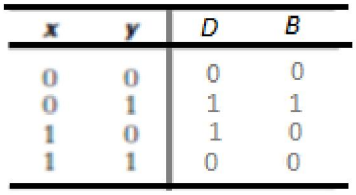
- The bits are added with full adders, starting from the least significant position (subscript 0), to form the sum bit and carry bit. The input carry C_0 in the least significant position must be 0.
- The value of C_{i+1} in a given significant position is the output carry of the full adder. This value is transferred into the input carry of the full adder that adds the bits one higher significant position to the left.
- The sum bits are thus generated starting from the rightmost position and are available as soon as the corresponding previous carry bit is generated. All the carries must be generated for the correct sum bits to appear at the outputs.



Four Bit Binary Adder

HALF SUBTRACTOR:-

- This circuit needs two binary inputs and two binary outputs.
- Symbols x and y are assigned to the two inputs and D (for difference) and B (for borrow) to the outputs.
- The truth table for the half subtractor is listed in the below table.



x	*y*	*D*	*B*
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

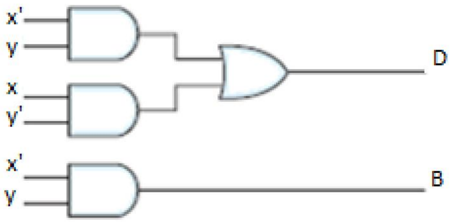
Truth Table

- The B output is 1 only when the inputs are 0 and 1. The D output represents the least significant bit of the subtraction.
- The subtraction operation is done by using the following rules as

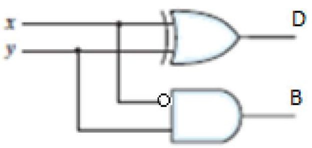
0-0=0;
0-1=1 with borrow 1;
1-0=1;
1-1=0.

- The simplified Boolean functions for the two outputs can be obtained directly from the truth table. The simplified sum-of-products expressions are

$$D = x'y + xy' \text{ and } B = x'y$$



D = x'y+xy'
B = x'y

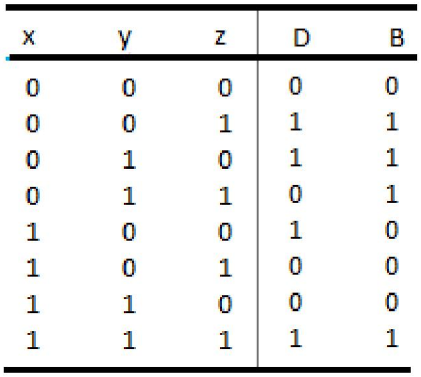


$D = x \oplus y$
B=x'y

- The logic diagram of the half adder implemented in sum of products is shown in the figure. It can be also implemented with an exclusive-OR and an AND gate with one inverted input.

FULL SUBTRACTOR:-

- A full subtractor is a combinational circuit that forms the arithmetic subtraction operation of three bits.
- It consists of three inputs and two outputs. Two of the input variables, denoted by x and y , represent the two significant bits to be subtracted. The third input, z , is subtracted from the result 0f the first subtraction.

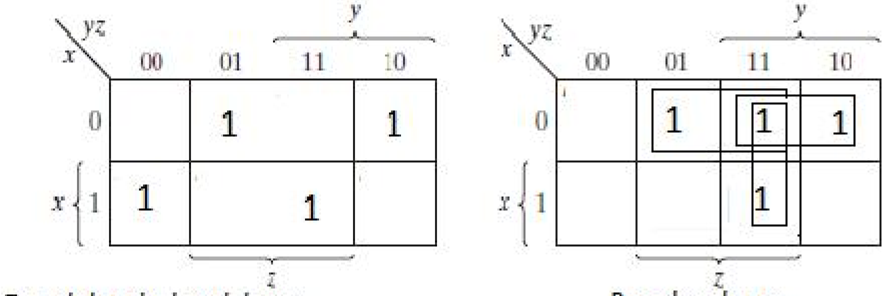


x	y	z	D	B
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Truth Table

- Two outputs are necessary because the arithmetic subtraction of three binary digits ranges in value from 0 to 3, and binary representation of 2 or 3 needs two bits. The two outputs are designated by the symbols D for difference and B for borrow.

- The binary variable D gives the value of the least significant bit of the difference. The binary variable B gives the output borrow formed during the subtraction process.



D = x'y'z+x'yz'+xy'z'+xyz

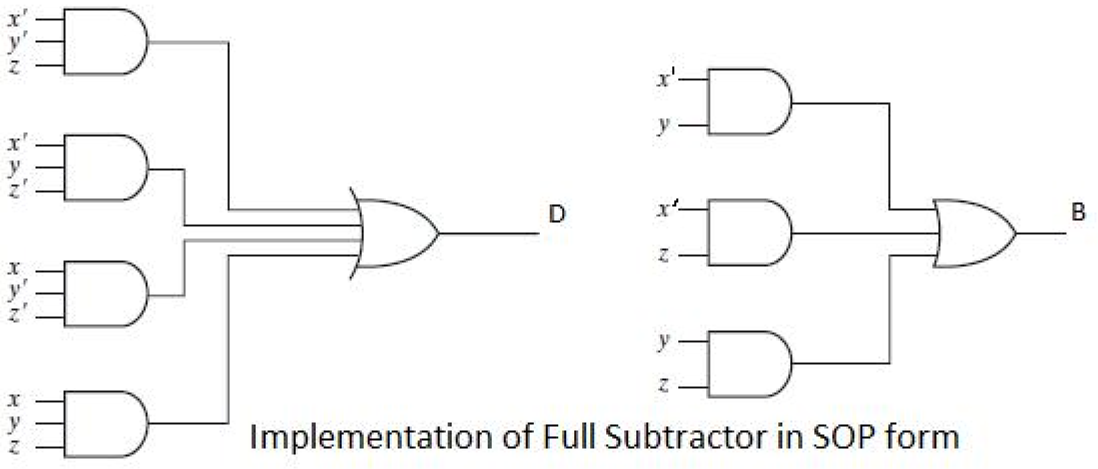
B = x'z+x'y+yz

K-Map for full Subtractor

- The eight rows under the input variables designate all possible combinations of the three variables. The output variables are determined from the arithmetic subtraction of the input bits.
- The difference D becomes 1 when any one of the input is 1or all three inputs are equal to1 and the borrow B is 1 when the input combination is (0 0 1) or (0 1 0) or (0 1 1) or (1 1 1).
- The simplified expressions are

D = x’y’z + x’yz’ + xy’z’ + xyz
B = x’z + x’y + yz

- The logic diagram for the full adder implemented in sum-of-products form is shown in figure.



Implementation of Full Subtractor in SOP form

MAGNITUDE COMPARATOR:-

- A magnitude comparator is a combinational circuit that compares two numbers A and B and determines their relative magnitudes.
- The following description is about a 2-bit magnitude comparator circuit.
- The outcome of the comparison is specified by three binary variables that indicate whether A < B, A = B, or A > B.
- Consider two numbers, A and B, with two digits each. Now writing the coefficients of the numbers in descending order of significance:

$$A = A_1\ A_0$$
$$B = B_1\ B_0$$

- The two numbers are equal if all pairs of significant digits are equal i.e. if and only if A1 = B1, and A0 = B0.
- When the numbers are binary, the digits are either 1 or 0, and the equality of each pair of bits can be expressed logically with an exclusive-NOR function as

$x1=A_1B_1+A_1'B_1'$

And $x0=A_0B_0+A_0'B_0'$

- The equality of the two numbers A and B is displayed in a combinational circuit by an output binary variable that we designate by the symbol (A = B).
- This binary variable is equal to 1 if the input numbers, A and B , are equal, and is equal to 0 otherwise.
- For equality to exist, all xi variables must be equal to 1, a condition that dictates an AND operation of all variables:

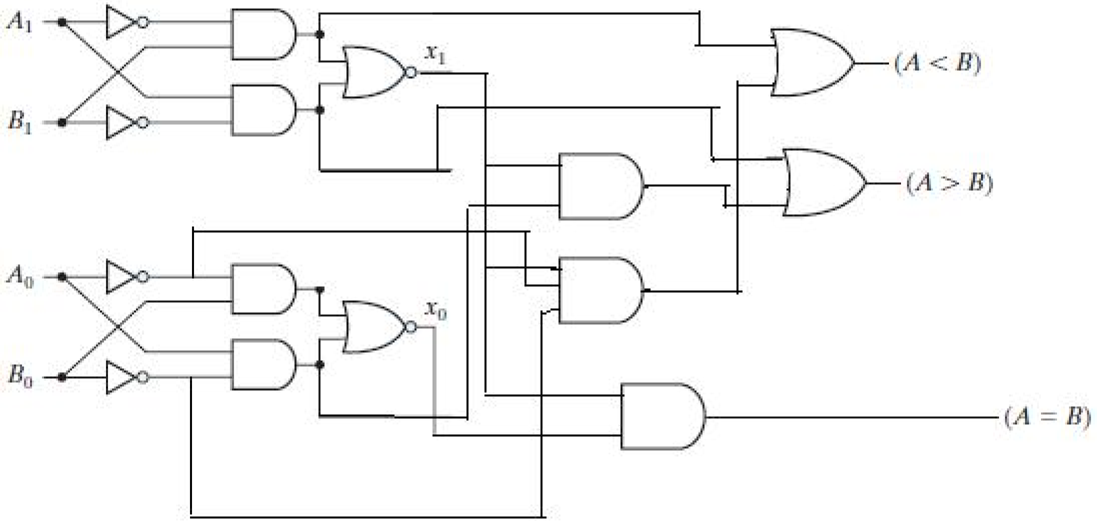
$$(A = B) = x_1x_0$$

- The binary variable (A = B) is equal to 1 only if all pairs of digits of the two numbers are equal.
- To determine whether A is greater or less than B, we inspect the relative magnitudes of pairs of significant digits, starting from the most significant position. If the two digits of a pair are equal, we compare the next lower significant pair of digits. If the corresponding digit of A is 1 and that of B is 0, we conclude that A > B. If the corresponding digit of A is 0 and that of B is 1, we have A < B. The sequential comparison can be expressed logically by the two Boolean functions

$$(A > B) = A_1B_1'+x_1A_0B'_0$$
$$(A < B) = A_1'\ B_1\ +x_1A_0'B_0'$$

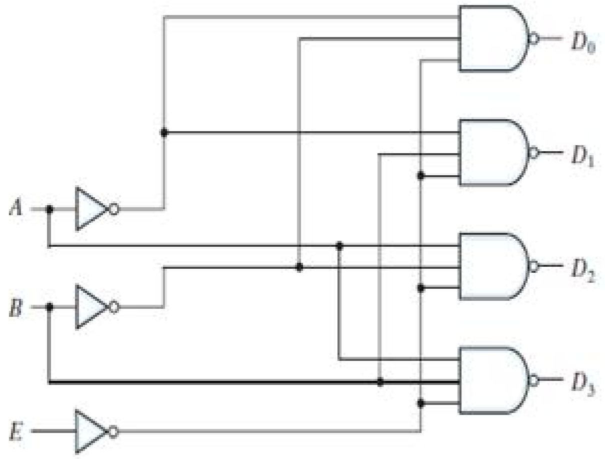
A_1	A_0	B_1	B_0	A>B	A<B	A=B
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	0	0	1
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	0	1

Truth Table



Logic Diagram of 2-bit Magnitude Comparator

DECODER:-

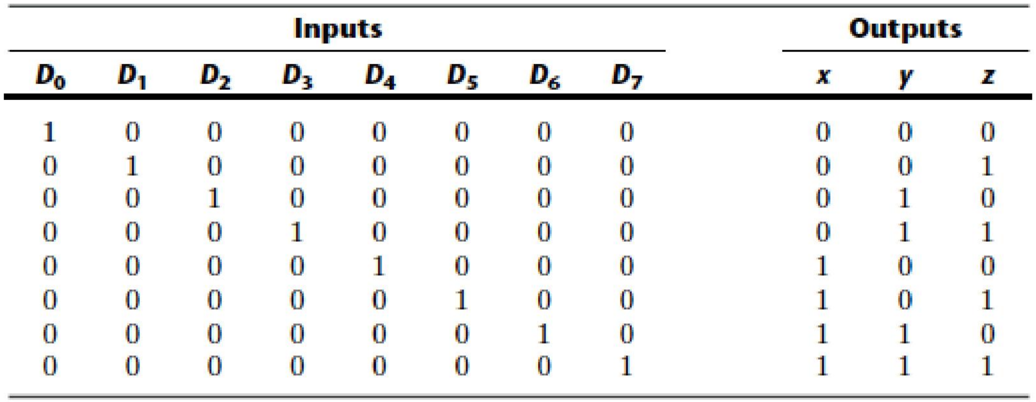


E	A	B	D_0	D_1	D_2	D_3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	1

- A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2^n unique output lines.
- If the n -bit coded information has unused combinations, the decoder may have fewer than 2n outputs.
- The decoders presented here are called n -to- m -line decoders, where m … 2n.
- Their purpose is to generate the 2n (or fewer) minterms of n input variables.
- Each combination of inputs will assert a unique output. The name decoder is also used in conjunction with other code converters, such as a BCD-to-seven-segment decoder.
- Consider the three-to-eight-line decoder circuit of three inputs are decoded into eight outputs, each representing one of the minterms of the three input variables.
- The three inverters provide the complement of the inputs, and each one of the eight AND gates generates one of the minterms.
- The input variables represent a binary number, and the outputs represent the eight digits of a number in the octal number system.
- However, a three-to-eight-line decoder can be used for decoding any three-bit code to provide eight outputs, one for each element of the code.
- A two-to-four-line decoder with an enable input constructed with NAND gates is shown in Fig.
- The circuit operates with complemented outputs and a complement enable input. The decoder is enabled when E is equal to 0 (i.e., active-low enable). As indicated by the truth table, only one output can be equal to 0 at any given time; all other outputs are equal to 1.
- The output whose value is equal to 0 represents the minterm selected by inputs A and B.
- The circuit is disabled when E is equal to 1, regardless of the values of the other two inputs.
- When the circuit is disabled, none of the outputs are equal to 0 and none of the minterms are selected.
- In general, a decoder may operate with complemented or un-complemented outputs.
- The enable input may be activated with a 0 or with a 1 signal.
- Some decoders have two or more enable inputs that must satisfy a given logic condition in order to enable the circuit.
- A decoder with enable input can function as a demultiplexer— a circuit that receives information from a single line and directs it to one of 2n possible output lines.
- The selection of a specific output is controlled by the bit combination of n selection lines.
- The decoder of Fig. can function as a one-to-four-line demultiplexer when E is taken as a data input line and A and B are taken as the selection inputs.
- The single input variable E has a path to all four outputs, but the input information is directed to only one of the output lines, as specified by the binary combination of the two selection lines A and B .
- This feature can be verified from the truth table of the circuit.
- For example, if the selection lines AB = 10, output D_2 will be the same as the input value E, while all other outputs are maintained at 1.
- Since decoder and demultiplexer operations are obtained from the same circuit, a decoder with an enable input is referred to as a decoder – demultiplexer.
- A application of this decoder is binary-to-octal conversion.

ENCODER:-

- An encoder is a digital circuit that performs the inverse operation of a decoder.
- An encoder has 2n (or fewer) input lines and n output lines.
- The output lines, as an aggregate, generate the binary code corresponding to the input value.



Inputs								Outputs		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

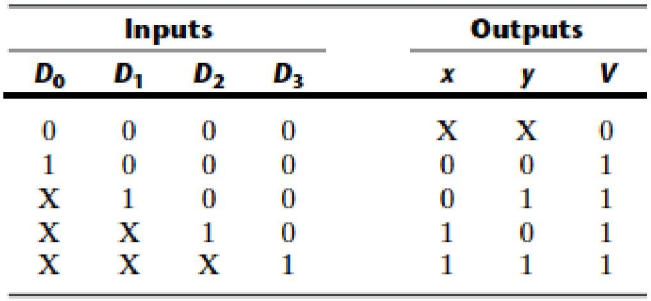
- The above Encoder has eight inputs (one for each of the octal digits) and three outputs that generate the corresponding binary number.
- It is assumed that only one input has a value of 1 at any given time.
- The encoder can be implemented with OR gates whose inputs are determined directly from the truth table.
- Output z is equal to 1 when the input octal digit is 1, 3, 5, or 7.
- Output y is 1 for octal digits 2, 3, 6, or 7, and output x is 1 for digits 4, 5, 6, or 7.
- These conditions can be expressed by the following Boolean output functions:

$$z = D_1 + D_3 + D_5 + D_7$$
$$y = D_2 + D_3 + D_6 + D_7$$
$$x = D_4 + D_5 + D_6 + D_7$$

- The encoder can be implemented with three OR gates.
- The encoder defined above has the limitation that only one input can be active at any given time.
- If two inputs are active simultaneously, the output produces an undefined combination.
- To resolve this ambiguity, encoder circuits must establish an input priority to ensure that only one input is encoded which is done in the Priority Encoder .

PRIORITY ENCODER:-

- A priority encoder is an encoder circuit that includes the priority function.
- The operation of the priority encoder is such that if two or more inputs are equal to 1 at the same time, the input having the highest priority will take precedence.



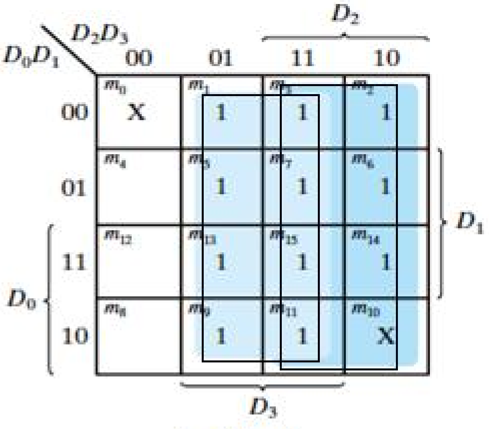
Inputs				Outputs		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

- In addition to the two outputs x and y , the circuit has a third output designated by V ; this is a valid bit indicator that is set to 1 when one or more inputs are equal to 1.

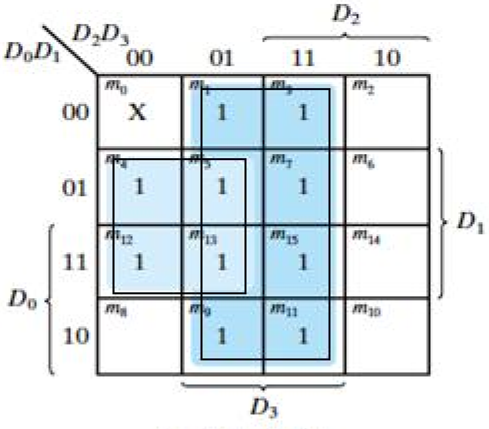
- If all inputs are 0, there is no valid input and V is equal to 0.
- The other two outputs are not inspected when V equals 0 and are specified as don't-care conditions.
- Here X 's in output columns represent don't-care conditions, the X 's in the input columns are useful for representing a truth table in condensed form.

Inputs				**Outputs**		
D_0	D_1	D_2	D_3	x	y	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

- Higher the subscript number, the higher the priority of the input.
- Input D3 has the highest priority, so, regardless of the values of the other inputs, when this input is 1, the output for xy is 11 (binary 3).
- If D2 = 1, provided that D3 = 0, regardless of the values of the other two lower priority inputs the output is 10.
- The output for D1 is generated only if higher priority inputs are 0, and so on down the priority levels.



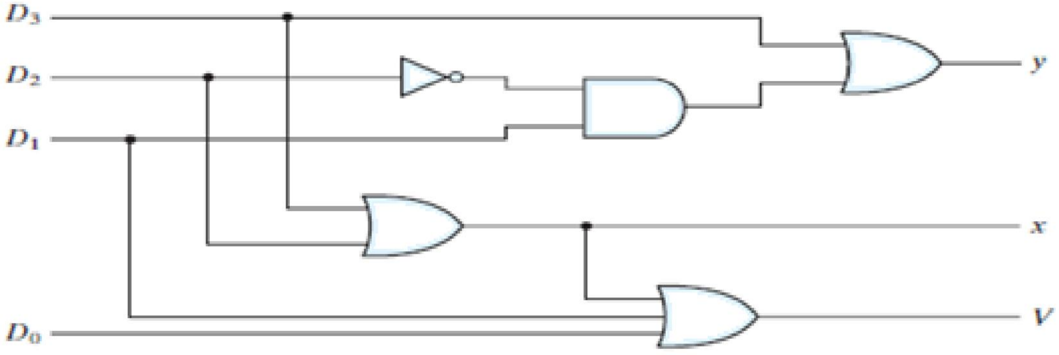
$x = D_2 + D_3$



$y = D_3 + D_1D'_2$

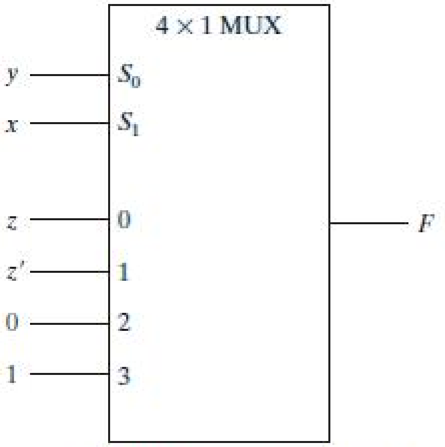
- The maps for simplifying outputs x and y are shown in above Fig.
- The minterms for the two functions are derived from its truth table.
- Although the table has only five rows, when each X in a row is replaced first by 0 and then by 1, we obtain all 16 possible input combinations.
- For example, the fourth row in the table, with inputs XX10, represents the four minterms 0010, 0110, 1010, and 1110. The simplified Boolean expressions for the priority encoder are obtained from the maps.
- The condition for output V is an OR function of all the input variables.
- The priority encoder is implemented according to the following Boolean functions:

$$x = D_2 + D_3$$
$$y = D_3 + D_1 D'_2$$
$$V = D_0 + D_1 + D_2 + D_3$$

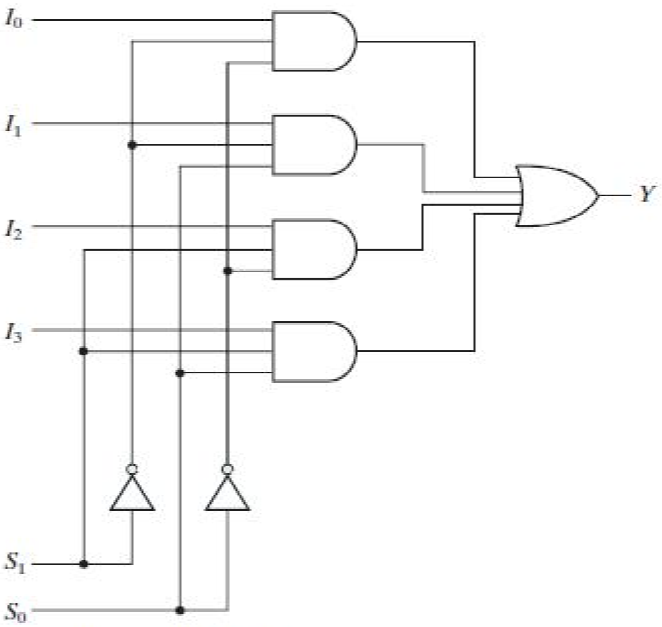


MULTIPLEXER:-

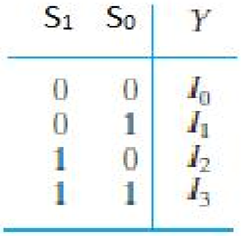
- A multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line.
- The selection of a particular input line is controlled by a set of selection lines.
- Normally, there are 2^n input lines and n selection lines whose bit combinations determine which input is selected.
- A four-to-one-line multiplexer is shown in the below figure. Each of the four inputs, I_0 through I_3, is applied to one input of an AND gate.
- Selection lines S_1 and S_0 are decoded to select a particular AND gate. The outputs of the AND gates are applied to a single OR gate that provides the one-line output.
- The function table lists the input that is passed to the output for each combination of the binary selection values.
- To demonstrate the operation of the circuit, consider the case when
  S_1S_0= 10.
- The AND gate associated with input I_2 has two of its inputs equal to 1 and the third input connected to I_2.
- The other three AND gates have at least one input equal to 0, which makes their outputs equal to 0. The output of the OR gate is now equal to the value of I_2, providing a path from the selected input to the output.
- A multiplexer is also called a data selector, since it selects one of many inputs and steers the binary information to the output line.



(b) Multiplexer implementation



Logic diagram



S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

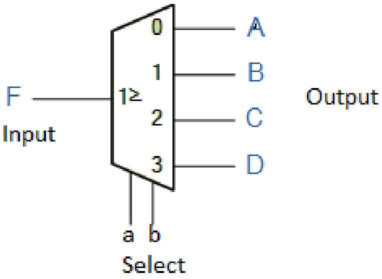
Truth table

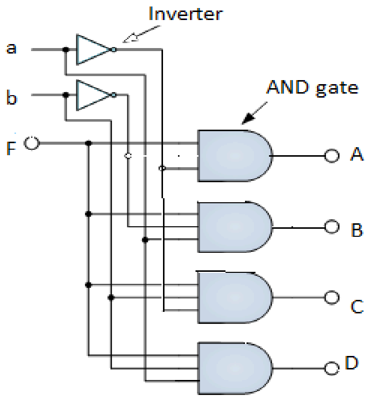
DEMULTIPLEXER:-

- The data distributor, known more commonly as a Demultiplexer or “Demux” for short, is the exact opposite of the Multiplexer.
- The demultiplexer takes one single input data line and then switches it to any one of a number of individual output lines one at a time. The demultiplexer converts a serial data signal at the input to a parallel data at its output lines as shown below.
- The Boolean expression for this 1-to-4 demultiplexer above with outputs A to D and data select lines a, b is given as:

$$F = (ab)'A + a'bB + ab'C + abD$$

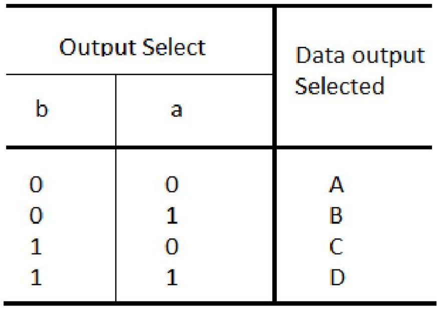
- The function of the demultiplexer is to switch one common data input line to any one of the 4 output data lines A to D in our example above. As with the multiplexer the individual solid state switches are selected by the binary input address code on the output select pins “a” and “b” as shown.





Logic Diagram

- Unlike multiplexers which convert data from a single data line to multiple lines and demultiplexers which convert multiple lines to a single data line, there are devices available which convert data to and from multiple lines and in the next tutorial about combinational logic devices.
- Standard demultiplexer IC packages available are the TTL 74LS138 1 to 8-output demultiplexer, the TTL 74LS139 Dual 1-to-4 output demultiplexer or the CMOS CD4514 1-to-16 output demultiplexer.



Output Select		Data output Selected
b	a	
0	0	A
0	1	B
1	0	C
1	1	D

Truth Table

LOGIC FAMILIES

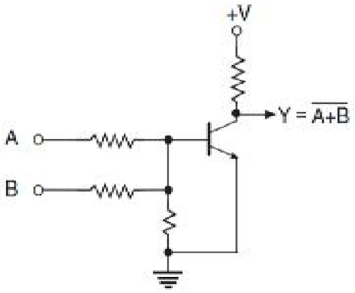
- A circuit configuration or approach used to produce a type of digital integrated circuit is called Logic Family.
- By using logic families we can generate different logic functions, when fabricated in the form of an IC with the same approach, or in other words belonging to the same logic family, will have identical electrical characteristics.
- The set of digital ICs belonging to the same logic family are electrically compatible with each other.
- Some common Characteristics of the Same Logic Family include Supply voltage range, speed of response, power dissipation, input and output logic levels, current sourcing and sinking capability, fan-out, noise margin, etc.
- Choosing digital ICs from the same logic family guarantees that these ICs are compatible with respect to each other and that the system as a whole performs the intended logic function.

TYPES OF LOGIC FAMILY:-

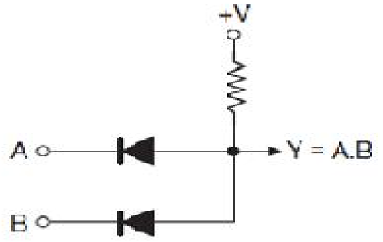
- The entire range of digital ICs is fabricated using either bipolar devices or MOS devices or a combination of the two.
- Bipolar families include:-

  Diode logic (DL)
  Resistor-Transistor logic (RTL)
  Diode-transistor logic (DTL)
  Transistor- Transistor logic (TTL)
  Emitter Coupled Logic (ECL),
  (also known as Current Mode Logic(CML))
  Integrated Injection logic (I2L)

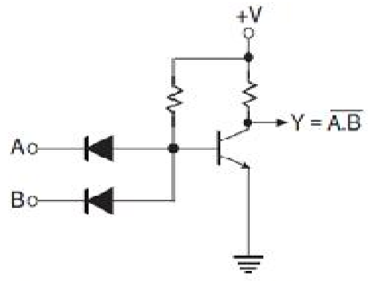
- The Bi-MOS logic family uses both bipolar and MOS devices.



Resistor -Transistor logic (RTL)



Diode Logic (DL)



Diode-Transistor logic (DTL)

- Above are some example of DL, RTL and DTL.
- MOS families include:-

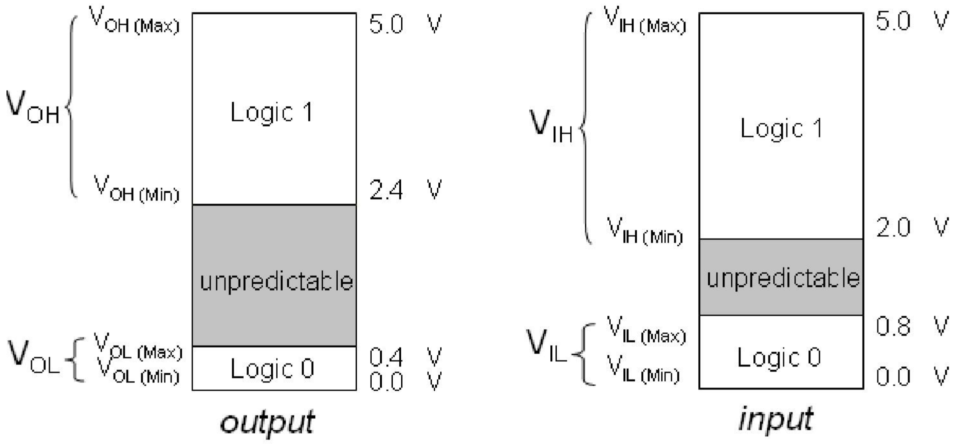
  ThePMOS family (using P-channel MOSFETs)
  The NMOS family (using N-channel MOSFETs)
  The CMOS family (using both N- and P-channel devices)

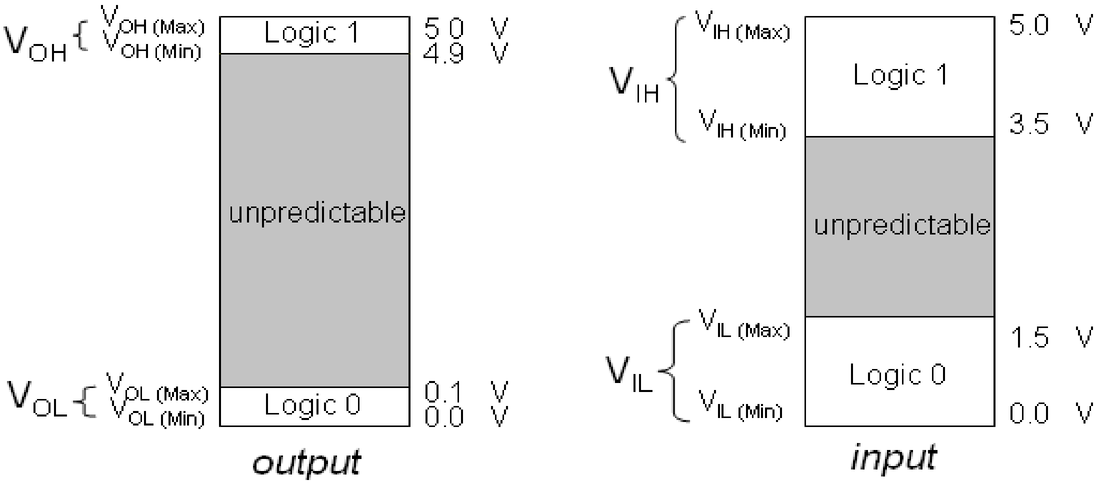
SOME OPERATIONAL PROPERTIES OF LOGIC FAMILY:-

DC Supply Voltage:-

- The nominal value of the dc supply voltage for TTL (transisitor-transistor logic) and CMOS (complementary metal-oxide semiconductor) devices is +5V. Although ommitted from logic diagrams for simplicity, this voltage is connected to Vcc or VDD pin of an IC package and ground is connected to the GND pin.

TTL Logic Levels





CMOS Logic Levels

Noise Immunity:-

- Noise is the unwanted voltage that is induced in electrical circuits and can present a threat to the poor operation of the circuit. In order not to be adversely effected by noise, a logic circuit must have a certain amount of 'noise immunity'.
- This is the ability to tolerate a certain amount of unwanted voltage fluctuation on its inputs without changing its output state is called Noise Immunity.

Noise Margin:-

- A measure of a circuit's noise immunity is called 'noise margin' which is expressed in volts.
- There are two values of noise margin specified for a given logic circuit: the HIGH (V_{NH}) and LOW (V_{NL}) noise margins.

These are defined by following equations :

$V_{NH} = V_{OH}$ (Min) - V_{IH} (Min) $V_{NL} = V_{IL}$ (Max) - V_{OL} (Max)

Power Dissipation:-

- A logic gate draws ICCH current from the supply when the gate is in the HIGH output state, draws ICCL current from the supply in the LOW output state.
- Average power is

  PD = VCC ICC where ICC = (ICCH + ICCL) / 2

Propagation Delay time:-

- When a signal passes (propagates) through a logic circuit, it always experiences a time delay as shown below. A change in the output level always occurs a short time, called 'propagation delay time' , later than the change in the input level that caused it.

Fan Out of Gates:-

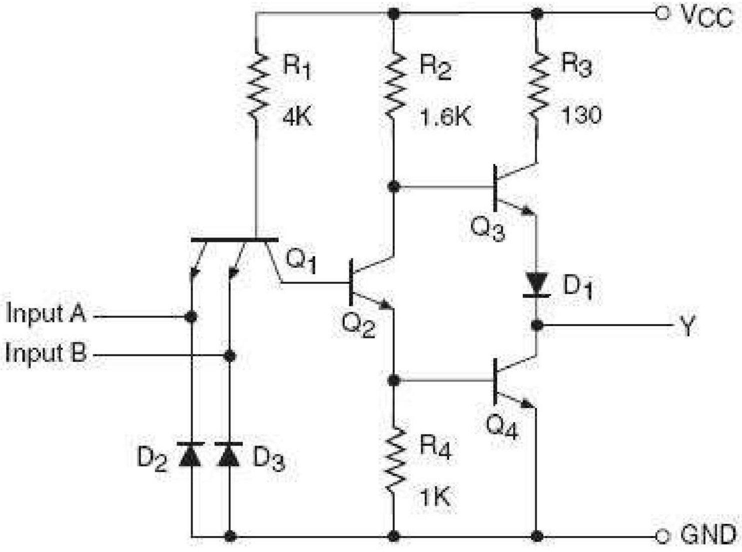
- When the output of a logic gate is connected to one or more inputs of other gates, a load on the driving gate is created. There is a limit to the number of load gates that a given gate can drive. This limit is called the 'Fan-Out' of the gate.

TRANSISTOR-TRANSISTOR LOGIC:-

- In Transistor-Transistor logic or just TTL, logic gates are built only around transistors.
- TTL was developed in 1965. Through the years basic TTL has been improved to meet performance requirements. There are many versions or families of TTL.
- For example
  - Standard TTL
  - High Speed TTL (twice as fast, twice as much power)
  - Low Power TTL (1/10 the speed, 1/10 the power of “standard" TTL)
  - Schhottky TTL etc. (for high-frequency uses)
- All TTL logic families have three configurations for outputs
  1. Totem pole output
  2. Open collector output
  3. Tristate output

Totem pole output:-

- Addition of an active pull up circuit in the output of a gate is called totem pole.
- To increase the switching speed of the gate which is limited due to the parasitic capacitance at the output totem pole is used.
- The circuit of a totem-pole NAND gate is shown below, which has got three stages
  1. Input Stage
  2. Phase Splitter Stage
  3. Output Stage



- Transistor Q1 is a two-emitter NPN transistor, which is equivalent two NPN transistors with their base and emitter terminals tied together.
- The two emitters are the two inputs of the NAND gate

  In TTL technology multiple emitter transistors are used for the input devices

  Diodes D2 and D3 are protection diodes used to limit negative input voltages.
- When there is large negative voltage at input, the diode conducts and shorting it to the ground Q2 provides complementary voltages for the output transistors Q3 and Q4.
- The combination of Q3 and Q4 forms the output circuit often referred to as a totem pole arrangement (Q4 is stacked on top of Q3). In such an arrangement, either Q3 or Q4 conducts at a time depending upon the logic status of the inputs

  Diode D1 ensures that Q4 will turn off when Q2 is on (HIGH input)

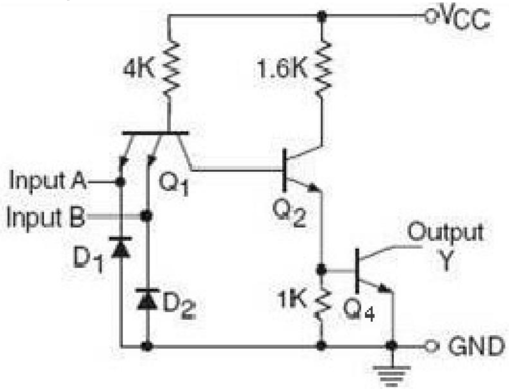
  The output Y is taken from the top of Q3

Advantages of Totem Pole Output:-

- The features of this arrangement are
  1. Low power consumption
  2. Fast switching
  3. Low output impedance

OPEN COLLECTOR OUTPUT:-

- Figure below shows the circuit of a typical TTL gate with open-collector output Observe here that the circuit elements associated with Q3 in the totem-pole circuit are missing and the collector of Q4 is left open-circuited, hence the name open-collector.



- An open-collector output can present a logic LOW output. Since there is no internal path from the output Y to the supply voltage V_{CC} , the circuit cannot present a logic HIGH on its own.

Advantages of Open Collector Outputs:-

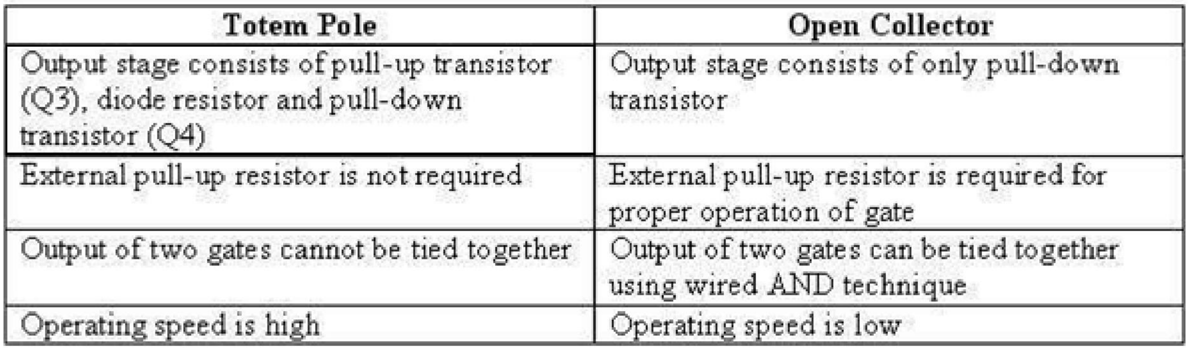
- Open-collector outputs can be tied directly together which results in the logical ANDing of the outputs. Thus the equivalent of an AND gate can be formed by simply connecting the outputs.
- Increased current levels - Standard TTL gates with totem-pole outputs can only provide a HIGH current output of 0.4 mA and a LOW current of 1.6 mA. Many open-collector gates have increased current ratings.
- Different voltage levels - A wide variety of output HIGH voltages can be achieved using open-collector gates. This is useful in interfacing different logic families that have different voltage and current level requirements.

Disadvantage of open-collector gates:-

- They have slow switching speed. This is because the value of pull-up resistor is in kW, which results in a relatively long time Constants

Comparison of Totem Pole and Open Collector Output:-

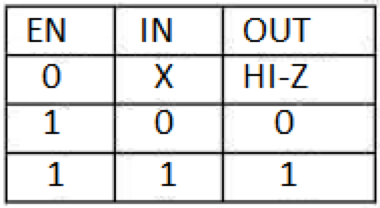
- The major advantage of using a totem-pole connection is that it offers low-output impedance in both the HIGH and LOW output states



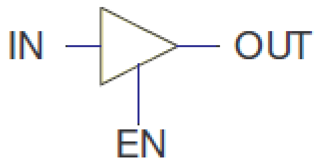
Totem Pole	Open Collector
Output stage consists of pull-up transistor (Q3), diode resistor and pull-down transistor (Q4)	Output stage consists of only pull-down transistor
External pull-up resistor is not required	External pull-up resistor is required for proper operation of gate
Output of two gates cannot be tied together	Output of two gates can be tied together using wired AND technique
Operating speed is high	Operating speed is low

TRISTATE (THREE-STATE) LOGIC OUPUT:-

- Tristate output combines the advantages of the totem-pole and open collector circuits.
- Three output states are HIGH, LOW, and high impedance (Hi-Z).

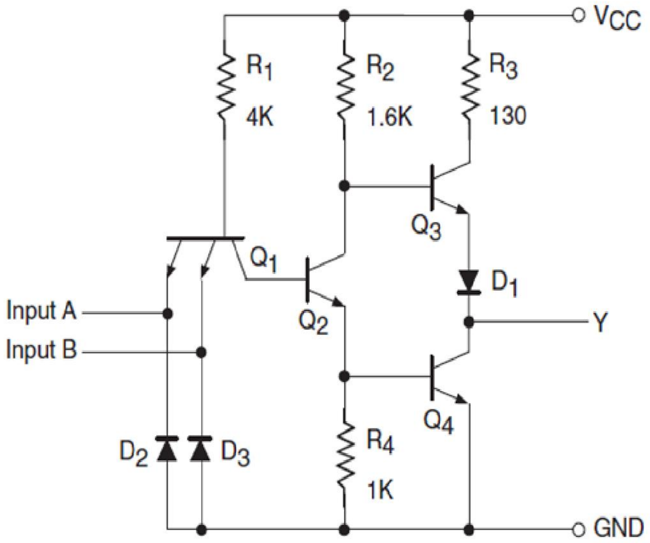


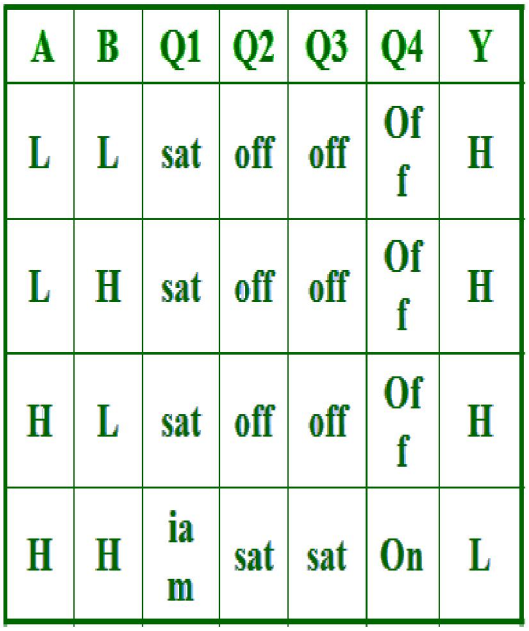
EN	IN	OUT
0	X	HI-Z
1	0	0
1	1	1



- For the symbol and truth table, IN is the data input, and EN, the additional enable input for control. For EN = 0, regardless of the value on IN(denoted by X), the output value is Hi-Z. For EN = 1, the output value follows the input value.
- Data input, IN, can be inverted. Control input, EN, can be inverted by addition of "bubbles" to signals IN OUT EN.
- This requires two inputs: input and enable EN is to make output Hi-Z or follow input.

STANDARD TTL NAND GATE:





A	B	Q1	Q2	Q3	Q4	Y
L	L	sat	off	off	Of f	H
L	H	sat	off	off	Of f	H
H	L	sat	off	off	Of f	H
H	H	ia m	sat	sat	On	L

CMOS TECHNOLOGY:-

- MOS stands for Metal Oxide Semiconductor and this technology uses FETs.
- MOS can be classified into three sub-families:

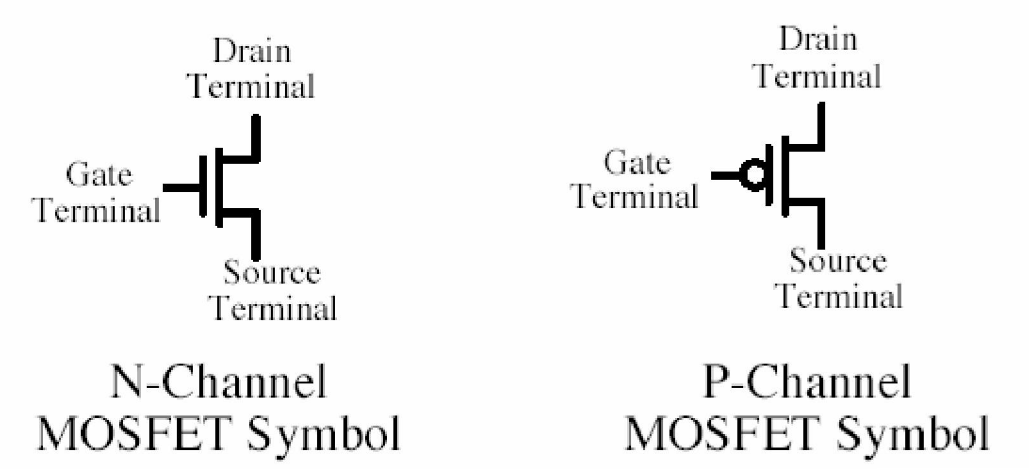
  PMOS (P-channel)
  NMOS (N-channel)
  CMOS (Complementary MOS, most common)
- The following simplified symbols are used to represent MOSFET transistors in most CMOS. The gate of a MOS transistor controls the flow of the current between the drain and the source. The MOS transistor can be viewed as a simple ON/OFF switch.

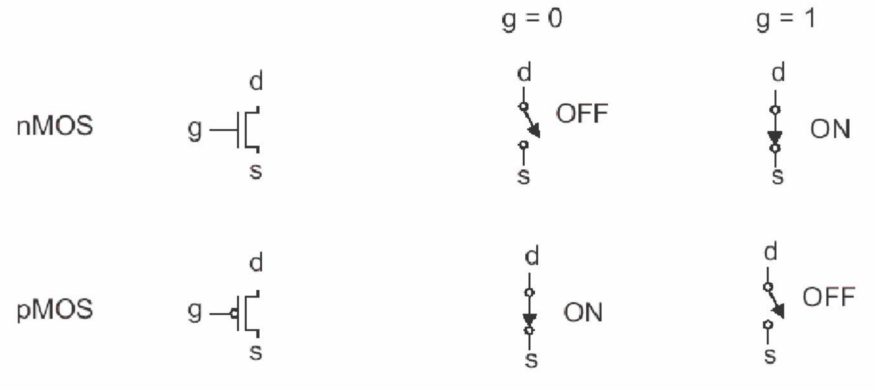
Advantages of MOS Digital ICs:-

- They are simple and inexpensive to fabricate.
- Can be used for Higher integration and consume little power.

Disadvantages of MOS Digital ICs:-

- There is possibility for Static-electricity damage.
- They are slower than TTL.



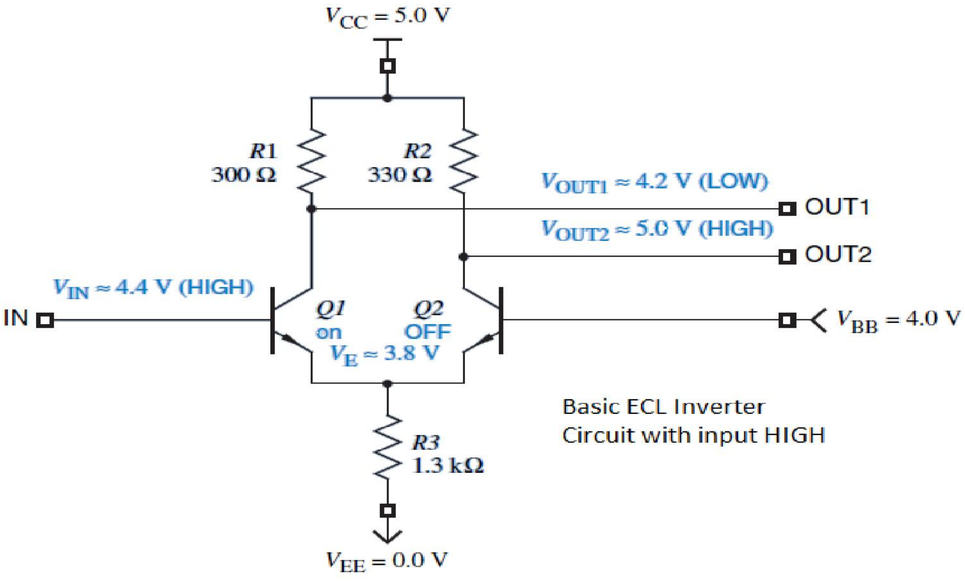


ECL: EMITTER-COUPLED LOGIC:-

- The key to reduce propagation delay in a bipolar logic family is to prevent a gate's transistors from saturating. It is possible to prevent saturation by using a radically different circuit structure, called current-mode logic (CML) or emitter-coupled logic (ECL).
- Unlike the other logic families in this chapter, ECL does not produce a large voltage swing between the LOW and HIGH levels but it has a small voltage swing, less than a volt, and it internally switches current between two possible paths, depending on the output state.

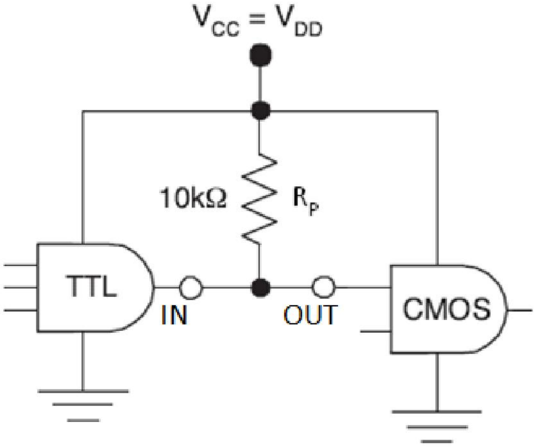
Basic ECL Circuit

- The basic idea of current-mode logic is illustrated by the inverter/buffer circuit in the figure. This circuit has both an inverting output (OUT1) and a non-inverting output (OUT2).
- Two transistors are connected as a differential amplifier with a common emitter resistor.
- The supply voltages for this example are VCC = 5.0, VBB = 4.0, and VEE = 0 V, and the input LOW and HIGH levels are defined to be 3.6 and 4.4 V. This circuit actually produces output LOW and HIGH levels that are 0.6 V higher (4.2 and 5.0 V).

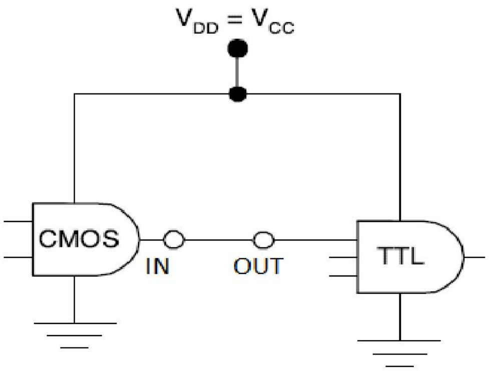


Basic ECL Inverter Circuit with input HIGH

INTERFACING OF TTL TO CMOS



INTERFACING OF CMOS TO TTL



TTL vs. CMOS:-

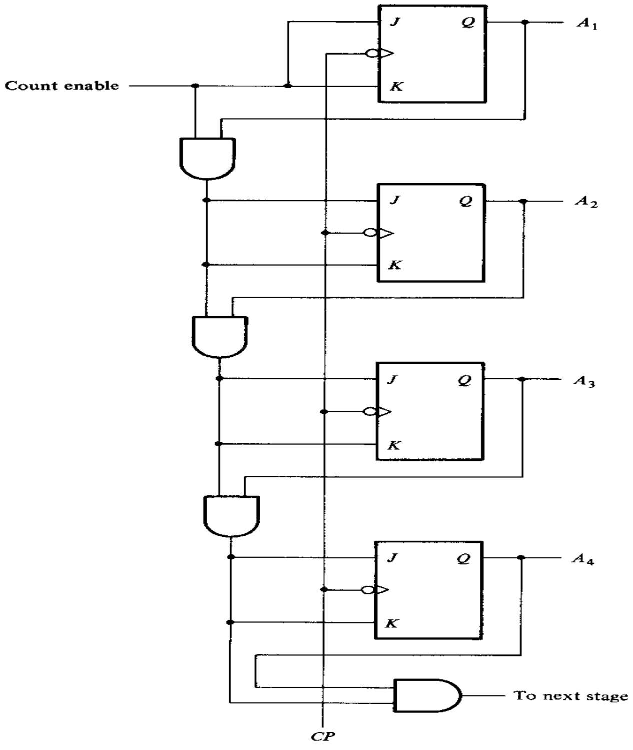
- TTL has less propagation delay than CMOS i.e. TTL is good where high speed is needed.
- And CMOS 4000 is good for Battery equipment and where speed is not so important.
- CMOS requires less power than TTL i.e. power dissipation and hence power consumption is less for CMOS.

COUNTER

- A counter is a device which stores (and sometimes displays) the number of times a particular event or process has occurred. In electronics, counters can be implemented quite easily using register-type circuits.
- There are different types of counters, viz.
    - Asynchronous (ripple) counter
    - Synchronous counter
    - Decade counter
    - Up/down counter
    - Ring counter
    - Johnson counter
    - Cascaded counter
    - Modulus counter.

Synchronous counter

- A 4-bit synchronous counter using JK flip-flops is shown in the figure.
- In synchronous counters, the clock inputs of all the flip-flops are connected together and are triggered by the input pulses. Thus, all the flip-flops change state simultaneously (in parallel).

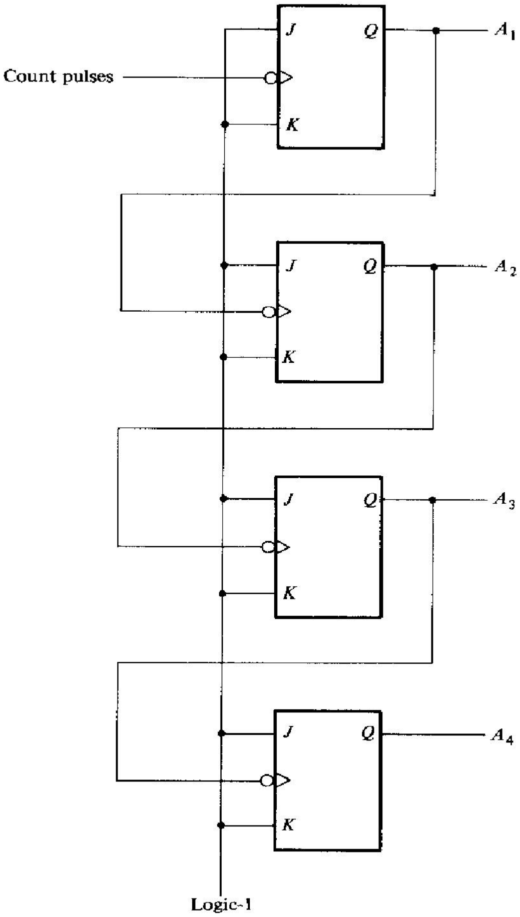


- The circuit below is a 4-bit synchronous counter.
- The J and K inputs of FF0 are connected to HIGH. FF1 has its J and K inputs connected to the output of FF0, and the J and K inputs of FF2 are connected to the output of an AND gate that is fed by the outputs of FF0 and FF1.
- A simple way of implementing the logic for each bit of an ascending counter (which is what is depicted in the image to the right) is for each bit to toggle when all of the less significant bits are at a logic high state.
- For example, bit 1 toggles when bit 0 is logic high; bit 2 toggles when both bit 1 and bit 0 are logic high; bit 3 toggles when bit 2, bit 1 and bit 0 are all high; and so on.

- Synchronous counters can also be implemented with hardware finite state machines, which are more complex but allow for smoother, more stable transitions.

Asynchronous Counter

- An asynchronous (ripple) counter is a single d-type flip-flop, with its J (data) input fed from its own inverted output.
- This circuit can store one bit, and hence can count from zero to one before it overflows (starts over from 0).



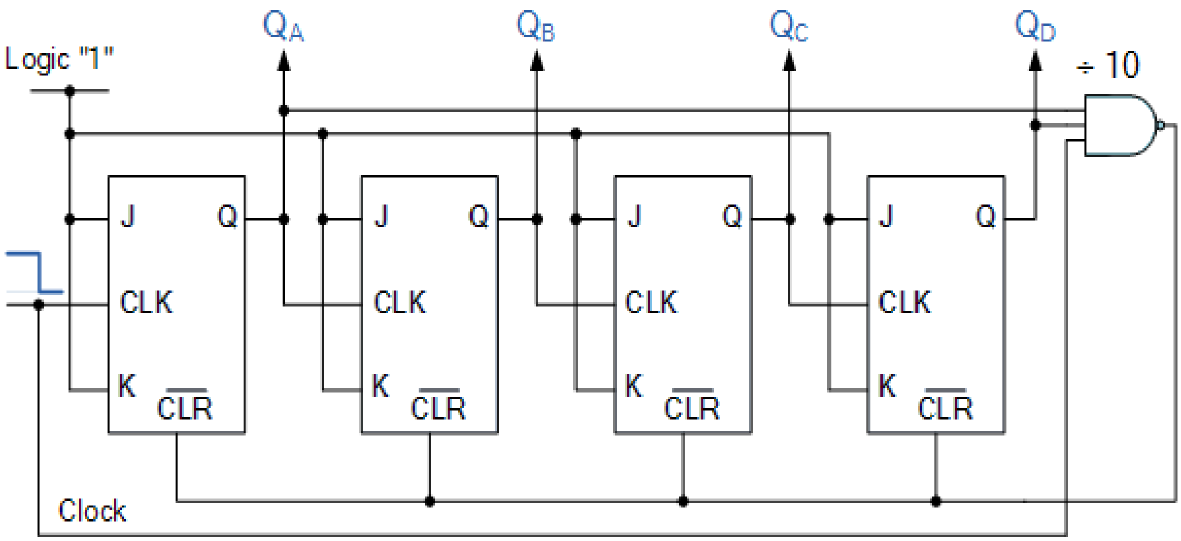
- This counter will increment once for every clock cycle and takes two clock cycles to overflow, so every cycle it will alternate between a transition from 0 to 1 and a transition from 1 to 0.
- This creates a new clock with a 50% duty cycle at exactly half the frequency of the input clock.
- If this output is then used as the clock signal for a similarly arranged D flip-flop, remembering to invert the output to the input, one will get another 1 bit counter that counts half as fast. These together yield a two-bit counter.
- Additional flip-flops can be added, by always inverting the output to its own input, and using the output from the previous flip-flop as the clock signal. The result is called a ripple counter, which can count to 2^n – 1, where n is the number of bits (flip-flop stages) in the counter.
- Ripple counters suffer from unstable outputs as the overflows "ripple" from stage to stage, but they find application as dividers for clock signals.

Modulus Counter

- A modulus counter is that which produces an output pulse after a certain number of input pulses is applied.
- In modulus counter the total count possible is based on the number of stages, i.e., digit positions.

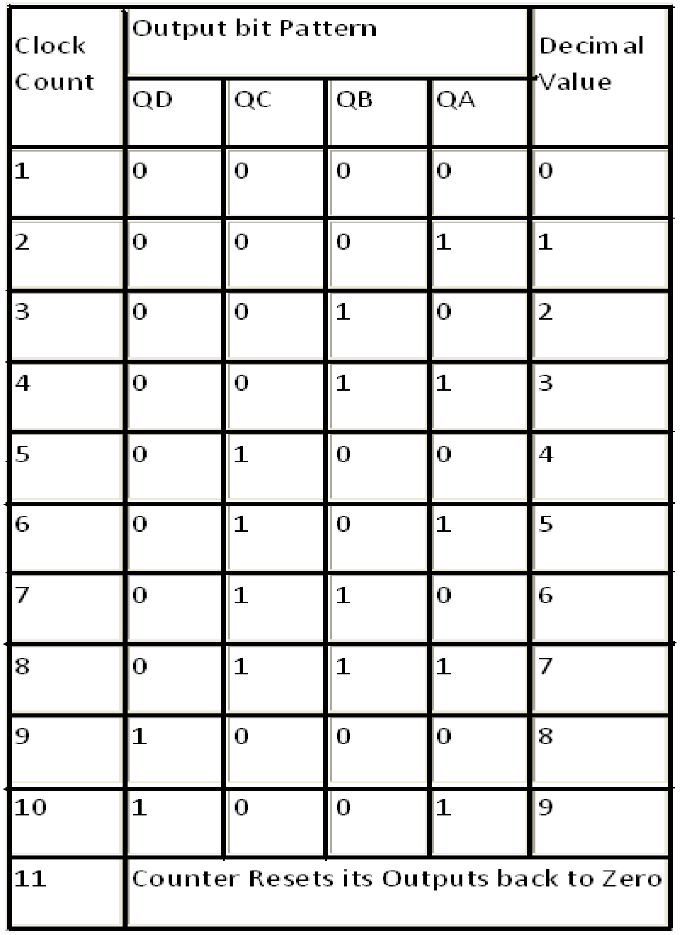
- Modulus counters are used in digital computers.
- A binary modulo-8 counter with three flip-flops, i.e., three stages, will produce an output pulse, i.e., display an output one-digit, after eight input pulses have been counted, i.e., entered or applied. This assumes that the counter started in the zero-condition.

Asynchronous Decade Counter



- A decade counter can count from BCD “0” to BCD “9”.
- A decade counter requires resetting to zero when the output count reaches the decimal value of 10, ie. when DCBA = 1010 and this condition is fed back to the reset input.
- A counter with a count sequence from binary “0000” (BCD = “0”) through to “1001” (BCD = “9”) is generally referred to as a BCD binary-coded-decimal counter because its ten state sequence is that of a BCD code but binary decade counters are more common.
- This type of asynchronous counter counts upwards on each leading edge of the input clock signal starting from 0000 until it reaches an output 1001 (decimal 9).
- Both outputs Q_A and Q_D are now equal to logic “1” and the output from the NAND gate changes state from logic “1” to a logic “0” level and whose output is also connected to the CLEAR (CLR) inputs of all the J-K Flip-flops.
- This signal causes all of the Q outputs to be reset back to binary 0000 on the count of 10. Once QA and QD are both equal to logic “0” the output of the NAND gate returns back to a logic level “1” and the counter restarts again from 0000. We now have a decade or Modulo-10 counter.

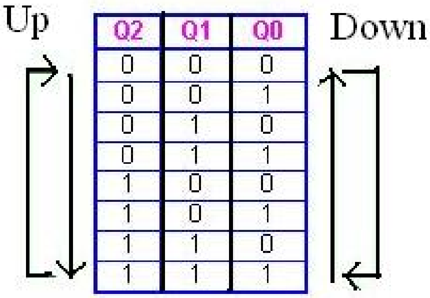
Decade Counter Truth Table



Clock Count	Output bit Pattern				Decimal Value
	QD	QC	QB	QA	
1	0	0	0	0	0
2	0	0	0	1	1
3	0	0	1	0	2
4	0	0	1	1	3
5	0	1	0	0	4
6	0	1	0	1	5
7	0	1	1	0	6
8	0	1	1	1	7
9	1	0	0	0	8
10	1	0	0	1	9
11	Counter Resets its Outputs back to Zero				

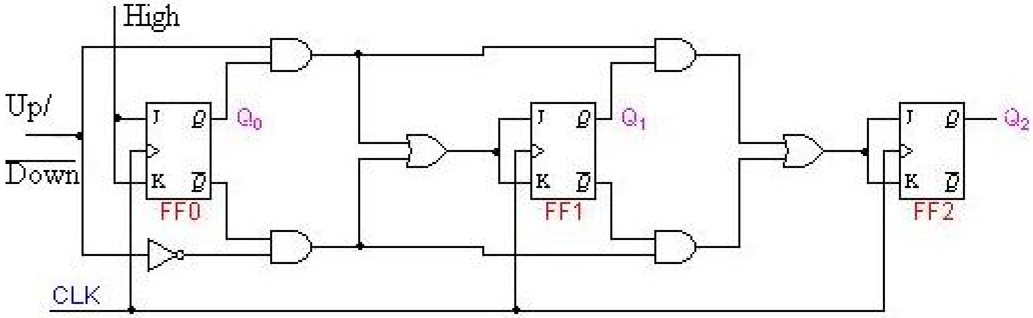
Up/Down Counter

- In a synchronous up-down binary counter the flip-flop in the lowest-order position is complemented with every pulse.
- A flip-flop in any other position is complemented with a pulse, provided all the lower-order pulse equal to 0.
- Up/Down counter is used to control the direction of the counter through a certain sequence.



Q2	Q1	Q0
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

- From the sequence table we can observe that:
    - For both the UP and DOWN sequences, Q_0 toggles on each clock pulse.
    - For the UP sequence, Q_1 changes state on the next clock pulse when Q_0=1.
    - For the DOWN sequence, Q_1 changes state on the next clock pulse when Q_0=0.
    - For the UP sequence, Q_2 changes state on the next clock pulse when Q_0=Q_1=1.
    - For the DOWN sequence, Q_2 changes state on the next clock pulse when Q_0=Q_1=0.



- These characteristics are implemented with the AND, OR & NOT logic connected as shown in the logic diagram above**.**

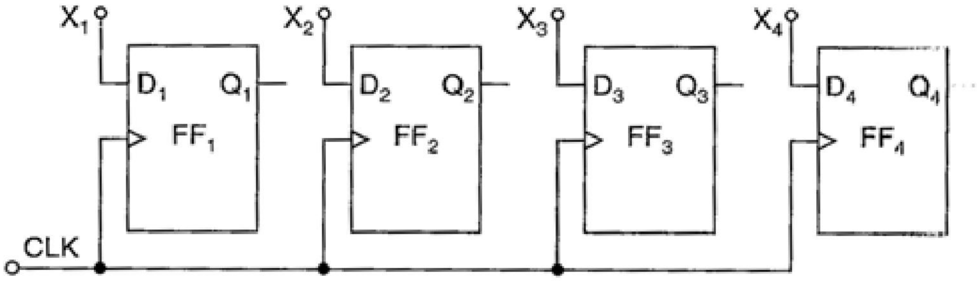
REGISTERS

INTRODUCTION:-

- The sequential circuits known as register are very important logical block in most of the digital systems.
- Registers are used for storage and transfer of binary information in a digital system.
- A register is mostly used for the purpose of storing and shifting binary data entered into it from an external source and has no characteristics internal sequence of states.
- The storage capacity of a register is defined as the number of bits of digital data, it can store or retain.
- These registers are normally used for temporary storage of data.

BUFFER REGISTER:-

- These are the simplest registers and are used for simply storing a binary word.
- These may be controlled by Controlled Buffer Register.
- D flip – flops are used for constructing a buffer register or other flip- flop can be used.
- The figure shown below is a 4- bit buffer register.



Logic diagram of a 4-bit buffer register.

- The binary word to be stored is applied to the data terminals.
- When the clock pulse is applied, the output word becomes the same as the word applied at the input terminals, i.e. the input word is loaded into the register by the application of clock pulse.
- When the positive clock edge arrives, the stored word becomes:

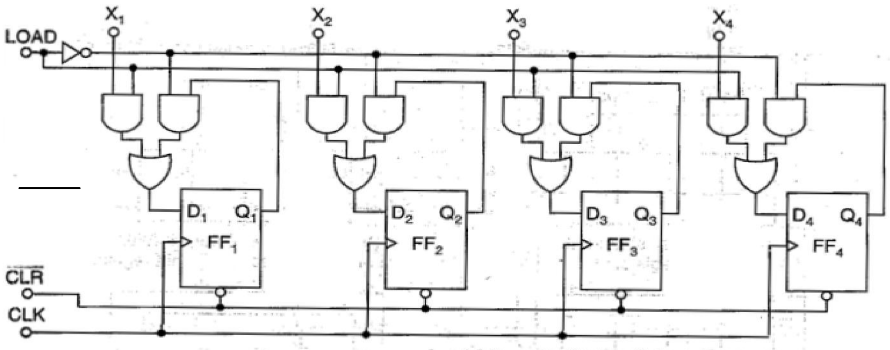
  Q4 Q3 Q2 Q1= X4 X3 X2 X1

  or Q = X .

  This circuit is too primitive to be of any use.

CONTROLLED BUFFER REGISTER:-

- The figure shows a controlled buffer register.



4-bit controlled buffer register.

- If $\overline{\text{CLR}}$ goes LOW, all the flip-flops are RESET and the output becomes, Q = 0000.
- When $\overline{\text{CLR}}$ is HIGH, the register is ready for action

- LOAD is control input.
- When LOAD is HIGH, the data bits X can reach the D inputs of FFs.
- At the positive going edge of the next clock pulse, the register is loaded, i.e.

  Q4 Q3 Q2 Q1= X4 X3 X2 X1

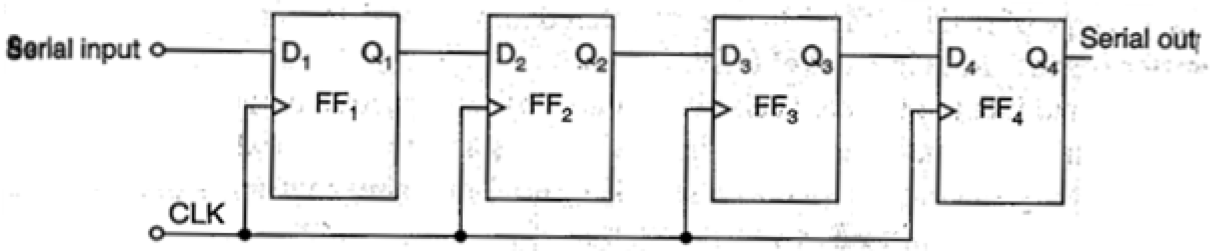
  or Q = X .
- When LOAD is LOW, the X bits cannot reach the FFs. At the same time the inverted signal $\overline{\text{LOAD}}$ is HIGH. This forces each flip-flop output to feedback to its data input.
- Therefore data is circulated or retained as each clock pulse arrives.
- In other words the content register remains unchanged in spite of the clock pulses.
- Longer buffer registers can built by adding more FFs.

CONTROLLED BUFFER REGISTER:-

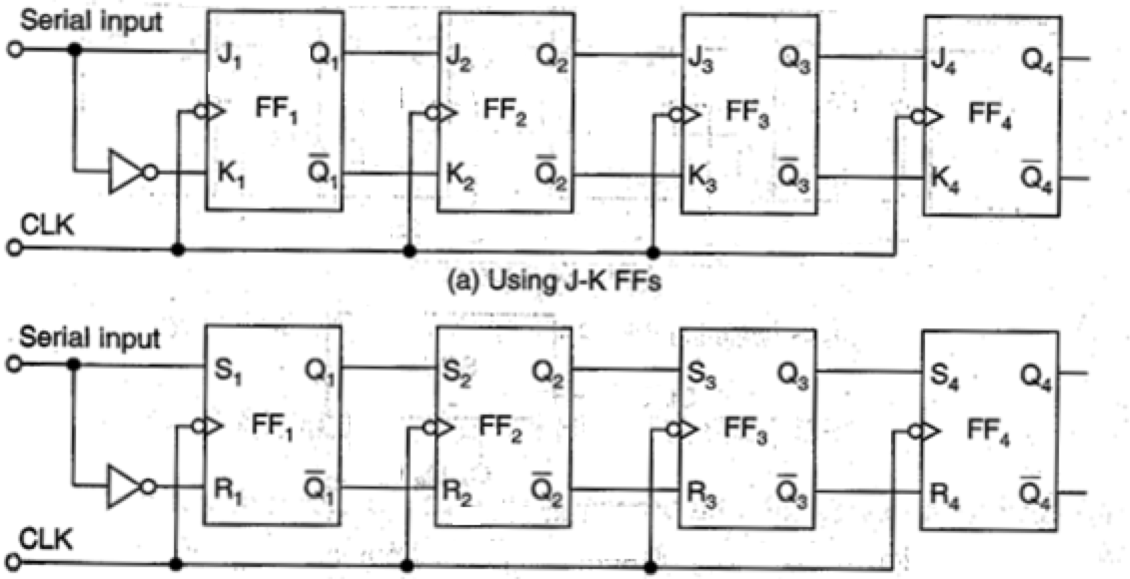
- A number of FFs connected together such that data may be shifted into and shifted out of them is called a shift register.
- Data may be shifted into or out of the register either in serial form or in parallel form.
- There are four basic types of shift registers
  1. Serial in, serial out
  2. Serial in, parallel out
  3. Parallel in, serial out
  4. Parallel in , parallel out

SERIAL IN, SERIAL OUT SHIFT REGISTER:-

- This type of shift register accepts data serially, i.e., one bit at a time and also outputs data serially.
- The logic diagram of a four bit serial in, serial out shift register is shown in below figure:
- In 4 stages i.e. with 4 FFs, the register can store upto 4 bits of data.
- Serial data is applied at the D input of the first FF. The Q output of the first FF is connected to the D input of the second FF, the output of the second FF is connected to the D input of the third FF and the Q output of the third FF is connected to the D input of the fourth FF. The data is outputted from the Q terminal of the last FF.
- When a serial data is transferred to a register, each new bit is clocked into the first FF at the positive going edge of each clock pulse.
- The bit that is previously stored by the first FF is transferred to the second FF.
- The bit that is stored by the second FF is transferred to the third FF, and so on.
- The bit that was stored by the last FF is shifted out.
- A shift register can also be constructed using J-K FFs or S-R FFs as shown in the figure below.



4-bit serial-in, serial-out, shift-right, shift register.



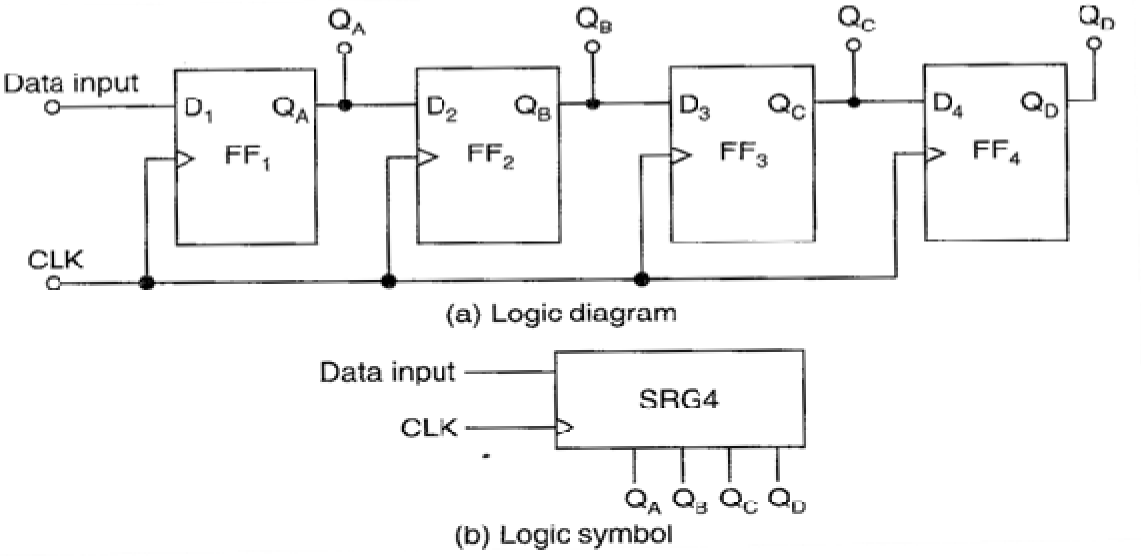
(a) Using J-K FFs

(b) Using S-R FFs

A 4-bit serial-in, serial-out, shift register.

SERIAL IN, PARALLEL OUT SHIFT REGISTER:-

- In this type of register, the data bits are entered into the register serially, but the data stored in the register serially, but the stored in the register is shifted out in the parallel form.
- When the data bits are stored once, each bits appears on its respective output line and all bits are available simultaneously, rather than bit – by – bit basis as in the serial output.
- The serial in, parallel out shift register can be used as a serial in, serial out shift register if the output is taken from the Q terminal of the last FF.
- The logic diagram and logic symbol of a 4 bit serial in, parallel out shift register is given below.



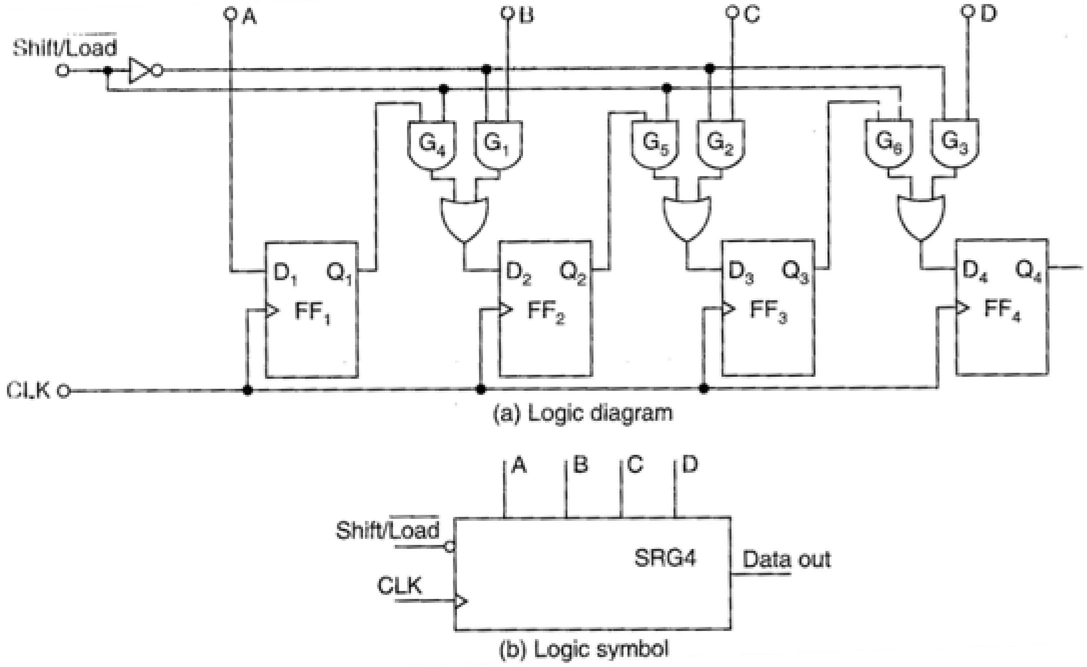
(a) Logic diagram

(b) Logic symbol

A 4- bit serial in, parallel out shift register

PARALLEL IN, SERIAL OUT SHIFT REGISTER:-

- For parallel in, serial out shift register the data bits are entered simultaneously into their respective stages on parallel lines, rather than on bit by bit basis on one line as with serial data inputs, but the data bits are transferred out of the register serially, i.e., on a bit by bit basis over a single line.
- The logic diagram and logic symbol of 4 bit parallel in, serial out shift register using D FFs is shown below.
- There are four data lines A, B, C and D through which the data is entered into the register in parallel form.
- The signal Shift /$\overline{\text{LOAD}}$ allows
    1. The data to be entered in parallel form into the register and
    2. The data to be shifted out serially from terminal Q_4.
- When Shift /$\overline{\text{LOAD}}$ line is HIGH, gates G1, G2, and G3 are disabled, but gates G4, G5 and G6 are enabled allowing the data bits to shift right from one stage to next.
- When Shift /LOAD line is LOW, gates G4, G5 and G6 are disabled, whereas gates G1, G2 and G3 are enabled allowing the data input to appear at the D inputs of the respective FFs.
- When clock pulse is applied, these data bits are shifted to the Q output terminals of the FFs and therefore the data is inputted in one step.
- The OR gate allows either the normal shifting operation or the parallel data entry depending on which AND gates are enabled by the level on the Shift /$\overline{\text{LOAD}}$ input.



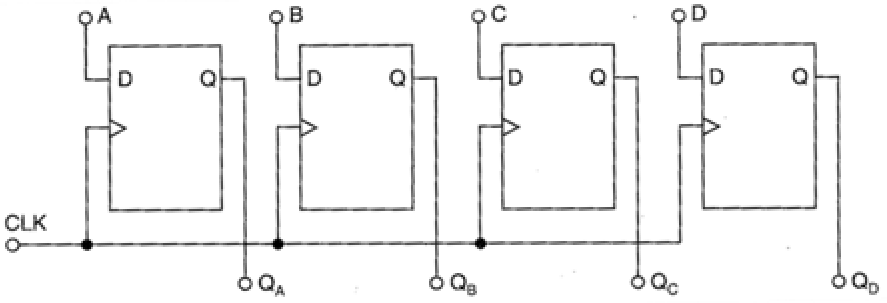
(a) Logic diagram

(b) Logic symbol

A 4- bit parallel in, serial out shift register

PARALLEL IN, PARALLEL OUT SHIFT REGISTER:-

- In a parallel in, parallel out shift register, the data entered into the register in parallel form and also the data taken out of the register in parallel form. Immediately following the simultaneous entry of all data bits appear on the parallel outputs.
- The figure shown below is a 4 bit parallel in parallel out shift register using D FFs.
- Data applied to the D input terminals of the FFs.
- When a clock pulse is applied at the positive edge of that pulse, the D inputs are shifted into the Q outputs of the FFs.
- The register now stores the data.
- The stored data is available instantaneously for shifting out in parallel form.

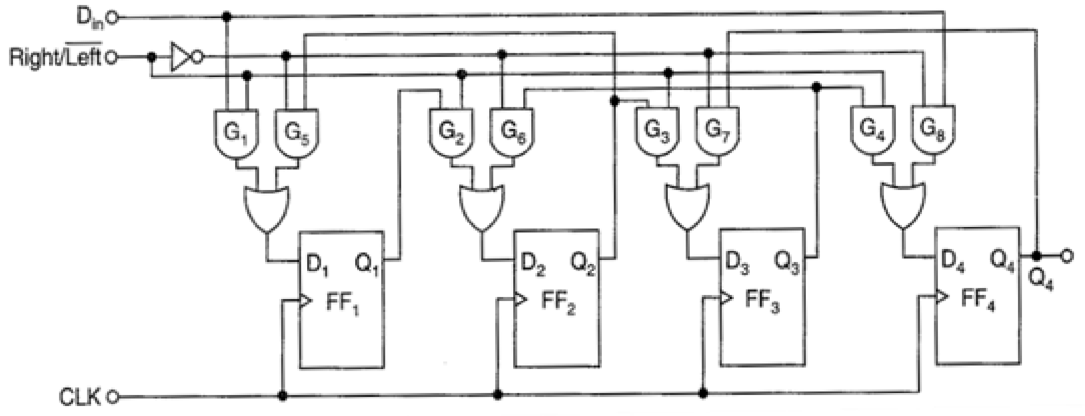


Logic diagram of a 4 – bit parallel in, parallel out shift register

BIDIRECTIONAL SHIFT REGISTER:-

- In bidirectional shift register is one in which the data bits can be shifted from left to right or from right to left.
- The figure shown below the logic diagram of a 4 bit serial in, serial out, bidirectional (shift-left, shift-right) shift register.
- Right /$\overline{\text{Left}}$ is the mode signal. When Right /$\overline{\text{Left}}$ is a 1, the logic circuit works as a shift right shift register. When Right /$\overline{\text{Left}}$ is a 0, the logic circuit works as a shift right shift register.

- The bidirectional is achieved by using the mode signal and two AND gates and one OR gate for each stage.
- A HIGH on the Right/$\overline{\text{Left}}$ control input enables the AND gates G_1, G_2, G_3 and G_4 and disables the AND gates G_5, G_6, G_7 and G_8 and the state of Q output of each FF is passed through the gate to the D input of the following FF. When clock pulse occurs, the data bits are effectively shifted one place to the right.
- A LOW Right/$\overline{\text{Left}}$ control input enables the AND gates G_5, G_6, G_7 and G_8 and disables the AND gates G_1, G_2, G_3 and G_4 and the Q output of each FF is passed to the D input of the preceding FF. When clock pulse occurs the data bits are then effectively shifted one place to the left.
- So, the circuit works as a bidirectional shift register.



Logic diagram of 4- bit bidirectional shift register

UNIVERSAL SHIFT REGISTERS:-

- The register which has both shifts and parallel load capabilities, it is referred as a universal shift register. So, universal shift register is a bidirectional register, whose input can be either in serial form or in parallel form and whose output also can be either in serial form or parallel form.
- The universal shift register can be realized using multiplexers.
- The figure shows the logic diagram of a 4 bit universal shift register that has all the capabilities of a general shift register.

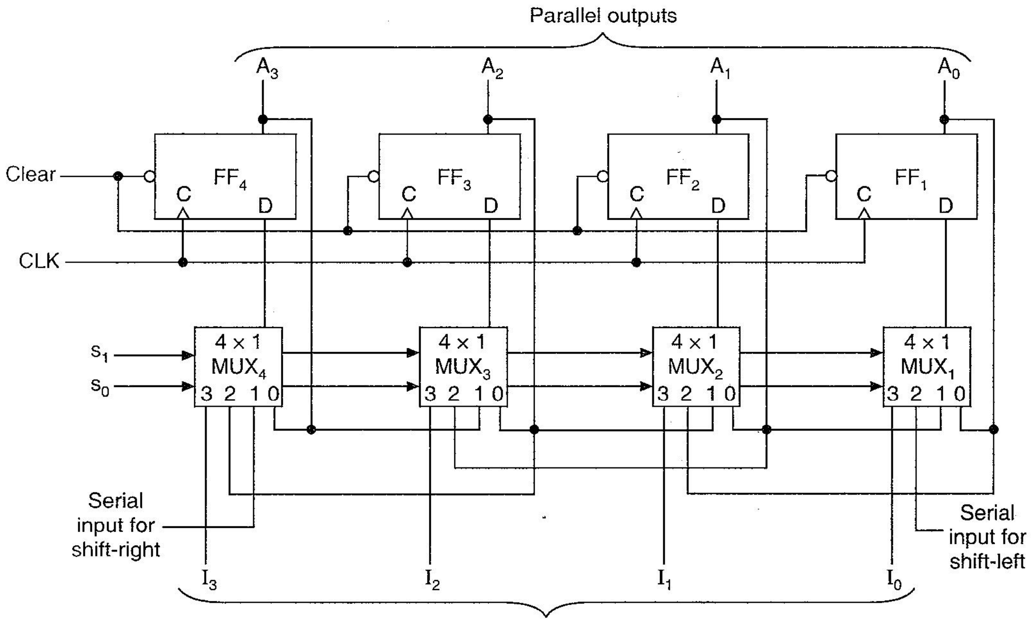
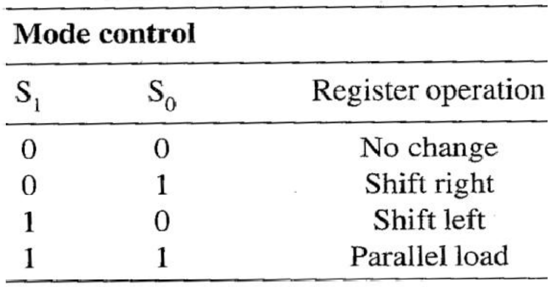


Fig- (a) 4 bit universal shift register

- It consists of four D flip- flops and four multiplexers.
- The four multiplexers have two common selection inputs S_1 and S_0.
- Input 0 in each multiplexer is selected when S_1S_0 = 00, input 1 is selected when S_1S_0 = 01, and input 2 is selected when S_1S_0 = 10 and input 3 is selected when S_1S_0= 11.
- The selection inputs control the mode of operation of the register is according to the function entries shown in the table.
- When S_1S_0 = 00 the present value of the register is applied to the D inputs of flip-flops. This condition forms a path from the output of each FF into the input of the same FF.
- The next clock edge transfers into each FF the binary value it held previously, and no change of state occurs.
- When S_1S_0 = 01, terminal 1 of the multiplexer inputs have a path of the D inputs of the flip- flops. This causes a shift right operation, with serial input transferred into FF_4.
- When S_1S_0 = 10 a shift left operation results with the other serial input going into the FF_1.
- Finally when S_1S_0 = 11, the binary information on the parallel input lines is transferred into the register simultaneously during the next clock edge.

Functional table for the register of fig – a:



Mode control		
S_1	S_0	Register operation
0	0	No change
0	1	Shift right
1	0	Shift left
1	1	Parallel load

<u>APPLICATIONS OF SHIFT REGISTERS</u>:-

1. **Time delays:**
   - In digital systems, it is necessary to delay the transfer of data until the operation of the other data have been completed, or to synchronize the arrival of data at a subsystem where it is processed with other data.
   - A shift register can be used to delay the arrival of serial data by a specific number of clock pulses, since the number of stages corresponds to the number of clock pulses required to shift each bit completely through the register.
   - The total time delay can be controlled by adjusting the clock frequency and by the number of stages in the register.
   - In practice, the clock frequency is fixed and the total delay can be adjusted only by controlling the number of stages through which the data is passed.
2. **Serial / Parallel data conversion:**
   - Transfer of data in parallel form is much faster than that in serial form.
   - Similarly the processing of data is much faster when all the data bits are available simultaneously. Thus in digital systems in which speed is important so to operate on data parallel form is used.
   - When large data is to be transmitted over long distances, transmitting data on parallel lines is costly and impracticable.
   - It is convenient and economical to transmit data in serial form, since serial data transmission requires only one line.

- Shift registers are used for converting serial data to parallel form, so that a serial input can be processed by a parallel system and for converting parallel data to serial form, so that parallel data can be transmitted serially.
- A serial in, parallel out shift register can be used to perform serial-to parallel conversion, and a parallel in, serial out shift register can be used to perform parallel- to –serial conversion.
- A universal shift register can be used to perform both the serial- to – parallel and parallel-to-serial data conversion.
- A bidirectional shift register can be used to reverse the order of data.

RING AND JOHNSON COUNTER:-

- Ring counters are constructed by modifying the serial-in, serial-out, shift register.
- There are two types of ring counters
  - i) Basic ring counter
  - ii) Johnson counter
- The basic ring counter can be obtained from a serial-in serial- out shift register by connecting the Q output of the last FF to the D input of the first FF.
- The Johnson counter can be obtained from serial-in, serial- out, shift register by connecting the Q output of the last FF to the D input of the first FF.
- Ring counter outputs can be used as a sequence of synchronizing pulses.
- The ring counter is a decimal counter.

D/A and A/D Converter

Weighted Register Network

The most significant bit (MSB) resistance is one-eighth of the least significant bit (LSB) resistance. R_L is much larger than 8R. The voltages V_A, V_B, V_C and V_D can be either equal to V (for logic 1) or 0 (for logical 0). Thus there are 2^4 = 16 input combinations from 0000 to 1111. The output voltage V_0, given by Millman's theorem is

V_0 =

When input is 0001, V_A = V_B= V_C= 0 and V_D = V and output is V/15. If input is 0010, V_A = V_B=V_D =0 and V_C=V giving an output of 2V/15. If input is 0011, V_A = V_B=0 and V_C=V_D = V giving an output of 3v/15. Thus, the output voltage varies from 0 to V in steps of V/15.

Binary Ladder Network

The weighted resistor network requires a range of resistor values. The binary ladder network requires only two resistance values. From node 1, the resistance to the digital source is 2R and resistance to ground is also 2R. From node 2, the resistance to digital source is 2R and resistance to ground =

R + (2R) (2R) / (2R+2R) = 2R

Thus, from each of the nodes 1,2,3,4, the resistance to source and ground is 2R each. A digital input 0001 means that D is connected to V and A, B, C are grounded. The output voltage V_0 is V/16. Thus as input varies from 0000 to 1111, the output varies from V/16 to V in steps of V/16.

A complete digital-to-analog converter circuit consists of a number of ladder networks (to deal with more bits of data), operational amplifier, gates etc.

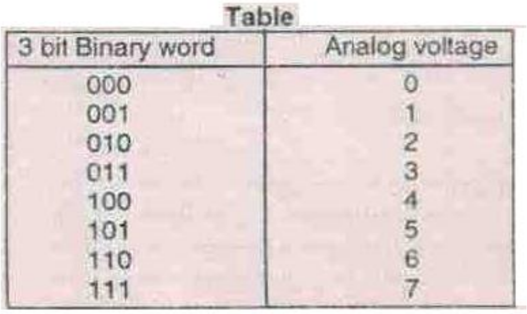
Performance Characteristics of D/A converters

The performance characteristics of D/A converters are resolution, accuracy, linear errors, monotonicity, setting time and temperature sensitivity.

(a) **Resolution:** It is the reciprocal of the number of discrete steps in the D/A output. Evidently resolution depends on the number of bits. The percentage resolution is [1/ (2^N-1)] * 100 where N is the number of bits. The percentage resolution for different values of N is given in table.

(b) **Accuracy:** It is a measure of the difference between actual output and expected output. It is expressed as a percentage of the maximum output voltage. If the maximum output voltage (or full scale deflection) is 5 V and accuracy is ±0.1%, then the maximum error is $\frac{0.1}{100}$ * 5 = 0.005 V or 5 mV. Ideally the accuracy should be better than ±0.5 of LSB. In an 8 bit converter, LSB is 1/256 or 0.39% of full scale. The accuracy should be better than 0.2%.

(c) **Setting Time:** When the input signal changes, it is desirable that analog output signal should immediately show the new output value. However in actual practice, the D/A converter takes some time to settle at the new position of the output voltage. Setting time is defined as the time taken by the D/A converter to settle with ±1/2 LSB of its final value when a change in input digital signal occurs. The final time taken to settle down to new value is due to the transients and oscillations in the output voltage. Figure shows the definition of setting time.



Table

3 bit Binary word	Analog voltage
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Fig 1

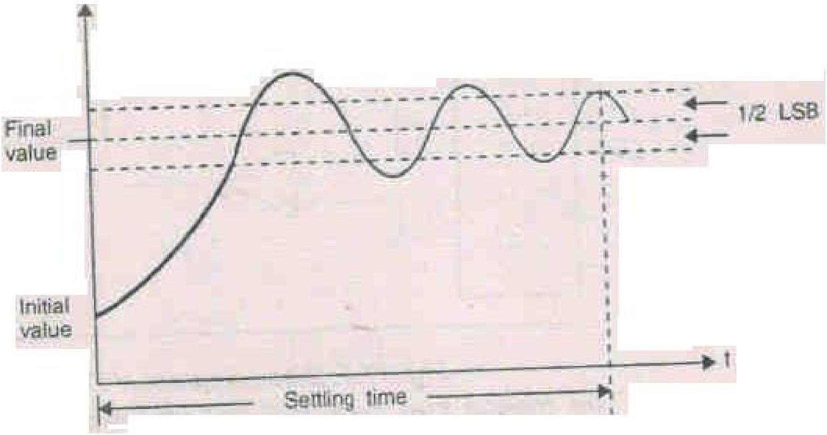


Fig. Settling time of D/A converter

Quantization error:

An analog to digital converter changes analog signal into digital signal. It is important to note that in D/A converter the number of input is fixed. In 4 bit D/a converter there are 16 possible inputs and in 6 bit D/A converter there are 64 possible inputs. However, in A/D converter the analog input voltage can have any value in the specified range but the digital output can have only 2^N discrete levels (for N bit converter). This means that there is a certain range of input voltage which correspond to every discrete output level.

Consider a 4 bit A/D converter having a resolution of 1 count per 100 mV. Fig (b) shows the analog input and digital output. It is seen that for input voltage range of 50 mV to 150 mV, the output is same i.e. 0001, for input voltage range of 150 mV to 250 mV, the output is the same, i.e. 0010. Thus we have one digital output for each 100 mV input range. If the digital signal of 0010 is fed to a D/A converter, it will show an output of 200 V whereas the original input voltage was between 150 V and 250 v. This error is called quntisation error and in this case this quntisation error can be ±50 mV and is equal to ±1/2 LSB.

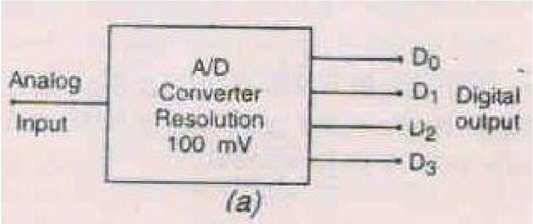


Fig (a) A/D Converter

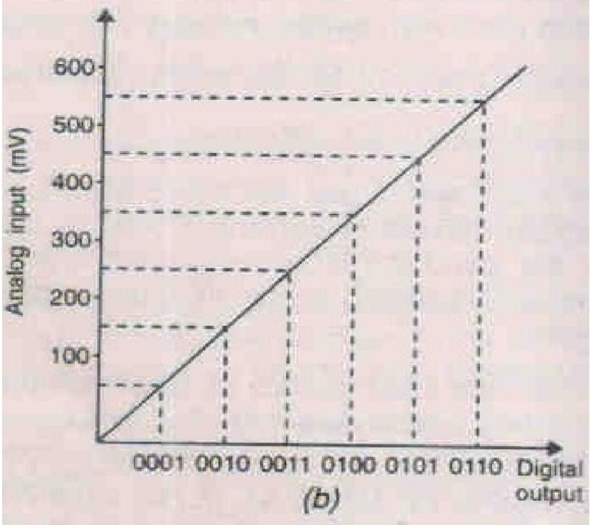


Fig (b) Quantisation error

Stair Step A/D Converter / Ramp A/D converter:

This converter is also called digital ramp or the counter type A/D converter. Figure shows the configuration for 8 bit converter. As seen in figure it uses a D/A converter and a binary counter to produce the digital number corresponding to analog input. The main components are comparator, AND gate, D/A converter, divide by 256 counter and latches. The analog input is given to non-inverting terminal of comparator. The D/A converter provides stair step reference voltage.

Let he counter be in reset state and output of D/A converter be zero. An analog input is given to non-inverting terminal of comparator. Since the reference input is 0, the comparator gives High output and enables the AND gate. The clock pulses cause advancing of counter through its binary states and stair step reference voltage is produced from D/A converter. As the counter keeps advancing, successively higher stair step output voltage is produced. When this stair step voltage reaches the level of analog input voltage, the comparator output goes Low and disables the AND gate. The clock pulses are cut off and counter stops. The state of counter at this point is equal to the number of steps in reference voltage at which comparison occurs. The binary number corresponding to this number of steps is the value of the analog input voltage. The control logic causes this binary number to be loaded into the latches and counter is reset.

This converter is rather slow in action because the counter has to pass through the maximum number of states before a conversion takes place. For 8 bit device this means 256 counter states.

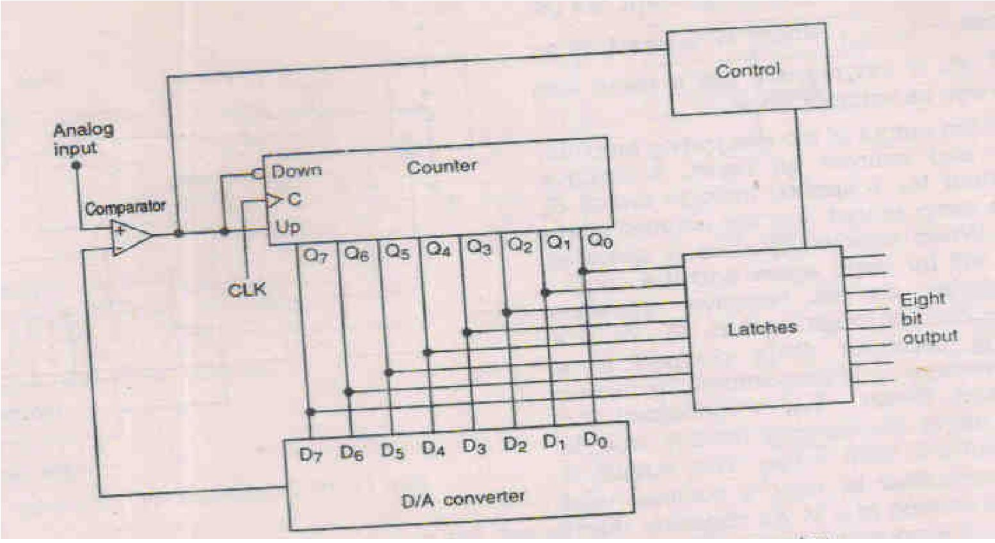


Fig (a) 8 bit up-down counter type A/D converter

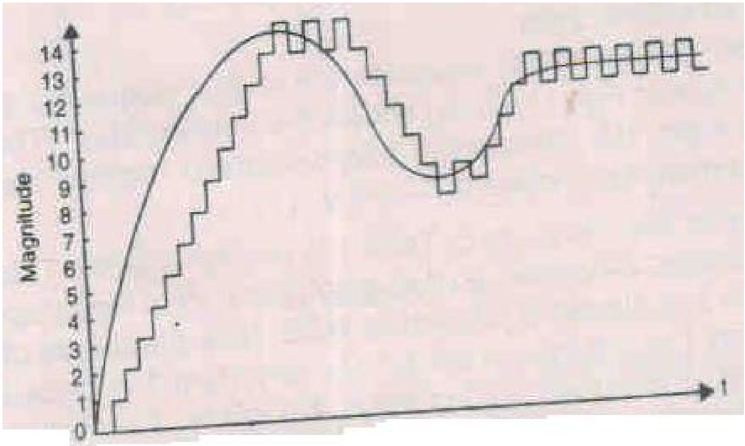


Fig (b) Tracking action of updown counter type A/D Converter

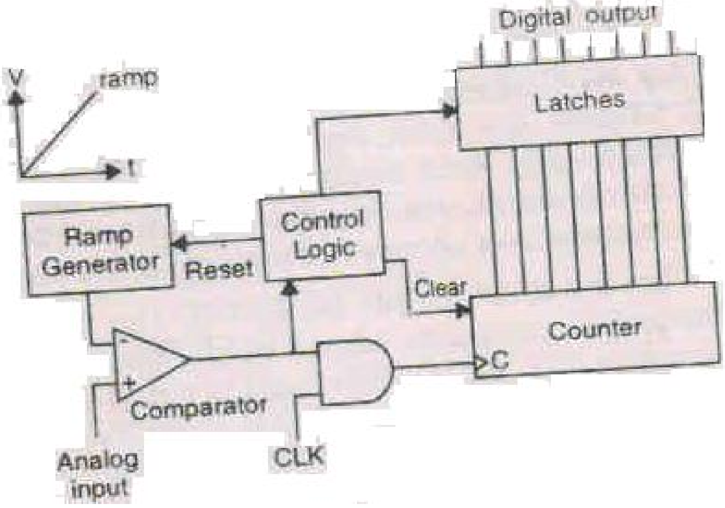


Fig (c) Single slope A/D converter

Dual slope A/D converter:

The single slope A/D converter is suscetible to noise. The dual slope converter is free from this problem. It uses an op-amp used as integreting amplifier for ramp generator. It is dual slope device because it uses a fixed slope ramp as well as variable slope ramp. Fig. Shows the configuration.

It is seen that the integreting op-amp uses a capacitor in the feedback path.

Output voltage of integreting op-amp = $-\frac{1}{C}\int i\,dt$ = $-\frac{1}{RC}\int V_{in}$ dt

Thus the output voltage is integral of analog input voltage. If V_{in} is constant, we get an output $-V_{in}\frac{t}{RC}$ which is a fixed slope ramp. If V_{in} is varing we get a ramp with fixed as well as variable slope.

Let the output of the integreting amplifier be zero and counter be reset. A positive analog input V_{in} is applied through switch S, we get a ramp output and the counter starts working. When counter reaches a specified count, it will be reset again and the control logic switches on the negative reference voltage $-V_{ref}$ (through switch S). At this instant the capacitor C is charged to a negative voltage - V proportional to analog input voltage. When $-V_{ref}$ is connected the capacitor starts discharging linearly due to constant current from $-V_{ref}$. The output of integreting amplifier is now a positive fixed slope ramp starting at – V. As capacitor discharges, the counter advances from the reset state. When the output of integretor becomes zero, the comparator output becomes Low and disables the clock signal to the AND gate. The counter is therefore stopped and the binary counter is latched. This completes one conversion cycle. The binary count is propor tional to analog input V_{in}.

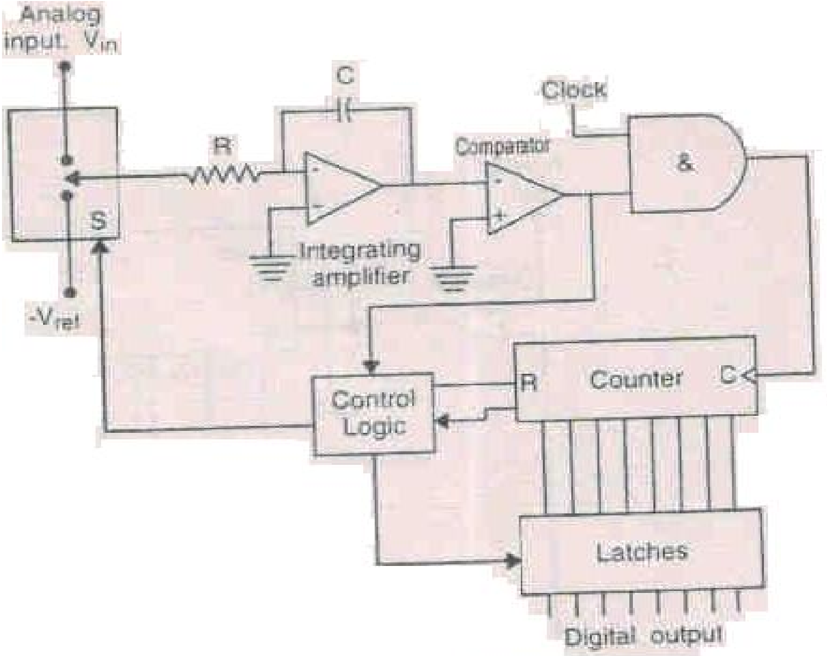


Fig. Dual slope A/D converter

Successive Approximation A/D Converter:

This is the most widely used A/D converter. As the name suggests the digital output tends towards analog input through successive approximations. Fig. Shows the configuration. The main components are op-amp comparator, control logic, SA (successive approximation) register and D/A converter. As shown it is a six bit device using a maximum reference of 64 V.

Let the analog input be 26.1 v. The SA register is first set to zero. Then 1 is placed in MSB. This is fed to D/A converter whose output goes to comparator. Since the analog input (26.1 V) is less than D/A output (i.e. 32 V) the MSB is set to zero. Then 1 is placed in bit next to MSB. Now the output of D/A is 16 V. Since analog input is more than 16 V, this 1 is retained in this bit position. Next 1 is placed in third bit position. Now the D/A output is 24 V which is less than analog input. Therefore this 1 bit is retained and 1 is placed in the next bit. Now the D/A output is 28 V, which is more than analog input. Therefore this 1 bit is set to zero and 1 is placed in 5th bit position producing a D/A output of 26 V. It is less than analog input. Therefore this 1 bit is retained. Now 1 is placed in LSB producing a D/A output of 27 V which is more than analog input. Therefore LSB is set to zero and the converter gives an output of 26 V.

The successive approximation method of A/D converter is very fast and takes only about 250 ns/ bit.

Performance Characteristics of A/D converters:

The performance characteristics of A/D converters are resolution, accuracy, A/D gain and drift and A/D speed.

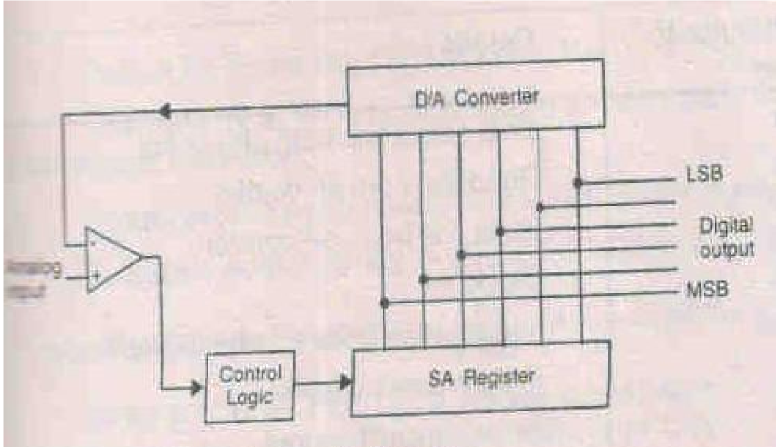
(a) Resolution: A/D rsolution is the change in voltage input necessary for a one bit change in output. It can also be expressed as percent.

(b) A/D Accuracy: The accuracy of A/D conversion is limited by the ±1/2 LSB due to quantisation error and the other errors of the system. It is defined as the maximum deviation of digital output from the ideal linear reference line. Ideally it aproaches ±1/2 LSB.

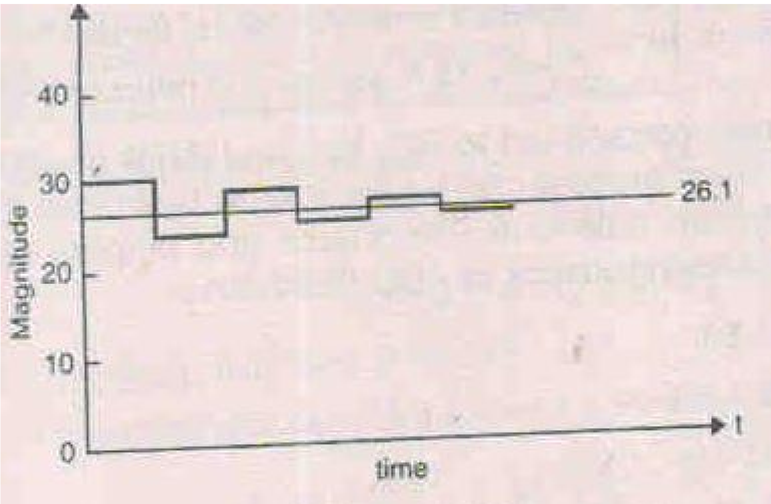
(c) A/D gain and Drift: A/D gain is the voltage output is devided by the voltage input at the linearity reference line. It can usually be zeroed out.

Drift means change in circuit parameters with time. Drift errors of upto ±1/2 LSB will cause a maximum errors of one LSB between the first and the last transition. Very low drift is quite difficult to achieve and increases cost of the device.

(d) A/D speed: It can be defined in two ways, i.e. either the time necessary to do one conversion or the line between successive conversion at the highest rate possible. Speed depends on the settling time of components and the speed of the logic.



(a) Block diagram



(b) Waveforms

GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

Digital Signal Processing

For 6th Semester

ELECTRONICS & TELECOMMUNICATION

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Miss. Jasmita Sahu

(Lecturer in Electronics & Telecommunication Engineering)

MODULE-1

Introduction

Signal:

A *signal* is defined as any physical quantity th a t varies with time, space, or any other in dependent variable or variables. Mathematically, we describe a signal as a function of one or mo re independent variables. For example, the functions

s(t)= 5t

describe a signal, one that varies linearly with the in dependent variable *t* (time).

$$s(x, y) = 3x + 2xy + 10y^2$$

This function describes a signal of two in dependent variables *x* and y that could represent the two spatial coordinates in a p lane.

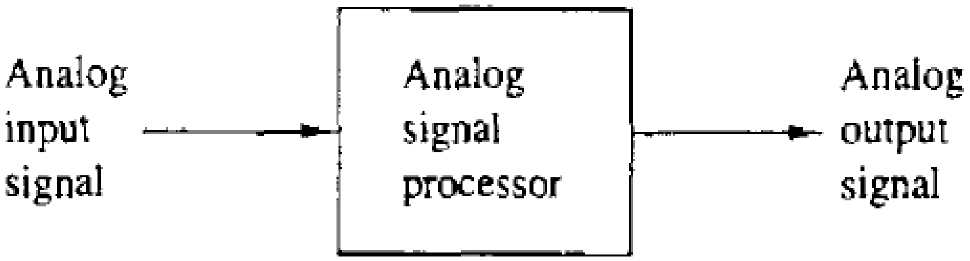
System:

A *system* may also be defined as a physical device th a t performs an operatioon a signal. For ex ample, a filter used to reduce the noise and interference corrupting desired in formation bearing signal is called a system .

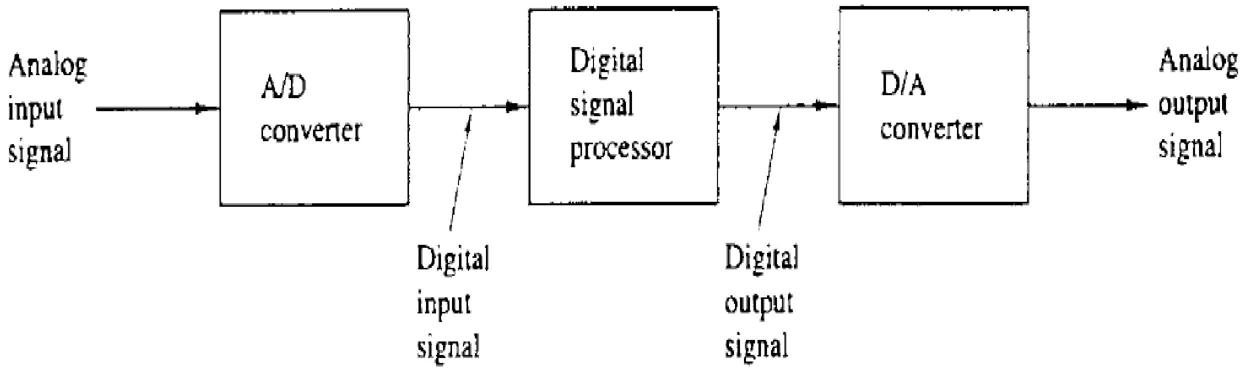
signal processing:

W h en we pass a signal thrugh a system , as in filtering, we say that we have processed the signal. In this case the processing of the signal involves filtering the noise and interference from the desired signal. If the operation on the signal is non linear, the system is said to be non linear, and so forth . Such operations are usually referred to as *signal rocessing.*

Analog signal processing:



Digital signal processing:



Advantages of Digital over Analog Signal Processing :

1- a digital programmable system allow s flexibility in re configuring the digital signal processing operations simply by changing the program .

2- a digital system provides much better control of accuracy .

3- Digital signals are easily stored on magnetic media (tape or disk) without deterioration or loss of signal fidelity beyond that introduced in the A /D conversion.

4- digital implementation of the signal processing system is cheaper than analog signal processing.

Limitations:

One practical limitation is the speed of operation of A /D converters and digital signal processors. We shall see that signals having extremely wide band widths require fast-sampling -rate A /D converters and fast digital signal processors. Hence there are analog signals with large bandwidths for which a digital processing approach is beyond the state of the art of digital hardware.

Discrete time signals and systems

CLASSIFICATION OF SIGNALS : There are 3 types of signals

Continuous-time signals: *Continuous-time signals* or *analog signals* are defined for every value of time.

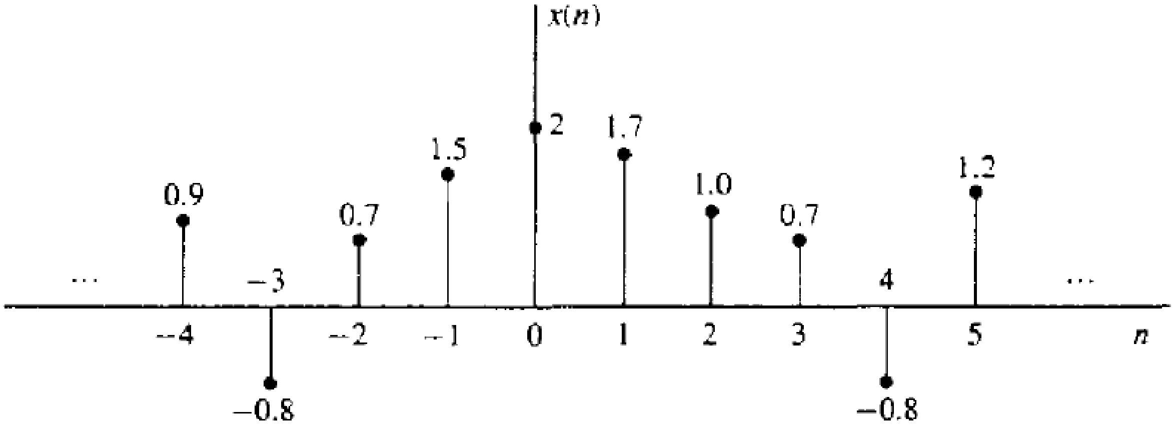
Discrete-time signals : Discrete-time signals are defined only at certain specific values of time.

Digital Signals: digital signal is defined as a function of an integer independent variable and its values are taken from a finite set of possible values, which are represented by a string of 0's and l's .

DISCRETE-TIME SIGNALS : A disc rete-time signal *x{n)* is a function of an in dependent variable that is an integer. discrete-time signal is *not defined* at instants between two successive samples. Simply, the signal *x (n)* is not defined for non integer values of *n*. So x(n) was obtained from sampling an analog signal x a(t), then .i(n) = x a(nT) , where T is the sampling period (i.e., the time between successive samples).

Representation of discrete-time signal :

A discrete-time signal can be represented in various way**.** But all can be represented graphically.



Graphical representation of a discrete-time signal.

Besides the graphical representation of a discrete-time signal or sequence as illustrated in above Fig. there are some alternative representations that are often more convenient to use. These are:

1. **Functional representation** :

$$x(n) = \begin{cases} 1, & \text{for } n = 1, 3 \\ 4, & \text{for } n = 2 \\ 0, & \text{elsewhere} \end{cases}$$

2. **Tabular representation :**

n	$\cdots$	-2	-1	0	1	2	3	4	5	$\ldots$
$x(n)$	$\cdots$	0	0	0	1	4	1	0	0	$\ldots$

3.Sequence representation : An infinite-duration signal or sequence with the time origin *(n* = 0) indicated by the symbol ↑ is represented as

$$x(n) = \{\ldots 0, \underset{\uparrow}{0}, 1, 4, 1, 0, 0, \ldots\}$$

A finite-duration sequence can be represented as

$$x(n) = \{3, -1, \underset{\uparrow}{-2}, 5, 0, 4, -1\}$$

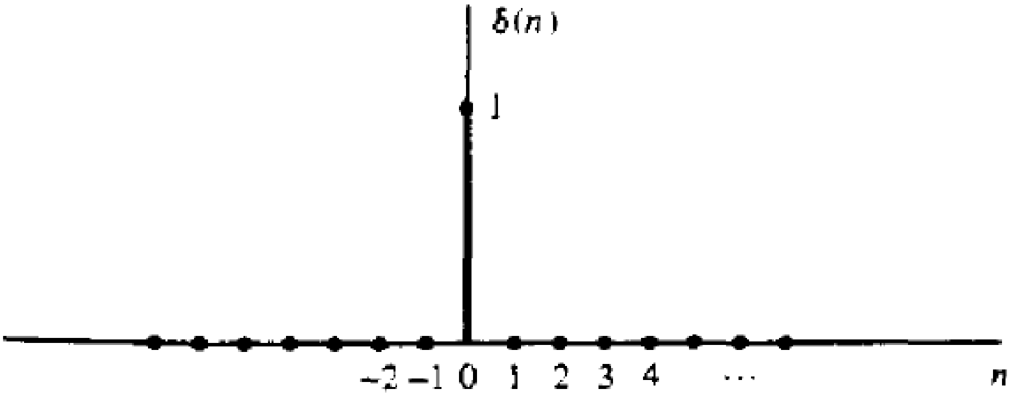
Some Elementary Discrete-Time Signals :

In discrete-time signals and systems there are a number of basic signals that appear often and play an important role. These signals are defined below .

1. Unit sample sequence/ unit impulse : It is denote d as δ(n) and is defined as

$$\delta(n) \equiv \begin{cases} 1, & \text{for } n = 0 \\ 0, & \text{for } n \neq 0 \end{cases}$$

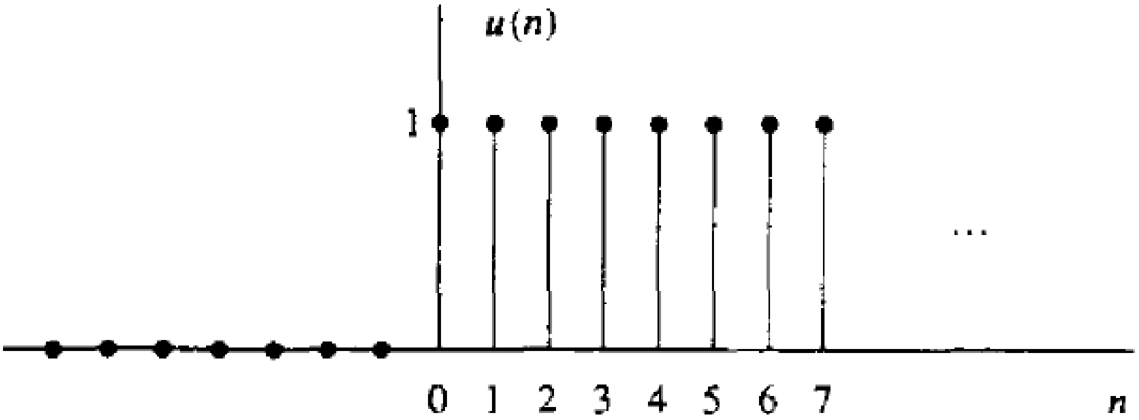
the unit impulse sequence is a signal that is zero every where, except at n =0 where its value is unity. The graphical representation of *δ(n)* is



2. **Unit step signal:** It is denoted as u(n) and is defined as

$$u(n) \equiv \begin{cases} 1, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases}$$

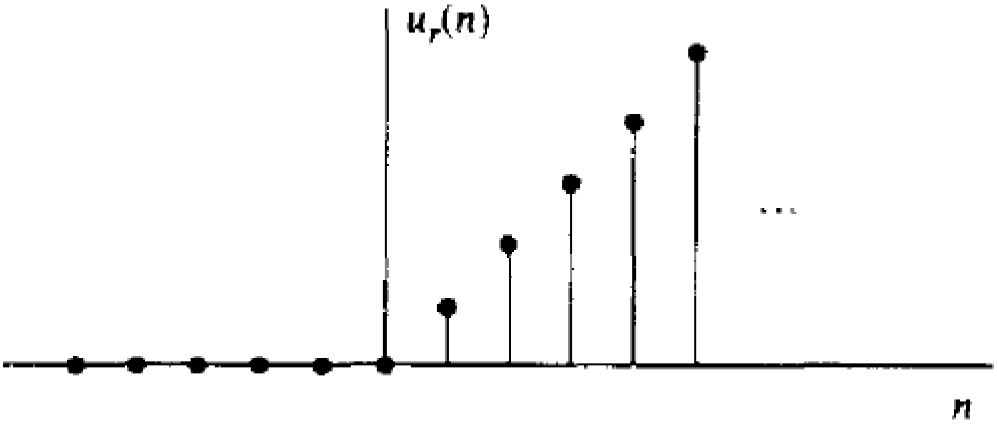
The graphical re presentation of *u(n)* is



3. **Unit ramp signal** *:* It is denoted as $u_r(n)$ and is defined as

$$u_r(n) \equiv \begin{cases} n, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases}$$

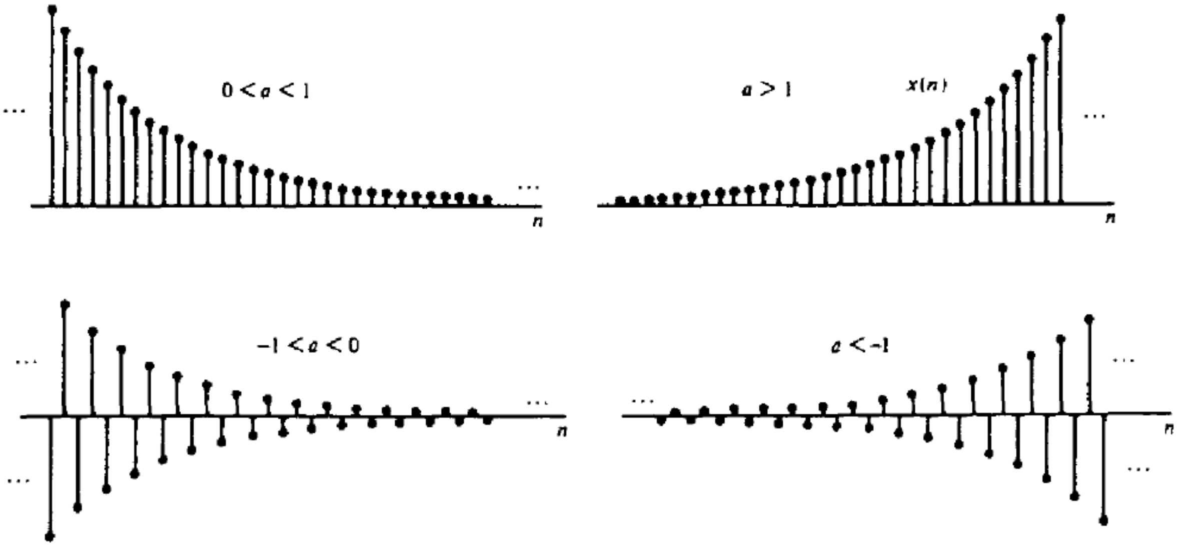
The graphical representation of u $_r$(n) is



4-Exponential signal : It is a sequence of the form

$$x(n) = a^n \qquad \text{for all } n$$

If the parameter *a* is real, then x(n) is a real signal. illustratation of *x(n)* for various values of the parameter *a is*



When the parameter *a* is complex valued , it can be expressed as

$$a \equiv re^{j\theta}$$

where *r* and *Ө* are now the parameters. Hence we can express *x(n)* as

$$\begin{aligned} x(n) &= r^n e^{j\theta n} \\ &= r^n(\cos\theta n + j\sin\theta n) \end{aligned}$$

Classification of Discrete-Time Signals:

1-**Energy signals and power signals:** The energy E of a signal *x(n)* is defined as

$$E \equiv \sum_{n=-\infty}^{\infty} |x(n)|^2$$

If *E* is finite (i.e., 0 <*E* <∞), if *E* is finite , *P* = 0. then x(n) is called an *energy signal.*

Many signals that possess infinite energy, have a finite average power. The average power of a d iscrete-time signal *x(n)* is defined as

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x(n)|^2$$

If we define the signal energy of *x(n)* over the finite interval —*N* < *n* < *N* as

$$E_N \equiv \sum_{n=-N}^{N} |x(n)|^2$$

the average power of the signal *x(n)* as

$$P \equiv \lim_{N\to\infty} \frac{1}{2N+1} E_N$$

if *E* is infinite and *P* is finite. the signal is called a *power signal.*

2-Periodic signals and aperiodic signals:

signal *x(n)* is periodic with period *N (N* > 0) if an d only if

$$x(n+N) = x(n) \text{ for all } n$$

the sinusoidal signal of the form

$$x(n) = A \sin 2\pi f_0 n$$

is periodic when f_0, is a rational number, that is, if f_0 can be expressed as

$$f_0 = \frac{k}{N}$$

where *k* and *N* are integers.

3-Symmetric (even) and antisymmetric (odd) signals :

A real valued signal *x (n*) is called symmetric (even) if

$$x(-n) = x(n)$$

O n the other hand , a signal *x(n)* is called antisymmetric (odd) if

$$x(-n) = -x(n)$$

We can illustrate that any arbitrary signal can be expressed as the sum of two signal components, one of which is even and the other odd. The even signal component is formed by adding x(n) to *x (—n)* and dividing by 2. that is.

$$x_e(n) = \tfrac{1}{2}[x(n) + x(-n)]$$

Similarly, w e form an odd signal component *$x_0(n)$* according to the relation

$$x_o(n) = \frac{1}{2}[x(n) - x(-n)]$$

So we obtain x(n),that is,

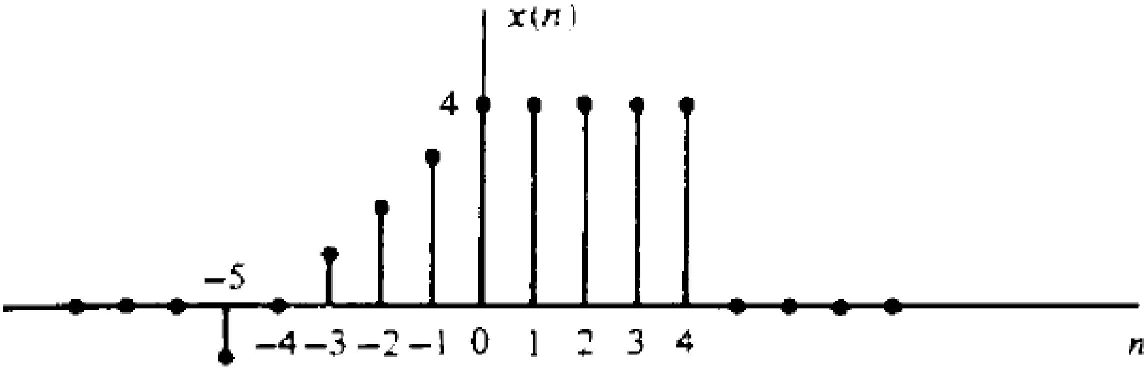
$$x(n) = x_e(n) + x_o(n)$$

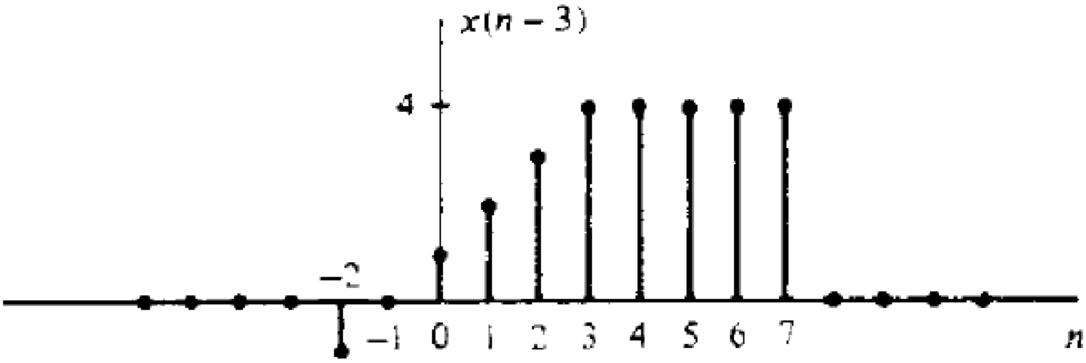
Simple Manipulations of Discrete-Time Signals :

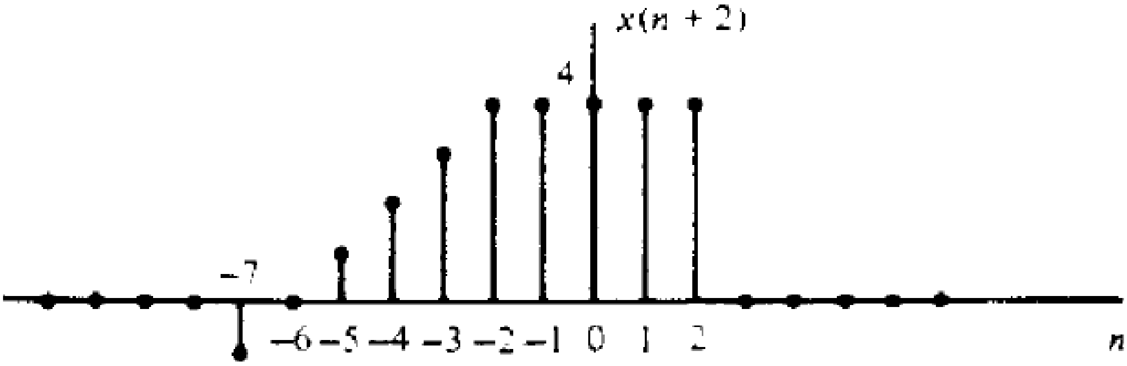
Time shifting :

A signal *x (n)* may be shifted in time by replacing the independent variable *n* by *n — k,* w here *k* is an integer. If k is a positive integer, the time shift results in a delay of the signal by *k* units o f time. If *k* is a negative integer, the time shift results in an advance of the signal by \k\\ units in time.

Ex- A signal *x(n)* is graphically illustrated in Fig. below. Show a graphical representation of the signals *x(n* — 3) and *x(n* + 2).





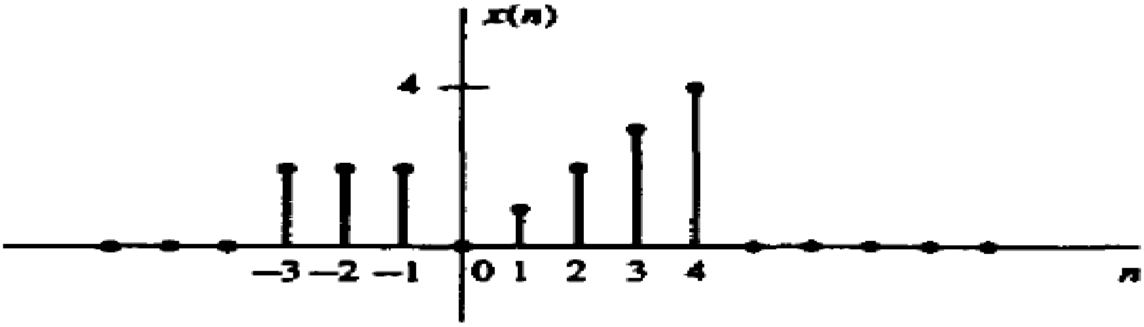


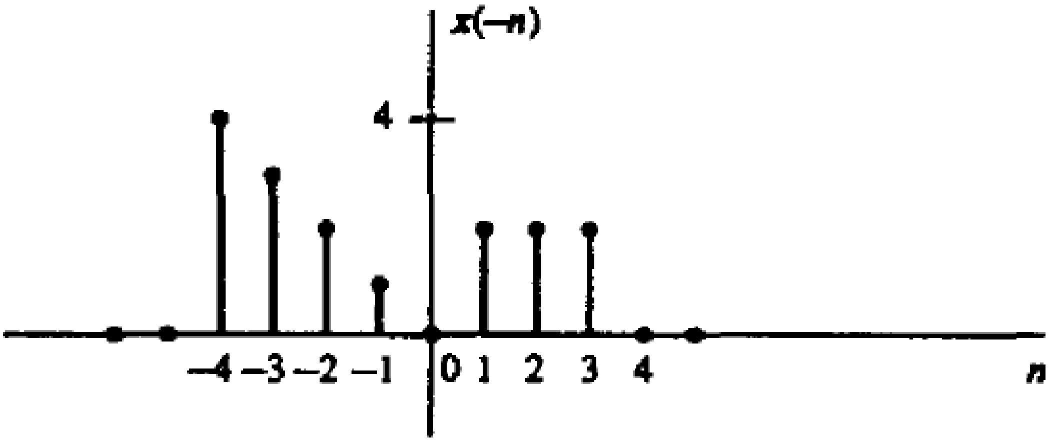
The signal *x* (n — 3) is obtained by delaying x(n) by three units in time. On the other hand, the signal *x(n* + 2) is obtained by advancing *x (n)* by two units in time. Note that delay corresponds to shifting a signal to the right, whereas advance implies shifting the signal to the left on the time axis.

Time Folding : The operations of folding is defined by

FD[x(n)] = *x (— n)*

Example:





Addition, multiplication, and scaling of sequences:

Amplitude modifications include *addition, multiplication,* and *scaling* o f discrete-time signals. *Amplitude scaling* o f a signal by a constant *A* is accomplished by multiplying the value o f every signal sample by *A.*

$$y(n) = Ax(n) \qquad -\infty < n < \infty$$

The sum of two signals x1(n) an d x2(n) is a signal y(n), whose value at any instant is equal to the sum of the values of these two signals at that instant, that is.

$$y(n) = x_1(n) + x_2(n) \qquad -\infty < n < \infty$$

The product of two signals is similarly defined on a sample -to -sample basis as

$$y(n) = x_1(n)x_2(n) \qquad -\infty < n < \infty$$

DISCRETE-TIME SYSTEMS :

A *discrete-time system* is a device or algorithm that operates on a discrete -time signal, called the *input* o r *excitation,* according to some w ell-defined rule, to produce another discrete-time signal called the *output* or *response* of the system .

We say that the input signal *x(n)* is *Transformed* by the system in to a signal y(n), and the general relationship Between x(n) and *y(n)* as

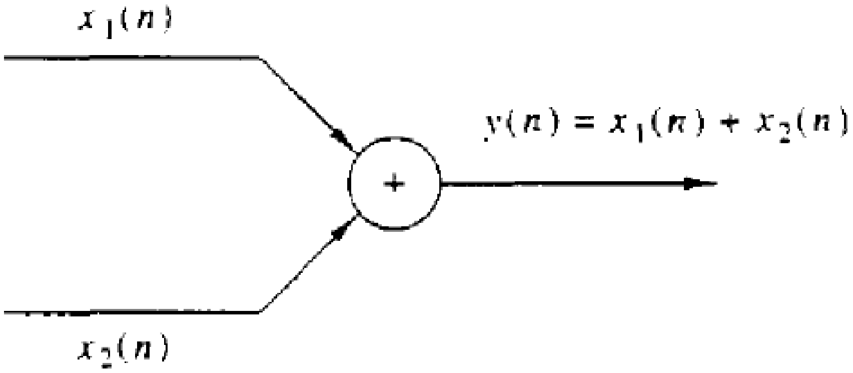
$$y(n) \equiv \mathcal{T}[x(n)]$$

where the symbol *T* denotes the transformation (also called an operator), or processing performed by the system on x(n) to produce y(n).

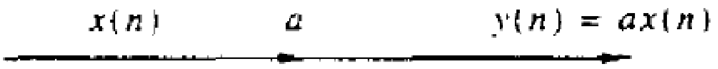
Representation of Discrete-Time Systems :

It is useful at this point to introduce a block diagram representation of discrete time systems. For this purpose we need to define some basic building blocks that can be interconnected to form complex systems.

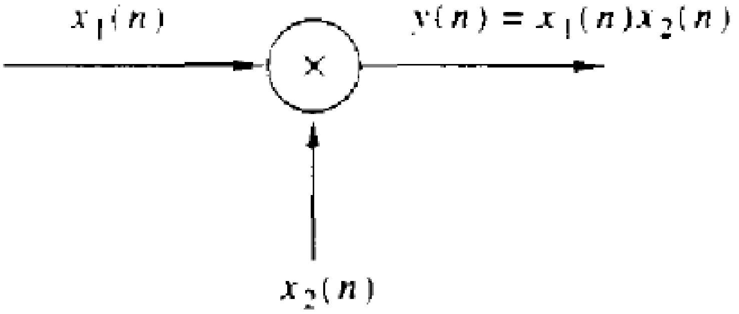
An adder: Figure below illustrates a system (adder) that perform s the addition o f two signal sequences to form another (the sum) sequence, which we denote as y(n).



A constant multiplier: This operation is depicted by below Fig., and simply represents applying a scale factor on the input *x (n).*



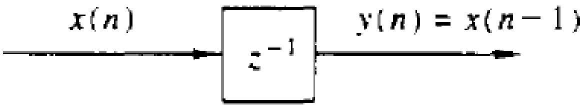
A signal multiplier: Figure below illustrates the multiplication of two signal sequences to form another (the product) sequence, denoted in the figure as y(n). we can view the multiplication operation as memory less.



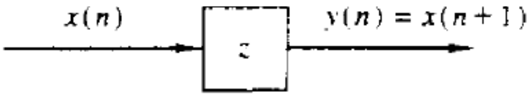
A unit delay element: The unit delay is a special system that simply delays the signal passing th rough it by one sample. Fig. below illustrates such a system .If the input signal is *x(n),* the output is *x(n* — 1). In fact, the sample *x{n* — 1) is stored in memory at time *n* — 1 an d it is recalled fro m memory at time *n* to form y(n),

$$y(n) = x(n-1)$$

T h e use o f the symbol z^{-1} to denote the unit of delay



A unit advance element: In contrast to the unit delay, a unit advance moves the input *x (n*) ahead by one sample in time to yield *x(n* + 1). Fig. below illustrates this operation , with the operator z being used to denote the unit advance.



Classification of Discrete-Time Systems :

There are various types of Discrete-Time Systems such as

1-Static versus dynamic systems:

A discrete-tim e system is called *static* or memory less if its output at any instant n depends at most on the input sample at the same time, but not on past or future samples of the input. In any other case, the system is said to be *dynamic* or to have memory .T h e systems described by the following input-output equations are both static or memory less

$$y(n) = a\,x\,\{n)$$
$$y\,(n) = nx(n) + b\,x^3(n)$$

On the other hand , the systems described by the following input-output relations are dynamic systems or systems with memory.

$$y(n) = x(n) + 3x(n-1)$$

$$y(n) = \sum_{k=0}^{n} x(n-k)$$

$$y(n) = \sum_{k=0}^{\infty} x(n-k)$$

Time-invariant versus time-variant systems: We can subdivide the general class of systems in to the two broad categories, time -invariant systems and time -variant systems. A system is called time-in variant if its input-output characteristics do not change with time.
A relaxed system *T* is *time invariant* o r *shift invariant* if and
only if

$$x(n) \xrightarrow{T} y(n)$$

implies that for every in p u t signal x(n) a n d every time shift *k.*

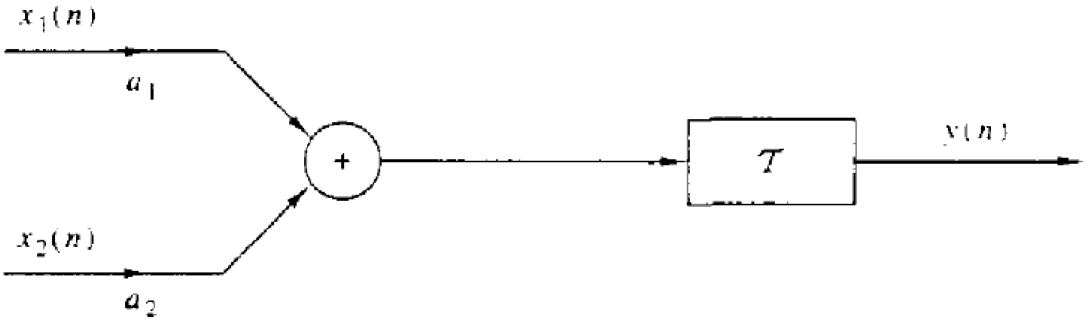
$$x(n-k) \xrightarrow{T} y(n-k)$$

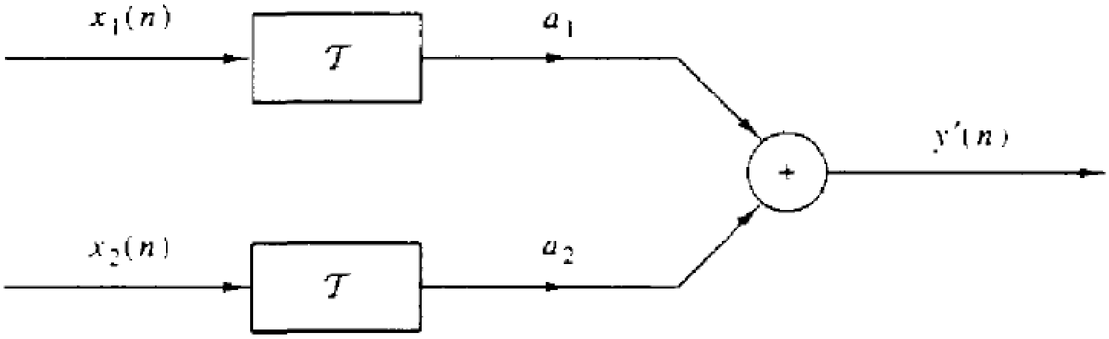
Now if this output y{n, k) = y{n — k), for all possible values o f k, the system is time invariant. O n the other hand , if the output y(n, k) ≠ y(n — k), even for one value o f k, the system is time variant.

Linear versus nonlinear systems: The general class o f system s can also be subdivided into linear system s and nonlinear system s. A linear system is one that satisfies the *superposition principle.* Simply stated, the principle o f superposition requires that the response o f the system to a weighted sum o f signals be equal to the corresponding weighted sum of the responses (outputs) of the system to each of the individual input signals**.**
A relaxed T system is linear if and only if

$$\mathcal{T}[a_1x_1(n) + a_2x_2(n)] = a_1\mathcal{T}[x_1(n)] + a_2\mathcal{T}[x_2(n)]$$

for any arbitrary input sequences x_1 *(n)* and x_2*(n),* and any arbitrary constants a_1 and a_2.





Causal versus noncausal systems:
A system is said to be *causal* if the output of the system at any time *n* [i.e., *y(n)]* depends only on present and past inputs [i.e., *x { n*), *x(n* - 1),*x(n* — 2) , . . .] , but does not depend on future inputs [i.e., *x(n* + 1), *x(n* + 2) , . . .] . In mathematical terms, the output of a causal system satisfies an equation of the form

$$y(n) = F[x(n), x(n-1), x(n-2), \ldots]$$

If a system does not satisfy this definition, it is called *noncausal.* Such a system has an output tha t depends not only on present and past inputs but also on future inputs.

Stable versus unstable systems:
An arbitrary relaxed system is said to be stable if an d only if every bounded input produces a bounded output (i:e; BIBO).

The conditions that the input sequence *x{n)* and the output sequence *y(n)* are bounded is transla ted mathematically to mean that there exist some finite numbers, say *M x* and *M y.* such that

$$|x(n)| \le M_x < \infty \qquad |y(n)| \le M_y < \infty$$

for all *n.* If. for some bounded input sequence ,x(n), the output is unbounded (infinite), the system is classified as unstable .

DISCRETE-TIME LINEAR TIME-INVARIANT SYSTEMS:
The linearity and time-invariance properties of the system , the response of the system to any arb itrary input signal can be expressed in terms of the unit sample response of the system . The gen eral form of the expression that relates the unit sample response of the system and the arbitrary input signal to the output signal, called the convolution sum or the convolution formula, is also derived. Thus we are able to determine the output of any linear, time-invariant system to any arbitrary input signal.

Response of LTI Systems to Arbitrary Inputs:

The Convolution Sum :
An arbitrary input signal *x(n)* in to a weighted sum of impulses, We are now ready to determine the response of any relaxed linear system to any Input signal. First, we denote the response y(n, k) of the system to the input unit Sample sequence at *n* = k by the special symbol *h(n, k), -*∞<*k* < ∞. That is,

$$y(n, k) \equiv h(n. k) = \mathcal{T}[\delta(n-k)]$$

if the input is the arbitrary signal x(n) that is expressed as a sum of weighted impulses, that is.

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k)$$

then the response of the system to *x(n)* is the corresponding sum of weighted outputs, that is,

$$y(n) = \mathcal{T}[x(n)] = \mathcal{T}\left[\sum_{k=-\infty}^{\infty} x(k)\delta(n-k)\right]$$

$$= \sum_{k=-\infty}^{\infty} x(k)\mathcal{T}[\delta(n-k)]$$

$$= \sum_{k=-\infty}^{\infty} x(k)h(n.k)$$

Clearly, the above equation follows from the superposition property of linear systems, and is know n as the *superposition summation.*th en by the time-invariance property , the response of the system to the delayed unit sample sequence δ(n - *k)* is

$$h(n-k) = \mathcal{T}[\delta(n-k)]$$

Consequently , the *superposition summation* formula in reduces to

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

The above formula gives the response *y(n)* of the LTI system as a function of the input signal *x (n*) and the unit sample (impulse) response *h(n)* is called a *convolution sum.*

To summarize, the process of computing the convolution between *x (k)* and *h(k)* involves the following four steps**.**

1. *Folding.* **Fold** *h(k)* **about** *k* **= 0 to obtain** *h (- k)* .

2. *Shifting,* Shift *h (—k)* by n_0 to the right (left) if n_0 is positive (negative), to obtain *h(*n_0— k).

3. *Multiplication.* Multiply *x (k)* by *h(*n_0— k) to obtain the product sequence$v_{n0}(k) = x(k)\, h(n_0 — k)$.

4. *Summation.* Sum all the values o f the product sequence $v_{n0}(k)$ to obtain the value of the output at time *n* = n_0.

Example:

The impulse response of a linear time-invariant system is

$$h(n) = \{1, \underset{\uparrow}{2}, 1, -1\}$$

Determine the response of the system to the input signal

$$x(n) = \{\underset{\uparrow}{1}, 2. 3, 1\}$$

Solution : We shall compute the convolution according to its formula. But we shall use graphs of the sequences to aid us in the computation. In Fig. below we illustrate the input signal sequence *x(k)* and the impulse response *h{k)* of the system, using *k* as the time index. The first step in the computation of the convolution sum is to fold *h(k).* The folded sequence *h(-k)* is illustrated inconsequent figs . Now we can compute the output at *n* = 0. according to the convolution formula which is

$$y(0) = \sum_{k=-\infty}^{\infty} x(k)h(-k)$$

Since the shift *n* = 0, we use *h(—k)* directly without shifting it. The product sequence

$$v_0(k) \equiv x(k)h(-k)$$

We continue the computation by evaluating the response of the system at *n* = 1.

$$y(1) = \sum_{h=-\infty}^{\infty} x(k)h(1-k)$$

Finally, the sum of all the values in the product sequence yields

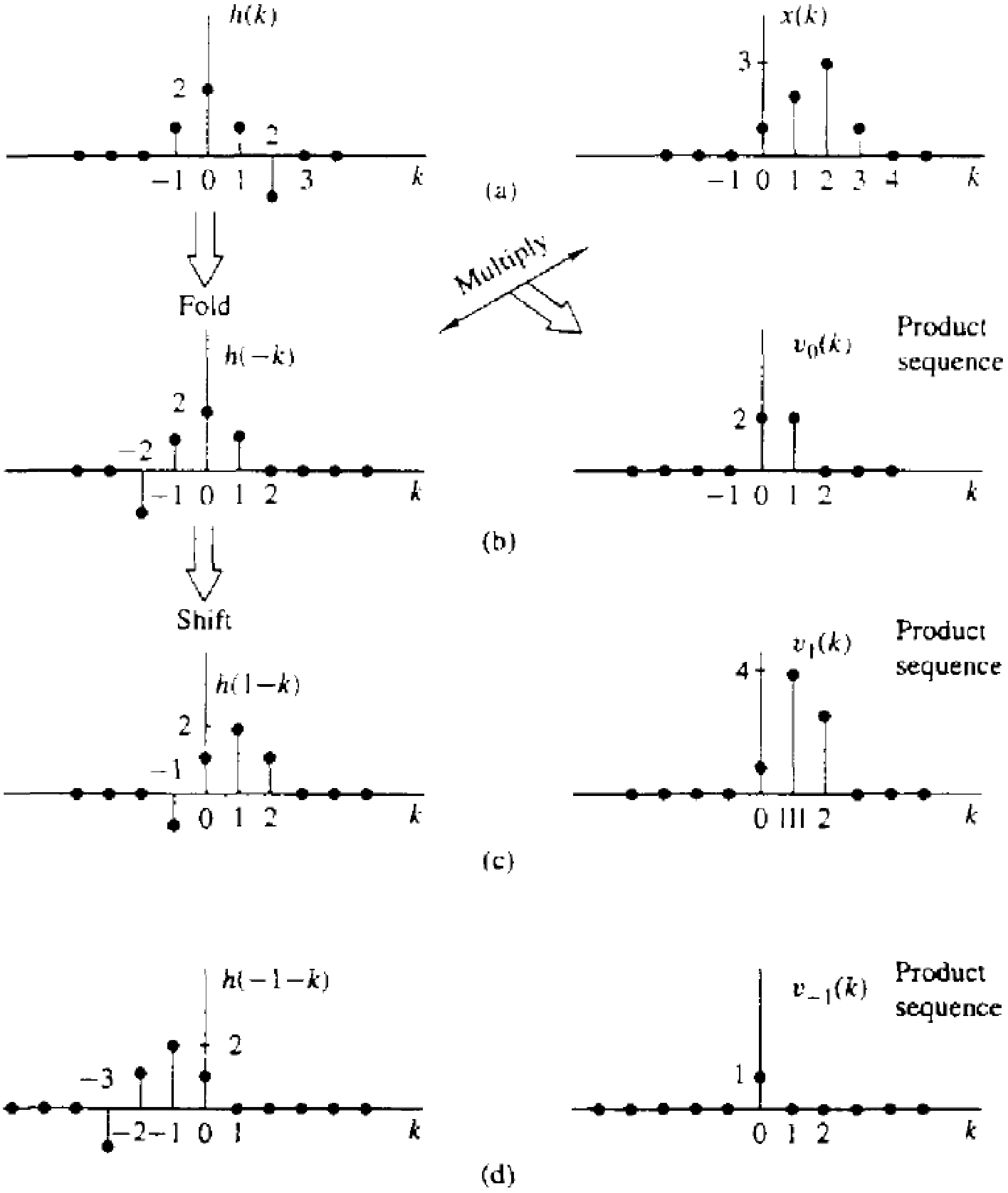
$$y(1) = \sum_{k=-\infty}^{\infty} v_1(k) = 8$$

In a similar manner, we can obtain y(2) by shifting *h (- k)* two units to the right. And y(2) = 8.

Then y(3) = 3. y(4) = - 2 , y(5) = -1 .For *n* >5, we find that y(n) = 0 because the product sequences contain all zeros.

Next we wish to evaluate y(n) for *n* < 0. We begin with *n =-1.Then*

$$y(-1) = \sum_{k=-\infty}^{\infty} x(k)h(-1-k)$$



$$y(0) = \sum_{h=-\infty}^{\infty} v_0(k) = 4$$

Finally, summing over the values of the product sequence, we obtain

$$y(-1) = 1$$

then

$$y(n) = 0 \qquad \text{for } n \le -2$$

Now we have the entire response of the system for -∞ <*n* < ∞. which we summarize below as

$$y(n) = \{\dots, 0, 0, 1, \underset{\uparrow}{4}, 8, 8, 3, -2, -1, 0, 0, \dots\}$$

Properties of Convolution:

1- **Commutative law :**

$$x(n) * h(n) = h(n) * x(n)$$

2- **Associative law :**

$$[x(n) * h_1(n)] * h_2(n) = x(n) * [h_1(n) * h_2(n)]$$

3-Distributive law :

$$x(n) * [h_1(n) + h_2(n)] = x(n) * h_1(n) + x(n) * h_2(n)$$

Finite-Duration and Infinite-Duration Impulse Response system:

Linear time-invariant system s into two types, those that have a finite-duration Impulse response (FIR) and those that have an infinite-duration impulse response(IIR). Thus an fir system has an impulse response that is zero outside o f some Finite time interval.

Stability and unstable Linear Time-Invariant Systems :

We defined an arbitrary relaxed system as BIBO stable if and only if its output sequence y(n) is bounded for every bounded input *x(n).*

The output is bounded if the impulse response of the system satisfies the condition

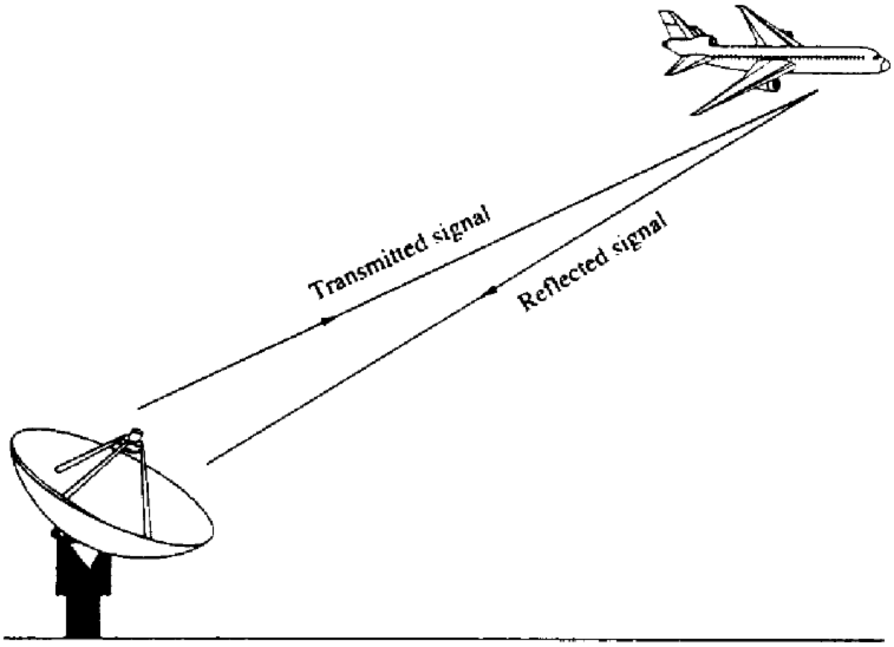
$$S_h \equiv \sum_{k=-\infty}^{\infty} |h(k)| < \infty$$

T hat is, *a linear time-invariant system is stable if its impulse response is absolutely summable* .

CORRELATION OF DISCRETE-TIME SIGNALS:

A mathematical operation that closely resembles convolution is correlation .Just as in the case of convolution , two signal sequences are involved in correlation. correlation between the two signals is to measure the degree to which the two signals are similar and thus to extract some in formation that depends to a large extent on the application. Correlation o f signals is often encountered in radar, sonar, digital communications, geology, an do the rare as in science and en gineering .

Let us suppose that we have two signal sequences x(n) and y(n) that we wish to compare. In radar and active sonar applications. x(n) can represent the sampled version of the transmitted signal and y{n) can represent the sampled version of the received signal at the output of the analog -to -digital (A /D) converter. If a target is p resent in the space being searched by the radar or sonar, the received signal y(n) consists of a delayed version of the transmitted signal, reflected from the target.



This comparison process is performed by means of the correlation operation of 2 different types.

Cross-correlation and Autocorrelation Sequences :

Suppose that we have two real signal sequences *x(n)* and *y(n)* each of which has finite energy. T he *cross-correlation* o f *x(n)* and y(n) is a sequence $r_{xy}(l)$, which is defined as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l) \qquad l = 0, \pm 1, \pm 2, \ldots$$

or, equivalently , as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n+l)y(n) \qquad l = 0, \pm 1, \pm 2, \ldots$$

The index l is the (time) shift (or *lag)* parameter and the subscripts *x y* on the cross-correlation se quence $r_{xy}(l)$, indicate the sequences being correlated .If we reverse the roles of x(n) an d y(n) and there fore reverse the order of the indices xy. we obtain the cross-correlation sequence

$$r_{yx}(l) = \sum_{n=-\infty}^{\infty} y(n)x(n-l)$$

or, equivalently ,

$$r_{yx}(l) = \sum_{n=-\infty}^{\infty} y(n+l)x(n)$$

By comparing the above 4 equations we conclude that

$$r_{xy}(l) = r_{yx}(-l)$$

Hence , $r_{yx}(l)$ provides exactly the same information as $r_{xy}(l)$,with respect to the similarity of *x (n)* to y(n).

Example:
Determine the cross-correlation sequence $r_{xy}(l)$ of the sequences

$$x(n) = \{\ldots, 0, 0, 2, -1, 3, 7, \underset{\uparrow}{1}, 2, -3, 0, 0, \ldots\}$$

$$y(n) = \{\ldots, 0, 0, 1, -1, 2, -2, \underset{\uparrow}{4}, 1, -2, 5, 0, 0, \ldots\}$$

Solution : Let us use the definition of cross-correlation to compute $r_{xy}(l)$. For $I = 0$ w e have

$$r_{xy}(0) = \sum_{n=-\infty}^{\infty} x(n)y(n)$$

The product sequence $v_0(n) = x(n)\, y(n)$ is

$$v_0(n) = \{\ldots, 0, 0, 2, 1, 6, -14, \underset{\uparrow}{4}, 2, 6, 0, 0, \ldots\}$$

and hence the sum over all values of *n* is

$$r_{xy}(0) = 7$$

For $I > 0$, we simply shift y(n) to the right relative to x(n) hy l units, compute the product sequence $v_l(n)$ = x(n)y(n — *I),* and finally, sum over all values o f the product sequence. Thus we obtain

$$\begin{array}{llll} r_{xy}(1) = 13, & r_{xy}(2) = -18, & r_{xy}(3) = 16, & r_{xy}(4) = -7 \\ r_{xy}(5) = 5, & r_{xy}(6) = -3, & r_{xy}(l) = 0, & l \geq 7 \end{array}$$

For l< 0, we shift *y(n)* to the left relative to x(n) by l units, compute the product sequence $v_l(n)$ = x(n)y(n — *I),* and sum over all values of the product sequence. Thus we obtain the values of the cross-correlation sequence

$$\begin{array}{llll} r_{xy}(-1) = 0, & r_{xy}(-2) = 33, & r_{xy}(-3) = -14, & r_{xy}(-4) = 36 \\ r_{xy}(-5) = 19, & r_{xy}(-6) = -9, & r_{xy}(-7) = 10, & r_{xy}(l) = 0, \ l \leq -8 \end{array}$$

Therefore, the cross-correlation sequence of *x{n)* and y(n) is

$$r_{xy}(l) = \{10, -9, 19, 36, -14, 33, 0, \underset{\uparrow}{7}, 13, -18, 16, -7, 5, -3\}$$

Then the convolution o f *x(n)* with y (—*n)* yields the cross-correlation $r_{xy}(l)$ that is,

$$r_{xy}(l) = x(l) * y(-l)$$

Autocorrelatio*n:*
when *y(n) = x(n),* we have the *autocorrelation* of x(n),which is defined as the sequence

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n)x(n-l)$$

or, equivalently, as

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n+l)x(n)$$

For finite-duration sequences,

$$r_{xy}(l) = \sum_{n=i}^{N-|k|-1} x(n)y(n-l)$$

and

$$r_{xx}(l) = \sum_{n=i}^{N-|k|-1} x(n)x(n-l)$$

where *i* = *l*, *k* = 0 for l> 0, and i = 0, *k = l* for l < 0.

Properties of the Autocorrelation and Crosscorrelation Sequences :

1-The cross-correlation sequence satisfies the condition that

$$|r_{xy}(l)| \leq \sqrt{r_{xx}(0)r_{yy}(0)} = \sqrt{E_x E_y}$$

when y(n) = *x (n*), reduces to

$$|r_{xx}(l)| \leq r_{xx}(0) = E_x$$

2-Th e normalized auto correlation sequence is defined as

$$\rho_{xx}(l) = \frac{r_{xx}(l)}{r_{xx}(0)}$$

Similarly, we define the normalized cross-correlation sequence

$$\rho_{xy}(l) = \frac{r_{xy}(l)}{\sqrt{r_{xx}(0)r_{yy}(0)}}$$

Now $\backslash\rho_{xx}\{l)\backslash$ <1 and $\backslash\rho_{xy}\{l)\backslash$ < 1, and hence these sequences are independent of signal scaling.

3-the cross-correlation sequence satisfies the property

$$r_{xy}(l) = r_{yx}(-l)$$

the autocorrelation sequence satisfies the property

$$r_{xx}(l) = r_{xx}(-l)$$

Hence the auto correlation function is an even function.

MODULE-2

The One-sided z-Transform:

The one-sided or unilateral z-transform of a signal x(n) is defined by

$$X^{+}(z) \equiv \sum_{n=0}^{\infty} x(n) z^{-n} \qquad \text{…………………………………..(1.1)}$$

Properties:

1. It does not contain information about the signal x(n) for negative values of time.
2. It is unique only for causal signals.
3. The one-sided z-transform $X^{+}(z)$ of x(n) is identical to the two-sided z-transform of the signal x(n)u(n).

Shifting Property:

❖ Time delay:

If $\quad x(n) \overset{z^{+}}{\leftrightarrow} X^{+}(z)$

then $\quad x(n-k) \overset{z^{+}}{\leftrightarrow} z^{-k}[X^{+}(z) + \sum_{n=1}^{k} x(-n) z^{n}] \quad$ k > 0 ……(1.2)

In case x(n) is a causal signal

then $\quad x(n-k) \overset{z^{+}}{\leftrightarrow} z^{-k} X^{+}(z) \quad$ k > 0 ………………………(1.3)

❖ Time advance:

$x(n+k) \overset{z^{+}}{\leftrightarrow} z^{k}[X^{+}(z) - \sum_{n=0}^{k-1} x(n) z^{-n}] \quad$ k > 0 …………………(1.4)

Final Value Theorem:

If $\quad x(n) \overset{z^{+}}{\leftrightarrow} X^{+}(z)$

then $\quad \lim_{n\to\infty} x(n) = \lim_{z\to 1}(z-1)\, X^{+}(z) \quad$ ……………....(1.5)

The limit exists if the ROC of $(z-1)X^{+}(z)$ includes the unit circle.

<u>Analysis of LTI System in z-domain:</u>

Response of Systems with Rational System:

We consider a linear constant coefficient difference equation:

$$y(n) = -\sum_{k=1}^{N} a_k\, y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.1)$$

corresponding system function H(z) is given by

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots .\ (2.2)$$

we apply an input signal x(n) whose z-transform is X(z). For $X(z) = \frac{N(z)}{Q(z)}$, $H(z) = \frac{B(z)}{A(z)}$ and zero initial conditions, the z-transform of the output of the system has the form

$$Y(z) = H(z)X(z) = \frac{B(z)N(z)}{A(z)Q(z)} \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots .... (2.3)$$

Suppose the system contains simple poles p_1 , p_2 , ……… , p_N and X(z) contains poles q_1 , q_2 , ……… , q_L , where $p_k \neq q_m$ for all k = 1, 2 , …… , N and m = 1, 2 , ……. , L. Assuming no pole-zero cancellation the partial fraction expansion of Y(z) yields

$$Y(z) = \sum_{k=1}^{N} \frac{A_k}{1-p_k z^{-1}} + \sum_{k=1}^{L} \frac{Q_k}{1-q_k z^{-1}} \ \ldots\ldots\ldots\ldots...................... (2.4)$$

The inverse transform of Y(z) is the output signal y(n) from the system:

$$y(n) = \sum_{k=1}^{N} A_k (p_k)^n u(n) + \sum_{k=1}^{L} Q_k (q_k)^n u(n) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots ... (2.5)$$

where scale factors {A$_k$}and {Q$_k$}are functions of both sets of poles {p$_k$}and {q$_k$}.

Response of Pole-Zero Systems with Non-zero Initial Conditions:

We consider the input signal x(n) to be a causal signal applied at n=0. The effects of all previous input signals to the system are reflected in the initial conditions y(-1), y(-2), ... , y(-N). We are interested in determining the output y(n) for $n \geq 0$.

$$Y^+(z) = -\sum_{k=1}^{N} a_k z^{-k} \left[Y^+(z) + \sum_{n=1}^{k} y(-n) z^n \right] + \sum_{k=0}^{M} b_k z^{-k} X^+(z)$$

Causality and Stability:

A causal linear time invariant system is one whose unit sample response h(n) satisfies the condition

$$h(n) = 0 \qquad\qquad n < 0$$

An LTI system is causal if and only if the ROC of the system function is the exterior of a circle of radius $r < \infty$, including the point $z = \infty$.

A necessary and sufficient condition for an LTI system to be BIBO stable is

$$\sum_{n=-\infty}^{\infty} |h(n)| < \infty$$

An LTI system is BIBO stable if and only if the ROC of the system function includes the unit circle.

Consequently, a causal and stable system must have a system function that converges for $|z| > r < 1$. Since the ROC cannot contain any poles of H(z) , it follows that ***a causal linear time-invariant system is BIBO stable if and only if all the poles of H(z) are inside the unit circle***.

The DFT as a Linear Transformation:

The formulas for the DFT and IDFT may be expressed as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad , \qquad k = 0,1, \dots ., N-1 \quad \dots \dots \dots \dots \dots .. (3.1)$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)\, W_N^{-kn} \ , \qquad n = 0,1, \dots , N-1 \quad \dots \dots \dots \dots \dots \dots (3.2)$$

where $\qquad W_N = e^{\frac{-j2\pi}{N}}$

which is an Nth root of unity.

The computation of each point of the DFT can be accomplished by N complex multiplications and (N-1) complex additions. Hence the N-point DFT values can be computed in a total of N^2 complex multiplications and N(N-1) complex additions.

Let us define an N-point vector $\mathbf{x}_N$ of the signal sequence x(n), n=0,1,…,N-1, an N-point vector $\mathbf{X}_N$ of frequency samples, and an $N \times N$ matrix $\boldsymbol{W}_N$ as

$$\mathbf{x}_N = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix} , \quad \mathbf{X}_N = \begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix}$$

$$\boldsymbol{W}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ \vdots & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots..(3.3)$$

With these definitions, the N-point DFT may be expressed in the matrix form as

$$\mathbf{X}_N = \mathbf{W}_N \mathbf{x}_N \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.(3.4)$$

where $\mathbf{W}_N$ is the matrix of the linear transformation. $\mathbf{W}_N$ is a symmetric matrix. If we assume that the inverse of $\mathbf{W}_N$ exists, then we also write

$$\mathbf{x}_N = \mathbf{W}_N{}^{-1} \mathbf{X}_N \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.(3.5)$$

IDFT can also be expressed as

$$\mathbf{x}_N = \frac{1}{N} \mathbf{W}_N^* \mathbf{X}_N \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.(3.6)$$

where $\mathbf{W}_N^*$ denotes the complex conjugate of the matrix $\mathbf{W}_N$. Comparison of equations 3.5 and 3.6 leads us to conclude that

$$\mathbf{W}_N{}^{-1} = \frac{1}{N} \mathbf{W}_N^* \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.7)$$

which in turn implies

$$\mathbf{W}_N \mathbf{W}_N^* = N \mathbf{I}_N \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.(3.8)$$

where $\mathbf{I}_N$ is a $N \times N$ identity matrix.

<u>Circular Convolution:</u>

Suppose that we have two finite-duration sequences of length N, $x_1(n)$ and $x_2(n)$. Their respective N point DFTs are

$$X_1(k) = \sum_{n=0}^{N-1} x_1(n) e^{-j2\pi nk/N} \quad , \quad k = 0,1,\ldots,N-1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.1)$$

$$X_2(k) = \sum_{n=0}^{N-1} x_2(n)e^{-j2\pi nk/N} \quad , \quad k = 0,1,\dots,N-1 \quad \dots\dots\dots\dots\dots\dots\dots (4.2)$$

Multiplying the above two DFTs we get:

$$X_3(k) = X_1(k)X_2(k) \ , \qquad k = 0,1,\dots,N-1 \quad \dots\dots\dots\dots\dots\dots\dots (4.2)$$

IDFT of $\{X_3(k)\}$ is

$$x_3(m) = \frac{1}{N}\sum_{n=0}^{N-1} X_3(k)e^{\frac{j2\pi km}{N}}$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} X_1(k)X_2(k)e^{\frac{j2\pi km}{N}} \quad \dots\dots\dots\dots\dots\dots\dots (4.3)$$

Substituting for $X_1(k)$ and $X_2(k)$ in (4.3) using DFTs given in (4.1) and (4.2), we obtain

$$x_3(m) = \frac{1}{N}\sum_{n=0}^{N-1}\left[\sum_{n=0}^{N-1} x_1(n)e^{-\frac{j2\pi nk}{N}}\right]\left[\sum_{n=0}^{N-1} x_2(n)e^{-\frac{j2\pi nk}{N}}\right]e^{\frac{j2\pi km}{N}} \quad \dots\dots\dots\dots (4.4)$$

The inner sum in the brackets in (4.4) has the form

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N\,, & a = 1 \\ \dfrac{1-a^N}{1-a}\,, & a \neq 1 \end{cases} \quad \dots\dots\dots\dots\dots\dots (4.5)$$

where a is defined as

$$a = e^{j2\pi(m-n-l)/N}$$

Consequently,

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N\,, & l = m - n + pN = ((m-n))_N\,, \quad p\ an\ integer \\ 0\,, & otherwise \end{cases} \quad \dots\dots\dots\dots (4.6)$$

If we substitute the result in (4.6) into (4.4) , we obtain

$$\boldsymbol{x_3(m)} = \sum_{\boldsymbol{n=0}}^{\boldsymbol{N-1}} \boldsymbol{x_1(n)x_2\,((m-n))_N} \quad , \qquad m = 0,1,\dots,N-1 \quad \dots\dots\dots\dots (4.7)$$

The above convolution sum is called ***circular convolution***. Thus we conclude that ***multiplication of the DFTs of two sequences is equivalent to the circular convolution of the two sequences in the time domain.***

Linear Filtering Methods Based on the DFT:

Use of the DFT in Linear Filtering:

Suppose we have a finite-duration sequence x(n) of length L which excites an FIR filter of length M. Let

$$x(n) = 0, \quad n < 0 \; and \; n \geq L$$

$$h(n) = 0, \quad n < 0 \; and \; n \geq M$$

where $h(n)$ is the impulse response of the FIR filter.

The output sequence $y(n)$ of the FIR filter:

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k) \qquad \ldots\ldots\ldots\ldots\ldots\ldots (5.1)$$

The duration of $y(n)$ is $L + M - 1$.

The frequency-domain equivalent to (5.1) is

$$Y(\omega) = X(\omega)H(\omega) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5.2)$$

If the sequence $y(n)$ is to be represented uniquely in the frequency domain by samples of its spectrum $Y(\omega)$ at a set of discrete frequencies, the number of distinct samples must equal or exceed $L + M - 1$. Therefore, a DFT of size $N \geq L + M - 1$ is required to represent {y(n)} in the frequency domain.

Now if

$$Y(k) \equiv Y(\omega)|_{\omega=\frac{2\pi k}{N}}, \qquad k = 0,1,\ldots,N-1$$

$$= X(\omega)H(\omega)|_{\omega=\frac{2\pi k}{N}}, \qquad k = 0,1,\ldots,N-1$$

then

$$Y(k) = X(k)H(k), \qquad k = 0,1,\ldots,N-1 \qquad \ldots\ldots\ldots\ldots\ldots..(5.3)$$

where $\{X(k)\}$ and $\{H(k)\}$ are the N-point DFTs of the corresponding sequences x(n) and h(n), respectively. Since the sequences x(n) and h(n) have a duration less than N, we simply pad these sequences with zeros to increase their length to N.

Since the ($N = L + M - 1$)-point DFT of the output sequence y(n) is sufficient to represent y(n) in the frequency domain, it follows that the multiplication of the N-point DFTs X(k) and H(k) followed by the computation of the N-point IDFT, must yield sequence {y(n)}.

Thus, ***the N-point circular convolution of x(n) with h(n) must be equivalent to the linear convolution of x(n) with h(n)***. *Thus with zero padding, the DFT can be used to perform linear filtering.*

Filtering of Long Data Sequences:

Let the FIR filter has duration M. The input data sequence is segmented into blocks of L points, where , by assumption, $L \gg M$.

Overlap-save method:

Size of input data blocks, $N = L + N - 1$

DFTs and IDFTs are of length N.

Each data block consists of the last $M - 1$ data points of the previous data block followed by L new data points to form a data sequence of length $N = L + N - 1$. An N-point DFT is computed for each data block.

The impulse response of the FIR filter is increased in length by appending $L - 1$ zeros and an N-point DFT of the sequence is computed once and stored. The multiplication of the two N-point DFTs $\{H(k)\}$ and $\{X_m(k)\}$ for the mth block of data yields

$$\hat{Y}_m(k) = H(k)X_m(k), \qquad k = 0,1,\dots,N-1 \qquad \dots\dots\dots\dots(5.4.1)$$

Then the N-point IDFT yields the result

$$\hat{Y}_m(n) = \{\hat{y}_m(0)\hat{y}_m(1)\dots\hat{y}_m(M-1)\hat{y}_m(M)\dots\hat{y}_m(N-1)\} \qquad \dots\dots\dots(5.4.2)$$

Since the data record is of length N, the first $M - 1$ points of $y_m(n)$ are corrupted by aliasing and must be discarded. The last L points of $y_m(n)$ are exactly same as the result from linear convolution and, as a consequence,

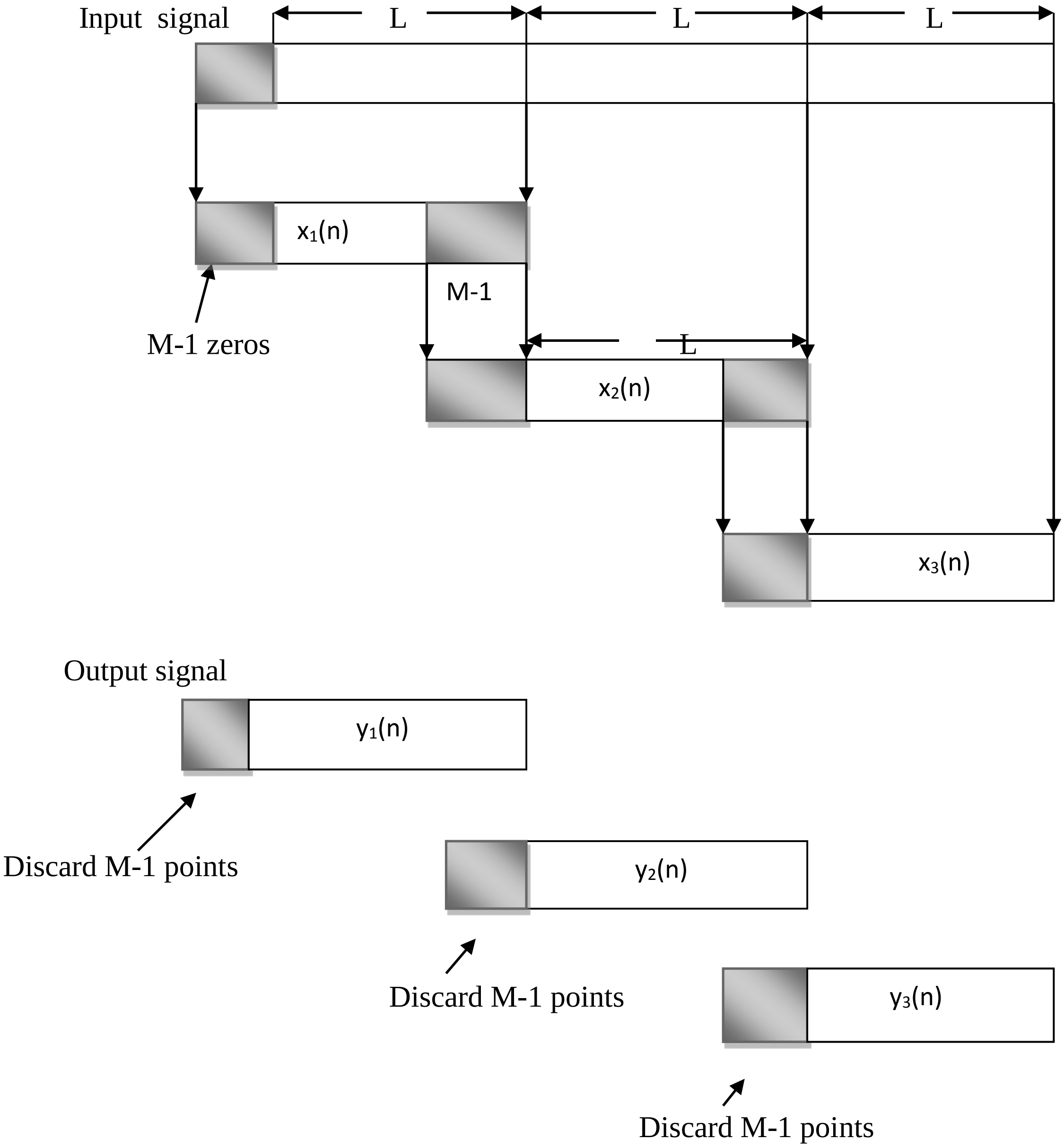
$$\hat{y}_m(n) = y_m(n)\,, \qquad n = M, M+1, \dots, N-1 \qquad \dots\dots\dots\dots(5.4.3)$$

To avoid loss of data due to aliasing, the last $M-1$ points of each data record are saved and these points become the first $M-1$ points of the subsequent record. To begin the processing, the first $M-1$ points of the first record are set to zero. Thus blocks of data sequences are:

$$x_1(n) = \left\{ \underbrace{0, 0, \dots, 0}_{M-1\ points}, x(0), x(1), \dots, x(L-1) \right\} \quad \dots\dots\dots.(5.4.4)$$

$$x_2(n) = \left\{ x\,\underbrace{(L-M+1), \dots, x(L-1)}_{M-1\ data\ points\ from\ x_1(n)}, \underbrace{x(L), \dots, x(2L-1)}_{L\ new\ data\ points} \right\} \quad \dots\dots\dots(5.4.5)$$

$$\Box_3(n) = \left\{ x\,\underbrace{(2L-M+1), \dots, x(2L-1)}_{M-1\ data\ points\ from\ x_2(n)}, \underbrace{x(2L), \dots, x(3L-1)}_{L\ new\ data\ points} \right\} \quad \dots\dots\dots(5.4.6)$$



(Linear FIR filtering by the overlap-save method)

Overlap-add method:

Size of input block = L

Size of the DFTs and IDFT is $N = L + M - 1$.

To each data block we append $M - 1$ zeros and compute the N-point DFT. The data blocks may be represented as

$$x_1(n) = \left\{x(0), x(1), \dots, x(L-1), \underbrace{0,0,\dots,0}_{M-1\ zeros}\right\} \quad \dots\dots\dots\dots (5.5.1)$$

$$x_2(n) = \left\{x(L), x(L-1), \dots, x(2L-1), \underbrace{0,0,\dots,0}_{M-1\ zeros}\right\} \quad \dots\dots\dots\dots (5.5.2)$$

$$x_3(n) = \left\{x(2L), \dots, x(3L-1), \underbrace{0,0,\dots,0}_{M-1\ zeros}\right\} \quad \dots\dots\dots\dots (5.5.3)$$

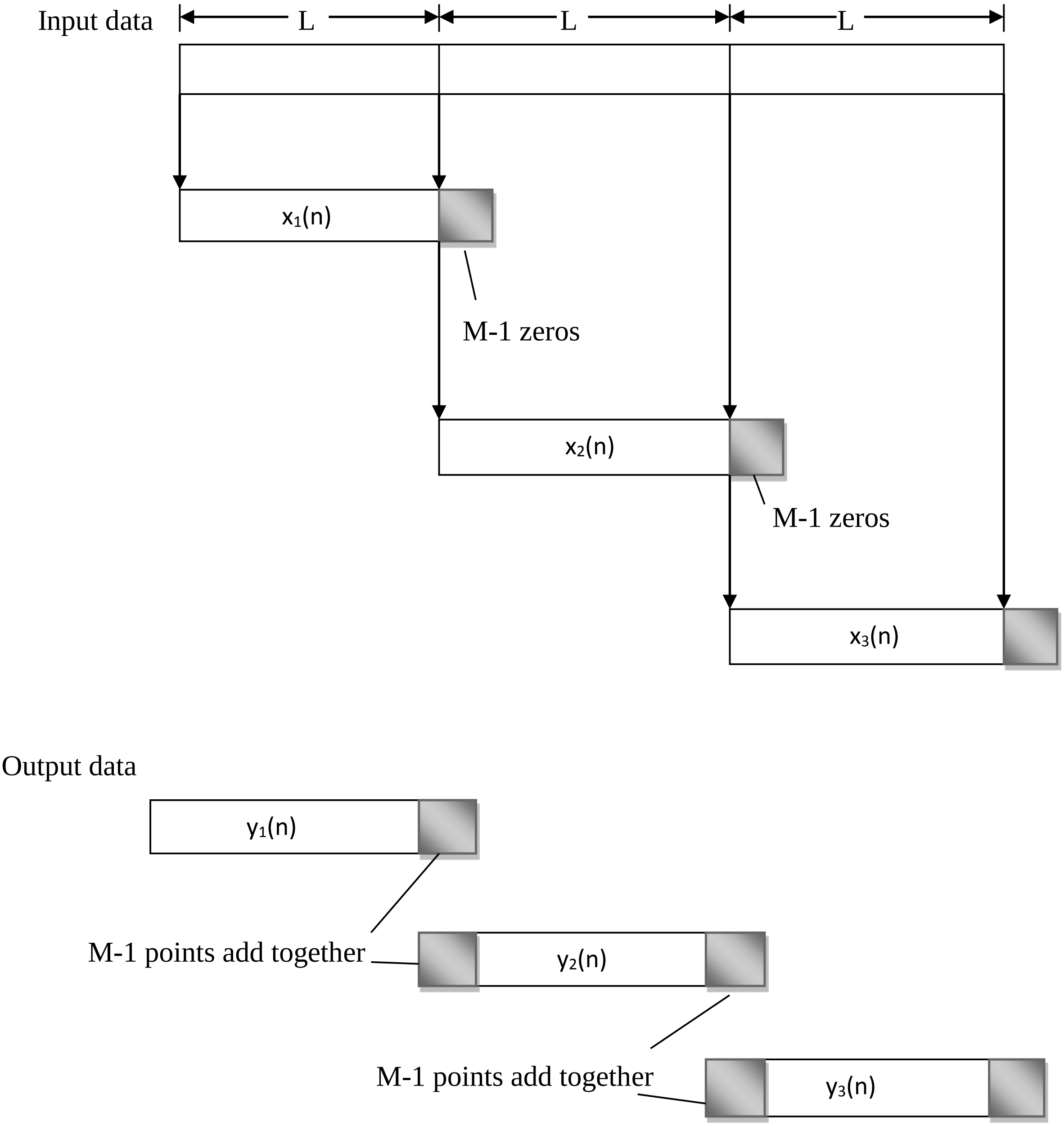
and so on. The two N-point DFTs are multiplied together to form

$$Y_m(k) = H(k)X_m(k), \qquad k = 0,1,\dots,N-1 \qquad \dots\dots..(5.5.4)$$

The IDFT yields data blocks of length N that are free of aliasing, since the size of the DFTs and IDFT is $N = L + M - 1$ and the sequences are increased to N-points by appending zeros to each block.

Since each data block is terminated with M-1 zeros , the last M-1 points from each output block must be overlapped and added to the first M-1 points of the succeeding block. Hence this method is called the overlap-add method. The output sequence is:

$$y(n) = \{y_1(0), y_1(1), \dots, y_1(L-1), y_1(L)+y_2(0), y_1(L+1)+y_2(1), \dots, y_1(N-1)+y_2(M-1), y_2(M), \dots\} \quad \dots\dots\dots\dots\dots..(5.5.5)$$



(Linear FIR filtering by the overlap-add method)

The Discrete Cosine Transform :

Forward DCT:

Let an N-point sequence x(n) which is real and even, that is,

$x(n) = x(N-n), 0 \leq n \leq N-1$

Let s(n) be a 2N-point even symmetric extension of x(n) defined by

$$s(n)=\begin{cases} x(n), & 0\le n\le N-1 \\ x(2N-n-1), & N\le n\le 2N-1 \end{cases} \quad \ldots\ldots\ldots\ldots.(6.1)$$

The DCT of x(n) can be computed by taking the 2N-point DFT of s(n) and multiplying the result by $W_{2N}^{k/2}$. The forward DCT is defined by

$$V(k)=2\sum_{n=0}^{N-1}x(n)\cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right], \quad 0\le k\le N-1 \quad \ldots\ldots\ldots\ldots\ldots\ldots..(6.2)$$

Inverse DCT

$$x(n)=\frac{1}{N}\left\{\frac{V(0)}{2}+\sum_{k=1}^{N-1}V(k)\cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]\right\}, \quad 0\le n\le N-1 \quad \ldots\ldots..(6.3)$$

DCT as an Orthogonal Transform

The $N\times N$ DCT matrix C_N of the sequence $x(n),\ 0\le n\le N-1$ is a real orthogonal matrix, that is, it satisfies

$$C_N^{-1}=C_N^{T} \quad \ldots\ldots\ldots..(6.4)$$

Orthogonality simplifies the computation of the inverse transform because it replaces matrix inversion by matrix transposition.

<u>Circular Correlation</u> :

If x(n) and y(n) are two periodic sequences, each with period N, then their cross correlation sequence is defined as

$$r_{xy}(l)=\frac{1}{N}\sum_{n=0}^{N-1}x(n)y(n-l) \quad \ldots\ldots\ldots\ldots(7.1)$$

Module-III

Fast Fourier Transform Algorithms:

1. Introduction

For a finite-duration sequence *x(n)* of length *N*, the DFT sum may be written as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}\ , k = 0,1,\dots,N-1$$

Where $W_N = e^{-j2\pi/N}$. There are a total of *N* values of *X(.)* ranging from *X(0)* to *X(N–1)*. The calculation of *X(0)* involves no multiplications at all since every product term involves $W_N^0 = e^{-j0} = 1$. Further, the first term in the sum always involves W_N^0 or $e^{-j0} = 1$ and therefore does not require a multiplication. Each *X(.)* calculation other than *X(0)* thus involves (*N-1*) complex multiplications. And each *X(.)* involves (*N–1*) complex additions. Since there are *N* values of *X(.)* the overall DFT requires (N-1)^2 complex multiplications and *N(N-1)* complex additions. For large *N* we may round these off to N^2 complex multiplications and the same number of complex additions.

Each complex multiplication is of the form

(A + jB) (C + jD) = (AC – BD) + j(BC + AD)

and therefore requires four real multiplications and two real additions. Each complex addition is of the form

(A + jB) + (C + jD) = (A + C) + j(B + D)

and requires two real additions. Thus the computation of all N values of the DFT requires 4N^2 real multiplications and 4N^2 (= 2N^2 + 2N^2) real additions. Efficient algorithms which reduce the number of multiply-and-add operations are known by the name of **fast Fourier transform** (FFT). The Cooley-Tukey and Sande-Tukey FFT algorithms exploit the following properties of the **twiddle factor** (phase factor), $W_N = e^{-j2\pi/N}$ (the factor $e^{-j2\pi/N}$ is called the *N*th principal root of 1):

1. Symmetry property $W_N^{k+N/2} = -W_N^k$
2. Periodicity property $W_N^{k+N} = W_N^k$

To illustrate, for the case of N = 8, these properties result in the following relations:

$W_8^0 = -W_8^4 = 1 \quad W_8^1 = -W_8^5 = \frac{1-j}{\sqrt{2}}$

$W_8^2 = -W_8^6 = -j \quad W_8^3 = -W_8^7 = -\frac{1+j}{\sqrt{2}}$

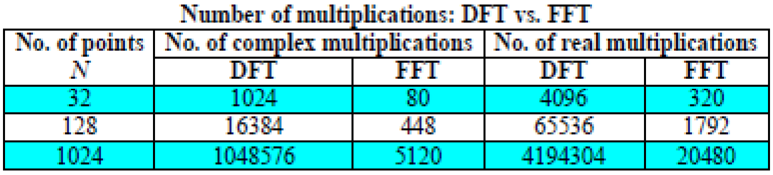
The use of these properties reduces the number of complex multiplications from N^2 to $\frac{N}{2}\log_2 N$ (actually the number of multiplications is less than this because several of the multiplications by W_N^r are really multiplications by ±1 or $\pm j$ and don't count); and the number of complex additions are reduced from $N^{\,2}$ to $N \log_2 N$. Thus, with each complex multiplication requiring four real multiplications and two real additions and each complex addition requiring two real additions, the computation of all N values of the DFT requires

Number of real multiplications = $4\left(\frac{N}{2}\log_2 N\right) = 2N\log_2 N$

Number of real additions = $2N\log_2 N + 2\left(\frac{N}{2}\log_2 N\right) = 3N\log_2 N$

We can get a rough comparison of the speed advantage of an FFT over a DFT by computing the number of multiplications for each since these are usually more time consuming than additions. For instance, for N = 8 the DFT, using the above formula, would need 82 = 64 complex multiplications, but the radix-2 FFT requires only $12\left(=\frac{8}{2}\log_2 8 = 4\times 3\right)$.

Number of multiplications: DFT vs. FFT



No. of points	No. of complex multiplications		No. of real multiplications	
N	DFT	FFT	DFT	FFT
32	1024	80	4096	320
128	16384	448	65536	1792
1024	1048576	5120	4194304	20480

We consider first the case where the length N of the sequence is an integral power of 2, that is, $N=2^v$ where v is an integer. These are called **radix-2 algorithms** of which the **decimation-in-time (DIT)** version is also known as the **Cooley-Tukey algorithm** and the **decimation-in-frequency (DIF)** version is also known as the **Sande-Tukey algorithm**. We show first how the algorithms work; their derivation is given later. For a radix of (r = 2), the **elementary computation** (*EC*) known as the **butterfly** consists of a single complex multiplication and two complex additions.

If the number of points, N, can be expressed as $N = r^{\,m}$, and if the computation algorithm is carried out by means of a succession of r-point transforms, the resultant FFT is called a **radix*r* algorithm**. In a radix-r FFT, an elementary computation consists of an r-point DFT followed by the multiplication of the r results by the appropriate twiddle factor. The number of *EC*s required is

$$C_r = \frac{N}{r}\log_r N$$

which decreases as r increases. Of course, the complexity of an *EC* increases with increasing r. For r = 4, the *EC* requires three complex multiplications and several complex additions.

Suppose that we desire an N-point DFT where N is a composite number that can be factored into the product of integers

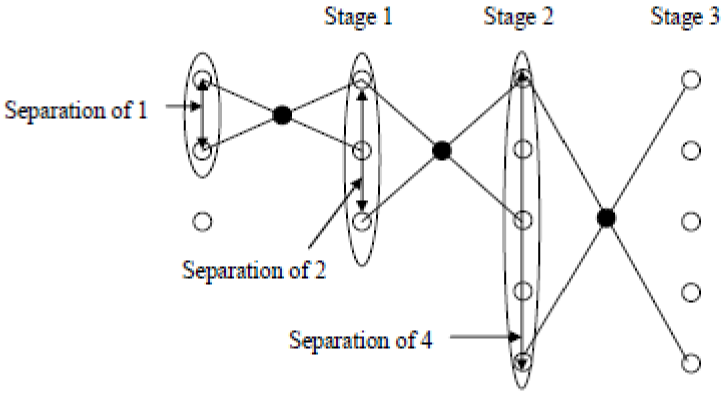
$N = N1\ N2 \dots Nm$

If, for instance, N = 64 and m = 3, we might factor N into the product 64 = 4 x 4 x 4, and the 64-point transform can be viewed as a three-dimensional 4 x 4 x 4 transform. If N is a prime number so that factorization of N is not possible, the original signal can be *zero-padded* and the resulting new composite number of points can be factored.

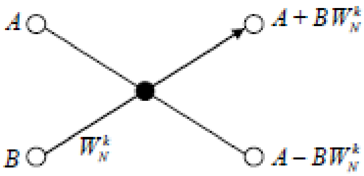
2. Radix-2 decimation-in-time FFT (Cooley-Tukey)

Procedure and important points

1. The number of input samples is $N = 2^v$ where v is an integer.
2. The input sequence is shuffled through bit-reversal. The index n of the sequence $x(n)$ is expressed in binary and then reversed.
3. The number of stages in the flow graph is given by $v = \log_2 N$.
4. Each stage consists of $N/2$ butterflies.
5. Inputs/outputs for each butterfly are separated as follows:
   Separation = 2^{m-1} samples where m = stage index, stages being numbered from left to right (that is, m = 1 for stage 1, m = 2 for stage 2 etc.). This amounts to separation increasing from left to right in the order 1, 2, 4… $N/2$.



6. The number of complex additions = $N \log_2 N$ and the number of complex multiplications $\frac{N}{2} \log_2 N$.
7. The elementary computation block in the flow graph, called the butterfly, is shown here. This is an **in-place calculation** in that the outputs ($A + B\,W_N^k$) and ($A - B\,W_N^k$) can be computed and stored in the same locations as A and B.



Example 1 Radix-2, 8-point, decimation-in-time FFT for the sequence

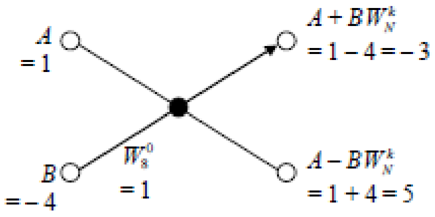
n→ 0 1 2 3 4 5 6 7
x(n) = {1, 2 3 4 –4 –3 –2 –1}

Solution The twiddle factors are

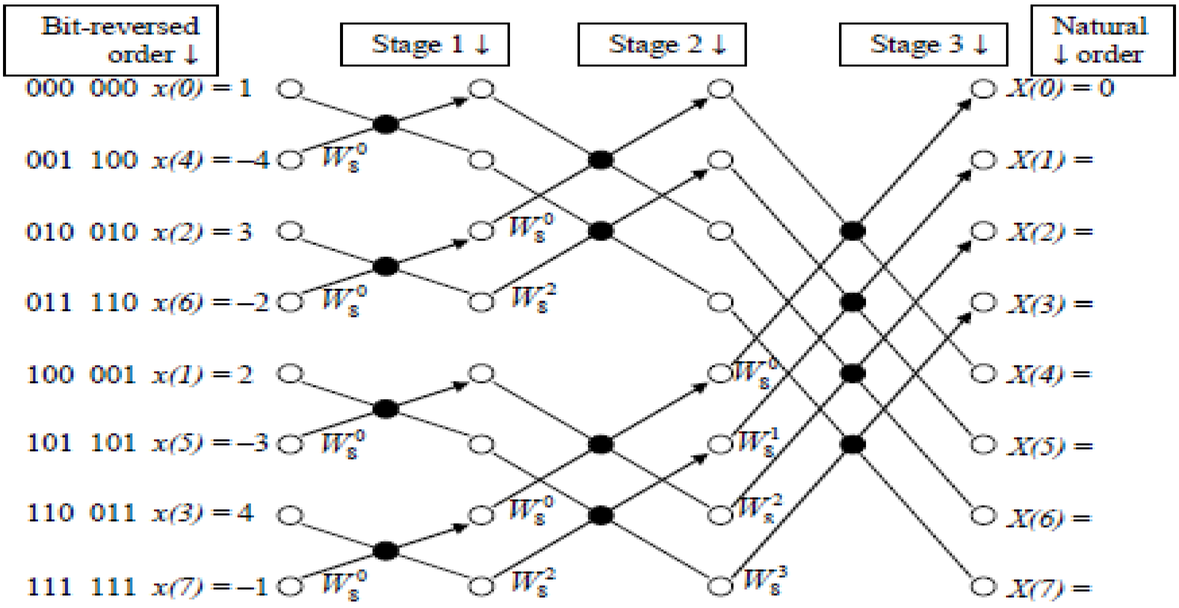
$W_8^0 = 1$ $\quad\quad W_8^1 = e^{-j2\pi/8} = e^{-j\pi/4} = \frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}$

$W_8^2 = \left(e^{-j2\pi/8}\right)^2 = e^{-j\pi/2} = -j$ $\quad\quad W_8^3 = \left(e^{-j2\pi/8}\right)^3 = e^{-j3\pi/4} = -\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}$

One of the elementary computations is shown below:



The signal flow graph follows:



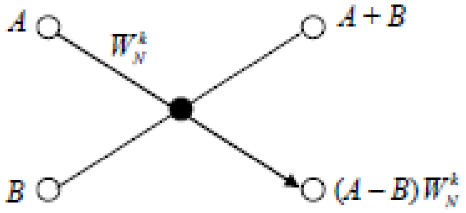
The DFT is

X(k) = {0, (5 – j12.07), (–4 + j4), (5 – j2.07), –4, (5 + j2.07), (–4 – j4), (5 + j12.07)}

3. Radix-2 decimation-in-frequency FFT (Sande-Tukey)

Procedure and important points

1. The number of input samples is $N = 2^v$ where v is an integer.
2. The input sequence is in natural order; the output is in bit-reversed order.
3. The number of stages in the flow graph is given by $v = \log_2 N$.
4. Each stage consists of $N/2$ butterflies.
5. Inputs/outputs for each butterfly are separated in the reverse order from that of the DIT. The separation decreases *from left to right* in the order $N/2, \ldots, 4, 2, 1$.
6. The number of complex additions = $N \log_2 N$ and the number of complex multiplications is $\frac{N}{2}\log_2 N$.
7. The basic computation block in the flow graph of the DIF FFT is the butterfly shown here. This is an **in-place calculation** in that the two outputs $(A + B)$ and $(A - B)\, W_N^k$ can be computed and stored in the same locations as A and B.



Example 2: Radix-2, 8-point, decimation-in-frequency FFT for the sequence

$n\rightarrow$ 0 1 2 3 4 5 6 7

$x(n)$ = {1, 2 3 4 –4 –3 –2 –1}

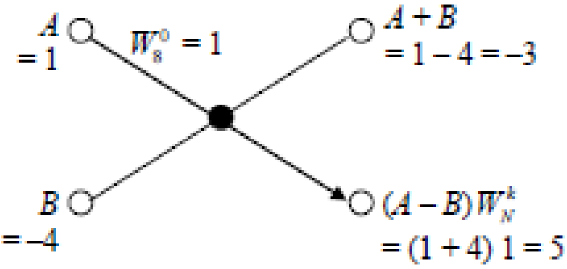
Solution :

The twiddle factors are the same as in the DIT FFT done earlier (both being 8-point DFTs):

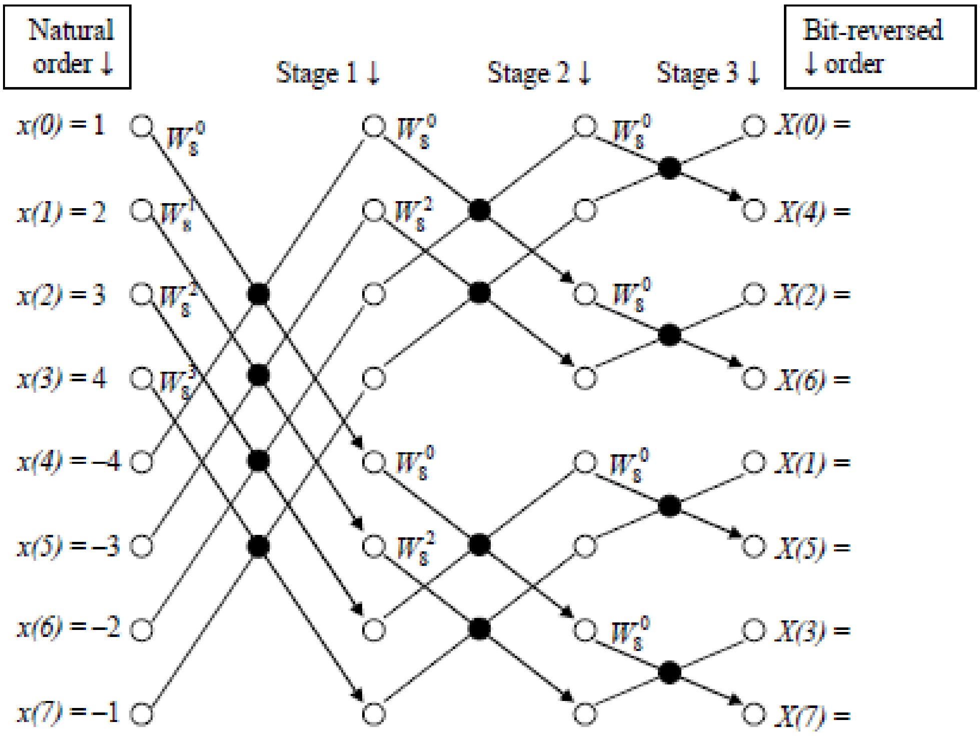
$W_8^0 = 1$ $\qquad W_8^1 = e^{-j2\pi/8} = e^{-j\pi/4} = \frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}$

$W_8^2 = \left(e^{-j2\pi/8}\right)^2 = e^{-j\pi/2} = -j$ $\qquad W_8^3 = \left(e^{-j2\pi/8}\right)^3 = e^{-j3\pi/4} = -\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}$

One of the elementary computations is shown below:



The signal flow graph follows:

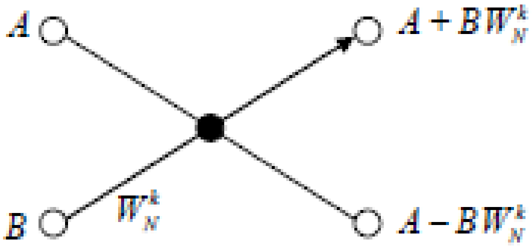


The DFT is

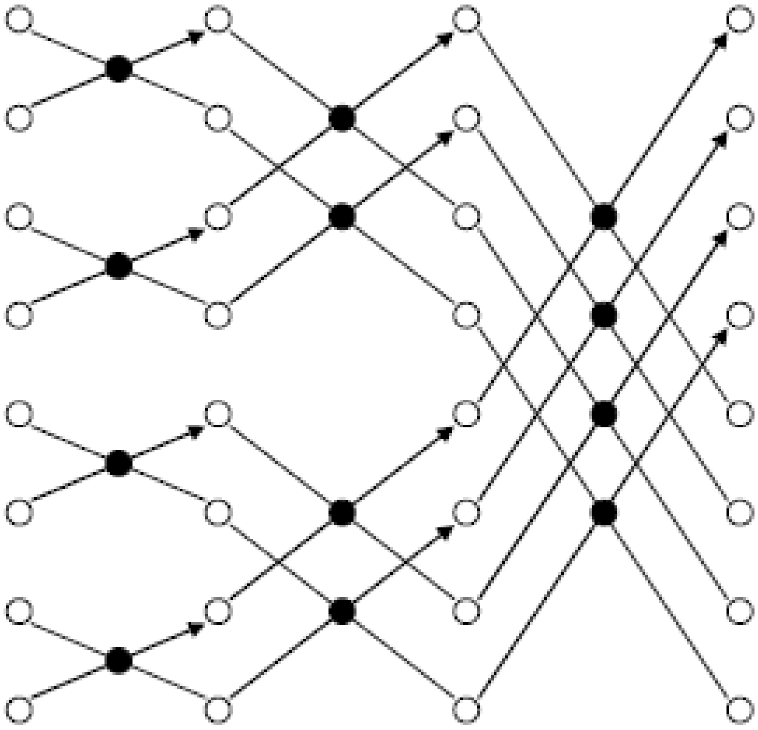
$X(k)$ = {0, (5 – j12.07), (–4 + j4), (5 – j2.07), –4, (5 + j2.07), (–4 – j4), (5 + j12.07)}

(DIT Template)

The elementary computation (Butterfly):

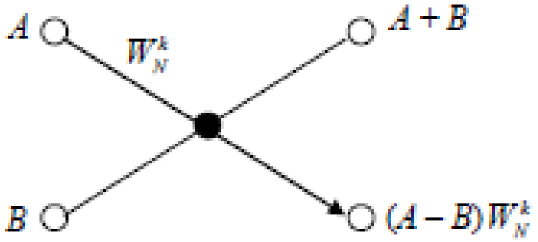


The signal flow graph:

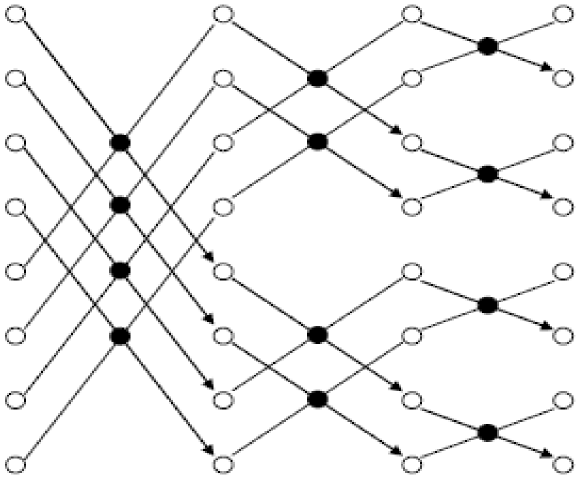


(DIF Template)

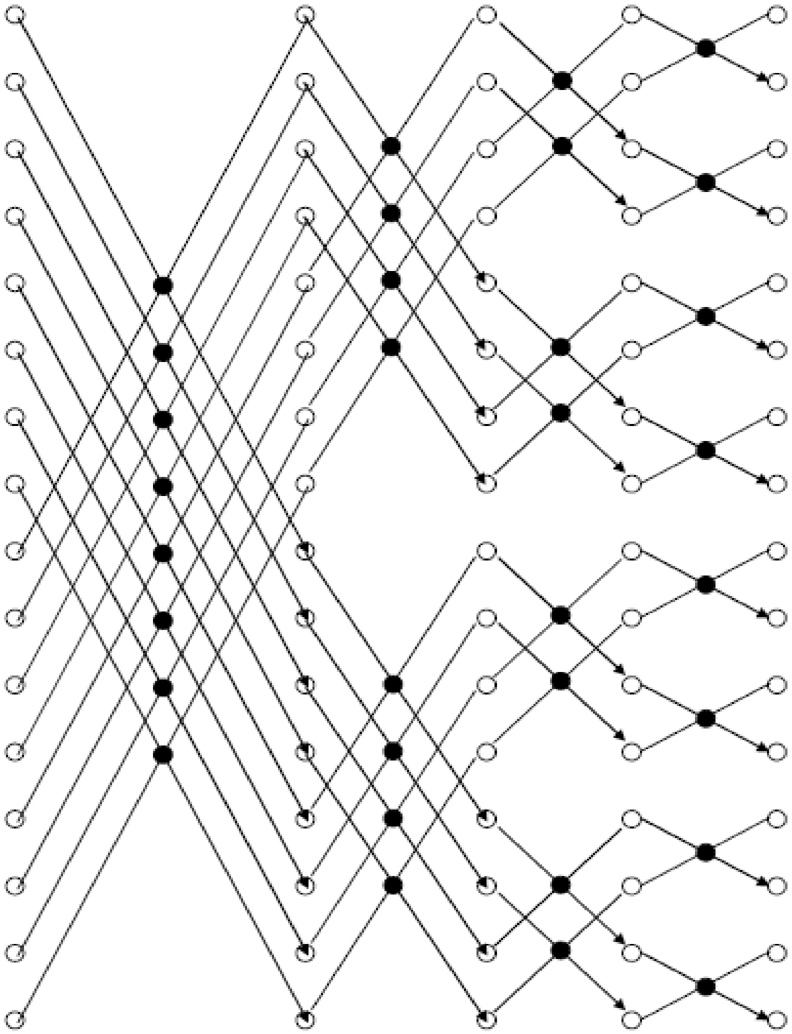
The elementary computation (Butterfly):



The signal flow graph:



16-point DIF FFT



4. Inverse DFT using the FFT algorithm

The inverse DFT of an *N*-point sequence {*X(k)*, *k* = 1, 2, … , (*N*–1)} is defined as

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)W_N^{-kn}\,, \qquad n = 0,1,\dots,N-1$$

Where $W_N = e^{-j2\pi/N}$. Take the complex conjugate of *x(n)* and multiply by *N* to get

$$Nx^*(n) = \sum_{k=0}^{N-1} X^*(k)W_N^{kn}$$

The right hand side of the above equation is simply the DFT of the sequence $X^*(k)$ and can be computed by using any FFT algorithm. The desired output sequence is then found by taking the conjugate of the result and dividing by *N*

$$x(n) = \frac{1}{N}\left(\sum_{k=0}^{N-1} X^*(k)W_N^{kn}\right)^*$$

Example 3: Given the DFT sequence *X(k)* = {0, (–1–j), j, (2+j), 0, (2–j), –j, (–1+j)} obtain the IDFT *x(n)* using the DIF FFT algorithm.

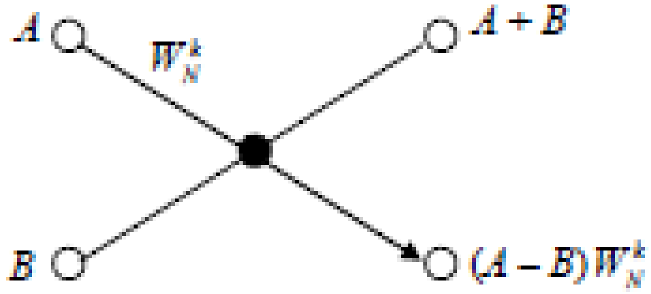
Solution:

This is an 8-point IDFT. The 8-point twiddle factors are, as calculated earlier,

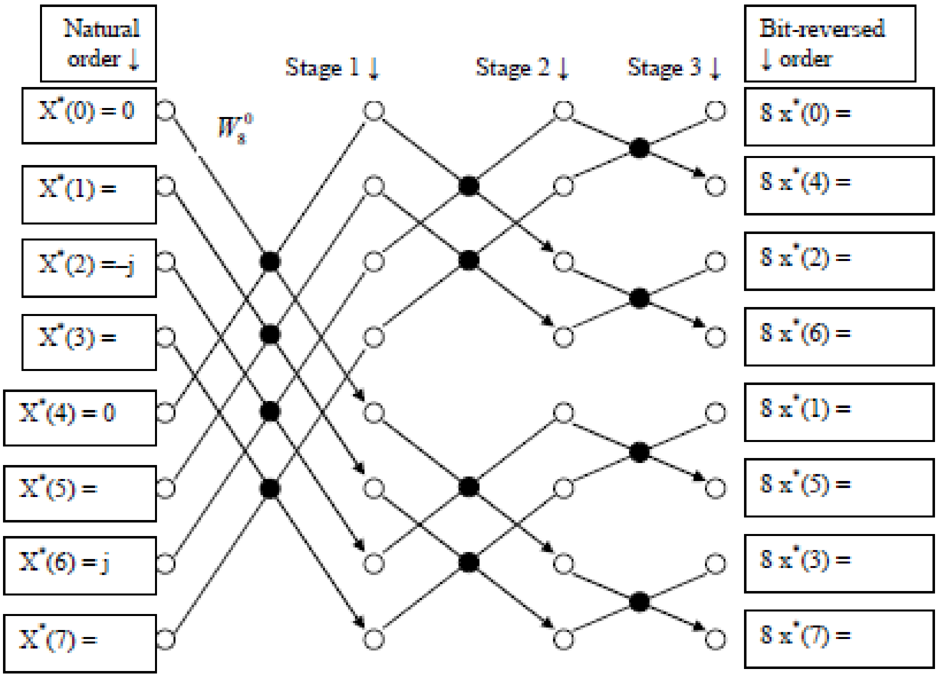
$$W_8^0 = 1 \qquad W_8^1 = e^{-j2\pi/8} = e^{-j\pi/4} = \tfrac{1}{\sqrt{2}} - j\tfrac{1}{\sqrt{2}}$$

$$W_8^2 = \left(e^{-j2\pi/8}\right)^2 = e^{-j\pi/2} = -j \qquad W_8^3 = \left(e^{-j2\pi/8}\right)^3 = e^{-j3\pi/4} = -\tfrac{1}{\sqrt{2}} - j\tfrac{1}{\sqrt{2}}$$

The elementary computation (Butterfly) is shown below:



The signal flow graph follows:



The output at stage 3 gives us the values $\{8x^*(n)\}$ in bit-reversed order:

$\{8x^*(n)\}_{bit\,rev\,order}$ = {2, –2, 4, –4, –6.24, 2.24, 6.24, –2.24}

The IDFT is given by arranging the data in normal order, taking the complex conjugate of the sequence and dividing by 8:

$\{8x^*(n)\}_{normalorder}$ = {2, –6.24, 4, 6.24, –2, 2.24, –4, –2.24}

$$x(n) = \left\{\frac{1}{4}, \frac{-6.24}{8}, \frac{1}{2}, \frac{6.24}{8}, \frac{1}{4}, \frac{2.24}{8}, -\frac{1}{2}, \frac{-2.24}{8}\right\}$$

$$x(n) = \{0.25, -0.78, 0.5, 0.78, -0.25, 0.28, -0.5, -0.28\}$$

Example 4: Given the DFT sequence *X(k)* = {0, (1–j), j, (2+j), 0, (2–j), (–1+j), –j}, obtain the IDFT *x(n)* using the DIF FFT algorithm.

Solution:

There is no conjugate symmetry in {*X(k)*}. Using MATLAB

```
X = [0, 1-1j, 1j, 2+1j, 0, 2-1j, -1+1j, -1j]
x = ifft(X)
```

The IDFT is

x(n) = {0.5, (-0.44 + 0.037i), (0.375 - 0.125i), (0.088 + 0.14i), (-0.75 + 0.5i), (0.44 + 0.21i), (-0.125 - 0.375i), (-0.088 - 0.39i)}

5. APPLICATIONS OF FFT ALGORITHMS:

1. Efficient Computation of the DFT of Two Real Sequences

The FFT algorithm is designed to perform complex multiplications and additions, even though the input data may be real valued. The basic reason for this situation is that the phase factors are complex and hence, after the first stage of the algorithm, all variables are basically complex-valued. In view of the fact that the algorithm can handle complex -valued input sequences, we can exploit this capability in the computation of the DFT of two real-valued sequences. Suppose that $x_1(n)$ and $x_2(n)$ are two real-valued sequences of length N, and let x(n) be a complex-valued sequence defined as

$$x(n) = x_1(n) + jx_2(n) \qquad 0 \le n \le N-1$$

The DFT operation is linear and hence the DFT of x(n) can be expressed as

$$X(k) = X_1(k) + jX_2(k)$$

The sequences $x_1(n)$ and $x_2(n)$can be expressed in terms of x(n) as follows:

$$x_1(n) = \frac{x(n) + x^*(n)}{2}$$

$$x_2(n) = \frac{x(n) - x^*(n)}{2j}$$

Hence the DFTs of $x_1(n)$ and $x_2(n)$ are

$$X_1(k) = \frac{1}{2}\{DFT[x(n)] + DFT[x^*(n)]\}$$

$$X_2(k) = \frac{1}{2j}\{DFT[x(n)] - DFT[x^*(n)]\}$$

Recall that the DFT of $x^*(n)$ is $X^*(N-k)$. Therefore

$$X_1(k) = \frac{1}{2}[X(k) + X^*(N-k)]$$

$$X_2(k) = \frac{1}{2j}[X(k) - X^*(N-k)]$$

Thus, by performing a single DFT on the complex-valued sequence x(n), we have obtained the DFT of the two real sequences with only a small amount of additional computation that is involved in computing $X_1(k)$ and $X_2(k)$from *X(k).*

2. Efficient Computation of the DFT of a 2N-Point Real Sequence

Suppose that g(n) is a real-valued sequence of 2N points. We now demonstrate how to obtain the 2N-point DFT of g(n) from computation of one N-point DFT involving complex-valued data. First, we define

$$x_1(n) = g(2n)$$

$$x_2(n) = g(2n+1)$$

Thus we have subdivided the 2N-point real sequence into two N-point real sequences. Now we can apply the method described in the preceding section.
Let x(n) be the N-point complex-valued sequence

$$x(n) = x_1(n) + jx_2(n)$$

From the results of the preceding section, we have

$$X_1(k) = \frac{1}{2}[X(k) + X^*(N-k)]$$

$$X_2(k) = \frac{1}{2j}[X(k) - X^*(N-k)]$$

Finally, we must express the 2N-point DFT in terms of the two N-point DFTs, $X_1(k)$ and $X_2(k)$. To accomplish this, we proceed as in the decimation-in-time FFT algorithm, namely,

$$G(k) = \sum_{n=0}^{N-1} g(2n)W_{2N}^{2nk} + \sum_{n=0}^{N-1} g(2n+1)W_{2N}^{(2n+1)k}$$
$$= \sum_{n=0}^{N-1} x_1(n)W_N^{nk} + W_{2N}^{k}\sum_{n=0}^{N-1} x_2(n)W_N^{nk}$$

Consequently,

$$G(k) = X_1(k) + W_2^kNX_2(k) \qquad k = 0,1,\dots.,N-1$$

$$G(k+N) = X_1(k) - W_2^kNX_2(k) \qquad k = 0,1,\dots.,N-1$$

Thus we have computed the DFT of a 2N-point real sequence from one N-point DFT and some additional computation.

6. The Chirp-z Transform Algorithm:

The DFT of an N-point data sequence *x(n)* has been viewed as the z-transform of $x_1(n)$ evaluated at N equally spaced points on the unit circle. It has also been viewed as N equally spaced samples of the Fourier transform of the data sequence *x(n).* In this section we consider the evaluation of *X(z)* on other contours in the z-plane, including the unit circle.

Suppose that we wish to compute the values of the z-transform of *x(n)* at a set of points {z_k}. Then,

$$X(\mathrm{z_k}) = \sum_{n=0}^{N-1} x(n) z_k^{-n} \qquad k = 0,1,\dots,L-1$$

For example, if the contour is a circle of radius r and the z_k are N equally spaced points, then

$$z_k = re^{j2\pi kn/N} \qquad k = 0,1,2,\dots\, ,N-1$$

$$X(\mathrm{z_k}) = \sum_{n=0}^{N-1} [x(n)r^{-n}] e^{-j2\pi kn/N} \qquad k = 0,1,2,\dots,N-1$$

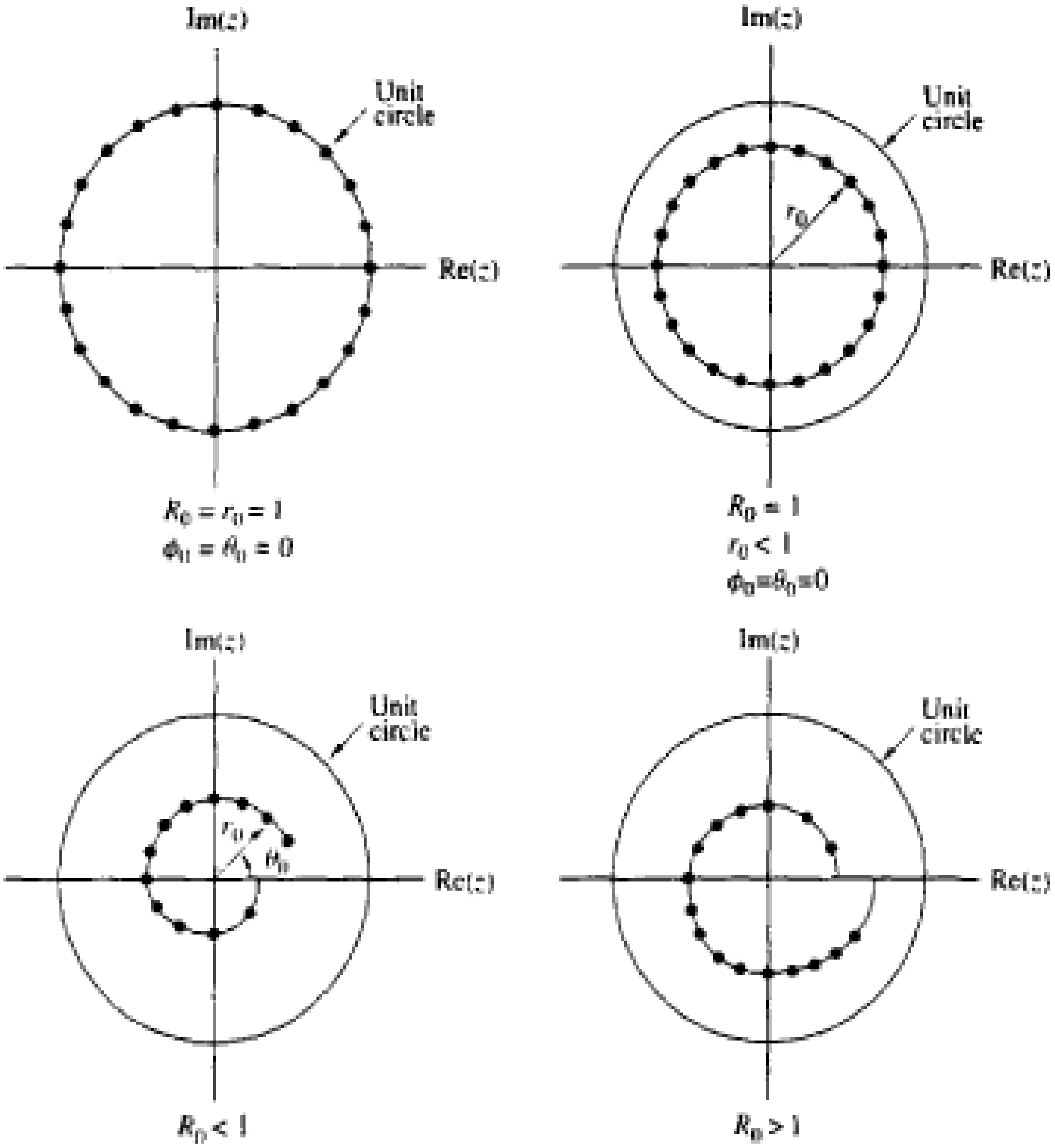
In this case the FFT algorithm can be applied on the modified sequence $x(n)r^{-n}$.
More generally, suppose that the points z_k in the z-plane fall on an arc which begins at some point

$$\mathrm{z_0} = \mathrm{r_0} e^{j\theta_0}$$

and spirals either in toward the origin or out away from the origin such that the points z_k are defined as

$$\mathrm{z_k} = \mathrm{r_0} e^{j\theta_0} \left(\mathrm{R_0} e^{j\phi_0}\right)^k \qquad k = 0,1,\dots,L-1$$

Note that if $R_0 < 1$, the points fall on a contour that spirals toward the origin and if $R_0 > 1$, the contour spirals away from the origin. If Ro = 1, the contour is a circular arc of radius r_0. If $r_0 = 1$ and $R_0 = l$, the contour is an arc of the unit circle. The latter contour would allow us to compute the frequency content of the sequence x(n) at a dense set of L frequencies in the range covered by the arc without having to compute a large DFT, that is, a DFT of the sequence x(n) padded with many zeros to obtain the desired resolution in frequency. Finally, if $r_0 = R_0 = 1$, = 0, $\Theta_0 = 0$, $\phi_0 = 2n / N$, and L = N, the contour is the entire unit circle and the frequencies are those of the DFT.



When points {z$_k$} are substituted into the expression for the z transform, we obtain

$$X(\mathrm{z_k}) = \sum_{n=0}^{N-1} x(n) z_k^{-n}$$

$$= \sum_{n=0}^{N-1} x(n) \left(\mathrm{r}_0 e^{j\theta_0}\right)^{-n} V^{-nk}$$

where, by definition, $V = \mathrm{R}_0 e^{j\phi_0}$

We can express the above equation in the form of a convolution, by noting that

$$nk = \frac{1}{2}[n^2 + k^2 - (k-n)^2]$$

$$X(\mathrm{z_k}) = V^{-k^2/2}\sum_{n=0}^{N-1}\left[x(n)\left(\mathrm{r}_0 e^{j\theta_0}\right)^{-n}V^{-n^2/2}\right]V^{(k-n)^2/2}$$

Let us define a new sequence g(n) as

$$g(n) = x(n)\left(\mathrm{r}_0 e^{j\theta_0}\right)^{-n}V^{-n^2/2}$$

Then,

$$X(\mathrm{z_k}) = V^{-k^2/2}\sum_{n=0}^{N-1} g(n)V^{(k-n)^2/2}$$

The summation in the above expression can be interpreted as the convolution of the sequence g(n) with the impulse response h(n) of a filter, where

$$h(n) = V^{n^2/2}$$

Hence,

$$X(\mathrm{z_k}) = V^{-k^2/2}y(k) = \frac{y(k)}{h(k)} \qquad k = 0,1,\dots,L-1$$

Where y(k) is the output of the filter

$$y(k) = \sum_{n=0}^{N-1} g(n)h(k-n) \qquad k = 0,1,\dots,L-1$$

We observe that both h(n) and g(n) are complex-valued sequences. The sequence h(n) with $R_0 = 1$ has the form of a complex exponential with argument $wn = n^2\phi_0/2 = (n\phi_0/2)n$. The quantity $n\phi_0/2$ represents the frequency of the complex exponential signal, which increases linearly with time. Such signals are used in radar systems and are called chirp signals. Hence the z-transform evaluated is called the chirp-z transform.

MODULE 4:

Structures for FIR and IIR Systems:

Structure for FIR Systems:

In general a FIR system is described by the difference equation

$$y(n) = \sum_{k=0}^{M-1} b_k x(n-k)$$

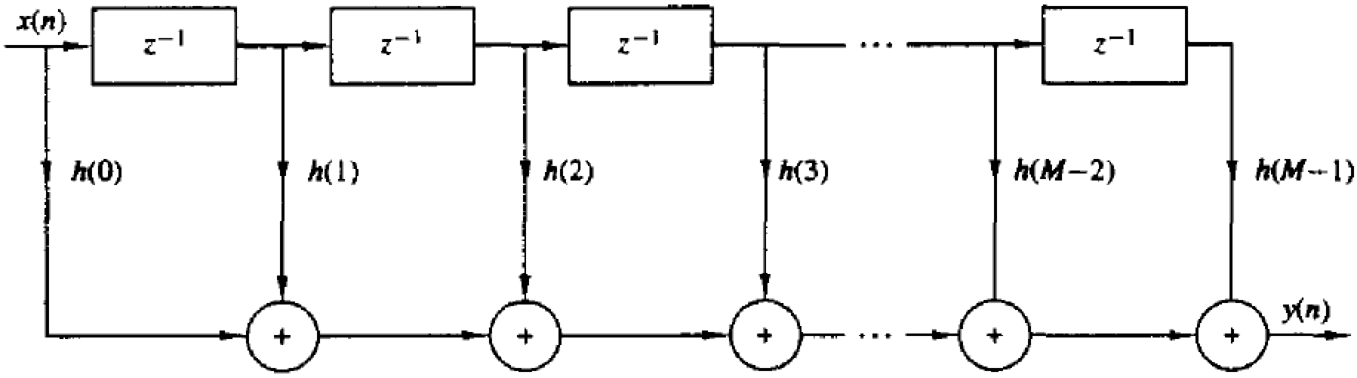
Or equivalently, by the system function

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k}$$

1. Direct-Form Structure:

The direct-form realization follows the convolution summation

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k)$$



Direct form realisation of FIR system

We observe that this structure requires M-1 memory locations for storing the M-1 previous inputs, and has a complexity of M multiplications and M-1 additions per output point. Since the output consists of a weighted linear combination of M-1 past values of the input and the weighted current value of the input, the structure in above figure, resembles a tapped delay line or a transversal system consequently, the direct-form realization is often called a transversal or tapped-delay-line filter.

2. Cascade-Form Structures:

The cascade realization follows naturally from the system function given by

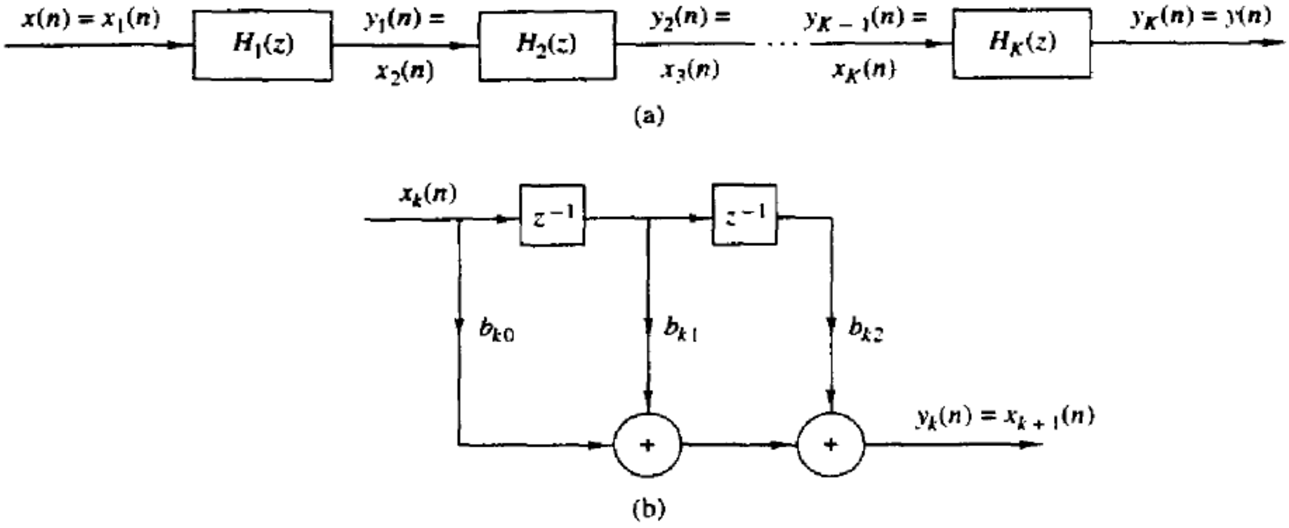
$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k}$$

It is simple matter to factor H(z) into second order FIR system so that

$$H(z) = \prod_{k=1}^{M} H_k(z)$$

Where H_k (z)=b_{k0}+$b_{k1}z^{-1}$+$b_{k2}z^{-2}$, k=1,2,3………..k

And K is the integer part of (M + l) /2. The filter parameter b_0 may be equally distributed among the K filter sections, such that $b_0 = b_{10}b_{20}\cdots b_{K0}$ or it may be assigned to a single filter section. The zeros of H (z) are grouped in pairs to produce the second-order FIR systems. It is always desirable to form pairs of complex-conjugate roots so that the coefficients {b_{ki}} are real valued. On the other hand, real-valued roots can be paired in any arbitrary manner. The cascade-form realization along with the basic second-order section is shown below.



Cascade Realisation of a FIR system

Design of Digital Filters:

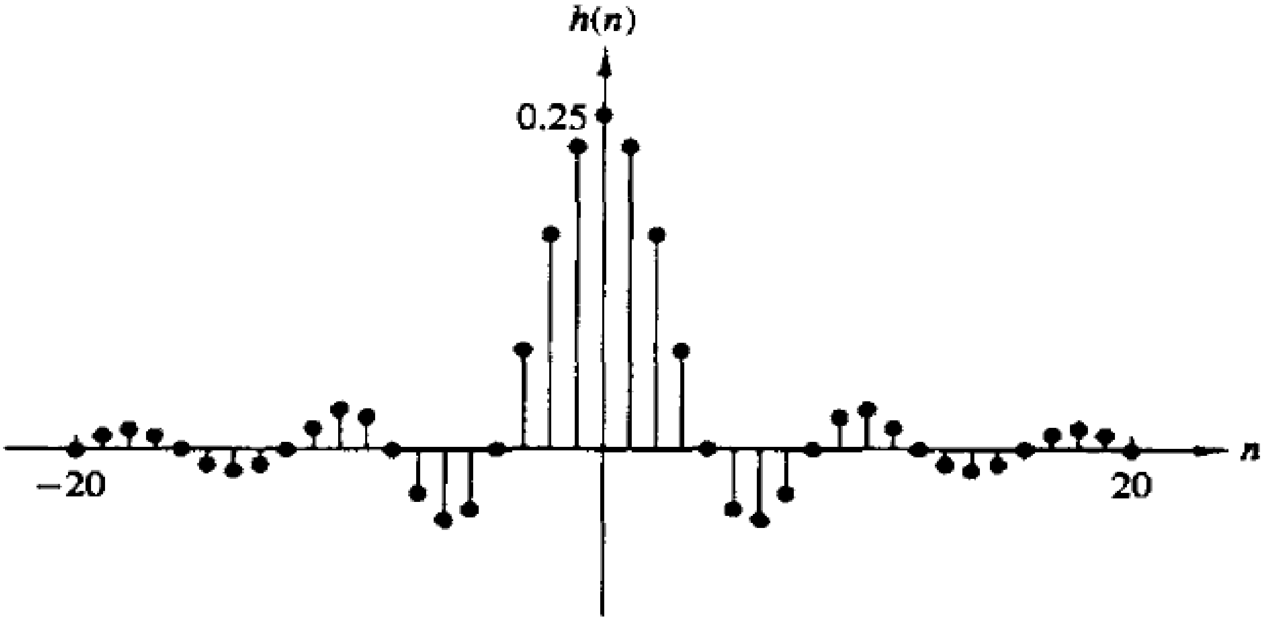
Causality and Its Implications:

Let us consider the issue of causality in more detail by examining the impulse response h(n) of an ideal low pass filter with frequency response characteristic

$$H(w)=\{ \begin{array}{ll} 1 & |\omega| \leq \omega_c \\ 0 & \omega_c < \omega < \pi \end{array}$$

The impulse response of the filter is

$$h(n)=\{ \begin{array}{l} \frac{\omega_c}{\pi}, n = 0 \\ \frac{\omega_c}{\pi}\frac{\sin \omega_c n}{\omega_c n}, n \neq 0 \end{array}$$



Unit sample response of an ideal low pass filter

A plot of h{n) for $w_c = \pi/ 4$ is illustrated in the above figure. It is clear that the ideal low pass filter is noncausal and hence it cannot be realized in practice.

One possible solution is to introduce a large delay n_0 in h(n) and arbitrarily to set h(n)=0 for $n < n_0$. However, the resulting system no longer has an ideal frequency response characteristic. Indeed, if we set h(n) = 0 for $n < n_0$, the Fourier series expansion of H(w) results in the Gibbs phenomenon.

Paley-Wiener Theorem:

If h(n) has finite energy and h(n) = 0 for n < 0, then

$$\int_{-\pi}^{\pi} |\ln|H(\omega)||d\omega < \infty$$

Conversely, if |H(ω)| is square integrable and if the integral in the above equation is finite, then we can associate with |H(ω)| a phase response $\Theta(\omega)$, so that the resulting filter with frequency response $H(\omega)=\left|H(\omega)\right| e^{j\theta(\omega)}$ is causal.

One important conclusion that we draw from the Paley-Wiener theorem is that the magnitude function |H(ω)| can be zero at some frequencies, but it can't be zero over any finite band of frequencies, since the integral then becomes infinite. Consequently any ideal filter is noncausal.

Apparently causalty imposes some tight constraints on a linear time invariant system. In addition to the Paley-Wiener condition causalty also implies a strong relation between $H_R(\omega)$ and $H_I(\omega)$, the real and imaginary components of the frequency response H(ω).To illustrate this dependence we decompose h(n).That iseven and an odd sequence, that is

$$H(n)=h_e(n)+h_o(n)$$

Where $h_e(n)=\frac{1}{2}[h(n)+h(-n)]$ and $\frac{1}{2}[h(n)-h(-n)]$

Now, if h(n) is causal ,it is possible to recover h(n) from its even part $h_e(n)$ for $0\leq n\leq\infty$ or from its odd component $h_o(n)$ for $1\leq n\leq\infty$.

Indeed, it can be easily seen that

$$h(n)=2h_e(n)u(n)-h_e(0)\delta(n) \qquad n\geq 0$$

and

$$h(n)=2h_o(n)u(n)-h_o(0)\delta(n) \qquad n\geq 1$$

Since h_0 (n) = 0 for n = 0, we cannot recover h(0) from h_0 (n) and hence we also must know h(0). In any case, it is apparent that h_0 (n) = h_e(n) for n > 1, so there is a strong relationship between h_0 (n) and h_e(n).

If h (n) is absolutely summable (i.e., BIBO stable), the frequency response H(w) exists, and

$$H(\omega) = H_R(\omega) + jH_I(\omega)$$

In addition, if h(n) is real valued and causal, the symmetry properties of the Fourier transform imply that

$$h_e(n) \overset{F}{\longleftrightarrow} H_R(\omega)$$

$$h_o(n) \overset{F}{\longleftrightarrow} H_I(\omega)$$

Since h(n) is completely specified by $h_e(n)$, it follows that H(ω) is completely determined if we know $H_R(\omega)$.alternatively H(ω) is completely determined from $H_I(\omega)$ and h(0).In short $H_R(\omega)$ and $H_I(\omega)$ are independent and cannot be specified independently if the system is causal. Equivalently the magnitude and phase responses of a causal filter are interdependent and hence cannot be specified independently.

Design of Linear Phase FIR filters using different windows:

In many cases a linear phase characteristics is required through the passband of the filter. It can be shown that causal IIR filter cannot produce a linear phase characteristics and only special forms of causal FIR filters can give linear phase. If {h[n]} represents the impulse response of a discrete time linear system a necessary and sufficient condition for linear phase is that {h[n]} have finite duration N, that it be symmetric about its midpoint, i.e.

$$h[n] = h[N-1-n], \quad n = 0, 1, 2, \ldots (N-1)$$

$$\begin{aligned} H(e^{jw}) &= \sum_{n=0}^{N-1} h[n]e^{-j\omega n} \\ &= \sum_{n=0}^{\frac{N}{2}-1} h[n]^{-j\omega n} + \sum_{n=N/2}^{N-1} h[n]e^{-j\omega n} \\ &= \sum_{n=0}^{N/2-1} h[n]e^{-j\omega n} + \sum_{m=0}^{N/2-1} h[m]e^{-j\omega}(N-1-m) \end{aligned}$$

For N even, we get

$$H(e^{j\omega}) = e^{-j\omega(N-1)/2} \sum_{n=0}^{N/2-1} 2h[N]\cos(\omega(n-(N-1)/2))$$

For N odd

$$H(e^{j\omega}) = e^{-j\omega(N-1)/2} \left\{ h[\frac{N-1}{2}] + \sum_{n=0}^{\frac{N-3}{2}} 2h[n]\cos[\omega(n-\frac{N-1}{2})] \right\}$$

For N even we get a non-integer delay, which will cause the value of the sequence to change.

One approach to design FIR filters linear phase is to use windows. The easiest way to obtain FIR filter is to simply truncate the impulse response of an IIR filter. If $\{h_d[n]\}$is the impulse response of the designed FIR filter then the fir filter with impulse response {h[n]} can be obtained as follows.

$$H[n]=\{ \begin{matrix} hd[n], N_1 \le n \le N_2 \\ 0, otherwise \end{matrix}$$

This can be thought of as being formed by a product of $\{h_d[n]\}$ and a window function {w[n]}

$\{h[n]\}= \{h_d[n]\}\ \{w[n]\}$

where {w[n]} is the window function.

Using modulation property of fourier transform

$$H(e^{j\omega})= \frac{1}{2\pi} [\ H_d(e^{j\omega}) \otimes -w(e^{j\omega})]$$

In general for smaller N values spreading of main lobe more, and for larger N narrower thr main lobe and $|H(e^{j\omega})|$ comes closer to $|H_d(e^{j\omega})|$.Much work has been done on adjusting {w[n]} to satisfy certain main lobe and side lobe requirements .Some of the commonly used windows are given below-

(a) Rectangular Window

$$W_R(n)=\{ \begin{matrix} 1, 0 \le n \le N-1 \\ 0, otherwise \end{matrix}$$

(b) Bartlett (Triangular)

$$W_B(n)=\{ \begin{matrix} \frac{2n}{N-1}, 0 \le n \le (N-1)/2 \\ 2-\frac{2n}{N-1}, (N-1)/2 \le n \le N-1 \\ 0, elsewhere \end{matrix}$$

(c) Hanning Window

$$W_{Han}(n)=\{\ \begin{matrix} \frac{1-\cos[2\pi n/(N-1)]}{2}, 0 \le n \le N-1 \\ 0, otherwise \end{matrix}$$

(d)Blackman Window

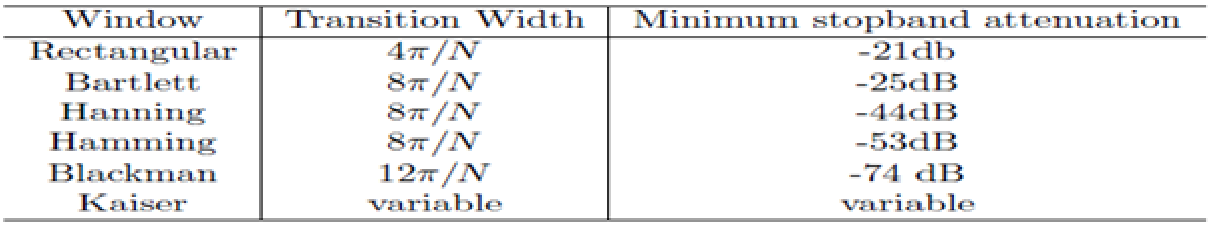
$$W_{Bl}(n)=\{\ \begin{matrix} .42-.5\cos\left[2\pi n \big/ N-1\right]+.08\cos\left[4\pi n \big/ N-1\right], 0 \le n \le N-1 \\ 0, otherwise \end{matrix}$$

(e)Kaiser Window

$$W_K(n)=\{\ \begin{matrix} \frac{I_0 \omega_a \left[\left(\frac{N-1}{2}\right)^2 - \left(n-\frac{N-1}{2}\right)^2\right]^{\frac{1}{2}}}{I_0\left\{w_a\left(\frac{N-1}{2}\right)\right\}}, 0 \le n \le N-1 \\ 0, otherwise \end{matrix}$$

Where $I_0(x)$ is the modified Zero Order Bessel Function of the first kind.

The Transition width and the minimum stopped attenuation for different windows are listed below-



Window	Transition Width	Minimum stopband attenuation
Rectangular	$4\pi/N$	-21db
Bartlett	$8\pi/N$	-25dB
Hanning	$8\pi/N$	-44dB
Hamming	$8\pi/N$	-53dB
Blackman	$12\pi/N$	-74 dB
Kaiser	variable	variable

We first choose a window that satisfies the minimum attenuation and the bandwidth that allows us to choose the appropriate value of N. Actual frequency response characteristics are then calculated and we check the requirments are met or not

<u>Design of IIR Filters:</u>

There are two methods for design the IIR filter.

1. Impulse Invariant Method

2. Bilinear Transformation Method

1. Filter design by impulse invariance:

Here the impulse response h[n] of the desire discrete time system is proportional to equally spaces samples of the continuous time filter i.e,

$H[n]=T_d h_a(nT_d)$

Where T_d represents a sample interval.Since the specification of the filter are given in discrete time domain it turns out that T_d has no role to play in design of the filter. From the sampling theorem the frequency response of the discrete time filter is given by

$$H(e^{j\omega})= \sum_{k=-\infty}^{\infty} H_a(j\frac{\omega}{T_d}+j\frac{2\pi k}{T_d})$$

Since any practical continuous time filter is not strictly band limited there is some aliasing. However if the continuous time filter approaches zero at high frequency the aliasing may be negligible. Then the frequency response of the discrete time filter is

$$H(e^{j\omega})\approx \sum_{k=-\infty}^{\infty} H_a(j\frac{\omega}{T_d}), |\omega|\le\pi$$ Type equation here.

We first convert digital filter specifications to continuous time filter specifications. Neglecting aliasing we get $H_a(j\Omega)$ specification by applying the relation $\Omega= \omega/T_d$. Where $H_a(j\Omega)$ is transferred to the designed filter H(z).

Let us assume that the poles of the continuous time filter are simple, then

$$H_a(s)= \sum_{k=1}^{N} \frac{A_k}{s-s_k}$$

The corresponding Impulse response is $h_a(t)=\{ \begin{array}{l} \sum_{k=1}^{N} A_k e^{s_k t}, t\ge 0 \\ 0, t<0 \end{array}$

Then $h[n]=T_d h_a(nT_d)= \sum_{k=1}^{N} T_d A_k e^{s_k n T_d} u[n]$

The system function function for this is $H(z)= \sum_{k=1}^{N} \frac{T_d A_k}{1-e^{s_k T_d} z^{-1}}$

We see that a pole at s= s_k in the s-plane is transferred to a pole at z= $e^{s_k T_d}$ in the z-plane. If the continuous time filter is stable i.e Re{s_k}<0, then the magnitude of $e^{s_k T_d}$ will be less than 1.So the pole will be inside the unit circle. Thus the causal discrete filter is stable. The mapping of zero is not so straight forward.

Bilinear Transformation:

This technique avoids the problem of aliasing by mapping jΩ axis in the s-plane to one revolution of unit circle in the z-plane. If $H_a(s)$ is the continuous time transfer function the discrete time transfer function is detained by replacing s with

$$S= \frac{2}{T_d}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)$$

From which we get z= $\frac{1+(T_d/2)s}{1-(T_d/2)s}$

Substituting s= σ+jΩ , we get $z=\frac{1+\sigma\frac{T_d}{2}+j\frac{\Omega T_d}{2}}{1-\sigma\frac{T_d}{2}-j\frac{\Omega T_d}{2}}$

If σ<0, it is then magnitude of the real part in the denominator is more than that of the numerator and so │z│<1. Similarly if σ>0 then │z│>1 for all Ω.Thus pole in the left half of the s-plane will get mapped to the poles inside the unit circle in z-plane. If σ=0 then

$$z=\frac{1+j\frac{\Omega T_d}{2}}{1-j\frac{\Omega T_d}{2}}$$

so │z│=1,writing z=$e^{j\omega}$ we get

$$e^{j\omega}=\frac{1+j\frac{\Omega T_d}{2}}{1-j\frac{\Omega T_d}{2}}$$

Rearranging we get $j\frac{\Omega T_d}{2}=\frac{e^{j\omega}-1}{e^{j\omega}-1}=\frac{e^{j\omega/2}(e^{+j\omega/2}-e^{-j\omega/2})}{e^{j\omega/2}(e^{+j\omega/2}+e^{-j\omega/2})}=j\frac{\sin\omega/2}{\cos\omega/2}$

Or $\Omega=\frac{2}{T_d}\tan\omega/2$ or $\omega=2\tan^{-1}\frac{\Omega T_d}{2}$.

GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

Microprocessor & Microcontroller

For 4th Semester

ELECTRONICS & TELECOMMUNICATION

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Mr. Manas Ranjan Biswal

(Lecturer in Electronics & Telecommunication Engineering)

MODULE: 1

1. INTRODUCTION TO MICROPROCESSOR AND MICROCOMPUTER ARCHITECTURE:

A *microprocessor* is a programmable electronics chip that has computing and decision making capabilities similar to central processing unit of a computer. Any microprocessor-based systems having limited number of resources are called *microcomputers*. Nowadays, microprocessor can be seen in almost all types of electronics devices like mobile phones, printers, washing machines etc. Microprocessors are also used in advanced applications like radars, satellites and flights. Due to the rapid advancements in electronic industry and large scale integration of devices results in a significant cost reduction and increase application of microprocessors and their derivatives.

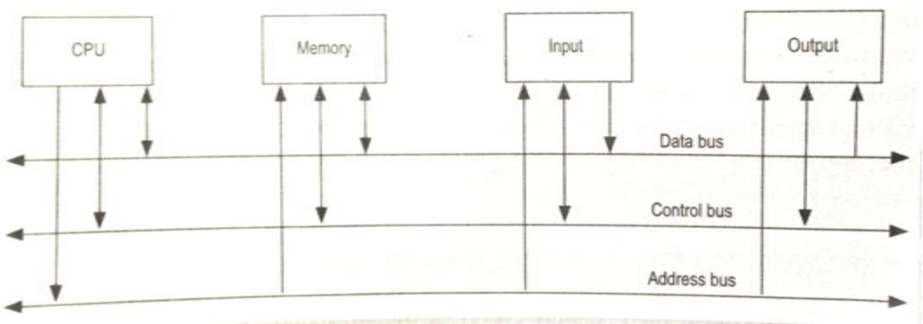


Fig.1 Microprocessor-based system

- **Bit**: A bit is a single binary digit.
- **Word**: A word refers to the basic data size or bit size that can be processed by the arithmetic and logic unit of the processor. A 16-bit binary number is called a word in a 16-bit processor.
- **Bus**: A bus is a group of wires/lines that carry similar information.
- **System Bus**: The system bus is a group of wires/lines used for communication between the microprocessor and peripherals.
- **Memory Word**: The number of bits that can be stored in a register or memory element is called a memory word.
- **Address Bus**: It carries the address, which is a unique binary pattern used to identify a memory location or an I/O port. For example, an eight bit address bus has eight lines and thus it can address 2^8 = 256 different locations. The locations in hexadecimal format can be written as 00H – FFH.
- **Data Bus**: The data bus is used to transfer data between memory and processor or between I/O device and processor. For example, an 8-bit processor will generally have an 8-bit data bus and a 16-bit processor will have 16-bit data bus.
- **Control Bus**: The control bus carry control signals, which consists of signals for selection of memory or I/O device from the given address, direction of data transfer and synchronization of data transfer in case of slow devices.

A typical microprocessor consists of arithmetic and logic unit (ALU) in association with control unit to process the instruction execution. Almost all the microprocessors are based on the principle of store-program concept. In *store-program concept*, programs or instructions are sequentially stored in the memory locations that are to be executed. To do any task using a microprocessor, it is to be programmed by the user. So the programmer must have idea about its internal resources, features and supported instructions. Each microprocessor has a set of instructions, a list which is provided by the microprocessor manufacturer. The instruction set of a microprocessor is provided in two forms: *binary machine code and mnemonics*.

Microprocessor communicates and operates in binary numbers 0 and 1. The set of instructions in the form of binary patterns is called a *machine language* and it is difficult for us to understand. Therefore, the binary patterns are given abbreviated names, called mnemonics, which forms the *assembly language*. The conversion of assembly-level language into binary machine-level language is done by using an application called *assembler.*

Technology Used:

The semiconductor manufacturing technologies used for chips are:

- Transistor-Transistor Logic (TTL)
- Emitter Coupled Logic (ECL)
- Complementary Metal-Oxide Semiconductor (CMOS)

Classification of Microprocessors:

Based on their specification, application and architecture microprocessors are classified.

Based on size of data bus:

- 4-bit microprocessor
- 8-bit microprocessor
- 16-bit microprocessor
- 32-bit microprocessor

Based on application:

- General-purpose microprocessor- used in general computer system and can be used by programmer for any application. Examples, 8085 to Intel Pentium.
- Microcontroller- microprocessor with built-in memory and ports and can be programmed for any generic control application. Example, 8051.
- Special-purpose processors- designed to handle special functions required for an application. Examples, digital signal processors and application-specific integrated circuit (ASIC) chips.

Based on architecture:

- Reduced Instruction Set Computer (RISC) processors
- Complex Instruction Set Computer (CISC) processors

2. 8085 MICROPROCESSOR ARCHITECTURE

The 8085 microprocessor is an 8-bit processor available as a 40-pin IC package and uses +5 V for power. It can run at a maximum frequency of 3 MHz. Its data bus width is 8-bit and address bus width is 16-bit, thus it can address 2^{16} = 64 KB of memory. The internal architecture of 8085 is shown is Fig. 2.

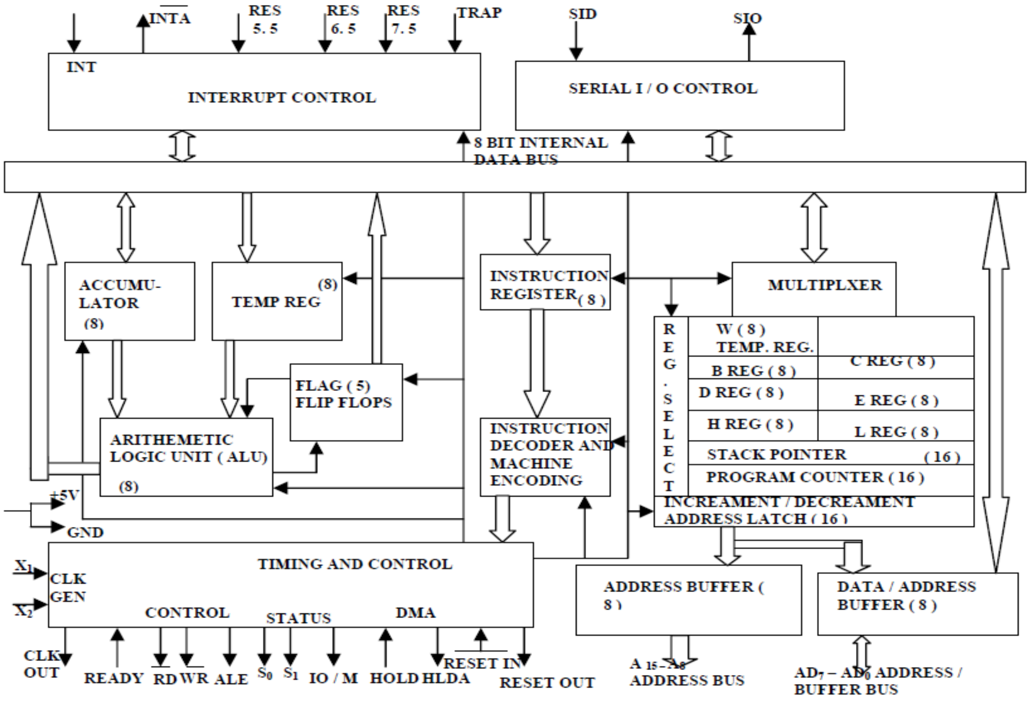


Fig. 2 Internal Architecture of 8085

Arithmetic and Logic Unit

The ALU performs the actual numerical and logical operations such as Addition (ADD), Subtraction (SUB), AND, OR etc. It uses data from memory and from Accumulator to perform operations. The results of the arithmetic and logical operations are stored in the accumulator.

Registers

The 8085 includes six registers, one accumulator and one flag register, as shown in Fig. 3. In addition, it has two 16-bit registers: stack pointer and program counter. They are briefly described as follows.

The 8085 has six general-purpose registers to store 8-bit data; these are identified as B, C, D, E, H and L. they can be combined as register pairs - BC, DE and HL to perform some

16-bit operations. The programmer can use these registers to store or copy data into the register by using data copy instructions.

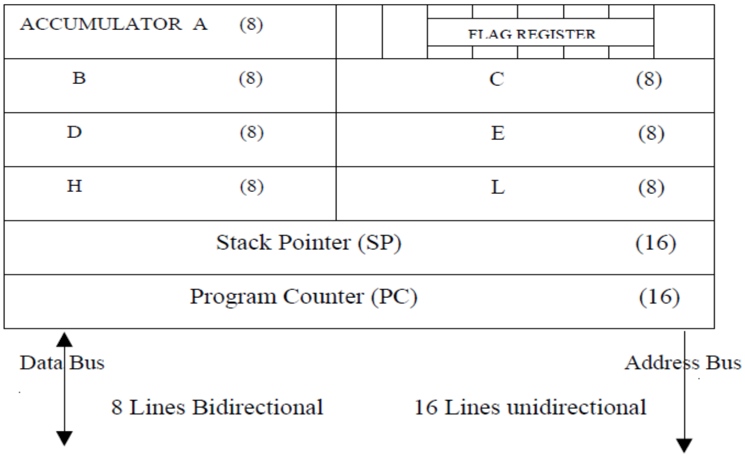


Fig. 3 Register organisation

Accumulator

The accumulator is an 8-bit register that is a part of ALU. This register is used to store 8-bit data and to perform arithmetic and logical operations. The result of an operation is stored in the accumulator. The accumulator is also identified as register A.

Flag register

The ALU includes five flip-flops, which are set or reset after an operation according to data condition of the result in the accumulator and other registers. They are called Zero (Z), Carry (CY), Sign (S), Parity (P) and Auxiliary Carry (AC) flags. Their bit positions in the flag register are shown in Fig. 4. The microprocessor uses these flags to test data conditions.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
S	**Z**		**AC**		**P**		**CY**

Fig. 4 Flag register

For example, after an addition of two numbers, if the result in the accumulator is larger than 8-bit, the flip-flop uses to indicate a carry by setting CY flag to 1. When an arithmetic operation results in zero, Z flag is set to 1. The S flag is just a copy of the bit D7 of the accumulator. A negative number has a 1 in bit D7 and a positive number has a 0 in 2's complement representation. The AC flag is set to 1, when a carry result from bit D3 and passes to bit D4. The P flag is set to 1, when the result in accumulator contains even number of 1s.

Program Counter (PC)

This 16-bit register deals with sequencing the execution of instructions. This register is a memory pointer. The microprocessor uses this register to sequence the execution of the instructions. The function of the program counter is to point to the memory address from which the next byte is to be fetched. When a byte is being fetched, the program counter is automatically incremented by one to point to the next memory location.

Stack Pointer (SP)

The stack pointer is also a 16-bit register, used as a memory pointer. It points to a memory location in R/W memory, called stack. The beginning of the stack is defined by loading 16-bit address in the stack pointer.

Instruction Register/Decoder

It is an 8-bit register that temporarily stores the current instruction of a program. Latest instruction sent here from memory prior to execution. Decoder then takes instruction and decodes or interprets the instruction. Decoded instruction then passed to next stage.

Control Unit

Generates signals on data bus, address bus and control bus within microprocessor to carry out the instruction, which has been decoded. Typical buses and their timing are described as follows:

- *Data Bus*: Data bus carries data in binary form between microprocessor and other external units such as memory. It is used to transmit data i.e. information, results of arithmetic etc between memory and the microprocessor. Data bus is bidirectional in nature. The data bus width of 8085 microprocessor is 8-bit i.e. 2^8 combination of binary digits and are typically identified as D0 – D7. Thus size of the data bus determines what arithmetic can be done. If only 8-bit wide then largest number is 11111111 (255 in decimal). Therefore, larger numbers have to be broken down into chunks of 255. This slows microprocessor.
- *Address Bus*: The address bus carries addresses and is one way bus from microprocessor to the memory or other devices. 8085 microprocessor contain 16-bit address bus and are generally identified as A0 - A15. The higher order address lines (A8 – A15) are unidirectional and the lower order lines (A0 – A7) are multiplexed (time-shared) with the eight data bits (D0 – D7) and hence, they are bidirectional.
- *Control Bus*: Control bus are various lines which have specific functions for coordinating and controlling microprocessor operations. The control bus carries control signals partly unidirectional and partly bidirectional. The following control and status signals are used by 8085 processor:
    I. ALE (output): Address Latch Enable is a pulse that is provided when an address appears on the AD0 – AD7 lines, after which it becomes 0.

II. $\overline{RD}$ (active low output): The Read signal indicates that data are being read from the selected I/O or memory device and that they are available on the data bus.

III. $\overline{WR}$ (active low output): The Write signal indicates that data on the data bus are to be written into a selected memory or I/O location.

IV. $IO/\overline{M}$ (output): It is a signal that distinguished between a memory operation and an I/O operation. When $IO/\overline{M} = 0$ it is a memory operation and $IO/\overline{M} = 1$ it is an I/O operation.

V. S1 and S0 (output): These are status signals used to specify the type of operation being performed; they are listed in Table 1.

Table 1 Status signals and associated operations

S1	S0	States
0	0	Halt
0	1	Write
1	0	Read
1	1	Fetch

The schematic representation of the 8085 bus structure is as shown in Fig. 5. The microprocessor performs primarily four operations:

I. Memory Read: Reads data (or instruction) from memory.
II. Memory Write: Writes data (or instruction) into memory.
III. I/O Read: Accepts data from input device.
IV. I/O Write: Sends data to output device.

The 8085 processor performs these functions using address bus, data bus and control bus as shown in Fig. 5.

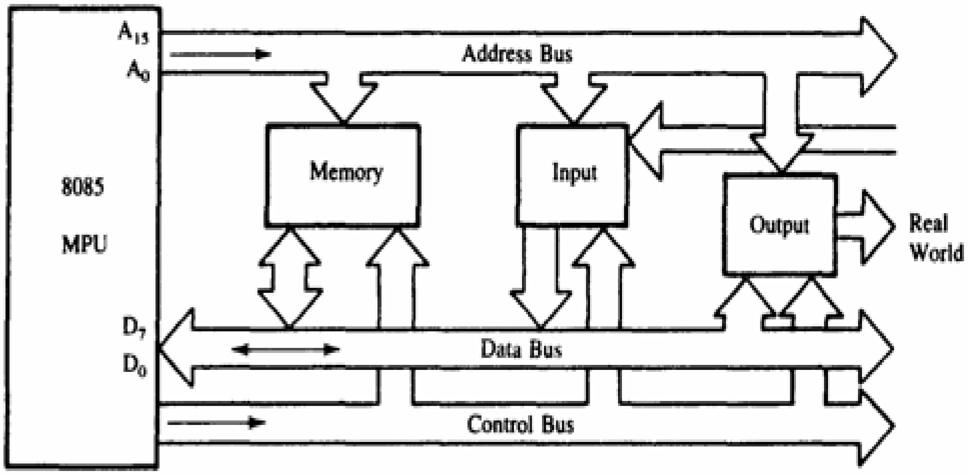


Fig. 5 The 8085 bus structure

3. 8085 PIN DESCRIPTION

Properties:

- It is a 8-bit microprocessor
- Manufactured with N-MOS technology
- 40 pin IC package
- It has 16-bit address bus and thus has 2^{16} = 64 KB addressing capability.
- Operate with 3 MHz single-phase clock
- +5 V single power supply

The logic pin layout and signal groups of the 8085nmicroprocessor are shown in Fig. 6. All the signals are classified into six groups:

- Address bus
- Data bus
- Control & status signals
- Power supply and frequency signals
- Externally initiated signals
- Serial I/O signals

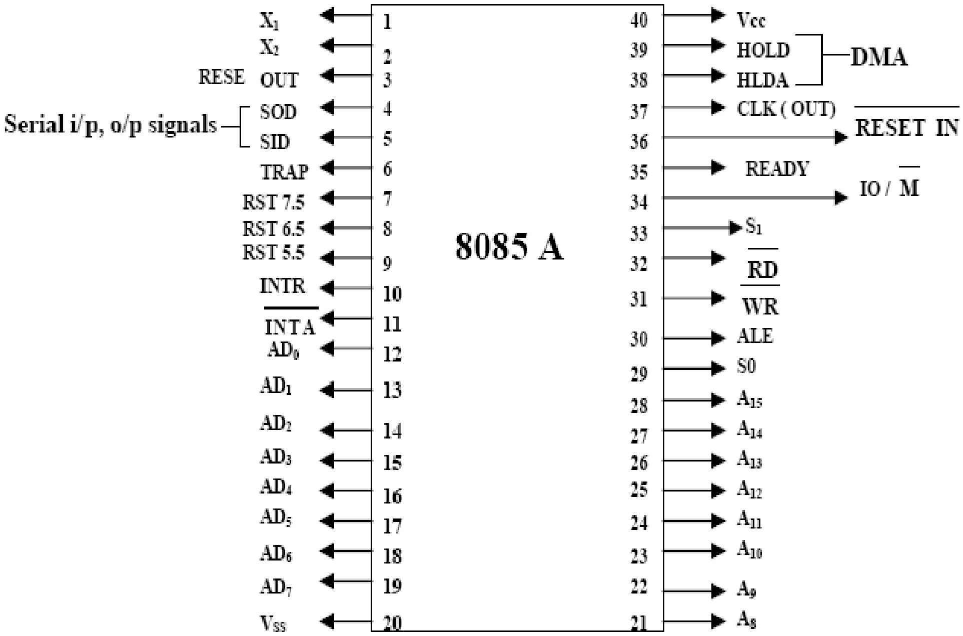


Fig. 6 8085 microprocessor pin layout and signal groups

Address and Data Buses:

- A8 – A15 (output, 3-state): Most significant eight bits of memory addresses and the eight bits of the I/O addresses. These lines enter into tri-state high impedance state during HOLD and HALT modes.
- AD0 – AD7 (input/output, 3-state): Lower significant bits of memory addresses and the eight bits of the I/O addresses during first clock cycle. Behaves as data bus

during third and fourth clock cycle. These lines enter into tri-state high impedance state during HOLD and HALT modes.

Control & Status Signals:

- ALE: Address latch enable
- $\overline{RD}$: Read control signal.
- $\overline{WR}$: Write control signal.
- IO/$\overline{M}$, S1 and S0 : Status signals.

Power Supply & Clock Frequency:

- Vcc: +5 V power supply
- Vss: Ground reference
- X1, X2: A crystal having frequency of 6 MHz is connected at these two pins
- CLK: Clock output

Externally Initiated and Interrupt Signals:

- $\overline{RESET\quad IN}$: When the signal on this pin is low, the PC is set to 0, the buses are tri-stated and the processor is reset.
- RESET OUT: This signal indicates that the processor is being reset. The signal can be used to reset other devices.
- READY: When this signal is low, the processor waits for an integral number of clock cycles until it goes high.
- HOLD: This signal indicates that a peripheral like DMA (direct memory access) controller is requesting the use of address and data bus.
- HLDA: This signal acknowledges the HOLD request.
- INTR: Interrupt request is a general-purpose interrupt.
- $\overline{INTA}$: This is used to acknowledge an interrupt.
- RST 7.5, RST 6.5, RST 5,5 – restart interrupt: These are vectored interrupts and have highest priority than INTR interrupt.
- TRAP: This is a non-maskable interrupt and has the highest priority.

Serial I/O Signals:

- SID: Serial input signal. Bit on this line is loaded to D7 bit of register A using RIM instruction.
- SOD: Serial output signal. Output SOD is set or reset by using SIM instruction.
-

4. INSTRUCTION SET AND EXECUTION IN 8085

Based on the design of the ALU and decoding unit, the microprocessor manufacturer provides instruction set for every microprocessor. The instruction set consists of both machine code and mnemonics.

An instruction is a binary pattern designed inside a microprocessor to perform a specific function. The entire group of instructions that a microprocessor supports is called instruction set. Microprocessor instructions can be classified based on the parameters such functionality, length and operand addressing.

Classification based on functionality:

I. Data transfer operations: This group of instructions copies data from source to destination. The content of the source is not altered.
II. Arithmetic operations: Instructions of this group perform operations like addition, subtraction, increment & decrement. One of the data used in arithmetic operation is stored in accumulator and the result is also stored in accumulator.
III. Logical operations: Logical operations include AND, OR, EXOR, NOT. The operations like AND, OR and EXOR uses two operands, one is stored in accumulator and other can be any register or memory location. The result is stored in accumulator. NOT operation requires single operand, which is stored in accumulator.
IV. Branching operations: Instructions in this group can be used to transfer program sequence from one memory location to another either conditionally or unconditionally.
V. Machine control operations: Instruction in this group control execution of other instructions and control operations like interrupt, halt etc.

Classification based on length:

I. One-byte instructions: Instruction having one byte in machine code. Examples are depicted in Table 2.
I. Two-byte instructions: Instruction having two byte in machine code. Examples are depicted in Table 3
II. Three-byte instructions: Instruction having three byte in machine code. Examples are depicted in Table 4.

Table 2 Examples of one byte instructions

Opcode	Operand	Machine code/Hex code
MOV	A, B	78
ADD	M	86

Table 3 Examples of two byte instructions

Opcode	Operand	Machine code/Hex code	Byte description
MVI	A, 7FH	3E	First byte
		7F	Second byte
ADI	0FH	C6	First byte
		0F	Second byte

Table 4 Examples of three byte instructions

Opcode	Operand	Machine code/Hex code	Byte description
JMP	9050H	C3	First byte
		50	Second byte
		90	Third byte
LDA	8850H	3A	First byte
		50	Second byte
		88	Third byte

Addressing Modes in Instructions:

The process of specifying the data to be operated on by the instruction is called addressing. The various formats for specifying operands are called addressing modes. The 8085 has the following five types of addressing:

I. Immediate addressing
II. Memory direct addressing
III. Register direct addressing
IV. Indirect addressing
V. Implicit addressing

Immediate Addressing:

In this mode, the operand given in the instruction - a byte or word – transfers to the destination register or memory location.

Ex: MVI A, 9AH

- The operand is a part of the instruction.
- The operand is stored in the register mentioned in the instruction.

Memory Direct Addressing:

Memory direct addressing moves a byte or word between a memory location and register. The memory location address is given in the instruction.

Ex: LDA 850FH

This instruction is used to load the content of memory address 850FH in the accumulator.

Register Direct Addressing:

Register direct addressing transfer a copy of a byte or word from source register to destination register.

Ex: MOV B, C

It copies the content of register C to register B.

Indirect Addressing:

Indirect addressing transfers a byte or word between a register and a memory location.

Ex: MOV A, M

Here the data is in the memory location pointed to by the contents of HL pair. The data is moved to the accumulator.

Implicit Addressing

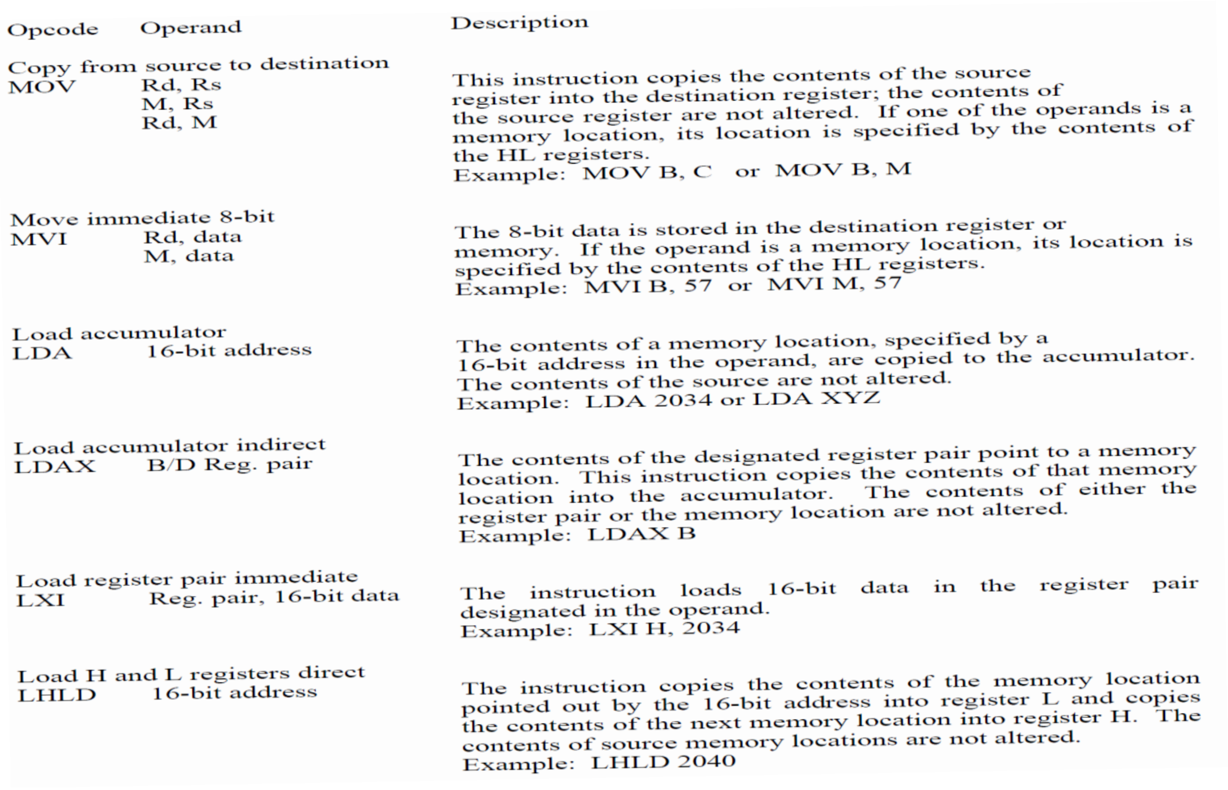
In this addressing mode the data itself specifies the data to be operated upon.

Ex: CMA

The instruction complements the content of the accumulator. No specific data or operand is mentioned in the instruction.

5. **INSTRUCTION SET OF 8085**

Data Transfer Instructions:



Opcode	Operand	Description
Copy from source to destination MOV	Rd, Rs M, Rs Rd, M	This instruction copies the contents of the source register into the destination register; the contents of the source register are not altered. If one of the operands is a memory location, its location is specified by the contents of the HL registers. Example: MOV B, C or MOV B, M
Move immediate 8-bit MVI	Rd, data M, data	The 8-bit data is stored in the destination register or memory. If the operand is a memory location, its location is specified by the contents of the HL registers. Example: MVI B, 57 or MVI M, 57
Load accumulator LDA	16-bit address	The contents of a memory location, specified by a 16-bit address in the operand, are copied to the accumulator. The contents of the source are not altered. Example: LDA 2034 or LDA XYZ
Load accumulator indirect LDAX	B/D Reg. pair	The contents of the designated register pair point to a memory location. This instruction copies the contents of that memory location into the accumulator. The contents of either the register pair or the memory location are not altered. Example: LDAX B
Load register pair immediate LXI	Reg. pair, 16-bit data	The instruction loads 16-bit data in the register pair designated in the operand. Example: LXI H, 2034
Load H and L registers direct LHLD	16-bit address	The instruction copies the contents of the memory location pointed out by the 16-bit address into register L and copies the contents of the next memory location into register H. The contents of source memory locations are not altered. Example: LHLD 2040

Store accumulator direct
STA 16-bit address

The contents of the accumulator are copied into the memory location specified by the operand. This is a 3-byte instruction, the second byte specifies the low-order address and the third byte specifies the high-order address.
Example: STA 4350 or STA XYZ

Store accumulator indirect
STAX Reg. pair

The contents of the accumulator are copied into the memory location specified by the contents of the operand (register pair). The contents of the accumulator are not altered.
Example: STAX B

Store H and L registers direct
SHLD 16-bit address

The contents of register L are stored into the memory location specified by the 16-bit address in the operand and the contents of H register are stored into the next memory location by incrementing the operand. The contents of registers HL are not altered. This is a 3-byte instruction, the second byte specifies the low-order address and the third byte specifies the high-order address.
Example: SHLD 2470

Exchange H and L with D and E
XCHG none

The contents of register H are exchanged with the contents of register D, and the contents of register L are exchanged with the contents of register E.
Example: XCHG

Copy H and L registers to the stack pointer
SPHL none

The instruction loads the contents of the H and L registers into the stack pointer register, the contents of the H register provide the high-order address and the contents of the L register provide the low-order address. The contents of the H and L registers are not altered.
Example: SPHL

Exchange H and L with top of stack
XTHL none

The contents of the L register are exchanged with the stack location pointed out by the contents of the stack pointer register. The contents of the H register are exchanged with the next stack location (SP+1); however, the contents of the stack pointer register are not altered.
Example: XTHL

Push register pair onto stack

PUSH Reg. pair

The contents of the register pair designated in the operand are copied onto the stack in the following sequence. The stack pointer register is decremented and the contents of the high-order register (B, D, H, A) are copied into that location. The stack pointer register is decremented again and the contents of the low-order register (C, E, L, flags) are copied to that location.
Example: PUSH B or PUSH A

Pop off stack to register pair

POP Reg. pair

The contents of the memory location pointed out by the stack pointer register are copied to the low-order register (C, E, L, status flags) of the operand. The stack pointer is incremented by 1 and the contents of that memory location are copied to the high-order register (B, D, H, A) of the operand. The stack pointer register is again incremented by 1.
Example: POP H or POP A

Output data from accumulator to a port with 8-bit address

OUT 8-bit port address

The contents of the accumulator are copied into the I/O port specified by the operand.
Example: OUT 87

Input data to accumulator from a port with 8-bit address

IN 8-bit port address

The contents of the input port designated in the operand are read and loaded into the accumulator.
Example: IN 82

Arithmetic Instructions:

Opcode Operand Description

Add register or memory to accumulator

ADD R
M

The contents of the operand (register or memory) are added to the contents of the accumulator and the result is stored in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the addition.
Example: ADD B or ADD M

Add register to accumulator with carry

ADC R
M

The contents of the operand (register or memory) and the Carry flag are added to the contents of the accumulator and the result is stored in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the addition.
Example: ADC B or ADC M

Add immediate to accumulator

ADI 8-bit data

The 8-bit data (operand) is added to the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the addition.
Example: ADI 45

Add immediate to accumulator with carry

ACI 8-bit data

The 8-bit data (operand) and the Carry flag are added to the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the addition.
Example: ACI 45

Add register pair to H and L registers

DAD Reg. pair

The 16-bit contents of the specified register pair are added to the contents of the HL register and the sum is stored in the HL register. The contents of the source register pair are not altered. If the result is larger than 16 bits, the CY flag is set. No other flags are affected.
Example: DAD H

Subtract register or memory from accumulator

SUB R
M

The contents of the operand (register or memory) are subtracted from the contents of the accumulator, and the result is stored in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the subtraction.
Example: SUB B or SUB M

Subtract source and borrow from accumulator

SBB R
M

The contents of the operand (register or memory) and the Borrow flag are subtracted from the contents of the accumulator and the result is placed in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the subtraction.
Example: SBB B or SBB M

Subtract immediate from accumulator

SUI 8-bit data

The 8-bit data (operand) is subtracted from the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the subtraction.
Example: SUI 45

Subtract immediate from accumulator with borrow

SBI 8-bit data

The 8-bit data (operand) and the Borrow flag are subtracted from the contents of the accumulator and the result is stored in the accumulator. All flags are modified to reflect the result of the subtracion.
Example: SBI 45

Increment register or memory by 1

INR R
M

The contents of the designated register or memory) are incremented by 1 and the result is stored in the same place. If the operand is a memory location, its location is specified by the contents of the HL registers.
Example: INR B or INR M

Increment register pair by 1

INX R

The contents of the designated register pair are incremented by 1 and the result is stored in the same place.
Example: INX H

Decrement register or memory by 1

DCR R
M

The contents of the designated register or memory are decremented by 1 and the result is stored in the same place. If the operand is a memory location, its location is specified by the contents of the HL registers.
Example: DCR B or DCR M

Decrement register pair by 1

DCX R

The contents of the designated register pair are decremented by 1 and the result is stored in the same place.
Example: DCX H

Decimal adjust accumulator

DAA none

The contents of the accumulator are changed from a binary value to two 4-bit binary coded decimal (BCD) digits. This is the only instruction that uses the auxiliary flag to perform the binary to BCD conversion, and the conversion procedure is described below. S, Z, AC, P, CY flags are altered to reflect the results of the operation.

If the value of the low-order 4-bits in the accumulator is greater than 9 or if AC flag is set, the instruction adds 6 to the low-order four bits.

If the value of the high-order 4-bits in the accumulator is greater than 9 or if the Carry flag is set, the instruction adds 6 to the high-order four bits.

Example: DAA

BRANCHING INSTRUCTIONS

Opcode Operand Description

Jump unconditionally

JMP 16-bit address

The program sequence is transferred to the memory location specified by the 16-bit address given in the operand.
Example: JMP 2034 or JMP XYZ

Jump conditionally

Operand: 16-bit address

The program sequence is transferred to the memory location specified by the 16-bit address given in the operand based on the specified flag of the PSW as described below.
Example: JZ 2034 or JZ XYZ

Opcode	Description	Flag Status
JC	Jump on Carry	CY = 1
JNC	Jump on no Carry	CY = 0
JP	Jump on positive	S = 0
JM	Jump on minus	S = 1
JZ	Jump on zero	Z = 1
JNZ	Jump on no zero	Z = 0
JPE	Jump on parity even	P = 1
JPO	Jump on parity odd	P = 0

Unconditional subroutine call

CALL 16-bit address

The program sequence is transferred to the memory location specified by the 16-bit address given in the operand. Before the transfer, the address of the next instruction after CALL (the contents of the program counter) is pushed onto the stack.
Example: CALL 2034 or CALL XYZ

Call conditionally

Operand: 16-bit address

The program sequence is transferred to the memory location specified by the 16-bit address given in the operand based on the specified flag of the PSW as described below. Before the transfer, the address of the next instruction after the call (the contents of the program counter) is pushed onto the stack.
Example: CZ 2034 or CZ XYZ

Opcode	Description	Flag Status
CC	Call on Carry	CY = 1
CNC	Call on no Carry	CY = 0
CP	Call on positive	S = 0
CM	Call on minus	S = 1
CZ	Call on zero	Z = 1
CNZ	Call on no zero	Z = 0
CPE	Call on parity even	P = 1
CPO	Call on parity odd	P = 0

Return from subroutine unconditionally

RET none

The program sequence is transferred from the subroutine to the calling program. The two bytes from the top of the stack are copied into the program counter, and program execution begins at the new address.
Example: RET

Return from subroutine conditionally

Operand: none

The program sequence is transferred from the subroutine to the calling program based on the specified flag of the PSW as described below. The two bytes from the top of the stack are copied into the program counter, and program execution begins at the new address.
Example: RZ

Opcode	Description	Flag Status
RC	Return on Carry	CY = 1
RNC	Return on no Carry	CY = 0
RP	Return on positive	S = 0
RM	Return on minus	S = 1
RZ	Return on zero	Z = 1
RNZ	Return on no zero	Z = 0
RPE	Return on parity even	P = 1
RPO	Return on parity odd	P = 0

Load program counter with HL contents

PCHL none The contents of registers H and L are copied into the program counter. The contents of H are placed as the high-order byte and the contents of L as the low-order byte.
Example: PCHL

Restart

RST 0-7 The RST instruction is equivalent to a 1-byte call instruction to one of eight memory locations depending upon the number. The instructions are generally used in conjunction with interrupts and inserted using external hardware. However these can be used as software instructions in a program to transfer program execution to one of the eight locations. The addresses are:

Instruction	Restart Address
RST 0	0000H
RST 1	0008H
RST 2	0010H
RST 3	0018H
RST 4	0020H
RST 5	0028H
RST 6	0030H
RST 7	0038H

The 8085 has four additional interrupts and these interrupts generate RST instructions internally and thus do not require any external hardware. These instructions and their Restart addresses are:

Interrupt	Restart Address
TRAP	0024H
RST 5.5	002CH
RST 6.5	0034H
RST 7.5	003CH

LOGICAL INSTRUCTIONS

Opcode Operand Description

Compare register or memory with accumulator

CMP R
M
The contents of the operand (register or memory) are compared with the contents of the accumulator. Both contents are preserved . The result of the comparison is shown by setting the flags of the PSW as follows:
if (A) < (reg/mem): carry flag is set, s=1
if (A) = (reg/mem): zero flag is set, s=0
if (A) > (reg/mem): carry and zero flags are reset, s=0
Example: CMP B or CMP M

Compare immediate with accumulator

CPI 8-bit data The second byte (8-bit data) is compared with the contents of the accumulator. The values being compared remain unchanged. The result of the comparison is shown by setting the flags of the PSW as follows:
if (A) < data: carry flag is set, s=1
if (A) = data: zero flag is set, s=0
if (A) > data: carry and zero flags are reset, s=0
Example: CPI 89

Logical AND register or memory with accumulator

ANA R
M
The contents of the accumulator are logically ANDed with the contents of the operand (register or memory), and the result is placed in the accumulator. If the operand is a memory location, its address is specified by the contents of HL registers. S, Z, P are modified to reflect the result of the operation. CY is reset. AC is set.
Example: ANA B or ANA M

Logical AND immediate with accumulator

ANI 8-bit data The contents of the accumulator are logically ANDed with the 8-bit data (operand) and the result is placed in the accumulator. S, Z, P are modified to reflect the result of the operation. CY is reset. AC is set.
Example: ANI 86

Exclusive OR register or memory with accumulator

XRA R
M

The contents of the accumulator are Exclusive ORed with the contents of the operand (register or memory), and the result is placed in the accumulator. If the operand is a memory location, its address is specified by the contents of HL registers. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.
Example: XRA B or XRA M

Exclusive OR immediate with accumulator

XRI 8-bit data

The contents of the accumulator are Exclusive ORed with the 8-bit data (operand) and the result is placed in the accumulator. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.
Example: XRI 86

Logical OR register or memory with accumulaotr

ORA R
M

The contents of the accumulator are logically ORed with the contents of the operand (register or memory), and the result is placed in the accumulator. If the operand is a memory location, its address is specified by the contents of HL registers. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.
Example: ORA B or ORA M

Logical OR immediate with accumulator

ORI 8-bit data

The contents of the accumulator are logically ORed with the 8-bit data (operand) and the result is placed in the accumulator. S, Z, P are modified to reflect the result of the operation. CY and AC are reset.
Example: ORI 86

Rotate accumulator left

RLC none

Each binary bit of the accumulator is rotated left by one position. Bit D_7 is placed in the position of D_0 as well as in the Carry flag. CY is modified according to bit D_7. S, Z, P, AC are not affected.
Example: RLC

Rotate accumulator right

RRC none

Each binary bit of the accumulator is rotated right by one position. Bit D_0 is placed in the position of D_7 as well as in the Carry flag. CY is modified according to bit D_0. S, Z, P, AC are not affected.
Example: RRC

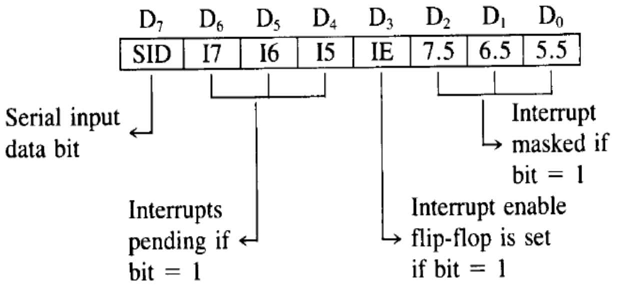
Rotate accumulator left through carry		
RAL	none	Each binary bit of the accumulator is rotated left by one position through the Carry flag. Bit D7 is placed in the Carry flag, and the Carry flag is placed in the least significant position D0. CY is modified according to bit D7. S, Z, P, AC are not affected. Example: RAL
Rotate accumulator right through carry		
RAR	none	Each binary bit of the accumulator is rotated right by one position through the Carry flag. Bit D0 is placed in the Carry flag, and the Carry flag is placed in the most significant position D7. CY is modified according to bit D0. S, Z, P, AC are not affected. Example: RAR
Complement accumulator		
CMA	none	The contents of the accumulator are complemented. No flags are affected. Example: CMA
Complement carry		
CMC	none	The Carry flag is complemented. No other flags are affected. Example: CMC
Set Carry		
STC	none	The Carry flag is set to 1. No other flags are affected. Example: STC

CONTROL INSTRUCTIONS

Opcode	Operand	Description
No operation		
NOP	none	No operation is performed. The instruction is fetched and decoded. However no operation is executed. Example: NOP
Halt and enter wait state		
HLT	none	The CPU finishes executing the current instruction and halts any further execution. An interrupt or reset is necessary to exit from the halt state. Example: HLT
Disable interrupts		
DI	none	The interrupt enable flip-flop is reset and all the interrupts except the TRAP are disabled. No flags are affected. Example: DI
Enable interrupts		
EI	none	The interrupt enable flip-flop is set and all interrupts are enabled. No flags are affected. After a system reset or the acknowledgement of an interrupt, the interrupt enable flip-flop is reset, thus disabling the interrupts. This instruction is necessary to reenable the interrupts (except TRAP). Example: EI

Read interrupt mask
RIM none

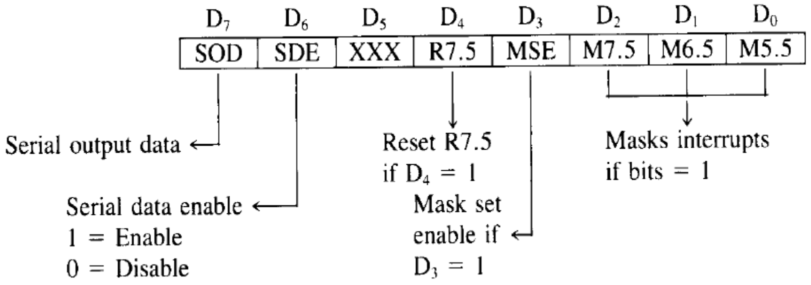
This is a multipurpose instruction used to read the status of interrupts 7.5, 6.5, 5.5 and read serial data input bit. The instruction loads eight bits in the accumulator with the following interpretations.
Example: RIM



D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0
SID I7 I6 I5 IE 7.5 6.5 5.5
Serial input data bit
Interrupts pending if bit = 1
Interrupt enable flip-flop is set if bit = 1
Interrupt masked if bit = 1

Set interrupt mask
SIM none

This is a multipurpose instruction and used to implement the 8085 interrupts 7.5, 6.5, 5.5, and serial data output. The instruction interprets the accumulator contents as follows.
Example: SIM



D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0
SOD SDE XXX R7.5 MSE M7.5 M6.5 M5.5
Serial output data
Serial data enable
1 = Enable
0 = Disable
Reset R7.5 if D_4 = 1
Mask set enable if D_3 = 1
Masks interrupts if bits = 1

- □ SOD—Serial Output Data: Bit D_7 of the accumulator is latched into the SOD output line and made available to a serial peripheral if bit D_6 = 1.
- □ SDE—Serial Data Enable: If this bit = 1, it enables the serial output. To implement serial output, this bit needs to be enabled.
- □ XXX—Don't Care
- □ R7.5—Reset RST 7.5: If this bit = 1, RST 7.5 flip-flop is reset. This is an additional control to reset RST 7.5.
- □ MSE—Mask Set Enable: If this bit is high, it enables the functions of bits D_2, D_1, D_0. This is a master control over all the interrupt masking bits. If this bit is low, bits D_2, D_1, and D_0 do not have any effect on the masks.
- □ M7.5—D_2 = 0, RST 7.5 is enabled.
  = 1, RST 7.5 is masked or disabled.
- □ M6.5—D_1 = 0, RST 6.5 is enabled.
  = 1, RST 6.5 is masked or disabled.
- □ M5.5—D_0 = 0, RST 5.5 is enabled.
  = 1, RST 5.5 is masked or disabled.

6. **INSTRUCTION EXECUTION AND TIMING DIAGRAM:**

Each instruction in 8085 microprocessor consists of two part- operation code (opcode) and operand. The opcode is a command such as ADD and the operand is an object to be operated on, such as a byte or the content of a register.

Instruction Cycle: The time taken by the processor to complete the execution of an instruction. An instruction cycle consists of one to six machine cycles.

Machine Cycle: The time required to complete one operation; accessing either the memory or I/O device. A machine cycle consists of three to six T-states.

T-State: Time corresponding to one clock period. It is the basic unit to calculate execution of instructions or programs in a processor.

To execute a program, 8085 performs various operations as:

- Opcode fetch
- Operand fetch
- Memory read/write
- I/O read/write

External communication functions are:

- Memory read/write
- I/O read/write
- Interrupt request acknowledge

Opcode Fetch Machine Cycle:

It is the first step in the execution of any instruction. The timing diagram of this cycle is given in Fig. 7.

The following points explain the various operations that take place and the signals that are changed during the execution of opcode fetch machine cycle:

T1 clock cycle

i. The content of PC is placed in the address bus; AD0 - AD7 lines contains lower bit address and A8 – A15 contains higher bit address.

ii. $IO/\overline{M}$ signal is low indicating that a memory location is being accessed. S1 and S0 also changed to the levels as indicated in Table 1.

iii. ALE is high, indicates that multiplexed AD0 – AD7 act as lower order bus.

T2 clock cycle

i. Multiplexed address bus is now changed to data bus.

ii. The $\overline{RD}$ signal is made low by the processor. This signal makes the memory device load the data bus with the contents of the location addressed by the processor.

T3 clock cycle

i. The opcode available on the data bus is read by the processor and moved to the instruction register.
ii. The $\overline{RD}$ signal is deactivated by making it logic 1.

T4 clock cycle

i. The processor decode the instruction in the instruction register and generate the necessary control signals to execute the instruction. Based on the instruction further operations such as fetching, writing into memory etc takes place.

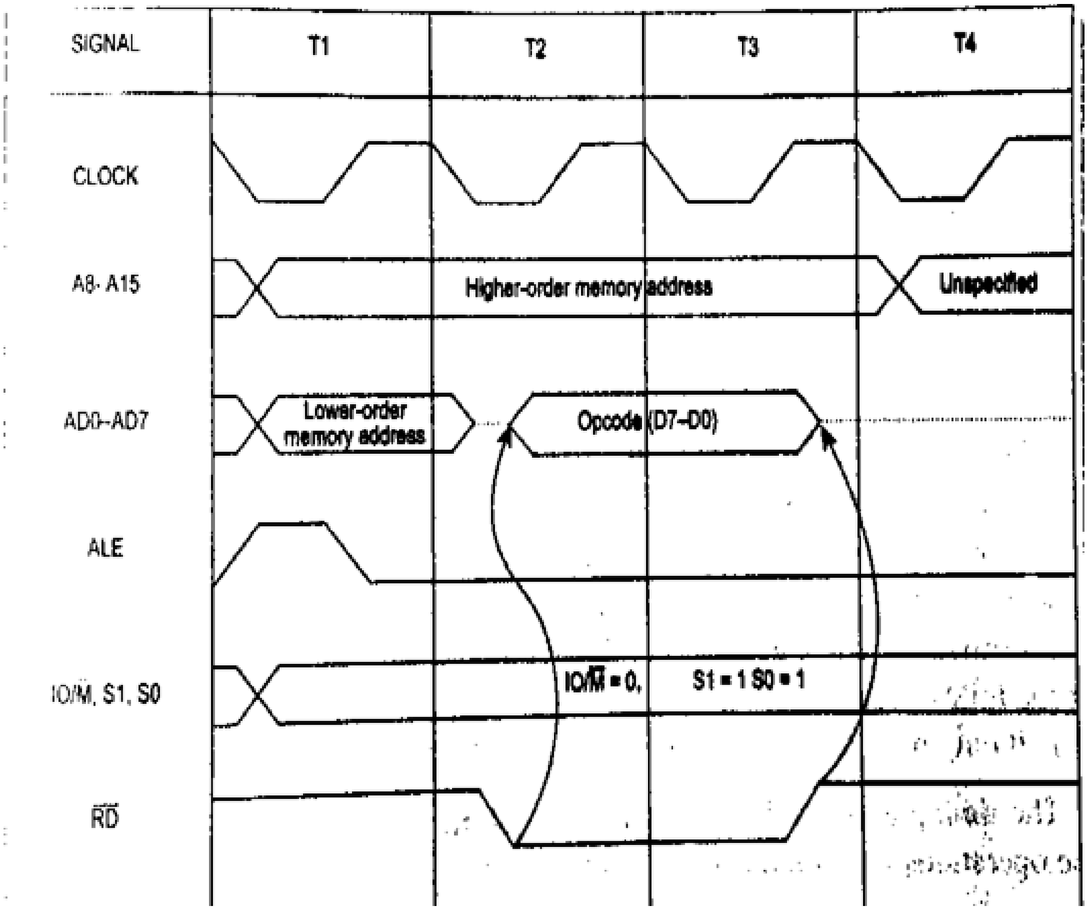


Fig. 7 Timing diagram for opcode fetch cycle

Memory Read Machine Cycle:

The memory read cycle is executed by the processor to read a data byte from memory. The machine cycle is exactly same to opcode fetch except: a) It has three T-states b) The S0 signal is set to 0. The timing diagram of this cycle is given in Fig. 8.

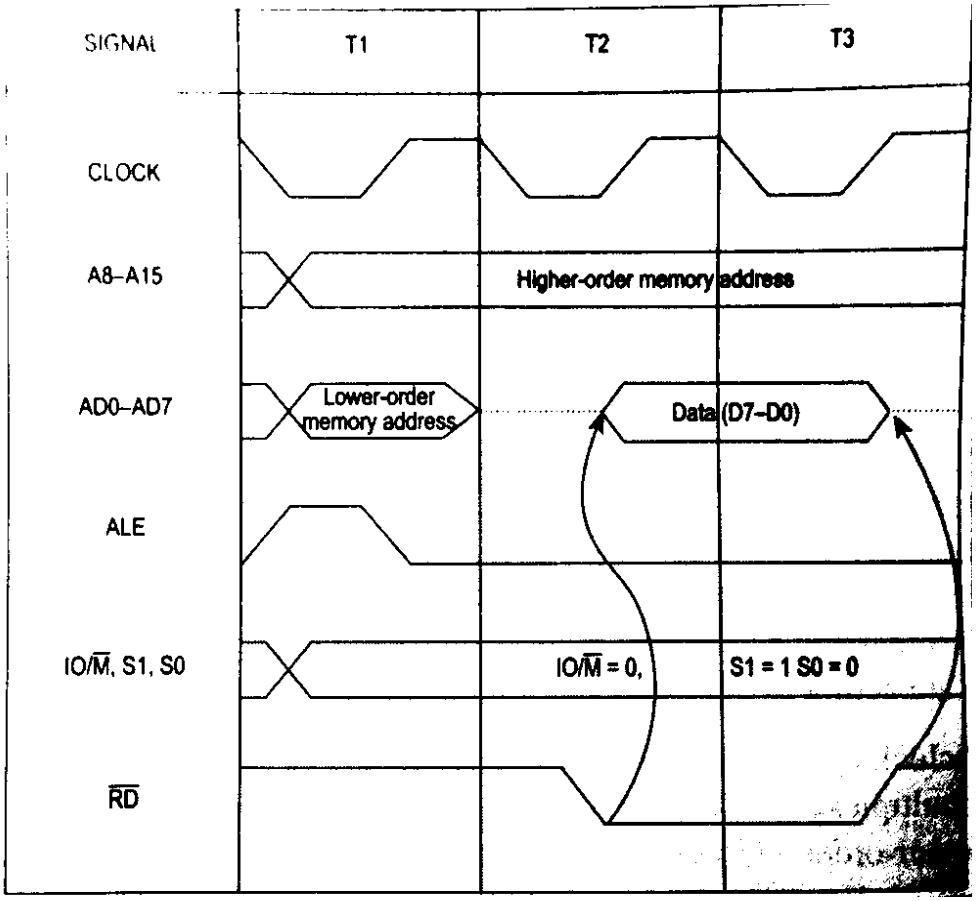


Fig. 8 Timing diagram for memory read machine cycle

Memory Write Machine Cycle:

The memory write cycle is executed by the processor to write a data byte in a memory location. The processor takes three T-states and $\overline{\text{WR}}$ signal is made low. The timing diagram of this cycle is given in Fig. 9.

I/O Read Cycle:

The I/O read cycle is executed by the processor to read a data byte from I/O port or from peripheral, which is I/O mapped in the system. The 8-bit port address is placed both in the lower and higher order address bus. The processor takes three T-states to execute this machine cycle. The timing diagram of this cycle is given in Fig. 10.

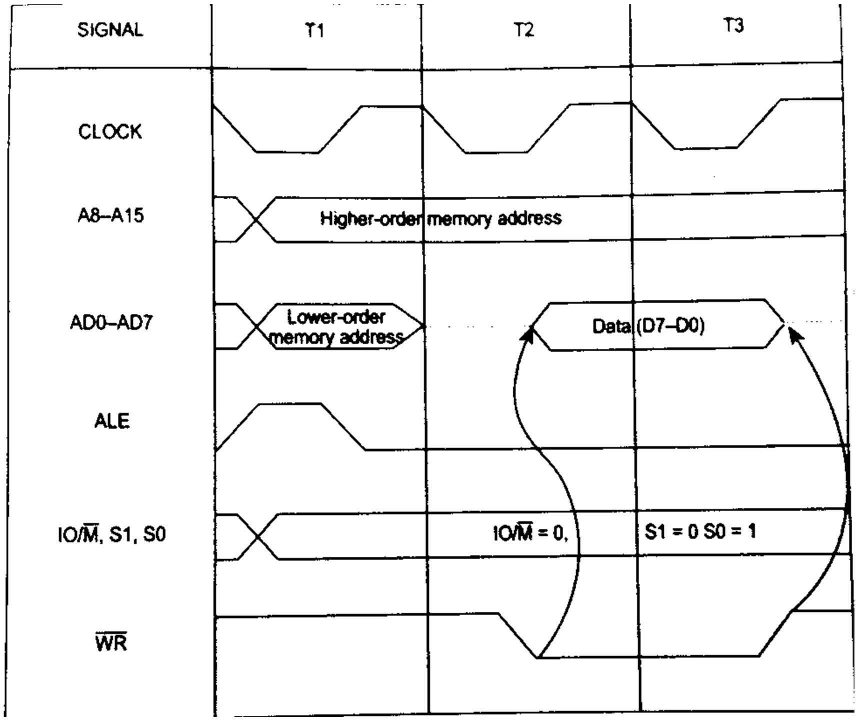


Fig. 9 Timing diagram for memory write machine cycle

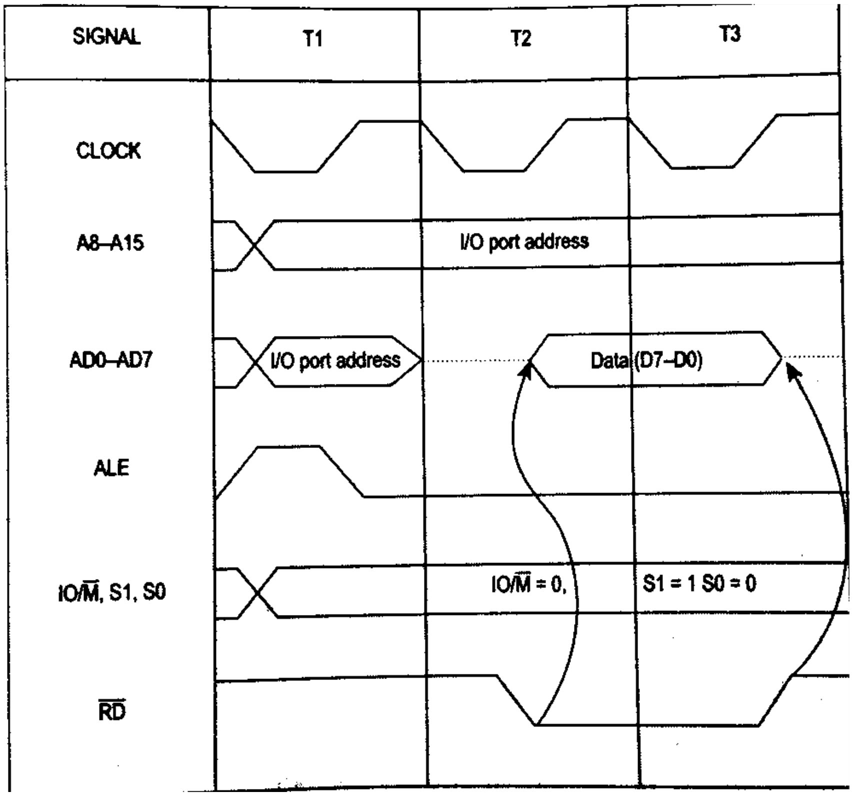


Fig. 10 Timing diagram I/O read machine cycle

I/O Write Cycle:

The I/O write cycle is executed by the processor to write a data byte to I/O port or to a peripheral, which is I/O mapped in the system. The processor takes three T-states to execute this machine cycle. The timing diagram of this cycle is given in Fig. 11.

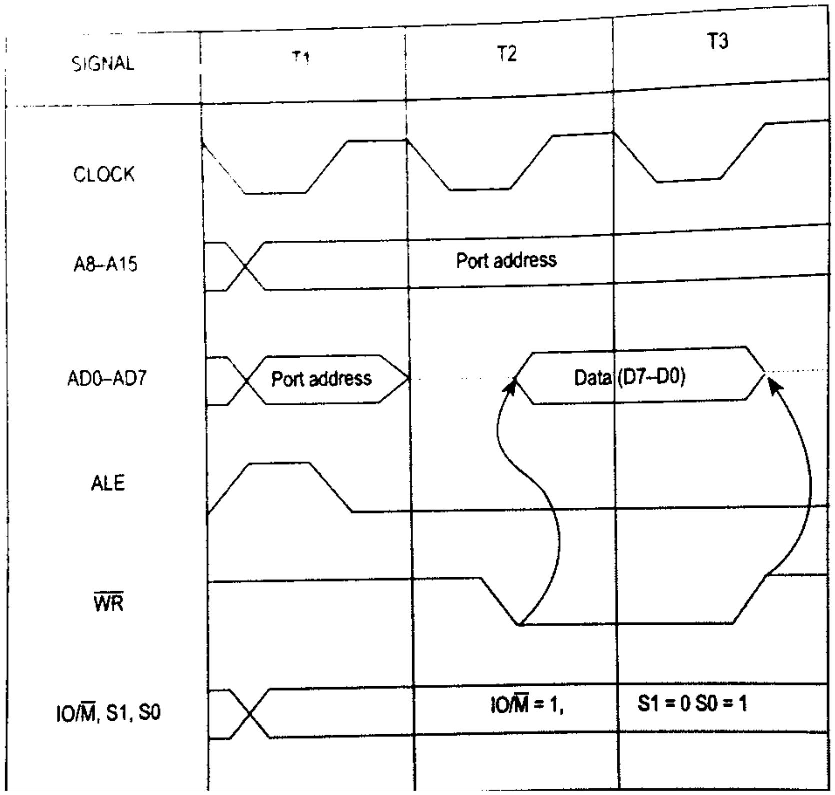


Fig. 11 Timing diagram I/O write machine cycle

Ex: Timing diagram for IN 80H.

The instruction and the corresponding codes and memory locations are given in Table 5.

Table 5 IN instruction

Address	Mnemonics	Opcode
800F	IN 80H	DB
8010		80

i. During the first machine cycle, the opcode DB is fetched from the memory, placed in the instruction register and decoded.
ii. During second machine cycle, the port address 80H is read from the next memory location.
iii. During the third machine cycle, the address 80H is placed in the address bus and the data read from that port address is placed in the accumulator.

The timing diagram is shown in Fig. 12.

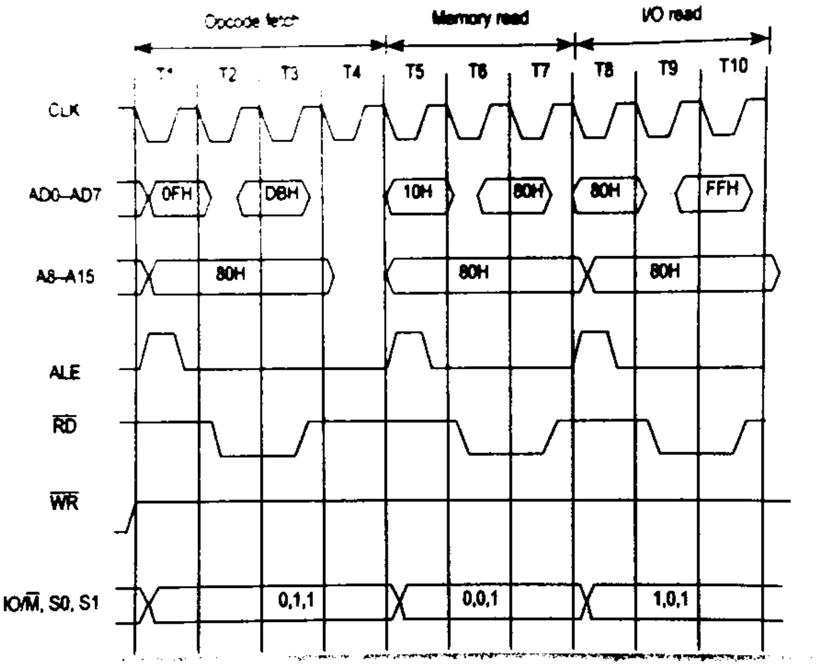


Fig. 12 Timing diagram for the IN instruction

7. 8085 INTERRUPTS

Interrupt Structure:

Interrupt is the mechanism by which the processor is made to transfer control from its current program execution to another program having higher priority. The interrupt signal may be given to the processor by any external peripheral device.

The program or the routine that is executed upon interrupt is called interrupt service routine (ISR). After execution of ISR, the processor must return to the interrupted program. Key features in the interrupt structure of any microprocessor are as follows:

i. Number and types of interrupt signals available.
ii. The address of the memory where the ISR is located for a particular interrupt signal. This address is called interrupt vector address (IVA).
iii. Masking and unmasking feature of the interrupt signals.
iv. Priority among the interrupts.
v. Timing of the interrupt signals.
vi. Handling and storing of information about the interrupt program (status information).

Types of Interrupts:

Interrupts are classified based on their maskability, IVA and source. They are classified as:

i. Vectored and Non-Vectored Interrupts
   - Vectored interrupts require the IVA to be supplied by the external device that gives the interrupt signal. This technique is vectoring, is implemented in number of ways.
   - Non-vectored interrupts have fixed IVA for ISRs of different interrupt signals.

ii. Maskable and Non-Maskable Interrupts
   - Maskable interrupts are interrupts that can be blocked. Masking can be done by software or hardware means.
   - Non-maskable interrupts are interrupts that are always recognized; the corresponding ISRs are executed.

iii. Software and Hardware Interrupts
   - Software interrupts are special instructions, after execution transfer the control to predefined ISR.
   - Hardware interrupts are signals given to the processor, for recognition as an interrupt and execution of the corresponding ISR.

Interrupt Handling Procedure:

The following sequence of operations takes place when an interrupt signal is recognized:

i. Save the PC content and information about current state (flags, registers etc) in the stack.
ii. Load PC with the beginning address of an ISR and start to execute it.
iii. Finish ISR when the return instruction is executed.
iv. Return to the point in the interrupted program where execution was interrupted.

Interrupt Sources and Vector Addresses in 8085:

Software Interrupts:

8085 instruction set includes eight software interrupt instructions called Restart (RST) instructions. These are one byte instructions that make the processor execute a subroutine at predefined locations. Instructions and their vector addresses are given in Table 6.

Table 6 Software interrupts and their vector addresses

Instruction	Machine hex code	Interrupt Vector Address
RST 0	C7	0000H
RST 1	CF	0008H
RST 2	D7	0010H
RST 3	DF	0018H
RST 4	E7	0020H
RST 5	EF	0028H
RST 6	F7	0030H
RST 7	FF	0032H

The software interrupts can be treated as CALL instructions with default call locations. The concept of priority does not apply to software interrupts as they are inserted into the program as instructions by the programmer and executed by the processor when the respective program lines are read.

Hardware Interrupts and Priorities:

8085 have five hardware interrupts – INTR, RST 5.5, RST 6.5, RST 7.5 and TRAP. Their IVA and priorities are given in Table 7.

Table 7 Hardware interrupts of 8085

Interrupt	Interrupt vector address	Maskable or non-maskable	Edge or level triggered	priority
TRAP	0024H	Non-makable	Level	1
RST 7.5	003CH	Maskable	Rising edge	2
RST 6.5	0034H	Maskable	Level	3
RST 5.5	002CH	Maskable	Level	4
INTR	Decided by hardware	Maskable	Level	5

Masking of Interrupts:

Masking can be done for four hardware interrupts INTR, RST 5.5, RST 6.5, and RST 7.5. The masking of 8085 interrupts is done at different levels. Fig. 13 shows the organization of hardware interrupts in the 8085.

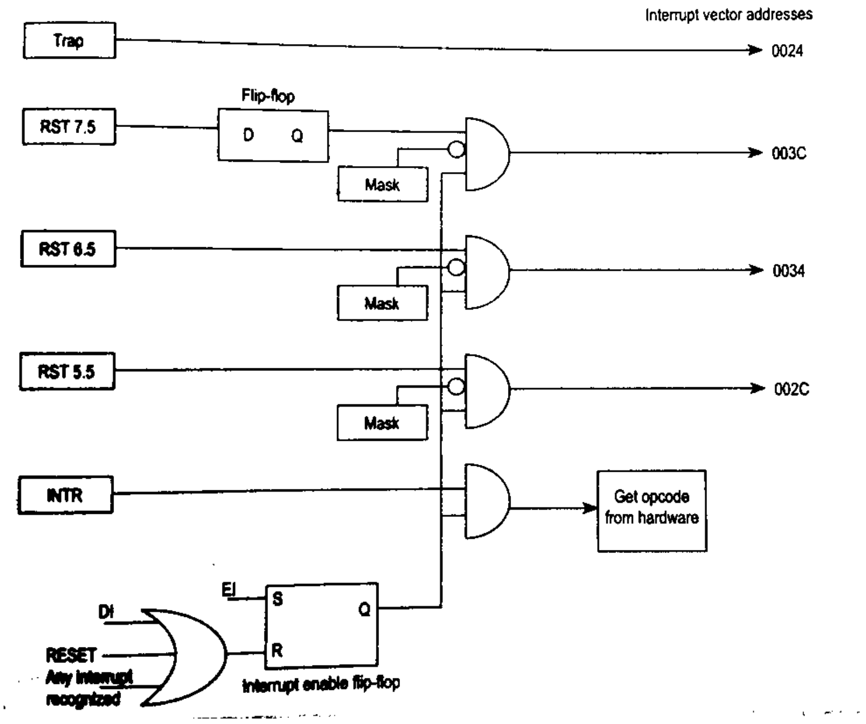


Fig. 13 Interrupt structure of 8085

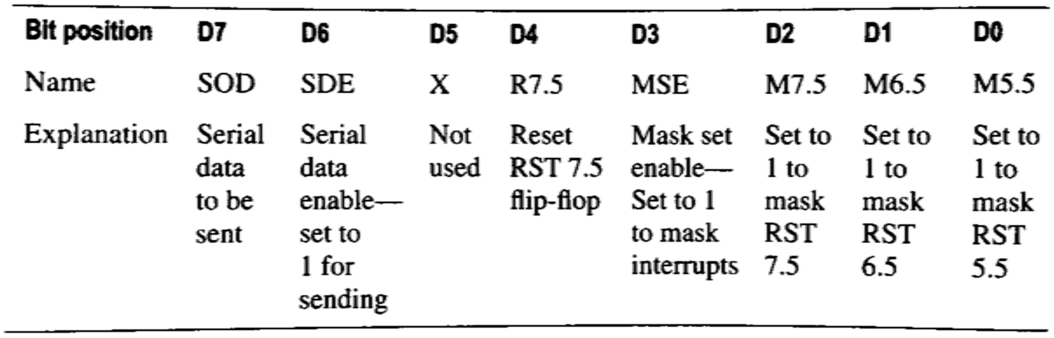
The Fig. 13 is explained by the following five points:

i. The maskable interrupts are by default masked by the Reset signal. So no interrupt is recognized by the hardware reset.
ii. The interrupts can be enabled by the EI instruction.
iii. The three RST interrupts can be selectively masked by loading the appropriate word in the accumulator and executing SIM instruction. This is called software masking.
iv. All maskable interrupts are disabled whenever an interrupt is recognized.
v. All maskable interrupts can be disabled by executing the DI instruction.

RST 7.5 alone has a flip-flop to recognize edge transition. The DI instruction reset interrupt enable flip-flop in the processor and the interrupts are disabled. To enable interrupts, EI instruction has to be executed.

SIM Instruction:

The SIM instruction is used to mask or unmask RST hardware interrupts. When executed, the SIM instruction reads the content of accumulator and accordingly mask or unmask the interrupts. The format of control word to be stored in the accumulator before executing SIM instruction is as shown in Fig. 14.



Bit position	D7	D6	D5	D4	D3	D2	D1	D0
Name	SOD	SDE	X	R7.5	MSE	M7.5	M6.5	M5.5
Explanation	Serial data to be sent	Serial data enable—set to 1 for sending	Not used	Reset RST 7.5 flip-flop	Mask set enable—Set to 1 to mask interrupts	Set to 1 to mask RST 7.5	Set to 1 to mask RST 6.5	Set to 1 to mask RST 5.5

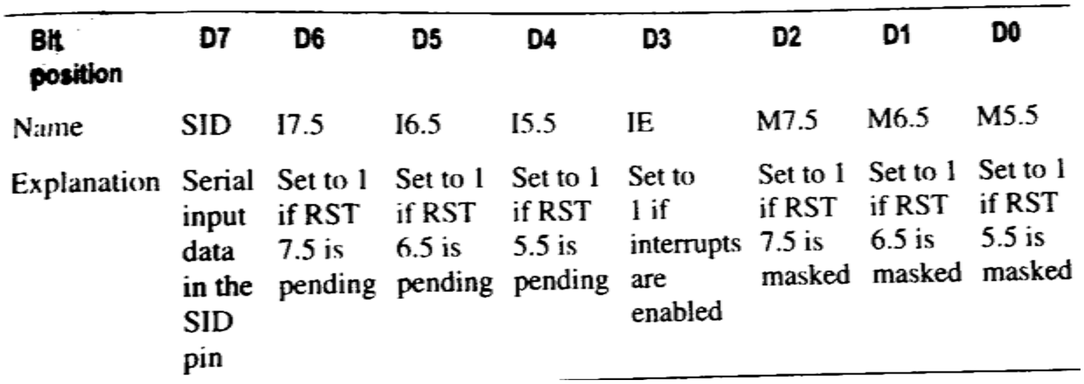
Fig. 14 Accumulator bit pattern for SIM instruction

In addition to masking interrupts, SIM instruction can be used to send serial data on the SOD line of the processor. The data to be send is placed in the MSB bit of the accumulator and the serial data output is enabled by making D6 bit to 1.

RIM Instruction:

RIM instruction is used to read the status of the interrupt mask bits. When RIM instruction is executed, the accumulator is loaded with the current status of the interrupt masks and the pending interrupts. The format and the meaning of the data stored in the accumulator after execution of RIM instruction is shown in Fig. 15.

In addition RIM instruction is also used to read the serial data on the SID pin of the processor. The data on the SID pin is stored in the MSB of the accumulator after the execution of the RIM instruction.



Bit position	D7	D6	D5	D4	D3	D2	D1	D0
Name	SID	I7.5	I6.5	I5.5	IE	M7.5	M6.5	M5.5
Explanation	Serial input data in the SID pin	Set to 1 if RST 7.5 is pending	Set to 1 if RST 6.5 is pending	Set to 1 if RST 5.5 is pending	Set to 1 if interrupts are enabled	Set to 1 if RST 7.5 is masked	Set to 1 if RST 6.5 is masked	Set to 1 if RST 5.5 is masked

Fig. 15 Accumulator bit pattern after execution of RIM instruction

Ex: Write an assembly language program to enables all the interrupts in 8085 after reset.

```
EI          : Enable interrupts

MVI A, 08H  : Unmask the interrupts

SIM         : Set the mask and unmask using SIM instruction
```

Timing of Interrupts:

The interrupts are sensed by the processor one cycle before the end of execution of each instruction. An interrupts signal must be applied long enough for it to be recognized. The longest instruction of the 8085 takes 18 clock periods. So, the interrupt signal must be applied for at least 17.5 clock periods. This decides the minimum pulse width for the interrupt signal.

The maximum pulse width for the interrupt signal is decided by the condition that the interrupt signal must not be recognized once again. This is under the control of the programmer.

QUESTIONS:

1. What is the function of a microprocessor in a system?
2. Why is the data bus in 8085 bidirectional?
3. How does microprocessor differentiate between data and instruction?
4. How long would the processor take to execute the instruction LDA 1753H if the T-state duration is 2µs?
5. Draw the timing diagram of the instruction LDAX B.
6. Sketch and explain the various pins of the 8085.
7. Explain direct addressing mode of 8085 with an example?
8. Draw and explain the timing diagram of the instruction IN 82H.
9. What is meant by 'priority of the interrupts'? Explain the operation of the interrupts structure of the 8085, with the help of a circuit diagram.
10. Explain the bit pattern for SIM instruction. Write the assembly language program lines to enable all the interrupts in the 8085 after reset.
11. Write the logical instructions which affect and which does not affect flags in 8085.
12. Write an ALP in 8085 MPU to reject all the negative readings and add all the positive reading from a set of ten reading stored in memory locations starting at XX60H. When the sum exceeds eight bits produce output FFH to PORT1 to indicate overload otherwise display the sum.
13. Write an ALP in 8085 to eliminate the blanks (bytes with zero value) from a string of eight data bytes. Use two memory pointers: one to get a byte and the other to store the byte.
14. Design an up-down counter to count from 0 to 9 and 9 to 0 continuously with a 1.5 second delay between each count, and display the count at one of the output ports.

MODULE: 2

1. INTERFACING MEMORY AND I/O DEVICES WITH 8085

The programs and data that are executed by the microprocessor have to be stored in ROM/EPROM and RAM, which are basically semiconductor memory chips. The programs and data that are stored in ROM/EPROM are not erased even when power supply to the chip is removed. Hence, they are called non-volatile memory. They can be used to store permanent programs.

In a RAM, stored programs and data are erased when the power supply to the chip is removed. Hence, RAM is called volatile memory. RAM can be used to store programs and data that include, programs written during software development for a microprocessor based system, program written when one is learning assembly language programming and data enter while testing these programs.

Input and output devices, which are interfaced with 8085, are essential in any microprocessor based system. They can be interfaced using two schemes: I/O mapped I/O and memory-mapped I/O. In the I/O mapped I/O scheme, the I/O devices are treated differently from memory. In the memory-mapped I/O scheme, each I/O device is assumed to be a memory location.

2. INTERFACING MEMORY CHIPS WITH 8085

8085 has 16 address lines (A0 - A15), hence a maximum of 64 KB (= 2^{16} bytes) of memory locations can be interfaced with it. The memory address space of the 8085 takes values from 0000H to FFFFH.

The 8085 initiates set of signals such as $\mathrm{IO}/\overline{\mathrm{M}}$, $\overline{\mathrm{RD}}$ and $\overline{\mathrm{WR}}$ when it wants to read from and write into memory. Similarly, each memory chip has signals such as $\overline{\mathrm{CE}}$ or $\overline{\mathrm{CS}}$ (chip enable or chip select), $\overline{\mathrm{OE}}$ or $\overline{\mathrm{RD}}$ (output enable or read) and $\overline{\mathrm{WE}}$ or $\overline{\mathrm{WR}}$ (write enable or write) associated with it.

Generation of Control Signals for Memory:

When the 8085 wants to read from and write into memory, it activates $\mathrm{IO}/\overline{\mathrm{M}}$, $\overline{\mathrm{RD}}$ and $\overline{\mathrm{WR}}$ signals as shown in Table 8.

Table 8 Status of $\mathrm{IO}/\overline{\mathrm{M}}$, $\overline{\mathrm{RD}}$ and $\overline{\mathrm{WR}}$ signals during memory read and write operations

$\mathrm{IO}/\overline{\mathrm{M}}$	$\overline{\mathrm{RD}}$	$\overline{\mathrm{WR}}$	Operation
0	0	1	8085 reads data from memory
0	1	0	8085 writes data into memory

Using $\mathrm{IO}/\overline{\mathrm{M}}$, $\overline{\mathrm{RD}}$ and $\overline{\mathrm{WR}}$ signals, two control signals $\overline{\mathrm{MEMR}}$ (memory read) and $\overline{\mathrm{MEMW}}$ (memory write) are generated. Fig. 16 shows the circuit used to generate these signals.

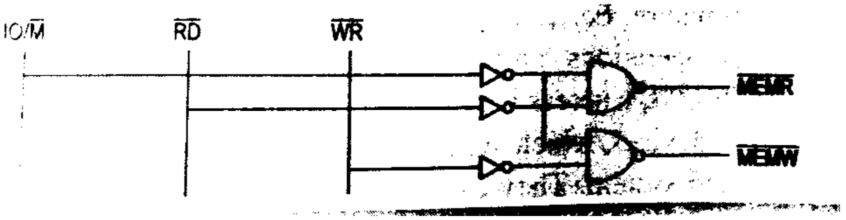


Fig. 16 Circuit used to generate $\overline{\text{MEMR}}$ and $\overline{\text{MEMW}}$ signals

When is $\text{IO}/\overline{\text{M}}$ high, both memory control signals are deactivated irrespective of the status of $\overline{\text{RD}}$ and $\overline{\text{WR}}$ signals.

Ex: Interface an IC 2764 with 8085 using NAND gate address decoder such that the address range allocated to the chip is 0000H – 1FFFH.

Specification of IC 2764:

- 8 KB (8 x 2^{10} byte) EPROM chip
- 13 address lines (2^{13} bytes = 8 KB)

Interfacing:

- 13 address lines of IC are connected to the corresponding address lines of 8085.
- Remaining address lines of 8085 are connected to address decoder formed using logic gates, the output of which is connected to the $\overline{\text{CE}}$ pin of IC.
- Address range allocated to the chip is shown in Table 9.
- Chip is enabled whenever the 8085 places an address allocated to EPROM chip in the address bus. This is shown in Fig. 17.

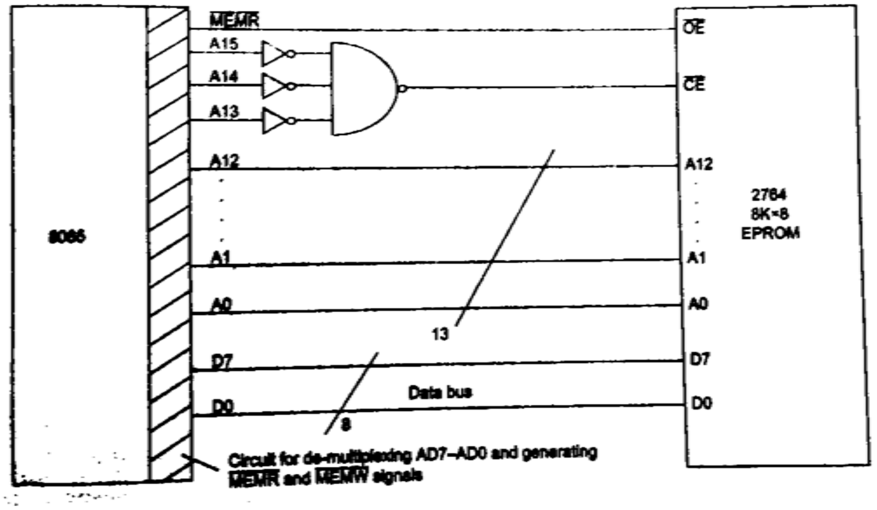
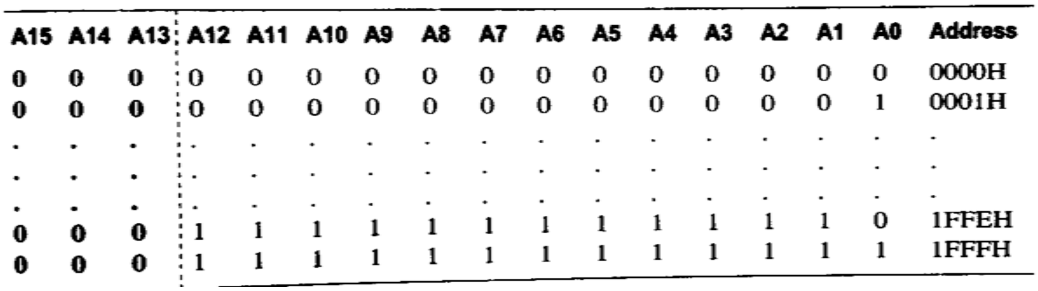


Fig. 17 Interfacing IC 2764 with the 8085

Table 9 Address allocated to IC 2764



A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Address
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0001H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1FFEH
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFFH

Ex: Interface a 6264 IC (8K x 8 RAM) with the 8085 using NAND gate decoder such that the starting address assigned to the chip is 4000H.

Specification of IC 6264:

- 8K x 8 RAM
- 8 KB = 2^{13} bytes
- 13 address lines

The ending address of the chip is 5FFFH (since 4000H + 1FFFH = 5FFFH). When the address 4000H to 5FFFH are written in binary form, the values in the lines A15, A14, A13 are 0, 1 and 0 respectively. The NAND gate is designed such that when the lines A15 and A13 carry 0 and A14 carries 1, the output of the NAND gate is 0. The NAND gate output is in turn connected to the $\overline{\text{CE1}}$ pin of the RAM chip. A NAND output of 0 selects the RAM chip for read or write operation, since CE2 is already 1 because of its connection to +5V. Fig. 18 shows the interfacing of IC 6264 with the 8085.

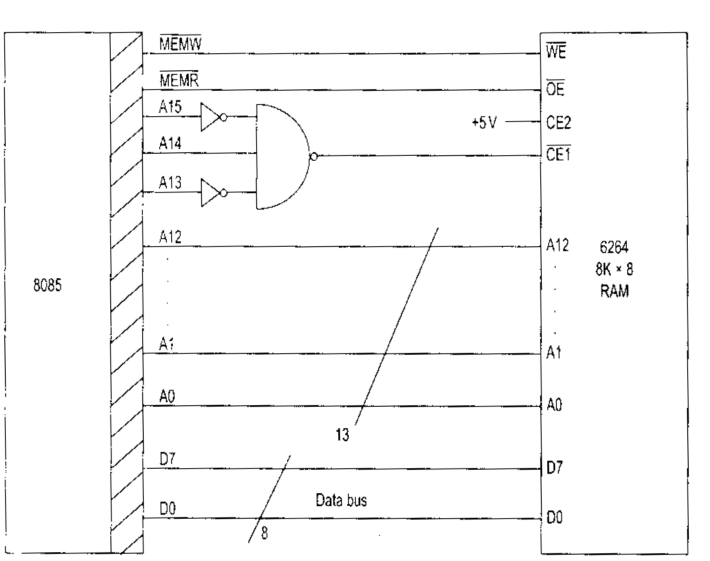


Fig. 18 Interfacing 6264 IC with the 8085

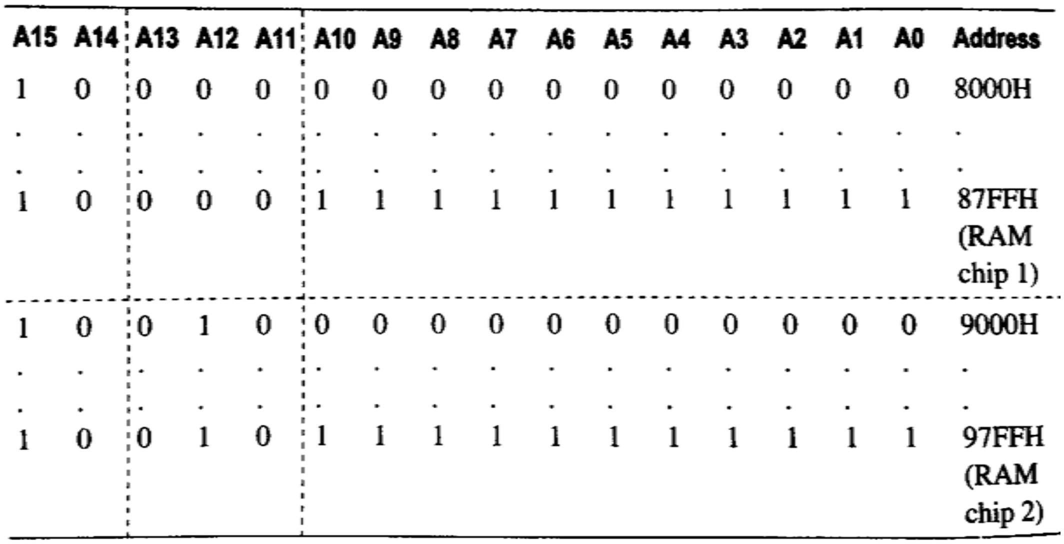
Ex: Interface two 6116 ICs with the 8085 using 74LS138 decoder such that the starting addresses assigned to them are 8000H and 9000H, respectively.

Specification of IC 6116:

- 2 K x 8 RAM
- 2 KB = 2^{11} bytes
- 11 address lines

6116 has 11 address lines and since 2 KB, therefore ending addresses of 6116 chip 1 is and chip 2 are 87FFH and 97FFH, respectively. Table 10 shows the address range of the two chips.

Table 10 Address range for IC 6116



A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Address
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	87FFH (RAM chip 1)
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	9000H
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	97FFH (RAM chip 2)

Interfacing:

- Fig. 19 shows the interfacing.
- A0 – A10 lines of 8085 are connected to 11 address lines of the RAM chips.
- Three address lines of 8085 having specific value for a particular RAM are connected to the three select inputs (C, B and A) of 74LS138 decoder.
- Table 10 shows that A13=A12=A11=0 for the address assigned to RAM 1 and A13=0, A12=1 and A11=0 for the address assigned to RAM 2.
- Remaining lines of 8085 which are constant for the address range assigned to the two RAM are connected to the enable inputs of decoder.
- When 8085 places any address between 8000H and 87FFH in the address bus, the select inputs C, B and A of the decoder are all 0. The Y0 output of the decoder is also 0, selecting RAM 1.
- When 8085 places any address between 9000H and 97FFH in the address bus, the select inputs C, B and A of the decoder are 0, 1 and 0. The Y2 output of the decoder is also 0, selecting RAM 2.

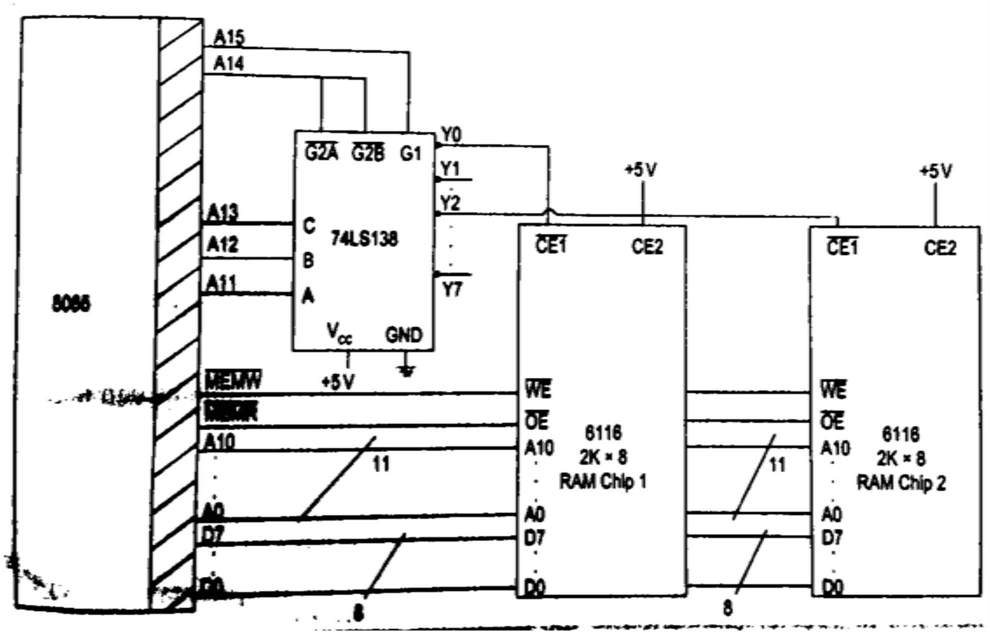


Fig. 19 Interfacing two 6116 RAM chips using 74LS138 decoder

3. PERIPHERAL MAPPED I/O INTERFACING

In this method, the I/O devices are treated differently from memory chips. The control signals I/O read ($\overline{IOR}$) and I/O write ($\overline{IOW}$), which are derived from the $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals of the 8085, are used to activate input and output devices, respectively. Generation of these control signals is shown in Fig. 20. Table 11 shows the status of $IO/\overline{M}$, $\overline{RD}$ and $\overline{WR}$ signals during I/O read and I/O write operation.

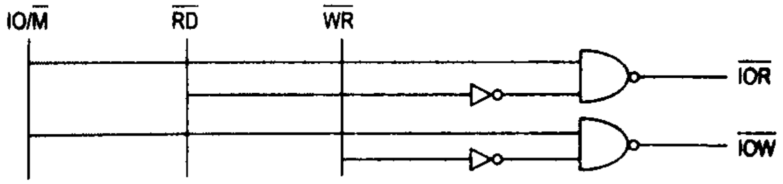


Fig. 20 Generation of $\overline{IOR}$ and $\overline{IOW}$ signals

IN instruction is used to access input device and OUT instruction is used to access output device. Each I/O device is identified by a unique 8-bit address assigned to it. Since the control signals used to access input and output devices are different, and all I/O device use 8-bit address, a maximum of 256 (2^8) input devices and 256 output devices can be interfaced with 8085.

Table 11 Status of $\overline{IOR}$ and $\overline{IOW}$ signals in 8085.

$IO/\overline{M}$	$\overline{RD}$	$\overline{WR}$	$\overline{IOR}$	$\overline{IOW}$	Operation
1	0	1	0	1	I/O read operation
1	1	0	1	0	I/O write operation
0	X	X	1	1	Memory read or write operation

Ex: Interface an 8-bit DIP switch with the 8085 such that the address assigned to the DIP switch if F0H.

IN instruction is used to get data from DIP switch and store it in accumulator. Steps involved in the execution of this instruction are:

i. Address F0H is placed in the lines A0 – A7 and a copy of it in lines A8 – A15.
ii. The $\overline{IOR}$ signal is activated ($\overline{IOR}$ = 0), which makes the selected input device to place its data in the data bus.
iii. The data in the data bus is read and store in the accumulator.

Fig. 21 shows the interfacing of DIP switch.

A7	A6	A5	A4	A3	A2	A1	A0	
1	1	1	1	0	0	0	0	= F0H

A0 – A7 lines are connected to a NAND gate decoder such that the output of NAND gate is 0. The output of NAND gate is ORed with the $\overline{IOR}$ signal and the output of OR gate is connected to $\overline{1G}$ and $\overline{2G}$ of the 74LS244. When 74LS244 is enabled, data from the DIP switch is placed on the data bus of the 8085. The 8085 read data and store in the accumulator. Thus data from DIP switch is transferred to the accumulator.

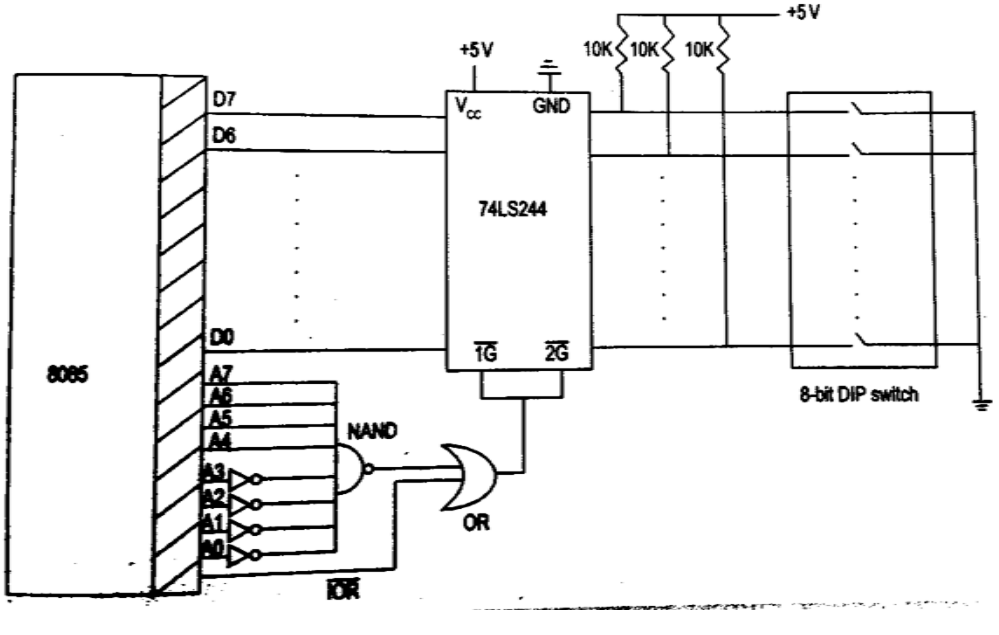


Fig. 21 interfacing of 8-bit DIP switch with 8085

4. MEMORY MAPPED I/O INTERFACING

In memory-mapped I/O, each input or output device is treated as if it is a memory location. The $\overline{MEMR}$ and $\overline{MEMW}$ control signals are used to activate the devices. Each input or output device is identified by unique 16-bit address, similar to 16-bit address assigned to memory location. All memory related instruction like LDA 2000H, LDAX B, MOV A, M can be used.

Since the I/O devices use some of the memory address space of 8085, the maximum memory capacity is lesser than 64 KB in this method.

Ex: Interface an 8-bit DIP switch with the 8085 using logic gates such that the address assigned to it is F0F0H.

Since a 16-bit address has to be assigned to a DIP switch, the memory-mapped I/O technique must be used. Using LDA F0F0H instruction, the data from the 8-bit DIP switch can be transferred to the accumulator. The steps involved are:

i. The address F0F0H is placed in the address bus A0 – A15.
ii. The $\overline{\text{MEMR}}$ signal is made low for some time.
iii. The data in the data bus is read and stored in the accumulator.

Fig. 22 shows the interfacing diagram.

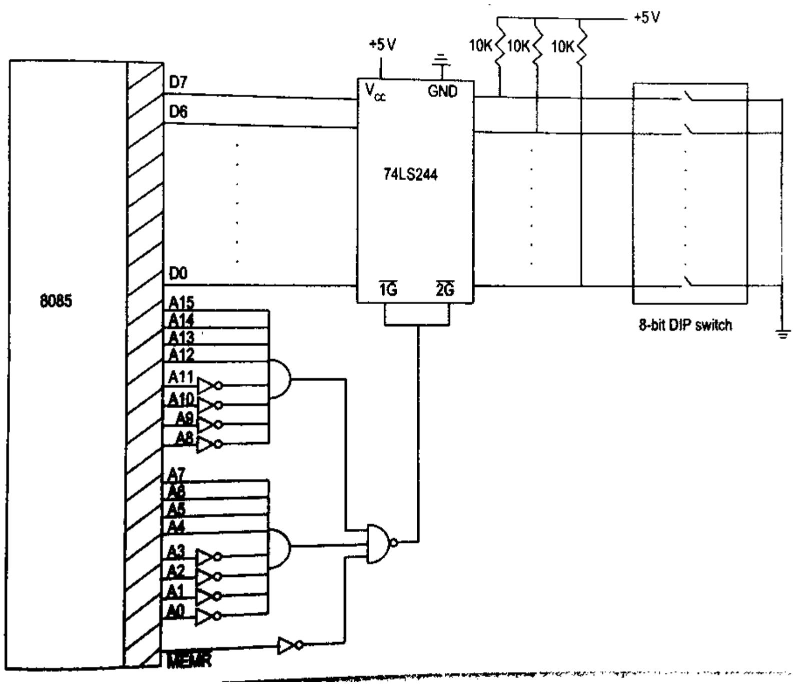


Fig. 22 Interfacing 8-bit DIP switch with 8085

When 8085 executes the instruction LDA F0F0H, it places the address F0F0H in the address lines A0 – A15 as:

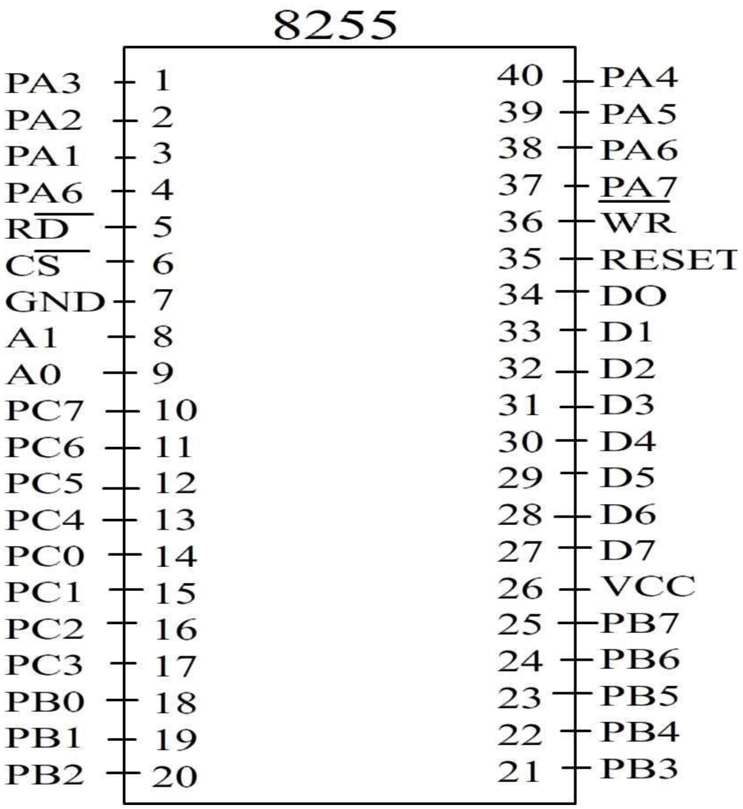
A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	= F0F0H

The address lines are connected to AND gates. The output of these gates along with $\overline{\text{MEMR}}$ signal are connected to a NAND gate, so that when the address F0F0H is placed in the address bus and $\overline{\text{MEMR}}$ = 0 its output becomes 0, thereby enabling the buffer 74LS244. The data from the DIP switch is placed in the 8085 data bus. The 8085 reads the data from the data bus and stores it in the accumulator.

INTEL 8255: (Programmable Peripheral Interface)

The 8255A is a general purpose programmable I/O device designed for use with Intel microprocessors. It consists of three 8-bit bidirectional I/O ports (24I/O lines) that can be configured to meet different system I/O needs. The three ports are PORT A, PORT B & PORT C. Port A contains one 8-bit output latch/buffer and one 8-bit input buffer. Port B is same as PORT A or PORT B. However, PORT C can be split into two parts PORT C lower (PC_0-PC_3) and PORT C upper (PC_7-PC_4) by the control word. The three ports are divided in two groups Group A (PORT A and upper PORT C) Group B (PORT B and lower PORT C). The two groups can be programmed in three different modes. In the first mode (mode 0), each group may be programmed in either input mode or output mode (PORT A, PORT B, PORT C lower, PORT C upper). In mode 1, the second's mode, each group may be programmed to have 8-lines of input or output (PORT A or PORT B) of the remaining 4-lines (PORT C lower or PORT C upper) 3-lines are used for hand shaking and interrupt control signals. The third mode of operation (mode 2) is a bidirectional bus mode which uses 8-line (PORT A only for a bidirectional bus and five lines (PORT C upper 4 lines and borrowing one from other group) for handshaking.

The 8255 is contained in a 40-pin package, whose pin out is shown below:



PIN Names

RESET – Reset input

$\overline{CS}$ - Chip selected

$\overline{RD}$ - Read input

$\overline{WR}$ - Write input

A_0 A_1 – Port Address

PA_7 – PA_0 – PORT A

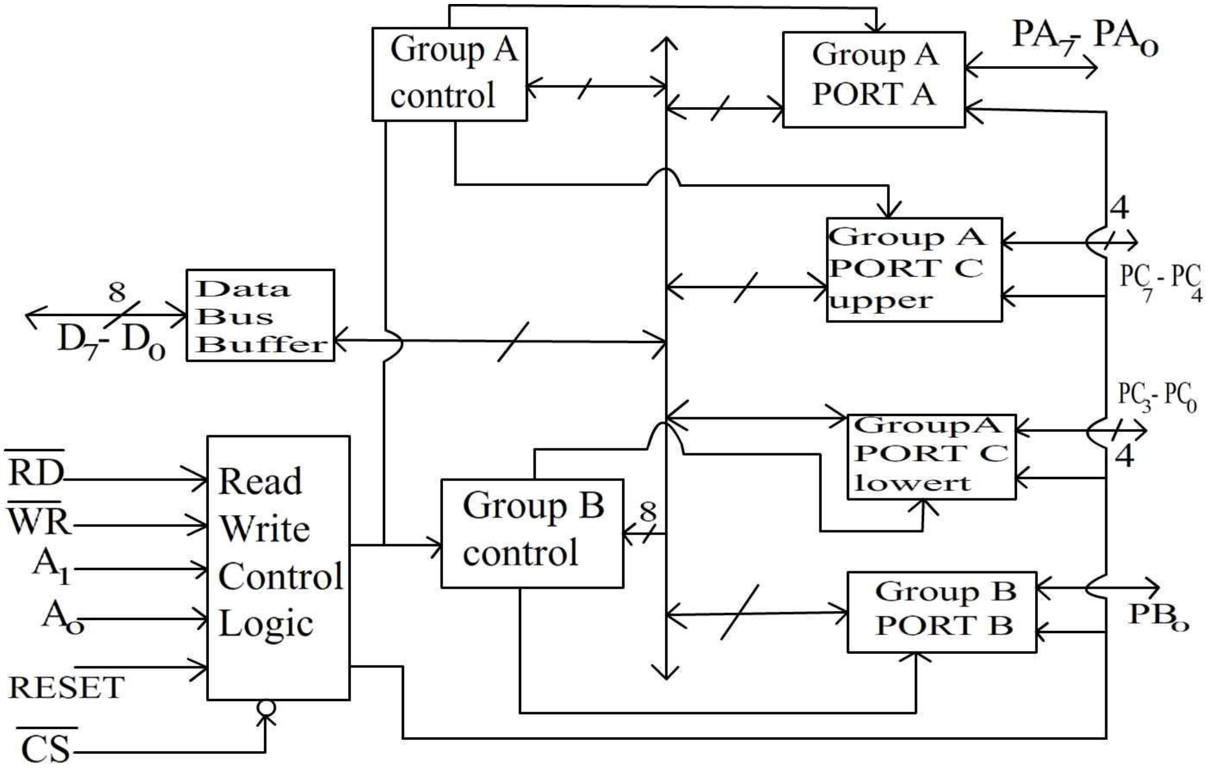
PB_7 – PB_0 – PORT B

PC_7 – PC_0 – PORT C

VCC - +5v

GND - Ground

The block diagram is shown below:



Functional Description:

This support chip is a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. It is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer:

It is a tri-state 8-bit buffer used to interface the chip to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer. The data lines are connected to BDB of μp.

Read/Write and logic control:

The function of this block is to control the internal operation of the device and to control the transfer of data and control or status words. It accepts inputs from the CPU address and control buses and in turn issues command to both the control groups.

$\overline{CS}$ Chip Select:

A low on this input selects the chip and enables the communication between the 8255 A & the CPU. It is connected to the output of address decode circuitry to select the device when it $\overline{RD}$ (Read). A low on this input enables the 8255 to send the data or status information to the CPU on the data bus.

$\overline{WR}$ (Write):

A low on this input pin enables the CPU to write data or control words into the 8255 A.

A_1, A_0 port select:

These input signals, in conjunction with the $\overline{RD}$ and $\overline{WR}$ inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus (A_0 and A_1).

Following Table gives the basic operation,

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Input operation
0	0	0	1	0	PORT A → Data bus
0	1	0	1	0	PORT B → Data bus
1	0	0	1	0	PORT C → Data bus
0	0	1	0	0	Output operation Data bus → PORT A
0	1	1	0	0	Data bus → PORT B
1	0	1	0	0	Data bus → PORT C
1	1	1	0	0	Data bus → control

All other states put data bus into tri-state/illegal condition.

RESET:

A high on this input pin clears the control register and all ports (A, B & C) are initialized to input mode. This is connected to RESET OUT of 8255. This is done to prevent destruction of circuitry connected to port lines. If port lines are initialized as output after a power up or reset, the port might try to output into the output of a device connected to same inputs might destroy one or both of them.

PORTs A, B and C:

The 8255A contains three 8-bit ports (A, B and C). All can be configured in a variety of functional characteristic by the system software.

PORTA:

One 8-bit data output latch/buffer and one 8-bit data input latch.

PORT B:

One 8-bit data output latch/buffer and one 8-bit data input buffer.

PORT C:

One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signals inputs in conjunction with ports A and B.

Group A & Group B control:

The functional configuration of each port is programmed by the system software. The control words outputted by the CPU configure the associated ports of the each of the two groups. Each control block accepts command from Read/Write content logic receives control words from the internal data bus and issues proper commands to its associated ports.

Control Group A – Port A & Port C upper

Control Group B – Port B & Port C lower

The control word register can only be written into No read operation if the control word register is allowed.

Operation Description:

Mode selection:

There are three basic modes of operation that can be selected by the system software.

Mode 0: Basic Input/output

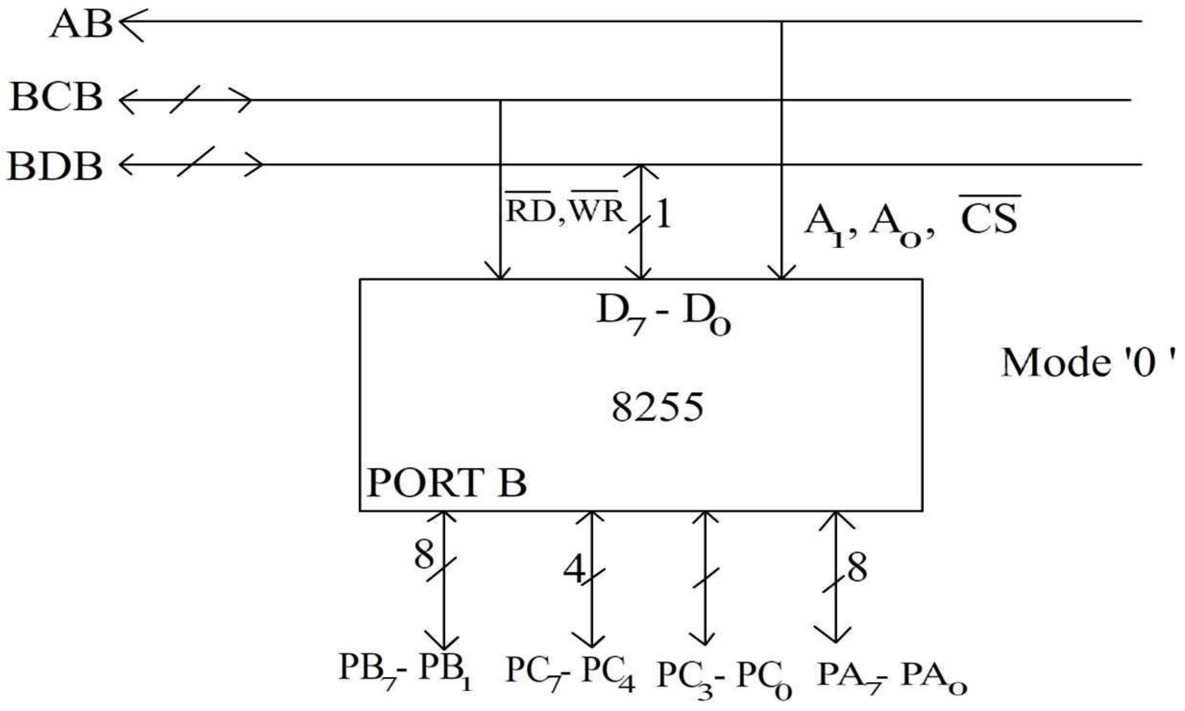
Mode 1: Strobes Input/output

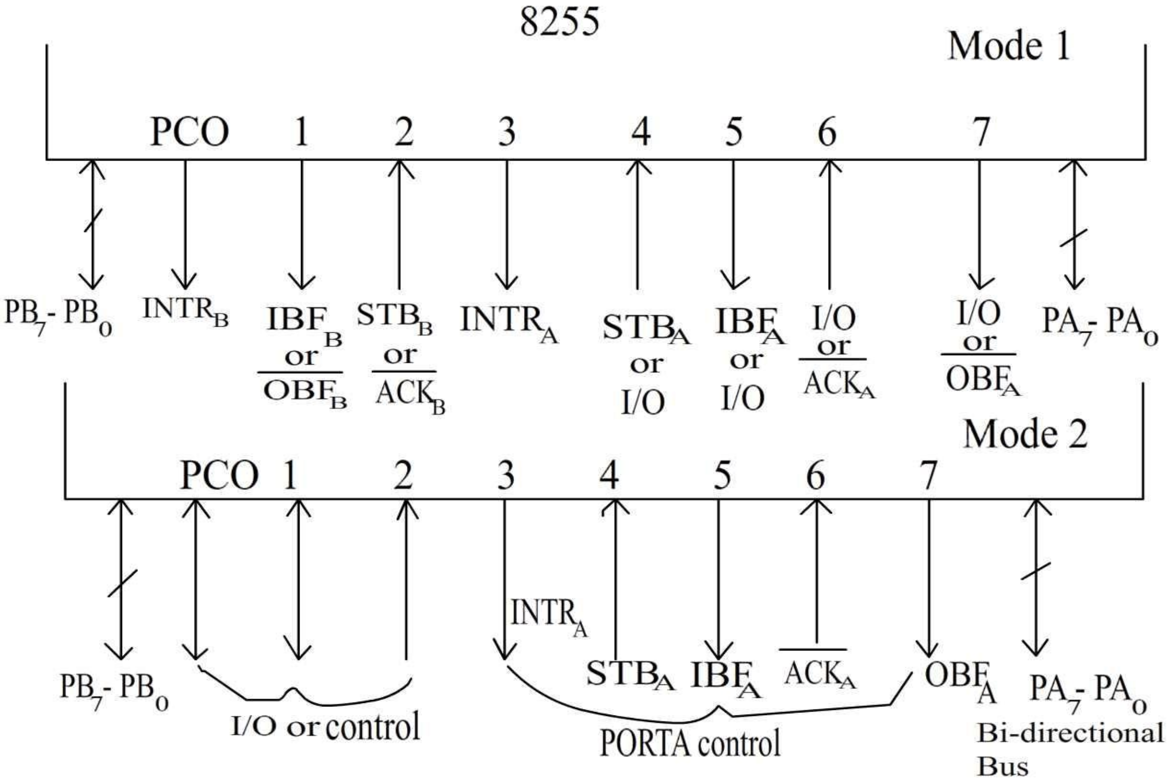
Mode 2: Bi-direction bus.

When the reset input goes HIGH all poets are set to mode'0' as input which means all 24 lines are in high impedance state and can be used as normal input. After the reset is removed the 8255A remains in the input mode with no additional initialization. During the execution of the program any of the other modes may be selected using a single output instruction.

The modes for PORT A & PORT B can be separately defined, while PORT C is divided into two portions as required by the PORT A and PORT B definitions. The ports are thus divided into two groups Group A & Group B. All the output register, including the status flip-flop will be reset whenever the mode is changed. Modes of the two group may be combined for any desired I/O operation e.g. Group A in mode '1' and group B in mode '0'.

The basic mode definitions with bus interface and the mode definition format are given in fig (a) & (b),


AB
BCB
BDB
$\overline{RD}$, $\overline{WR}$
1
A_1, A_o, $\overline{CS}$
D_7 - D_O
8255
PORT B
Mode '0 '
8
4
8
PB_7- PB_1
PC_7- PC_4
PC_3- PC_0
PA_7- PA_o

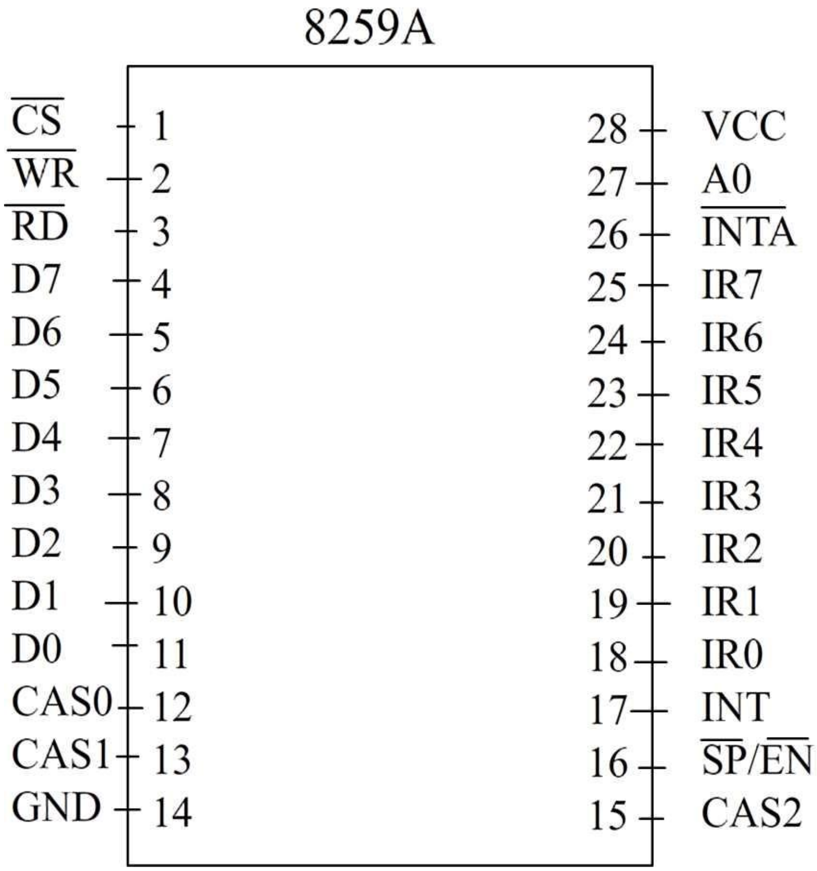

8255
Mode 1
PCO
1
2
3
4
5
6
7
PB_7- PB_0
$INTR_B$
IBF_B or $\overline{OBF_B}$
STB_B or $\overline{ACK_B}$
$INTR_A$
STB_A or I/O
IBF_A or I/O
I/O or $\overline{ACK_A}$
I/O or $\overline{OBF_A}$
PA_7- PA_0
Mode 2
PCO
1
2
3
4
5
6
7
$INTR_A$
STB_A
IBF_A
$\overline{ACK_A}$
OBF_A
PB_7- PB_0
I/O or control
PORTA control
PA_7- PA_0
Bi-directional Bus

INTEL 8259A Programmable Interrupt Controller

The 8259A is a programmable interrupt controller designed to work with Intel microprocessor 8080 A, 8085, 8086, 8088. The 8259 A interrupt controller can

1) Handle eight interrupt inputs. This is equivalent to providing eight interrupt pins on the processor in place of one INTR/INT pin.
2) Vector an interrupt request anywhere in the memory map. However, all the eight interrupt are spaced at the interval of either four or eight location. This eliminates the major drawback, 8085 interrupt, in which all interrupts are vectored to memory location on page 00_H.
3) Resolve eight levels of interrupt priorities in a variety of modes.
4) Mask each interrupt request individually.
5) Read the status of pending interrupts, in service interrupts, and masked interrupts.
6) Be set up to accept either the level triggered or edge triggered interrupt request.
7) Mine 8259 as can be cascade in a master slave configuration to handle 64 interrupt inputs.

The 8259 A is contained in a 28-element in line package that requires only a compatible with 8259. The main difference between the two is that the 8259 A can be used with Intel 8086/8088 processor. It also induces additional features such as level triggered mode, buffered mode and automatic end of interrupt mode. The pin diagram and interval block diagram is shown below:



The pins are defined as follows:

$\overline{CS}$: Chip select

To access this chip, $\overline{CS}$ is made low. A LOW on this pin enables $\overline{RD}$ & $\overline{WR}$ communication between the CPU and the 8259A. This pin is connected to address bus through the decoder logic circuits. INTA functions are independent of $\overline{CS}$

$\overline{WR}$:

A low on this pin. When $\overline{CS}$ is low enables the 8259 A to accept command words from CPU.

$\overline{RD}$:

A low on this pin when $\overline{CS}$ is low enables these 8259 A to release status on to the data bus for the CPU. The status in dudes the contents of IMR, ISR or TRR register or a priority level.

D_7-D_0:

Bidirectional data bus control status and interrupt in a this bus. This bus is connected to BDB of 8085.

CAS_0-CAS_2:

Cascade lines: The CAS lines form a private 8259A bus to control a multiple 8259A structure ie to identify a particular slave device. These pins are outputs of a master 8259A and inputs for a slave 8259A.

$\overline{SP}$/ $\overline{EN}$: Salve program/enable buffer:

This is a dual function pin. It is used as an input to determine whether the 8259A is to a master ($\overline{SP}$/ $\overline{EN}$ = 1) or as a slave ($\overline{SP}$/ $\overline{EN}$ = 0). It is also used as an output to disable the data bus transceivers when data are being transferred from the 8259A to the CPU. When in buffered mode, it can be used as an output and when not in the buffered mode it is used as an input.

INT:

This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the CPU, thus it is connected to the CPU's interrupt pin (INTR).

$\overline{INTA}$:

Interrupt: Acknowledge. This pin is used to enable 8259A interrupt vector data on the data bus by a sequence of interrupt request pulses issued by the CPU.

IR_0-IR_7:

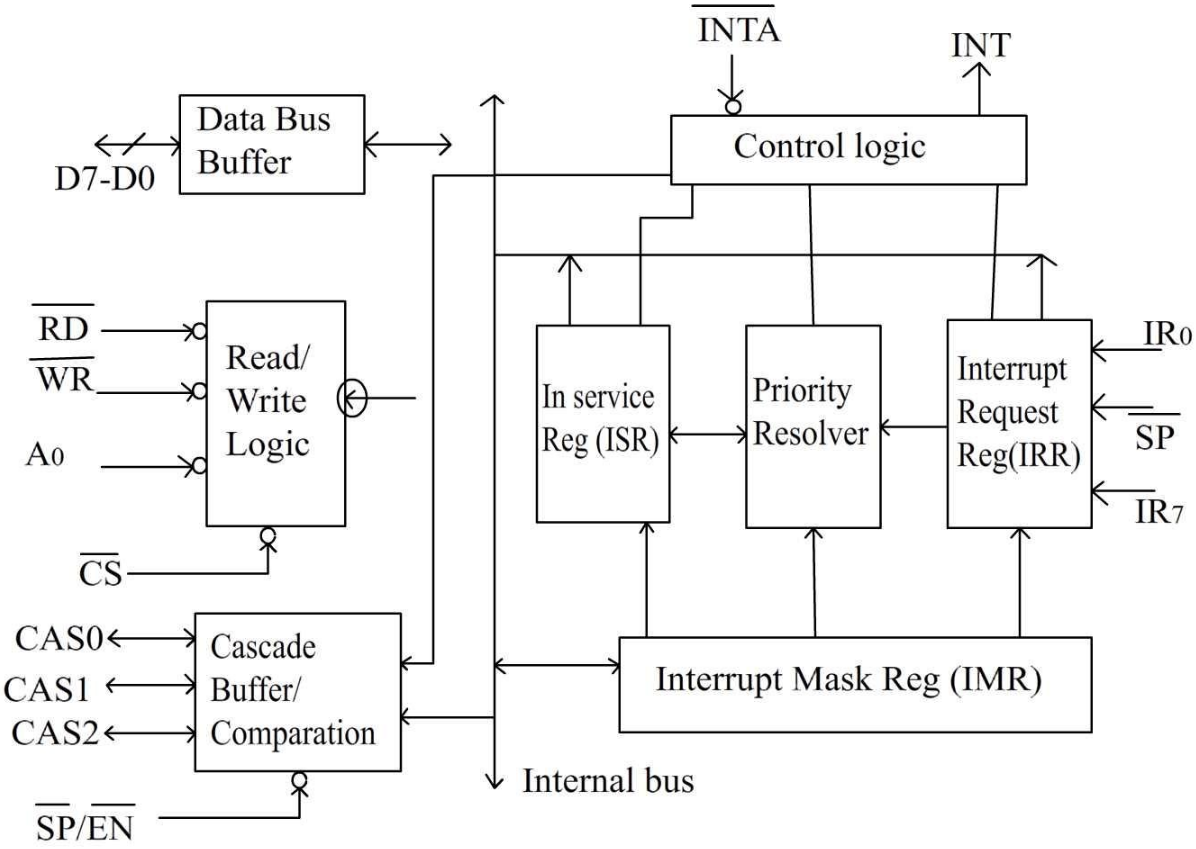
Interrupt Requests: Asynchronous interrupt inputs. An interrupt request is executed by raising an IR input (low to high), and holding it high until it is acknowledged. (Edge triggered mode).or just by a high level on an IR input (levels triggered mode).

A_0:

A_0 address line: This pin acts in conjunction with the $\overline{RD}$, $\overline{WR}$ & $\overline{CS}$ pins. It is used by the 8259A to send various command words from the CPU and to read the status. If is connected to the CPU A_0 address line. Two addresses must be reserved in the I/O address space for each 8259 in the system.

Functional Description:

The 8259 A has eight interrupt request inputs, TR2 IR0. The 8259 A uses its INT output to interrupt the 8085A via INTR pin. The 8259A receives interrupt acknowledge pulses from the μp at its $\overline{INTA}$ input. Vector address used by the 8085 A to transfer control to the service subroutine of the interrupting device, is provided by the 8259 A on the data bus. The 8259A is a programmable device that must be initialized by command words sent by the. After initialization the 8259 A mode of operation can be changed by operation command words from the.



The descriptions of various blocks are,

Data bus buffer:

This 3- state, bidirectional 8-bit buffer is used to interface the 8259A to the system data bus. Control words and status information are transferred through the data bus buffer.

Read/Write & control logic:

The function of this block is to accept OUTPUT commands from the CPU. It contains the initialization command word (ICW) register and operation command word (OCW) register which store the various control formats for device operation. This function block also allows the status of 8159A to be transferred to the data bus.

Interrupt request register (IRR):

IRR stores all the interrupt inputs that are requesting service. Basically, it keeps track of which interrupt inputs are asking for service. If an interrupt input is unmasked, and has an interrupt signal on it, then the corresponding bit in the IRR will be set.

Interrupt mask register (IMR):

The IMR is used to disable (Mask) or enable (Unmask) individual interrupt inputs. Each bit in this register corresponds to the interrupt input with the same number. The IMR operation on the IRR. Masking of higher priority input will not affect the interrupt request lines of lower priority. To unmask any interrupt the corresponding bit is set ‘0’.

In service register (ISR):

The in service registers keeps tracks of which interrupt inputs are currently being serviced. For each input that is currently being serviced the corresponding bit will be set in the in service register. Each of these 3-reg can be read as status reg.

Priority Resolver:

This logic block determines the priorities of the set in the IRR. The highest priority is selected and strobed into the corresponding bit of the ISR during $\overline{INTA}$ pulse.

Cascade buffer/comparator:

This function blocks stores and compare the IDS of all 8259A’s in the reg. The associated 3-I/O pins (CAS0-CAS2) are outputs when

8259A is used a master. Master and are inputs when 8259A is used as a slave. As a master, the 8259A sends the ID of the interrupting slave device onto the cas2-cas0. The slave thus selected will send its pre-programmed subroutine address on to the data bus during the next one or two successive $\overline{INTA}$ pulses.

8257: Direct Memory Access Controller

The *Direct Memory Access* or DMA mode of data transfer is the fastest amongst all the modes of data transfer. In this mode, the device may transfer data directly to/from memory without any interference from the CPU. The device requests the CPU (through a DMA controller) to hold its data, address and control bus, so that the device may transfer data directly to/from memory.

The DMA data transfer is initiated only after receiving HLDA signal from the CPU. Intel's 8257 is a four channel DMA controller designed to be interfaced with their family of microprocessors. The 8257, on behalf of the devices, requests the CPU for bus access using local bus request input i.e. HOLD in minimum mode. In maximum mode of the microprocessor RQ/GT pin is used as bus request input.

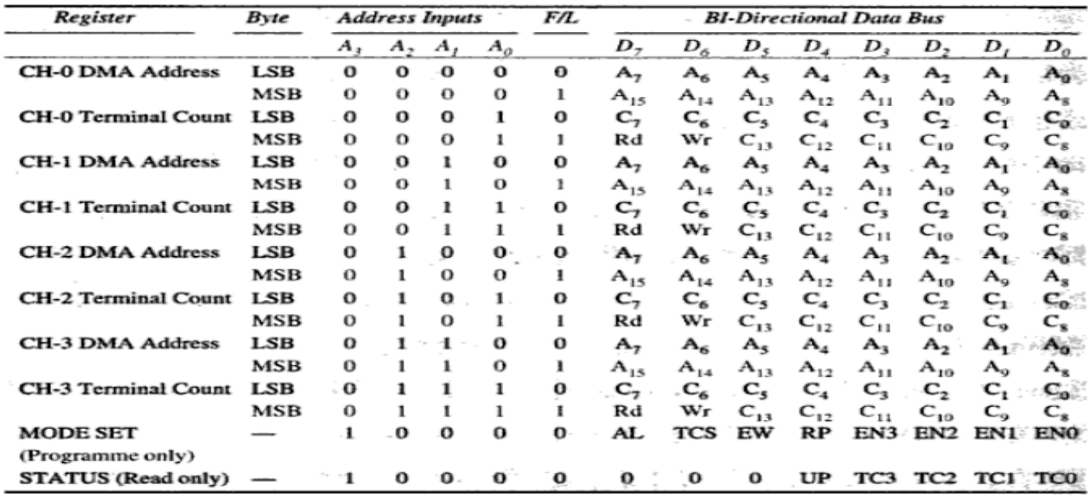
On receiving the HLDA signal (in minimum mode) or RQ/GT signal (in maximum mode) from the CPU, the requesting devices gets the access of the bus, and it completes the required number of DMA cycles for the data transfer and then hands over the control of the bus back to the CPU.

Internal Architecture of 8257

The internal architecture of 8257 is shown in figure. The chip support four DMA channels, i.e. four peripheral devices can independently request for DMA data transfer through these channels at a time. The DMA controller has 8-bit internal data buffer, a read/write unit, a control unit, a priority resolving unit along with a set of registers.

Register Organization of 8257

Table *8257 Register Selection*



Register	*Byte*	*Address Inputs*				*F/L*	*BI-Directional Data Bus*							
		A_3	A_2	A_1	A_0		D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
CH-0 DMA Address	LSB	0	0	0	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	0	0	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-0 Terminal Count	LSB	0	0	0	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	0	0	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
CH-1 DMA Address	LSB	0	0	1	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	0	1	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-1 Terminal Count	LSB	0	0	1	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	0	1	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
CH-2 DMA Address	LSB	0	1	0	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	1	0	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-2 Terminal Count	LSB	0	1	0	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	1	0	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
CH-3 DMA Address	LSB	0	1	1	0	0	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
	MSB	0	1	1	0	1	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8
CH-3 Terminal Count	LSB	0	1	1	1	0	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0
	MSB	0	1	1	1	1	Rd	Wr	C_{13}	C_{12}	C_{11}	C_{10}	C_9	C_8
MODE SET (Programme only)	—	1	0	0	0	0	AL	TCS	EW	RP	EN3	EN2	EN1	EN0
STATUS (Read only)	—	1	0	0	0	0	0	0	0	UP	TC3	TC2	TC1	TC0

The 8257 performs the DMA operation over four independent DMA channels. Each of four channels of 8257 has a pair of two 16-bit registers, viz. DMA *address* *register* and *terminal count register*.

There are two common registers for all the channels, namely, *mode set register* and *status register*. Thus there are a total of ten registers. The CPU selects one of these ten registers using address lines Ao-A3. Table shows how the Ao-A3 bits may be used for selecting one of these registers.

DMA Address Register

Each DMA channel has one DMA address register. The function of this register is to store the address of the starting memory location, which will be accessed by the DMA channel. Thus the starting address of the memory block which will be accessed by the device is first loaded in the DMA address register of the channel.

The device that wants to transfer data over a DMA channel, will access the block of the memory with the starting address stored in the DMA Address Register.

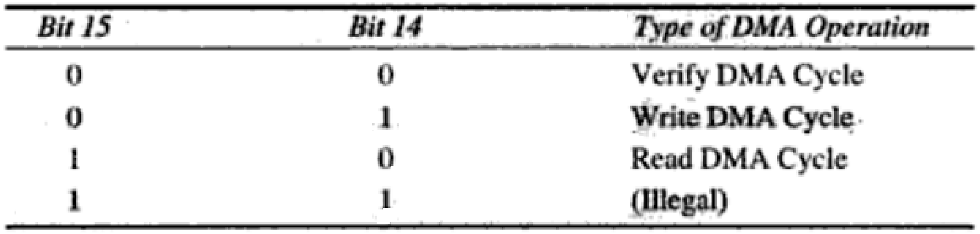
Terminal Count Register

Each of the four DMA channels of 8257 has one terminal count register (TC). This 16-bit register isused for ascertaining that the data transfer through a DMA channel ceases or stops after the required number of DMA cycles. The low order 14-bits of the terminal count register are initialised with the binary equivalent of the number of required DMA cycles minus one.

After each DMA cycle, the terminal count register content will be decremented by one and finally it becomes zero after the required number of DMA cycles are over. The bits 14 and 15 of this register indicate the type of the DMA operation (transfer). If the device wants to write data into the memory, the DMA operation is called DMA write operation. Bit 14 of the register in this case will be set to one and bit 15 will be set to zero.

Table gives detail of DMA operation selection and corresponding bit configuration of bits 14 and 15 of the TC register.

Table *DMA Operation Selection Using A_{15}/RD and A_{14}/WR*



Bit 15	*Bit 14*	*Type of DMA Operation*
0	0	Verify DMA Cycle
0	1	Write DMA Cycle
1	0	Read DMA Cycle
1	1	(Illegal)

Mode Set Register

The mode set register is used for programming the 8257 as per the requirements of the system. The function of the mode set register is to enable the DMA channels individually and also to set the various modes of operation.

The DMA channel should not be enabled till the DMA address register and the terminal count register contain valid information, otherwise, an unwanted DMA request may initiate a DMA cycle, probably destroying the valid memory data. The bits Do-D3 enable one of the four DMA channels of 8257. for example, if Do is ‘1’, channel 0 is enabled. If

bit 4 is set, rotating priority is enabled, otherwise, the normal, i.e. fixed priority is enabled.

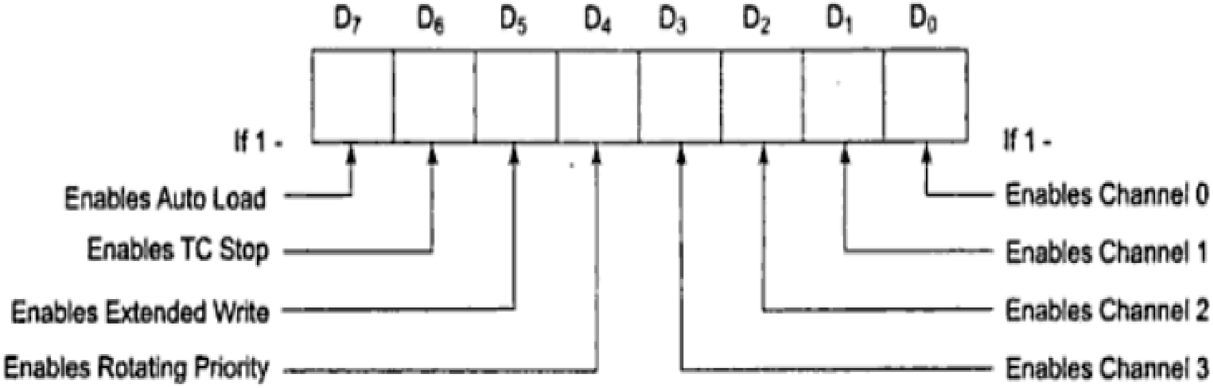


Fig. *Bit Definitions of the Mode Set Register*

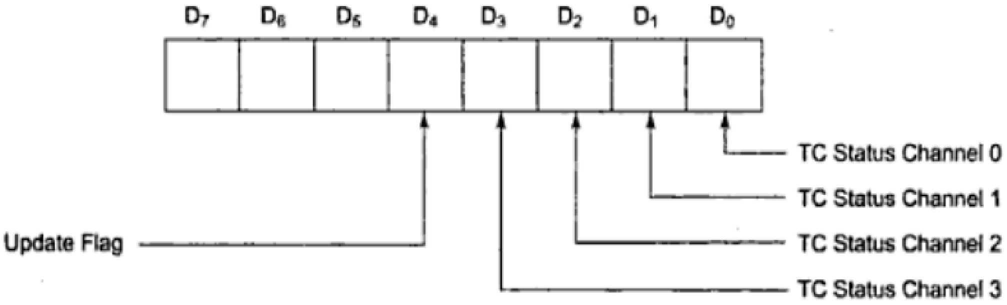
If the TC STOP bit is set, the selected channel is disabled after the *terminal count* condition is reached, and it further prevents any DMA cycle on the channel. To enable the channel again, this bit must be reprogrammed. If the TC STOP bit is programmed to be zero, the channel is not disabled, even after the count reaches zero and further request are allowed on the same channel.

The auto load bit, if set, enables channel 2 for the repeat block chaining operations, without immediate software intervention between the two successive blocks. The channel 2 registers are used as usual, while the channel 3 registers are used to store the block reinitialisation parameters, i.e. the DMA starting address and terminal count. After the first block is transferred using DMA, the channel 2 registers are reloaded with the corresponding channel 3 registers for the next block transfer, if the *update* flag is set. The extended write bit, if set to '1', extends the duration of MEMW and IOW signals by activating them earlier, this is useful in interfacing the peripherals with different access times.

If the peripheral is not accessed within the stipulated time, it is expected to give the 'NOT READY' indication to 8257, to request it to add one or more wait states in the DMA CYCLE. The mode set register can only be written into.

Status Register

The status register of 8257 is shown in figure. The lower order 4-bits of this register contain the terminal count status for the four individual channels. If any of these bits is set, it indicates that the specific channel has reached the terminal count condition.



If 1, the respective channel has reached the terminal count condition.

These bits remain set till either the status is read by the CPU or the 8257 is reset. The update flag is not affected by the read operation. This flag can only be cleared by resetting 8257 or by resetting the auto load bit of the mode set register. If the update flag is set, the contents of the channel 3 registers are reloaded to the corresponding registers of channel 2 whenever the channel 2 reaches a terminal count condition, after transferring one block and the next block is to be transferred using the autoload feature of 8257.

The update flag is set every time, the channel 2 registers are loaded with contents of the channel 3 registers. It is cleared by the completion of the first DMA cycle of the new block. This register can only read.

Data Bus Buffer, Read/Write Logic, Control Unit and Priority Resolver

The 8-bit. Tristate, bidirectional buffer interfaces the internal bus of 8257 with the external system bus under the control of various control signals.

In the slave mode, the read/write logic accepts the I/O Read or I/O Write signals, decodes the Ao-A3 lines and either writes the contents of the data bus to the addressed internal register or reads the contents of the selected register depending upon whether IOW or IOR signal is activated.

In master mode, the read/write logic generates the IOR and IOW signals to control the data flow to or from the selected peripheral. The control logic controls the sequences of operations and generates the required control signals like AEN, ADSTB, MEMR, MEMW, TC and MARK along with the address lines A4-A7, in master mode. The priority resolver resolves the priority of the four DMA channels depending upon whether normal priority or rotating priority is programmed.

Signal Description of 8257

DRQo-DRQ3 :

These are the four individual channel DMA request inputs, used by the peripheral devices for requesting the DMA services. The DRQo has the highest priority while DRQ**3** has the lowest one, if the fixed priority mode is selected.

DACKo-DACK3 :

These are the active-low DMA acknowledge output lines which inform the requesting peripheral that the request has been honoured and the bus is relinquished by the CPU. These lines may act as strobe lines for the requesting devices.

Do-D7:

These are bidirectional, data lines used to interface the system bus with the internal data bus of 8257. These lines carry command words to 8257 and status word from 8257, in slave mode, i.e. under the control of CPU.

The data over these lines may be transferred in both the directions. When the 8257 is the bus master (master mode, i.e. not under CPU control), it uses Do-D7 lines to send higher byte of the generated address to the latch. This address is further latched using ADSTB signal. the address is transferred over Do-D7 during the first clock cycle of the DMA cycle. During the rest of the period, data is available on the data bus.

IOR:

This is an active-low bidirectional tristate input line that acts as an input in the slave mode. In slave mode, this input signal is used by the CPU to read internal registers of 8257.this line acts output in master mode. In master mode, this signal is used to read data from a peripheral during a memory write cycle.

IOW :

This is an active low bidirection tristate line that acts as input in slave mode to load the contents of the data bus to the 8-bit mode register or upper/lower byte of a 16-bit DMA address register or terminal count register. In the master mode, it is a control output that loads the data to a peripheral during DMA memory read cycle (write to peripheral).

CLK:

This is a clock frequency input required to derive basic system timings for the internal operation of 8257.

RESET :

This active-high asynchronous input disables all the DMA channels by clearing the mode register and tristates all the control lines.

Ao-A3:

These are the four least significant address lines. In slave mode, they act as input which select one of the registers to be read or written. In the master mode, they are the four least significant memory address output lines generated by 8257.

CS:

This is an active-low chip select line that enables the read/write operations from/to 8257, in slave mode. In the master mode, it is automatically disabled to prevent the chip from getting selected (by CPU) while performing the DMA operation.

A4-A7 :

This is the higher nibble of the lower byte address generated by 8257 during the master mode of DMA operation.

READY:

This is an active-high asynchronous input used to stretch memory read and write cycles of 8257 by inserting wait states. This is used while interfacing slower peripherals..

HRQ:

The hold request output requests the access of the system bus. In the non-cascaded 8257 systems, this is connected with HOLD pin of CPU. In the cascade mode, this pin of a slave is connected with a DRQ input line of the master 8257, while that of the master is connected with HOLD input of the CPU.

HLDA :

The CPU drives this input to the DMA controller high, while granting the bus to the device. This pin is connected to the HLDA output of the CPU. This input, if high, indicates to the DMA controller that the bus has been granted to the requesting peripheral by the CPU.

MEMR:

This active –low memory read output is used to read data from the addressed memory locations during DMA read cycles.

MEMW :

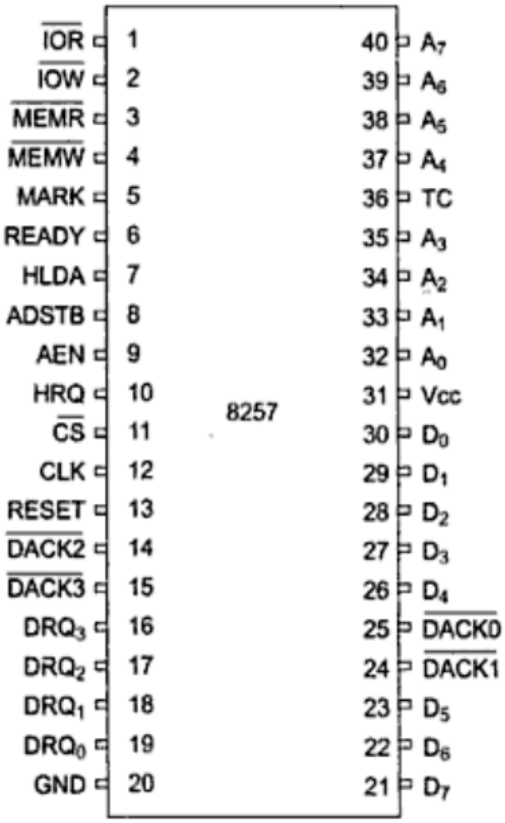
This active-low three state output is used to write data to the addressed memory location during DMA write operation.

ADST :

This output from 8257 strobes the higher byte of the memory address generated by the DMA controller into the latches.

AEN:

This output is used to disable the system data bus and the control the bus driven by the CPU, this may be used to disable the system address and data bus by using the enable input of the bus drivers to inhibit the non-DMA devices from responding during DMA operations. If the 8257 is I/O mapped, this should be used to disable the other I/O devices, when the DMA controller addresses is on the address bus.



Pin diagram of 8257

TC:

Terminal count output indicates to the currently selected peripherals that the present DMA cycle is the last for the previously programmed data block. If the TC STOP bit in the mode set register is set, the selected channel will be disabled at the end of the DMA cycle.

The TC pin is activated when the 14-bit content of the terminal count register of the selected channel becomes equal to zero. The lower order 14 bits of the terminal count register are to be programmed with a 14-bit equivalent of (n-1), if n is the desired number of DMA cycles.

MARK :

The modulo 128 mark output indicates to the selected peripheral that the current DMA cycle is the 128th cycle since the previous MARK output. The mark will be activated after each 128 cycles or integral multiples of it from the beginning if the data block (the first DMA cycle), if the total number of the required DMA cycles (n) is completely divisible by 128.

Vcc :

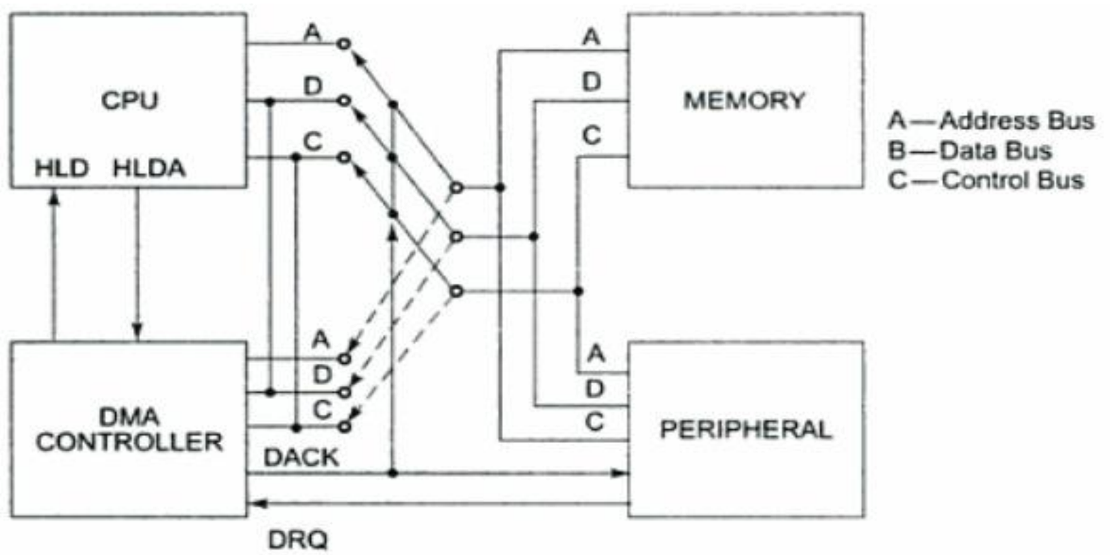
This is a +5v supply pin required for operation of the circuit.

GND :

This is a return line for the supply (ground pin of the IC).

Interfacing 8257 with 8086

Once a DMA controller is initialised by a CPU property, it is ready to take control of the system bus on a DMA request, either from a peripheral or itself (in case of memory-to-memory transfer). The DMA controller sends a HOLD request to the CPU and waits for the CPU to assert the HLDA signal. The CPU relinquishes the control of the bus before asserting the HLDA signal.



A conceptual implementation of the system is shown in Figure

Once the HLDA signal goes high, the DMA controller activates the DACK signal to the requesting peripheral and gains the control of the system bus. The DMA controller is the sole master of the bus, till the DMA operation is over. The CPU remains in the HOLD status (all of its signals are tristate except HOLD and HLDA), till the DMA controller is the master of the bus.

In other words, the DMA controller interfacing circuit implements a switching arrangement for the address, data and control busses of the memory and peripheral subsystem from/to the CPU to/from the DMA controller.

GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

V.L.S.I.

For 5th Semester

ELECTRONICS & TELECOMMUNICATION

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Miss. Jasmita Sahu

(Lecturer in Electronics & Telecommunication Engineering)

VLSI ENGINEERING (3-1-0)

MODULE 1:

Issues and Challenges in VLSI Design:

Integration density and performance of integrated circuits have gone through an astounding revolution in the last couple of decades. In the 1960s, Gordon Moore, then with Fairchild Corporation and later cofounder of Intel, predicted that the number of transistors that can be integrated on a single die would grow exponentially with time. This prediction, later called *Moore's law*, has proven to be amazingly visionary. Its validity is best illustrated with the aid of a set of graphs. Figure 1 plots the integration density of both logic IC's and memory as a function of time. As can be observed, integration complexity doubles approximately every 1 to 2 years. As a result, memory density has increased by more than a thousand fold since 1970.

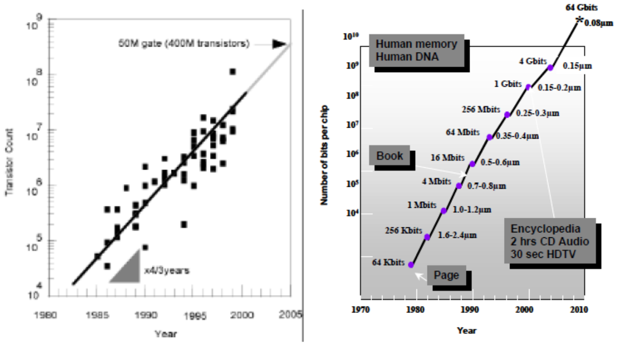


Figure 1: Evolution of integration complexity of logic ICs and memories as a function of time.

Typically used abstraction levels in digital circuit design are, in order of increasing abstraction, the device, circuit, gate, functional module (e.g., adder) and system levels (e.g., processor), as illustrated in Figure 2. A semiconductor device is an entity with a very complex behavior.

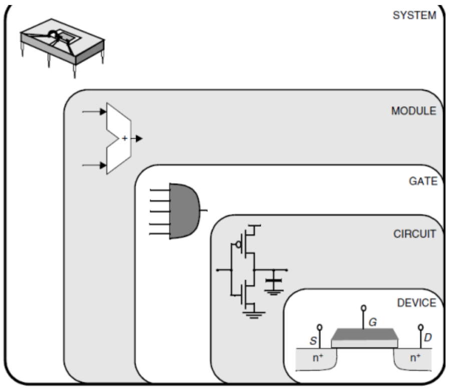
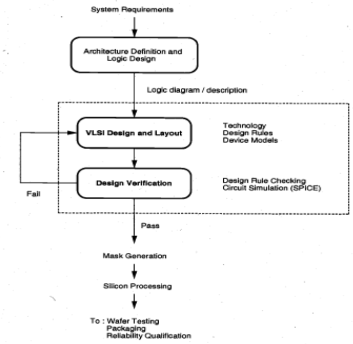


Figure 2 Design abstraction levels in digital circuits.

VLSI Design Flow:



VLSI Fabrication Technology:

The CMOS process requires a large number of steps, each of which consists of a sequence of basic operations. A number of these steps and/or operations are executed very repetitively in the course of the manufacturing process.

The Silicon Wafer

The base material for the manufacturing process comes in the form of a single-crystalline, lightly doped *wafer*. These wafers have typical diameters between 4 and 12 inches (10 and 30 cm, respectively) and a thickness of at most 1 mm, and are obtained by cutting a single crystal ingot into thin slices.

Photolithography:

In each processing step, a certain area on the chip is masked out using the appropriate optical mask so that a desired processing step can be selectively applied to the remaining regions. The processing step can be any of a wide range of tasks including oxidation, etching, metal and polysilicon deposition, and ion implantation. The technique to accomplish this selective masking, called *photolithography*, is applied throughout the manufacturing process. Figure 3 gives a graphical overview of the different operations involved in a typical photolithographic process. The following steps can be identified:

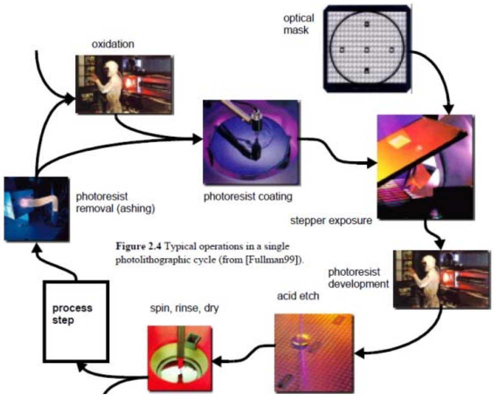


Figure 3: Graphical Overview

1. Oxidation layering — this optional step deposits a thin layer of SiO2 over the complete wafer by exposing it to a mixture of high-purity oxygen and hydrogen at approximately 1000°C. The oxide is used as an insulation layer and also forms transistor gates.
2. Photoresist coating — a light-sensitive polymer (similar to latex) is evenly applied while spinning the wafer to a thickness of approximately 1 mm. This material is originally soluble in an organic solvent, but has the property that the polymers crossoxidation link when exposed to light, making the affected regions insoluble. A photoresist of this type is called negative. A positive photoresist has the opposite properties;

originally insoluble, but soluble after exposure. By using both positive and negative resists, a single mask can sometimes be used for two steps, making complementary regions available for processing. Since the cost of a mask is increasing quite rapidly with the scaling of technology, a reduction of the number of masks is surely of high priority.

3. Stepper exposure — a glass mask (or reticle), containing the patterns that we want to transfer to the silicon, is brought in close proximity to the wafer. The mask is opaque in the regions that we want to process, and transparent in the others (assuming a negative photo resist). The glass mask can be thought of as the negative of one layer of the microcircuit. The combination of mask and wafer is now exposed to ultra-violet light. Where the mask is transparent, the photo resist becomes insoluble.
4. Photoresist development and bake — the wafers are developed in either an acid or base solution to remove the non-exposed areas of photoresist. Once the exposed photoresist is removed, the wafer is "soft-baked" at a low temperature to harden the remaining photoresist.
5. Acid Etching — material is selectively removed from areas of the wafer that are not covered by photoresist. This is accomplished through the use of many different types of acid, base and caustic solutions as a function of the material that is to be removed. Much of the work with chemicals takes place at large wet benches where special solutions are prepared for specific tasks. Because of the dangerous nature of some of these solvents, safety and environmental impact is a primary concern.
6. Spin, rinse, and dry — a special tool (called SRD) cleans the wafer with deionized water and dries it with nitrogen. The microscopic scale of modern semiconductor devices means that even the smallest particle of dust or dirt can destroy the circuitry. To prevent this from happening, the processing steps are performed in ultra-clean rooms where the number of dust particles per cubic foot of air ranges between 1 and 10. Automatic wafer handling and robotics are used whenever possible. This explains why the cost of a state-of-the-art fabrication facility easily ranges in the multiple billions of dollars. Even then, the wafers must be constantly cleaned to avoid contamination, and to remove the left-over of the previous process steps.
7. Photoresist removal (or ashing) — a high-temperature plasma is used to selectively remove the remaining photoresist without damaging device layers.

CMOS Process Flow:

The process starts with the definition of the active regions; this is the regions where transistors will be constructed. All other areas of the die will be covered with a thick layer of silicon dioxide (SiO2), called the field oxide. This oxide acts as the insulator between neighboring devices, and is either grown (as in the process of Figure 2.1), or deposited in etched trenches (Figure 2.2) — hence the name trench insulation. Further insulation is provided by the addition of a reverse-biased np-diode, formed by adding an extra p+ region, called the channel-stop implant (or field implant) underneath the field oxide. Next, lightly doped p- and n-wells are formed through ion implantation.

To construct an NMOS transistor in a p-well, heavily doped n-type source and drain regions are implanted (or diffused) into the lightly doped p-type substrate. A thin layer of SiO2, called the gate oxide, separates the region between the source and drain, and is itself covered by conductive polycrystalline silicon (or polysilicon, for short).

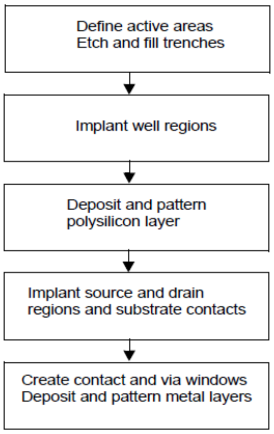
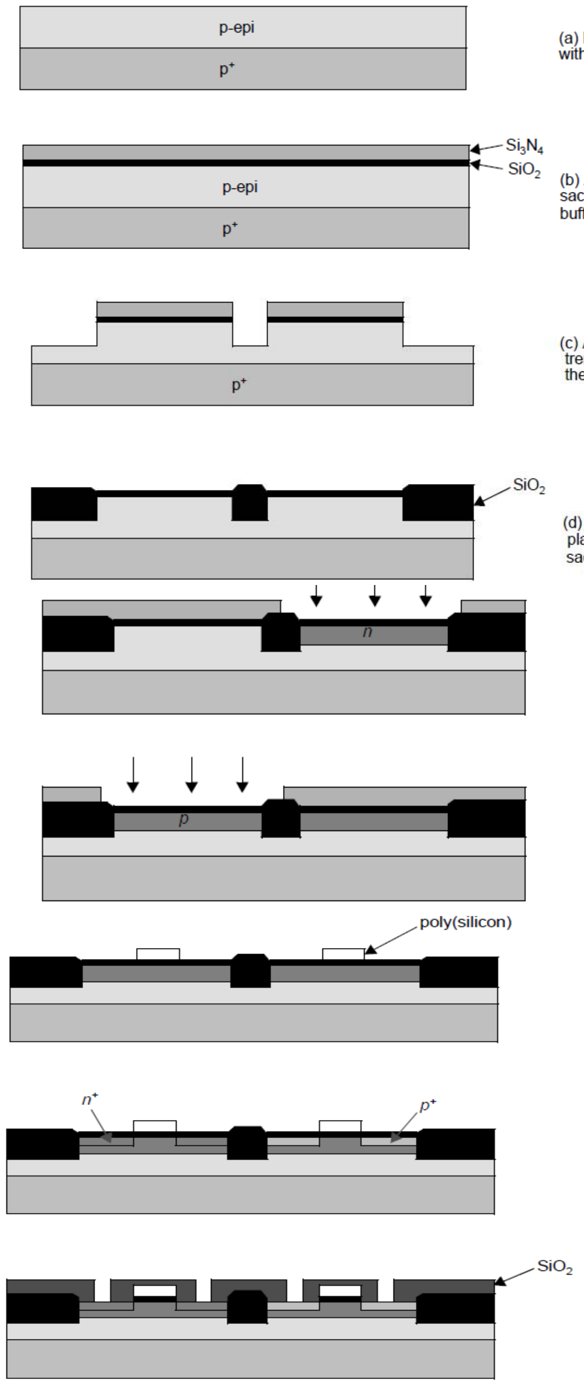


Figure 4: Simplified process sequence for the manufacturing of a ndual-well CMOS circuit.



(a) Base material: p+ substrate with p-epi layer

(b) After deposition of gate-oxide sacrificial nitride (acts as a buffer layer)

(c) After plasma etch of insulating trenches using the inverse of the active area mask

(d) After trench filling, CMP planarization, and removal of sacrificial nitride

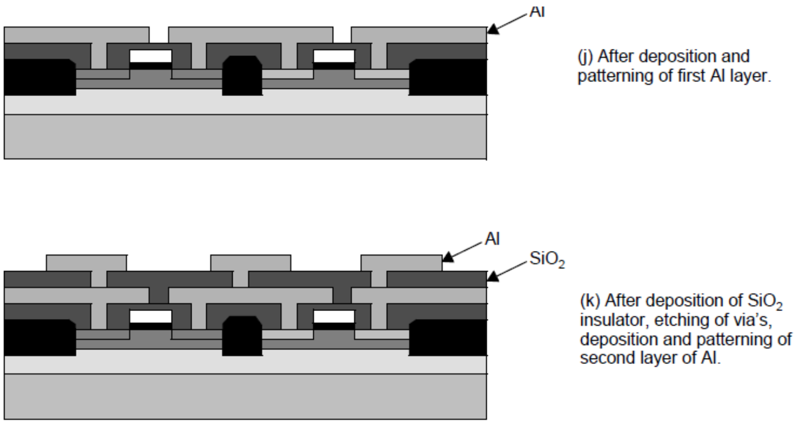
(e) After *n*-well and V_{Tp} adjust implants

(f) After *p*-well and V_{Tn} adjust implants

(g) After polysilicon deposition and etch

(h) After n^+ source/drain and p^+ source/drain implants. These steps also dope the polysilicon.

(i) After deposition of SiO_2 insulator and contact hole etch.



(j) After deposition and patterning of first Al layer.

(k) After deposition of SiO$_2$ insulator, etching of via's, deposition and patterning of second layer of Al.

Figure 5: Process flow for the fabrication of an NMOS and a PMOS transistor in a dual-well CMOS process.

Design Rules:

The goal of defining a set of design rules is to allow for a ready translation of a circuit concept into an actual geometry in silicon. The design rules act as the interface or even the contract between the circuit designer and the process engineer. Circuit designers in general want tighter, smaller designs, which lead to higher performance and higher circuit density. The process engineer, on the other hand, wants a reproducible and high-yield process. Design rules are, consequently, a compromise that attempts to satisfy both sides. The design rules provide a set of guidelines for constructing the various masks needed in the patterning process. They consist of minimum-width and minimum-spacing constraints and requirements between objects on the same or on different layers.

The fundamental unity in the definition of a set of design rules is the *minimum line width*. It stands for the minimum mask dimension that can be safely transferred to the semiconductor material. In general, the minimum line width is set by the resolution of the patterning process, which is most commonly based on optical lithography.

Mead and Conway [Mead80], defines all rules as a function of a single parameter, most often called **λ based rule**. The rules are chosen so that a design is easily ported over a cross section of industrial processes. Scaling of the minimum dimension is accomplished by simply changing the value of λ. This results in a *linear scaling* of all dimensions. For a given process, λ is set to a specific value, and all design dimensions are consequently translated into absolute numbers.
This approach, while attractive, suffers from some disadvantages

- Linear scaling is only possible over a limited range of dimensions (for instance, between 0.25 mm and 0.18 mm). When scaling over larger ranges, the relations between the different layers tend to vary in a nonlinear way that cannot be adequately covered by the linear scaling rules.

As circuit density is a prime goal in industrial designs, most semiconductor companies tend to use ***micron rules*,** which express the design rules in absolute dimensions and can therefore exploit the features of a given process to a maximum degree. Scaling and porting designs between technologies under these rules is more demanding and has to be performed either manually or using advanced CAD tools.

Layout Design:

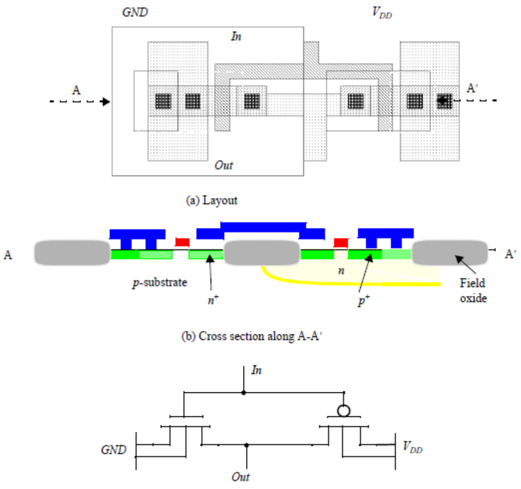
The layer concept translates the intractable set of masks currently used in CMOS into a simple set of conceptual layout levels that are easier to visualize by the circuit designer. From a designer's viewpoint, all CMOS designs are based on the following entities:

1. Substrates and/or wells, being p-type (for NMOS devices) and n-type (for PMOS)
2. Diffusion regions (n+ and p+) defining the areas where transistors can be formed.These regions are often called the active areas. Diffusions of an inverse type are needed to implement contacts to the wells or to the substrate. These are called select regions.

3. One or more polysilicon layers, which are used to form the gate electrodes of the transistors (but serve as interconnect layers as well).
4. A number of metal interconnect layers.
5. Contact and via layers to provide interlayer connections.

A layout consists of a combination of polygons, each of which is attached to a certain layer. The functionality of the circuit is determined by the choice of the layers, as well as the interplay between objects on different layers. For instance, an MOS transistor is formed by the cross section of the diffusion layer and the polysilicon layer. An interconnection between two metal layers is formed by a cross section between the two metal layers and an additional contact layer. To visualize these relations, each layer is assigned a standard color (or stipple pattern for a black-and-white representation).

Layout Exmple:



(a) Layout

(b) Cross section along A-A'

(c) Circuit diagram

Figure 6 A detailed layout examples, including vertical process cross section and circuit diagram.

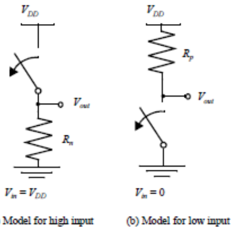
Module-II

CMOS Inverter:

The inverter is truly the nucleus of all digital designs. Once its operation and properties are clearly understood, designing more intricate structures such as NAND gates, adders, multipliers, and microprocessors is greatly simplified. The electrical behavior of these complex circuits can be almost completely derived by extrapolating the results obtained for inverters. The analysis of inverters can be extended to explain the behavior of more complex gates such as NAND, NOR, or XOR, which in turn form the building blocks for modules such as multipliers and processors.

The Static CMOS Inverter — An Intuitive Perspective

Figure 7 shows the circuit diagram of a static CMOS inverter. Its operation is readily understood with the aid of the simple switch model of the MOS transistor, the transistor is nothing more than a switch with an infinite off resistance (for $|VGS| < |VT|$), and a finite on-resistance (for $|VGS| > |VT|$). This leads to the following interpretation of the inverter. When V_{in} is high and equal to V_{DD}, the NMOS transistor is on, while the PMOS is off. This yields the equivalent circuit of Figure 8. A direct path exists between V_{out} and the ground node, resulting in a steady-state value of 0 V. On the other hand, when the input voltage is low (0 V), NMOS and PMOS transistors are off and on, respectively. The equivalent circuit of Figure 9 shows that a path exists between VDD and V_{out}, yielding a high output voltage. The gate clearly functions as an inverter.



(a) Model for high input (b) Model for low input

Figure 7.Switch models of CMOS inverter.

A number of other important properties of static CMOS can be derived from this switch level view:

- The high and low output levels equal *VDD* and *GND*, respectively; in other words, the voltage swing is equal to the supply voltage. This results in high noise margins.
- The logic levels are not dependent upon the relative device sizes, so that the transistors can be minimum size. Gates with this property are called *ratioless.* This is in contrast with *ratioed logic*, where logic levels are determined by the relative dimensions of the composing transistors.
- In steady state, there always exists a path with finite resistance between the output and either *VDD* or *GND*. A well-designed CMOS inverter, therefore, has a *low output impedance*, which makes it less sensitive to noise and disturbances. Typical values of the output resistance are in kW range.
- The *input resistance* of the CMOS inverter is extremely high, as the gate of an MOS transistor is a virtually perfect insulator and draws no dc input current. Since the input node of the inverter only connects to transistor gates, the steady-state input current is nearly zero. A single inverter can theoretically drive an infinite number of gates (or have an infinite fan-out) and still be functionally operational; however, increasing the fan-out also increases the propagation delay, as will become clear below. So, although fan-out does not have any effect on the steady-state behavior, it degrades the transient response.
- No direct path exists between the supply and ground rails under steady-state operating conditions (this is, when the input and outputs remain constant). The absence of current flow (ignoring leakage currents) means that the gate does not consume anystatic power.

MOS Device Model with Sub-micron Effects:

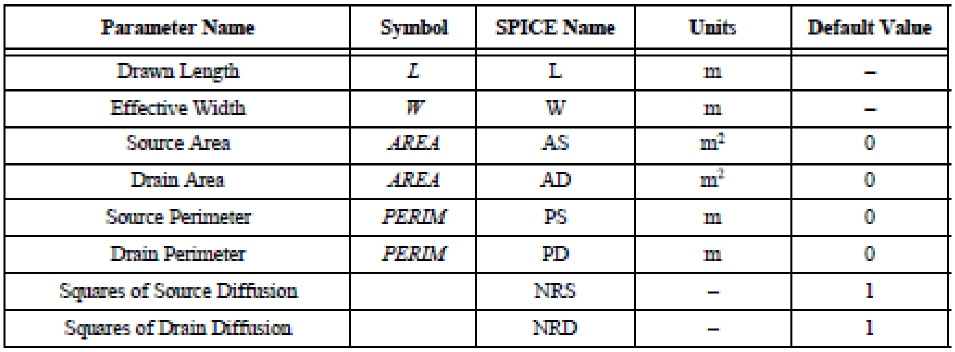
The complexity of the behavior of the short-channel MOS transistor and its many parasitic effects has led to the development of a wealth of models for varying degrees of accuracy and computing efficiency. In general, more accuracy also means more complexity and, hence, an increased run time. In this section, we briefly discuss the characteristics of the more popular MOSFET models, and describe how to instantiate a MOS transistor in a circuit description.

SPICE Models:

SPICE has three built-in MOSFET models, selected by the LEVEL parameter in the model card. Unfortunately, all these models have been rendered obsolete by the progression to short-channel devices. They should only be used for first-order analysis, and we therefore limit ourselves to a short discussion of their main properties.

- The LEVEL 1 SPICE model implements the *Shichman-Hodges model*, which is based on the square law long-channel expressions, derived earlier in this chapter. It does not handle short-channel effects.
- The LEVEL 2 model is a geometry-based model, which uses detailed device physics to define its equations. It handles effects such as velocity saturation, mobility degradation, and drain-induced barrier lowering. Unfortunately, including all 3D-effects of an advanced submicron process in a pure physics-based model becomes complex and inaccurate.
- LEVEL 3 is a semi-empirical model. It relies on a mixture of analytical and empirical expressions, and uses measured device data to determine its main parameters. It works quite well for channel lengths down to 1 μm.

In response to the inadequacy of the built-in models, SPICE vendors and semi-conductor manufacturers have introduced a wide range of accurate, but proprietary models.



Parameter Name	Symbol	SPICE Name	Units	Default Value
Drawn Length	*L*	L	m	–
Effective Width	*W*	W	m	–
Source Area	*AREA*	AS	m^2	0
Drain Area	*AREA*	AD	m^2	0
Source Perimeter	*PERIM*	PS	m	0
Drain Perimeter	*PERIM*	PD	m	0
Squares of Source Diffusion		NRS	–	1
Squares of Drain Diffusion		NRD	–	1

Table 1: Some SPICE transistor parameters

VTC Parameters (DC Characteristics):

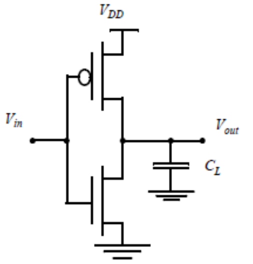


Figure 8: Static CMOS inverter

The Voltage-Transfer Characteristic:

Assume now that a logical variable *in* serves as the input to an inverting gate that produces the variable *out*. The electrical function of a gate is best expressed by its *voltage-transfer characteristic* (VTC) (sometimes called the *DC transfer characteristic*), which plots the output voltage as a function of the input voltage $V_{out} = f(V_{in})$. An example of an inverter VTC is shown in Figure 9. The high and low nominal voltages, V_{OH} and V_{OL}, can readily be identified—$V_{OH} = f(V_{OL})$ and $V_{OL} = f(V_{OH})$. Another point of interest of the VTC is the *gate or switching threshold voltage VM* (not to be confused with the threshold voltage of a transistor), that is defined as $V_M = f(V_M)$. V_M can also be found graphically at the intersection of the VTC curve and the line given by $V_{out} = V_{in}$. The gate threshold voltage presents the midpoint of the switching characteristics, which is obtained when the output of a gate is short-circuited to the input. This point will prove to be of particular interest when studying circuits with feedback (also called *sequential circuits*).

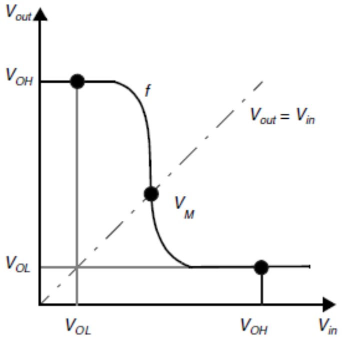
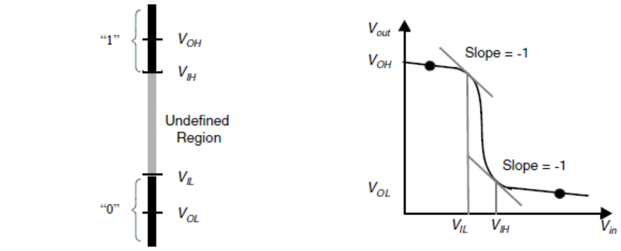


Figure 9:Inverter Voltage Transfer Characteristics(VTC)

Even if an ideal nominal value is applied at the input of a gate, the output signal often deviates from the expected nominal value. These deviations can be caused by noise or by the loading on the output of the gate (i.e., by the number of gates connected to the output signal). Figure 10(a) illustrates how a logic level is represented in reality by a range of acceptable voltages, separated by a region of uncertainty, rather than by nominal levels alone. The regions of acceptable high and low voltages are delimited by the V_{IH} and V_{IL} voltage levels, respectively. These represent by definition the points where the gain (= dV_{out} / dV_{in}) of the VTC equals −1 as shown in Figure 10(b). The region between V_{IH} and V_{IL} is called the *undefined region* (sometimes also referred to as *transition width,* or *TW*). Steady-state signals should avoid this region if proper circuit operation is to be ensured.



(a) Relationship between voltage and logic levels (b) Definition of V_{IH} and V_{IL}

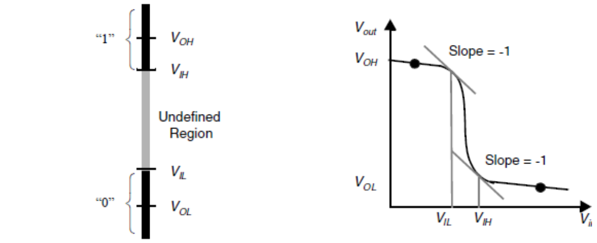
Figure 10: Mapping logic levels to the voltage domain.

Noise Margins:

For a gate to be robust and insensitive to noise disturbances, it is essential that the "0" and "1" intervals be as large as possible. A measure of the sensitivity of a gate to noise is given by the noise margins NM_L (*noise margin low*) and NM_H (*noise margin high*), which quantize the size of the legal "0" and "1", respectively, and set a fixed maximum threshold on the noise value:

$$NM_L = V_{IL} - V_{OL}$$
$$NM_H = V_{OH} - V_{IH}$$

The noise margins represent the levels of noise that can be sustained when gates are cascaded as illustrated in Figure 11. It is obvious that the margins should be larger than 0 for a digital circuit to be functional and by preference should be as large as possible.



(a) Relationship between voltage and logic levels (b) Definition of V_{IH} and V_{IL}

Figure 11: Mapping logic levels to the voltage domain.

CMOS Propagation Delay:

The input and output voltage waveforms of a typical inverter circuit are shown in Figure 12. The propagation delay times $\boldsymbol{\gamma_{PHL}}$ and γ_{pLH} determine the input-to-output signal delay during the high-to-low and low-to-high transitions of the output, respectively. By definition, $\boldsymbol{\gamma_{PHL}}$ is the time delay between the $V_{50\%}$-transition of the *rising* input voltage and the $V_{50\%}$ -transition of *the falling* output voltage. Similarly, γ_{pLH} is defined as the time delay between the $V_{50\%}$ transition of *the falling* input voltage and the $V_{50\%}$ transition of the *rising* output voltage.

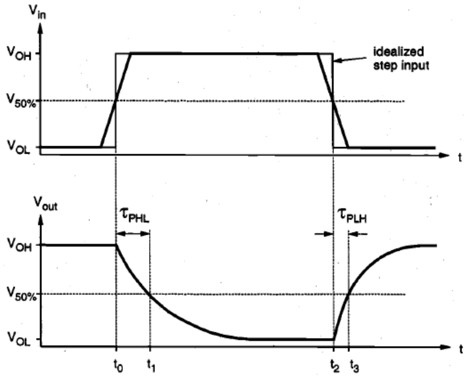


Figure 12: Input and output voltage waveforms of a typical inverter

Thus, the propagation delay times γ_{PHL} and γ_{PLH} are

$$\gamma_{PHL} = t3 - to$$
$$\gamma_{PLH} = t3 - t2$$

Calculation of parasitic capacitances:

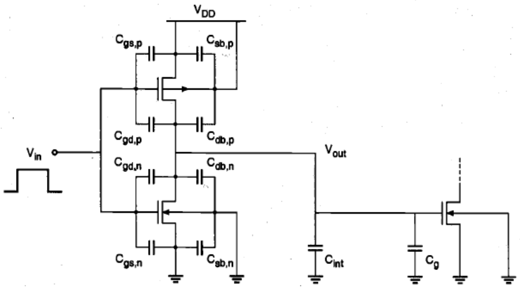


Figure 13: **Cascaded CMOS inverter stages.**

The problem of analyzing the output voltage waveform is fairly complicated, even for this relatively simple circuit, because a number of nonlinear, voltage-dependent capacitances are involved. To simplify the problem, we first combine the capacitancesseen in Fig. 6.1 into an equivalent *lumped* linear capacitance, connected between theoutput node of the inverter and the ground. This combined capacitance at the output node will be called the load capacitance, $\boldsymbol{C_{load}}$.

$$C_{load} = C_{gd},. + C_{gdp} + C_{db},,n + C_{dbp} + C_{int} + C_g$$

Layout of an Inverter:

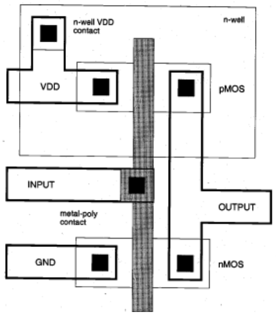


Figure 14: Complete mask layout of the CMOS inverter.

Switching Power Dissipation of CMOS Inverters:

The static power dissipation of the CMOS inverter is quite negligible. During switching events where the output load capacitance is alternatingly charged up and charged down, on the other hand, the CMOS inverter inevitably dissipates power. In the following section, we will derive the expressions for the dynamic power consumption of the CMOS inverter.Consider the simple CMOS inverter circuit shown in Fig. 15. We will assume that the input voltage is an ideal step waveform with negligible rise and fall times.

Typical input and output voltage waveforms and the expected load capacitor current waveform are shown in Fig. 16. When the input voltage switches from low to high, the pMOS transistor in the circuit is turned off, and the nMOS transistor starts conducting. During this phase, the output load capacitance C_{load} is being discharged through the nMOS transistor.

Thus, the capacitor current equals the instantaneous drain current of the nMOS transistor. When the input voltage switches from high to low, the nMOS transistor in the circuit is turned off, and the pMOS transistor starts conducting. During this phase, the output load capacitance C_{load} is being charged up through the pMOS transistor; therefore, the capacitor current equals the instantaneous drain current of the pMOS transistor.

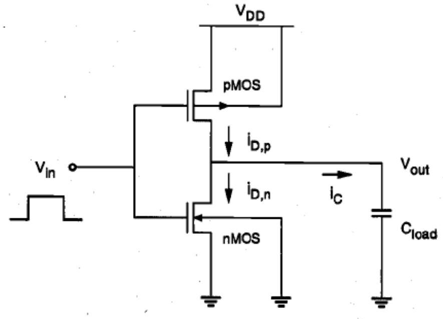


Figure 15: CMOS inverter used in the dynamic power-dissipation analysis.

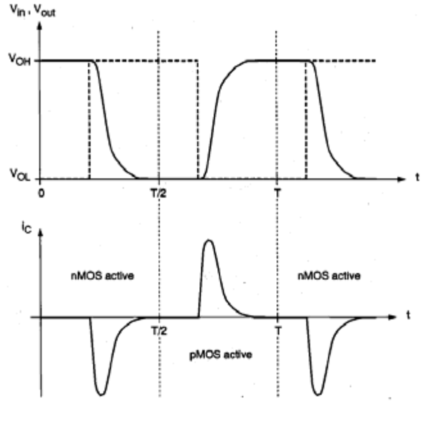


Figure 16: Typical input and output voltage waveforms and the capacitor current waveform during switching

Assuming periodic input and output waveforms, the average power dissipated by any device over one period can be found as follows:

$$P_{avg} = \frac{1}{T}\int_0^T V(t)i(t)dt$$

Since during switching, the nMOS transistor and the pMOS transistor in a CMOS inverter conduct current for one-half period each, the average power dissipation of the CMOS inverter can be calculated as the power required to charge up and charge down the output load capacitance.

By solving we get $P_{avg}=C_{load}\, V_{DD}^2 f$

Estimation of Interconnect Parasitics:

The classical approach for determining the switching speed of a logic gate is based on the assumption that the loads are mainly capacitive and lumped. We have examined the relatively simple delay models for inverters with purely capacitive load at the output node, which can be used to estimate the transient behavior of the circuit once the load is determined. The conventional delay estimation approaches seek to classify three main components of the output load, all of which are assumed to be purely capacitive, as: (i) internal parasitic capacitances of the transistors, (ii) interconnect (line) capacitances, and (iii) input capacitances of the fan-out gates. Of these three components, the load conditions imposed by the interconnection lines present serious problems, especially in submicron circuits.

Figure 17 shows a simple situation where an inverter is driving three other inverters, linked by interconnection lines of different length and geometry. If the load from each interconnection line can be approximated by a lumped capacitance, then the total load seen by the primary inverter is simply the sum of all capacitive components described above. In most cases, however, the load conditions imposed by the interconnection line are far from being simple. The line, itself a three-dimensional structure in metal and/or polysilicon, usually has a non-negligible resistance in addition to its capacitance. The (length/width) ratio of the wire usually dictates that the parameters are distributed, making the interconnect a true transmission line. Also, an interconnect is rarely isolated from other influences. In realistic conditions, the interconnection line is in very close proximity to a number of other lines, either on the same level or on different levels. The capacitive/inductive coupling and the signal interference between neighboring lines should also be taken into consideration for an accurate estimation of delay.

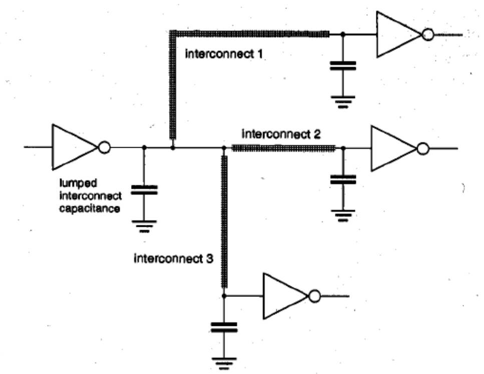


Figure 17: An inverter driving three other inverters over interconnection lines.

In general, if the time of flight across the interconnection line (as determined by the speed of light) is much shorter than the signal rise/fall times, then the wire can be modeled as a capacitive load, or as a lumped or distributed RC network. If the interconnection lines are sufficiently long and the rise times of the signal waveforms are comparable to the time of flight across the line, then the inductance also becomes important, and the interconnection lines must be modeled as transmission lines. The following is a simple rule of thumb which can be used to determine when to use transmission-line models.

$\gamma_{rise}(\gamma_{fall})<2.5\text{x}(L/V)$ => Transmission line modeling

$2.5\text{x}(L/V) < \gamma_{rise}(\gamma_{fall}) < 5\text{x}(L/V)$ Either transmission line or lumped modeling

$\gamma_{rise}(\gamma_{fall})> 5\text{x}(L/V)$

Here, L is the interconnect line length, and V is the propagation speed. Note that transmission line analysis always gives the correct result irrespective of the rise/fall time and the interconnect length; yet the same result can be obtained with the same accuracy using lumped approximation when rise/fall times are sufficiently large.

The transmission-line effects have not been a serious concern in CMOS VLSI chips until recently, since the gate delays due to capacitive load components dominated the line delay in most cases. But as the fabrication technologies move to finer submicron design rules, the intrinsic gate delays tend to decrease significantly. By contrast, the overall chip size and the worst-case line length on a chip tend to increase mainly due to increasing chip complexity, thus, the importance of interconnect delay increases in submicron technologies. In addition, as the widths of metal lines shrink, the transmission line effects and signal coupling between neighboring lines become even more pronounced.

This fact is illustrated in Figure 18, where typical intrinsic gate delay and interconnect delay are plotted qualitatively, for different technologies. It can be seen that for submicron technologies, the interconnect delay starts to dominate the gate delay. In order to deal with the implications and to optimize a system for speed, chip designers must have reliable and efficient means for (i) estimating the interconnect parasitics in a large chip, and (ii) simulating the transient effects.

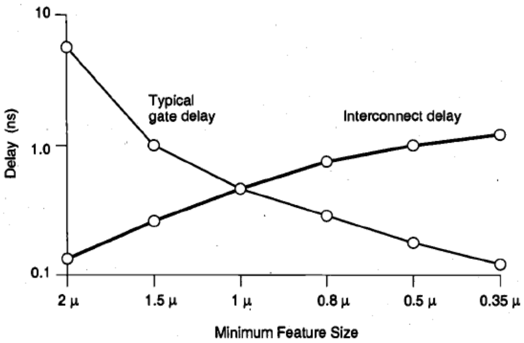


Figure 18: Interconnect delay dominates gate delay in submicron CMOS technologies.

Interconnect Capacitance Estimation:

In a large-scale integrated circuit, the parasitic interconnect capacitances are among the most difficult parameters to estimate accurately. Each interconnection line (wire) is a three-dimensional structure in metal and/or polysilicon, with significant variations of shape, thickness, and vertical distance from the ground plane (substrate). Also, each interconnect line is typically surrounded by a number of other lines, either on the same level or on different levels. Figure 19 shows a simplified view of six interconnections on three different levels, running in close proximity of each other. The accurate estimation of the parasitic capacitances of these wires with respect to the ground plane, as-well as with respect to each other, is a complicated task.

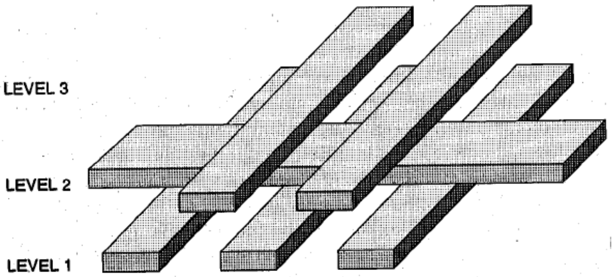


Figure 19: An example of six interconnect lines running on three different levels.

First, consider the section of a single interconnect which is shown in Figure 20. It is assumed that this wire segment has a length of *(1)* in the current direction, a width of (w) and a thickness of (t). Moreover, we assume that the interconnect segment runs parallel to the chip surface and is separated from the ground plane by a dielectric (oxide) layer of height *(h).* Now, the correct estimation of the parasitic capacitance with respect to ground is an important issue. Using the basic geometry given in Figure 20, one can calculate the parallel-plate capacitance C_{pp} of the interconnect segment. However, in interconnect lines where the wire thickness (t) is comparable in magnitude to the ground-plane distance *(h), fringing electric fields* significantly increase the total parasitic capacitance (figure 21).

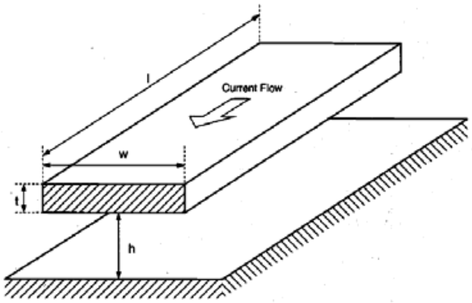


Figure 20: Interconnect segment running parallel to the surface, which is used for parasitic resistance and capacitance estimations.

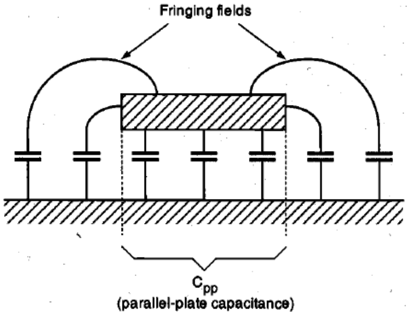
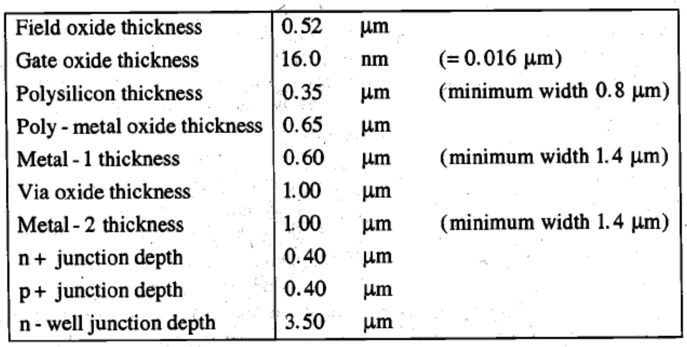


Figure 21: Influence of fringing electric fields upon the parasitic wire capacitance.

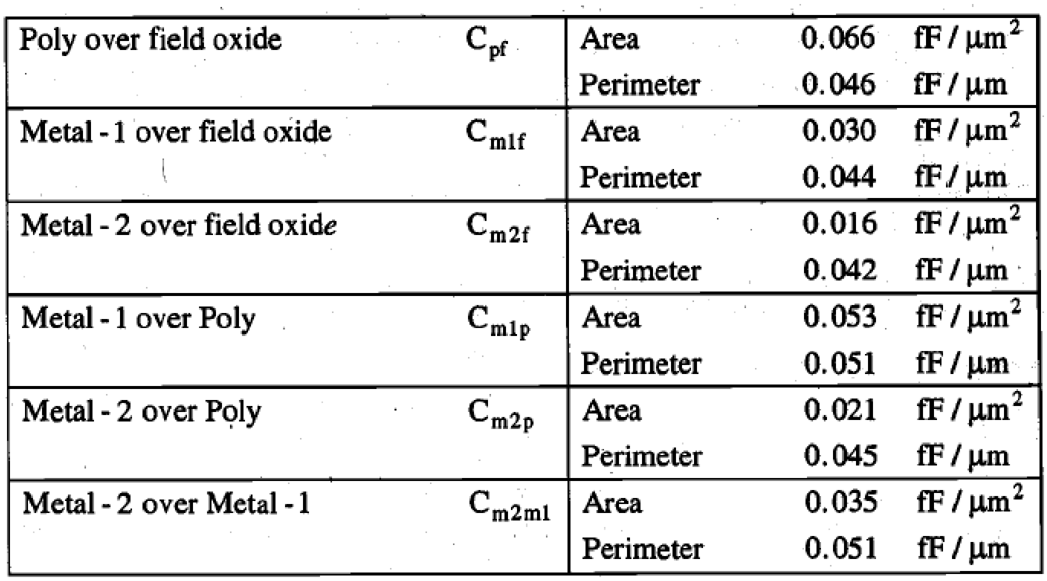


Field oxide thickness	0.52	µm	
Gate oxide thickness	16.0	nm	(= 0.016 µm)
Polysilicon thickness	0.35	µm	(minimum width 0.8 µm)
Poly - metal oxide thickness	0.65	µm	
Metal - 1 thickness	0.60	µm	(minimum width 1.4 µm)
Via oxide thickness	1.00	µm	
Metal - 2 thickness	1.00	µm	(minimum width 1.4 µm)
n + junction depth	0.40	µm	
p + junction depth	0.40	µm	
n - well junction depth	3.50	µm	

Table 2: Thickness values of different layers in a typical 0.8 micron CMOS process.

For the estimation of interconnect capacitances in a complicated three-dimensional structure, the exact geometry must be taken into account for every portion of the wire. Yet this requires an excessive amount of computation in a large circuit, even if simple formulas are applied for the calculation of capacitances. Usually, chip manufacturers supply the area capacitance (parallel-plate capacitance) and the perimeter capacitance (fringing-field capacitance) figures for each layer, which are backed up by measurement of capacitance test structures. These figures can be used to extract the parasitic capacitance from the mask layout.

It is often prudent to include test structures on chip that enable the designer to independently calibrate a process to a set of design tools. In some cases where the entire chip performance is influenced by the parasitic capacitance or coupling of a specific line, accurate 3-D simulation of interconnect parasitic effects is the only reliable solution.



Poly over field oxide	C_{pf}	Area	0.066	$fF/\mu m^2$
		Perimeter	0.046	$fF/\mu m$
Metal - 1 over field oxide	C_{m1f}	Area	0.030	$fF/\mu m^2$
		Perimeter	0.044	$fF/\mu m$
Metal - 2 over field oxide	C_{m2f}	Area	0.016	$fF/\mu m^2$
		Perimeter	0.042	$fF/\mu m$
Metal - 1 over Poly	C_{m1p}	Area	0.053	$fF/\mu m^2$
		Perimeter	0.051	$fF/\mu m$
Metal - 2 over Poly	C_{m2p}	Area	0.021	$fF/\mu m^2$
		Perimeter	0.045	$fF/\mu m$
Metal - 2 over Metal - 1	C_{m2m1}	Area	0.035	$fF/\mu m^2$
		Perimeter	0.051	$fF/\mu m$

Table 3: Parasitic capacitance values between various layers, for atypical double-metal 0.8 micron CMOS technology.

Interconnect Resistance Estimation:

The parasitic resistance of a metal or polysilicon line can also have a significant influence on the signal propagation delay over that line. The resistance of a line depends on the type of material used (e.g., polysilicon, aluminum, or gold), the dimensions of the line and finally, the number and locations of the contacts on that line. Consider again the interconnection line shown in Figure 20. The total resistance in the indicated current direction can be found as

$$R_{wire}=\rho x(L/W.t)=R_{sheet}(L/W)$$

where (ρ) represents the characteristic resistivity of the interconnect material, and R_{sheet} represents the sheet resistivity of the line, in (Ω/square).

$$R_{sheet}= \rho/t$$

Calculation of Interconnect Delay: Two models are introduced here for calculation of interconnected delay.

- RC Delay Models
- The Elmore Delay

RC Delay Models

As we have already discussed in the previous Section, an interconnect line can be modeled as a lumped RC network if the time of flight across the interconnection line is significantly shorter than the signal rise/fall times. This is usually the case in most on-chip interconnects, thus, we will mainly concentrate on the calculation of delay in RC networks, in the following. The simplest model which can be used to represent the resistive and capacitive parasitics of the interconnect line consists of one lumped resistance and one lumped capacitance (Figure 22). Assuming that the capacitance is discharged initially, and assuming that the input signal is a rising step pulse at time t = 0, the output voltage waveform of this simple RC circuit is found as

$$V_{out}(t)=V_{DD}\left(1-e^{-\frac{t}{RC}}\right)$$

And the propagation delay for the simple lumped RC network is found as $\tau = 0.69\ RC$

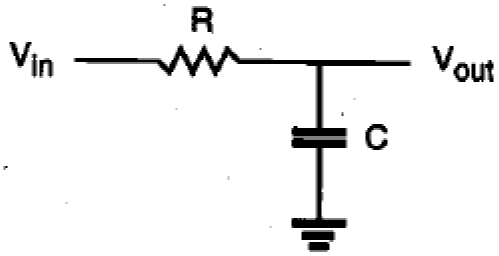


Figure 22: Simple lumped RC model of an interconnect line, where R and C represent the total line resistance and capacitance, respectively

The Elmore Delay:

Consider a general RC tree network, as shown in Figure 23. Note that (i) there are no resistor loops in this circuit, (ii) all of the capacitors in an RC tree are connected between a node and the ground, and (iii) there is one input node in the circuit. Also notice that there is a unique resistive path, from the input node to any other node in the circuit. Inspecting the general topology of this RC tree network, we can make the following path definitions:

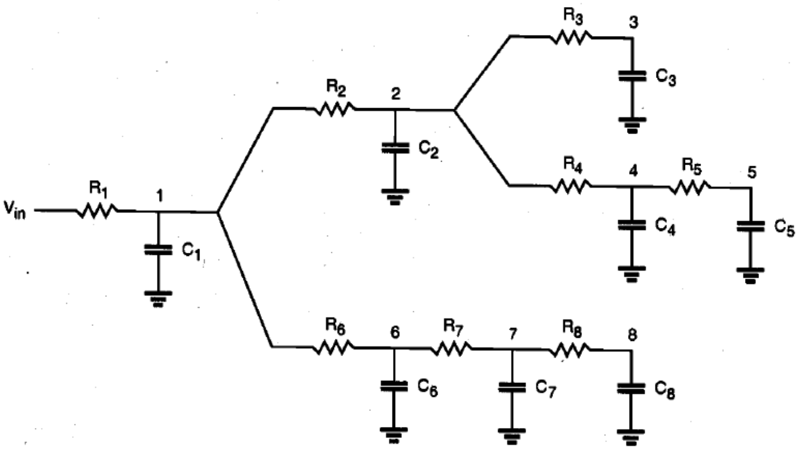


Figure 23. A general RC tree network consisting of several branches.

Assuming that the input signal is a step pulse at time t = 0, the Elmore delay at node i of this RC tree is given by the following expression.

$$\tau_{Di} = \sum_{j=1}^{N} C_j \sum_{\substack{\text{for all} \\ k \in P_{ij}}} R_k$$

Calculation of the Elmore delay is equivalent to deriving the first-order time constant (first moment of the impulse response) of this circuit. Note that although this delay is still an *approximation* for the actual signal propagation delay from the input node to node i, it provides a fairly simple and accurate means of predicting the behavior of the RC line. The procedure to calculate the delay at any node in the circuit is very straightforward. For example, the Elmore delay at node 7 can be found according to the above equation.

$$\tau_{D7} = R_1 C_1 + R_1 C_2 + R_1 C_3 + R_1 C_4 + R_1 C_5 + (R_1 + R_6) C_6 \\ + (R_1 + R_6 + R_7) C_7 + (R_1 + R_6 + R_7) C_8$$

as in Elmore Delay model is τ = 0.50 RC

Module 3

Combinational Logic Circuits

Combinational logic circuits, or gates, which perform Boolean operations on multiple input variables and determine the outputs as Boolean functions of the inputs, are the basic building blocks of all digital systems.In this chapter, we will examine the static and dynamic characteristics of various combinational MOS logic circuits.

The first major class of combinational logic circuits to be presented in this chapter is the nMOS depletion-load gates. Our purpose for including nMOS depletion-load circuits here is mainly pedagogical, to emphasize the load concept, which is still being widely used in many areas in digital circuit design. We will examine simple circuit configurations such as two-input NAND and NOR gates and then expand our analysis to more general cases of multiple-input circuit structures. Next, the CMOS logic circuits will be presented in a similar fashion. We will stress the similarities and differences between the nMOS depletion-load logic and CMOS logic circuits and point out the advantages of CMOS gates with examples. The design of complex logic gates, which allows the realization of complex Boolean functions of multiple variables, will be examined in detail. Finally, we will devote the last section to CMOS transmission gates and to transmission gate (TG) logic circuits.

In its most general form, a combinational logic circuit, or gate, performing a Boolean function can be represented as a multiple-input single-output system.

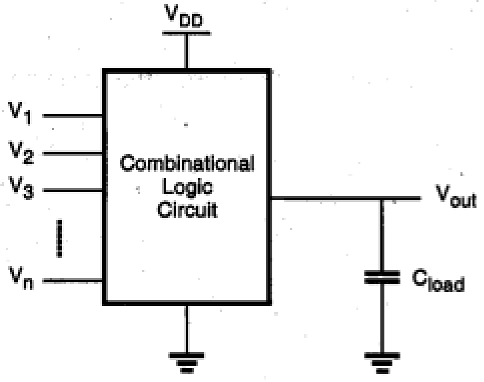


Figure 24: Generic combinational logic circuit (gate).

All input variables are represented by node voltages, referenced to the ground potential.Using positive logic convention, the Boolean(or logic) value of "1" can be represented by a high voltage of VDD, and the Boolean(or logic) value of "0" can be represented by a low voltage of 0. The output node is loaded with capacitance CL, which represents the combined parasitic device capacitances in the circuit and the interconnect capacitance components seen by the output node. This output load capacitance certainly plays a very significant role in the dynamic operation of the logic gate.

As in the simple inverter case, the voltage transfer characteristic (VTC) of a combinational logic gate provides valuable information on the DC operating performance of the circuit. Critical voltage points such as VOL or Vth are considered to be important design parameters for combinational logic circuits. Other design parameters and concerns include the dynamic (transient) response characteristics of the circuit, the silicon area occupied by the circuit, and the amount of static and dynamic power dissipation.

Static CMOS Logic Circuits:

Complementary CMOS

Introduction- Complementary Static CMOS design offers low noise sensitivity, speed and low power consumption. Static CMOS design means that at any time, the output of the gate is directly connected to VSS or VDD. In comparison, Dynamic CMOS gates depend upon the device capacitance to temporarily hold charge at the output.

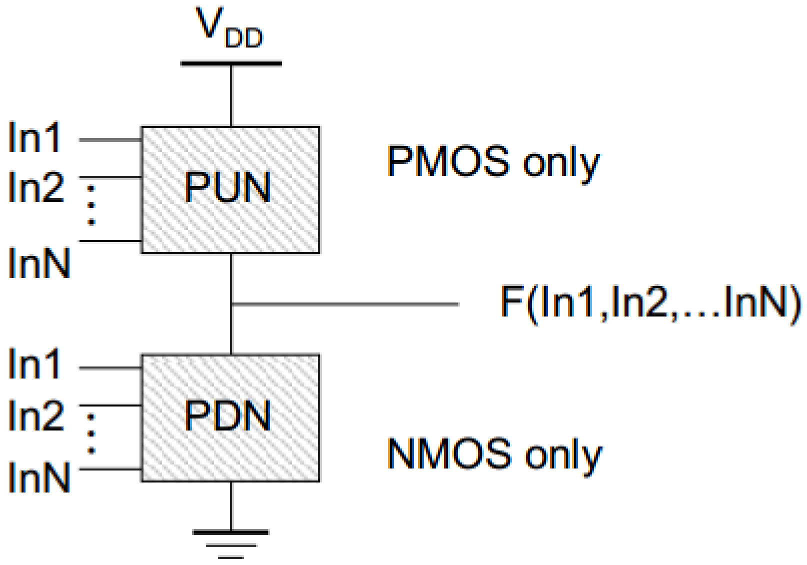


Figure 25: Pull up and Pull down networks

Properties of Complementary Static CMOS

1) Contains a pull up network (PUP) and pull down network (PDN)
2) PUP networks consist of PMOS transistors
3) PDN networks consist of NMOS transistors
4) Each network is the dual of the other network
5) The output of the complementary gate is inverted.

PDN Design - Any function is first designed by the Pull Down network. The Pull Up network can be designed by simply taking the dual of the Pull Down network, once it is found.An NMOS gate will pass current between source and drain when 5 volts is presented at the gate.

1) All functions are composed of either and'ed or or'ed sub-functions.
2) The And function is composed of NMOS transistors in series.
3) The Or function is composed of NMOS transistors in parallel.

CMOS Logic Circuits

CMOS NOR2 (Two-Input NOR) Gate

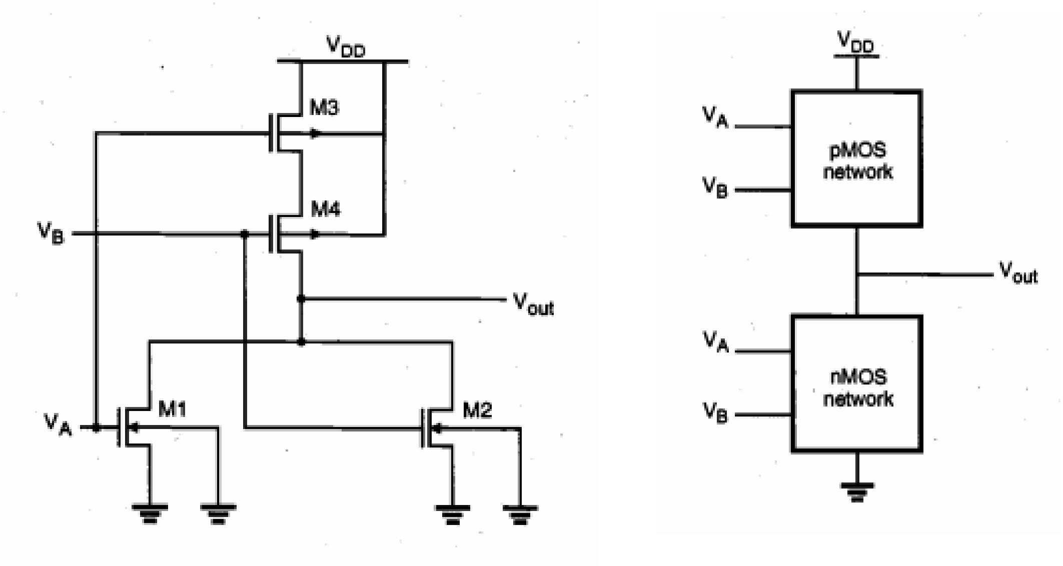


Figure 26: A CMOS NOR2 gate and its complementary operation: Either the nMOS network is on and the pMOS network is off, or the pMOS network is on and the nMOS network is off.

The circuit consists of a parallel-connected n-net and a series-connected complementary pnet. The input voltages VA and VB are applied to the gates of one nMOS and one pMOS transistor.

The complementary nature of the operation can be summarized as follows: When either one or both inputs are high, i.e., when the n-net creates a conducting path between the output node and the ground, the p-net is cut-off. On the other hand, if both input voltages are low, i.e., the n-net is cut-off, then the p-net creates a conducting path between the output node and the supply voltage VDD. Thus, the dual or complementary circuit structure allows that, for any given input combination, the output is connected either to VDD or to ground via a low-resistance path. A DC current path between the VDD and ground is not established for any of the input combinations.

The output voltage of the CMOS NOR2 gate will attain a logic-low voltage of VOL= 0 and a logic-high voltage of VOH = VDD. For circuit design purposes, the switching threshold voltage Vth of the CMOS gate emerges as an important design criterion. We start our analysis of the switching threshold by assuming that both input voltages switch simultaneously, i.e.VA =VB. Furthermore, it is assumed that the device sizes in each block are identical, (W/L)n,A = (W/L)n,B and (W/L)p,A = (W/L)p,B and the substrate-bias effect for the pMOS transistors is neglected for simplicity.

By definition, the output voltage is equal to the input voltage at the switching threshold.

$$V_A = V_B = V_{out} = V_{th}$$

It is obvious that the two parallel nMOS transistors are saturated at this point, because VGS=VDS. The combined drain current of the two nMOS transistors is

$$I_D = k_n \left(V_{th} - V_{T,n}\right)^2$$

Thus, we obtain the first equation for the switching threshold Vth

$$V_{th} = V_{T,n} + \sqrt{\frac{I_D}{k_n}}$$

Examination of the p-net shows that the pMOS transistor M3 operates in the linear region, while the other pMOS transistor, M4, is in saturation for Vth=Vout.Thus,

$$I_{D3} = \frac{k_p}{2}\left[2\left(V_{DD} - V_{th} - \left|V_{T,p}\right|\right)V_{SD3} - V_{SD3}^2\right]$$

$$I_{D4} = \frac{k_p}{2}\left(V_{DD} - V_{th} - \left|V_{T,p}\right| - V_{SD3}\right)^2$$

The drain currents of bothpMOS transistors are identical, i.e., ID3=ID4=ID. Thus,

$$V_{DD} - V_{th} - \left|V_{T,p}\right| = 2\sqrt{\frac{I_D}{k_p}}$$

This yields the second equation of the switching threshold voltage Vth

$$V_{th}(\text{NOR2}) = \frac{V_{T,n} + \frac{1}{2}\sqrt{\frac{k_p}{k_n}}\left(V_{DD} - \left|V_{T,p}\right|\right)}{1 + \frac{1}{2}\sqrt{\frac{k_p}{k_n}}}$$

If kn = kp and V T,n = |VT,p|, the switching threshold of the CMOS inverter is equal to VDD/2. Using the same parameters, the switching threshold of the NOR2 gate is

$$V_{th}(\mathrm{NOR2}) = \frac{V_{DD} + V_{T,n}}{3}$$

which is not equal to VDD/2. For example, when VDD = 5 V and VT,n =|VT,p| = 1 V, the switching threshold voltages of the NOR2 gate and the inverter are

$$V_{th}(\mathrm{NOR2}) = 2\ \mathrm{V}$$

$$V_{th}(\mathrm{INR}) = 2.5\ \mathrm{V}$$

The switching threshold voltage of the NOR2 gate can also be obtained by using the equivalent-inverter approach. When both inputs are identical, the parallel-connected nMOS transistors can be represented by a single nMOS transistor with 2kb. Similarly, the series-connected pMOS transistors are represented by a single pMOS transistor with Kp/2.

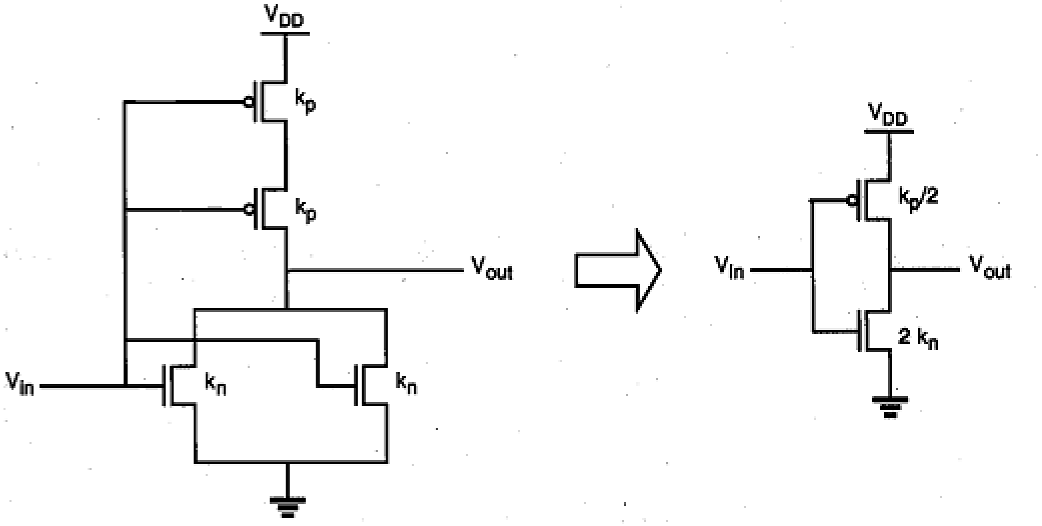


Figure 27: A CMOS NOR2 gate and its inverter equivalent.

CMOS NAND2 (Two-Input NAND) Gate

The operating principle of this circuit is the exact dual of the CMOS NOR2 operation examined earlier. The n-net consisting of two series-connected nMOS transistors creates a conducting path between the output node and the ground only if both input voltages are logic-high, i.e., are equal to VOH In this case, both of the parallel-connected pMOS transistors in the p-net will be off. For all other input combinations, either one or both of the pMOS transistors will be turned on, while the n-net is cut-off, thus creating a current path between the output node and the power supply voltage.

Using an analysis similar to the one developed for the NOR2 gate, we can easily calculate the switching threshold for the CMOS NAND2 gate. Again, we will assume that the device sizes in each block are identical, with (W/L)n,A = (W/L)n,B and (W/L)p,A = (W/L)p,B. The switching threshold for this gate is then found as

$$V_{th}(\mathrm{NAND2}) = \frac{V_{T,n} + 2\sqrt{\frac{k_p}{k_n}}\left(V_{DD} - \left|V_{T,p}\right|\right)}{1 + 2\sqrt{\frac{k_p}{k_n}}}$$

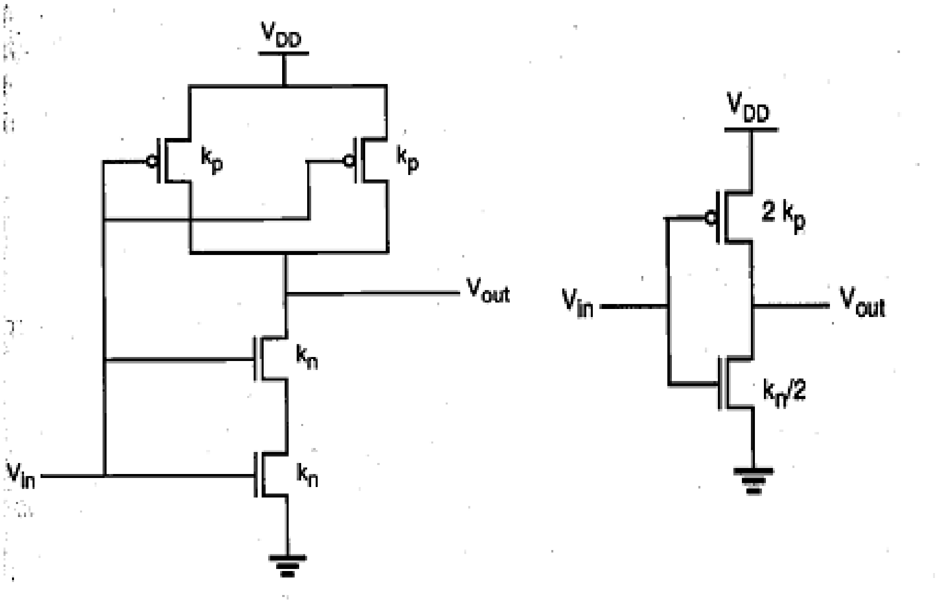


Figure 28: A CMOS NAND2 gate and its inverter equivalent.

Ratioed Logic

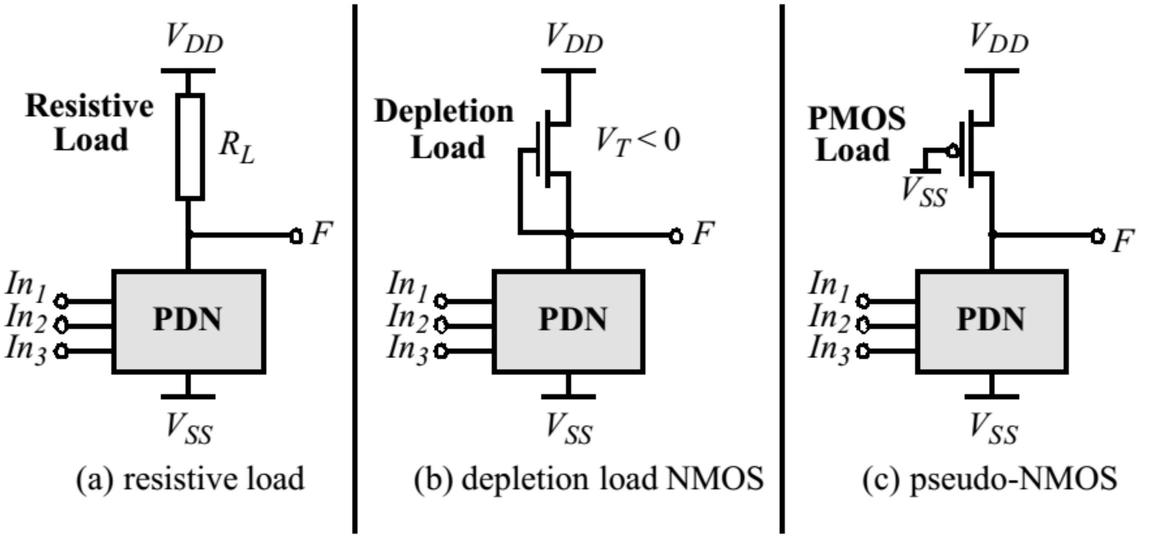
It is one method to reduce the circuit complexity of static CMOS. Here, the logic function is built in the PDN and used in combination with a simple load device

Comparison of Logic Families

Complementary CMOS	Ratioed Logic
Full Swing	Reduced Swing (smaller noise margin)
No Static Power	Static Power

tpHL= tpLH
if sized correctly
	tpLH>tpHL

Large PMOS parasitic capactiances	much smaller parasitic capactiances



(a) resistive load (b) depletion load NMOS (c) pseudo-NMOS

DC characteristics:

ƒ VOH= VDD

ƒ VOL depends on PMOS to NMOS ratio

Let's assume the load can be represented as linearized resistors. When the PDN is on, the output voltage is determined by:

$$V_{OL} = \frac{R_{PDN}}{R_L + R_{PDN}} V_{DD}$$

This logic style is called ratioed because care must be taken in scaling the impedances properly. Note that full complementary CMOS is ratioless, since the output signals do not depend on the size of the transistors. In order to keep the noise margins high, RL >> RPDN

However, RL must be able to provide as much current as possible to minimize delay.

$$t_{pLH} = 0.69R_LC_L$$

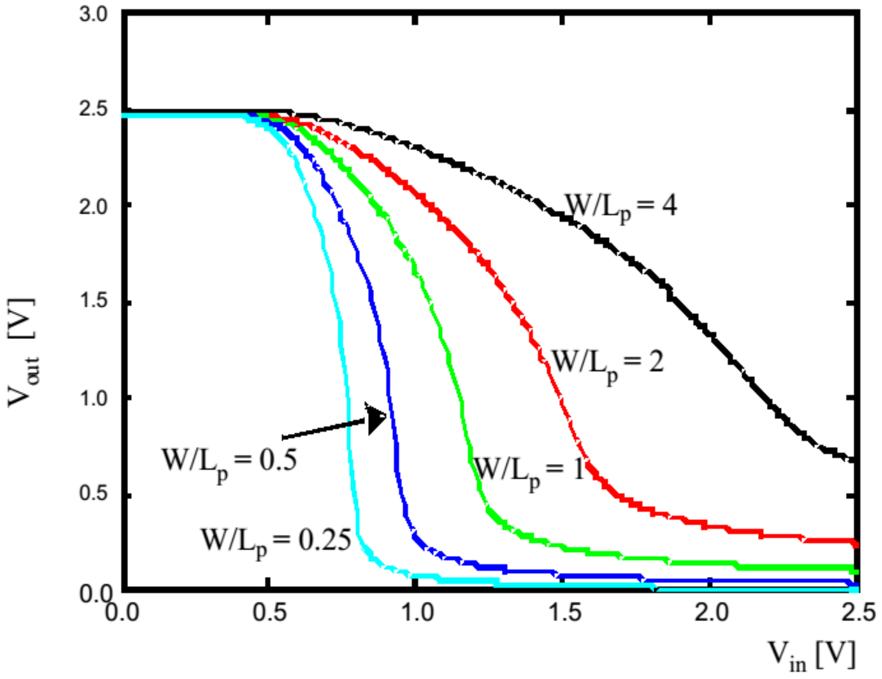
$$t_{pHL} = 0.69(R_L // R_{PDN})C_L$$

These are conflicting requirements:

RL large: Noise margins.

RL small: performance and power dissipation.

Pseudo-NMOS VTC



Pass-Transistor Logic

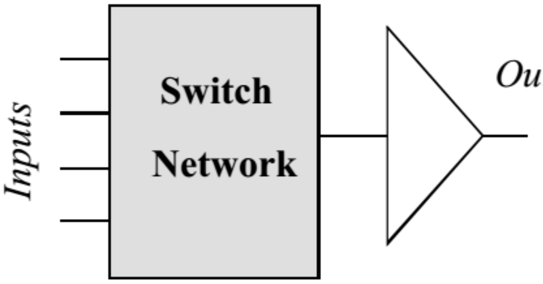
Introduction

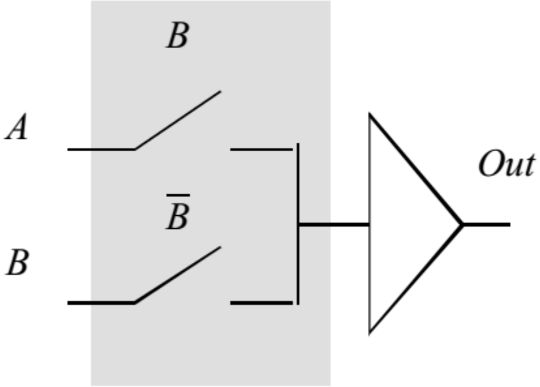
Realizing boolean functions using transistors as a switch, this is typically known as pass transistor logic circuits.

Characteristics of an ideal switch

- It has zero on resistance (Vin=Vout)
- Infinite(very very high) OFF resistance (Vout=0 v), previous value retained in case of capacitive load.

- Relay is an ideal switch. In relay logic, presence of high voltage level is considered to be “1”. Absence of high voltage level is considered to be “0”.
- Pass transitor logic is different from relay logic.



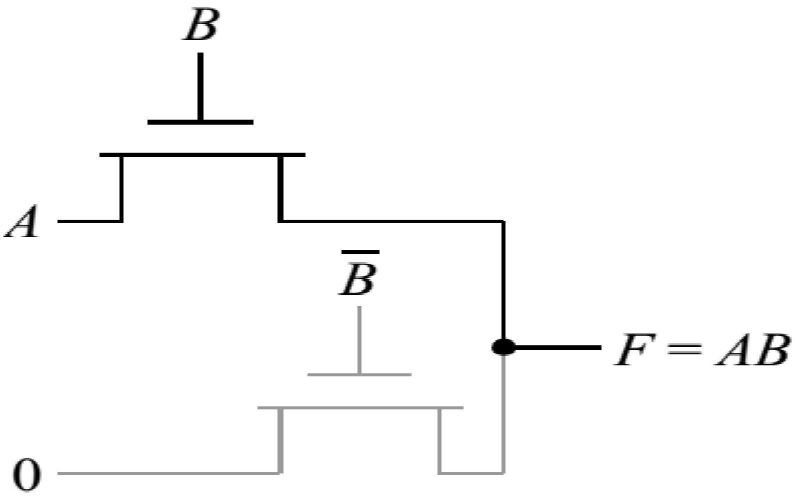


- N inputs, N transistors
- No static power consumption (ideally)

Pass transistor logic Design rules

- One must not drive the gate of a pass transistor by the output of another pass transistor.
- It is essential to provide both charging and discharging path for the load capacitance.
- When a signal is steered through several stages of pass transistors, the delay can be considerable.

Example: AND Gate



When B is “1”, top device turns on and copies the input A to output F. When B is low, bottom device turns on and passes a “0”.

The presence of the switch driven by B is essential to ensure that the gate is static – a low-impedance path must exist to supply rails.

Advantage:

Fewer devices to implement some functions.

Example: AND2 requires 4 devices (including inverter to invert B) vs. 6 for complementary CMOS (lower total capacitance).

NMOS is effective at passing a 0, but poor at pulling a node to Vdd. When the pass transistor a node high, the output only charges up to Vdd- Vtn. This becomes worse due to the body effect. The node will be charged up to Vdd– Vtn(Vs)

Complementary Pass Transistor Logic

The complexity of full-CMOS pass-gate logic circuits can be reduced dramatically by adopting another circuit concept, called Complementary Pass-transistor Logic (CPL). The main idea behind CPL is to use a purely nMOS pass-transistor network for the logic operations, instead of a CMOS TG network. All inputs are applied in complementary form, i.e., every input signal and its inverse must be provided; the circuit also produces complementary outputs, to be used by subsequent CPL stages. Thus, the CPL circuit essentially consists of complementary inputs, an nMOS pass transistor logic network to generate complementary outputs, and CMOS output inverters to restore the output signals.

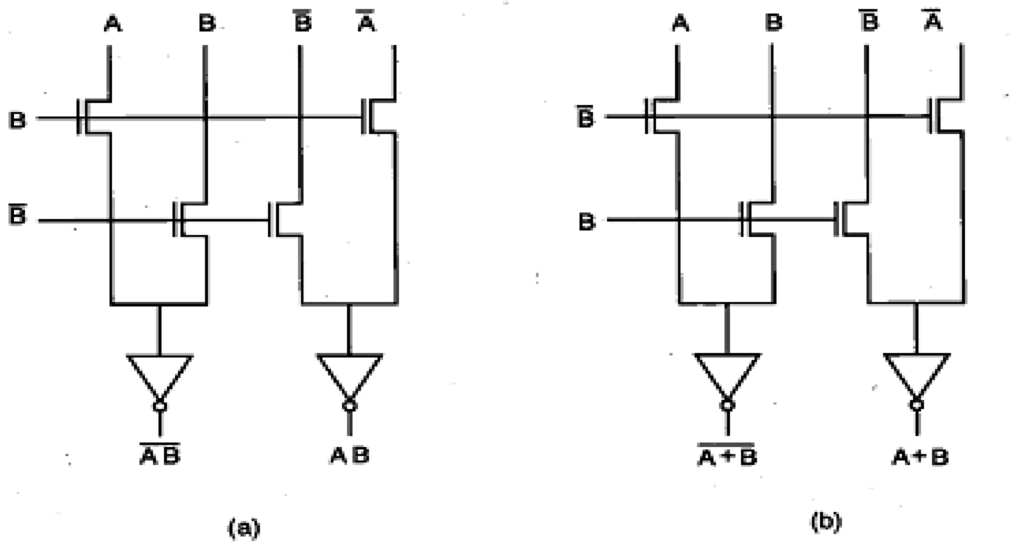


Figure 29: Circuit diagram of (a) CPL NAND2 gate and (b) CPL NOR2 gate

The elimination of pMOStransistors from the pass-gate network significantly reduces the parasitic capacitances associated with each node in the circuit, thus, the operation speed is typically higher compared to a full-CMOS counterpart. But the improvement in transient characteristics comes at a price of increased process complexity. In CPL circuits, the threshold voltages of the nMOStransistors in the pass-gate network must be reduced to about 0 V through threshold-adjustment implants, in order to eliminate the threshold-voltage drop. This, on the other hand, reduces the overall noise immunity and makes the transistors more susceptible to subthreshold conduction in the off-mode. Also note that the CPL design style is highly modular, a wide range of functions can be realized by using the same basic pass-transistor structures.

Transmission Gate Logic

The CMOS transmission gate consists of one nMOS and one pMOS transistor, connected in parallel. The gate voltages applied to these two transistors are also set to be complementary signals. As such, the CMOS TG operates as a bidirectional switch between the nodes A and B which is controlled by signal C.

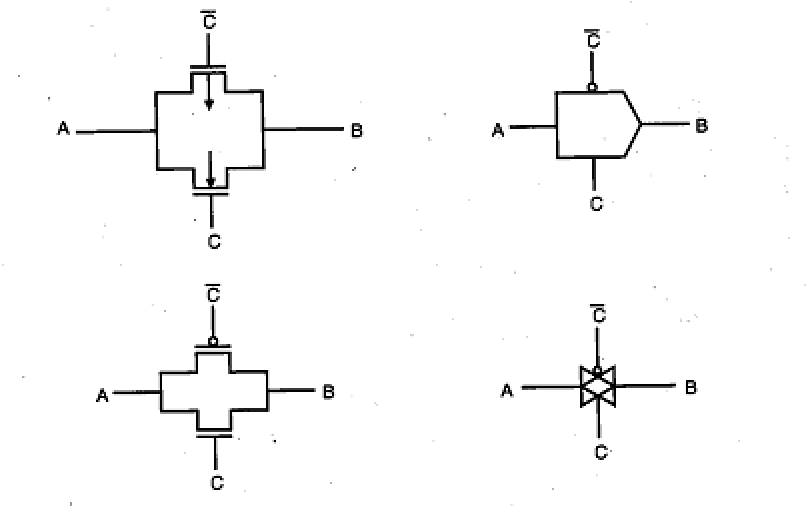
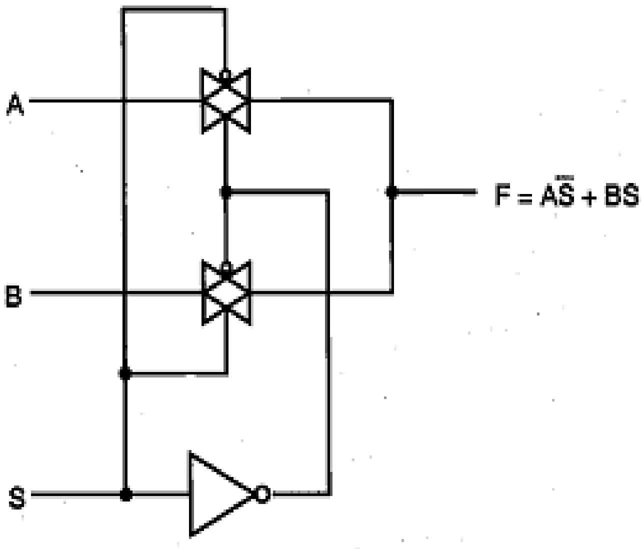


Figure 30: Four different representation of CMOS transmission gate(TG)

If the control signal C is logic-high, i.e., equal to VDD, then both transistors are turned on and provide a low-resistance current path between the nodes A and B. If, on the other hand, the control signal C is low, then both transistors will be off, and the path between the nodes A and B will be an open circuit. This condition is also called the high-impedance state.

Charecteristics

- NMOS passes a strong “0”
- PMOS passes a strong “1”
- Transmission gates enable rail-to-rail swing
- These gates are particularly efficient in implementing MUXs



Two-input multiplexor circuit implemented using two CMOS TGs.

DCVS (Differential Cascode Voltage Switch Logic) Logic

It requires mainly N-channel MOSFET transistors to implement the logic using true and complementary input signals, and also needs two P-channel transistors at the top to pull one of the outputs high.

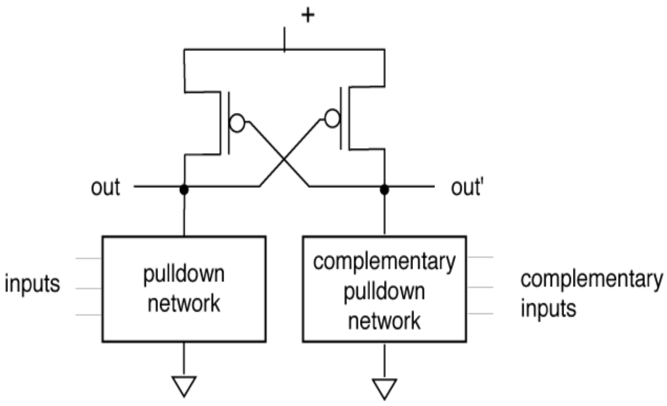
Static logic—consumes no dynamic or static power.

Uses latch to compute output quickly.

Requires true/complement inputs, produces true/complement outputs.

The cascode (sometimes verbified to cascoding) is a universal technique for improving analog circuit performance, applicable to both vacuum tubes and transistors. The word was first used in an article by F.V. Hunt and R.W. Hickman in 1939, in a discussion for application in low-voltage stabilizers. They proposed a cascade of two triodes (first one with common cathode, the second one with common grid) as a replacement of a pentode.

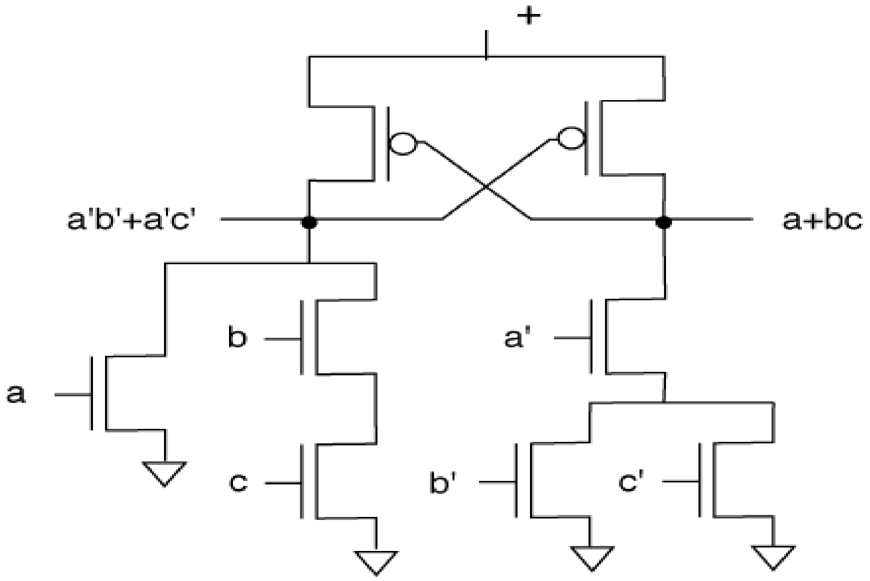
DCVS structure



DCVS operation

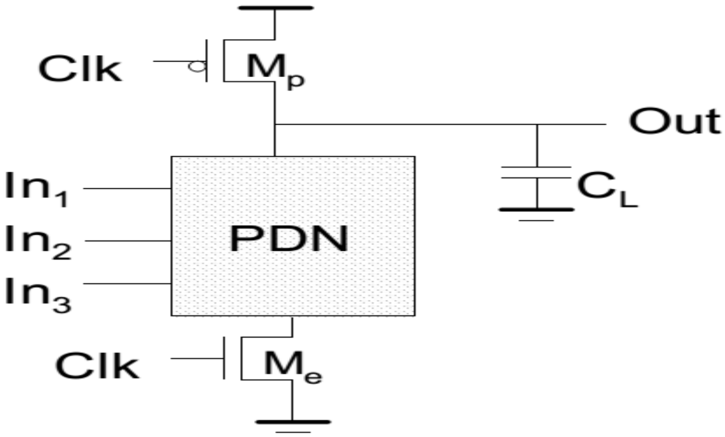
- Exactly one of true/complement pulldown networks will complete a path to the power supply.
- Pulldown network will lower output voltage, turning on other p-type, which also turns off p-type for node which is going down.

DCVS example



Dynamic CMOS Logic Circuits

- Output not permanently connected to Vdd or Vss
- Output value partly relies on storage of signal values on the capacitance of high impedance circuit nodes.
- Input only active when clockis active
- Requires N+2 transistors for N inputs



Two phase operation

Precharge (CLK = 0) – Me is off => no static power

Evaluate (CLK = 1) –Mp is off

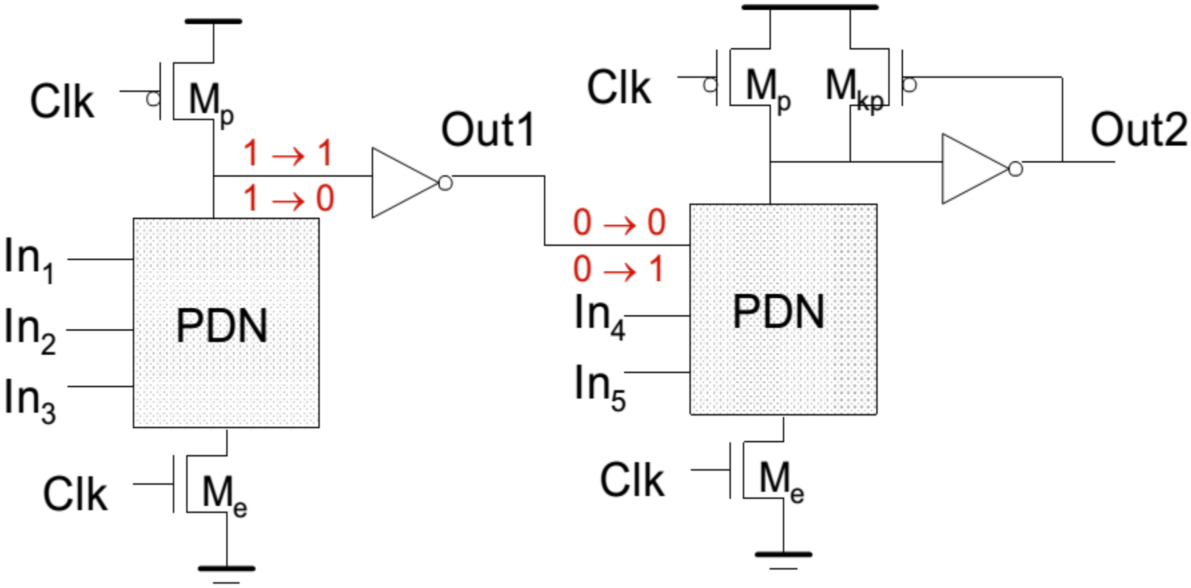
Conditions on Output

- Once the output of a dynamic gate is discharged, it cannot be charged again until the next precharge operation**.**
- Inputs to the gate can make at most one transition during evaluation.
- Output can be in the high impedance state during and after evaluation (PDN off), state is stored on CL (fundamentally different than static gates)

Properties of Dynamic Gates

- Logic function is implemented by the PDN only(number of transistors is N + 2 (versus 2N for static complementary CMOS)
- Full swing outputs (VOL= GND and VOH= VDD)
- Non-ratioed - sizing of the devices does not affect the logic levels-Sizing the PMOS doesn't impact correct functionality. Sizing it up improves the low- to-high transition time (not too critical), but trades- off increase in clock -power dissipation.
- Faster switching speeds
    - reduced load capacitance due to lower input capacitance (Cin)
    - reduced load capacitance due to smaller output loading (Cout)
    - no Isc, so all the current provided by PDN goes into discharging CL
- Overall power dissipation usually higher than static CMOS
    - no static current path ever exists between VDD and GND (including Psc)
    - higher transition probabilities
    - extra load on Clock -> higher dynamic power consumption
- Needs a precharge /evaluate clock

Domino Logic



The inverter:

(a) ensures that all inputs are set to "0" at the end of the precharge phase

(b) has a low impedance output, which increases the noise immunity.

(c) Can be used to drive a keeper device to combat leakage and charge redistribution.

Why Domino?

- Since each dynamic gate has a static inverter, only non -inverting logic can be implemented (there are ways to deal with this)
- Very high speed
    - Input capacitance reduced

Sequential Logic Circuits

Introduction:

Combinational logic circuits that were described earlier have the property that the output of a logic block is only a function of the current input values, assuming that enough time has elapsed for the logic gates to settle. Yet virtually all useful systems require storage of state information, leading to another class of circuits called sequential logic circuits. In these circuits, the output not only depends upon the current values of the inputs, but also upon preceding input values. In other words, a sequential circuit remembers some of the past history of the system—it has memory.

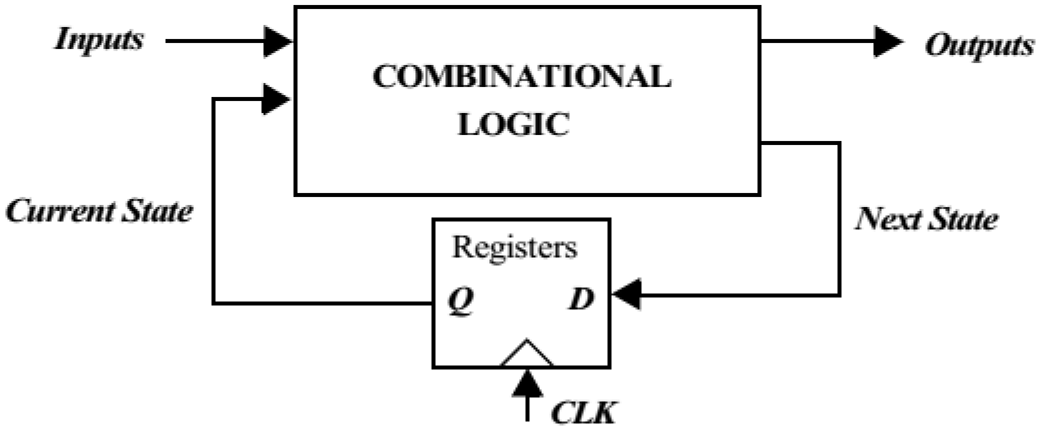


Figure 31:Block diagram of a finite state machine using positive edge-triggered registers

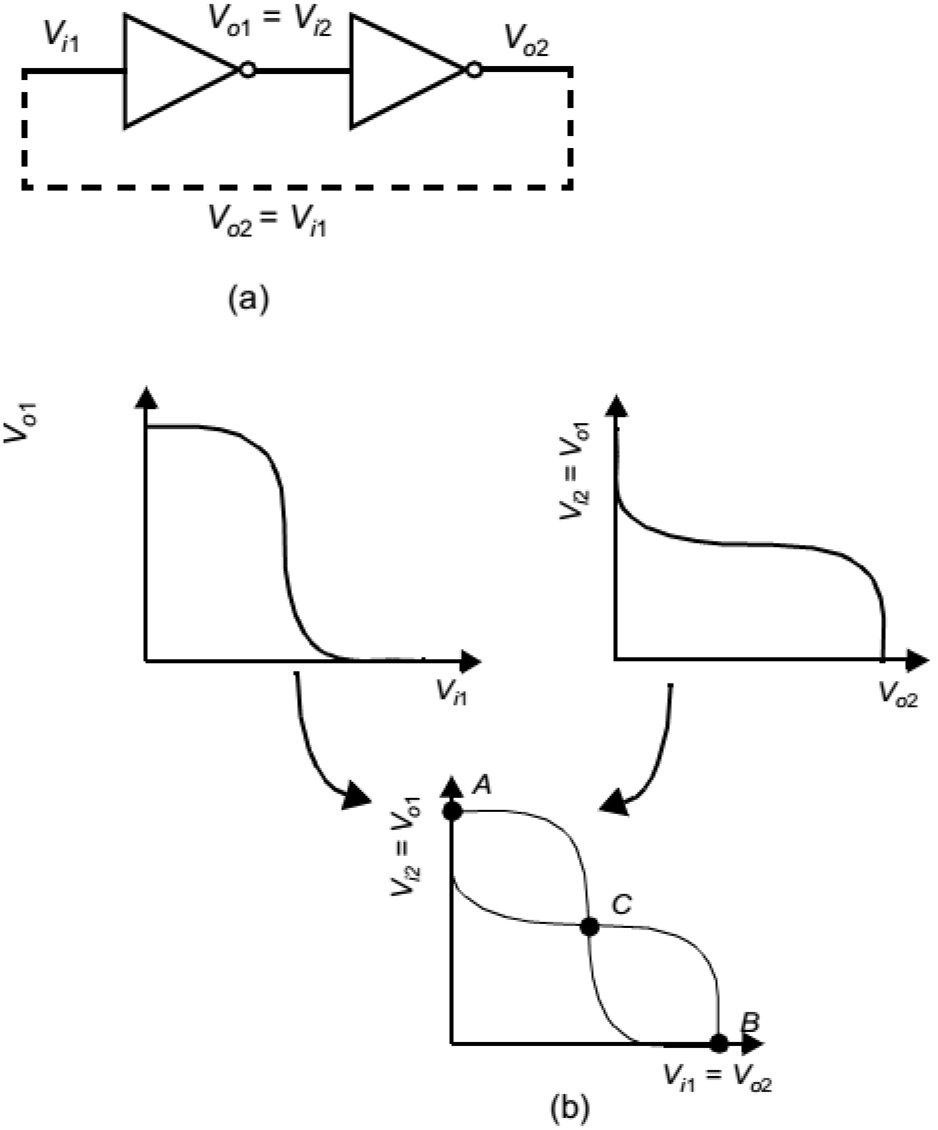
Above figure shows a block diagram of a generic finite state machine(FSM) that consists of combinational logic and registers that hold the system state. The system depicted here belongs to the class of synchronous sequential systems, in which all registers are under control of a single global clock. The outputs of the FSM are a function of the current Input sand the Current State. The Next State is determined based on the Current State and the current Input sand is fed to the inputs of registers. On the rising edge of the clock, the Next State bits are copied to the outputs of the registers (after some propagation delay), and a new cycle begins. The register then ignores changes in the input signals until the next rising edge. In general, registers can be positive edge-triggered(where the input data is copied on the positive edge) or negative edge triggered (where the input data is copied on the negative edge of the clock, as is indicated by a small circle at the clock input).

This chapter discusses the CMOS implementation of the most important sequential building blocks. A variety of choices in sequential primitives and clocking methodologies exist; making the correct selection is getting increasingly important in modern digital circuits, and can have a great impact on performance, power, and/or design complexity. Before embarking on a detailed discussion on the various design options, a revision of the design metrics, and a classification of the sequential elements is necessary.

Static Latches and Registers

The Bistability Principle

Static memories use positive feedback to create abistable circuit —a circuit having two stable states that represent 0 and 1. The basic idea is two inverters connected in cascade along with a voltage-transfer characteristic typical of such a circuit.



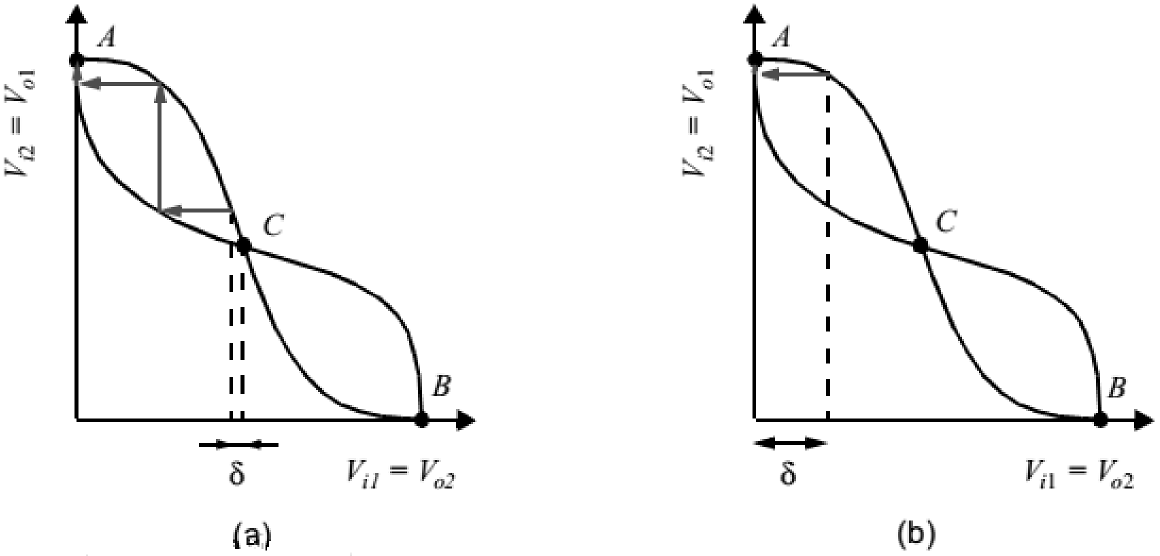
(a)

(b)

Also plotted are the VTCs of the first inverter, that is Vo1 versusVi1and the second inverter (Vo2 versusVo1). The latter plot is rotated to accentuate that Vi2 =Vo1 Assume now that the output of the second inverterVo2 is connected to the input of the first Vi1. The resulting circuit has only three possible operation points (A, B and C), as demonstrated on the combined VTC. The following important conjecture is easily proven to be valid:

Under the condition that the gain of the inverter in the transient region is larger than 1, only A and B are stable operation points, and C is a metastable operation point.

Suppose that the cross-coupled inverter pair is biased at point C. A small deviation from this bias point, possibly caused by noise, is amplified and regenerated around the circuit loop. This is a consequence of the gain around the loop being larger than 1.A small deviation δ is applied toVi1(biased in C). This deviation is amplified by the gain of the inverter. The enlarged divergence is applied to the second inverter and amplified once more. The bias point moves away from C until one of the operation points A or B is reached. In conclusion, C is an unstable operation point. Every deviation (even the smallest one) causes the operation point to run away from its original bias. The chance is indeed very small that the cross-coupled inverter pair is biased at C and stays there. Operation points with this property are termed metastable.



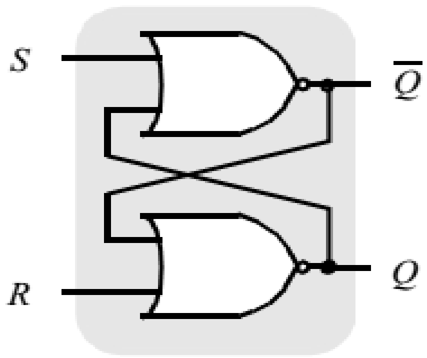
On the other hand, A and B are stable operation points. In these points, the loop gain is much smaller than unity. Even a rather large deviation from the operation point is reduced in size and disappears.Hence the cross-coupling of two inverters results in a bistable circuit, that is, a circuit with two stable states, each corresponding to a logic state. The circuit serves as a memory, storing either a 1 or a 0 (corresponding to positions A and B).

In order to change the stored value, we must be able to bring the circuit from state A to Band vice-versa. Since the precondition for stability is that the loop gain G is smaller than unity, we can achieve this by making A (or B) temporarily unstable by increasing G to a value larger than 1. This is generally done by applying a trigger pulse atVi1 or Vi2. For instance, assume that the system is in position A(Vi1 =0,Vi2 =1).ForcingVi1 to 1 causes both inverters to be on simultaneously for a short time and the loop gain G to be larger than 1. The positive feedback regenerates the effect of the trigger pulse, and the circuit moves to the other state (Bin this case). The width of the trigger pulse need be only a little larger than the total propagation delay around the circuit loop, which is twice the average propagation delay of the inverters.

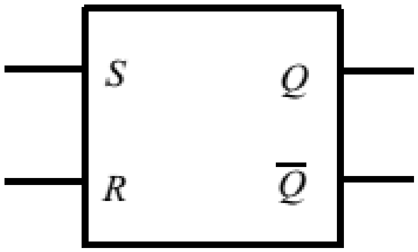
SR Flip-Flops

The cross-coupled inverter pair shown in the previous section provides an approach to store a binary variable in a stable way. However, extra circuitry must be added to enable control of the memory states. The simplest incarnation accomplishing this is the well knowSR —or set-reset— flip-flop.

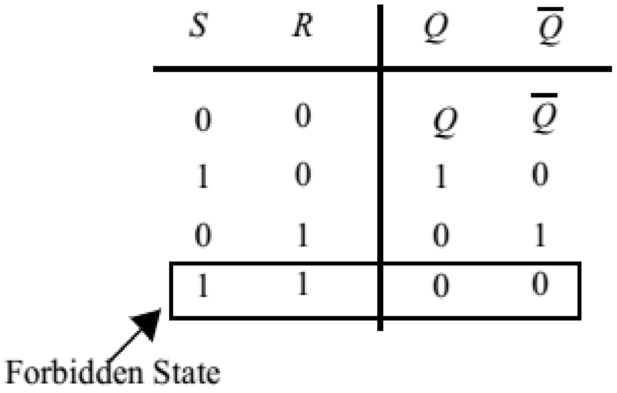
This circuit is similar to the cross-coupled inverter pair with NOR gates replacing the inverters. The second input of the NOR gates is connected to the trigger inputs (S and R), that make it possible to force the outputs Q and Qbarto a given state. These outputs are complimentary (except for theSR= 11 state). When both S andRare 0, the flip-flop is in a quiescent state and both outputs retain their value (a NOR gate with one of its input being 0 looks like an inverter, and the structure looks like a cross coupled inverter). If a positive (or 1) pulse is applied to theSinput, theQoutput is forced into the 1 state (withQbar going to 0). Vice versa, a 1 pulse onRresets the flip-flop and theQoutput goes to 0.



(a) Schematic diagram



(b) Logic symbol



S	R	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
1	0	1	0
0	1	0	1
1	1	0	0

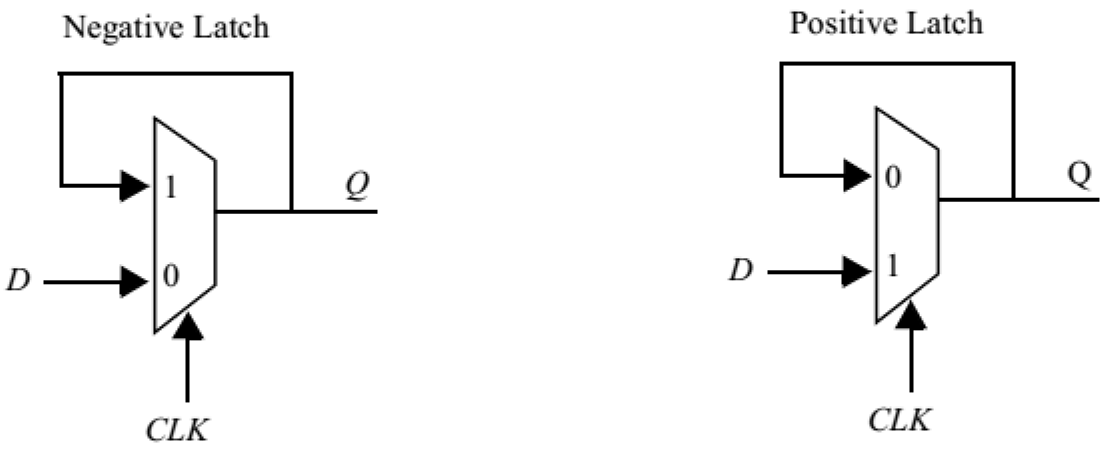
Forbidden State

(c) Characteristic table

These results are summarized in the characteristic table of the flip-flop. The characteristic table is the truth table of the gate and lists the output states as functions of all possible input conditions. When both S and R are high, both Q and Qbar are forced to zero. Since this does not correspond with our constraint that Q and Qbar must be complementary, this input mode is considered to be forbidden. An additional problem with this condition is that when the input triggers return to their zero levels, the resulting state of the latch is unpredictable and depends on whatever input is last to go low.

Multiplexer Based Latches

There are many approaches for constructing latches. One very common technique involves the use of transmission gate multiplexers. Multiplexer based latches can provide similar functionality to the SR latch, but has the important added advantage that the sizing of devices only affects performance and is not critical to the functionality**.**



Negative and positive latches based on multiplexers.

For a negative latch, when the clock signal is low, the input 0 of the multiplexer is selected, and the D input is passed to the output. When the clock signal is high, the input 1 of the multiplexer, which connects to the output of the latch, is selected. The feedback holds the output stable while the clock signal is high. Similarly in the positive latch, the D input is selected when clock is high, and the output is held (using feedback) when clock is low.

A transistor level implementation of a positive latch based on multiplexers is shown below.When CLK is high, the bottom transmission gate is on and the latch is transparent-that is, the D input is copied to the Q output. During this phase, the feedback loop is open since the top transmission gate is off. Unlike the SRFF, the feedback does not have to be overridden to write the memory and hence sizing of transistors is not critical for realizing correct functionality. The number of transistors that the clock touches is important since it has an activity factor of 1. This particular latch implementation is not particularly efficient from this metric as it presents a load of 4 transistors to the CLK signal.

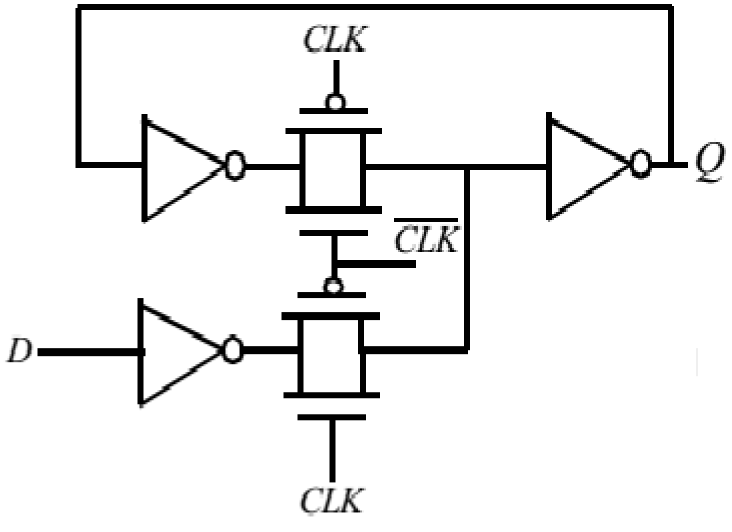
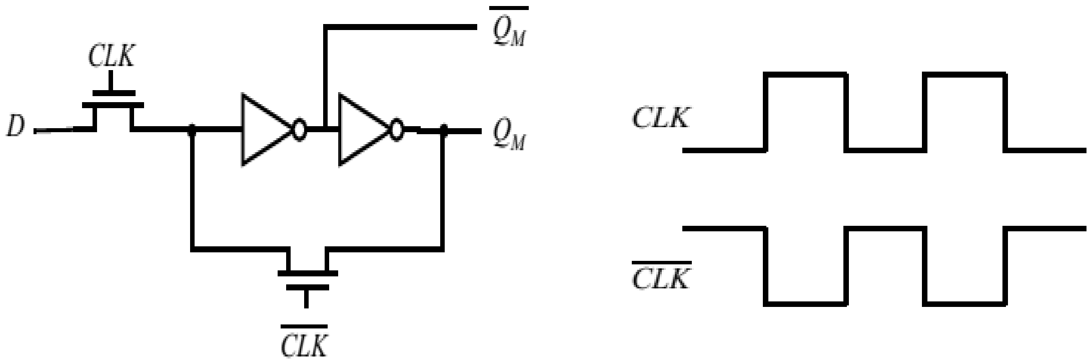


Figure 32: Transistor level implementation ofa positive latch built using transmission gates.

It is possible to reduce the clock load to two transistors by using implement multiplexers using NMOS only pass transistor as shown in Figure 7.13. The advantage of this approach is the reduced clock load of only two NMOS devices. When CLK is high, the latch samples the D input, while a low clock-signal enables the feedback-loop, and puts the latch in the hold mode. While attractive for its simplicity, the use of NMOS only pass transistors results in the passing of a degraded high voltage of VDD-VTn to the input of the first inverter. This impacts both noise margin and the switching performance, especially in the case of low values of VDD and high values of VTn. It also causes static power dissipation in first inverter, as already pointed out in Chapter 6. Since the maximum input-voltage to the inverter equals VDD-VTn, the PMOS device of the inverter is never turned off, resulting is a static current flow.



.

(a) Schematic diagram (b) Non overlapping clocks

Fig: Multiplexer based NMOS latch using NMOS only pass transistors for multiplexers

Master-Slave Based Edge Triggered Register

The most common approach for constructing an edge-triggered register is to use a master-slave configuration. The register consists of cascading a negative latch (master stage) with a positive latch (slave stage). A multiplexer based latch is used in this particular implementation, though any latch can be used to realize the master and slave stages. On the low phase of the clock, the master stage is transparent and the D input is passed to the master stage output, QM. During this period, the slave stage is in the hold mode, keeping its previous value using feedback. On the rising edge of the clock, the master slave stops sampling the input, and the slave stage starts sampling. During the high phase of the clock, the slave stage samples the output of the master stage (QM), while the master stage remains in a hold mode. Since QM is constant during the high phase of the clock, the output Q makes only one transition per cycle. The value of Q is the value of D right before the rising edge of the clock, achieving the positive edge-triggered effect. A negative edge-triggered register can be constructed using the same principle by simply switching the order of the positive and negative latch (i.e., placing the positive latch first).

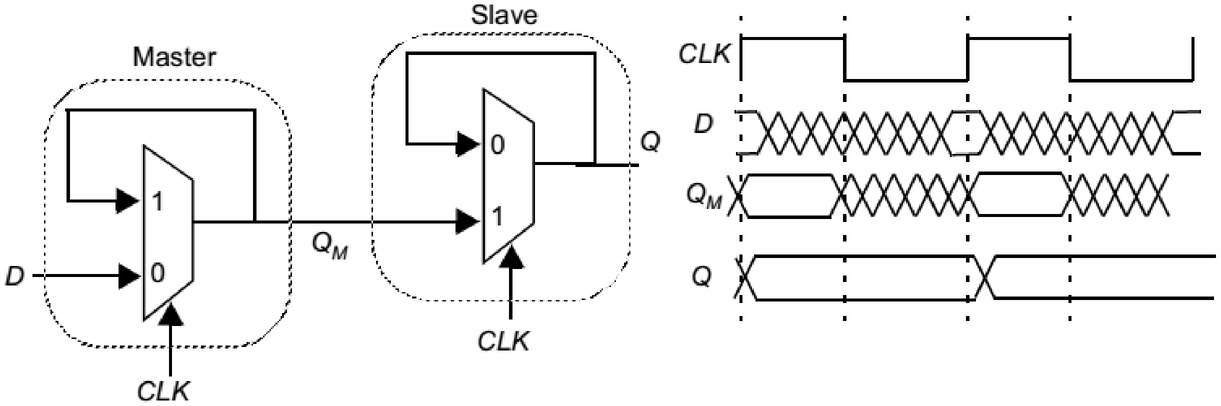


Fig: Positive edge-triggered register based on a master-slave configuration.

A complete transistor level implementation of a the master-slave positive edge-triggered register is shown. The multiplexer is implemented using transmission gates as discussed in the previous section. When clock is low (CLK=0).T1 is on and T2 is off, and the D input is sampled onto node QM. During this period, T3 is off and T4 is on and the cross-coupled inverters (I5, I6) holds the state of the slave latch. When the clock goes high, the master stage stops sampling the input and goes into aholdmode.T1 is off and T2 is on, and the cross coupled inverters I3 and I4 holds the state of QM. Also, T3 is on and T4 is off, and QM is copied to the output Q.

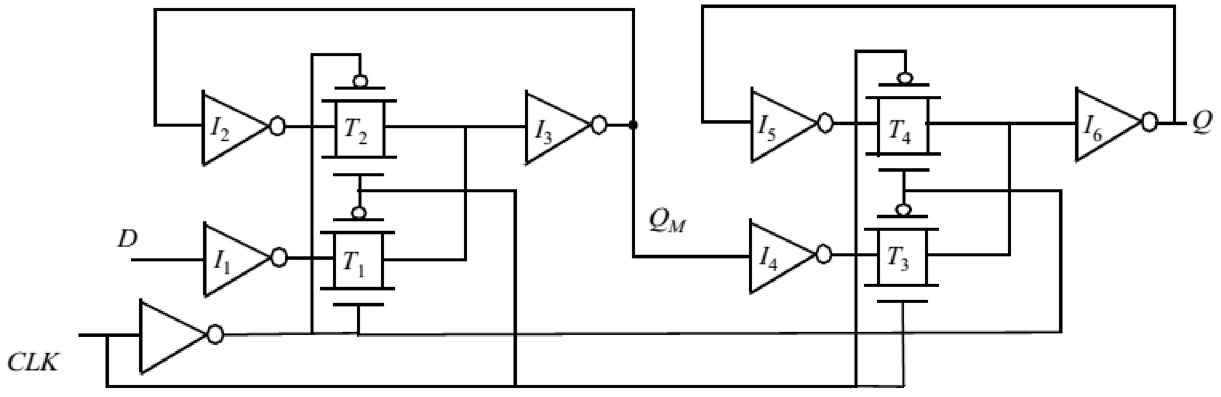


Figure 33: Transistor-level implementation of a master-slave postive edge-triggered register using multiplexers.

DYNAMIC LATCHES AND REGISTERS

Storage in a static sequential circuit relies on the concept that a cross-coupled inverter pair produces a bistable element and can thus be used to memorize binary values. This approach has the useful property that a stored value remains valid as long as the supply voltage is applied to the circuit, hence the name static. The major disadvantage of the static gate, however, is its complexity. When registers are used in computational structures that are constantly clocked such as pipelined data path, the requirement that the memory should hold state for extended periods of time can be significantly relaxed.

This results in a class of circuits based on temporary storage of charge on parasitic capacitors. The principle is exactly identical to the one used in dynamic logic — charge stored on a capacitor can be used to represent a logic signal. The absence of charge denotes a 0, while its presence stands for a stored 1. No capacitor is ideal, unfortunately, and some charge leakage is always present. A stored value can hence only be kept for a limited amount of time, typically in the range of milliseconds. If one wants to preserve signal integrity, a periodic refresh of its value is necessary. Hence the name dynamic storage. Reading the value of the stored signal from a capacitor without disrupting the charge requires the availability of a device with a high input impedance.

Dynamic Transmission-Gate Based Edge-triggered Registers

A fully dynamic positive edge-triggered register based on the master-slave concept is shown.When CLK= 0, the input data is sampled on storage node 1, which has an equivalent capacitance of C1 consisting of the gate capacitance of I1 , the junction capacitance of T1, and the overlap gate capacitance of T1. During this period, the slave stage is in a hold mode, with node 2 in a high-impedance (floating) state. On the rising edge of clock, the transmission gate T2 turns on, and the value sampled on node 1 right before the rising edge propagates to the output Q(note that node 1 is stable during the high phase of the clock since the first transmission gate is turned off). Node 2 now stores the inverted version of node 1. This implementation of an edge-triggered register is very efficient as it requires only 8 transistors. The sampling switches can be implemented using NMOS-only pass transistors, resulting in an even-simpler 6 transistor implementation. The reduced transistor count is attractive for high-performance and low-power systems.

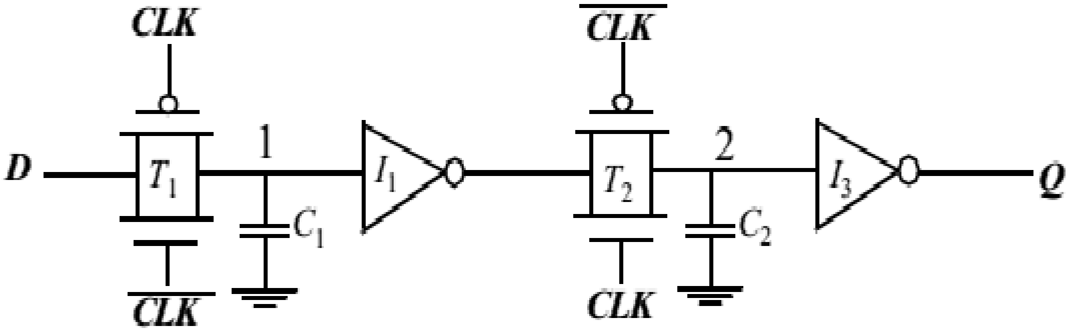


Figure 33: Dynamic edge-triggered register.

The set-up time of this circuit is simply the delay of the transmission gate, and corresponds to the time it takes node 1 to sample the D input. The hold time is approximately zero, since the transmission gate is turned off on the clock edge and further inputs changes are ignored. The propagation delay(tc-q) is equal to two inverter delays plus the delay of the transmission gate T2.

One important consideration for such a dynamic register is that the storage nodes (i.e., the state) has to be refreshed at periodic intervals to prevent a loss due to charge leakage, due to diode leakage as well as sub-threshold currents. In data path circuits, the refresh rate is not an issue since the registers are periodically clocked, and the storage nodes are constantly updated.

Clock overlap is an important concern for this register. Consider the clock waveforms shown.

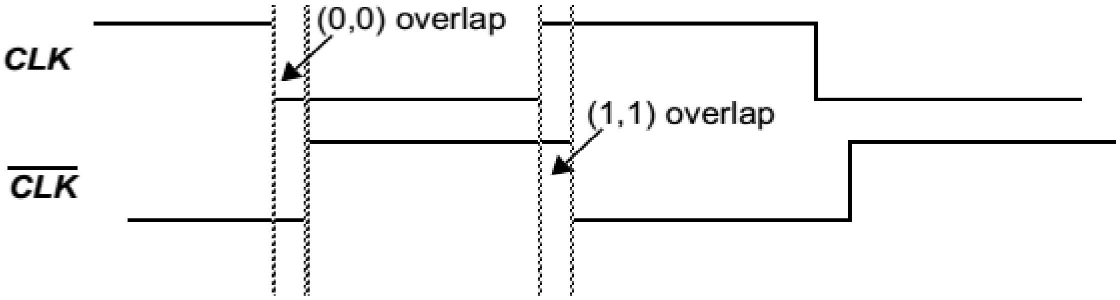


Figure 35: Impact of non-overlapping clocks.

During the 0-0 overlap period, the NMOS of T1 and the PMOS ofT2 are simultaneously on, creating a direct path for data to flow from the D input of the register to the Q output. This is known as a race condition. The output Q can change on the falling edge if the overlap period is large — obviously an undesirable effect for a positive edge-triggered register. The same is true for the 1-1 overlap region, where an input-output path exists through the PMOS of T1 and the NMOS ofT2. The latter case is taken care off by enforcing a hold time constraint. That is, the data must be stable during the high-high overlap period. The former situation (0-0 overlap) can be addressed by making sure that there is enough delay between the D input and node 2 ensuring that new data sampled by the master stage does not propagate through to the slave stage. Generally the built in single inverter delay should be sufficient and the overlap period constraint is given as:

$$t_{overlap0-0} < t_{T1} + t_{I1} + t_{T2}$$

Similarly, the constraint for the 1-1 overlap is given as:

$$t_{hold} > t_{overlap1-1}$$

C^{2}MOS Dynamic Register: A Clock Skew Insensitive Approach

The C^{2}MOS Register

The register operates in two phases.

1. CLK =0: The first tri-state driver is turned on, and the master stage acts as an inverter sampling the inverted version of D on the internal node X. The master stage is in the evaluation mode. Meanwhile, the slave section is in a high-impedance mode, or in ahold mode. Both transistors M7 andM8 are off, decoupling the output from the input. The output Q retains its previous value stored on the output capacitor $C_{L2.}$

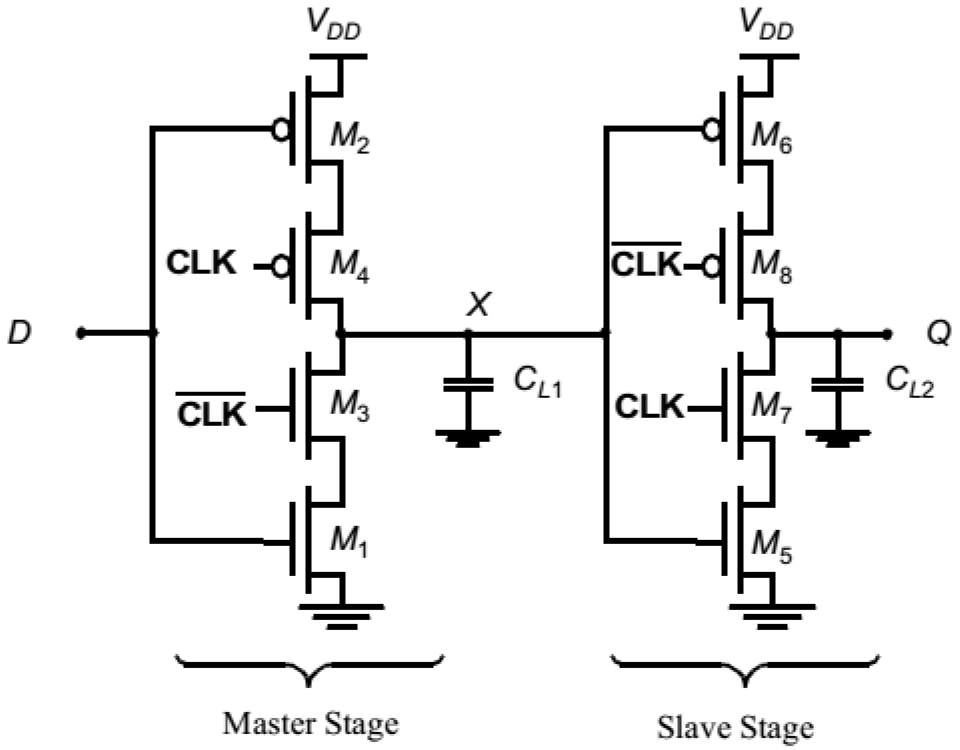


Figure 36: C^2MOS master-slave positive edge-triggered register.

2.The roles are reversed when CLK = 1: The master stage section is in hold mode (M3-M4 off), while the second section evaluates (M7-M8 on). The value stored onCL1 ropagates to the output node through the slave stage which acts as an inverter.

The overall circuit operates as a positive edge-triggered master-slave register — very similar to the transmission-gate based register presented earlier. However, there is an important difference:

A C^2MOS register with CLK-~~CLK~~ clocking is insensitive to overlap, as long as the rise and fall times of the clock edges are sufficiently small.

Dual-edge Triggered Registers

It is also possible to design sequential circuits that sample the input on both edges. The advantage of this scheme is that a lower frequency clock (half of the original rate) is distributed for the same functional throughput, resulting in power savings in the clock distribution network.

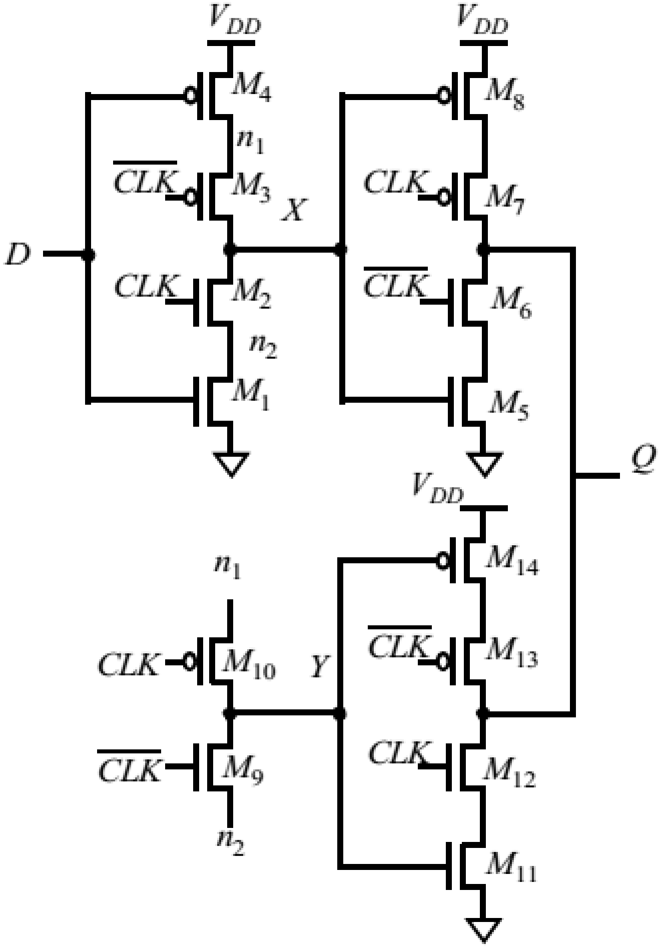


Figure 37: C^2MOS based dual-edge triggered register

This is a modification of the C^2MOS register to enable sampling on both edges. It consists of two parallel master-slavebased edge-triggered registers, whose outputs are multiplexed using the tri-state drivers.

When clock is high, the positive latch composed of transistorsM1-M4 is sampling the inverted D input on node X. Node Y is held stable, since devices M9 and M10 are turned off. On the falling edge of the clock, the top slave latch M5-M8 turns on, and drives the inverted value of X to the Q output. During the low phase, the bottom master latch (M1,M4, M9, M10) is turned on, sampling the inverted D input on node Y. Note that the devices M1 and M4 are reused, reducing the load on the D input. On the rising edge, the bottom slave latch conducts, and drives the inverted version of Y on node Q. Data hence changes on both edges. Note that the slave latches operate in a complementary fashion — this is, only one of them is turned on during each phase of the clock.

True Single-Phase Clocked Register (TSPCR)

In the two-phase clocking schemes described above, care must be taken in routing the two clock signals to ensure that overlap is minimized. While the C2MOS provides a skew-tolerant solution, it is possible to design registers that only use a single phase clock. The True Single-Phase Clocked Register (TSPCR) proposed by Yuan and Svensson uses a single clock(without an inverse clock).

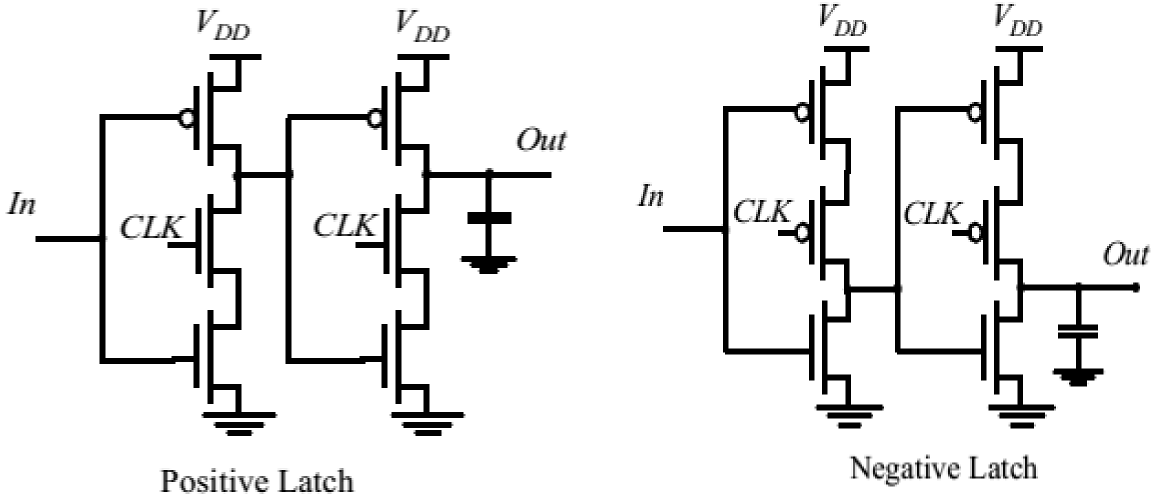


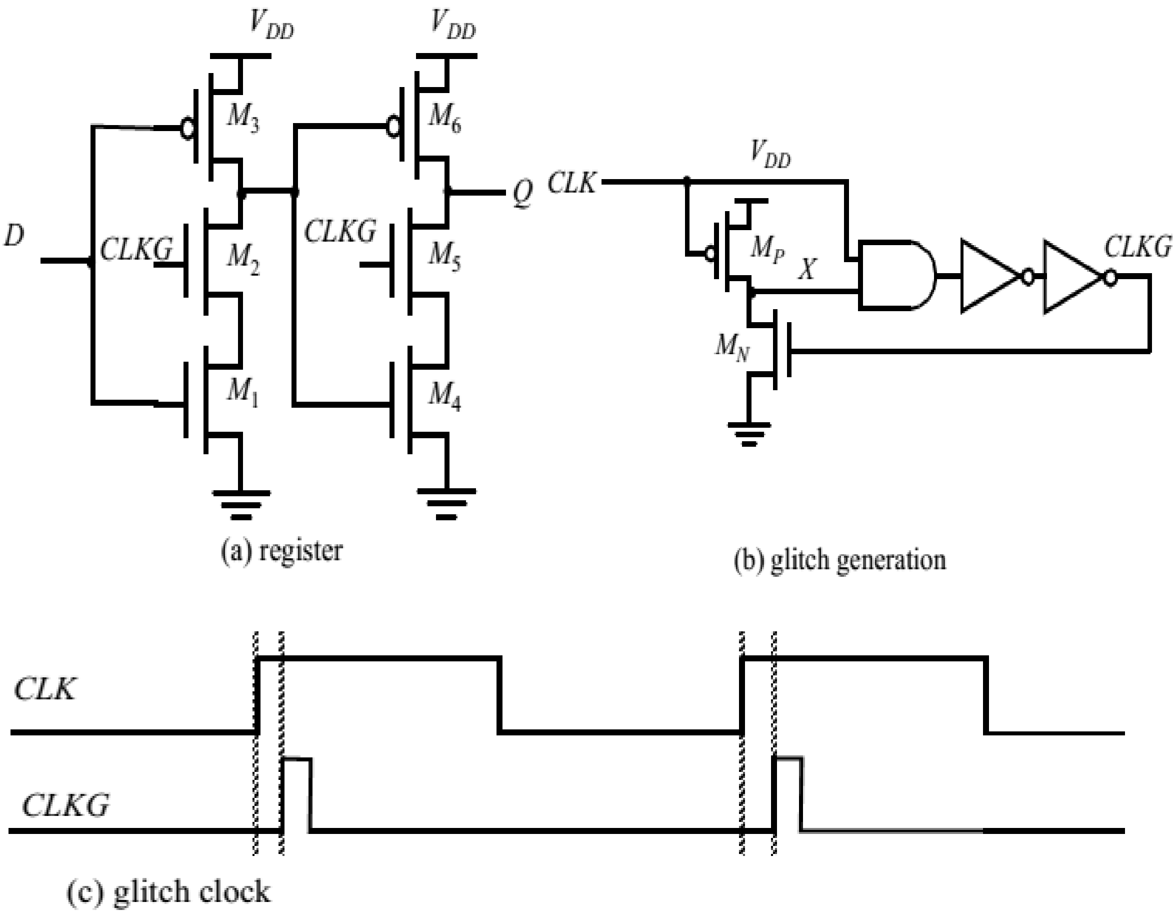
Figure 38: True Single Phase Latches.

For the positive latch, when CLK is high, the latch is in the transparent mode and corresponds to two cascaded inverters; the latch is non-inverting, and propagates the input to the output. On the other hand, when CLK= 0, both inverters are disabled, and the latch is in hold-mode. Only the pull-up networks are still active, while the pull-down circuits are deactivated. As a result of the dual-stage approach, no signal can ever propagate from the input of the latch to the output in this mode. A register can be constructed by cascading positive and negative latches. The clock load is similar to a conventional transmission gate register, or C^2MOS register. The main advantage is the use of a single clock phase. The disadvantage is the slight increase in the number of transistors — 12 transistors are required.

TSPC offers an additional advantage: the possibility of embedding logic functionality into the latches. This reduces the delay overhead associated with the latches. In addition to performing the latching function. While the set-up time of this latch has increased over the one, the overall performance of the digital circuit (that is, the clock period of a sequential circuit) has improved: the increase inset-up time is typically smaller than the delay of an AND gate. This approach of embedding logic into latches has been used extensively in the design of the EV4 DEC Alpha microprocessor and many other high performance processors.

PULSE BASED REGISTERS

A fundamentally different approach for constructing a register uses pulse signals. The idea is to construct a short pulse around the rising (or falling) edge of the clock. This pulse acts as the clock input to a latch (e.g., a TSPC flavor), sampling the input only in a short window. Race conditions are thus avoided by keeping the opening time (i.e, the transparent period) of the latch very short. The combination of the glitch generation circuitry and the latch results in a positive edge-triggered register.



(a) register

(b) glitch generation

(c) glitch clock

.

Figure 39: Glitch latch - timing generation and register.

When CLK= 0, node X is charged up to VDD(MN is off since CLKG is low). On the rising edge of the clock, there is a short period of time when both inputs of the AND gate are high, causing CLKG to go high. This in turn activates MN, pulling X and eventually CLKG low.The length of the pulse is controlled by the delay of the AND gate and the two inverters. Note that there exists also a delay between the rising edges of the input clock (CLK) and the glitch clock (CLKG)— alsoequaltothedelayoftheANDgateandthetwoinverters.Ifeveryregisteronthechip uses the same clock generation mechanism, this sampling delay does not matter. However, process variations and load variations may cause the delays through the glitch clock circuitry to be different.

If set-up time and hold time are measured in reference to the rising edge of the glitch clock, the set-up time is essentially zero, the hold time is equal to the length of the pulse (if the contamination delay is zero for the gates), and the propagation delay(tc-q) equals two gate delays. The advantage of the approach is the reduced clock load and the small number of transistors required. The glitch-generation circuitry can be amortized over multiple register bits. The disadvantage is a substantial increase in verification complexity. This has prevented a wide-spread use. They do however provide an alternate approach to conventional schemes, and have been adopted in some high performance processors

SENSE AMPLIFIER BASED REGISTER

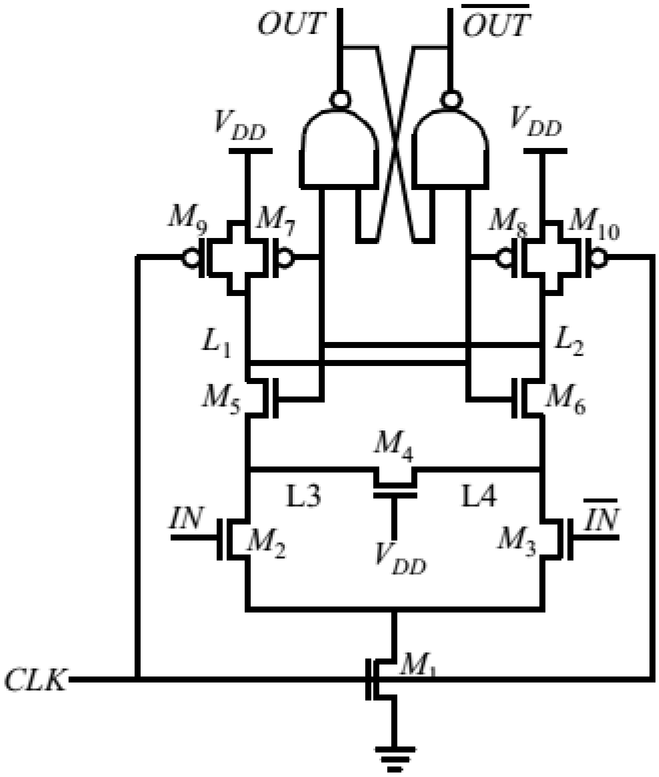


Figure 40: Positive edge-triggered register based on sense-amplifier.

Sense amplifier circuits accept small input signals and amplify them to generate rail-to-rail swings. As we will see, sense amplifier circuits are used extensively in memory cores and in low swing bus drivers to amplify small voltage swings present in heavily loaded wires. There are many techniques to construct these amplifiers,with the use of feedback (e.g., cross-coupled inverters) being one common approach. The circuit shown uses a precharged front-end amplifier that samples the differential input signal on the rising edge of the clock signal. The outputs of front-end are fed into a NAND cross-coupled SR FF that holds the data and gurantees that the differential outputs switch only once per clock cycle. The differential inputs in this implementation don’t have to have rail-to-rail swing and hence this register can be used as a receiver for a reduced swing differential bus.

The core of the front-end consists of a cross-coupled inverter (M5-M8) whose outputs (L1andL2) are precharged using devices M9andM10 during the low phase of the clock. As a result, PMOS transistorsM7 andM8 to be turned off and the NAND FF is holding its previous state. Transistor M1is similar to an evaluate switch in dynamic circuits and is turned off ensuring that the differential inputs don’t affect the output during the low phase of the clock. On the rising edge of the clock, the evaluate transistor turns on and the differential input pair (M2andM3) is enabled, and the difference between the input signals is amplified on the output nodes onL1 andL2. The cross-coupled inverter pair flips to one of its the stable states based on the value of the inputs. For example, if IN is 1, L1 is pulledto0,andL2 remains at VDD. Due to the amplifying properties of the input stage, it is not necessary for the input to swing all the way up to VDD and enables the use of low swing signaling on the input wires.

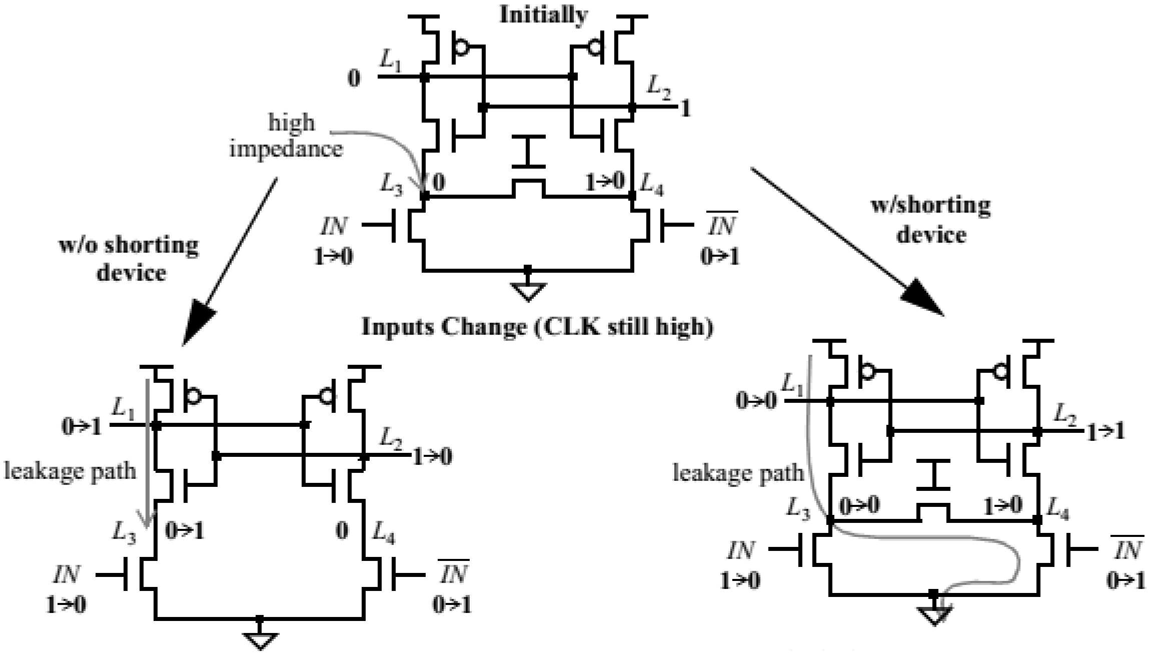


Figure 41: The need for the shorting transistor M4

The shorting transistor,M4, is used to provide a DC leakage path from either node L3,orL4, to ground. This is necessary to accommodate the case where the inputs change their value after the positive edge of CLK has occurred, resulting in eitherL3orL4 being left in a high-impedance state with a logical low voltage level stored on the node. Without the leakage path that node would be susceptible to charging by leakage currents. The latch could then actually change state prior to the next rising edge of CLK!

NORA-CMOS—A Logic Style for Pipelined Structures

The latch-based pipeline circuit can also be implemented using C^{2}MOS latches. The operation is similar to the one discussed above. This topology has one additional, important property:

A C^{2}MOS-based pipelined circuit is race-free as long as all the logic functions F(implemented using static logic) between the latches are noninverting.

The reasoning for the above argument is similar to the argument made in the construction of a C^{2}MOS register. During a (0-0) overlap between CLK and CLK, al l C^{2}MOS latches, simplify to pure pull-up networks. The only way a signal can race from stage to stage under this condition is when the logic function F is inverting, where F is replaced by a single, static CMOS inverter. Similar considerations are valid for the (1-1) overlap.

Based on this concept, a logic circuit style called NORA-CMOS was conceived It combines C^{2}MOS pipeline registers and NOR A dynamic logic function blocks. Each module consists of a block of combinational logic that can be a mixture of static and dynamic logic, followed by a C^{2}MOS latch. Logic and latch are clocked in such a way that both are simultaneously in either evaluation, or hold (precharge) mode. A block that is in evaluation during CLK= 1 is called a CLK-module, while the inverse is called a ~~CLK~~-module.

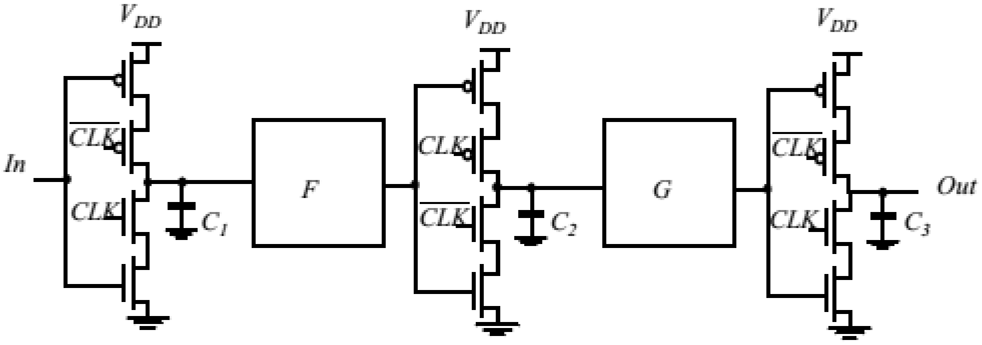
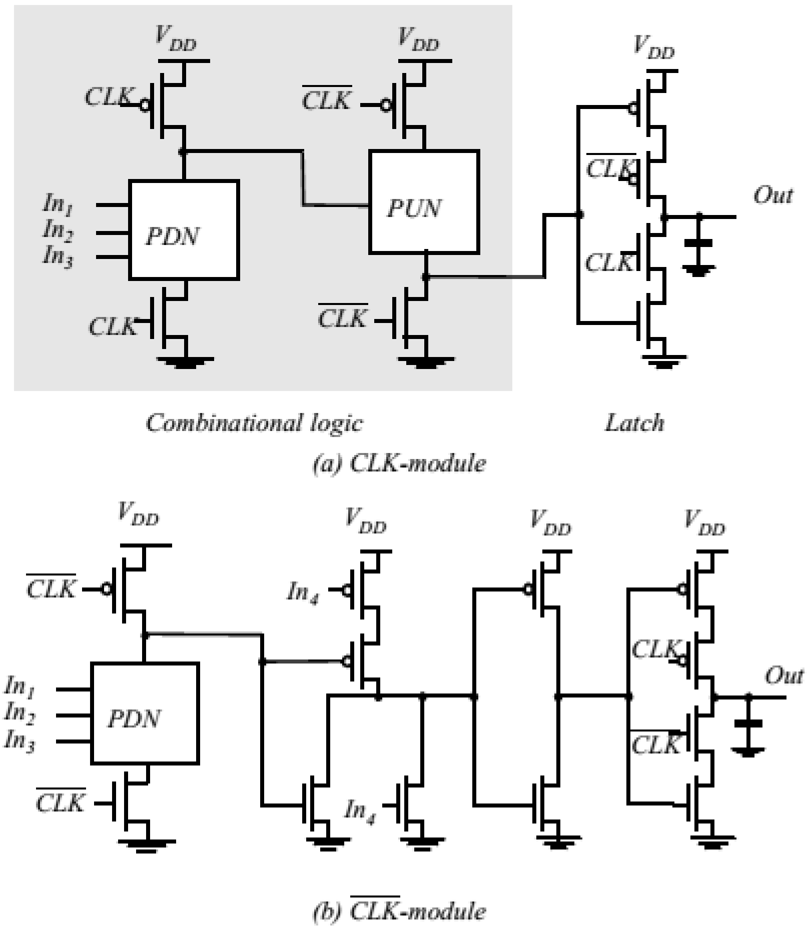


Figure 7.42 Pipelined datapath using C^2MOS latches.

A NORA data path consists of a chain of alternating CLK and ~~CLK~~ modules. While one class of modules is precharging with its output latch in hold mode, preserving the previous output value, the other class is evaluating. Data is passed in a pipelined fashion from module to module. NORA offers designers a wide range of design choices. Dynamic and static logic can be mixed freely, and both CLKp and CLKn dynamic blocks can be used in cascaded or in pipelined form. With this freedom of design, extra inverter stages, as required in DOMINO-CMOS, are most often avoided.



Combinational logic

Latch

(a) CLK-module

(b) $\overline{CLK}$-module

Figure 42: Examples of NORA CMOS Modules

Semiconductor Memories:

Semiconductor based electronics is the foundation to the information technology society we live in today. Ever since the first transistor was invented way back in 1948, the semiconductor industry has been growing at a tremendous pace. Semiconductor memories and microprocessors are two major fields, which are benefited by the growth in semiconductor technology.

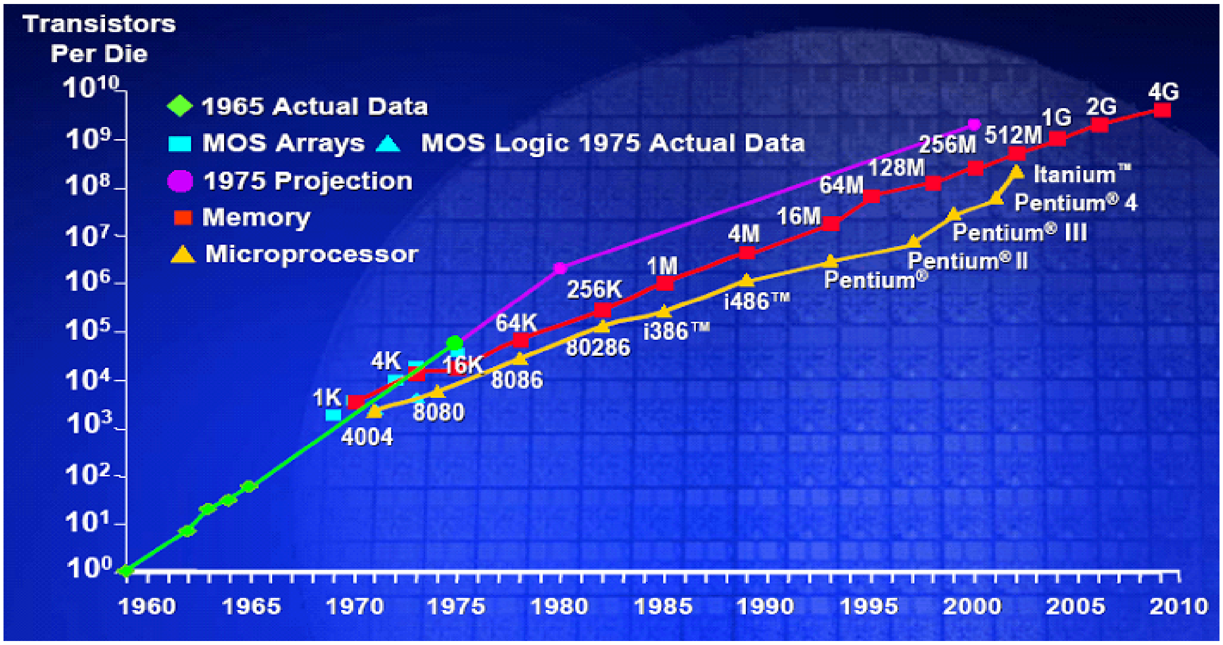


Fig: Increasing memory capacity over the years

The technological advancement has improved performance as well as packing density of these devices over the years Gordon Moore made his famous observation in 1965, just four years after the first planar integrated circuit was discovered. He observed an exponential growth in the number of transistors per integrated circuit in which the number of transistors nearly doubled every couple of years. This observation, popularly known as Moore's Law, has been maintained and still holds true today. Keeping up with this law, the semiconductor memory capacity also increases by a factor of two every year.

Memory Classification

Size: Depending upon the level of abstraction, different means are used to express the size of the memory unit. A circuit designer usually expresses memory in terms of bits, which are equivalent to the number of individual cells need to store the data. Going up one level in the hierarchy to the chip design level, it is common to express memory in terms of bytes, which is a group of 8 bits. And on a system level, it can be expressed in terms of words or pages, which are in turn collection of bytes.

Function: Semiconductor memories are most often classified on the basis of access patterns, memory functionality and the nature of the storage mechanism. Based on the access patterns, they can be classified into random access and serial access memories. A random access memory can be accessed for read/write in a random fashion. On the other hand, in serial access memories, the data can be accessed only in a serial fashion. FIFO (First In First Out) and LIFO (Last In Last Out) are examples of serial memories. Most of the memories fall under the random access types.

Based on their functionalities, memory can be broadly classified into Read/Write memories and Read-only memories. As the name suggests, Read/Write memory offers both read and write operations and hence is more flexible. **SRAM (Static RAM)** and **DRAM (Dynamic RAM)** come under this category. A Read-only memory on the other hand encodes the information into the circuit topology. Since the topology is hardwired, the data cannot be modified; it can only be read. However **ROM** structures belong to the class of the **nonvolatile memories**. Removal of the supply voltage does not result in a loss of the stored data. Examples of such structures include **PROMs, ROMs and PLDs**. The most recent entry in the filed are memory modules that can be classified as nonvolatile, yet offer both read and write functionality. Typically, their write operation takes substantially longer time than the read operation. An **EPROM, EEPROM** and **Flash memory** fall under this category.

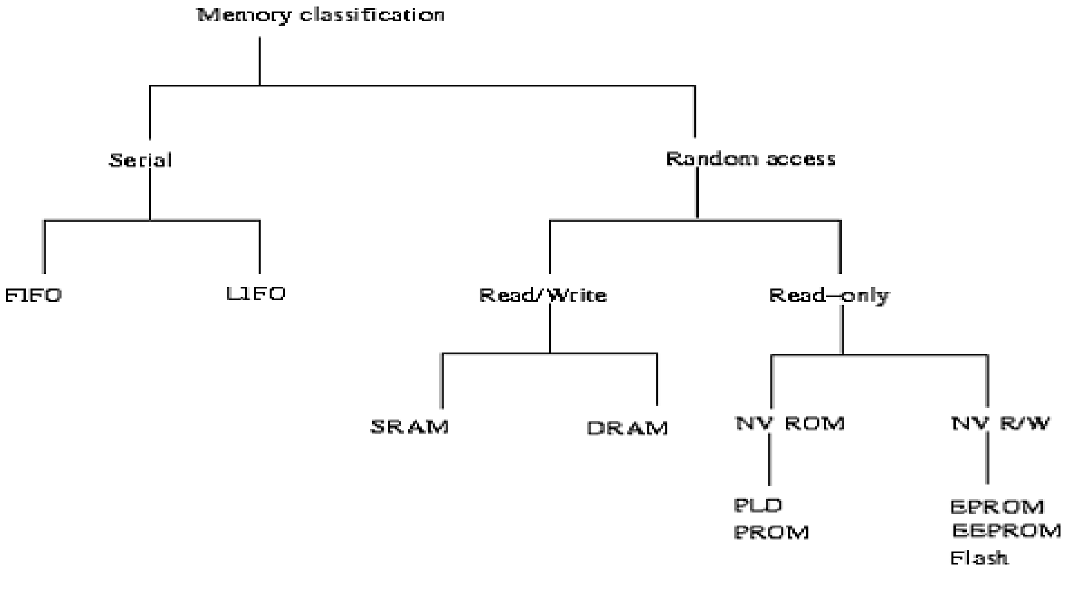
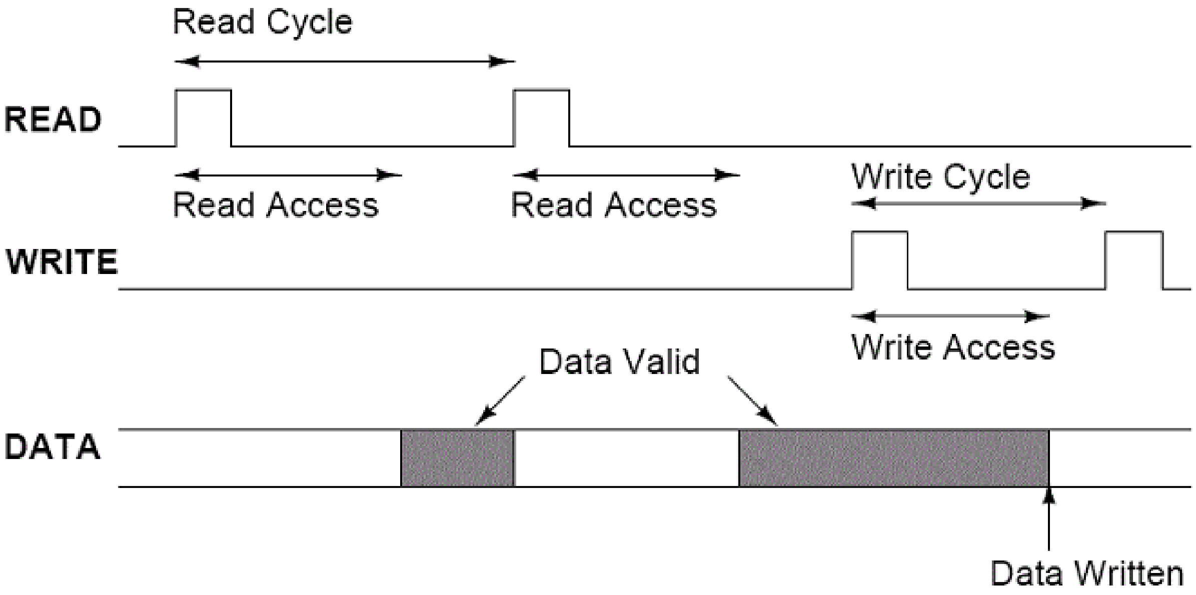


Figure 44: Classification of memories

Timing Parameters: The time it takes to retrieve data from the memory is called the readaccess time. This is equal to the delay between the read request and the moment the data is available at the output. Similarly, write-access time is the time elapsed between a write request and the final writing of the input data into the memory. Finally, there is another important parameter, which is the cycle time (read or write), which is the minimum time required between two successive read or write cycles. This time is normally greater than the access time.



Memory Architecture and Building Blocks

The straightforward way of implementing a N-word memory is to stack the words in a linear fashion and select one word at a time for reading or writing operation by means of a select bit. Only one such select signal can be high at a time. Though this approach is quite simple, one runs into a number of problems when trying to use it for larger memories. The number of interface pins in the memory module varies linearly with the size of the memory and this can easily run into huge values.

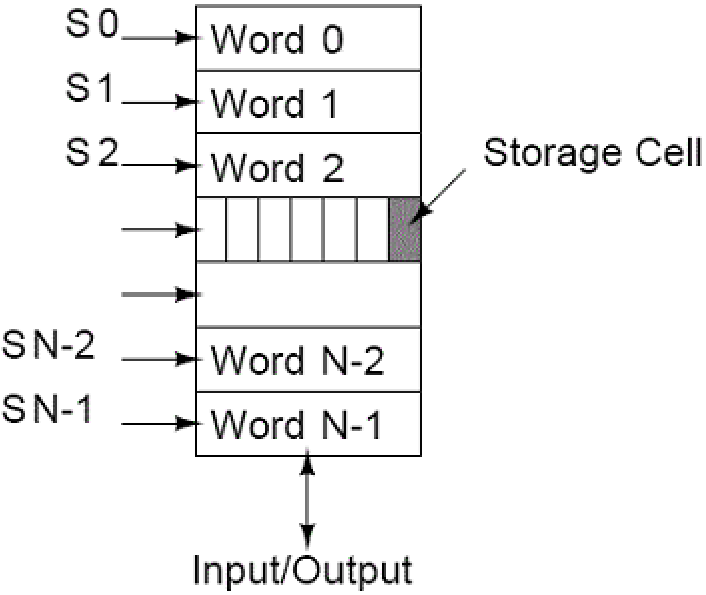


Figure 45: Basic Memory Organization

To overcome this problem, the address provided to the memory module is generally encoded. A decoder is used internally to decode this address and make the appropriate select line high. With 'k' address pins, 2 K number of select pins can be driven and hence the number of interface pins will get reduced by a factor of log2N.

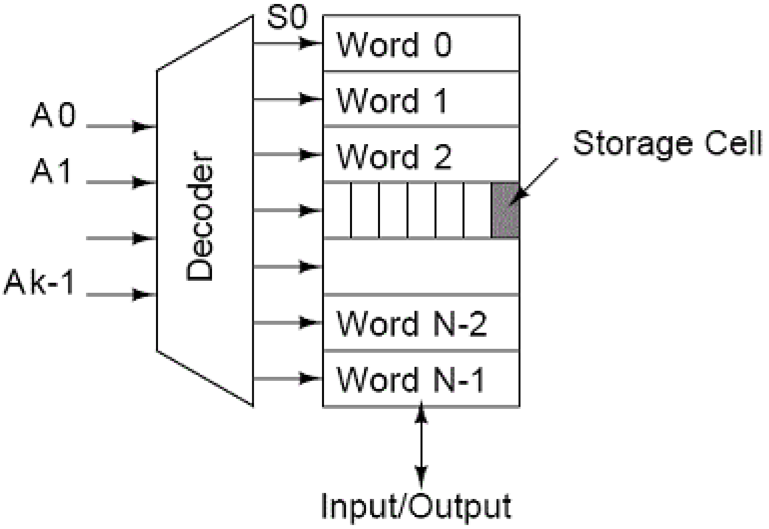


Figure 46: Memory with decoder logic

Though this approach resolves the select problem, it does not address the issues of the memory aspect ratio. For an N-word memory, with a word length of M, the aspect ratio will be nearly N:M, which is very difficult to implement for large values of N. Also such sort of a design slows down the circuit very much. This is because,the vertical wires connecting the storage cells to the inputs/outputs become excessively long. To address this problem, memory arrays are organized so that the vertical and horizontal dimensions are of the same order of magnitude, making the aspect ratio close to unity. To route the correct word to the input/output terminals, an extra circuit called column decoder is needed. The address word is partitioned into column address (A0 to AK-1) and row address (AK-1 to AL-1). The row address enables one row of the memory for read/write, while the column address picks one particular word from the selected row.

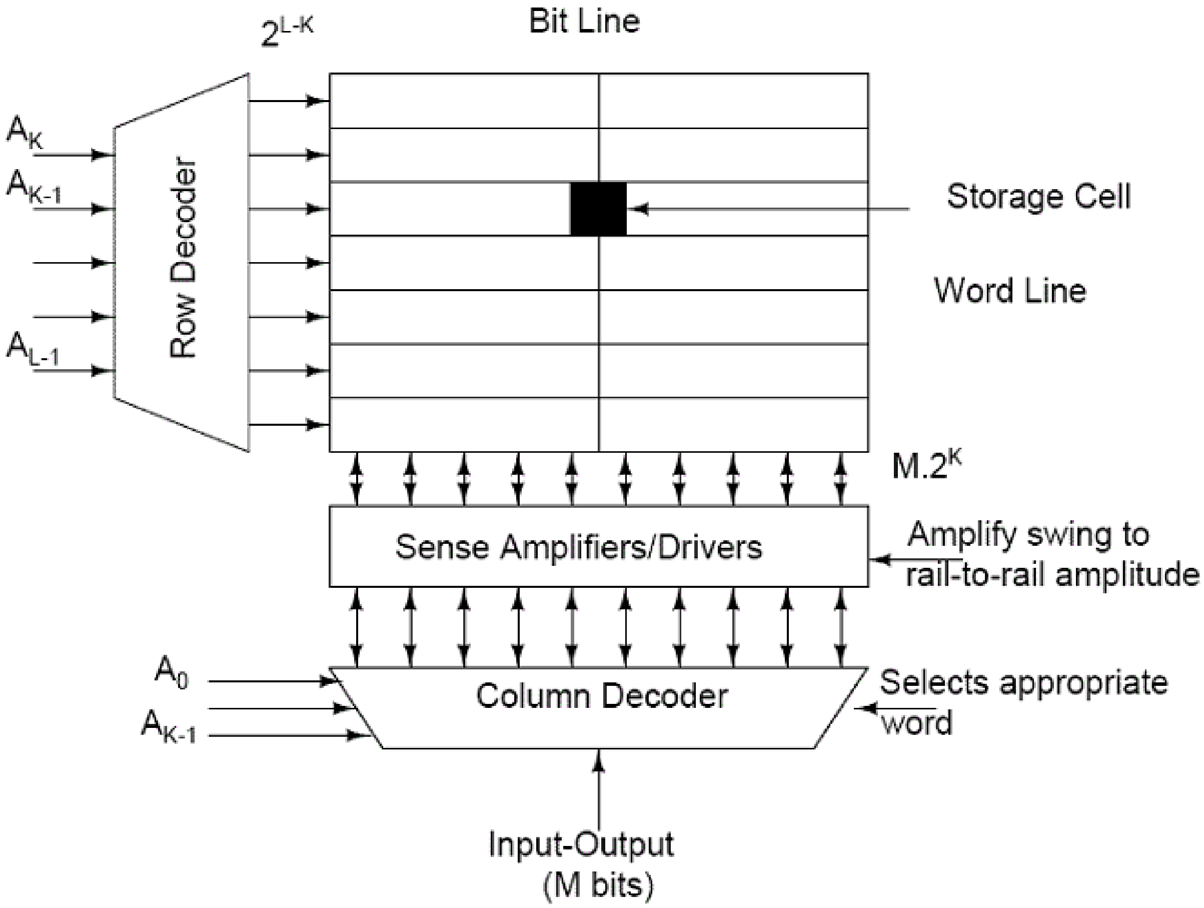


Figure 47: Memory with row and column decoders

Static and Dynamic RAMs

RAMs are of two types, static and dynamic. Circuits similar to basic D flip-flop are used to construct static RAMs (SRAMs) internally. A typical SRAM cell consists of six transistors which are connected in such a way as to form a regenerative feedback. In contrast to DRAM, the information stored is stable and does not require clocking or refresh cycles to sustain it. Compared to DRAMs, SRAMs are much faster having typical access times in the order of a few nanoseconds. Hence SRAMs are used as level 2 cache memory.

Dynamic RAMs do not use flip-flops, but instead are an array of cells, each containing a transistor and a tiny capacitor. '0's and '1's can be stored by charging or discharging the capacitors. The electric charge tends to leak out and hence each bit in a DRAM must be refreshed every few milliseconds to prevent loss of data. This requires external logic to take care of refreshing which makes interfacing of DRAMs more complex than SRAMs. This disadvantage is compensated by their larger capacities. A high packing density is achieved since DRAMs require only one transistor and one capacitor per bit. This makes them ideal to build main memories. But DRAMs are slower having delays in the order tens of nanoseconds. Thus the combination of static RAM cache and a dynamic RAM main memory attempts to combine the good properties of each.

Static Random Access Memory (SRAM)

SRAM Basics

The memory circuit is said to be static if the stored data can be retained indefinitely, as long as the power supply is on, without any need for periodic refresh operation. The data storage cell, i.e., the one-bit memory cell in the static RAM arrays, invariably consists of a simple latch circuit with two stable operating points. Depending on the preserved state of the two inverter latch circuit, the data being held in the memory cell will be interpreted either as logic '0' or as logic '1'. To access the data contained in the memory cell via a bit line, we need atleast one switch, which is controlled by the corresponding word line.

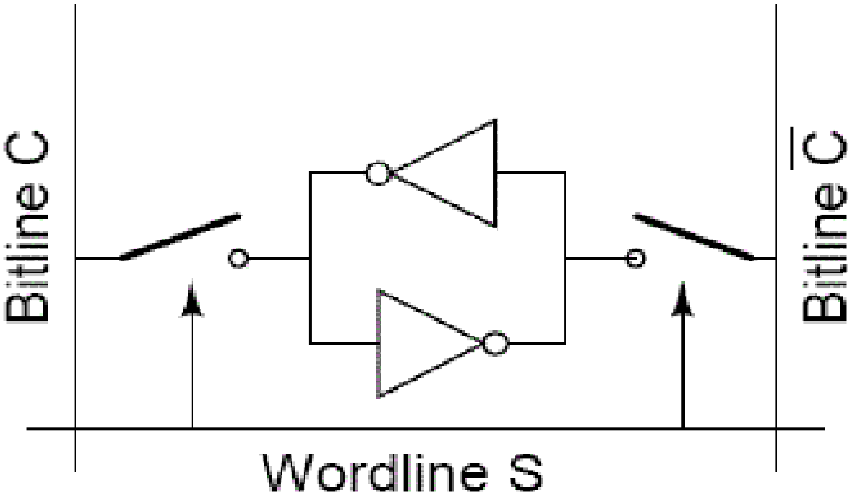


Figure 48: SRAM Cell

CMOS SRAM Cell

A low power SRAM cell may be designed by using cross-coupled CMOS inverters. The most important advantage of this circuit topology is that the static power dissipation is very small; essentially, it is limited by small leakage current. Other advantages of this design are high noise immunity due to larger noise margins, and the ability to operate at lower power supply voltage. The major disadvantage of this topology is larger cell size. The memory cell consists of simple CMOS inverters connected back to back, and two access transistors. The access transistors are turned on whenever a word line is activated for read or write operation, connecting the cell to the complementary bit line columns.

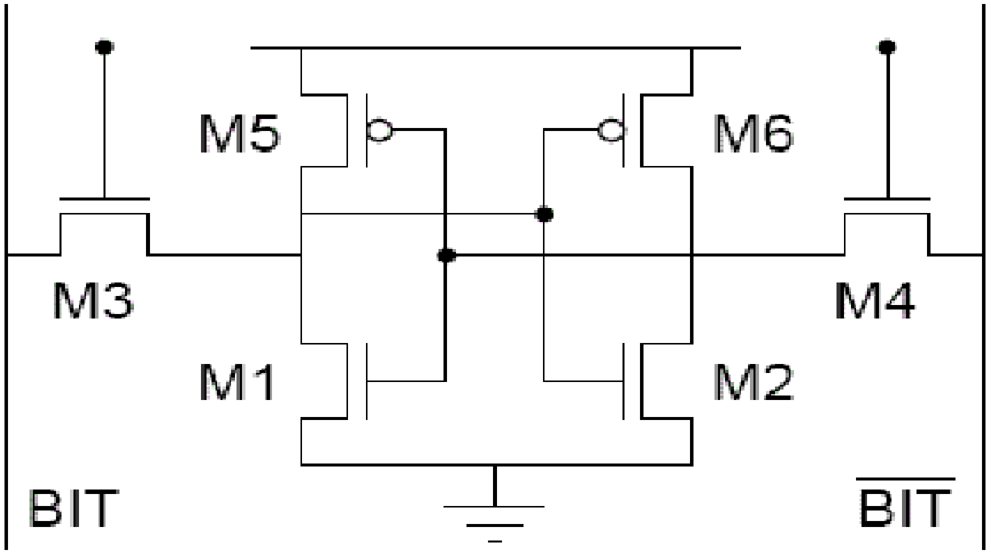


Figure 49: Full CMOS SRAM cell

CMOS SRAM Cell Design

To determine W/L ratios of the transistors, a number of design criteria must be taken into consideration. The two basic requirements, which dictate W/L ratios, are that the data read operation should not destroy the stored information in the cell. The cell should allow stored information modification during write operation. In order to consider operations of SRAM, we have to take into account, the relatively large parasitic column capacitance $C_{bit}$and $C_{\overline{bit}}$column pull-up transistors

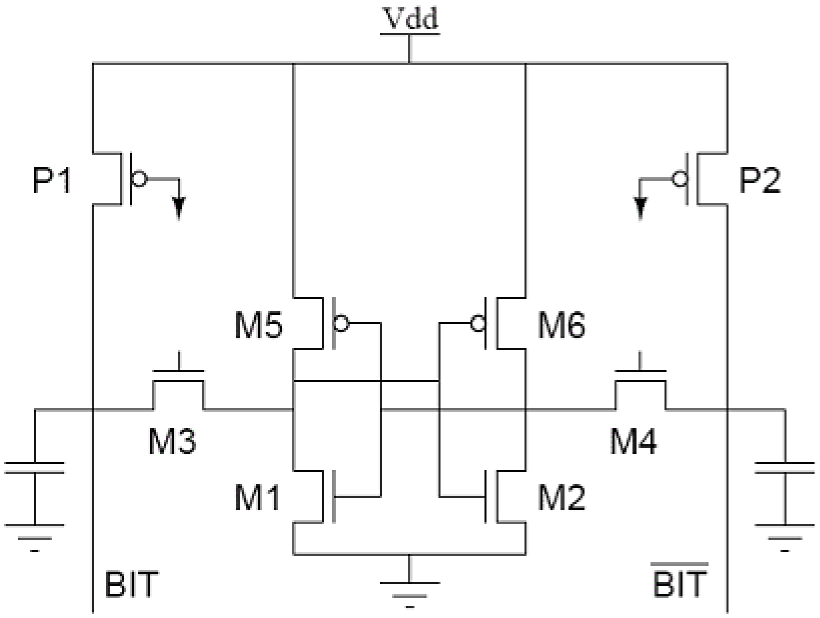
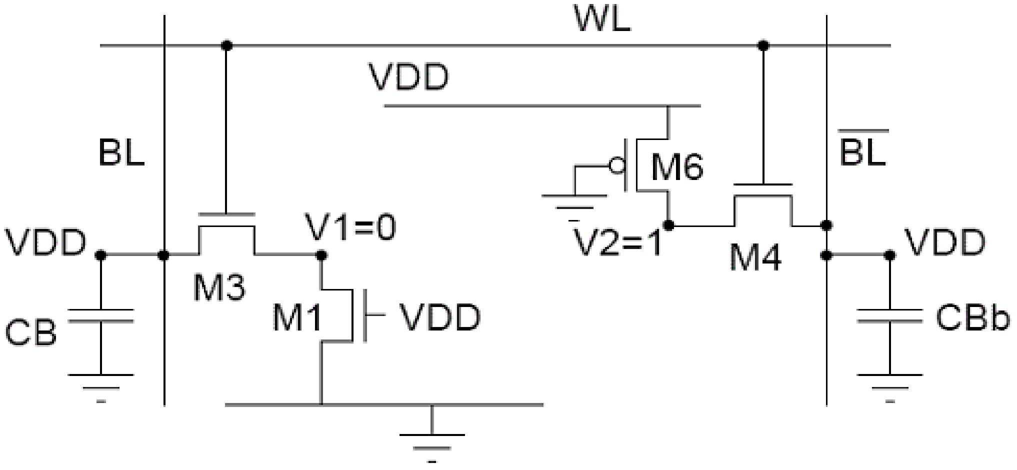


Figure 50: CMOS SRAM cell with precharge transistors

When none of the word lines is selected, the pass transistors M3 and M4 are turned off and the data is retained in all memory cells. The column capacitances are charged by the pull-up transistors P1 and P2. The voltages across the column capacitors reach VDD - VT.

READ Operation

Consider a data read operation, shown in Figure 28.41, assuming that logic '0' is stored in the cell. The transistors M2 and M5 are turned off, while the transistors M1 and M6 operate in linear mode. Thus internal node voltages are V1 = 0 and V2 = VDD before the cell access transistors are turned on.



Read Operation

After the pass transistors M3 and M4 are turned on by the row selection circuitry, the voltageC_{Bb} of will not change any significant variation since no current flows through M4. On the other hand M1 and M3 will conduct a nonzero current and the voltage level ofC_B will begin to drop slightly. The node voltage V1 will increase from its initial value of '0'V. The node voltage V1 may exceed the threshold voltage of M2 during this process, forcing an unintended change of the stored state. Therefore voltage must not exceed the threshold voltage of M2, so the transistor M2 remains turned off during read phase.

WRITE Operation

Consider the write '0' operation assuming that logic '1' is stored in the SRAM cell initially. Figure 28.51 shows the voltage levels in the CMOS SRAM cell at the beginning of the data write operation. The transistors M1 and M6 are turned off, while M2 and M5 are operating in the linear mode. Thus the internal node voltage V1 = VDD and V2 = 0 before the access transistors are turned on. The column voltage V_b is forced to '0' by the write circuitry. Once M3 and M4 are turned on, we expect the nodal voltage V2 to remain below the threshold voltage of M1, since M2 and M4 are designed.

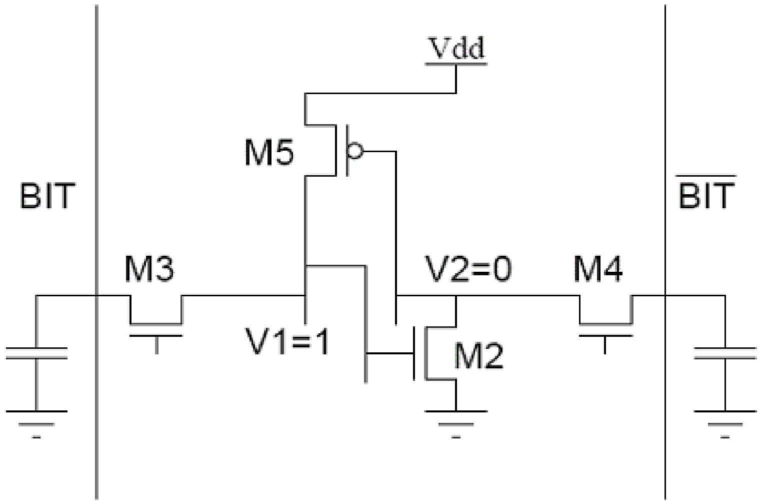


Figure 50: SRAM start of write '0'

The voltage at node 2 would not be sufficient to turn on M1. To change the stored information, i.e., to force V1 = 0 and V2 = VDD, the node voltage V1 must be reduced below the threshold voltage of M2, so that M2 turns off. When $V_1=V_{T,n}$ the transistor M3 operates in linear region while M5 operates in saturation region.

WRITE Circuit

The principle of write circuit is to assert voltage of one of the columns to a low level. This can be achieved by connecting either BIT or BIT' to ground through transistor M3 and either of M2 or M1. The transistor M3 is driven by the column decoder selecting the specified column. The transistor M1 is on only in the presence of the write enable signal and when the data bit to be written is '0'. The transistor M2 is on only in the presence of the write signal W'=0 and when the data bit to be written is '1'.

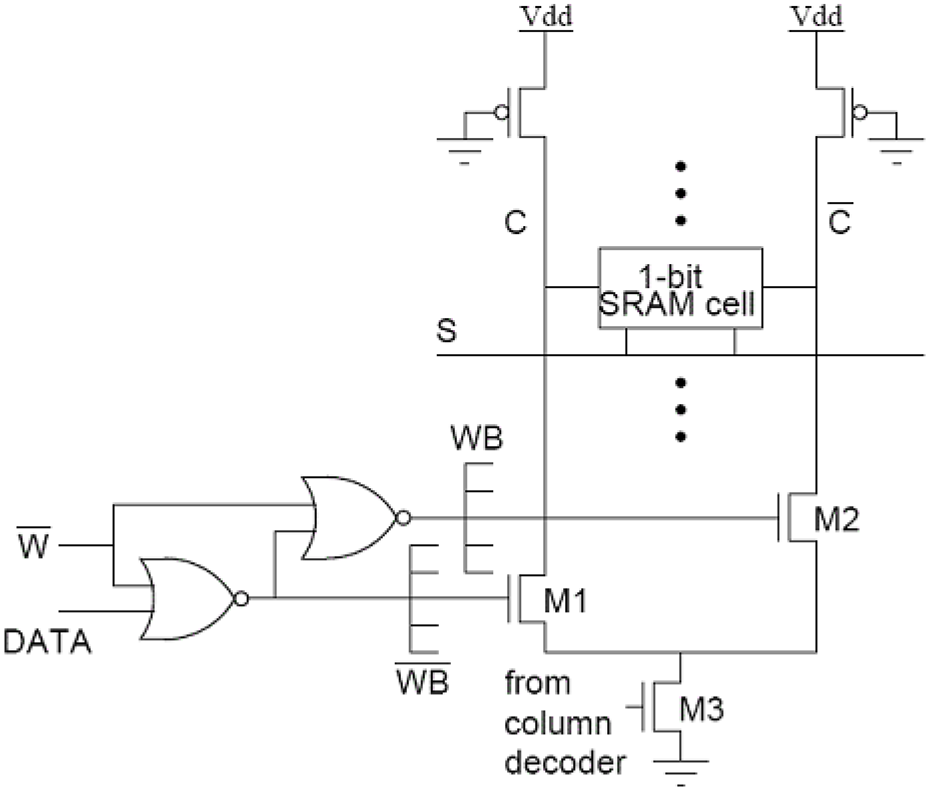


Figure 51: Circuit for write operation

DYNAMIC RANDOM ACCESS MEMORY (DRAM)

DRAM Basics

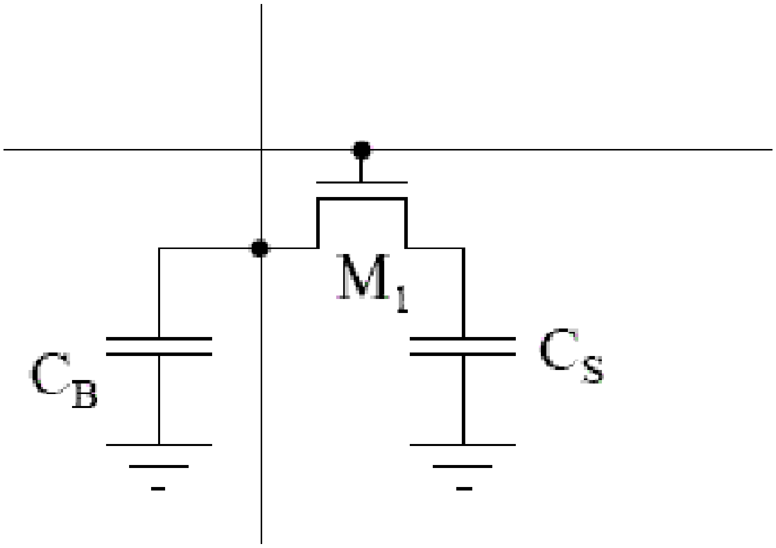


Fig: DRAM Cell

The C_S capacitor stores the charge for the cell. Transistor M1 gives the R/W access to the cell. C_B is the capacitance of the bit line per unit length.

Memory cells are etched onto a silicon wafer in an array of columns (bit lines) and rows (word lines). The intersection of a bit line and word line constitutes the address of the memory cell.

DRAM works by sending a charge through the appropriate column (CAS) to activate the transistor at each bit in the column. When writing, the row lines contain the state the capacitor should take on. When reading, the sense amplifier determines the level of charge in the capacitor. If it is more than 50%, it reads it as "1"; otherwise it reads it as "0". The counter tracks the refresh sequence based on which rows have been accessed in what order. The length of time necessary to do all this is so short that it is expressed in nanoseconds (billionths of a second). e.g. a memory chip rating of 70ns means that it takes 70 nanoseconds to completely read and recharge each cell.

The capacitor in a dynamic RAM memory cell is like a leaky bucket. Dynamic RAM has to be dynamically refreshed all of the time or it forgets what it is holding. This refreshing takes time and slows down the memory.

Semiconductor ROMs

Introduction

Read only memories are used to store constants, control information and program instructions in digital systems. They may also be thought of as components that provide a fixed, specified binary output for every binary input.

The read only memory can also be seen as a simple combinational Boolean network, which produces a specified output value for each input combination, i.e. for each address. Thus storing binary information at a particular address location can be achieved by the presence or absence of a data path from the selected row (word line) to the selected column (bit line), which is equivalent to the presence or absence of a device at that particular location.

The two different types of implementations of ROM array are:

- NOR-based ROM array
- NAND-based ROM array

NOR-based ROM Array

There are two different ways to implement MOS ROM arrays. Consider the first 4-bit X 4-bit memory array as shown in Figure 31.21. Here, each column consists of a pseudo nMOS NOR gate driven by some of the row signals, i.e., the word line.

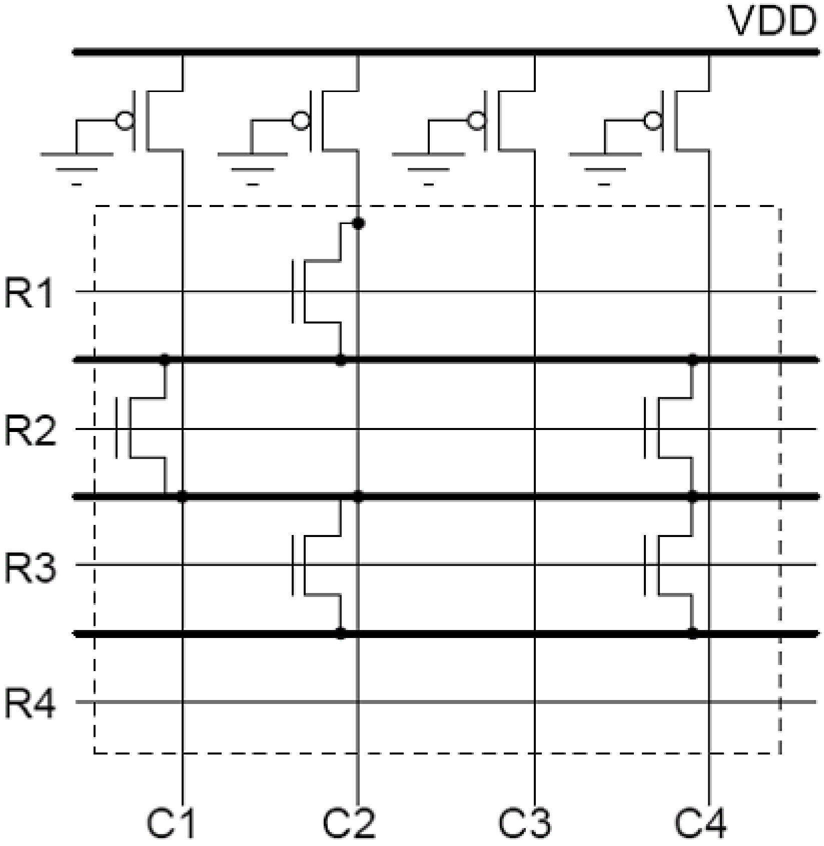


Fig: NOR-based ROM array

As we know, only one word line is activated at a time by raising tis voltage to VDD, while all other rows are held at a low votlage level. If an active transistor exists at the cross point of a column and the selected row, the column voltage is pulled down to the logic LOW level by that transistor. If no active transistor exists at the cross point, the column voltage is pulled HIGH by the pMOS load device. Thus, a logic "1"-bit is stored as the absence of an active transistor, while a logic "0"-bit is stored as the presence of an active transistor at the cross point.

NAND-based ROM Array

In this types of ROM array which is shown in Figure 31.31, each bit line consists of a depletion-load NAND gate, driven by some of the row signals, i.e. the word lines. In normal operation, all word lines are held at the logic HIGH voltage level except for the selected line, which is pulled down to logic LOW level. If a transistor exists at the cross point of a column and the selected row, that transistor is turned off and column voltage is pulled HIGH by the load device. On the other hand, if no transistor exists (shorted) at that particular cross point, the column voltage is pulled LOW by the other nMOS transistors in the multi-input NAND structure. Thus, a logic "1"-bit is stored by the presence of a transistor that can be deactivated, while a logic "0"-bit is stored by a shorted or normally ON transistor at the cross point.

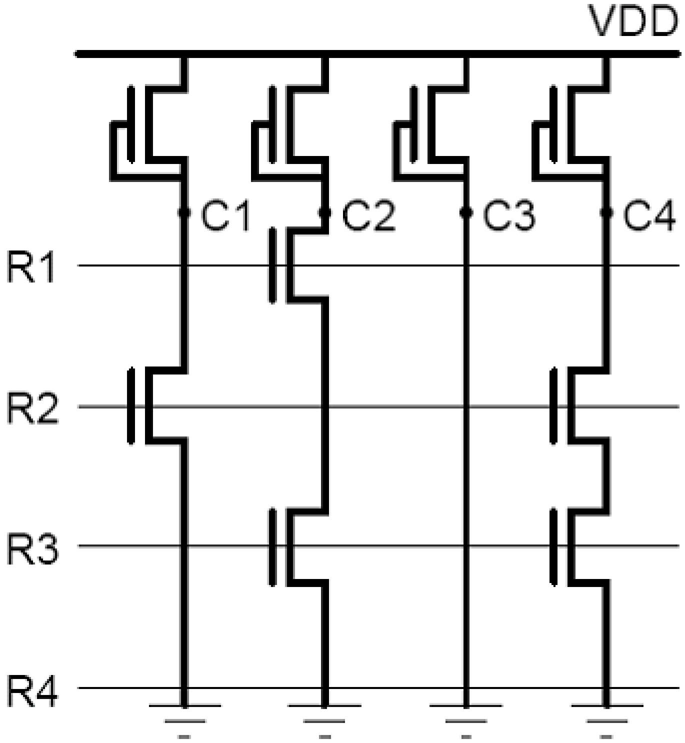


Figure 51: NAND-based ROM

Few special Examples of Memories

Erasable Programmable Read Only Memory (EPROM)

An EPROM is erased by shining ultraviolet light on the cells through a transparent window in the package. The UV radiation renders the oxide slightly conductive by direct generation of electron-hole pair in the material. The erasure process is slow and can take from seconds to several minutes, depending on the intensity of the UV source. Programming takes several (5-10) microseconds/word. Another problem with the process is limited endurance, that is, the number of erase/program cycles is generally limited to maximum of 1000, mainly as a result of UV erase procedure. Reliability is also an issue. The device threshold might vary with repeated programming cycles. Most EPROM memories therefore contain on-chip circuitry to control the value of thresholds to within a specified range during programming. Finally, the injection always entails a large channel current, as high as 0.5mA at a control gate voltage of 12.5V. This causes high power dissipation during programming. The EPROM cell is extremely simple and dense, making it possible to fabricate large memories at a low cost. EPROMs were therefore attractive in applications that do not require regular programming. Due to cost and reliability issues, EPROMs have fallen out of favor and have been replaced by Flash Memories.

Electrically Erasable Programmable Read Only Memory (EEPROM)

The major disadvantage of the EPROM approach is that erasure procedure has to occur "off system". This means the memory must be removed from the board and placed in the EPROM programmer for programming. The EEPROM approach avoids this labor intensive and annoying procedure by using another mechanism to inject or remove charges from the floating gate viz. tunneling. procedure.

Module 4

Design Capture Tool

Computer aided design (CAD) tools are essential for timely development of integrated circuits.Although CAD tools cannot replace the creative and inventive parts of the design activities,the majority of time consuming and computation intensive mechanistic parts of the design can be executed by using CAD tools.The CAD technology for VLSI chip design can be categorized into the following areas:

a)High level synthesis

b)Logic synthesis

c)circuit optimization

d)Layout

e)simulation

f)Design rule checking

g)Formal verification

Synthesis tools

The high level synthesis tools using hardware description languages(HDLs) such as VHDL or Verilog address the automation of the design phase in the top level of the design hierarchy,With an accurate estimation of lower level design features such as chip area and signal delay it can very effectively determine the types and quantities of modules to be included in the chip design.

Layout tools

The tools for circuit optimization are concerned with transistor sizing for minimization of delays and with process variation ,noise and reliability hazards.The layout CAD tools include floorplanning,place and route and module generation.

Simulation and verification tools

The simulation category ,which is the most mature area of VLSI CAD,includes many tools ranging from circuit level simulation,timimg level simulation,logic level simulation and behavirol simulation.

Hardware Description Language: VHDL

What is VHDL?

- A hardware description language that can be used to model a digital system.

• VHDL = VHSIC (Very High Speed Integrated Circuit) Hardware Description Language

• Can describe:

– behavior

– structure, and

– timing of a logic circuit.

Hardware Modelling in VHDL

VHDL is NOT a programming language like C or Java. It is used to model the physical hardware used in digital systems.Therefore you must always think about the hardware you wish to implement when designing systems using VHDL.

Data Objects

A data object holds a value of a specified type. The data objects that can be synthesized directly in hardware are:

1. **Signal**: Represents a physical wire in a circuit. It holds a list of values which includes its current value and a set of possible future values.

2. **Variable**: Used to hold results of computations. It does not necessarily represent a wire in a circuit.

3. **Constant**: Contains a value that cannot be changed. It is set before the beginning of a simulation.

Predefined Data Types

• Standard logic type:	STD_LOGIC, STD_LOGIC_VECTOR (Can hold 0, 1, Z, and —.)
• Bit type:	BIT, BIT_VECTOR
• Integer type:	INTEGER
• Floating–point type:	REAL
• Physical type:	TIME
• Enumeration type:	BOOLEAN, CHARACTER

- To use the STD_LOGIC and STD_LOGIC_VECTOR types, the std_logic_1164 package must be included in the VHDL design file.

- We can define our own data types. This is especially useful when designing finite–state machines (FSMs).

- An object declaration is used to declare an object, it type, its class, and optionally to assign it a value

Object Declaration Examples

- Signal declarations:
  SIGNAL sresetn : STD_LOGIC;
  SIGNAL address : STD_LOGIC_VECTOR(7 downto 0);
- Variable declarations:
  VARIABLE index : INTEGER range 0 to 99 :=20;
  VARIABLE memory : BIT_MATRIX(0 to 7, 0 to 1023);
- Constant declarations:
  CONSTANT cycle_time : TIME := 100 ns;
  CONSTANT cst : UNSIGNED(3 downto 0);

Operators

Operator Class	Operator
Miscellaneous	**, ABS, NOT
Multiplication	*, /, MOD, REM
Unary Arithmetic (Sign)	+, −
Addition	+, −, &
Shift/Rotate	sll, srl, sla, sra, rol, ror
Relational	=, /=, <, <=, >, >=
Logical	and, or, nand, nor, xor, xnor

- The individual operators in each class have the same precedence.

VHDL Design Entity

Entity Declaration

− Specifies the interface of entity to the outside world.

− Has a name.

− Includes PORT statement which specifies the entity's input and output signals (ports)

− Ports can have different modes: IN, OUT, INOUT, BUFFER

Architecture

− Provides circuit details for an entity

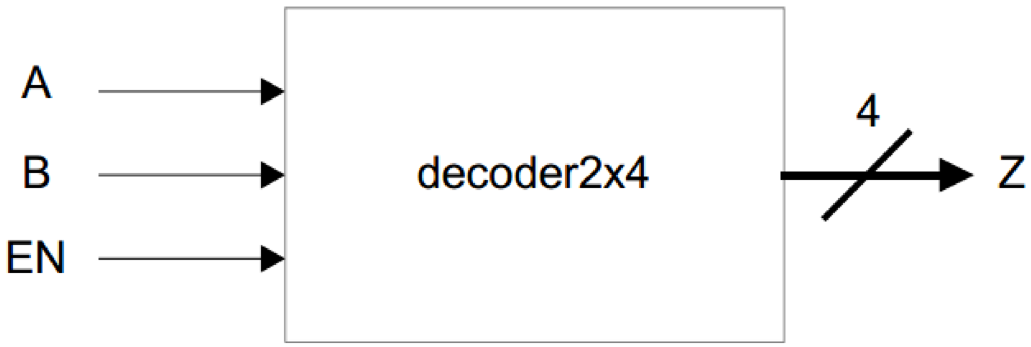
General form:
ARCHITECTURE arch_name OF entity_name IS
        Signal declarations

```
          Constant declarations
          Type declarations
          Component declarations
BEGIN
          Component instantiations
          Concurrent assignment statements
          Process statements
END arch_name
```

Concurrent Assignment Statements

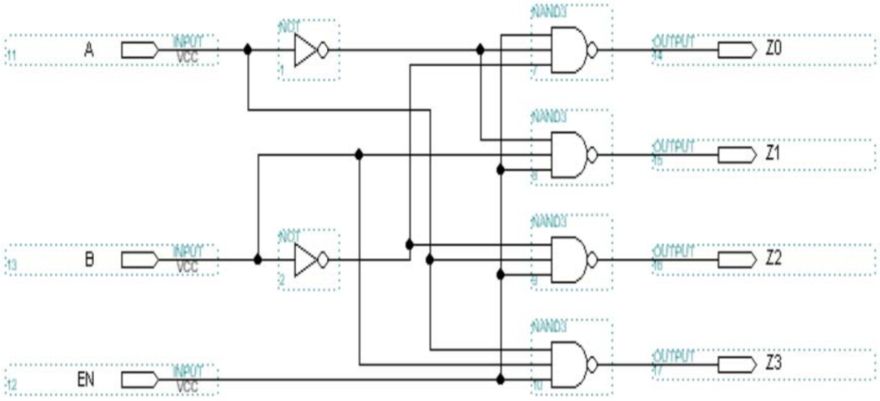
- A concurrent assignment statement is used to assign a value to a signal in an architecture body.
- Used to model combinational circuits.
- The order in which these statements occur does not affect the meaning of the code.

2x4 Decoder Example



```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY decoder2x4 IS
PORT (A, B, EN : IN STD_LOGIC;
Z : OUT STD_LOGIC_VECTOR(3 downto 0));
END decoder2x4;
—this is a comment
```

Architecture Body of 2x4 Decoder



```
ARCHITECTURE dec_df OF decoder2x4 IS
SIGNAL ABAR, BBAR : STD_LOGIC;
BEGIN
——Order of concurrent signal assignment
statements is not important.
Z(3) <= not (A and B and EN);
Z(0) <= not (ABAR and BBAR and EN);
BBAR <= not B;
Z(2) <= not (A and BBAR and EN);
ABAR <= not A;
Z(1) <= not (ABAR and B and EN);
END dec_df;
```

Sequential Assignment Statements

- Sequential assignment statements assign values to signals and variables. The order in which these statements appear can affect the meaning of the code.
- Can be used to model combinational circuits and sequential circuits.
- Require use of the PROCESS statement.
- Include three variants: IF statements, CASE statements, and LOOP statements. 2x4 Decoder Revisited

IF and CASE Statements

- IF and CASE statements are used to model multiplexers, decoders, encoders, and comparators.
- Can only be used in a PROCESS.

Modelling a 4−1 Multiplexer

Using an IF statement:

```
PROCESS (Sel, A, B, C, D)
```

```
BEGIN
IF (Sel = "00") THEN
Y <= A;
ELSIF (Sel = "01")
THEN
Y <= B;
ELSIF (Sel = "10")
THEN
Y <= C;
ELSE
Y <= D;
END IF;
END PROCESS;
```

Using a CASE statement:

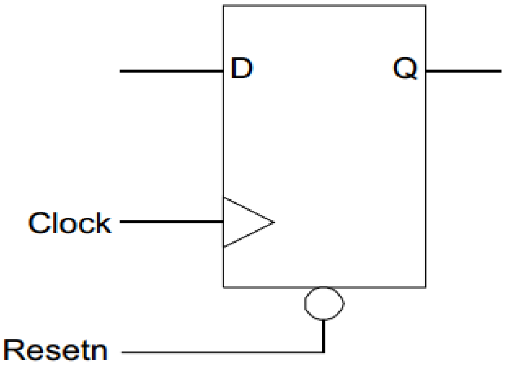
```
PROCESS (Sel, A, B, C, D)
BEGIN
CASE Sel IS
WHEN "00" => Y <= A;
WHEN "01" => Y <= B;
WHEN "10" => Y <= C;
WHEN "11" => Y <= D;
WHEN OTHERS =>Y
<=A;
END CASE;
END PROCESS;
```

- Can also model multiplexers with WHEN/ELSE clause and WITH/SELECT clause. These can only be used outside of a PROCESS.

Behavioural vs. Structural Modelling

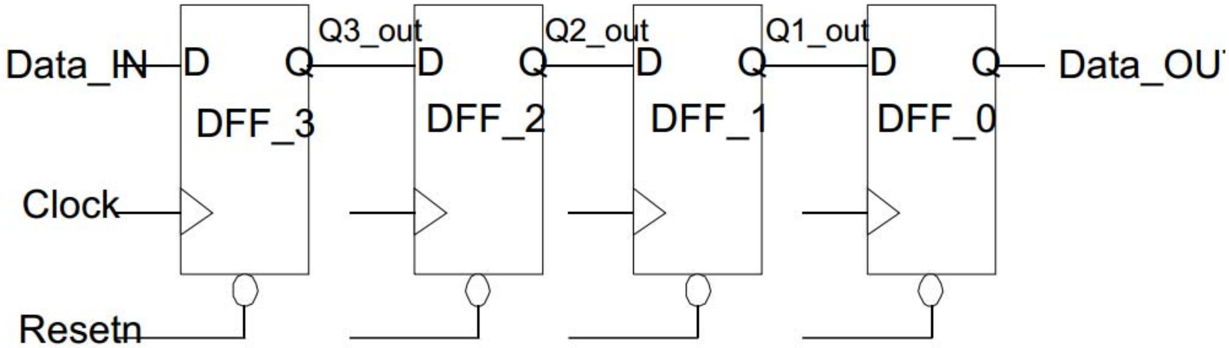
- With VHDL, we can describe the behaviour of simple circuit building blocks and then use these to build up the structure of a more complex circuit.
- Behavioural modelling is useful because it allows the designer to build a logic circuit without having to worry about the low−level details.
- Structural modelling is useful because it tells the synthesis tools exactly how to construct a desired circuit.

Behavioural Model of a D Flip−Flop



```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY my_dff IS
PORT ( D : IN  STD_LOGIC;
Q : OUT STD_LOGIC;
Clock : IN  STD_LOGIC;
Resetn: IN STD_LOGIC);
END my_dff;
ARCHITECTURE Behaviour OF my_dff IS
BEGIN
PROCESS(Clock, Resetn)
BEGIN
IF Resetn=’0’ THEN
Q <= ’0’;
ELSIF (Clock’EVENT AND Clock=’1’) THEN
Q <= D;
END IF;
END PROCESS;
END Behaviour;
```

Structural Model of a 4−Bit Shift Register



```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
```

```
ENTITY shift_reg_struct IS
PORT (Data_IN : IN STD_LOGIC;
Data_OUT: OUT STD_LOGIC;
CLK, RESN : IN STD_LOGIC);
END shift_reg_struct;
ARCHITECTURE Structure OF shift_reg_struct IS
COMPONENT my_dff
PORT ( D : IN  STD_LOGIC;
Q : OUT STD_LOGIC;
Clock : IN  STD_LOGIC;
Resetn: IN STD_LOGIC);
END COMPONENT;
SIGNAL Q3_out, Q2_out, Q1_out : STD_LOGIC;
BEGIN
DFF_3 : my_dff PORT MAP (Data_IN, Q3_out, CLK, RESN);
DFF_2 : my_dff PORT MAP (Q3_out, Q2_out, CLK, RESN);
DFF_1 : my_dff PORT MAP (D=>Q2_out, Q=>Q1_out,
Clock=>CLK, Resetn=>RESN);
DFF_0 : my_dff PORT MAP (D=>Q1_out, Q=>Data_OUT,
Clock=>CLK, Resetn=>RESN);
END Structure;
```

Testing and Verification:

Defects, Errors and Faults

Models bridge the gap between *physical reality* and a *mathematical abstraction* and allow for development of analytical tools.

Definitions:

- **Defect**: An untended difference between the implemented hardware and its intended function.
- **Error**: A wrong *output signal* produced by a defective system.
- **Fault** is a *logic level abstraction* of a **physical defect**. Used to describe the change in the logic function caused by the defect. *Fault abstractions* reduce the number of conditions that must be considered in deriving tests.

A *collection of faults*, all of which are based on the same set of assumptions concerning the nature of defects, is called a **fault model**.

For example:

- **Defect**: *A* shorted to GND.
- **Fault**: *A* Stuck-at logic 0.
- **Error**: *A=1*, *B=1 => C=1*

Correct output is *C=0*.

Note that the **error** is *not* permanent

For example, no error occurs if at least one input is 0.

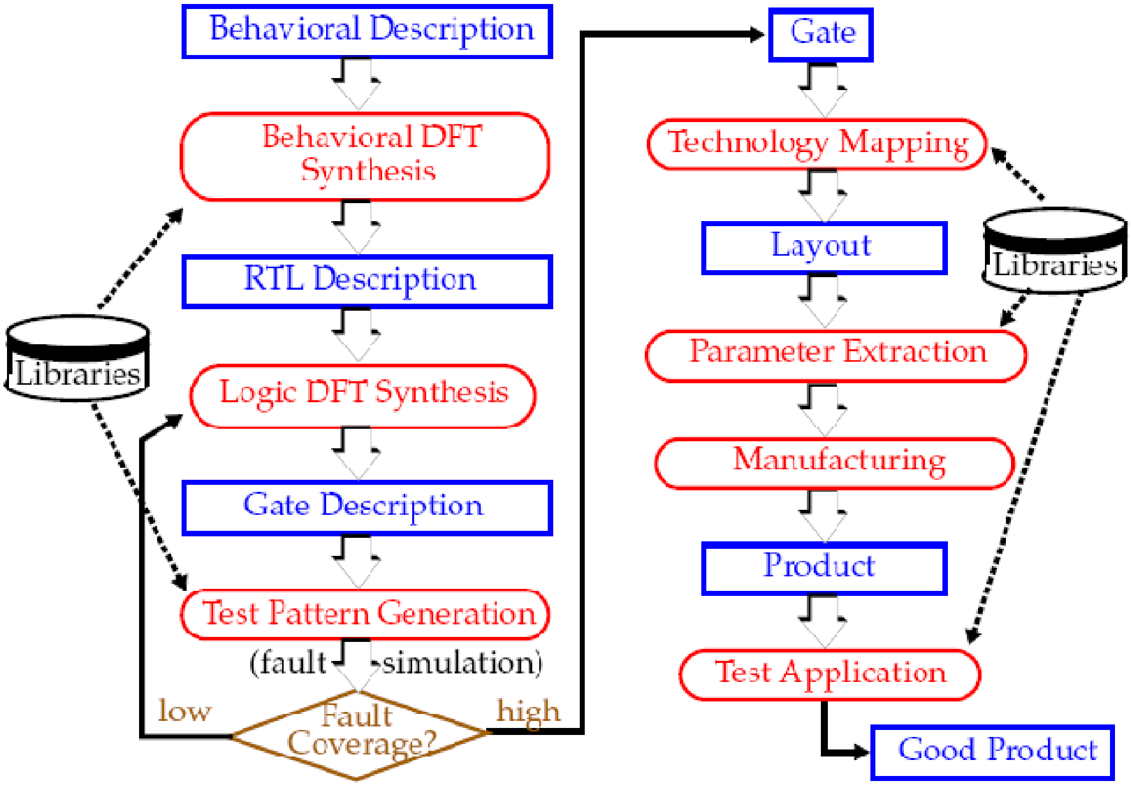
Functional Vs. Structural Testing

Design verification may need to use exhaustive functional tests. Fortunately, for hardware testing, we **can assume** the function is correct. The focus on **structure** makes it possible to develop algorithms that are independent of the design. The algorithms are based on **fault models**.

Fault models can be formulated at the various levels of design abstraction.

- *Behavioral level*: Faults may not have any obvious correlation to defects.
- *RTL and Logic level*: Stuck-at faults most popular, followed by bridging and delay fault models.
- *Transistor (switch) level*: Technology-dependent faults.

Design Levels from Testability Perspective:



Fault Models

The text describes these and other fault models:

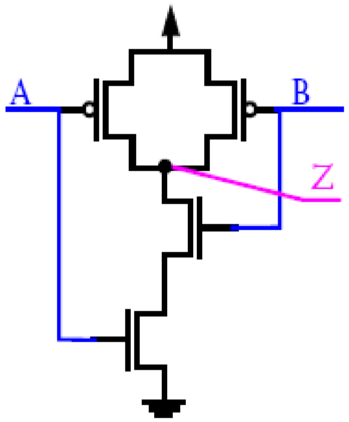
- Bridging fault

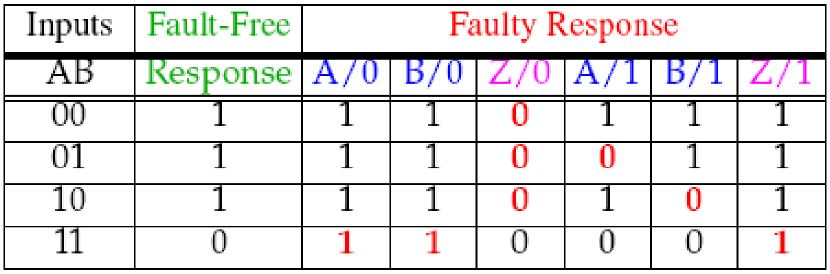
- Defect-oriented fault (e.g. bridging, stuck-open, IDDQ).
- Delay fault (transition, gate-delay, line-delay, segment-delay, path-delay).
- Intermittent fault
- Logical fault (often stuck-at)
- Memory fault (single cell SA0/1, pattern sensitive, cell coupling faults).
- Non-classical fault (stuck-open or stuck-on, transistor faults for CMOS).
- Pattern sensitive fault
- Pin fault (SA faults on the signal pins of all modules in the circuit).
- Redundant fault
- Stuck-at fault

Single stuck-at faults (SSF)

Assumes defects cause the signal net or line to remain at a fixed voltage level. Model includes **stuck-at-0** (SA0) or **stuck-at-1** (SA1) faults and assumes only one fault exists.

For example, how many *SSF* faults can occur on an *n-input* NAND gate?





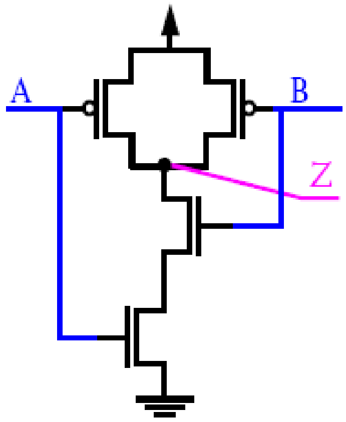
Inputs	Fault-Free	Faulty Response					
AB	Response	A/0	B/0	Z/0	A/1	B/1	Z/1
00	1	1	1	0	1	1	1
01	1	1	1	0	0	1	1
10	1	1	1	0	1	0	1
11	0	1	1	0	0	0	1

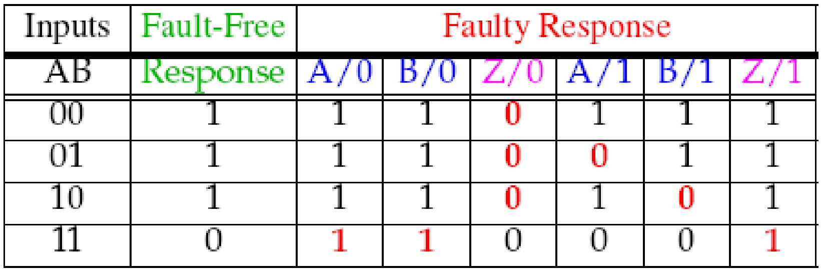
What fault(s) does the pattern *AB* = *01* detect?

What is the *minimum* number of tests needed to "detect" all of them?

What are the tests?

A **dominant** input value is defined as the value that *determines* the state of the output independent of the other values of the inputs.





Inputs	Fault-Free	Faulty Response					
AB	Response	A/0	B/0	Z/0	A/1	B/1	Z/1
00	1	1	1	0	1	1	1
01	1	1	1	0	0	1	1
10	1	1	1	0	1	0	1
11	0	1	1	0	0	0	1

What is the dominant input value for the NAND?

How many tests do you need to diagnosis the fault (hint: deduction can be used to distinguish)?

Can you distinguish between all of the faults? An *n*-line circuit can have at most *2n* SSF faults.

This number can be further reduced through **fault collapsing**.

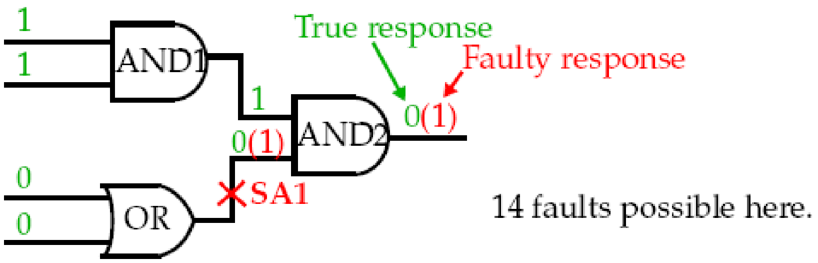
Fault detection requires:

- A test *t* activates or provokes the fault *f*.
- *t* propagates the error to *observation point* (primary output (PO)/scan latch).

A line that changes with *f* is said to be **sensitized** to the fault site.

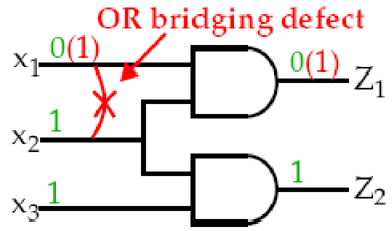
Fault propagation requires *off-path* inputs be set to *non-dominant* values.

01, *10*, and *11*do **not** provoke the fault



Let *Z(t)* represent the response of a circuit *N* under input vector *t*.

Fault *f* transforms the circuit to *Nf* and its response to *Zf(t)*.



$Z_1 = x_1x_2 \longrightarrow Z_{1f} = x_1 + x_2$

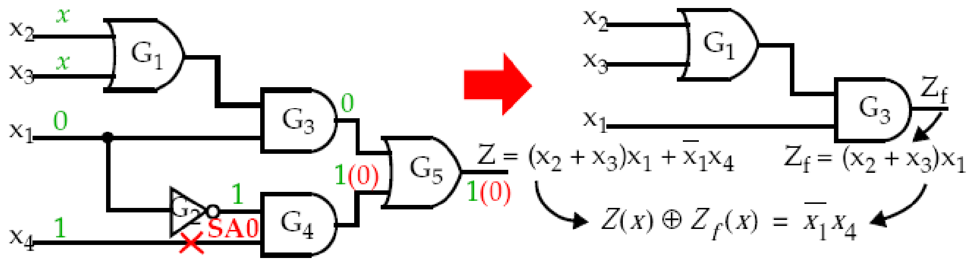
$Z_2 = x_2x_3 \longrightarrow Z_{2f} = (x_1 + x_2)x_3$

011 defects *f* because $Z(011) = 01$ and $Z_f(011) = 11$.

The set of all tests *x* that detect *f* is given by

$$Z(x) \oplus Z_f(x) = 1$$

In this example, any test in which *x1* = 0 and *x4* = 1 is a test for *f*.



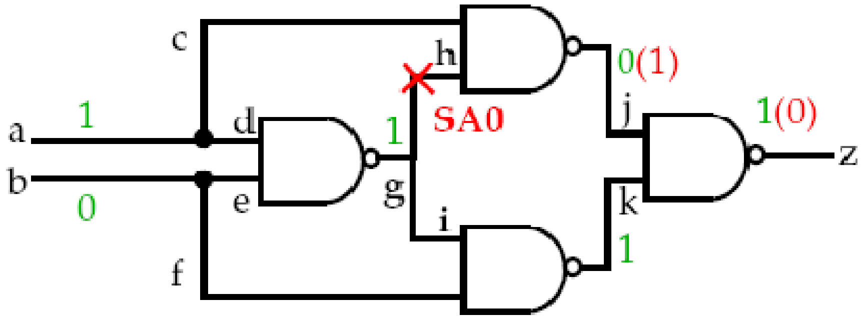
$Z = (x_2 + x_3)x_1 + \overline{x}_1x_4$ $\quad Z_f = (x_2 + x_3)x_1$

$$Z(x) \oplus Z_f(x) = \overline{x}_1x_4$$

The expression (x1compliment)x4 represents 4 tests (*0001*, *0011*, *0101*, *0111*).

Note that faults on *fanout stems* and *fanout branches* are **not** the same. This will be important later when we develop a method to *collapse* faults that are equivalent.

In this example, we assume line *g*, *h* and *i* carry the same signal value.

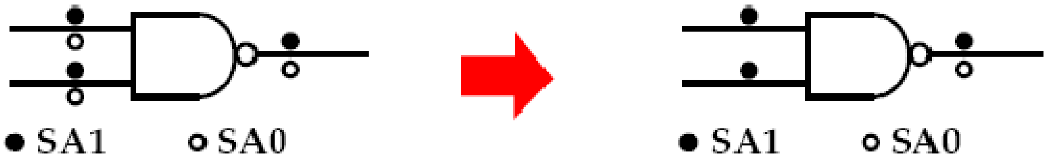


Input *ab* = (*10*) activates *h SA0* and is detectable at *z*. This input also activates *SA0* faults on *g* and *i* .However, *i SA0* is **not** detectable since it is blocked by *f* = 0.

We consider faults *h* and *i* as different because the tests that detect *h* and *i* are not the same -- fault *g* is also different because both vectors detect it.

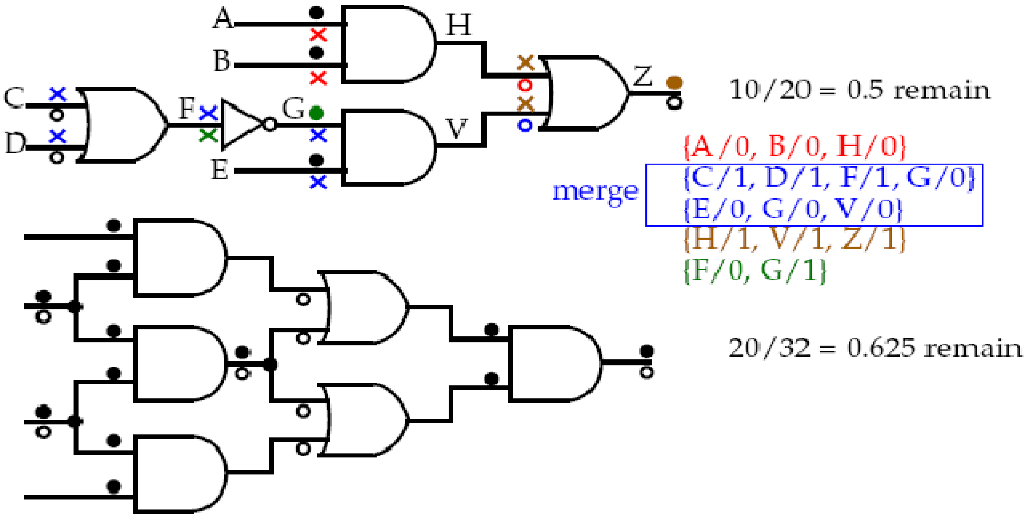
Fault Equivalence

Two faults *f* and *g* are considered **functionally equivalent** if Zf(x) = Zg(x).
There is no test that can distinguish between *f* and *g*. i.e., *all tests* that detect *f* also detect *g*. For large circuits, determining this is computationally intensive. However, equivalence can be determined for simple gates and applied to large circuits.
Any *n*-input gate has 2(n+1) SA faults. For the NAND gate, the *SA0* on the inputs are equivalent to *SA1* on the output and all three are detected by the same test pattern *AB*=(*11*). For any *n*-input primitive gate with *n*>1, only ***n+2*** single SA faults need to be considered.



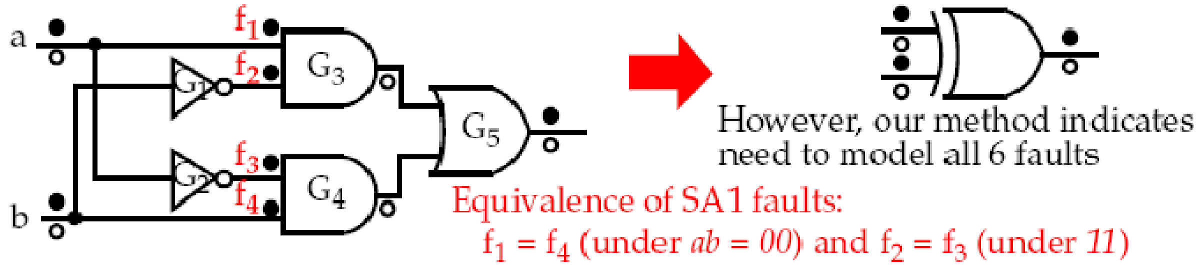
Fault Equivalence

Equivalence fault collapsing is performed from inputs to output.



Reduction is between 50-60% and is larger, in general, for fanout free circuits.

Functional equivalence partitions the set of all faults into *functional equivalence classes*, from each of which only one fault needs to be considered. This property is useful for test generation programs. Fault equivalence reduces the size of the fault list. Fault equivalence is important for fault location analysis as well. A *complete location test* set can diagnose a fault to within a functional equivalence class. This represents the *maximal diagnostic* resolution achievable. Fault collapsing using this simple method cannot find all equivalencies.

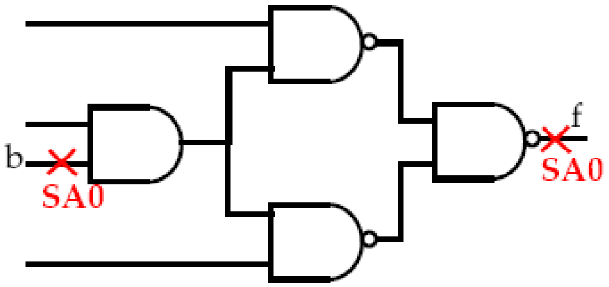


However, our method indicates need to model all 6 faults

Equivalence of SA1 faults:
$f_1 = f_4$ (under *ab* = *00*) and $f_2 = f_3$ (under *11*)

Therefore, the equivalence classes derived are not maximal.

Our method will not identify faults *b* SA0 and *f* SA0 as equivalent

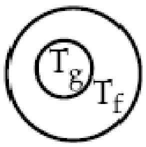


Algorithms are available to solve the more general case of **functional equivalence** based on: **SA0** *Z f* (*x*) Å *Zg*(*x*) = 0

The benefit of identifying these additional fault equivalences is usually not worth the effort, however.

See text for ISCAS’85 benchmark circuit results. If fault detection is the objective (not diagnosis), then **fault dominance** can be used to further reduce the fault list.

A fault *f* dominates another fault *g* if the set of all tests that detect *g*, Tg, is a subset of the test set of Tf.



$T_g \subseteq T_f$

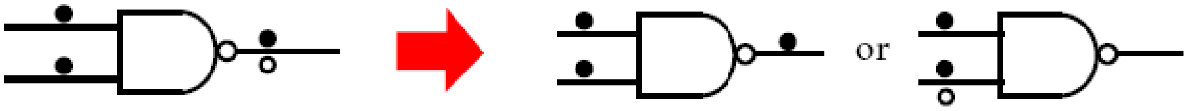
Def: *f* dominates *g* iff
f and *g* are functionally equivalent
under T_g.

Therefore, any test that detects *g* will also detect *f*.

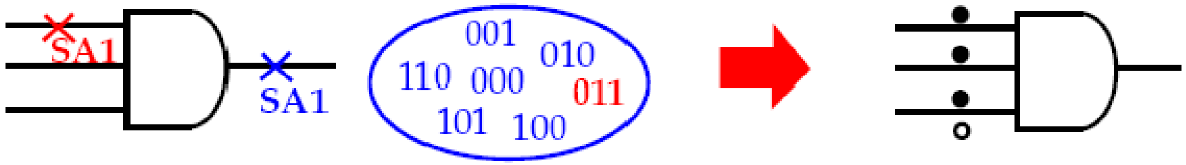
Since *g* implies *f*, it is sufficient to include *g* in the fault list.

For example, the *SA1* test (*g*) for input *A* of the NAND gate also detects *SA0* (*f*) on the output (the same is true for input *B*.)

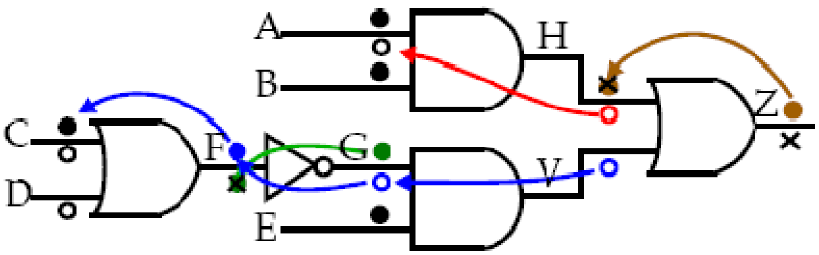
Therefore, *Z-SA0* can be dropped from the list.



For larger input gates.



Dominance fault collapsing is performed from outputs to inputs.



7/20 = 0.3 remain

Here, the *x* indicates the collapsing of the fault via dominance.

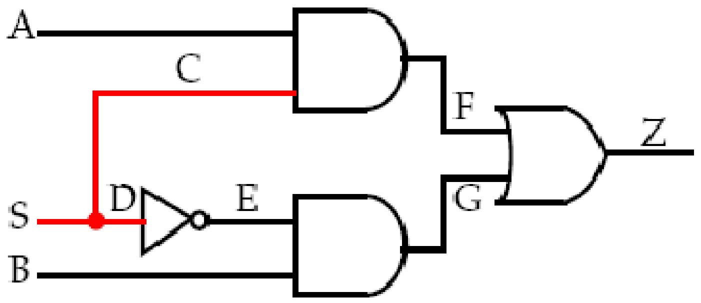
We can also, optionally, choose to move the output fault preserved in the equivalence fault collapsing to an arbitrary input.

One such fault list may be: {*A/0*, *A/1*, *B/1*, *C/0*, *C/1*, *D/0*, *E/1*}

Or another may be: {*B/0*, *A/1*, *B/1*, *C/0*, *D/1*, *D/0*, *E/1*}

Note that these lists contains **only** faults on the PIs.Any test set that detects all SSFs on the PIs of a **fanout free** combinational circuit C detects all SSFs in C.

What happens in the presence of fanout?



Eliminating S, reduces the circuit to a fanout free circuit.

SAB = (*010*) detects C SA1, *SAB* = (*001*) detects D SA1 => both detect S SA1.

SAB = (*110*) detects C SA0, *SAB* = (*101*) detects D SA0 => both detect S SA0.

Therefore, SA faults on *stem* **dominate** SA faults on the *fanout branches*.

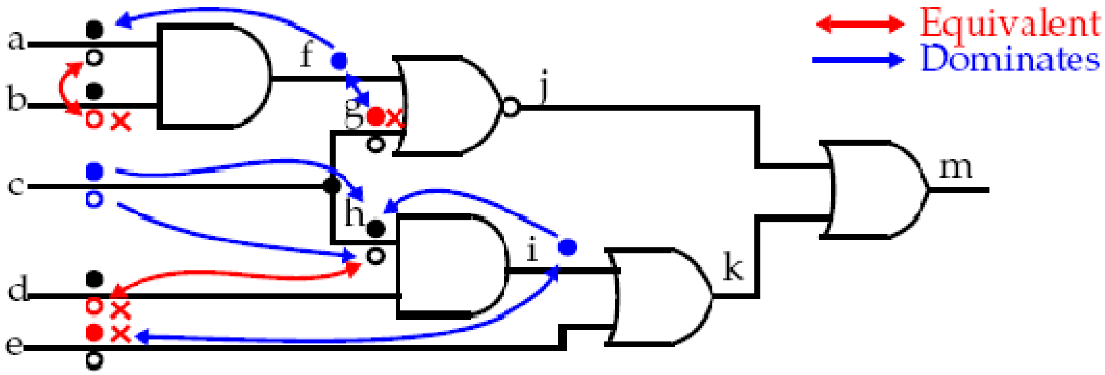
More generally, any test set that detects all SSF on the PIs and fanout branches of C detects all SSF in C.

The PIs and fanout branches are called **checkpoints**. Therefore, it is sufficient to target faults only at the *checkpoints*.

Equivalence and dominance relations can then be used to further collapse the list of faults.

For example, this circuit has 24 SSFs.

But it only has 14 checkpoint faults (the 5 PIs) + *g* and *h*.



This leaves 8 faults from the original list of 14 checkpoint faults.

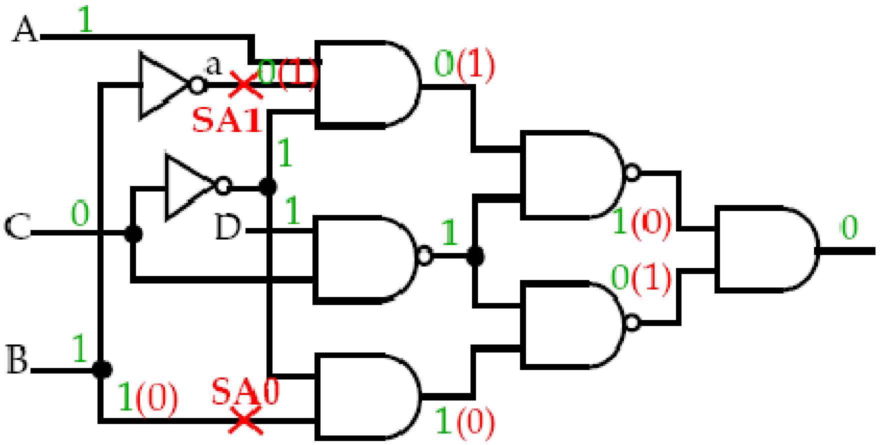
Undetectable Faults

A fault is **detectable** if there exists a test *t* that defects *f*. It is **undetectable**, if **no** test simultaneously activates *f* and creates a sensitized path to a PO.

Undetectable faults may appear to be harmless.

However, a *complete* test set may not be sufficient if one is present in a chip with *multiple faults*.

The fault *b* SA0 is **no longer detectable** by the test *t* = (*1101*) if the **undetectable fault** *a* SA1 is present.

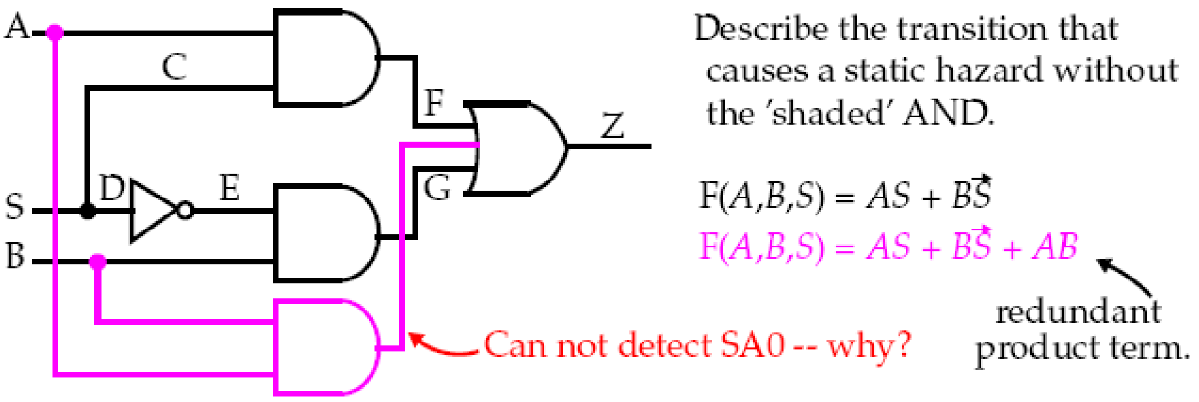


If test *t* is the only test in the complete test set *T* that detects *b* SA0, then *T* is no longer complete in the presence of *a* SA1.

Redundancy: A combinational circuit that contains an *undetectable fault* is said to be redundant.

Redundant faults cause ATPG algorithms to exhibit worst-case behavior.

Redundancy is not always undesirable.



Describe the transition that causes a static hazard without the 'shaded' AND.

$F(A,B,S) = AS + B\vec{S}$

$F(A,B,S) = AS + B\vec{S} + AB$

redundant product term.

Design stratergies for testing

Design for Testability Techniques Design for testability (DFT) refers to those design techniques that make the task of subsequent testing easier. There is definitely no single methodology that solves all embedded system-testing problems. There also is no single DFT technique, which is effective for all kinds of circuits. DFT techniques can largely be divided into two categories, i.e., ad hoc techniques and structured (systematic) techniques. DFT methods for digital circuits:

1) Ad-hoc methods

2) Structured methods:

- Scan

- Partial Scan

- Built-in self-test

- Boundary scan Ad-hoc DFT methods

Ad Hoc Testable Design Techniques

One way to increase the testability is to make nodes more accessible at some cost by physically inserting more access circuits to the original design. Listed below are some of the ad hoc testable design techniques.

Partition-and-Mux Technique

Since the sequence of many serial gates, functional blocks, or large circuits are difficult to test, such circuits can be partitioned and multiplexors (muxes) can be inserted such that some of the primary inputs can be fed to partitioned parts through multiplexers with accessible control signals. With this design technique, the number of accessible nodes can be increased and the number of test patterns can be reduced. A case in point would be the 32-bit counter. Dividing this counter into two 16-bit parts would reduce the testing time in principle by a factor of 215. However, circuit partitioning and addition of multiplexers may increase the chip area and circuit delay. This practice is not unique and is similar to the divide-and-conquer approach to large, complex problems.

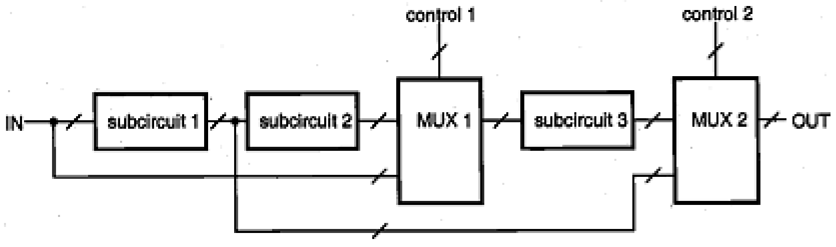


Figure 51: Partition-and-mux method for large circuits.

Initialize Sequential Circuit

When the sequential circuit is powered up, its initial state can be a random, unknown state. In this case, it is not possible to start the test sequence correctly. The state of a sequential circuit can be brought to a known state through initialization. In many designs, the initialization can be easily done by connecting asynchronous preset or clear-input signals from primary or controllable inputs to flip-flops or latches.

Disable Internal Oscillators and Clocks

To avoid synchronization problems during testing, internal oscillators and clocks should be disabled. For example, rather than connecting the circuit directly to the on-chip oscillator, the clock signal can be ORed with a disabling signal followed by an insertion of a testing signal.

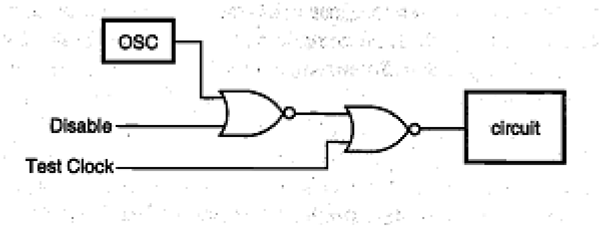


Figure 51: Avoid synchronization problems-via disabling of the oscillator.

Avoid Asynchronous Logic and Redundant Logic

The enhancement of testability requires serious tradeoffs. The speed of an asynchronous logic circuit can be faster than that of the synchronous logic circuit counterpart. However, the design and test of an asynchronous logic circuit are more difficult than for a synchronous logic circuit, and its state transition times are difficult to predict. Also, the operation of an asynchronous logic circuit is sensitive to input test patterns, often causing race problems and hazards of having momentary signal values opposite to the expected values. Sometimes, designed-in logic redundancy is used to mask a static hazard condition for reliability. However, the redundant node cannot be observed since the primary output value cannot be made dependent on the value of the redundant node. Hence, certain faults on the redundant node cannot be tested or detected. The bottom NAND2 gate is redundant and the stuck-at- fault on its output line cannot be detected. If a fault is undetectable, the associated line or gate can be removed without changing the logic function.

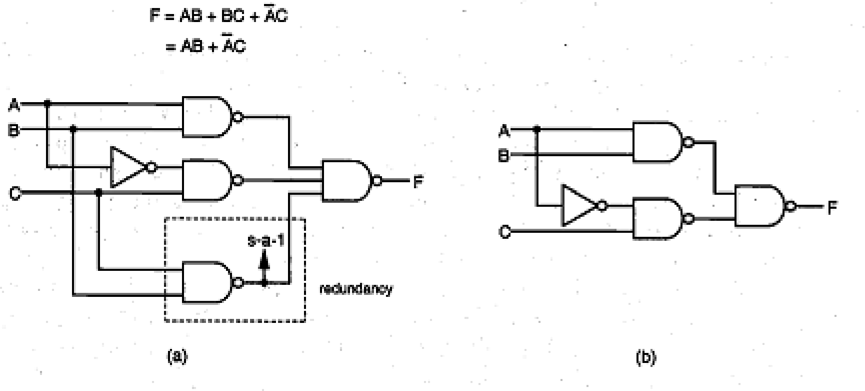


Figure 52: (a) A redundant logic gate example. (b) Equivalent gate with redundancy removed.

Avoid Delay-Dependent Logic

Chains of inverters can be used to design in delay times and use AND operation of their outputs along with inputs to generate pulses.

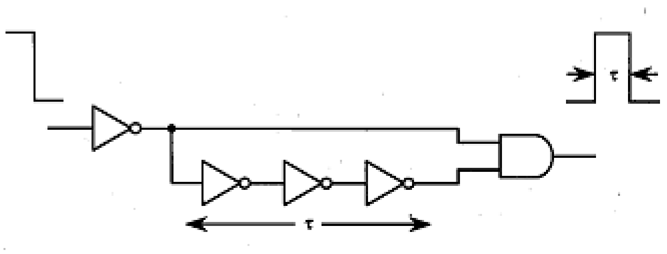


Figure 53: A pulse-generation circuit using a delay chain of three inverters.

Most automatic test pattern generation (ATPG) programs do not include logic delays to minimize the complexity of the program. As a result, such delay-dependent logic is viewed as redundant combinational logic, and the output of the reconvergent gate is always set to logic 0, which is not correct. Thus, the use of delay-dependent logic should be avoided in design for testability

Scan-Based Technique

As discussed earlier, the controllability and observability can be enhanced by providing more accessible logic nodes with use of additional primary input lines and multiplexors. However, the use of additional I/O pins can be costly not only for chip fabrication but also for packaging. A popular alternative is to use scan registers with both shift and parallel load capabilities. The scan design technique is a structured approach to design sequential circuits for testability. The storage cells in registers are used as observation points, control points, or both. By using the scan design techniques, the testing of a sequential circuit is reduced to the problem of testing a combinational circuit.

In general, a sequential circuit consists of a combinational circuit and some storage elements. In the scan-based design, the storage elements are connected to form a long serial shift register, the so-called scan path, by using multiplexors and a mode (test/ normal) control signal.

In the test mode, the scan-in signal is clocked into the scan path, and the output of the last stage latch is scanned out. In the normal mode, the scan-in path is disabled and the circuit functions as a sequential circuit. The testing sequence is as follows:

Step 1: Set the mode to test and, let latches accept data from scan-in input,

Step 2: Verify the scan path by shifting in and out the test data.

Step 3: Scan in (shift in) the desired state vector into the shift register.

Step 4: Apply the test pattern to the prim ary input pins. I : ;

Step 5: Set the mode to normal and observe the primary outputs of the circuit after

sufficient time for propagation.,

Step 6: Assert the circuit clock, for one machine cycle to capture the outputs of the

combinational logic into the registers.

Step 7: Return to test mode; scan out the contents of the registers, and at the same time scan in the next pattern.

Step 8: Repeat steps 3-7 until all test patterns are applied.

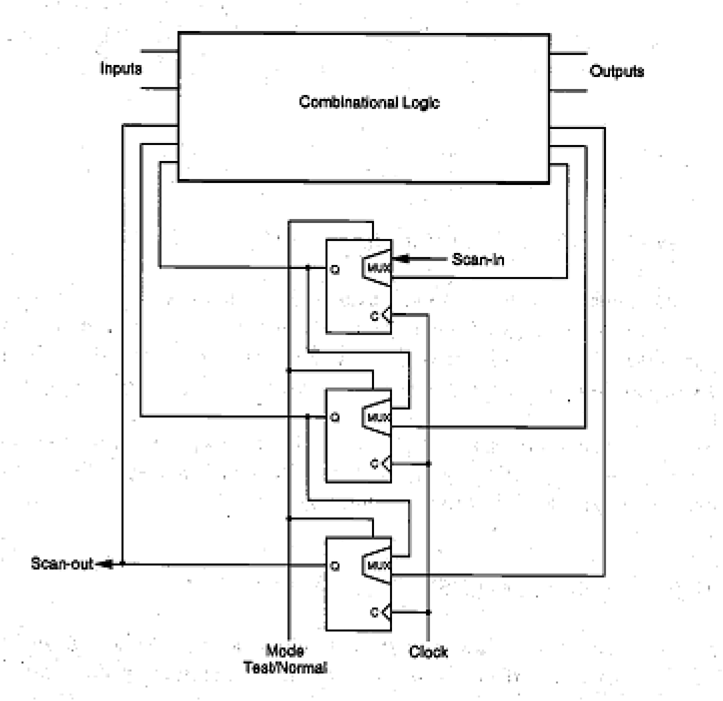


Figure 54: The general structure of scan-based design

The storage cells in scan design can be implemented using edge-triggered D flipflops, master-slave flip-flops, or level-sensitive latches controlled by complementary clock signals to ensure race-free operation. D flip-flop. In large high-speed circuits, optimizing a single clock signal for skews, etc., both for normal operation and for shift operation, is difficult. To overcome this difficulty, two separate clocks, one for normal operation and one for shift operation, are used. Since the shift operation does not have to be performed at the target speed, its clock is much less constrained.

An important approach among scan-based designs is the level sensitive scan design (LSSD), which incorporates both the level sensitivity and the scan path approach using shift registers. The level sensitivity is to ensure that the sequential circuit response is independent of the transient characteristics of the circuit, such as the component and wire delays. Thus, LSSD removes hazards and races. Its ATPG is also simplified since tests have to be generated only for the combinational part of the circuit.

The boundary scan test method is also used for testing printed circuit boards (PCBs) and multichip modules (MCMs) carrying multiple chips. Shift registers are placed in each chip close to I/O pins in order to form a chain around the board for testing. With successful implementation of the boundary scan method, a simpler tester can be used for PCB testing.

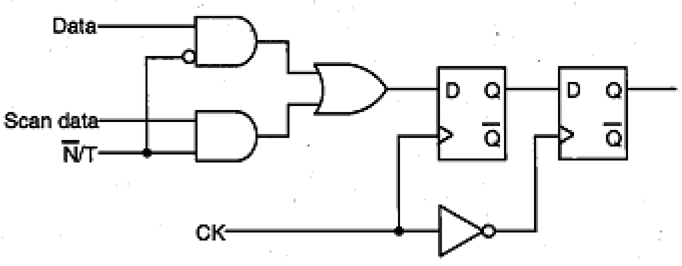


Figure 55: Scan-based design of an edge-triggered D flip-flop

On the negative side, scan design uses more complex latches, flip-flops, I/O pins, and interconnect wires and, thus, requires more chip area. The testing time per test pattern is also increased due to shift time in long registers.

Built-in Self-Test (BIST)

- Capability of a circuit to test itself
- On-line: – Concurrent : simultaneous with normal operation – Nonconcurrent : idle during normal operation
- Off-line: – Functional : diagnostic S/W or F/W – Structural : LFSR-based
- We deal primarily with structural off-line testing

Basic Architecture of BIST

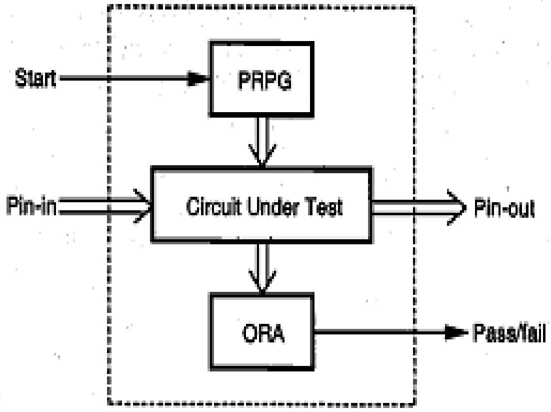


Fig: A procedure for BIST.

- PRPG: Pseudo random pattern generator
- ORA :Output response analyzer

Built-in Self Testing

- Test pattern generation
  - Exhaustive
  - Pseudoexhaustive
  - Pseudorandom
- Test response compression
  - One's count

– Transition count
– Parity checking
– Syndrome checking
– Signature analysis

]

Test Pattern Generation for BIST

- Exhaustive testing
- Pseudorandom testing
  - Weighted and Adaptive TG
- Pseudoexhaustive testing
  - Syndrome driver counter
  - Constant-weight counter
  - Combined LFSR and shift register
  - Combined LFSR and XOR
  - Cyclic LFSR

Pseudo Random Pattern Generator

To test the circuit, test patterns first have to be generated either by using a pseudo random pattern generator, a weighted test generator, an adaptive test generator, or other means.
A pseudo random test generator circuit can use an LFSR.

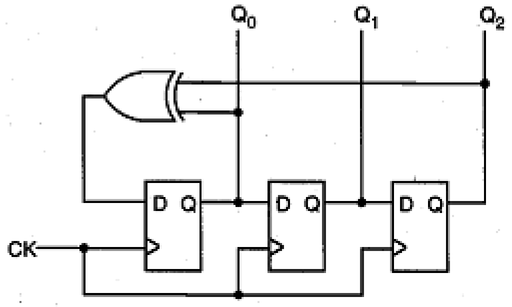


Figure 56: A pseudo-random sequence generator using LFSR

Linear Feedback Shift Register (LFSR) as an ORA

To reduce the chip area penalty, data compression schemes are used to compare the compacted test responses instead of the entire raw test data. One of the popular data compression schemes is the signature analysis, which is based on the concept of cyclic redundancy checking. It uses polynomial division, which divides the polynomial representation of the test output data by a characteristic polynomial and then finds the remainder as the signature. The signature is then compared with the expected signature to determine whether the device under test is faulty. It is known that compression can cause some loss of fault coverage. It is possible that the output of a faulty circuit can match the output of the fault-free circuit; thus, the fault can go undetected in the signature analysis. Such a phenomenon is called aliasing.

In its simplest form, the signature generator consists of a single-input linear feedback shift register (LFSR), all the latches are edge-triggered. In this case, the signature is the content of this register after the last input bit has been sampled. The input sequence $\{a_n)$ is represented by polynomial G(x) and the output sequence by Q(x). It can be shown that G(x) = Q(x) P(x) + R(x), where P(x) is the characteristic polynomial of LFSR and R(x) is the remainder, the degree of which is lower than that of P(x).

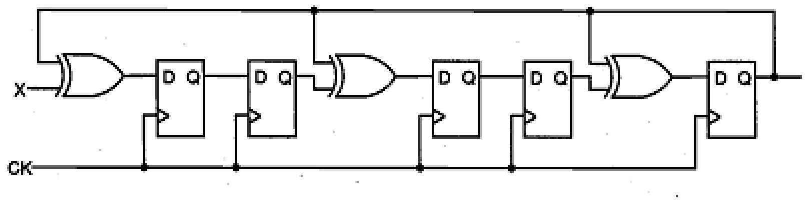


Figure 57: Polynomial division using LFSR for signature analysis.

The characteristic polynomial is

$$P(x)=1+x^2+x^4+x^5$$

For the 8-bit input sequence { 1 1 1 1 0 1 0 1, the corresponding input polynomial is

$$G(x)=x^7+x^6+x^5+x^4+x^2+1$$

and the remainder term becomes $R(x) = x^4 + x^2$, which corresponds to the register contents of {0 0 1 0 11}

Built-In Logic Block Observer

The built-in logic block observer (BILBO) register is a form of ORA which can be used in each cluster of partitioned registers. A basic BILBO circuit allows four different modes controlled by $C_0$and C_1 signals.

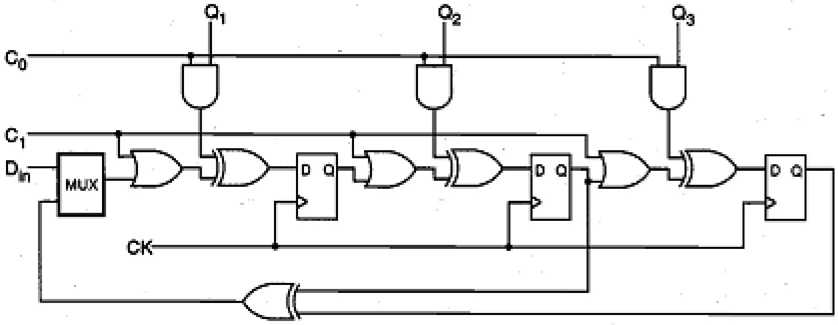


Figure 58: 3-bit built-in logic observer (BILBO) example.

C_0	C_1	Mode
0	0	linear shift
1	0	signature analysis

1	1	data(complemented) latch
0	1	reset

The BILBO operation allows monitoring of circuit operation through exclusive ORing into LFSR at multiple points, which corresponds to the signature analyzer with multiple inputs.

Packaging Technology

Proper packaging technology is critical to the success of the chip development.Package issues have to be taken into consideration in early stages of chip development.Ensure sufficient design margins to accommodate the parasitics of the package

Important packaging concerns:

-Hermetic seals to prevent the penetration of moisture

-Thermal conductivity

-Thermal expansion coefficient

-Pin density

-Parasitic inductance and capacitance

-particle protection

-Cost

Types of packaging technology

Classified by the method used to solder the package on the printed PCB

-Pin-through-hole (PTH)

-Surface-mounted technology (SMT)

Dual in-line packages (DIP)

Advantage of low cost.Not applicable for high-speed operations due to the inductance of the bond wires-Maximum pin count is typically limited to 64

Pin grid array (PGA) packages

-Offers a higher pin count (several houndreds)

-High thermal conductivity especially with a passive or active heat sink

-Requires large PCB area

-Cost is higher than DIP

Chip carrier packages (CCP)

-Leadless chip carrier:

-Chip mounted on PCB directly

-Supports higher pin count

-Problem with difference in thermal coefficient

-Leaded chip carrier:

Quad flat packages (QFP)

-Similar to leaded chip carrier with leads extending outward

Multi-chip modules (MCM)

-Used for very high performance in special applications

-Multiple chips are assembled on a common substrate in a single package

-A large number of critical interconnects among the chips are made within the package

Important features:

-Significant reduction in the overall system size

-Reduced package lead counts

-Faster operation allowed

-Higher implementation cos

GANDHI ACADEMY OF TECHNOLOGY AND ENGINEERING

GOLANTHARA, BERHAMPUR

LECTURE NOTES

ON

Wave Propagation & Broadband Communication

For 5th Semester

ELECTRONICS & TELECOMMUNICATION

(As per Syllabus prescribed by SCTE&VT, Odisha)

Prepared By

Mr. Manas Ranjan Biswal

(Lecturer in Electronics & Telecommunication Engineering)

Microwave

The signal deals with very small wave wavelength is called microwave signal, this implies signal has:

Wavelength (ƛ) =speed/frequency

With due increase in frequency the wavelength decrease and vice versa; we can say that wavelength is inversely proportional to frequency.

In communication system,it generally consist of three main components: Transmitter, Receiver and Channel.

There are two type mediums: transmission line & waveguide.

Transmission line used for small range frequencies. Waveguide used for large range frequencies.

Transmission Line (TL)

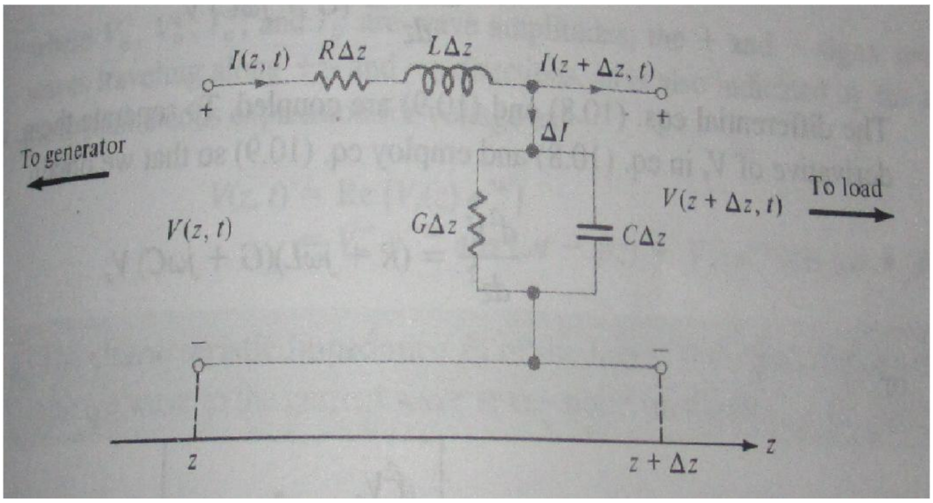
- The wave is bounded at low frequency in transmission line and hence called low pass filter.
- Transmission line mainly supports electromagnetic field.

Types of Transmission Line:

- Coaxial cable
- Parallel wire cable
- Microstrip line

Lumped Element Circuit Model of Transmission Line

- A current carrying conductor produce a magnetic field i.e., inductance which opposes the flow of current hence resistance 'R' is in series with inductance 'L'.
- Because of dielectric separation; there exist a capacitance and the loss in the dielectric medium give rise to conductance.
  R= Resistance of the conductor (Ω/m)
  L= Self Inductance of the conductor (μ/m)
  C= Capacitance across the conductor (F/m)
  G= Dielectric Loss between conductor (Ʊ/m)



Applying KVL in the loop;

$v(z, t) = R\,\Delta z\, I(z, t) + L\,\Delta z\,(d\,I(z, t)/dt) + v(z + \Delta z, t)$

$v(z, t) - v(z + \Delta z, t) = R\,\Delta z\, I(z, t) + L\,\Delta z\,(d\,I(z, t)/dt)$

$(v(z, t) - v(z + \Delta z, t))/\,\Delta z = R\, I(z, t) + L\,(d\,I(z, t)/dt)$

Taking limit;

$\lim_{\Delta z \to 0} \quad (v(z, t) - v(z + \Delta z, t))/\,\Delta z = \lim_{\Delta z \to 0} \quad R\, I(z, t) + L\,(d\,I(z, t)/dt)$

$-dv(z, t)/dz = R\, I(z, t) + L\, dI(z, t)/dtC$

$-dV/dz = RI + LdI/dt$ -------------①

Applying KCL at first node

$I(z, t) = I(z + \Delta z, t) + \Delta I$

$I(z, t) = \Delta I_1 + \Delta I_2 + I(z + \Delta z, t)$

$I(z, t) = G\,\Delta z\, V(z + \Delta z, t) + C\,\Delta z\,(d\,V(z + \Delta z, t)/dt) + I(z + \Delta z, t)$

$I(z, t) - I(z + \Delta z, t) = G\,\Delta z\, V(z + \Delta z, t) + C\,\Delta z\,(d\,V(z + \Delta z, t)/dt)$

$(I(z, t) - I(z + \Delta z, t))/\Delta z = G\, V(z + \Delta z, t) + C\,(d\,V(z + \Delta z, t)/dt)$

$\lim_{\Delta z \to 0} \quad (I(z, t) - I(z + \Delta z, t))/\Delta z = \lim_{\Delta z \to 0} \quad G\, V(z + \Delta z, t) + C\,(d\,V(z + \Delta z, t)/dt)$

$-d\,I(z, t)/dz = G\, V(z,t) + C\,(d\,V(z, t)/dt)$

$-d\,I/dz = G\,V + C\,d\,V/dt$ -------------②

In equation ① & ②

$V= V(z, t)= \mathrm{Re}\{V_s(z)e^{j\dot{\omega}t}\}$

$I= I(z, t)= \mathrm{Re}\{I_s(z)e^{j\dot{\omega}t}\}$

Putting V & I in equation ①

$-d\,V/dz= R\,I + L\,d\,I/dt$

$-d\,[\mathrm{Re}\{V_s(z)e^{j\dot{\omega}t}\}]/dz= R\,[\mathrm{Re}\{I_s(z)e^{j\dot{\omega}t}\}] + L\,d\,[\mathrm{Re}\{I_s(z)e^{j\dot{\omega}t}\}]/dt$

$-\,\mathrm{Re}\{d\,V_s(z)e^{j\dot{\omega}t}/dz\}= R\,I_s(z)\,[\mathrm{Re}\{\,e^{j\dot{\omega}t}\}] + L\,\,[\mathrm{Re}\{j\,\dot{\omega}\,I_s(z)e^{j\dot{\omega}t}\}]$

$-d\,V_s(z)/dz= (R + j\dot{\omega}L)I_s(z)$ -------------③

Similarly,

$-d\,I_s(z)/dz= (G + j\dot{\omega}C)V_s(z)$ -------------④

Equation ③ & ④ are called **Telegraphers Equation** or low frequency equation.

The transmission line discussed so far were of lossy type in which the conductors comprising the line are imperfect ($\omega_c = \infty$)and the dielectric in which the conductors are embedded is lossy ($\omega_c = 0$).

Having considered this general case , we may now consider two special cases:

1. Lossless line(R=0=G)
2. Distortionless line(R/l=G/c)

Case-1:Lossless line(R=0=G):- The transmission line is said to be lossless if the conductors of the line are perfect $\omega_c = \infty$ and the dielectric separating between them is lossless($\omega_c = 0$).

For such a line R=0=G .This is the necessary condition for a line to be lossless.

Hence for this line the attenuation constant α=0. But the propagation constant $\gamma=\sqrt{}(R+j\omega L)(G+ j\omega C)$

But we also know that $\gamma=\alpha+j\beta$.

so on solving we get:-

$\sqrt{}\{(RG +R\,j\omega C +G\,j\omega L+j^2\omega^2LC=j\beta$ (∵α=0)

So we get $\sqrt{(j^2\omega^2LC)}=j\beta$

this shows that the phase constant $\beta=\omega\sqrt{LC}$ and for the characteristic impedance $Z_0=\sqrt{\{(R+j\omega L)/(G+j\omega C)\}}$

so $Z_{0=}$ $\sqrt{(L/C)}$

and the phase velocity $v_p=\omega/\beta=1/\sqrt{(LC)}=f\lambda$

Case-2:Distortionless Line(R/L=G/C):- A distortionless line is the one in which the attenuation constant α is frequency independent while the phase constant β is linearly dependent on frequency.

From the expression for α and β a distortionless line results if the parameters are such:- $R/L=G/C$

So $\gamma=\sqrt{(R+j\omega L)(G+j\omega C)}$

Hence $\gamma=\sqrt{RG(1+j\omega L/R)(1+j\omega C/G)}$

$=\sqrt{RG(1+j\omega C/G)^2}=\alpha+j\beta$

so $\alpha=\underline{\sqrt{RG}}$ and $\beta\underline{=\omega\sqrt{LC}}$

The above values shows that α is not dependent on frequency and β is a linearly dependent on frequency.

Also characteristic impedance$=Z_0=\sqrt{(R+j\omega L)/(G+j\omega C)}$

$Z_0=\sqrt{[R(1+j\omega L/R)]/[G(1+j\omega C/G)]}$

$Z_0\underline{=\sqrt{R/G}=\sqrt{L/C}}$

Also the phase velocity $v_p=\omega/\beta=1/\sqrt{LC}=f\lambda$

Or $\lambda=1/f\sqrt{LC}$

Microwave Line:- A microwave line is the one where the parameters are such :-$R<<\omega L$ and $G<<\omega C$

So the characteristic impedance $Z_0=\sqrt{(R+j\omega L)/(G+j\omega C)}$

So substituting the parameters we get:- $\underline{Z_0=\sqrt{L/C}}$

And propagation constant $\gamma=\sqrt{(R+j\omega L)(G+j\omega C)}$

So $\gamma=\sqrt{j\omega L(1+R/j\omega L)\, j\omega c(1+G/j\omega C)}$

$=j\omega\sqrt{LC}[(1+R/j\omega L)^{1/2}(1+G/j\omega C)^{1/2}$

$=j\omega\sqrt{}LC(1+R/2j\omega L)(1+G/2j\omega L)$

$=j\omega\sqrt{}LC \quad [1+R/2j\omega L+G/2j\omega C]$

$\gamma = j\omega\sqrt{}LC+ (R/2)(\sqrt{}C/L) + G\sqrt{}L/C$

but we also know that $\gamma=\alpha+j\beta$

so we have $\alpha\underline{=(R/2)(\sqrt{}C/L)}$ and $\underline{\beta=j\omega\sqrt{}LC}$

and the phase velocity $v_p=\omega/\beta=2\Pi/(j\omega\sqrt{}LC)$

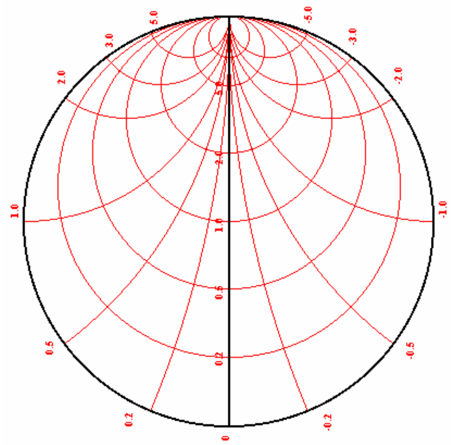
so $v_p= 2\Pi/(j2\Pi f\sqrt{}LC$

hence $v_p= \underline{1/(jf\sqrt{}LC)}$

SMITH CHART

For evaluating the rectangular components, or the magnitude and phase of an input impedance or admittance, voltage, current, and related transmission functions at all points along a transmission line, including:

- Complex voltage and current reflections coefficients
- Complex voltage and current transmission coefficents
- Power reflection and transmission coefficients
- Reflection Loss
- Return Loss
- Standing Wave Loss Factor
- Maximum and minimum of voltage and current, and SWR
- Shape, position, and phase distribution along voltage and current standing waves
- Evaluating effects of shunt and series impedances on the impedance of a transmission line.
- For displaying and evaluating the input impedance characteristics of resonant and anti-resonant stubs including the bandwidth and Q.
- Designing impedance matching networks using single or multiple open or shorted stubs.
- Designing impedance matching networks using quarter wave line sections.
- Designing impedance matching networks using lumped L-C components.
- For displaying complex impedances verses frequency.
- For displaying s-parameters of a network verses frequency.



- It is the most useful graphical tool for transmission line problems.
- From mathematical point of view, the Smith chart is simply a representation of all possible complex impedances with respect to coordinates defined by the reflection coefficient.
- It can be used to convert from reflection coefficients to normalized impedances (or admittances), and vice versa using the impedance (or admittance) circle printed on the chart. When dealing with impedances on a Smith chart, normalized quantities are generally used denoted by lowercase letters. The normalization constant is usually the characteristic impedance of the line($z=Z/Z_0$)
- The domain of definition of the reflection coefficient is a circle of radius 1 in the complex plane.

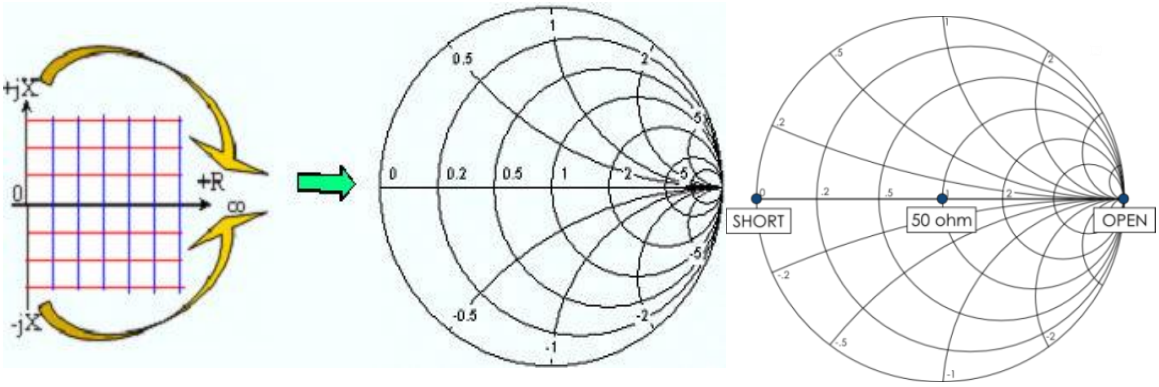


Fig-1: Smith Chart

- At point of Short, r = 0, x = 0 So z_L= r+jx=0
- At point of Open, r = ∞, x = ∞ So z_L= r+jx=∞
- If a lossless line of characteristics impedance Z_0 is terminated with a load impedance Z_L, the reflection coefficient at the load can be written as

$$\Gamma = \frac{z_L - 1}{z_L + 1} = |\Gamma| e^{j\theta}$$

Where $z_L = Z_L / Z_0$ is the normalized impedance

$$z_L = \frac{1+|\Gamma| e^{j\theta}}{1-|\Gamma| e^{j\theta}}$$

Writing z_L and Γ in terms of their real and imaginary parts as $\Gamma = \Gamma_r + j\Gamma_i$ and $z_L = r_L + jx_L$ then

$$z_L = r_L + jx_L = \frac{1+\Gamma_r + j\Gamma_i}{1-\Gamma_r - j\Gamma_i}$$

Multiplying the numerator and denominator by the complex conjugate of the denominator and rearranging gives

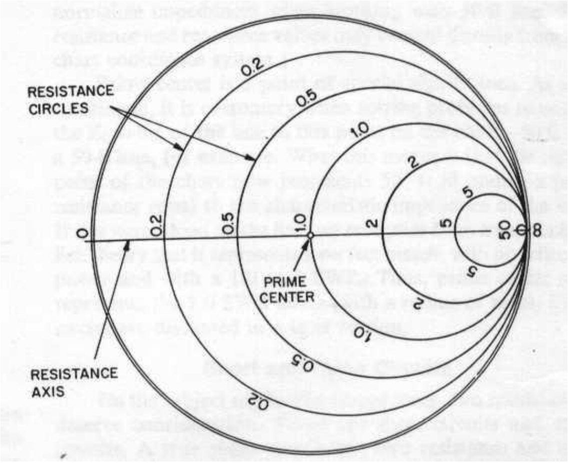
$$\left(\Gamma_r - \frac{r_L}{1+r_L}\right)^2 + \Gamma_i^2 = \left(\frac{1}{1+r_L}\right)^2$$

$$\left(\Gamma_i - \frac{1}{x_L}\right)^2 + \left(\Gamma_r - 1\right)^2 = \left(\frac{1}{x_L}\right)^2$$

The above two equation represents two families of a circle in the Γ_r , Γ_i plane.

Constant Resistance Circle : Center $\left(\frac{r}{r+1},0\right)$ and Radius $\left(\frac{1}{r+1}\right)$

- Constant Reactance Circle : Center $\left(1,\frac{1}{x}\right)$ and Radius $\left(\frac{1}{x}\right)$



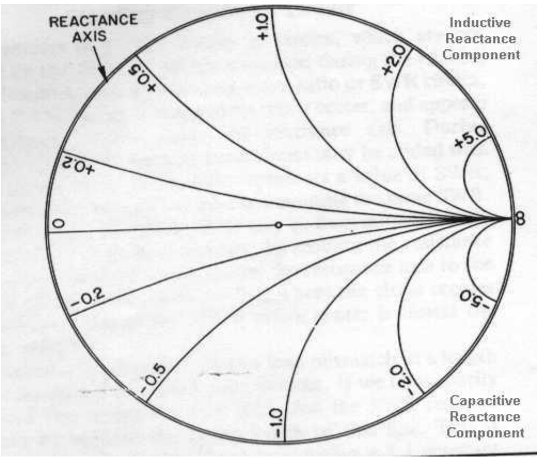


Fig-2: Constant Resistance Circle

Fig-3: Constant Reactance Circle

- Complete revolution on the smith chart is $\frac{\lambda}{2}$ on Transmission line. Distance between Vmax and Vmin is $\frac{\lambda}{4}$ i.e 180.
- Movement from load towards generator is clockwise.

WAVEGUIDE

The transmission line can't propagate high range of frequencies in GHz due to skin effect. Waveguides are generally used to propagate microwave signal and they always operate beyond certain frequency that is called "*cut off frequency*". so they behaves as high pass filter.

SKIN EFFECT :$x_{c=}\frac{1}{(2*\pi*f*c)}$

According to this relation,as frequency increases, x_c tends to zero, that is short circuit. Hence signal becomes grounded and can't propagate further which is called skin effect.

Types of waveguides: - (1)rectangular waveguide (2)cylindrical waveguide (3)elliptical waveguide (4)parallel waveguide

RECTANGULAR WAVEGUIDE :

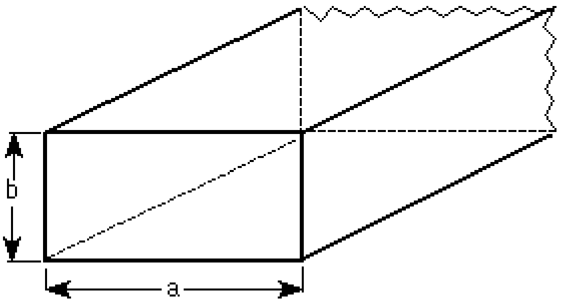
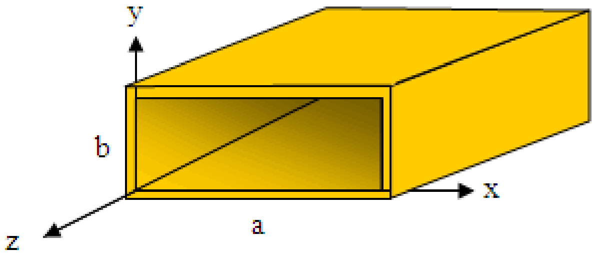


Figure 1. The Rectangular Waveguide



Let us assume that the wave is travelling along z-axis and field variation along z-direction is equal to e^{-Yz},where z=direction of propagation and Y= propagation constant.

Assume the waveguide is lossless (α=0) and walls are perfect conductor (σ=∞). According to maxwell's equation: $\nabla \times$ H=J+$\partial D/\partial t$ and $\nabla \times$ E = -$\partial B/\partial t$.

So $\nabla \times$ H=J$\omega \in E---(1.a)$, $\nabla \times$ E $= -$J$\omega\mu H.$---------(1.b)

Expanding equation (1),

$$\begin{array}{ccc} Ax & Ay & Az \\ |\partial/\partial x & \partial/\partial y & \partial/\partial z| \\ Hx & Hy & Hz \end{array} = \text{J}\omega \in [ExAx + EyAy + EzAz]$$

By equating coefficients of both sides we get,

$\frac{\partial}{\partial y}Hz - \frac{\partial}{\partial z}Hy$ =Jω ∈Ex -----------------2(a)

$-\frac{\partial}{\partial x}Hz + \frac{\partial}{\partial z}Hx = \text{J}\omega \in Ey$ -----------2(b)

$\frac{\partial}{\partial x}Hy - \frac{\partial}{\partial y}Hx = \mathrm{J}\omega \in Ez$ -----------------2(c)

As the wave is travelling along z-direction and variation is along –Ɣz direction.

=> $\frac{\partial}{\partial z}(e^{-\mathrm{Ɣz}})$= -$\gamma e^{-\mathrm{Ɣz}}$.

Comparing above equations, $\frac{\partial}{\partial z} = -\mathrm{Ɣ}$.

So by putting this value of $\frac{\partial}{\partial z}$ in equations 2(a,b,c),we will get

$\frac{\partial}{\partial y}Hz + \mathrm{Ɣ}\, Hy$ =jω ∈Ex -----------------3(a)

$\frac{\partial}{\partial x}Hz + \mathrm{Ɣ}\, Hx = -\mathrm{j}\omega \in Ey$ -----------3(b)

$\frac{\partial}{\partial x}Hy - \frac{\partial}{\partial y}Hx = \mathrm{j}\omega \in Ez$ -----------------3(c)

Similarly from relation $\nabla \times \mathrm{E} = -\mathrm{j}\omega\mu H$ and $\frac{\partial}{\partial z} = -\mathrm{Ɣ}$,we will get

$\frac{\partial}{\partial y}Ez + \mathrm{Ɣ}\, Ey$ = - j$\omega\mu$Hx -----------------4(a)

$\frac{\partial}{\partial x}Ez + \mathrm{Ɣ}\, Ex = \mathrm{j}\omega\mu Hy$ -----------4(b)

$\frac{\partial}{\partial x}Ey - \frac{\partial}{\partial y}Ex$ = - j$\omega\mu Hz$ -----------------4(c)

From equation sets of (3) , we will get : $\frac{\partial}{\partial y}Hz + \mathrm{Ɣ}\, Hy$ =jω ∈Ex

Ex=$\frac{1}{\mathrm{j}\,\omega\in}[\frac{\partial}{\partial y}Hz + \mathrm{Ɣ}\, Hy]$ ---------(5)

From equation sets of (4) ,we will get : $\frac{\partial}{\partial x}Ez + \mathrm{Ɣ}\, Ex = \mathrm{j}\omega\mu Hy$

Ex=$\frac{1}{Ɣ}[\mathrm{j}\omega\mu Hy - \frac{\partial}{\partial x}Ez]$ ---------(6)

Equating equations (5) and (6), we will get

=> $\frac{1}{\mathrm{j}\,\omega\in}[\frac{\partial}{\partial y}Hz + \mathrm{Ɣ}\, Hy]$ =$\frac{1}{Ɣ}[\mathrm{j}\omega\mu Hy - \frac{\partial}{\partial x}Ez]$

=> $\frac{\mathrm{Ɣ}}{\mathrm{j}\,\omega\in}\frac{\partial}{\partial y}Hz + \frac{\mathrm{Ɣ}^2}{\mathrm{j}\,\omega\in}$ Hy = j$\omega\mu Hy - \frac{\partial}{\partial x}Ez$

=> $(\frac{\mathrm{Ɣ}^2}{\mathrm{j}\,\omega\in} - \mathrm{j}\omega\mu)$Hy = $-\frac{\partial}{\partial x}Ez - \frac{\mathrm{Ɣ}}{\mathrm{j}\,\omega\in}\frac{\partial}{\partial y}Hz$

=> $(\frac{\Upsilon^2+\omega^2\mu\in}{j\omega\in})$ Hy $=-\frac{\partial}{\partial x}Ez-\frac{\Upsilon}{j\,\omega\in}\frac{\partial}{\partial y}Hz$

Let $(\Upsilon^2+\omega^2\mu\in)=h^2$

=> $(\frac{h^2}{j\omega\in})$Hy $=-\frac{\partial}{\partial x}Ez-\frac{\Upsilon}{j\omega\in}\frac{\partial}{\partial y}Hz$

Hy $=-\frac{j\omega\in}{h^2}\frac{\partial}{\partial x}Ez-\frac{\Upsilon}{h^2}\frac{\partial}{\partial y}Hz$ ------------ (7)

Similarly we will get by simplifying other equations

Hx $=-\frac{j\omega\in}{h^2}\frac{\partial}{\partial y}Ez-\frac{\Upsilon}{h^2}\frac{\partial}{\partial x}Hz$ ------------ (8)

Ex $=-\frac{j\omega\mu}{h^2}\frac{\partial}{\partial y}Hz-\frac{\Upsilon}{h^2}\frac{\partial}{\partial x}Ez$ ------------ (9)

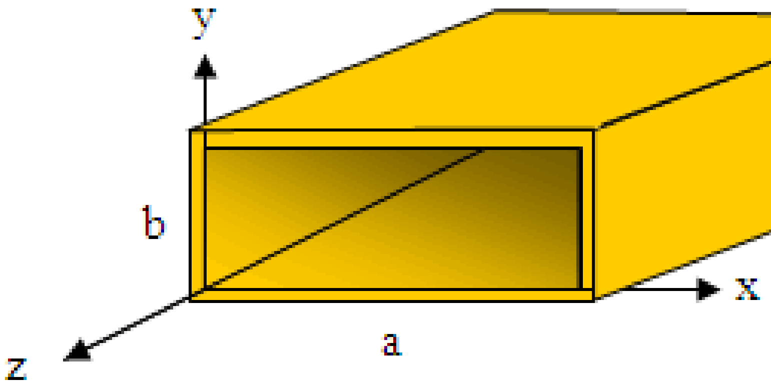
Ey $=\frac{j\omega\mu}{h^2}\frac{\partial}{\partial x}Hz-\frac{\Upsilon}{h^2}\frac{\partial}{\partial y}Ez$ ----------- (10)

<u>IMPORTANT QUESTION</u> :

(Q) Transverse electromagnetic mode(TEM mode)is not possible in a waveguide.why?

(A)Let the wave propagate along z-direction,then the waveguide field equation 7,8,9,10.as wave propagate along z-direction Ez and Hz=0.From equations 7,8,9,10 :Ex,Hx,Ey,Hy=0.hence it is not possible practically.so Ez,Hz both can't be zero.if Ez=0,transverse electric mode exists and Hz=0,transverse magnetic mode exists.

<u>Field solutions of rectangular waveguide :</u>



∗To find the solution we have to assume waveguide is lossless that is $(\alpha=0)$ and walls are perfect conductor$(\sigma=\infty)$. According the Poisson's equation $\nabla^2E=\gamma^2E\ and\ \nabla^2H=\gamma^2H$

*As $\gamma=\alpha+j\beta$ and as $\alpha=0\ so$ the equations becomes $\gamma=j\beta$ and $\gamma^2=-\beta^2=-k^2$(let) *now the Poisson's equation are $\nabla^2 E + k^2 E = 0 - - - - - (a)$

$$and\ \nabla^2 H + k^2 H = 0 - - - -(b)$$

Expanding equation (a) $\frac{\partial^2 Ex}{\partial x^2} + \frac{\partial^2 Ey}{\partial y^2} + \frac{\partial^2 Ez}{\partial z^2} + k^2 E = 0 - - - -(c)$

$\frac{\partial^2 E}{\partial x^2} + \frac{\partial^2 E}{\partial y^2} + \frac{\partial^2 E}{\partial z^2} + k^2 E = 0$ -----it is a second order partial differential equation whose solution let it be E=XYZ where X=X(x),Y=Y(y),Z=Z(z)

Putting E in equation (b) we will get

$$\frac{\partial^2 (XYZ)}{\partial x^2} + \frac{\partial^2 (XYZ)}{\partial y^2} + \frac{\partial^2 (XYZ)}{\partial z^2} + k^2 (XYZ) = 0$$

=>$YZ\frac{\partial^2 X}{\partial x^2} + XZ\frac{\partial^2 Y}{\partial y^2} + XY\frac{\partial^2 Z}{\partial z^2} + k^2 (XYZ) = 0$

Dividing XYZ in both sides,We will get

$$\frac{1}{X}\frac{\partial^2 X}{\partial x^2} + \frac{1}{Y}\frac{\partial^2 Y}{\partial y^2} + \frac{1}{Z}\frac{\partial^2 Z}{\partial z^2} + k^2 = 0$$

$Kx^2 + ky^2 + kz^2 = k^{\ 2}$

As $\frac{1}{X}\frac{\partial^2 X}{\partial x^2} = -\mathrm{Kx}^2$, $\frac{1}{Y}\frac{\partial^2 Y}{\partial y^2} = -\mathrm{Ky}^2$, $\frac{1}{Z}\frac{\partial^2 Z}{\partial z^2}$= -Kz^2 ------------(d)

As the wave is travelling along z-direction Z(z)=$e^{-\gamma z}$

$\frac{\partial^2\ Z(z)}{\partial z^2}$=$\gamma^2 e^{-\gamma z}$

⇨ $\frac{\partial^2\ Z(z)}{\partial z^2}$=$Z(z)\ \gamma^2$

by comparing it we will get $\partial^2/\partial z^2$=γ^2

⇨ $\frac{1}{Z}\frac{\partial^2 Z}{\partial z^2}$= -k$_z$^2 =$\gamma^2$

Similarly $\frac{1}{x}\frac{\partial^2 X}{\partial x^2}$+k$_x$^2 =0

⇨ $\frac{\partial^2 X}{\partial x^2}$+x k$_x$^2 =0

It is a second order homogenous differential equation whose solution is

X(x)=C$_1$cos kxx +C$_2$sin k_xx ------(e)

X(x)=C$_3$cos kyy +C$_4$sin k_yy ------(f)

As E=XYZ ;

$E_{XYZ} = (C_1\cos kxx + C_2\sin k_x x)(C_3\cos kyy + C_4\sin k_y y)e^{-\gamma z}$ ------(g)

Similarly by solving for magnetic field we will get

$H_{XYZ} = (B_1\cos kxx + B_2\sin k_x x)(B_3\cos kyy + B_4\sin k_y y)\ e^{-\gamma z}$ ------(h)

Equations (g) and (h) are the field solutions foe rectangular waveguide.

CASE-1

FIELD SOLUTIONS FOR TRANSVERSE MAGNETIC FIELD IN RECTANGULAR WAVEGUIDE :

Hz=0 and $E_z \neq 0$

$E_{XYZ} = (C_1\ \cos kxx + C_2\sin k_x x)(C_3\cos kyy + C_4\sin k_y y)\ e^{-\gamma z}$

The values of C1,C2,C3,C4,K_x,K_y are found out from boundary equations.as we know that the tangential component of E are constants across the boundary,then

$$E= \begin{cases} 0, x = 0\ and\ x = a \\ 0, y = 0\ and\ y = b \end{cases}$$

AT x=0 AND y=0 ;

$E=C_1C_3e^{-\gamma z} = 0$ but we know that $e^{-\gamma z} \neq 0$ wave is travelling along z-direction.

So either C_1=0 or C_3=0 oterwise $C_1C_{3=}0$

AT x=0 AND y=b ;

$E= C_1(C_3\cos kyb + C_4\sin k_y b)\ e^{-\gamma z}=0$

So $C_1C_{3=}0$

So equation (g)becomes

$E_{XYZ} = (C_2\sin k_x x \times\ C_4\sin k_y y)e^{-\gamma z}$ ------(i)

Hence for x=0,E=0

So $(C_2\sin k_x a \times\ C_4\sin k_y y)e^{-\gamma z} = 0$

$=>\sin kxa = 0 => k_x = \frac{m*\pi}{a}$

In equation (i) for y=b=>E=0;

So (C$_2$sin k_xx $\times$ C$_4$sin k_yb)$e^{-\gamma z} = 0$

=>sin $kyb = 0 => k_y$=$\frac{n*\pi}{b}$

So finally solutions for TRANSVERSE MAGNETIC MODE is given by

$$\boxed{E_{z=}C\ (\sin(\frac{m*\pi}{a})x \ \times\ \sin(\frac{n*\pi}{b})y \times\ e^{-\gamma z}}$$

Where C$_2$ $\times$ C_4=C

CUT-OFF FREQUENCY :

It is the minimum frequency after which propagation occurs inside the waveguide.

As we know that => K$_x$^2 +k$_y$^2+k$_z$^2=k^2

=> K$_x$^2 +k$_y$^2= k^2-k$_z$^2

=> K$_x$^2 +k$_y$^2=k^2+γ^2

As we know that $\beta = -j\omega\sqrt{\mu\varepsilon}\ and\ k^2 = \beta^2$

So we will get that : => K$_x$^2 +k$_y$^2=k^2+γ^2=$\omega^2\mu\varepsilon + \gamma^2$

So $\gamma = \sqrt{\left[\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2 - \omega^2\mu\varepsilon\right]}$

At f=f$_c$ or w=w$_c$,at cut off frequency propagation is about to start. So $\gamma = 0$

=>0=$\sqrt{\left[\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2 - \omega c^2\mu\varepsilon\right]}$

=>$\omega c^2\mu\varepsilon = \left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2$

W$_c$=1/$\sqrt{\mu\varepsilon}$ $\left(\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right)^{1/2}$

$$\boxed{\text{So } f_c=1/2\pi\sqrt{\mu\varepsilon}\ \left(\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right)^{1/2} \text{---------}\textbf{\textit{cut off frequency equation}}}$$

where m=n=0,1,2,3……

At free space f$_c$=1/2$\pi\sqrt{\mu o\varepsilon o}$ $\left(\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right)^{1/2}$

$$\boxed{=>f_c=c/2\left(\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right)^{1/2} \text{---------}\textbf{\textit{cut off frequency equation in free space}}}$$

CUT – OFF WAVELENGTH:

This is given by

$$\boxed{\not{y}c = \frac{c}{f} = 2 \times (\frac{1}{\left(\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right)^{1/2}})}$$

DOMINANT MODE :

The mode having lowest cut-off frequency or highest cut-off wavelength is called DOMINANT MODE.

*the mode can be TM$_{01}$,TM$_{10,}$TM$_{11,}$But for TM$_{10}$ and TM$_{01}$,wave can't exist.

*hence TM$_{11}$has lowest cut-off frequency and is the DOMINANT MODE in case of all TM modes only.

PHASE CONSTANT :

As we know that $\gamma = \left(\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2 - \omega^2\mu\varepsilon \quad \right)^{1/2}$

So j$\beta = \sqrt{\omega^2\mu\varepsilon - \omega c^2\mu\varepsilon}$

This condition satisfies that only $\omega c^2\mu\varepsilon > \omega^2\mu\varepsilon$

$$\boxed{\text{So that } \beta = \sqrt{\omega^2\mu\varepsilon - \omega c^2\mu\varepsilon}}$$

PHASE VELOCITY :

It is given by Vp=ω/ β

V$_p$=$\frac{\omega}{(\omega^2\mu\varepsilon - \omega c^2\mu\varepsilon)}$

$$\boxed{V_p=1/ [\sqrt{}\omega\varepsilon(1 - fc^2/f^2)]}$$

GUIDE WAVELENGTH :

It is given by $\lambda g = \frac{2\pi}{\beta} = \frac{2\pi}{\sqrt{\omega^2\mu\varepsilon - \omega c^2\mu\varepsilon}}$

λg =1/$\sqrt{}$$(f^2\mu\varepsilon - fc^2\mu\varepsilon)$

$$\lambda g = \frac{c}{f}/(1 - fc^2/f^2)$$

$$\boxed{\Rightarrow \lambda g = \frac{\lambda o}{[1-\left(\frac{\lambda o}{\lambda c}\right)^2]^{\frac{1}{2}}}}$$

⇨ $\frac{1}{\lambda g^2}$=1/ $\lambda o^2 - \frac{1}{\lambda c^2}$

CASE -2

SOLUTIONS OF TRANSVERSE ELECTRIC MODE :

Here E_z=0 and $H_z \neq 0$

H_z=(B_1cos kxx +B_2sin k_xx)(B_3cos kyy +B_4sin k_yy) $e^{-\gamma z}$

B_1,B_2,B_3,B_4,K_X,K_Y are found from boundary conditions.

$$(Ex = 0|for\ y = 0\ and\ y = b)$$

$$(Ey = 0|for\ x = 0\ and\ x = a)$$

At x=0 and y=0 ;

$\mathrm{Ey} = \frac{\mathrm{J}\omega\mu}{\mathrm{h}^2}\frac{\partial}{\partial x}Hz - \frac{\Upsilon}{h^2}\ \frac{\partial}{\partial y}Ez$,as $\frac{\Upsilon}{h^2}\ \frac{\partial}{\partial y}Ez = 0$

So $\mathrm{Ey} = \frac{\mathrm{J}\omega\mu}{\mathrm{h}^2}\frac{\partial}{\partial x}Hz$

Here

$\frac{\partial}{\partial x}Hz = [\mathrm{B1} * \mathrm{kx} * (-\sin kxx) + \mathrm{B2} * \mathrm{kx} * \cos k\ \mathrm{xx})(\ \mathrm{B3}\cos kyy + \mathrm{B4}\sin k\ \mathrm{yy}\ )\ e^{-\gamma z}]$

So $\mathrm{Ey} = \frac{\mathrm{J}\omega\mu}{\mathrm{h}^2}[\mathrm{B1} * \mathrm{kx} * (-\sin kxx) + \mathrm{B2} * \mathrm{kx} * \cos k\ \mathrm{xx})(\ \mathrm{B3}\cos kyy + \mathrm{B4}\sin k\ \mathrm{yy}\ )\ e^{-\gamma z}]$

At x=0, $\frac{\partial}{\partial x}Hz = 0$

0=[B_2*K_x][(B_3cosK_yy+B_4sinK_yy)$e^{-\gamma z}$]

From this B_2=0

So , $\mathrm{Ex} = -\frac{\mathrm{j}\omega\mu}{\mathrm{h}^2}\ \frac{\partial}{\partial y}Hz - \frac{\Upsilon}{h^2}\ \frac{\partial}{\partial x}Ez$ [as $\frac{\Upsilon}{h^2}\ \frac{\partial}{\partial x}Ez = 0$];

$\mathrm{Ex} = -\frac{\mathrm{j}\omega\mu}{\mathrm{h}^2}\ \frac{\partial}{\partial y}Hz$

Here

$\frac{\partial}{\partial y}Hz = [\mathrm{B1}\ (\cos kxx) + \mathrm{B2}\sin k\ \mathrm{xx})(-\ \mathrm{B3} * \mathrm{ky} * \sin kyy + \mathrm{B4} * \mathrm{ky} * \cos k\ \mathrm{yy}\ )\ e^{-\gamma z}]$

$\mathrm{Ex} = -\frac{\mathrm{J}\omega\mu}{\mathrm{h}^2}[\mathrm{B1}\ (\cos kxx) + \mathrm{B2}\sin k\ \mathrm{xx})(-\ \mathrm{B3} * \mathrm{ky} * \sin kyy + \mathrm{B4} * \mathrm{ky} * \cos k\ \mathrm{yy}\ )\ e^{-\gamma z}]$

$$\mathrm{B1}\ (\cos kxx) + \mathrm{B2}\sin k\ \mathrm{xx})(-\ \mathrm{B3} * \mathrm{ky} * \sin kyy + \mathrm{B4} * \mathrm{ky} * \cos k\ \mathrm{yy}\ )\ e^{-\gamma z}\ = 0$$

At y=0 , $\frac{\partial}{\partial y}Hz = 0$

$$[\mathrm{B1}\ (\cos kxx) + \mathrm{B2} \sin k\ \mathrm{xx})(\mathrm{B4} * \mathrm{ky}\ )e^{-\gamma z} = 0$$

From this B_4=0

So H_z= $\mathrm{B1}\ (\cos kxx) * \ \mathrm{B3} \cos kyy\ \ * e^{-\gamma z}$

$$\frac{\partial}{\partial x}Hz = [\mathrm{B1} * \mathrm{kx} * (-\sin kxx)\ (\ \mathrm{B3} \cos kyy\ )\ e^{-\gamma z}]$$

Here we know that at x=a,E_y=0

So E_y= $\frac{\mathrm{J}\omega\mu}{\mathrm{h}^2}[-\mathrm{B1} * \mathrm{kx} * (-\sin kxa) * \ \ \mathrm{B3} \cos kyy * \ \ e^{-\gamma z}]$=0

$$\sin kxa = 0 => kx = \frac{m\pi}{a}$$

At y=b,E_x=0

So E_x=$-\frac{\mathrm{J}\omega\mu}{\mathrm{h}^2}[\mathrm{B1}\ (\cos kxx)\ (\mathrm{B3} * \mathrm{ky} * \sin kyb)\ e^{-\gamma z}]$=0

So here

$$\sin kyb = 0 => ky = \frac{n\pi}{b}$$

So the general TRANSVERSE ELECTRIC MODE solution is given by

$$\boxed{H_z = \mathrm{B}\ (\cos \frac{m\pi}{a}x)\ (\cos \frac{n\pi}{b}y)\ e^{-\gamma z}}$$

Where B=B_1B_3

CUT-OFF FREQUENCY :

The cut-off frequency is given as

$\Rightarrow$ f_c=c/2 $\left(\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right)^{1/2}$ ---------***cut off frequency equation in free space***

DOMINANT MODE :

The mode having lowest cut-off frequency or highest cut-off wavelength is called DOMINANT MODE.here TE_{00} where wave can’t exist.

So $f_{c(TE01)}$=c/2b

$f_{c(TE10)}$=c/2a

for rectangular waveguide we know that a>b so TE_{10} is the dominant mode in all rectangular waveguide.

DEGENERATE MODE :

The modes having same cut-off frequency but different field equations are called degenerate modes.

WAVE IMPEDANCE :

Impedance offered by waveguide either in TE mode or TM mode when wave travels through ,it is called wave impedance.

For TE mode

$$\boxed{\eta TE = \frac{\eta i}{\sqrt{}(1 - \frac{fc^2}{f^2})}}$$

And

$$\boxed{\eta TM = \eta i * \sqrt{}(1 - \frac{fc^2}{f^2})}$$

Where ηi = intrinsic impedance = 377ohm = 120π

CYLINDRICAL WAVEGUIDES

A circular waveguide is a tubular, circular conductor. A plane wave propagating through a circular waveguide results in transverse electric (TE) or transverse magnetic field(TM) mode.

Assume the medium is lossless(α=0) and the walls of the waveguide is perfect conductor(σ=∞).

The field equations from MAXWELL'S EQUATIONS are:-

$\nabla x E = -j\omega\mu H$ ------(1.a)

$\nabla x H = j\omega\epsilon E$ -----(1.b)

Taking the first equation,

$\nabla x E = -j\omega\mu H$

Expanding both sides of the above equation in terms of cylindrical coordinates, we get

$$\frac{1}{\rho} * \begin{vmatrix} A\rho & \rho A\varphi & Az \\ \partial/\partial\rho & \partial/\partial\varphi & \partial/\partial z \\ E\rho & \rho E\varphi & Ez \end{vmatrix} = -j\omega\mu[H\rho A\rho + H\varphi A\varphi + HzAz]$$

Equating :-

$$\frac{1}{\rho}\left\{\frac{\partial(Ez)}{\partial\varphi} - \frac{\partial(\rho E\varphi)}{\partial z}\right\} = -j\omega\mu H\rho \quad \text{------ (2.a)}$$

$$\left\{\frac{\partial(E\rho)}{\partial z} - \frac{\partial(Ez)}{\partial\rho}\right\} = -j\omega\mu H\varphi \quad \text{------ (2.b)}$$

$$\frac{1}{\rho}\left\{\frac{\partial(\rho E\varphi)}{\partial\rho} - \frac{\partial(E\rho)}{\partial\varphi}\right\} = -j\omega\mu Hz \quad \text{------- (2.c)}$$

Similarly expanding $\nabla x H = j\omega\epsilon E$,

$$\frac{1}{\rho} * \begin{vmatrix} A\rho & \rho A\varphi & Az \\ \partial/\partial\rho & \partial/\partial\varphi & \partial/\partial z \\ H\rho & \rho H\varphi & Hz \end{vmatrix} = j\omega\epsilon[E\rho A\rho + E\varphi A\varphi + EzAz]$$

Equating:

$$\frac{1}{\rho}\left\{\frac{\partial(Hz)}{\partial\varphi} - \frac{\partial(\rho H\varphi)}{\partial z}\right\} = j\omega\epsilon E\rho \quad \text{------ (3.a)}$$

$$\left\{\frac{\partial(H\rho)}{\partial z} - \frac{\partial(Hz)}{\partial\rho}\right\} = -j\omega\epsilon E\varphi \quad \text{------- (3.b)}$$

$$\frac{1}{\rho}\left\{\frac{\partial(\rho H\varphi)}{\partial\rho} - \frac{\partial(H\rho)}{\partial\varphi}\right\} = j\omega\epsilon Ez \quad \text{------- (3.c)}$$

Let us assume that the wave is propagating along z direction. So,

$$\text{Hz}=e^{-\gamma z}\ ;$$

$$=>\frac{\partial Hz}{\partial z}=-\gamma e^{-\gamma z}$$

$$\boxed{\frac{\partial}{\partial z}=-\gamma}$$

Putting in equation 2 and 3:

$$\left\{\frac{\partial(Ez)}{\partial\varphi}+\gamma\rho E\varphi\right\}=\text{-j}\omega\mu\rho\text{H}\rho\quad\text{------ (4.a)}$$

$$\left\{\gamma E\rho+\frac{\partial(Ez)}{\partial\rho}\right\}=\text{j}\omega\mu\rho\text{H}\varphi\quad\text{------- (4.b)}$$

$$\left\{\frac{\partial(\rho E\varphi)}{\partial\rho}-\frac{\partial(E\rho)}{\partial\varphi}\right\}=\text{-j}\omega\mu\rho\text{Hz}\quad\text{------- (4.c)}$$

And

$$\frac{1}{\rho}\left\{\frac{\partial(Hz)}{\partial\varphi}+\gamma\rho H\varphi\right\}=\text{j}\omega\epsilon\text{E}\rho\quad\text{------ (5.a)}$$

$$\left\{\gamma H\rho+\frac{\partial(Hz)}{\partial\rho}\right\}=\text{j}\omega\epsilon\text{E}\varphi\quad\text{------- (5.b)}$$

$$\frac{1}{\rho}\left\{\frac{\partial(\rho H\varphi)}{\partial\rho}-\frac{\partial(H\rho)}{\partial\varphi}\right\}=\text{j}\omega\epsilon\text{E}z\quad\text{------- (5.c)}$$

Now from eq(4.a) and eq(5.b),we get

$$\text{H}\rho=\frac{1}{-\text{j}\omega\mu\rho}\left\{\frac{\partial(Ez)}{\partial\varphi}+\gamma\rho E\varphi\right\};\ \ \text{H}\rho=\frac{1}{\gamma}[\text{-j}\omega\epsilon\text{E}\varphi\text{-}\frac{\partial(Hz)}{\partial\rho}]$$

$$\therefore\frac{1}{-\text{j}\omega\mu\rho}\left\{\frac{\partial(Ez)}{\partial\varphi}+\gamma\rho E\varphi\right\}=\ \frac{1}{\gamma}[\text{-j}\omega\epsilon\text{E}\varphi\text{-}\frac{\partial(Hz)}{\partial\rho}]$$

$$=>\frac{1}{\rho}\frac{\partial(Ez)}{\partial\varphi}+\gamma E\varphi=\text{ - }\frac{\omega^2\mu\varepsilon}{\gamma}\text{E}\varphi\ +\frac{\text{j}\omega\mu}{\gamma}\frac{\partial\text{Hz}}{\partial\rho}$$

$$=>(\frac{\gamma^2+\omega^2\mu\varepsilon}{\gamma})\text{E}\varphi=\frac{\text{j}\omega\mu}{\gamma}\frac{\partial\text{Hz}}{\partial\rho}\text{ - }\frac{1}{\rho}\frac{\partial\text{Hz}}{\partial\rho}$$

Let $(\gamma^2+\omega^2\mu\varepsilon)=\text{h}^2=\text{Kc}^2$;

For lossless medium $\alpha=0;\gamma=\text{j}\beta$;

Now the final equation for Eφ is

$$E\varphi = \frac{-j}{Kc^2}\left(\frac{\beta}{\rho}\frac{\partial Ez}{\partial \varphi} - \omega\mu\frac{\partial Hz}{\partial \rho}\right) \quad \text{----- (6.a)}$$

$$H\varphi = \frac{-j}{Kc^2}\left(\omega\varepsilon\frac{\partial Ez}{\partial \rho} + \frac{\beta}{\rho}\frac{\partial Hz}{\partial \varphi}\right) \quad \text{----- (6.b)}$$

$$E\rho = \frac{-j}{Kc^2}\left(\frac{\omega\mu}{\rho}\frac{\partial Hz}{\partial \varphi} + \beta\frac{\partial Ez}{\partial \rho}\right) \quad \text{----- (6.c)}$$

$$H\rho = \frac{j}{Kc^2}\left(\frac{\omega\varepsilon}{\rho}\frac{\partial Ez}{\partial \varphi} - \beta\frac{\partial Hz}{\partial \rho}\right) \quad \text{------ (6.d)}$$

Equations (6.a),(6.b),(6.c),(6.d) are the field equations for cylindrical waveguides.

TE MODE IN CYLINDRICAL WAVEGUIDE :-

For TE mode, $E_z=0$, $H_z \neq 0$.

As the wave travels along z-direction,$e^{-\gamma z}$ is the solution along z-direction.

As, $\gamma^2 + \omega^2\mu\varepsilon = h^2$;

⇨ $-\beta^2+\omega^2\mu\epsilon = Kc^2$;

⇨ $-\beta^2+K^2 = Kc^2$ (as $K^2= \omega^2\mu\varepsilon$)

According to maxwell's equation, the laplacian of H_z :

$$\nabla^2 H_z = -\omega^2\mu\varepsilon H_z;$$

⇨ $\nabla^2 H_z + \omega^2\mu\varepsilon H_z = 0$;

⇨ $\nabla^2 H_z + K^2 H_z = 0$

Expanding the above equation, we get:

$$\frac{\partial^2 Hz}{\partial \rho^2} + \frac{1}{\rho}\frac{\partial Hz}{\partial \rho} + \frac{1}{\rho^2}\frac{\partial^2 HZ}{\partial \varphi^2} + \frac{\partial^2 Hz}{\partial z^2} + K^2 H_z = 0;$$

Now, $H_Z= e^{-\gamma z}$; $\frac{\partial^2 Hz}{\partial z^2} = (-\gamma)^2\, e^{-\gamma z}$; $\frac{\partial^2 Hz}{\partial z^2} = -\beta^2\, e^{-\gamma z}$;

$$\frac{\partial^2 Hz}{\partial z^2} = -\beta^2 Hz \quad => \frac{\partial^2}{\partial z^2} = -\beta^2;$$

Putting this value in the above equations, we get:-

$$\frac{\partial^2 Hz}{\partial \rho^2} + \frac{1}{\rho}\frac{\partial Hz}{\partial \rho} + \frac{1}{\rho^2}\frac{\partial^2 HZ}{\partial \varphi^2} - \beta^2 Hz + K^2 H_z = 0;$$

$$\frac{\partial^2 Hz}{\partial \rho^2} + \frac{1}{\rho}\frac{\partial Hz}{\partial \rho} + \frac{1}{\rho^2}\frac{\partial^2 HZ}{\partial \varphi^2} + Kc^2 H_z = 0; \quad [\text{as } -\beta^2+K^2 = Kc^2\ ]$$

$\frac{\partial^2 Hz}{\partial\rho^2} + \frac{1}{\rho}\frac{\partial Hz}{\partial\rho} + Kc^2 H_z = -\frac{1}{\rho^2}\frac{\partial^2 HZ}{\partial\varphi^2};$

The partial differential with respect to ρ and φ in the above equation are equal only when the individuals are constant (Let it be K_o^2).

⇨ $-\frac{1}{\rho^2}\frac{\partial^2 HZ}{\partial\varphi^2} = K_o^2$

⇨ $\frac{\partial^2 HZ}{\partial\varphi^2} + \rho^2 K_o^2 = 0$

Solutions to the above differential equation is:-

$Hz = B_1 \sin(K_o\varphi) + B_2\cos(K_o\varphi)$-----{solution along φ direction}.

Now,

$$\frac{\partial^2 Hz}{\partial\rho^2} + \frac{1}{\rho}\frac{\partial Hz}{\partial\rho} + (Kc^2 H_z - K_o^2) = 0$$

This equation is similar to BESSEL'S EQUATION,so the solution of this equation is

$Hz=CnJn(K_c\,\rho)$ --------{solution along ρ-direction}

Hence,

$Hz=Hz(\rho)Hz(\varphi)e^{-\gamma z};$

So, the final solution is,

$$\boxed{Hz= CnJn(K_c\,\rho)[\,B_1\sin(K_o\varphi)+B_2\cos(K_o\varphi)]\,e^{-\gamma z}}$$

Applying boundary conditions:-

At ρ=a, Eφ=0 $\Rightarrow \frac{\partial Hz}{\partial\rho}=0,$

=> $J_n(K_c\,\rho) = 0$

=> $J_n(K_c\,a) = 0$

If the roots of above equation are defined as P_{mn}',then

$K_c = \frac{Pmn'}{a};$

∴ $$\boxed{Hz= CnJn\left(\frac{Pmn'}{a}\rho\right)[\,B_1\sin(K_o\varphi)+B_2\cos(K_o\varphi)]\,e^{-\gamma z}}$$

<u>CUT-OFF FREQUENCY</u> :- It is the minimum frequency after which the propagation occurs inside the cavity.

∴ $\gamma=0;$

But we know that $\gamma^2 + \omega^2\mu\varepsilon = \text{Kc}^2$;

⇨ $\omega^2\mu\varepsilon = \text{Kc}^2$

⇨ $\omega^2 = \frac{\text{Kc}^2}{\mu\varepsilon}$

⇨ $2\pi f_c = \sqrt{\frac{\text{Kc}^2}{\mu\varepsilon}}$

⇨ $f_c = \frac{1}{2\pi}\sqrt{\frac{\text{Kc}^2}{\mu\varepsilon}}$

$$\boxed{\therefore \quad f_c = \frac{\text{Pmn}'}{2\pi a\sqrt{\mu\varepsilon}}}$$

CUTOFF WAVELENGTH :- $\frac{c}{fc} = \frac{\frac{1}{\sqrt{\mu\varepsilon}}}{\frac{\text{Pmn}'}{2\pi a\sqrt{\mu\varepsilon}}} = \frac{2\pi a}{\text{Pmn}'}$

The experimental values of Pmn' are:-

n \ m	1	2	3
0	3.832	7.016	10.173
1	1.841	5.331	8.536
2	3.054	6.706	9.969
3	3.054	6.706	9.970

As seen from this table, TE_{11} mode has the lowest cut off frequency, hence $\mathbf{TE_{11}}$ is the dominating mode.

$$\boxed{\textbf{WAVE IMPEDANCE}(\mathbf{Z_{TE}}) = \frac{\eta k}{\beta}}$$

TM MODE IN CYLINDRICAL WAVEGUIDE :-

For TM mode Hz=0,Ez≠0.

From the calculation in TE mode, we got the equation as,

$$\frac{\partial^2 \text{Ez}}{\partial \rho^2} + \frac{1}{\rho}\frac{\partial \text{Ez}}{\partial \rho} + \frac{1}{\rho^2}\frac{\partial^2 \text{Ez}}{\partial \varphi^2} + \text{Kc}^2 E_z = 0; \text{-------}(2)$$

⇨ $\frac{\partial^2 \text{Ez}}{\partial \varphi^2} + \rho^2 K_1{}^2 = 0$ (let k_1 be a constant)

So, the general solution of the above equation is

$E_z=C_1\sin(k_1\varphi)+C_2\cos(k_1\varphi)$

Applying the boundary value conditions,i.e at $\rho=a$,$Ez=0$.

We get $J_n(K_c\, a)=0$;

So,if P_{mn} is the root of the above equation, then

$$K_c=\frac{\mathrm{Pmn}}{a}$$

The experimental values of Pmn' are:-

n m	1	2	3
0	2.405	5.520	8.654
1	3.832	7.016	10.174
2	5.135	8.417	11.620

So, we can conclude that the TE_{01} having a value of 2.405 is the lowest one. But since this value is greater than the TE_{11} mode, so mode **TE_{11}** is the dominant mode for the cylindrical TM modes. Here $m\geq1$,so there is no TM_{10} mode.

CUTOFF FREQUENCY:

The cut off frequency is given by

$$f_c=\frac{\mathrm{Pmn}}{2\pi a\sqrt{\mu\varepsilon}}$$

CUTOFF WAVELENGTH:-

The cut off wavelength is given by

$$\frac{c}{fc}=\frac{\frac{1}{\sqrt{\mu\varepsilon}}}{\frac{\mathrm{Pmn}}{2\pi a\sqrt{\mu\varepsilon}}}=\frac{2\pi a}{\mathrm{Pmn}}$$

WAVE IMPEDANCE :-

The wave impedance is given by $\frac{\eta k}{\beta}$.

MICROWAVE COMPONENTS

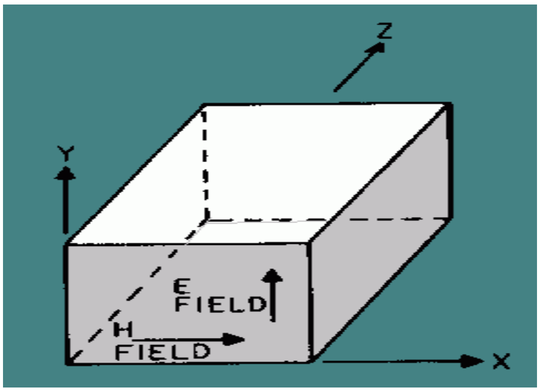
MICROWAVE RESONATOR:

- They are used in many applications such as oscillators, filters, frequency meters, tuned amplifiers and the like.
- A microwave resonator is a metallic enclosure that confines electromagnetic energy and stores it inside a cavity that determines its equivalent capacitance and inductance and from the energy dissipated due to finite conductive walls we can determine the equivalent resistance.
- The resonator has finite number of resonating modes and each mode corresponds to a particular resonant frequency.
- When the frequency of input signal equals to the resonant frequency, maximum amplitude of standing wave occurs and the peak energy stored in the electric and magnetic field are calculated.

RECTANGULAR WAVEGUIDE CAVITY RESONATOR:

- Resonator can be constructed from closed section of waveguide by shorting both ends thus forming a closed box or cavity which store the electromagnetic energy and the power can be dissipated in the metallic walls as well as the dielectric medium

DIAGRAM:



- The geometry of rectangular cavity resonator spreads as

  0≤x≤a;

  0≤y≤b;

  0≤z≤d

- Hence the expression for cut-off frequency will be

$$\omega_0^2 \mu € = (m\pi/a)^2 + (n\pi/b)^2 + (l\pi/d)^2$$

or $$\omega_0 = \frac{1}{\sqrt{\mu €}} [(m\pi/a)^2 + (n\pi/b)^2 + (l\pi/d)^2]^{1/2}$$

or $$f_0 = \frac{1}{2\pi\sqrt{\mu €}} [(m\pi/a)^2 + (n\pi/b)^2 + (l\pi/d)^2]^{1/2}$$

or $$f_0 = \frac{c'}{2} [(m\pi/a)^2 + (n\pi/b)^2 + (l\pi/d)^2]^{1/2}$$

- This is the expression for resonant frequency of cavity resonator.
- The mode having lowest resonant frequency is called DOMINANT MODE and for TE AND TM the dominant modes are TE-101 and TM-110 respectively.

QUALITY FACTOR OF CAVITY RESONATOR:

- $Q=2\pi \times \frac{\text{maximum energy stored per cycle}}{\text{energy dissipated per cycle}}$

FACTORS AFFECTING THE QUALITY FACTOR:

Quality factor depends upon 2 factors:

- Lossy conducting walls
- Lossy dielectric medium of a waveguide

1) LOSSY CONDUCTING WALL:

- The Q-factor of a cavity with lossy conducting walls but lossless dielectric medium i.e. $\sigma_c \neq \infty$ and $\sigma = 0$

Then $Q_c = (2\omega_0 W_e/P_c)$

Where w_0-resonant frequency

W_e-stored electrical energy

P_c-power loss in conducting walls

- From dimensional point of view:

$$Q_c = (kad^3)(b\eta) \times \{ 1/(2l^2a^3d + l^3a^3d + ad^3 + 2bd^3) \} /(2\pi^2 R_s)$$

Where $k = (\omega^2 \mu €)^{1/2}$

$$\eta = \eta_i/\sqrt{\epsilon_r} = 377/\sqrt{\epsilon_r}$$

2) LOSSY DIELECTRIC MEDIUM:

The Q-factor of a cavity with lossy dielectric medium but lossless conducting walls

i.e. $\sigma_c = \infty$ and $\sigma \neq 0$

$$\mathbf{Q_d = 2\omega_0 W_e / P_d = (1/\tan\delta)}$$

Where $\tan\delta = \frac{\sigma}{\omega\epsilon}$

P_d=power loss in dielectric medium

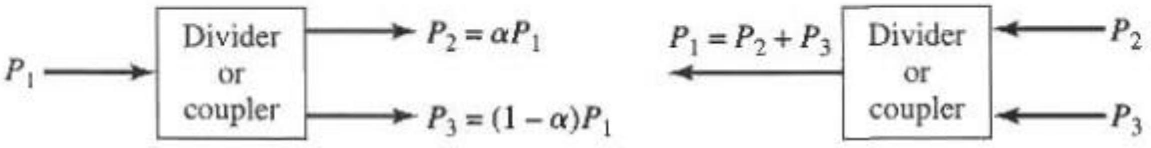
- When both the conducting walls and the dielectric medium are lossy in nature then

Total power loss = $\mathbf{P_c + P_d}$

$$\mathbf{1/Q_{total} = 1/Q_c + 1/Q_d}$$

$$\mathbf{or\ Q_{total} = 1/\,(1/Q_c + 1/Q_d)}$$

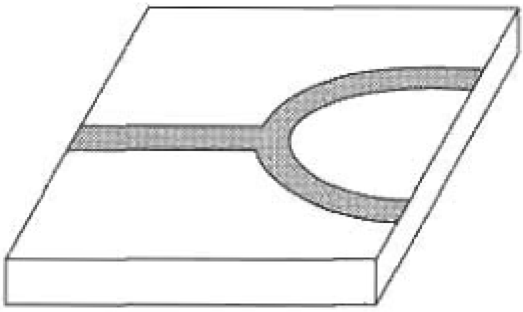
POWER DIVIDERS:



- Power dividers and combiners are the passive microwave components that are used for power division and combination in microwave frequency range
- In this case, the input power is divided into two or more signals of lesser power.
- The power divider has certain basic parameters like isolation, coupling factor and directivity.

T-JUNCTION POWER DIVIDER USING WAVEGUIDE:

The T-junction power divider is a 3-port network that can be constructed either from a transmission line or from the waveguide depending upon the frequency of operation.

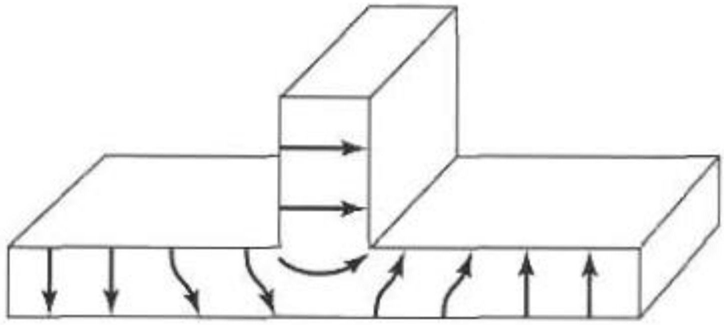


For very high frequency, power divider using waveguide is of 4 types

⇨ E-Plane Tee
⇨ H-Plane Tee
⇨ E-H Plane Tee/Magic Tee
⇨ Rat Race Tee

E-PLANE TEE:

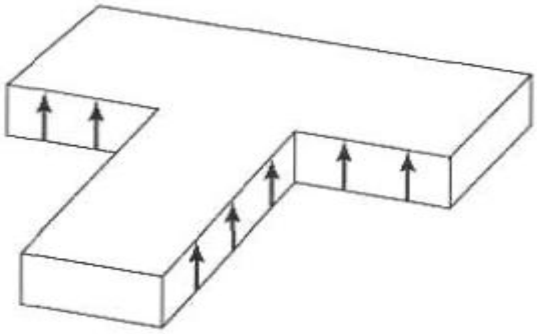
Diagram



- It can be constructed by making a rectangular slot along the wide dimension of the main waveguide and inserting another auxiliary waveguide along the direction so that it becomes a 3-port network.
- Port-1 and Port-2 are called collinear ports and Port-3 is called the E-arm.
- E-arm is parallel to the electric field of the main waveguide.
- If the wave is entering into the junction from E-arm it splits or gets divided into Port-1 and Port-2 with equal magnitude but opposite in phase
- If the wave is entering through Port-1 and Port-2 then the resulting field through Port-3 is proportional to the difference between the instantaneous field from Port-1 and Port-2

H-PLANE TEE:

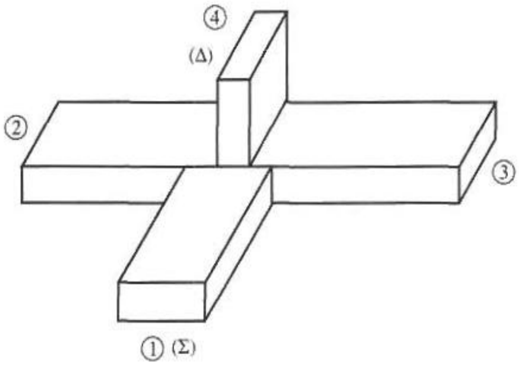
Diagram:



- An H-plane tee is formed by making a rectangular slot along the width of the main waveguide and inserting an auxiliary waveguide along this direction.
- In this case, the axis of the H-arm is parallel to the plane of the main waveguide.
- The wave entering through H-arm splits up through Port-1 and Port-2 with equal magnitude and same phase
- If the wave enters through Port-1 and Port-2 then the power through Port-3 is the phasor sum of those at Port-1 and Port-2.
- E-Plane tee is called PHASE DELAY and H-Plane tee is called PHASE ADVANCE.

E-H PLANE TEE/MAGIC TEE:

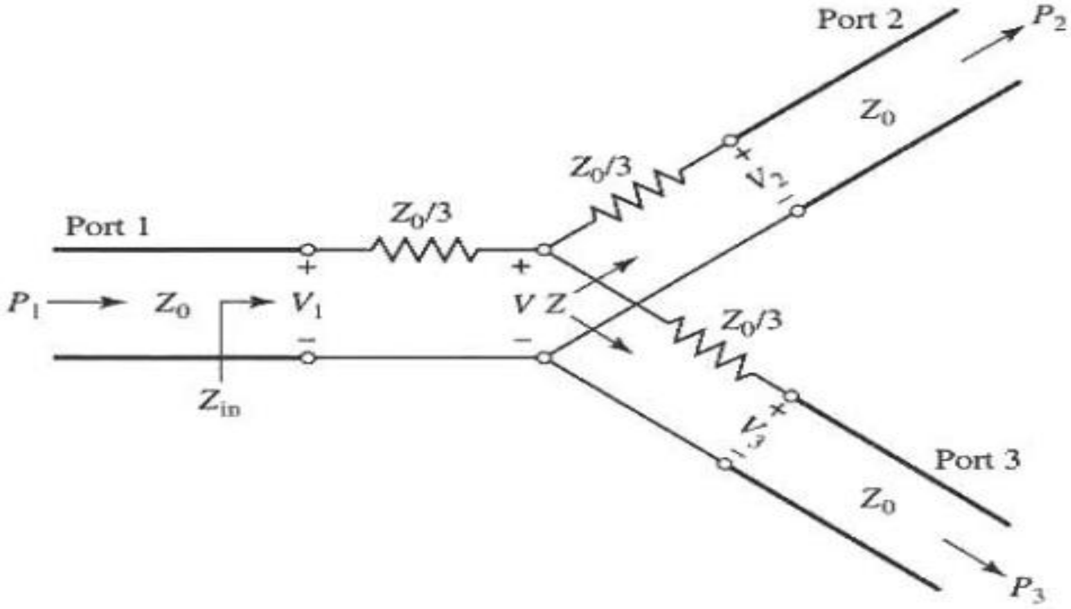
Diagram:



- It is a combination of E-Plane tee and H-Plane tee.
- If two waves of equal magnitude and the same phase are fed into Port-1 and Port-2, the output will be zero at Port-3 and additive at Port-4.
- If a wave is fed into Port-4 (H-arm) then it will be divided equally between Port-1 and Port-2 of collinear arms (same in phase) and will not appear at Port-3 or E-arm.
- If a wave is fed in Port-3 then it will produce an output of equal magnitude and opposite phase at Port-1 and Port-2 and the output at Port-4 will be zero.
- If a wave is fed in any one of the collinear arms at Port-1 or Port-2, it will not appear in the other collinear arm because the E-arm causes a phase delay and the H-arm causes phase advance.

T-JUNCTION POWER DIVIDER USING TRANSMISSION LINE:

Diagram:



- It is a junction of 3 transmission lines

- In this case, if P1 is the input port power then P2 and P3 are the power of output Port-2 and Port-3 respectively.
- To transfer maximum power from port-1 to port-2 and port-3 the impedance must match at the junction.

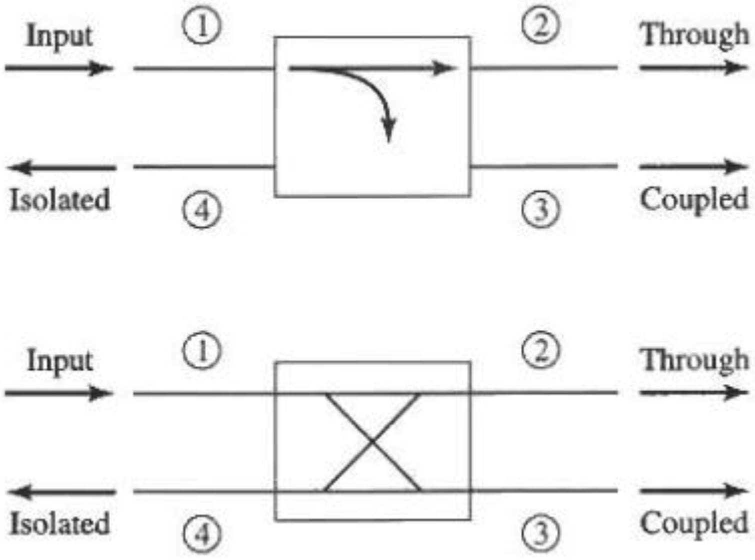
⇨ For maximum power transfer

$Z_{01} = Z_{02} \| Z_{03}$

⇨ $1/Z_{01} = 1/Z_{02} + 1/Z_{03}$ (Condition for lossless power division)

DIRECTIONAL COUPLER:

Diagram:



- It is a 4- port waveguide junction consisting of a primary waveguide 1-2 and a secondary waveguide 3-4.
- When all the ports are terminated in their characteristic impedance there is free transmission of power without reflection between port-1 and port-2 and no power transmission takes place between port-1 and port-3 or port-2 and port-4 a sno coupling exists.
- The characteristic of a directional coupler is expressed in terms of its coupling factor and directivity.
- The coupling factor is the measure of ratio of power levels in primary and secondary lines.
- Directivity is the measure of how well the forward travelling wave in the primary waveguide couples only to a specific port of the secondary waveguide.
- In ideal case, directivity is infinite i.e. power at port-3 =0 because port-2 and port-4 are perfectly matched.
- Let wave propagates from port-1 to port-2 in primary line then:

Coupling factor (dB) =10 $\log_{10}$ (P1/P4)

Directivity (dB) = 10 $\log_{10}$ (P4/P3)

Where P1=power input to port-1

P3=power output from port-3 and P4=power output from port-4

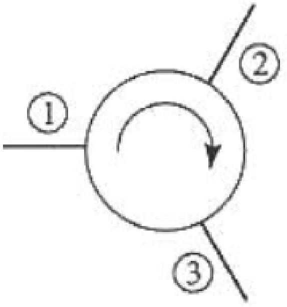
CIRCULATORS AND ISOLATORS:

- Both microwave circulators and microwave isolators are non-reciprocal transmission devices that use Faraday rotation in the ferrite material.

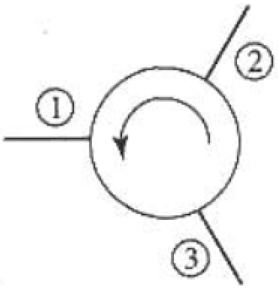
CIRCULATOR:

- A microwave circulator is a multiport waveguide junction in which the wave can flow only in one direction i.e. from the nth port to the (n+1)th port.
- It has no restriction on the number of ports
- 4-port microwave circulator is most common.
- One of its types is a combination of two 3-dB side hole directional couplers and a rectangular waveguide with two non reciprocal phase shifters.

Diagram:



$$[S]=\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}$$



$$[S]=\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}$$

- Each of the two 3db couplers introduce phase shift of 90 degrees
- Each of the two phase shifters produce a fixed phase change in a certain direction.
- Wave incident to port-1 splits into 2 components by coupler-1.
- The wave in primary guide arrives at port-2 with 180 degrees phase shift.
- The second wave propagates through two couplers and secondary guide and arrives at port-2 with a relative phase shift of 180 degrees.
- But at port-4 the wave travelling through primary guide phase shifter and coupler-2 arrives with 270 degrees phase change.
- Wave from coupler-1 and secondary guide arrives at port-4 with phase shift of 90 degrees.

- Power transmission from port-1 to port-4 =0 as the two waves reaching at port-4 are out of phase by 180 degrees.

$$\mathbf{w_1 - w_3 = (2m+1)\,\pi \text{ rad/s}}$$

$$\mathbf{w_2 - w_4 = 2n\pi \text{ rad/s}}$$

Power flow sequence: 1-> 2 -> 3 -> 4-> 1

MICROWAVE ISOLATOR:

- A non reciprocal transmission device used to isolate one component from reflections of other components in the transmission line.
- Ideally complete absorption of power takes place in one direction and lossless transmission is provided in the opposite direction
- Also called UNILINE, it is used to improve the frequency stability of microwave generators like klystrons and magnetrons in which reflections from the load affects the generated frequency.
- It can be made by terminating ports 3 and 4 of a 4-port circulator with matched loads.
- Additionally it can be made by inserting a ferrite rod along the axis of a rectangular waveguide.

REFLEX KLYSTRON

The reflex klystron is a single cavity variable frequency time-base generator of low power and load effiency

APPLICATION:

- It is widely used as in radar receiver
- Local oscillators in microwave receiver
- Portable microwave rings
- Pump oscillator in parametric amplifier

CONSTRUCTION:

- Reflex cavity klystron consists of an electron gun , filament surrounded by cathode and a floating electron at cathode potential
- Electron gun emits electron with constant velocity

$$\frac{1}{2}\mathrm{m}v^2=\mathrm{qVa}$$

$$\mathrm{V}=\sqrt{\frac{2qVa}{m}}\ \mathrm{m/s}$$

OPERATION:

- The electron that are emitted from cathode with constant velocity enter the cavity where the velocity of electrons is changed or modified depending upon the cavity voltage.
- The oscillations is started by the device due to high quality factor and to make it sustained we have to apply the feedback.

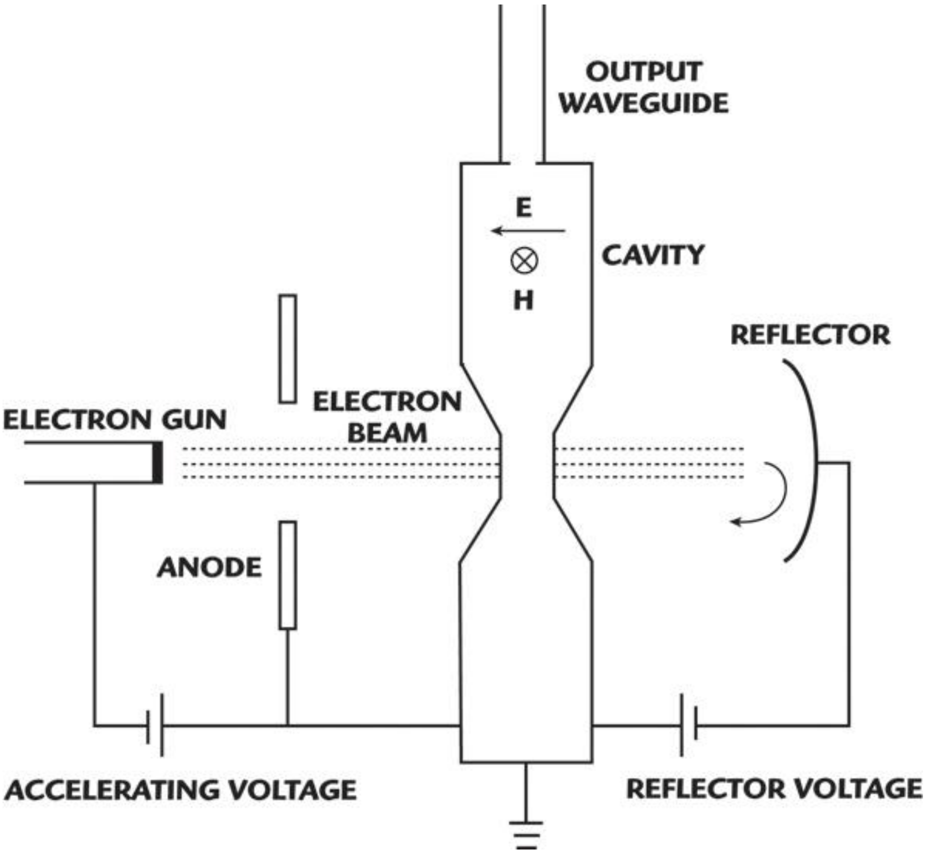
  Hence there are the electrons which will bunch together to deliver the energy act a time to the RF signal.

  - Inside the cavity velocity modulation takes place. Velocity modulation is the process in which the velocity of the emitted electrons are modified or change with respect to cavity voltage. The exit velocity or velocity of the electrons after the cavity is given as

$$\mathrm{V'}=\sqrt{\frac{2q(Va+Vi\sin(\omega t))}{m}}$$

  - In the cavity gap the electrons beams get velocity modulated and get bunched to the drift space existing between cavity and repellar.
  - Bunching is a process by which the electrons take the energy from the cavity at a different time and deliver to the cavity at the same time.

Diagram:



Bunching continuously takes place for every negative going half cycle and the most appropriate time for the electrons to return back to the cavity ,when the cavity has positive peak .So that it can give maximum retardation force to electron.

- It is found that when the electrons return to the cavity in the second positive peak that is 1 whole ¾ cycle.(n=1π).It is obtained max power and hence it is called dominant mode.

The electrons are emitted from cathode with constant anode voltage Va, hence the initial entrance velocity of electrons is

$$V=\sqrt{\frac{2qVa}{m}}\text{ m/s}$$

Inside the cavity the velocity is modulated by the cavity voltage Visin(ωt) as,

$$V= \sqrt{\frac{2q(Va+Vi\sin(\omega t))}{m}}$$

$$V= \sqrt{\frac{2qVa}{m}+\frac{2qVaKaVi\ \sin\omega t}{Vam}}$$

$$V= \sqrt{Vo^2+\frac{Vo^2KiVi\sin\omega t}{Va}}$$

$$V= \ Vo\sqrt{1+\frac{KiVi\ \sin\omega t}{Va}}$$

$$V= \ \sqrt{1+\frac{Vo^2iVi\sin\omega t}{Va}}$$

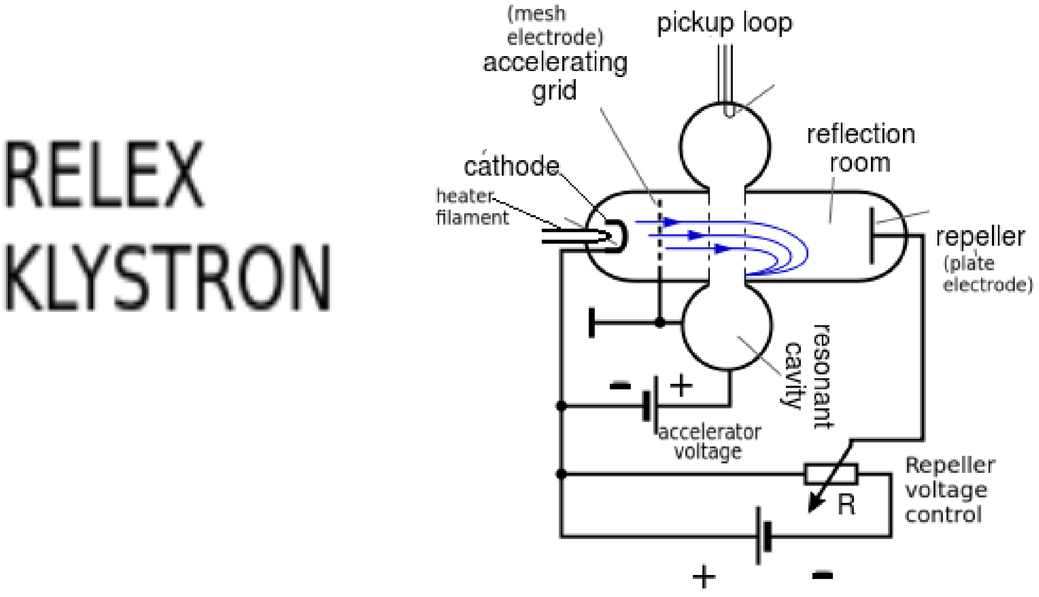
When, $\frac{kivi}{Va}$ =Depth of velocity

$$V{=}Vo\ (\ 1+\frac{Vo^2iVi\sin\omega t}{2Va})$$

TRANSIENT TIME:

Transit time is defined as the time spent by the electrons in the cavity space or, time taken by the electrons to leave the cavity and again return to the cavity.

(FIGURE)



If t_1 is the time at which electrons leave the cavity and t_2 is the time at which electrons bunch in the cavity then, transit time

$$t_{r=}t_1-t_2$$

During this time the net displacement by electrons is zero. That the potential of two point A and B is VA and VB(plate) as known in figure,then,

$$V_{AB}\ =\ V_A-V_B$$

$$=V_A+\ V_i\sin wt+\ VR$$

$$=V_A + VR + V_i \sin wt$$

Neglecting the AC compoment,

$$V_{AB} = V_a + V_R$$

$$E = \frac{\partial V_{AB}}{\partial x} = \frac{-V_{AB}}{S} = \frac{-(V_a+V_R)}{S} \quad \text{-------(a)}$$

The force experienced on an electron

$F=qe=\frac{-q(V_a+V_R)}{S}$

$F=\frac{m\,\partial^2 x}{\partial t^2}$ -------(b)

From equations a and b we get

$\frac{m\,\partial^2 x}{\partial t^2}=\frac{-q(V_a+V_R)}{S}$

Integrating both sides we get,

$\int \frac{m\,\partial^2 x}{\partial t^2} = \frac{-q(V_a+V_R)}{S}$

$\int m\,\partial x^2=\frac{-q(V_a+V_R)}{S} \begin{cases} t \\ t1 \end{cases} \partial t$

$m\frac{\partial x}{\partial t}=\frac{-q(V_a+V_R)}{S}\ {}_{t1}^{t}t$

$\frac{\partial x}{\partial t}=\frac{-q(V_a+V_R)}{mS}$ (t-t_1)+ k_1

$\therefore$ $k_1$is called velocity constant and assumed to velocity at (t-t_1)

$\frac{\partial x}{\partial t}=\frac{-q(V_a+V_R)}{mS}$ (t-t_1)+$V(t1)$

$$\int \partial x = \frac{-q(V_{a+}V_R)}{mS}\int (\mathrm{t}-t_1)\,\partial\, t + V(t1)$$

$X=\frac{-q(V_a+V_R)}{mS}\ {}_{t1}^{t}[\frac{t^2}{2}] - t_1\,{}_{t1}^{t}[t]+\int V(t1)\ \partial t$

$=\frac{-q(V_a+V_R)}{mS}\,[\left[\frac{t^2-t1^2}{2}\right] - (t_2 - t_2)t_1] +\int V(t1)\ \partial t$

$= \frac{-q(V_a+V_R)}{mS}\left[\frac{t^2-t1^2-2t_2t_1+2t_2{}^2}{2}\right]+\int V(t1)\ \partial t$

$=\frac{-q\,(V_a+V_R)}{mS}\left[\frac{t^2+t1^2-2t_2t_1}{2}\right]+\int V(t1)\ \partial t$

$=\frac{-q\,(V_a+V_R)}{mS}\left[\frac{(t-t1)^2}{2}\right]+\int V(t1)\ \partial t$

$=\frac{-q\,(V_a+V_R)}{mS}\left[\frac{(t-t1)^2}{2}\right]+\int V(t1)\ (t-t1)+k_2$

Where,k_2 is displacement constant at $t=t_2,x=0$ .In practice k_2=the cavity width which is negligible with respect to cavity space s.Here we can neglect k_2 in the expression of x

At $t=t_2,x=0$

$0=\frac{-q\,(V_a+V_R)}{mS}\left[\frac{(t-t1)^2}{2}\right]+V(t1)\,(t-t1)$

- ➢ $\frac{-q\,(V_a+V_R)}{mS}\left[\frac{(t_2-t1)^2}{2}\right] = V(t1)\,(t_2-t1)$
- ➢ $(t_2-t1)=\frac{2V(t1)ms}{q\,(V_{a+}V_R)}$

TRANSIT ANGLE:

$$\omega t_r = \omega(t_2-t1) = \omega\frac{2V(t1)ms}{q(V_{a+}V_R)}$$

$$\omega t_r = \omega\frac{2V(t1)ms}{q(V_{a+}V_R)}$$

We know,

n=+3/4, for n=0,1,2,3,4…..

n-1/4. for n=0,1,2,3,4…..

$2\times\pi(t_2-t1)=\left(n-\frac{1}{4}\right)2\pi\times T$

$2\times\pi\times f(t_2-t1)=\left(2n\pi-\frac{\pi}{2}\right)$

$$\omega t_r = \left(2n\pi-\frac{\pi}{2}\right) = \omega\frac{2V(t1)ms}{q(V_{a+}V_R)}$$

OUTPUT POWER:

The beam current of Reflex klystron is given as

$Ib=I_O+\sum_{n=1}^{\infty}(2I_0I_n(x')\cos(n\omega t-\varphi)$

$I_0$is dc current due to cavity voltage is given by

$P_{dc=}V_aI_o$ -------(1)

The ac component of the current is given by

$I_{ac}=\sum_{n=1}^{\infty}(2I_0 I_n(x') \cos(n\omega t-\varphi)$

For (n=1),we have fundamental current component ie,

If $(2I_0 I_1(x') \cos(n\omega t-\varphi)$

For n=2; $\cos(n\omega t-\varphi)=1$

$I_2=2I_o K_i J_i(X_i)$

Power$=\frac{v_i I_2}{2}=\frac{v_i}{2}(2I_o k_i Ji(Xi))$

$$\omega(t_2-t1)=\ \omega\frac{2V(t1)ms}{q(V_{a+}V_R)}$$

$\omega t_2=\omega t_1+\frac{2V_0 ms\omega}{q(V_{a+}V_R)}[1+\frac{k_i v_i}{2v_a}\sin\omega t_1]$

$\therefore[\frac{2V_0 ms\omega}{q(V_{a+}V_R)}]=\alpha$

→ $\omega t_2=\omega t_1+\alpha[1+\frac{k_i v_i}{2v_a}\sin\omega t_1]$

→ $\omega t_2=\omega t_1+\alpha+\frac{k_i v_i\alpha}{2v_a}\sin\omega t_1]$

Where , $\frac{k_i v_i\alpha}{2v_a}$=x, bunching parameter

$V_i=\frac{2v_a x}{\alpha k_i}=\frac{2v_a x}{(2n\pi-\frac{\pi}{2})k_i}$

$P_{o/p}=\frac{2v_a x I_0\times J_i(x)}{(2n\pi-\frac{\pi}{2})}$

Efficiency: $\eta\%=\frac{P_{output}}{P_{input}}\times 100\%$

$=\frac{2v_a x I_0\times J_i(x)}{(2n\pi-\frac{\pi}{2})v_a x I_0}$

$=\frac{2\times J_i(x)}{(2n\pi-\frac{\pi}{2})}\times 100\%$

Electronics admittance of reflex klystron

It is defined as the ratio of current induced in the cavity by the modulation of electron beam to the voltage across the cavity gap.

$Y_e=\frac{I_2}{v_2}$,

- $I_2=2I_o k_i J_i(x')\cos(\omega t-\varphi)$
- $\quad =2I_o k_i J_i e^{-j\varphi}$
- $V_{2=}V_1 e^{-\pi/2}=\frac{2\ v_{ax'}}{\alpha k_i}e^{-j\pi/2}$
- $Y_e=\frac{\alpha Io ki^2}{v_a}\frac{J_{i(x)'}\sin\ \varphi}{x'}+j\frac{\alpha Io ki^2}{v_a}\frac{J_{i(x)'}\cos\ \varphi}{x'}$

$Y_e=G_e+j\beta_e$

MAGNETRON

DIFFERENCE BETWEEN REFLEX KLYSTRON AND MAGNETRON:

REFLEX KLYSTRON	MAGNETRON
❖ It is a linear tube in which the magnetic field is applied to focus the electron and electric field is applied to drift the electron. ❖ In klystron the bunching takes places only inside the cavity which is very small ,hence generate low power and low frequency.	❖ In magnetron the magnetic field and electric field are perpendicular to each other hence it is called as cross field device. ❖ In magnetron the interacting or bunching space is extended so the efficiency can be increase.

APPLICATION:

- Used as oscillator.
- Used in radar communication.
- Used in missiles.
- Used in microwave oven (in the range of frequency of 2.5Ghz).

Types of magnetron:

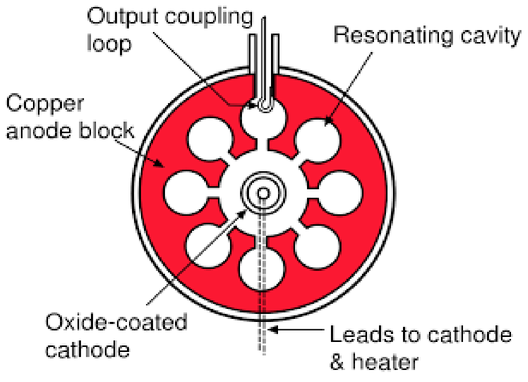
Magnetron is of 3 types:

- ✓ Negative resistance type.
- ✓ Cyclotron frequency type.
- ✓ Cavity type.

Construction of cavity magnetron:

Cavity type magnetron depends upon the interaction of electron with a rotating magnetic field with constant angular velocity.

FIGURE:



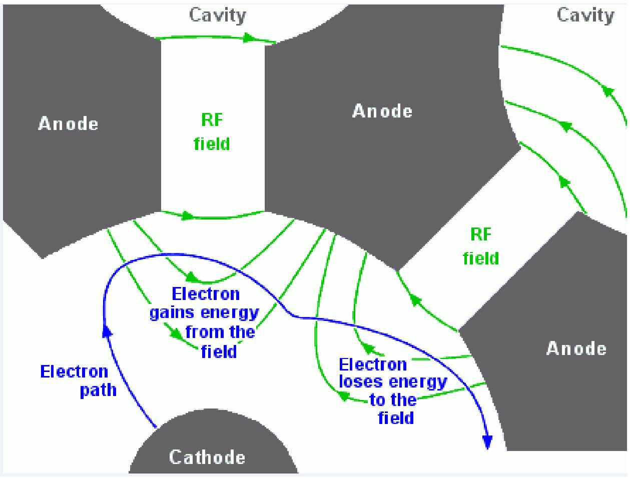
Resonant cavity magnetron high-power high-frequency oscillator

A magnetron consists of a cathode which is used to emit electrons and a number of anode cavities a permanent magnet is placed on the backside of cathode. The space between anode cavity and cathode is called interacting space. The electron which are emitted from cathode moves in different path in the interacting space depending upon the strength of electron and magnetic field applied to the magnetron.

OPERATION:

EFFECT OF ELECTRIC FIELD ONLY:

- In the absence of magnetic field(B=0) the electron travel straight from the cathode to the anode due to the radial electric field force acting on it(indicated by path A).
- If the magnetic field strength increases slightly it will exert a lateral force which bends the path of the as indicated in path B.



- The radius of the path is given as

$$r=\frac{mv}{eB}$$

where v=velocity of electron

B=magnetic field strength

- If from reaching the anode current become zero(indicated by path D).the strength of magnetic field is made sufficiently high enough, so to prevent the electron
- The magnetic field required to return the electron back to the cathode just touching the surface of anode is called critical magnetic field or cut off magnetic field(Bc).
- If B>Bc the electron experiences a grater rotational force and may return back to the cathode quite faster this results is heating of cathode.

Effect of magnetic field:

The force experience by the electron because of magnetic field only.

$$F= q(v\times B)$$

= qvBsinθ

- For maximum force θ=90°

  Fmax = qvB
- And hence the electron which are emitted, moved in a right angle with respect to force.
- If the magnetic field strength is sufficiently large enough, then the electrons emitted will return back to the cathode with high velocity which may destroy the cathode this effect is called Back heating of cathode.

π MODE OF OSCILLATION:

- The shape consisting of oscillation can maintain if the phase difference between anode cavity is $\frac{n\pi}{4}$ where n is the mode of operation and the best result can be obtained for n=4 => $\frac{n\pi}{4}$=π (for n=4 hence it is called π mode operation).
- It is assume that each anode cavity is of $\frac{\lambda}{4}$ length, hence a voltage antinodes will exist at the opening of anode cavity and the lines of forces present due to the oscillation started by high quality factor device.
- In the above figure the electron followed by path 'b' is so emitted that is not influenced by the electric lines of forces hence it will spend very less time inside the cavity and doesn't contribute to the oscillation so it is called unfavourable electron.
- The electron followed by path a is so emitted that is influence by the electric lines of forces at position 1,2&3 respectively where the velocity increases or decreases, hence more time spend inside the cavity therefore it is called favourable electron.
- Any favourable electron which are emitted earlier or later with respect to reference electron (let a) may be bunch together due to change in velocity by the effect of electric lines of forces. This type of bunching is called phase focusing effect.
- Electrons emitted from the cathode may rotate around itself in a confined area in a shape of spoke (spiral) at a angular velocity and before delivering the energy to the anode cavity. They will rotate until they reach the anode and completely absorbed by them. Hence the magnetron are also called travelling wave magnetron.

CUT OFF MAGNETIC FIELD (B_C):

- Assume a cylindrical magnetron whose inner radius is 'a' and outer radius is 'b' and the magnetic field is as shown in the figure. Under the effect of magnetic field the electrons will rotate in a circular path at any point the force electron will be balance by the centrifugal force.

$$\frac{mv^2}{r} = qvB_z$$

$$\Rightarrow v = \frac{q\mathrm{Bz}}{m}r$$

$$\Rightarrow \frac{v}{r} = \frac{q\mathrm{Bz}}{m}$$

$$\Rightarrow \omega = \frac{q\mathrm{Bz}}{m}$$

$$\Rightarrow f = \frac{1}{2\pi}(\frac{q\mathrm{Bz}}{m})$$

In cylindrical coordinate system the equation of motion is given as

$$=> \frac{1}{\partial}\frac{d}{dt}(\gamma^2\frac{d\theta}{dt})=\omega\frac{dr}{dt}$$

$$=>\int \frac{d}{dt}(r^2\frac{d\theta}{dt})=w\int \frac{dr}{dt}.r$$

$$=> r^2 (\frac{d\theta}{dt}) =\frac{\omega r^2}{2}+k$$

The constant k can be calculated using boundary condition.

At r=a,$\frac{d\theta}{dt}$=0

⇨ $0=\frac{\omega a^2}{2}+k$

⇨ $K=-\frac{\omega a^2}{2}$............(1)

The value of k in equation (1)

⇨ $r^2\frac{d\theta}{dt}=\frac{\omega}{2}(r^2-a^2)$...............(2)

At r=b the equation (2) becomes

⇨ $b^2\frac{d\theta}{dt}=\frac{\omega}{2}(b^2-a^2)$

⇨ $b\frac{d\theta}{dt}=\frac{\omega}{2b}(b^2-a^2)$................(3)

Due to electric field only, the electrons move radially from cathode to anode.

⇨ $\frac{1}{2}mv^2=q\,V_{\mathrm{dc}}$

⇨ $v=\sqrt{\frac{2qV_{\mathrm{dc}}}{m}}$

But inside the cavity the velocity of electron has two components

⇨ $v^2=v_r^2+v\theta^2$

⇨ $\frac{2qV_{\mathrm{dc}}}{m}=(\frac{dr}{dt})^2+(r\frac{d\theta}{dt})$

At r=b, $\frac{dr}{dt}$=0

⇨ $\frac{2qV_{\mathrm{dc}}}{m}=0+(b\frac{d\theta}{dt})^2$

⇨ $b\frac{d\theta}{dt}=\sqrt{\frac{2qV_{\mathrm{dc}}}{m}}$..............(4)

Comparing equation (3) & (4)

$$\frac{\omega}{2b}(b^2-a^2)=\sqrt{\frac{2qV_{\mathrm{dc}}}{m}}$$

⇨ $\frac{1}{2b}\frac{qB_z}{m}(b^2-a^2)=\sqrt{\frac{2qV_{\mathrm{dc}}}{m}}$(at r=b ,$B_z = B_c$)

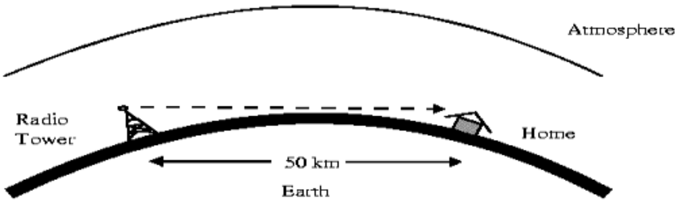
$$B_c = \frac{\frac{8mVdc}{b(1-{a^2}/{b^2})}}{b(1-{a^2}/{b^2})}$$

$$V_{dc} = \frac{q}{8m} b^2 \left(1 - {a^2}/{b^2}\right) B_z^2$$

MICROWAVE PROPAGATION

Line of Sight (LOS) Propagation:

Analog microwave communication system include LOS and over the horizon (OTH) analog communication systems. LOS communication is limited by horizon due to earth's curvature i.e. the transmission distance is determined by the height of the antenna above the earth in view of the horizon limitation. Microwaves are normally bent or refracted beyond the optical horizon (horizon visible to eyes). However due to atmospheric refractive changes, the radio horizon could even be less than the optical horizon at times.



The height of the microwave tower should be such that the radio beam is not obstructed by high rise buildings, trees or mountains. LOS microwave systems have shorter installation time, high flexible channel capacity and better adaptation to difficult terrains and natural barriers. LOS microwave system operates in the 14Hz to 10GHz frequency range. Above 10GHz however absorption due to rain, fog or snow may affect the system performance. Above 20GHz it is the absorption due to water vapour and atmospheric oxygen that limits the performance of an LOS system. Frequency Division Multiplexing (FDM) is used in all analog microwave LOS systems that enables several telephone signals to be transmitted over the same carrier. Normally the 4 KHz telephone channels are grouped as a 12 channel basic group in the frequency band 60KHz to 108KHz. Further , 5 basic groups form a 60-channel super group in the frequency band from 312KHz to 552KHz. 16 super groups are clubbed together to obtain a 960 channel master group.

ABSORPTION OF MICROWAVES BY ATMOSPHERIC GASES:

The transfer of electromagnetic radiation through an atmosphere is linked to its state (temperature, pressure, and composition) by the refractive index and by coefficient of absorption and scattering, if any. The absorption coefficient in a medium is a macroscopic parameter that represents the interaction of incident electromagnetic energy with the constituent molecules. The interaction is governed by three general principles.

1. Bohr's Frequency Condition:
   The frequency γ of a photon emitted or absorbed by the gas is equal to the difference of two energy levels (E_a-E_b) of the gas, divided by Plank's constant *h*.

2. Einstein's laws of emission and absorption:
   If E_a is higher than E_b, the probability, given initial state a, of stimulated emission of a photon by a transition from state a to state b is equal to the probability, given initial state b, of absorption of a photon by a transition from b to a. Hence the net absorption is proportional to the difference in thermodynamic probabilities (P_a-P_b) of two states.

3. Dirac's Perturbation Theory:
   For the electromagnetic field to include transitions between states a and b, the operator with which the field interacts must have a non-zero matrix element linking the two states. For wavelengths that are very long compared to molecular dimensions, this operator is the dipole moment.

ABSORPTION OF MICROWAVES BY WATER VAPOUR AND PRECIPITATES:

Water vapor and fog are the most influencing parameters when microwave propagates through the atmosphere. At high frequencies above about 10GHz, electromagnetic radiation starts interacting with the neutral atmosphere and also with the various metrological parameters, in particular, precipitation, producing absorption of energy and thus attenuation of signal levels. In addition to absorption by molecular oxygen, molecules of water vapor also interact with electromagnetic radiation in the microwave and millimeter wave regions. The water vapor molecule being a permanent electric dipole produces rotational transitions of the order of 10^4 times stronger than that of the magnetic transitions of the oxygen molecule. So, even though the abundance of water vapor in the atmosphere is considerably less than the oxygen, it can produce significant level of attenuation near the resonant frequencies. Water vapor attenuation is found to depend quadratically on water vapor density, particularly at high densities, above about $12g/m^3$.

SATTELITE SYSTEM

CHAPTER 1

1.1 Introduction:-

Satellites offer a number of features not readily available with other means of communications. Because very large areas of the earth are visible from a satellite, the satellite can form the star point of a communications net, simultaneously linking many users who may be widely separated geographically. The same feature enables satellites to provide communications links to remote communities in sparsely populated areas that are difficult to access by other means. Of course, satellite signals ignore political boundaries as well as geographic ones, which may or may not be a desirable feature.

Satellites are also used for remote sensing, examples being the detection of water pollution and the monitoring and reporting of2 Chapter One weather conditions. Some of these remote sensing satellites also form a vital link in search and rescue operations for downed aircraft and the like. Satellites are specifically made for telecommunication purpose. They are used for mobile applications such as communication to ships, vehicles, planes, hand-held terminals and for TV and radio broadcasting. They are responsible for providing these services to an assigned region (area) on the earth. The power and bandwidth of these satellites depend upon the preferred size of the footprint, complexity of the traffic control protocol schemes and the cost of ground stations. A satellite works most efficiently when the transmissions are focused with a desired area. When the area is focused, then the emissions do not go outside that designated area and thus minimizing the interference to the other systems. This leads more efficient spectrum usage.

Satellite’s antenna patterns play an important role and must be designed to best cover the designated geographical area (which is generally irregular in shape). Satellites should be designed by keeping in mind its usability for short and long term effects throughout its life time. The earth station should be in a position to control the satellite if it drifts from its orbit it is subjected to any kind of drag from the external forces.

1.2 History of Satellite Communications

The first artificial satellite used solely to further advances in global communications was a balloon named Echo 1. Echo 1 was the world's first artificial communications satellite capable of relaying signals to other points on Earth. The first American satellite to relay communications was Project SCORE in 1958, which used a tape recorder to store and forward voice messages. It was used to send a Christmas greeting to the world from U.S. President Dwight D. Eisenhower. NASA launched the Echo satellite in 1960; the 100-foot (30 m) aluminised PET film balloon served as a passive reflector for radio communications. Courier 1B, built by Philco, also launched in 1960, was the world's first active repeater satellite. The first communications satellite was Sputnik 1. Put into orbit by the Soviet Union on October 4, 1957, it was equipped with an onboard radio-transmitter that worked on two frequencies: 20.005 and 40.002 MHz. Sputnik 1 was launched as a step in the exploration of space and rocket development. While incredibly important it was not placed in orbit for the purpose of sending data from one point on earth to another. And it was the first artificial satellite in the steps leading to today's satellite communications. Telstar was the second active, direct relay communications satellite. Belonging to AT&T as part of a multi-national agreement between AT&T, Bell Telephone Laboratories, NASA, the British General Post Office, and the French National PTT (Post Office) to develop satellite communications, it was launched by NASA from Cape Canaveral on July 10, 1962, the first privately sponsored space launch. Relay 1 was launched on December 13, 1962, and became the first satellite to broadcast across the Pacific on November 22, 1963.

CHAPTER 2: ORBITAL MECHANICS

Satellites (spacecraft) orbiting the earth follow the same laws that govern the motion of the planets around the sun. From early times much has been learned about planetary motion through careful observations. Johannes Kepler (1571–1630) was able to derive empirically three laws describing planetary motion. Later, in 1665, Sir Isaac Newton (1642–1727) derived Kepler's laws from his own laws of mechanics and developed the theory of gravitation.

2.1 Kepler's Laws of Planetary Motion

Kepler's First Law:- *Kepler's first law* states that the path followed by a satellite around the primary will be an ellipse. An ellipse has two focal points shown as *F1* and *F2* in Fig.1.

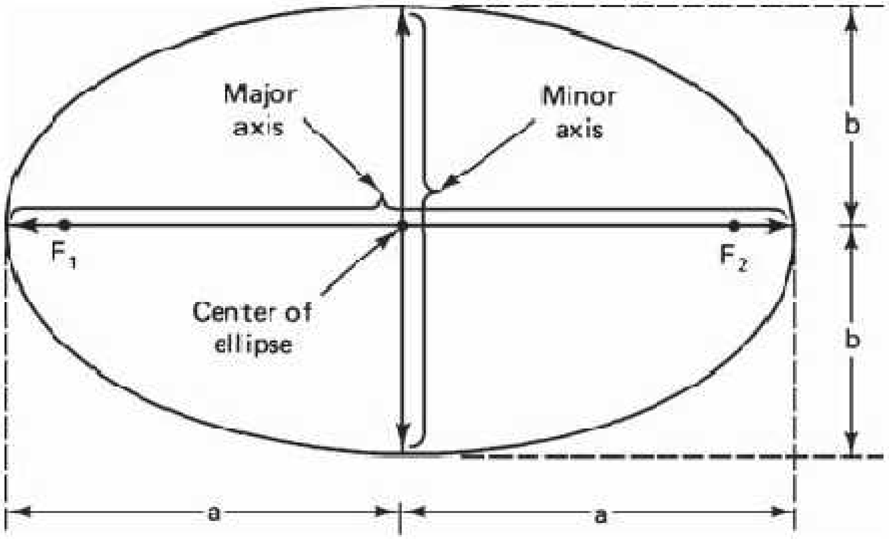


Fig.2.1 The foci *F1* and *F2*, the semimajor axis *a*, and the semiminor axis *b* of an ellips

The foci *F* The eccentricity and the semimajor axis are two of the orbital parameters specified for satellites (spacecraft) orbiting the earth. For an elliptical orbit, 0<*e*< 1. When *e* = 0, the orbit becomes circular.

Kepler's Second Law:- *Kepler's second law* states that, for equal time intervals, a satellite will sweep out equal areas in its orbital plane, focused at the barycenter. The center of mass of the two-body system, termed the *barycenter*, is always centered on one of the foci.

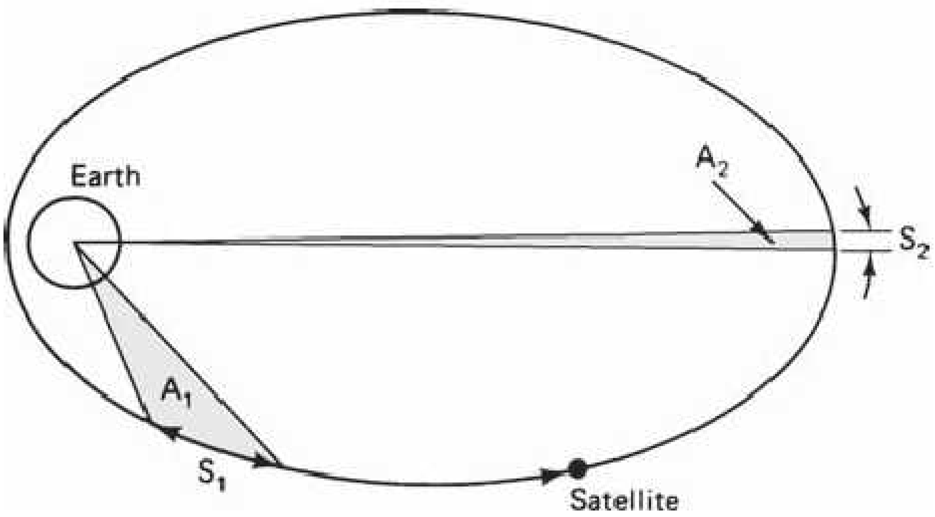


Figure 2.2. Kepler's second law. The areas *A*1 and *A*2 swept out in unit time are equal.

Kepler's Third Law:- *Kepler's third law* states that the square of the periodic time of orbit is proportional to the cube of the mean distance between the two bodies. The mean distance is equal to the semimajor axis *a*. For the
artificial satellites orbiting the earth, Kepler's third law can be written as follows

$$a^3 = \frac{«}{n^2} \qquad \dots (1)$$

where *n* is the mean motion of the satellite in radians per second and *μ* is the earth's geocentric gravitational constant.

$$« = 3.986005 \times 10^{14} m^3 / s^3 \qquad \dots (2)$$

The importance of Kepler's third law is that it shows there is a fixed relationship between period and semimajor axis.

2.2 Satellite Orbits

There are many different satellite orbits that can be used. The ones that receive the most attention are the geostationary orbit used as they are stationary above a particular point on the Earth. The orbit that is chosen for a satellite depends upon its application. These orbits are given in table 1.

Geostationary or geosynchronous earth orbit (GEO)

A satellite in a geostationary orbit appears to be stationary with respect to the earth, hence the name *geostationary*. GEO satellites are synchronous with respect to earth. Looking from a fixed point from Earth, these satellites appear to be stationary. These satellites are placed in the space in such a way that only three satellites are sufficient to provide connection throughout the surface of the Earth. GEO satellite travels eastward at the same rotational speed as the earth in circular orbit with zero inclination.

A geostationary orbit is useful for communications because ground antennas can be aimed at the satellite without their having to track the satellite's motion. This is relatively inexpensive. In applications that require a large number of ground antennas, such as DirectTVdistribution, the savings in ground equipment can more than outweigh the cost and complexity of placing a satellite into orbit.

Table: 1

STELLITE ORBIT NAME	ORBIT	SATELLITE ORBIT ALTITUDE (KM ABOVE EARTH'S SURFACE)	APPLICATION
Low Earth Orbit	LEO	200 - 1200	Satellite phones, Navstar or Global Positioning (GPS) system
Medium Earth Orbit	MEO	1200 - 35790	High-speed telephone signals
Geosynchronous Orbit	GSO	35790	Satellite Television
Geostationary Orbit	GEO	35790	Direct broadcast television

Low Earth Orbit (LEO) satellites

A low Earth orbit (LEO) typically is a circular orbit about 200 kilometres (120 mi) above the earth's surface and, correspondingly, a period (time to revolve around the earth) of about 90 minutes. Because of their low altitude, these satellites are only visible from within a radius of roughly 1000 kilometers from the sub-satellite point. In addition, satellites in low earth orbit change their position relative to the ground position quickly. So even for local applications, a large number of satellites are needed if the mission requires uninterrupted connectivity. s. LEO systems try to ensure a high elevation for every spot on earth to provide a high quality communication link. Each LEO satellite will only be visible from the earth for around ten minutes.

Low-Earth-orbiting satellites are less expensive to launch into orbit than geostationary satellites and, due to proximity to the ground, do not require as high signal strength (Recall that signal strength falls off as the square of the distance from the source, so the effect is dramatic). Thus there is a trade off between the number of satellites and their cost. In addition, there are important differences in the onboard and ground equipment needed to support the two types of missions. One general problem of LEOs is the short lifetime of about five to eight years due to atmospheric drag and radiation from the inner Van Allen belt1.

Medium Earth Orbit (MEO) satellites

A MEO satellite is in orbit somewhere between 8,000 km and 18,000 km above the earth's surface. MEO satellites are similar to LEO satellites in functionality. MEO satellites are visible for much longer periods of time than LEO satellites, usually between 2 to 8 hours. MEO satellites have a larger coverage area than LEO satellites. A MEO satellite's longer duration of visibility and wider footprint means fewer satellites are needed in a MEO network than a LEO network. One disadvantage is that a MEO satellite's distance gives it a longer time delay and weaker signal than a LEO satellite, though not as bad as a GEO satellite. Due to the larger distance to the earth, delay increases to about 70–80 ms. so these satellites need higher transmit power and special antennas for smaller footprints.

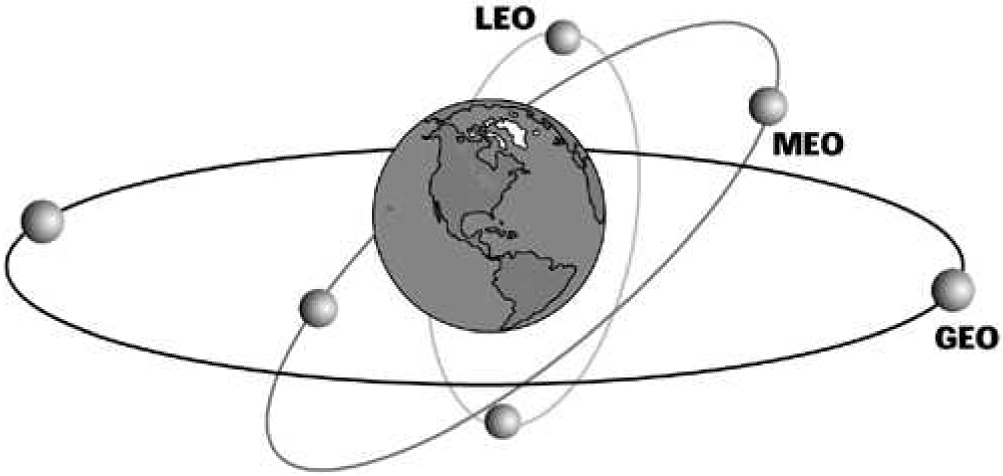


Fig. 2.3 Satellite Orbits

2.3 Spacing and Frequency Allocation

Allocating frequencies to satellite services is a complicated process which requires international coordination and planning. This is carried out under the supervision of the *International Telecommunication Union* (ITU). This frequency allocation is done based on different areas. So this world is divided into three areas.

Area 1:- : Europe, Africa, Soviet Union, and Mongolia

Area 2: North and South America and Greenland

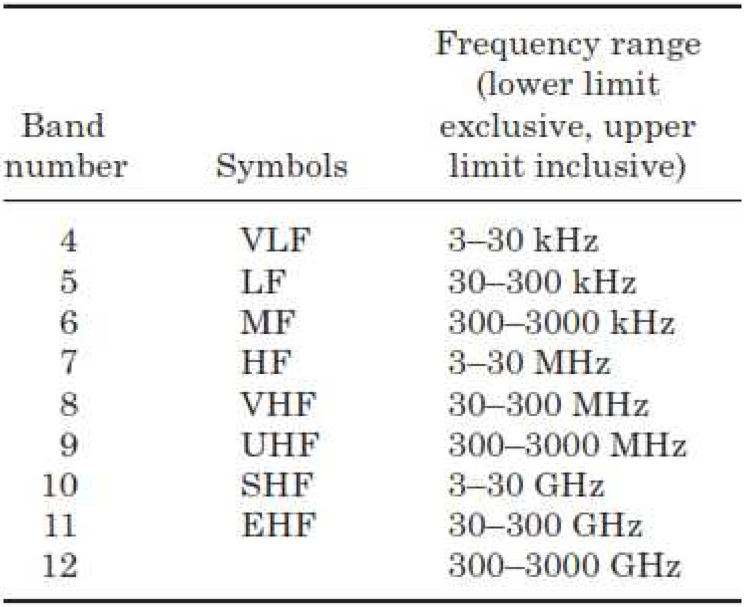
Area 3: Asia (excluding area 1 areas), Australia, and the south-west Pacific

Within these regions, frequency bands are allocated to various satellite services, although a given service may be allocated different frequency bands in different regions. Some of the services provided by satellites are:

- ***Fixed satellite service* (FSS)**

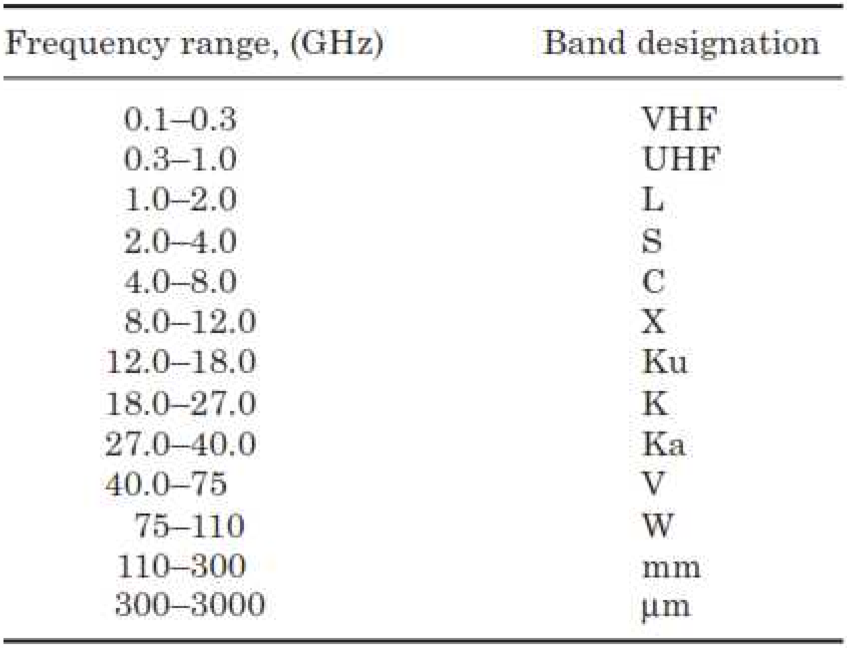
  The FSS provides links for existing telephone networks as well as for transmitting television signals to cable companies for distribution over cable systems. Broadcasting satellite services are intended mainly for direct broadcast to the home, sometimes referred to as *direct broadcast satellite* (DBS) service [in Europe it may be known as *direct-to-home* (DTH) service]. Mobile satellite services would include land mobile, maritime mobile, and aeronautical mobile. Navigational satellite services include *global positioning systems* (GPS), and satellites intended for the meteorological services often provide a search and rescue service.

TABLE 2: ITU Frequency Band Designations



Band number	Symbols	Frequency range (lower limit exclusive, upper limit inclusive)
4	VLF	3–30 kHz
5	LF	30–300 kHz
6	MF	300–3000 kHz
7	HF	3–30 MHz
8	VHF	30–300 MHz
9	UHF	300–3000 MHz
10	SHF	3–30 GHz
11	EHF	30–300 GHz
12		300–3000 GHz

TABLE 3: Frequency Band Designations



Frequency range, (GHz)	Band designation
0.1–0.3	VHF
0.3–1.0	UHF
1.0–2.0	L
2.0–4.0	S
4.0–8.0	C
8.0–12.0	X
12.0–18.0	Ku
18.0–27.0	K
27.0–40.0	Ka
40.0–75	V
75–110	W
110–300	mm
300–3000	µm

- ***Broadcasting satellite service*** **(BSS)**

  Provides Direct Broadcast to homes. E.g. Live Cricket matches etc.

- ***Mobile satellite services***
  - Land Mobile
  - Maritime Mobile
  - Aeronautical mobile
- ***Navigational satellite services***
  - Include Global Positioning systems
- ***Meteorological satellite services***
  - They are often used to perform Search and Rescue service.

2.4 Look Angle Determination

The satellite look angle refers to the angle that one would look for a satellite at a given time from a specified position on the Earth. The look angles for the ground station antenna are Azimuth and Elevation angles. They are required at the antenna so that it points directly at the satellite. Look angles are calculated by considering the elliptical orbit. These angles change in order to track the satellite.

Azimuth angle:- The azimuth angle is an angle measured from North direction in the local horizontal plane.

Elevation angle:- The elevation angle is the angle measured perpendicular to the horizontal plane (in the vertical plane) to the line-of-sight to the satellite.

The three pieces of information that are needed to determine the look angles for the geostationary orbit are

1. The earth-station latitude, denoted here by ª$_{E}$

2. The earth-station longitude, denoted here by ¤$_{E}$

3. The longitude of the subsatellite point, denoted here by ¤$_{SS}$ (this is just referred to as the satellite longitude)

4. ES: Position of Earth Station

5. SS: Sub-Satellite Point

6. S: Satellite

7. d: Range from ES to S

8. σ: angle to be determined

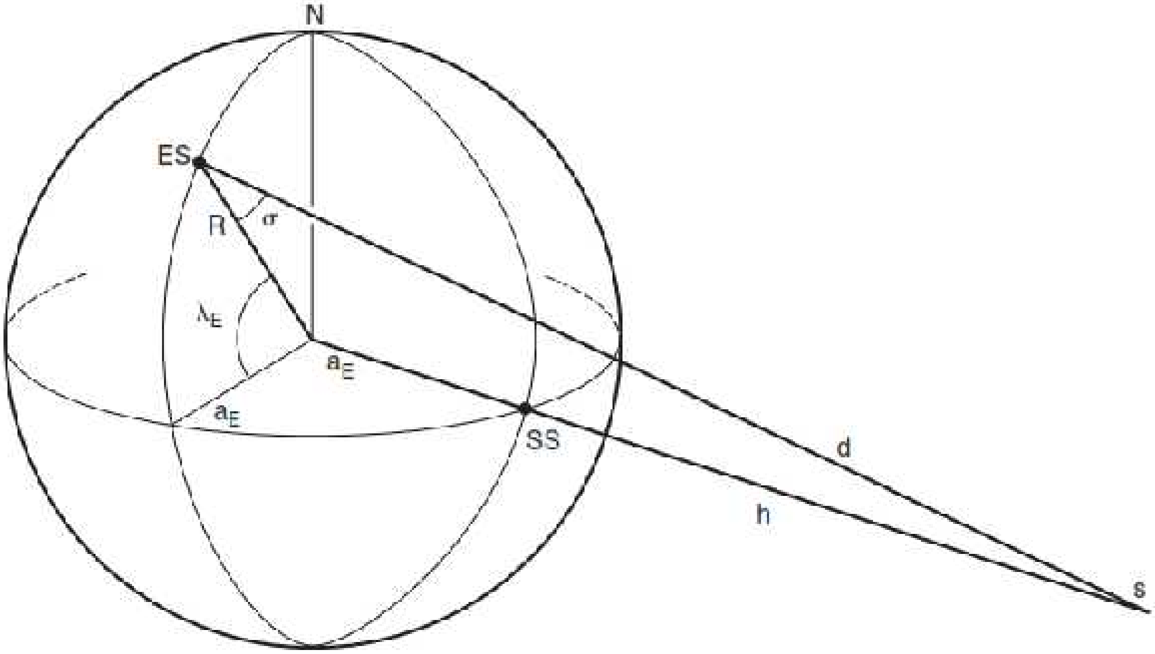


Fig. 2.4:- The geometry used in determining the look angles for a geostationary satellite.

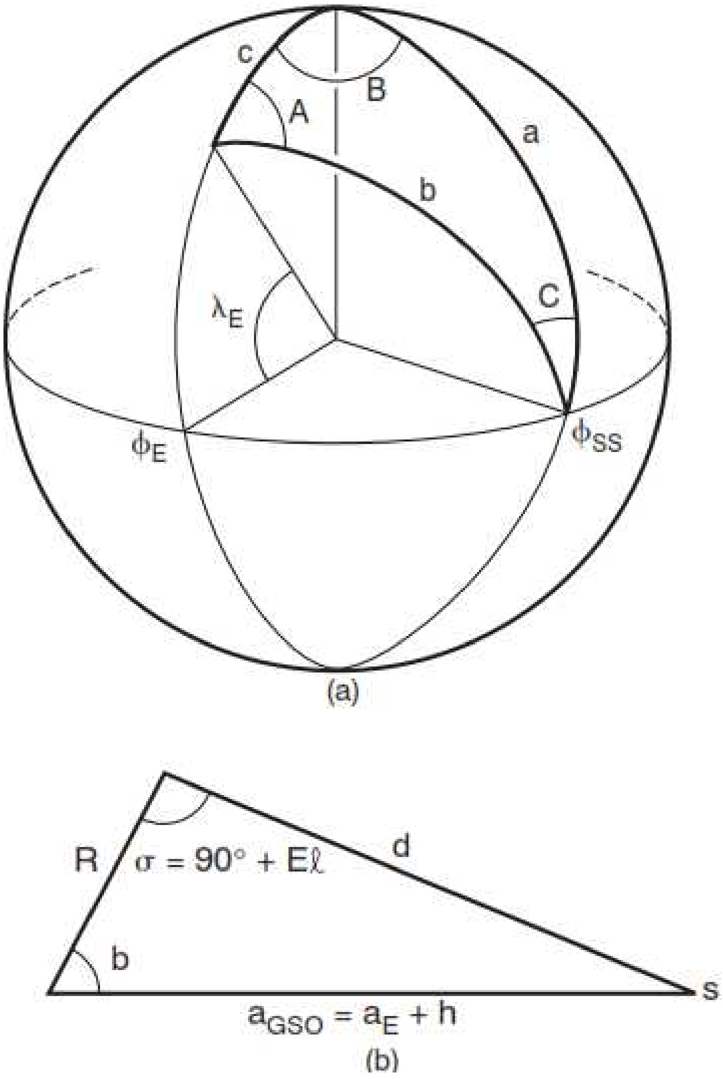


Figure 4.5 (*a*) The spherical geometry related to Fig. 4.4. (*b*) The plane triangle obtained from Fig. 4.4.

There are six angles in all defining the spherical triangle. The three angles *A, B,*and *C* are the angles between the planes. Angle *A* is the angle between the plane containing *c* and the plane containing *b.* Angle *B* is the angle between the plane containing *c* and the plane containing *a*.

Considering figure 5 (b), it's a spherical triangle. All sides are the arcs of a great circle. Three sides of this triangle are defined by the angles subtended by the centre of the earth.

Side a: angle between North Pole and radius of the sub-satellite point.

Side b: angle between radius of Earth and radius of the sub-satellite point.

Side c: angle between radius of Earth and the North Pole.

a = 90° and such a spherical triangle is called quadrantal triangle. c = 90° – λ

Angle B is the angle between the plane containing c and the plane containing a.

Thus, $B = ¤_E - ¤_{SS}$

Angle A is the angle between the plane containing b and the plane containing c.

Angle C is the angle between the plane containing a and the plane containing b.

Thus,

$a=90^0$

$c=90^0 -\lambda_E$

$B=f_E -f_{SS}$

Thus, b = arcos (cos B cos a_E

A = arc sin (sin |B| / sin b)

2.5 Orbital Perturbation

The *keplerian orbit* described so far is ideal in the sense that it assumes that the earth is a uniform spherical mass and that the only force acting is the centrifugal force resulting from satellite motion balancing the gravitational pull of the earth. In practice, other forces which can be significant are the gravitational forces of the sun and the moon and atmospheric drag. The gravitational pulls of sun and moon have negligible effect on low-orbiting satellites, but they do affect satellites in the geostationary orbit.

There are two types of perturbation:-

1- **Gravitational:- when considering third body interaction and the non-spherical shape of the earth.**

   The earth is very far away from perfectly spherical. This depends on the earth rotation, earth gravitational potential.

2- **Non-gravitational:- like Atmospheric drag, solar radiation pressure and tidal friction.**

For near-earth satellites, below about 1000 km, the effects of atmospheric drag are significant. Because the drag is greatest at the perigee, the drag acts to reduce the velocity at this point, with the result that the satellite does not reach the same apogee height on successive revolutions.

CHAPTER: 3 SATELLITES

3.1 Satellite Launching

A satellite is sent into space on top of a rocket. When a satellite is put into space, we say that it is "launched." The rocket that is used to launch a satellite is called a "launch vehicle." This satellite launching needs the earth stations in order to operate the satellite operation. The satellite launching can be divided into four stages.

1- **First Stage:-** The first stage of the launch vehicle contains the rockets and fuel that are needed to lift the satellite and launch vehicle off the ground and into the sky.

2- **Second Stage:-** The second stage contains smaller rockets that ignite after the first stage is finished. The rockets of the second stage have their own fuel tanks. The second stage is used to send the satellite into space.

3- **Third Stage (Upper Stage):-** The upper stage of the launch vehicle is connected to the satellite itself, which is enclosed in a metal shield, called a "fairing." The fairing protects the satellite while it is being launched and makes it easier for the launch vehicle to travel through the resistance of the Earth's atmosphere.

4- **Fourth Stage (Firing):-** Once the launch vehicle is out of the Earth's atmosphere, the satellite separates from the upper stage. The satellite is then sent into a "transfer orbit" that sends the satellite higher into space. Once the satellite reaches its desired orbital height, it unfurls its solar panels and communication antennas, which had been stored away during the flight. The satellite then takes its place in orbit with other satellites and is ready to provide communications to the public.

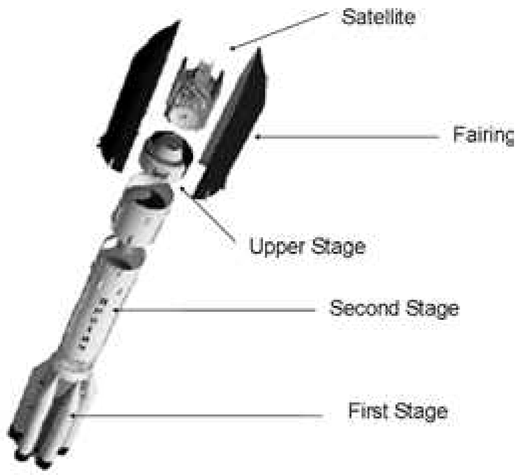


Figure 3.1 Steps of Satellite Launching

The launch process can be divided into two phases: the launch phase and the orbit injection phase.

1- **The Launch Phase**

   The launch vehicle places the satellite into the transfer orbit. An eliptical orbit that has at its farthest point from earth (apogee) the geosynchronous elevation of 22,238 miles and at its nearest point (perigee) an elevation of usually not less than 100 miles.

2- **The Orbit Injection Phase**

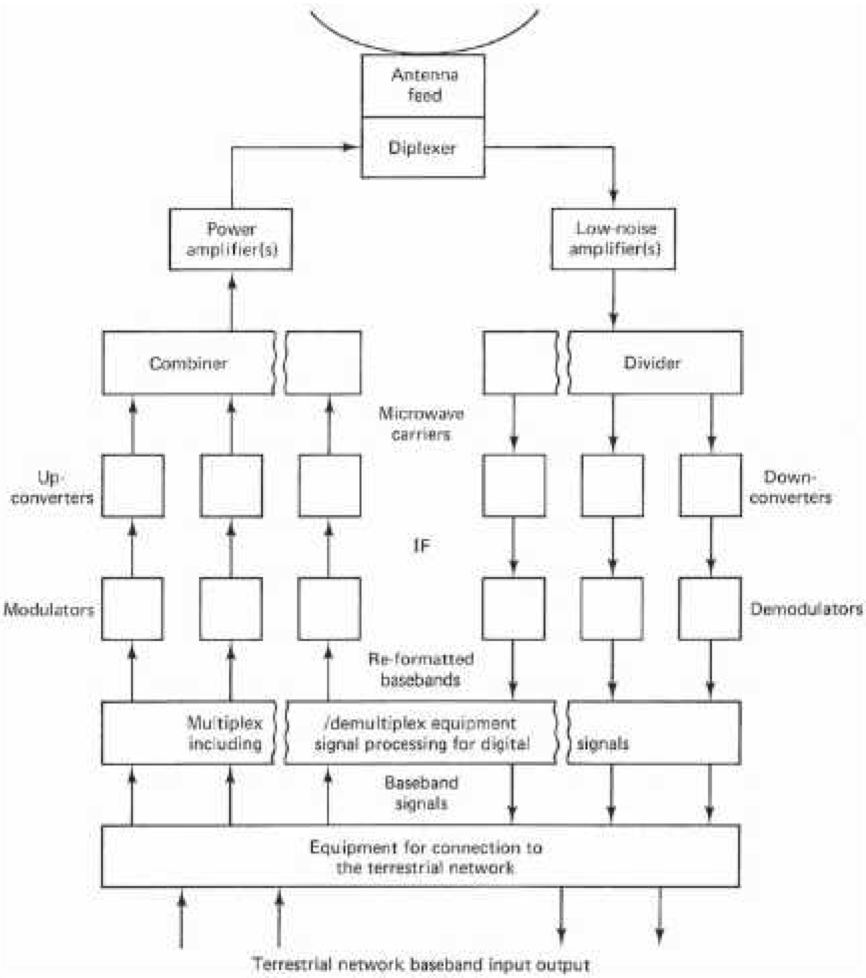
   The energy required to move the satellite from the elliptical transfer orbit into the geosynchronous orbit is supplied by the satellite's apogee kick motor (AKM). This is known as the orbit injection phase.

3.2 Earth Station

The earth segment of a satellite communications system consists of the transmit and receive earth stations. The station's antenna functions in both, the transmit and receive modes, but at different frequencies.

An earth station is generally made up of a multiplexor, a modem, up and downconverters, a high power amplifier (HPA) and a low noiseamplifier (LNA). Almost all transmission to satellites is d igital, and the digital data streams are combined in a multiplexor and fed to a modemthat modula tes a carrier frequency in the 50 to 180 MHz range. An upconverter bumps the carrier into the gi gahertz range, which goes to the HPA and antenna.

For receiving, the LNA boosts the signals to the downconverter, which lowers the freque ncy and sends itto the modem. The modemdemodulates the carrier, and the digital output goes to the demultiplexing device and then to its destinations. See earth station on board vessel and base station. A detailed block diagram is shown in fig. 3.2.



and dish.

Figure 3.2:- Block diagram of a transmit-receive earth station

3.3 Satellite Sub-systems

A satellite communications system can be broadly divided into two segments—a ground segment and a space segment. The space segment will obviously include the satellites, but it also includes the ground facilities needed to keep the satellites operational, these being referred to as the *tracking, telemetry, and command* (TT&C) facilities. In many networks it is common practice to employ a ground station solely for the purpose of TT&C.

In a communications satellite, the equipment which provides the connecting link between the satellite's transmit and receive antennas is referred to as the *transponder*. The transponder forms one of the main sections of the payload, the other being the antenna subsystems.

PAYLOAD:- The payload comprises of a Repeater and Antenna subsystem and performs the primary function of communication.

1- **REPEATER:-** It is a device that receives a signal and retransmits it to a higher level and/or higher power onto the other side of the obstruction so that the signal can cover longer distance.
2- **Transparent Repeater:-** It only translates the uplink frequency to an appropriate downlink frequency. It does so without processing the baseband signal. The main element of a typical transparent repeater is a single beam satellite. Signals from antenna and the feed system are fed into the low-noise amplifier through a bandpass filter.
3- **Regenerative Repeater :-** A repeater, designed for digital transmission, in which digital signals are amplified, reshaped, retimed, and retransmitted.
   Regenerative Repeater can also be called as a device which regenerates incoming digital signals and then retransmits these signals on an outgoing circuit.
4- **Antennas :-** The function of an antenna of a space craft is to receive signals and transmit signals to the ground stations located within the coverage area of the satellite. The choice of the antenna system is therefore governed by the size and shape of the coverage area. Consequently, there is also a limit to the minimum size of the antenna footprint.

3.4 Satellite System Link Models

System Link Budget calculations basically relate two quantities, the transmit power and the receive power, and show in detail how the difference between these two powers is accounted for. Link-power budget calculations also need the additional losses and noise factor which is incorporated with the transmitted and the received signals. Along with losses, this unit also discusses the system noise parameters. Various components of the system add to the noise in the signal that has to be transmitted.

3.4.1 EQUIVALENT ISOTROPIC RADIATED POWER

The key parameter in link-power budget calculations is the equivalent isotropic radiated power factor, commonly denoted as EIRP. Is the amount of power that a theoretical isotropic antenna (which evenly distributes power in all directions) would emit to produce the peak power density observed in the direction of maximum antenna gain. EIRP can be defined as the power input to one end of the transmission link and the problem to find the power received at the other end.

$$\text{EIRP} = \text{G Ps}$$

Where,
G - Gain of the Transmitting antenna and G is in decibels.
Ps- Power of the sender (transmitter) and is calculated in watts.

$$\mathbf{[EIRP] = [G] + [Ps]\ dBW}$$

3.4.2 TRANSMISSION LOSSES:-

As EIRP is thought of as power input of one end to the power received at the other, the problem here is to find the power which is received at the other end. Some losses that occur in the transmitting – receiving process are constant and their values can be pre – determined.

3.4.2.1 Free-Space Transmission Losses (FSL)

This loss is due to the spreading of the signal in space. Going back to the power flux density equation

$$\cdot\ _{m} = P_s / 4\text{®}\, r^2$$

The power that is delivered to a matched receiver is the power flux density. It is multiplied by the effective aperture of the receiving antenna. Hence, the received power is:

$$P_R = \cdot\ _M A_{eff}$$

$$= \frac{EIRP \bullet \text{a}^2 G_R}{4\text{®}\, r^2}$$

Where
r- distance between transmitter and receiver, G_R - power gain at the receiver
In decibels, the above equation becomes:

$$\left[P_R\right] = \left[EIRP\right] + \left[G_R\right] - 10\log\left(\frac{4\text{®}\, r}{\text{a}}\right)^2$$

$$[FSL] = 10\log\left(\frac{4\text{®}\, r}{\text{a}}\right)^2$$

$$\left[P_R\right] = \left[EIRP\right] + \left[G_R\right] - [FSL]$$

3.4.2.2 Feeder Losses (RFL):- This loss is due to the connection between the satellite receiver device and the receiver antenna is improper. Losses here occur is connecting wave guides, filers and couplers. The receiver feeder loss values are added to free space loss.

3.4.2.1 Antenna Misalignment Losses (AML):- To attain a good communication link, the earth station‟s antenna and the communicating satellite‟s antenna must face each other in such a way that the maximum gain is attained.

3.4.2.1 Fixed Atmospheric (AA) and Ionospheric losses (PL):-The gases present in the atmosphere absorb the signals. This kind of loss is usually of a fraction of decibel in quantity. Along with the absorption losses, the ionosphere introduces a good amount of depolarization of signal which results in loss of signal.

3.5 Link Equations

The EIRP can be considered as the input power to a transmission link. Due to the above discussed losses, the power at the receiver that is the output can be considered as a simple calculation of EIRP– losses.

$$\text{Losses} = [FSL] + [RFL] + [AML] + [AA] + [PL]$$

The received power that is P

$$[P_R] = [EIRP] + [G_R] - [Losses]$$

Where;

$[P_R]$ - Received power in dB, [EIRP] - equivalent isotropic radiated power in dBW.

$[G_R]$- Isotropic power gain at the receiver and its value is in dB.

[FSL]-Free-space transmission loss in dB.

[RFL] -Receiver feeder loss in dB.

[AA] -Atmospheric absorption loss in dB.

[AML] -Antenna misalignment loss in dB.

[PL] - Depolarization loss in dB.

CHAPTER 4: MODULATION AND MULTIPLEXING TECHNIQUES

4.1 Multiple Access

Multiple accesses is defined as the technique where in more than one pair of earth stations can simultaneously use a satellite transponder. A multiple access scheme is a method used to distinguish among different simultaneous transmissions in a cell. A radio resource can be a different time interval, a frequency interval or a code with a suitable power level.

If the different transmissions are differentiated for the frequency band, it will bedefined as the Frequency Division Multiple Access (FDMA). Whereas, if transmissions are distinguished on the basis of time, then it is considered as Time Division Multiple Access (TDMA). If a different code is adopted to separate simultaneous transmissions, it will be Code Division Multiple Access (CDMA).

4.1.1 Frequency Division Multiple Access (FDMA)

Frequency Division Multiple Access or FDMA is a channel access method used in multiple-access protocols as a channelization protocol. FDMA gives users an individual allocation of one or several frequency bands, or channels. It is particularly commonplace in satellite communication.

- In FDMA all users share the satellite transponder or frequency channel simultaneously but each user transmits at single frequency.
- FDMA can be used with both analog and digital signal.
- FDMA requires high-performing filters in the radio hardware.
- FDMA is not vulnerable to the timing problems that TDMA has. Since a predetermined frequency band is available for the entire period of communication, stream data (a continuous flow of data that may not be packetized) can easily be used with FDMA.
- Each user transmits and receives at different frequencies as each user gets a unique frequency slots.

4.1.2 Time Division Multiple Access (TDMA)

Time division multiple access (TDMA) is a channel access method for shared medium networks. It allows several users to share the same frequency channel by dividing the signal into different time slots. This allows multiple stations to share the same transmission medium (e.g. radio frequency channel) while using only a part of its channel capacity.

- Shares single carrier frequency with multiple users.
- Slots can be assigned on demand in dynamic TDMA.
- Less stringent power control than CDMA due to reduced intra cell interference
- Higher synchronization overhead than CDMA
- Cell breathing (borrowing resources from adjacent cells) is more complicated than in CDMA.
- Frequency/slot allocation complexity.

4.1.3 Code Division Multiple Access (CDMA)

Code division multiple access (CDMA) is a channel access method used by various radio communication technologies. CDMA is an example of multiple access, which is where several transmitters can send information simultaneously over a single communication channel. This allows several users to share a band of frequencies (see bandwidth). CDMA is used as the access method in many mobile phone standards such as cdmaOne, CDMA2000 (the 3G evolution of cdmaOne), and WCDMA (the 3G standard used by GSM carriers), which are often referred to as simply *CDMA*.

- One of the early applications for code division multiplexing is in the Global Positioning System (GPS). This predates and is distinct from its use in mobile phones.
- The Qualcomm standard IS-95, marketed as cdmaOne.
- The Qualcomm standard IS-2000, known as CDMA2000, is used by several mobile phone companies, including the Globalstar satellite phone network.
- The UMTS 3G mobile phone standard, which uses W-CDMA.
- CDMA has been used in the Omni TRACS satellite system for transportation logistics.

4.2 Direct Broadcast Satellite Services

Direct-broadcast satellite (DBS) is a type of artificial satellite which usually sends satellite television signals for home reception through geostationary satellites. The type of satellite television which uses direct-broadcast satellites is known as direct-broadcast satellite television (DBSTV) or direct-to-home television (DTHTV). This has initially distinguished the transmissions directly intended for home viewers from cable television distribution services that are sometimes carried on the same satellite.

A DBS subscriber installation consists of a dish antenna two to three feet (60 to 90 centimeters) in diameter, a conventional TV set, a signal converter placed next to the TV set, and a length of coaxial cable between the dish and the converter. The dish intercepts microwave signals directly from the satellite. The converter produces output that can be viewed on the TV receiver. Broadcast services include audio, television, and Internet services. Direct broadcast television, which is digital TV, is the subject of this chapter. A Typical DBS system block diagram is shown in fig. 4.1. The home receiver consists of two units—an outdoor unit and an indoor unit.

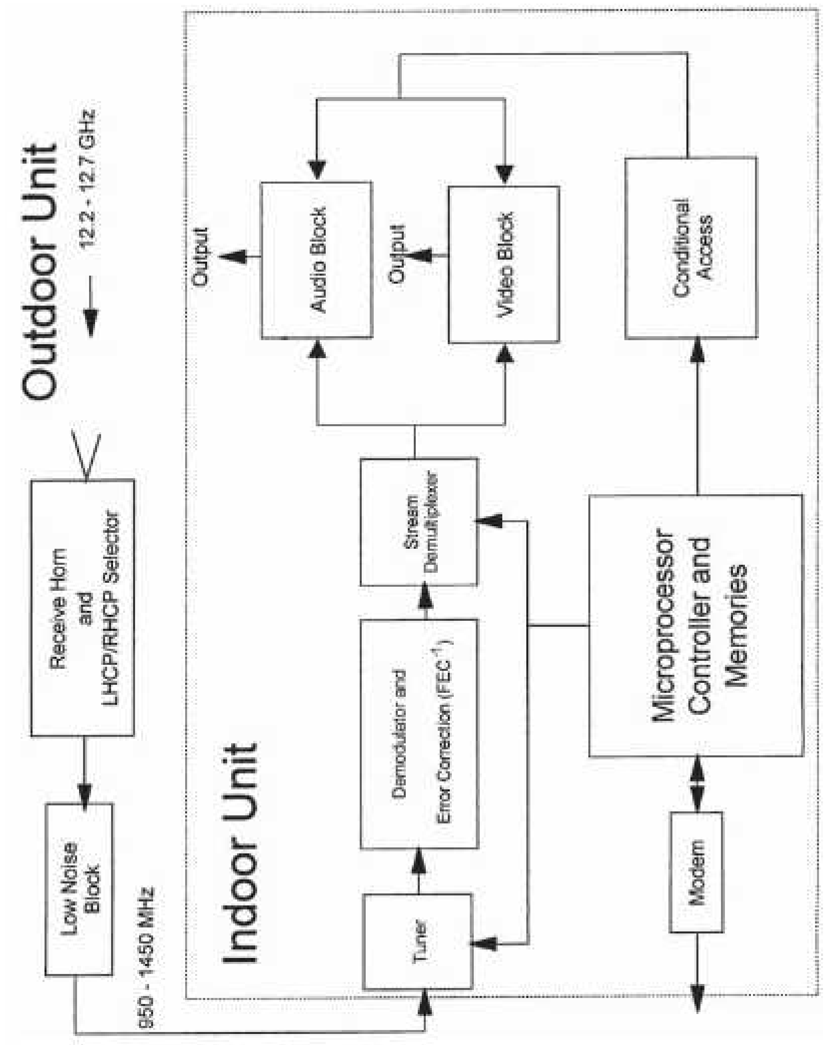


Fig. 4.1 Block schematic for the outdoor unit (ODU) and IDU unit of DBS.

The Home Receiver Outdoor Unit (ODU):- The downlink signal, covering the frequency range 12.2 to 12.7 GHz, is focused by the antenna into the receive horn. The horn feeds into a polarizer that can be switched to pass either left-hand circular or right-hand circular polarized signals. The low-noise block that follows the polarizer contains a *low-noise amplifier* (LNA) and a down converter. The down converter converts the 12.2- to 12.7-GHz band to 950 to 1450 MHz, a frequency range better suited to transmission through the connecting cable to the indoor unit.

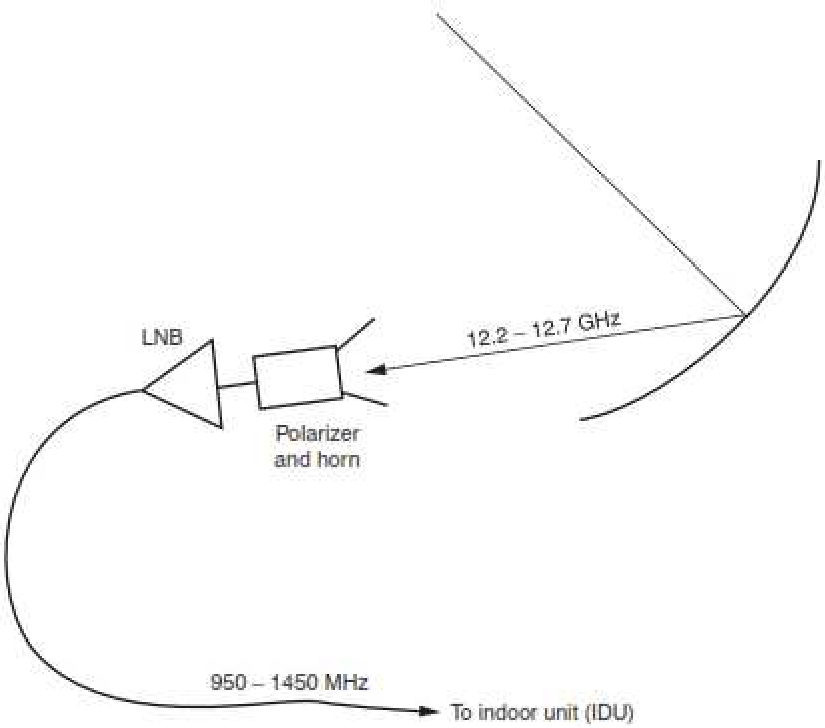


Fig. 4.2 Block schematic for the outdoor unit (ODU)

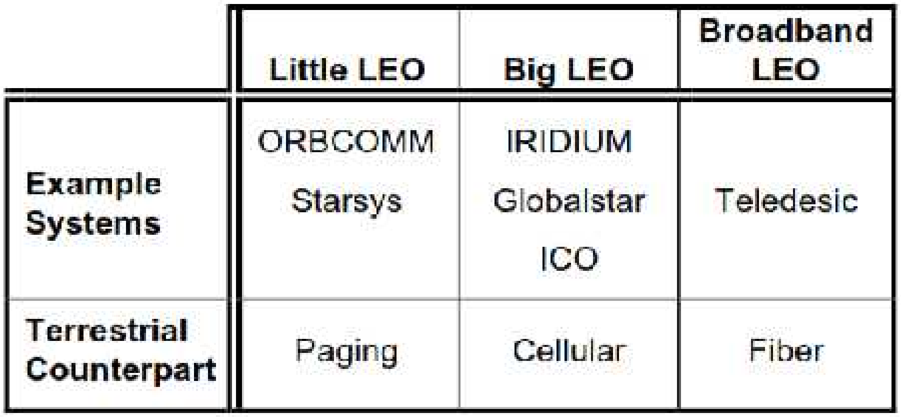
The Home Receiver Indoor Unit (IDU):- The transponder frequency bands shown in Fig. 16.2 are down converted to be in the range 950 to 1450 MHz, but of course, each transponder retains its 24-MHz bandwidth. The IDU must be able to receive any of the 32 transponders, although only 16 of these will be available for a single polarization. The tuner selects the desired transponder. It should be recalled that the carrier at the center frequency of the transponder is QPSK modulated by the bit stream, which itself may consist of four to eight TV programs TDM. Following the tuner, the carrier is demodulated, the QPSK modulation being converted to a bit stream. Error correction is carried out in the decoder block labeled FEC.

4.3 Application of LEO

The evolution from geo-stationary to low-Earthorbit (LEO) satellites has resulted in a number of proposed global satellite systems, which can be grouped into three distinct types - Little LEOs, Big LEOs, and Broadband LEOs. These systems can best be distinguished by reference to their terrestrial counterparts: paging, cellular, and fiber, as shown in Table 4.1. On the ground, paging, cellular, and fiber services are complementary, not competitive, because they offer fundamentally different kinds of services. Similarly, the Little LEOs, Big LEOs, and Broadband

LEOs are complementary rather than competitive because they are providing distinctly different services targeted at different markets, and have different pricing structures.

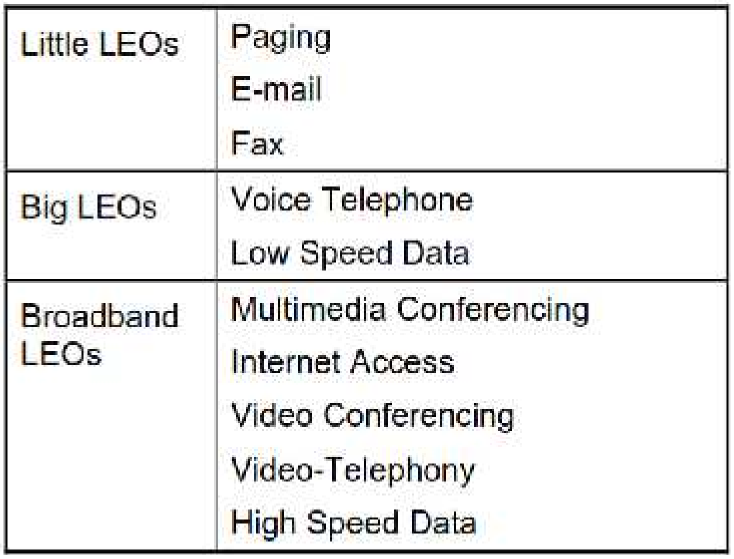
Table 4.1 Terrestrial Counterparts



	Little LEO	Big LEO	Broadband LEO
Example Systems	ORBCOMM Starsys	IRIDIUM Globalstar ICO	Teledesic
Terrestrial Counterpart	Paging	Cellular	Fiber

Typical applications of the various types of LEO systems are shown in Table 2. Of course the Big LEOs can support the Little LEO applications, and the Broadband LEOs can support both the Big and Little LEO applications.

Table 4.2 Application of LEO Satellites



Little LEOs	Paging E-mail Fax
Big LEOs	Voice Telephone Low Speed Data
Broadband LEOs	Multimedia Conferencing Internet Access Video Conferencing Video-Telephony High Speed Data

4.4 MEO and GEO Satellites

Medium Earth orbit (**MEO**), sometimes called **intermediate circular orbit** (**ICO**), is the region of space around the Earth above low Earth orbit (altitude of 2,000 kilometres (1,243 mi)) and below geostationary orbit (altitude of 35,786 kilometres (22,236 mi)).The most common use for satellites in this region is for navigation, communication, and geodetic/space environment science.[1] The most common altitude is approximately 20,200 kilometres (12,552 mi)), which yields an orbital period of 12 hours, as used, for example, by the Global Positioning System (GPS). Other satellites in Medium Earth Orbit include Glonass (with an altitude of 19,100 kilometres (11,868 mi)) and Galileo (with an altitude of 23,222 kilometres (14,429 mi)) constellations.[*citation needed*] Communications satellites that cover the North and South Pole are also put in MEO.

Geostationary satellites appear to be fixed over one spot above the equator. Receiving and transmitting antennas on the earth do not need to track such a satellite. These antennas can be fixed in place and are much less expensive than tracking antennas. These satellites have revolutionized global communications, television broadcasting and weather forecasting, and have a number of important defenseand intelligence applications.

One disadvantage of geostationary satellites is a result of their high altitude: radio signals take approximately 0.25 of a second to reach and return from the satellite, resulting in a small but significant signal delay. This delay increases the difficulty of telephone conversation and reduces the performance of common network protocols such as TCP/IP, but does not present a problem with non-interactive systems such as television broadcasts. There are a number of proprietary satellite data protocols that are designed to proxy TCP/IP connections over long-delay satellite links—these are marketed as being a partial solution to the poor performance of native TCP over satellite links. TCP presumes that all loss is due to congestion, not errors, and probes link capacity with its "slow-start" algorithm, which only sendspackets once it is known that earlier packets have been received. Slow start is very slow over a path using a geostationary satellite. There are approximately 600 geosynchronous satellites, some of which are not operational

Table 4.3 Application of MEO and GEO Satellites

Satellites	Application
Medium Earth Orbit	High-speed telephone signals
Geosynchronous Orbit	Satellite Television
Geostationary Orbit	Direct broadcast television

References

1- http://www.williamcraigcook.com/satellite/index.html

2- http://transition.fcc.gov/cgb/kidszone/satellite/kidz/into_space.html

3- Satellite Communications Dennis Roddy 3rd edition, McGraw Hill publication.

4- http://www.radio-electronics.com/info/satellite/satellite-orbits/satellites-orbit-definitions.php.

5- www.wikipedia.com

6- LEOs - THE COMMUNICATIONS SATELLITES of 21Century by Mark A. Sturza

OPTICAL FIBRE SYSTEM

An optical fiber (or optical fibre) is a flexible, transparent fiber made of extruded glass (silica) or plastic, slightly thicker than a human hair. It can function as a waveguide, or “light pipe”, to transmit light between the two ends of the fiber.The field of applied science and engineering concerned with the design and application of optical fibers is known as fiber optics.

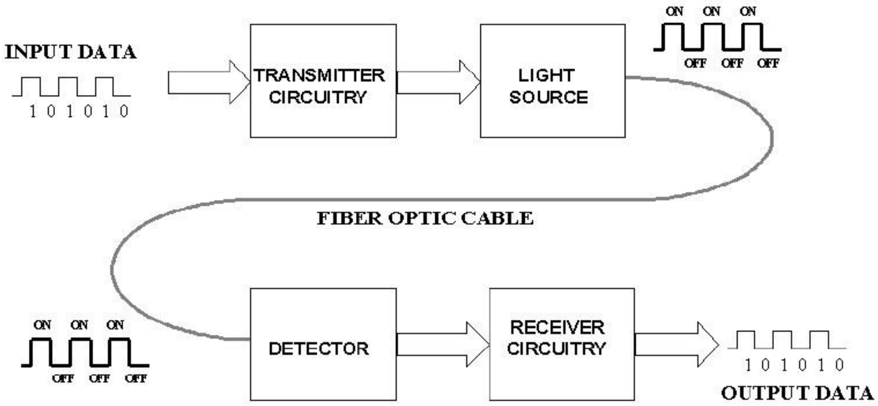
Optical fibers are widely used in fiber-optic communications, where they permit transmission over longer distances and at higher bandwidths (data rates) than wire cables. Fibers are used instead of metal wires because signals travel along them with less loss and are also immune to electromagnetic interference. Fibers are also used for illumination, and are wrapped in bundles so that they may be used to carry images, thus allowing viewing in confined spaces. Specially designed fibers are used for a variety of other applications, including sensors and fiber lasers.

Optical fibers typically include a transparent core surrounded by a transparent cladding material with a lower index of refraction. Light is kept in the core by total internal reflection. This causes the fiber to act as a waveguide. Fibers that support many propagation paths or transverse modes are called multi-mode fibers (MMF), while those that only support a single mode are called single-mode fibers (SMF). Multi-mode fibers generally have a wider core diameter, and are used for short-distance communication links and for applications where high power must be transmitted. Single-mode fibers are used for most communication links longer than 1,000 meters (3,300 ft).

How a Fiber Optic Communication Works?

Unlike copper wire based transmission where the transmission entirely depends on electrical signals passing through the cable, the fiber optics transmission involves transmission of signals in the form of light from one point to the other. Furthermore, a fiber optic communication network consists of transmitting and receiving circuitry, a light source and detector devices like the ones shown in the figure.

When the input data, in the form of electrical signals, is given to the transmitter circuitry, it converts them into light signal with the help of a light source. This source is of LED whose amplitude, frequency and phases must remain stable and free from fluctuation in order to have efficient transmission. The light beam from the source is carried by a fiber optic cable to the destination circuitry wherein the information is transmitted back to the electrical signal by a receiver circuit.



The Receiver circuit consists of a photo detector along with an appropriate electronic circuit, which is capable of measuring magnitude, frequency and phase of the optic field. This type of communication uses the wave lengths near to the infrared band that are just above the visible range. Both LED and Laser can be used as light sources based on the application.

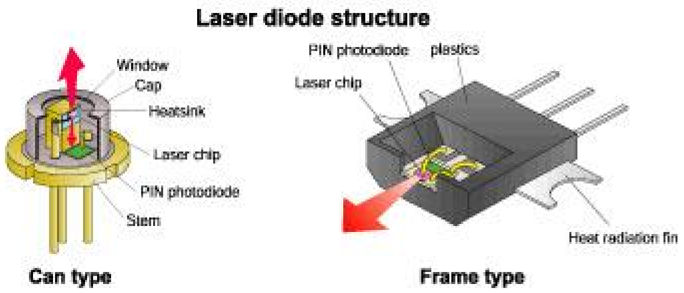
3 Basic Elements of a Fiber Optic Communication System

There are three main basic elements of fiber optic communication system. They are

1. Compact Light Source
2. Low loss Optical Fiber
3. Photo Detector

Accessories like connectors, switches, couplers, multiplexing devices, amplifiers and splices are also essential elements in this communication system.

1. Compact Light Source

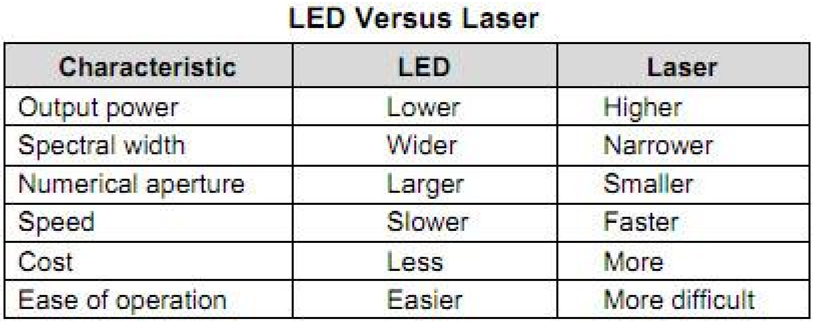


Laser Diodes

Depending on the applications like local area networks and the long haul communication systems, the light source requirements vary. The requirements of the sources include power, speed, spectral line width, noise, ruggedness, cost, temperature, and so on. Two components are used as light sources: light emitting diodes (LED’s) and laser diodes.

The light emitting diodes are used for short distances and low data rate applications due to their low bandwidth and power capabilities. Two such LEDs structures include Surface and Edge Emitting Systems. The surface emitting diodes are simple in design and are reliable, but due to its broader line width and modulation frequency limitation edge emitting diode are mostly used. Edge emitting diodes have high power and narrower line width capabilities.

For longer distances and high data rate transmission, Laser Diodes are preferred due to its high power, high speed and narrower spectral line width characteristics. But these are inherently non-linear and more sensitive to temperature variations.



LED Versus Laser

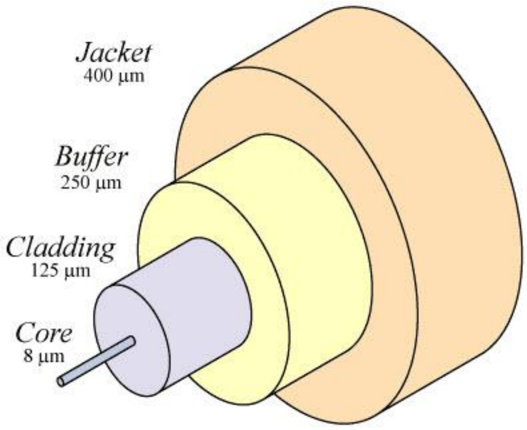
Characteristic	LED	Laser
Output power	Lower	Higher
Spectral width	Wider	Narrower
Numerical aperture	Larger	Smaller
Speed	Slower	Faster
Cost	Less	More
Ease of operation	Easier	More difficult

LED vs Laser Diodes

Nowadays many improvements and advancements have made these sources more reliable. A few of such comparisons of these two sources are given below. Both these sources are modulated using either direct or external modulation techniques.

2. Low Loss Optical Fiber

Optical fiber is a cable, which is also known as cylindrical dielectric waveguide made of low loss material. An optical fiber also considers the parameters like the environment in which it is operating, the tensile strength, durability and rigidity. The Fiber optic cable is made of high quality extruded glass (si) or plastic, and it is flexible. The diameter of the fiber optic cable is in between 0.25 to 0.5mm (slightly thicker than a human hair).



A Fiber Optic Cable consists of four parts.

- Core
- Cladding
- Buffer
- Jacket

Core

The core of a fiber cable is a cylinder of plastic that runs all along the fiber cable's length, and offers protection by cladding. The diameter of the core depends on the application used. Due to internal reflection, the light travelling within the core reflects from the core, the cladding boundary. The core cross section needs to be a circular one for most of the applications.

Cladding

Cladding is an outer optical material that protects the core. The main function of the cladding is that it reflects the light back into the core. When light enters through the core (dense material) into the cladding(less dense material), it changes its angle, and then reflects back to the core.

Buffer

The main function of the buffer is to protect the fiber from damage and thousands of optical fibers arranged in hundreds of optical cables. These bundles are protected by the cable's outer covering that is called jacket.

JACKET

Fiber optic cable's jackets are available in different colors that can easily make us recognize the exact color of the cable we are dealing with. The color yellow clearly signifies a single mode cable, and orange color indicates multimode.

2 Types of Optical Fibers

Single-Mode Fibers: Single mode fibers are used to transmit one signal per fiber; these fibers are used in telephone and television sets. Single mode fibers have small cores.

Multi-Mode Fibers: Multimode fibers are used to transmit many signals per fiber; these signals are used in computer and local area networks that have larger cores.

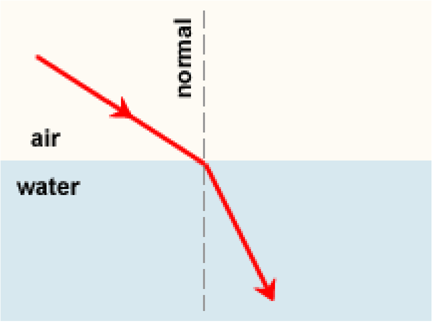
3. Photo Detectors

The purpose of photo detectors is to convert the light signal back to an electrical signal. Two types of photo detectors are mainly used for optical receiver in optical communication system: PN photo diode and avalanche photo diode. Depending on the application's wavelengths, the material composition of these devices vary. These materials include silicon, germanium, InGaAs, etc.

Basic optical laws

Refraction of light

As a light ray passes from one transparent medium to another, it changes direction; this phenomenon is called refraction of light. How much that light ray changes its direction depends on the refractive index of the mediums.



Refractive Index

Refractive index is the speed of light in a vacuum (abbreviated **c**, **c**=299,792.458km/second) divided by the speed of light in a material (abbreviated **v**). Refractive index measures how much a material refracts light. Refractive index of a material, abbreviated as **n**, is defined as

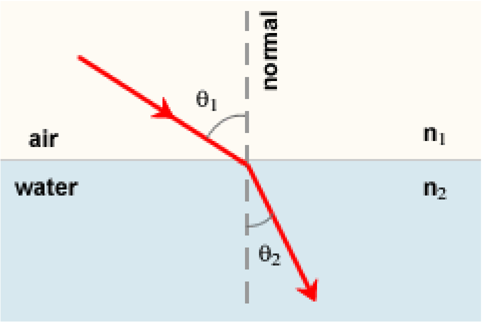
n=c/v

Snell's Law

In 1621, a Dutch physicist named Willebrord Snell derived the relationship between the different angles of light as it passes from one transparent medium to another. When light passes from one transparent material to another, it bends according to Snell's law which is defined as:

$$\boldsymbol{n_1 \sin(\theta_1) = n_2 \sin(\theta_2)}$$

where:
$\boldsymbol{n_1}$ is the refractive index of the medium the light is leaving
$\boldsymbol{\theta_1}$ is the incident angle between the light beam and the normal (normal is 90° to the interface between two materials)
$\boldsymbol{n_2}$ is the refractive index of the material the light is entering
$\boldsymbol{\theta_2}$ is the refractive angle between the light ray and the normal

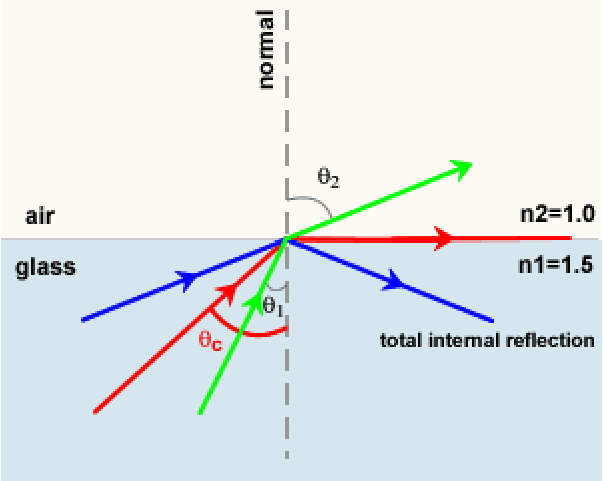


Note:

For the case of θ_1 = 0° (i.e., a ray perpendicular to the interface) the solution is θ_2 = 0° regardless of the values of n_1 and n_2. That means a ray entering a medium perpendicular to the surface is never bent.

The above is also valid for light going from a dense (higher n) to a less dense (lower n) material; the symmetry of Snell's law shows that the same ray paths are applicable in opposite direction.

Total Internal Reflection



When a light ray crosses an interface into a medium with a higher refractive index, it bends towards the normal. Conversely, light traveling cross an interface from a higher refractive index medium to a lower refractive index medium will bend away from the normal.

This has an interesting implication: at some angle, known as the **critical angle θ_c**, light traveling from a higher refractive index medium to a lower refractive index medium will be refracted at 90°; in other words, refracted along the interface.

If the light hits the interface at any angle larger than this critical angle, it will not pass through to the second medium at all. Instead, all of it will be reflected back into the first medium, a process known as **total internal reflection**.

The critical angle can be calculated from Snell's law, putting in an angle of 90° for the angle of the refracted ray $\boldsymbol{\theta_2}$. This gives $\boldsymbol{\theta_1}$:

$$\theta_1 = \arcsin[(n_2/n_1)*\sin(\theta_2)]$$

Since

$\boldsymbol{\theta_2} = 90°$

So

$\mathbf{sin}(\boldsymbol{\theta_2}) = 1$

Then

$\boldsymbol{\theta_c = \theta_1 = \arcsin(n_2/n_1)}$

For example, with light trying to emerge from glass with $\boldsymbol{n_1}$=1.5 into air ($\boldsymbol{n_2}$ =1), the critical angle $\boldsymbol{\theta_c}$ is arcsin(1/1.5), or 41.8°.

For any angle of incidence larger than the critical angle, Snell's law will not be able to be solved for the angle of refraction, because it will show that the refracted angle has a sine larger than 1, which is not possible. In that case all the light is totally reflected off the interface, obeying the law of reflection.

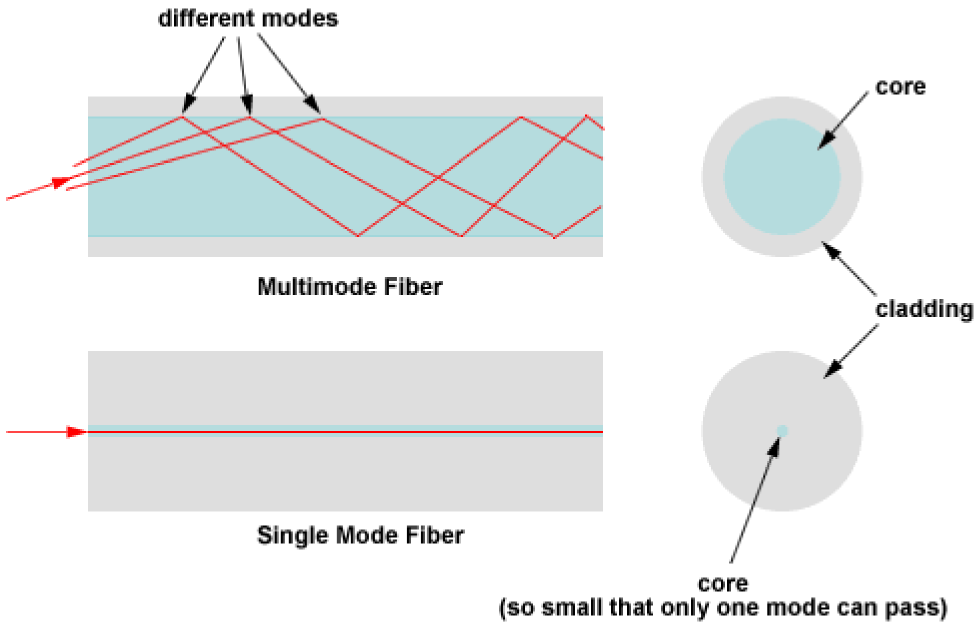
Optical Fiber Mode

What is Fiber Mode?

An optical fiber guides light waves in distinct patterns called *modes*. Mode describes the distribution of light energy across the fiber. The precise patterns depend on the wavelength of light transmitted and on the variation in refractive index that shapes the core. In essence, the variations in refractive index create boundary conditions that shape how light waves travel through the fiber, like the walls of a tunnel affect how sounds echo inside.

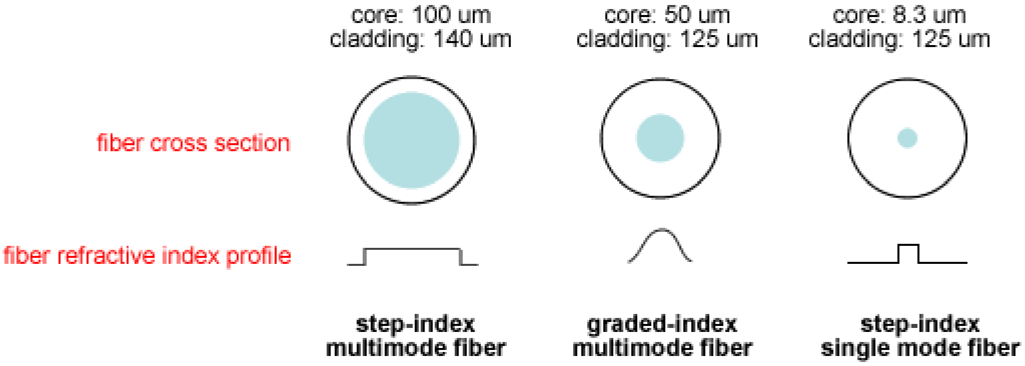
We can take a look at large-core step-index fibers. Light rays enter the fiber at a range of angles, and rays at different angles can all stably travel down the length of the fiber as long as they hit the core-cladding interface at an angle larger than critical angle. These rays are different modes.

Fibers that carry more than one mode at a specific light wavelength are called multimode fibers. Some fibers have very small diameter core that they can carry only one mode which travels as a straight line at the center of the core. These fibers are single mode fibers. This is illustrated in the following picture.



Optical Fiber Index Profile

Index profile is the refractive index distribution across the core and the cladding of a fiber. Some optical fiber has a step index profile, in which the core has one uniformly distributed index and the cladding has a lower uniformly distributed index. Other optical fiber has a graded index profile, in which refractive index varies gradually as a function of radial distance from the fiber center. Graded-index profiles include power-law index profiles and parabolic index profiles. The following figure shows some common types of index profiles for single mode and multimode fibers.



Multimode Fibers

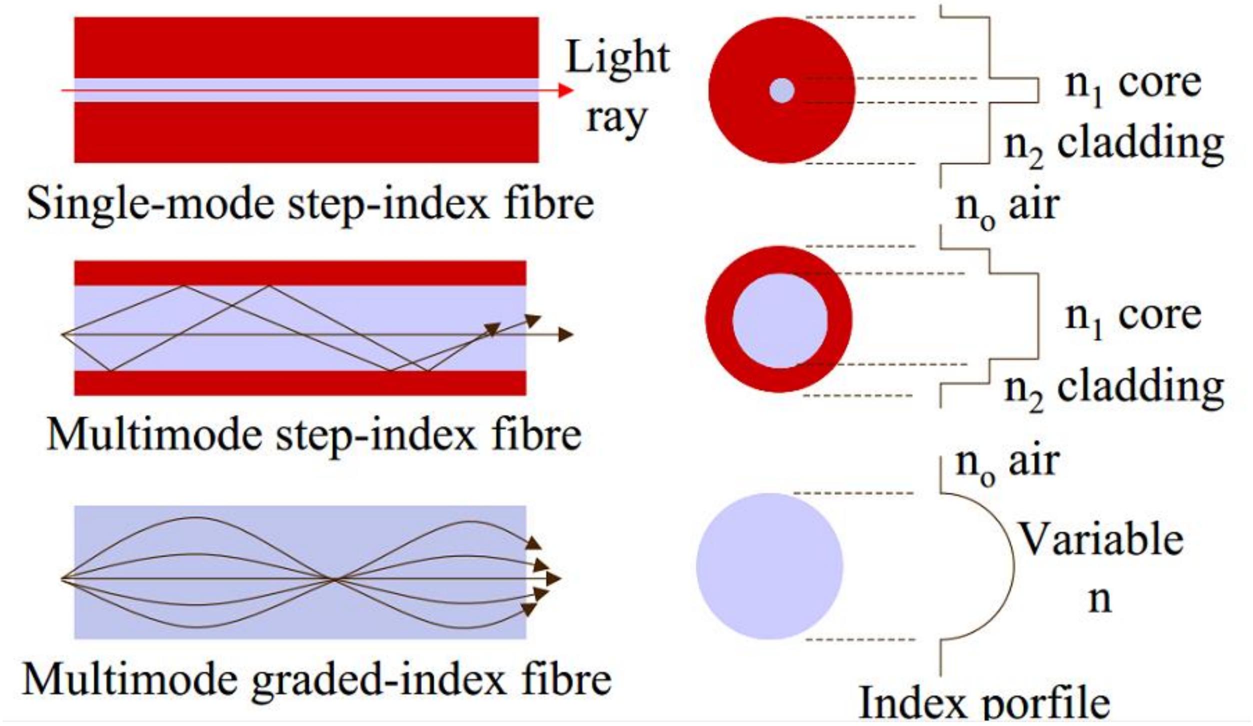
As their name implies, multimode fibers propagate more than one mode. Multimode fibers can propagate over 100 modes. The number of modes propagated depends on the core size and numerical aperture (NA).

As the core size and NA increase, the number of modes increases. Typical values of fiber core size and NA are 50 to 100 micrometer and 0.20 to 0.29, respectively.

Single Mode Fibers

The core size of single mode fibers is small. The core size (diameter) is typically around 8 to 10 micrometers. A fiber core of this size allows only the fundamental or lowest order mode to propagate around a 1300 nanometer (nm) wavelength. Single mode fibers propagate only one mode, because the core size approaches the operational wavelength. The value of the normalized frequency parameter (V) relates core size with mode propagation.

In single mode fibers, V is less than or equal to 2.405. When V = 2.405, single mode fibers propagate the fundamental mode down the fiber core, while highorder modes are lost in the cladding. For low V values (<1.0), most of the power is propagated in the cladding material. Power transmitted by the cladding is easily lost at fiber bends. The value of V should remain near the 2.405 level.



Multimode Step Index Fiber

Core diameter range from 50-1000 m .Light propagate in many different ray paths, or modes, hence the name multimode Index of refraction is same all across the core of the fiber Bandwidth range 20-30 MHz . Multimode Graded Index Fiber The index of refraction across the core is gradually changed from a maximum at the center to a minimum near the edges, hence the name “Graded Index” Bandwidth ranges from 100MHz-Km to 1GHz-Km

Pulse dispersion in a step index optical fiber is given by

$$\text{pulse dispersion} = \frac{\triangle\, n_1\, \ell}{c}$$

where

$\triangle$is the difference in refractive indices of core and cladding.

n1is the refractive index of core

$\ell$is the length of the optical fiber under observation

$$c = 3 \times 10^8\, \text{ms}^{-1}$$

Graded-Index Multimode Fiber

Contains a core in which the refractive index diminishes gradually from the center axis out toward the cladding. The higher refractive index at the center makes the light rays moving down the axis advance more slowly than those near the cladding. Due to the graded index, light in the core curves helically rather than zigzag off the cladding, reducing its travel distance. The shortened path and the higher speed allow light at the periphery to arrive at a receiver at about the same time as the slow but straight rays in the core axis. The result: digital pulse suffers less dispersion. This type of fiber is best suited for local-area networks.

Pulse dispersion in a graded index optical fiber is given by

$$\text{Pulse dispersion} = \frac{k\delta n\ n_1\ l}{c},$$

where

$\delta n$is the difference in refractive indices of core and cladding,

$n_1$is the refractive index of the cladding,

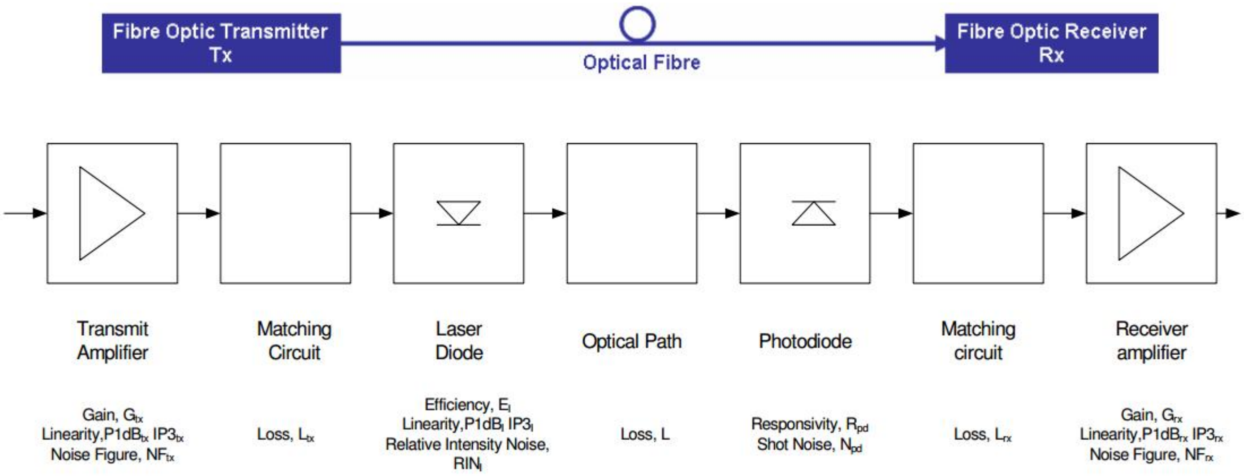
$l$is the length of the fiber taken for observing the pulse dispersion,

$c \approx 3 \times 10^8\ \mathrm{m/s}$is the speed of light, and

k is the constant of graded index profile.

Fibre Optic Link Budget

The FOL budget provides the design engineer with quantitative performance information about the FOL.It is determined by computing the FOL power budget and overall link gain.



Fibre Optic Power Budget

The FOL power budget (PB) is simply the difference between the maximum and minimum signals that the FOL can transport.

Fibre Optic Link Gain

FOL link gain is a summation of gains and losses derived from the different elements of the FOL as shown in above figure . Gains and losses attributed to the Tx, Rx, optical fibre and connectors, as well as any additional in-line components such as splitters, multiplexers, splices etc, must be taken into accounts when computing the linkloss budget.

In the case of a simple point-to-point link described in Above figure , and resistively matched (50 ohms) components,

the link gain (G) is expressed as:-

G = T + R - 2LO (1)

Where T is the gain of the Tx, R is the gain of theRx, and LO is the insertion loss attributed to the fibre link. Note the factor of two in this last optical term, meaning that for each dB optical loss there is a corresponding 2dB RF loss.

To calculate LO the following information is needed.

Standard Corning SMF28 single mode fibre has an insertion loss 0.2dB/km at 1310nm and 0.15dB/km at 1550nm. Optical connectors such as FC/APC typically have an insertion loss of 0.25dB. Optical splices introduce a further 0.25dB loss. Refer to TIA 568 standard forInterfacility and Premise cable specifications.

Output Noise Power

The output noise power of an analogue FOL must alsobe considered when quantifying the overall link budget. The measured output noise power is defined as:-

Output Noise Power = ONF + 10log10 (BW)

Where ONF (Optical Noise Floor) is the noise output of the link on its own, defined in a bandwidth of 1Hz,and BW is the bandwidth of the service transported over fibre. In a real installation, the NF, or Noise Figure is used to define the noise performance of the fibre optic link and is related to the output noise floor as follows:

ONF = -174dBm + NF + G (3)

-174dBm, is the noise contribution from an ideal 1ohm resistive load at zero degrees Kelvin.

The measured output noise power is given as:-

= -174dBm + NF + G + 10log10 (MBW)

ATTENUATION ON OPTICAL FIBER

The signal on optical attenuates due to following mechanisms.

1. Intrinsic loss in the fiber material.
2. Scattering due to micro irregularities inside the fiber.
3. Micro-bending losses due to micro-deformation of the fiber.
4. Bending or radiation losses on the fiber.

The first two losses are intrinsically present in any fiber and the last two depend on the environment in which the fiber is laid.

Material Loss

(a) Due to impurities: The material loss is due to the impurities present in glass used for making fibers. Inspite of best purification efforts, there are always impurities like Fe, Ni, Co, Al which are present in the fiber material. The Fig. shows attenuation due to various molecules inside glass as a function of wavelength. It can be noted from the figure that the material loss due to impurities reduces substantially beyond about 1200nm wavelength.

(b)Due to OH molecule: In addition, the OH molecule diffuses in the material and causes absorption of light. The OH molecule has main absorption peak somewhere in the deep infra-red wavelength region. However, it shows substantial loss in the range of 1000 to 2000nm.

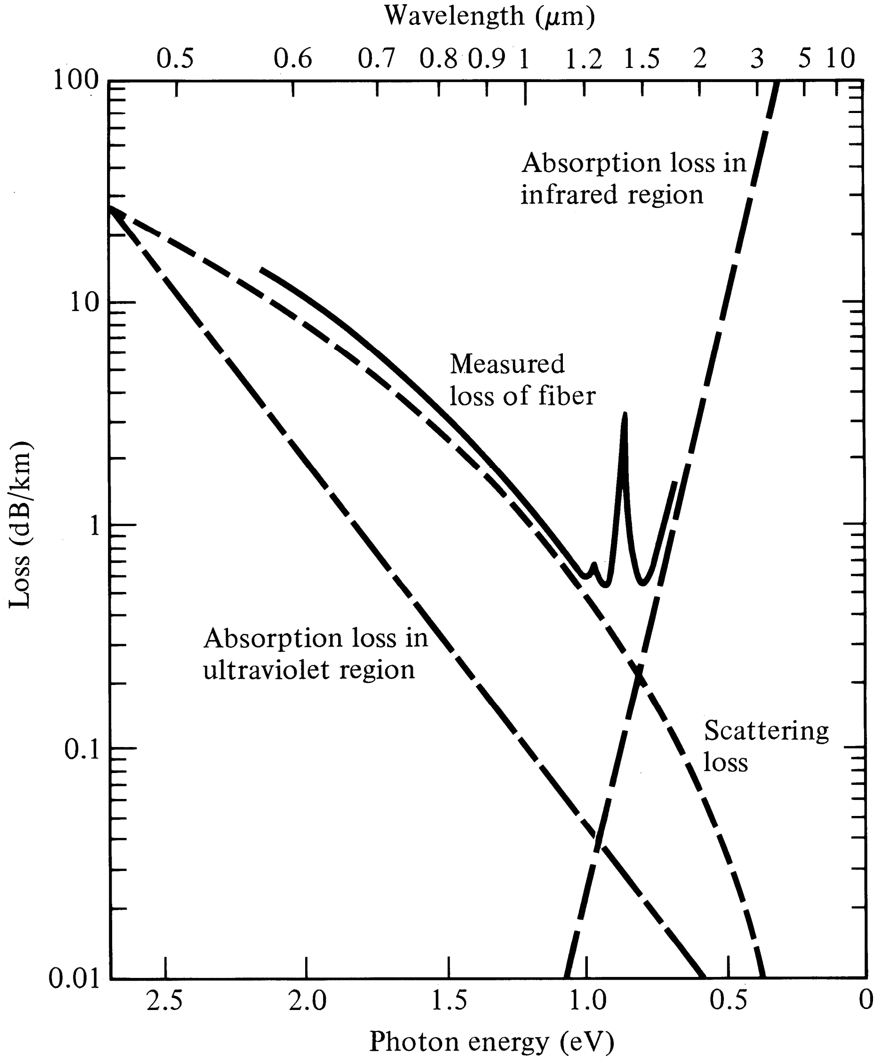
(b) Due to infra-red absorption : Glass intrinsically is a good infra-red absorber. As we increase the wavelength the infra-red loss increases rapidly.

SCATTERING LOSS

The scattering loss is due to the non-uniformity of the refractive index inside the core of the fiber. The refractive index of an optical fiber has fluctuation of the order of 10^{-4} over spatial scales much smaller than the optical wavelength. These fluctuations act as scattering centres for the light passing through the fiber. The process is, Rayleigh Scattering . A very tiny fraction of light gets scattered and therefore contributes to the loss.

The Rayleigh scattering is a very strong function of the wavelength. The scattering loss varies as λ^{-4}. This loss therefore rapidly reduces as the wavelength increases. For each doubling of the wavelength, the scattering loss reduces by a factor of 16. It is then clear that the scattering loss at 1550nm is about factor of 16 lower than that at 800nm.

The following Fig. shows the infrared, scattering and the total loss as a function of wavelength.

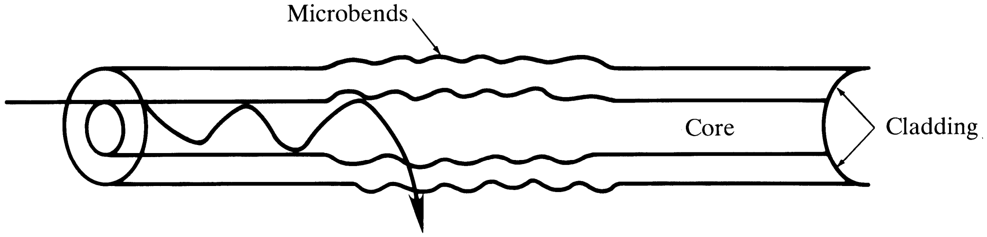


It is interesting to see that in the presence of various losses, there is a natural window in the optical spectrum where the loss is as low as 0.2-0.3dB/Km. This window is from 1200nm to 1600nm.

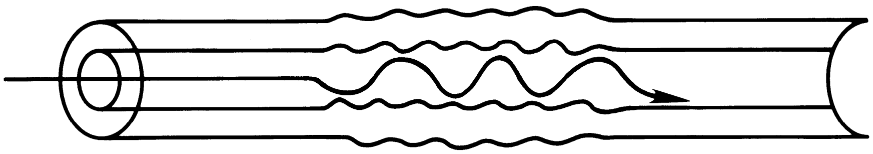
There is a local attenuation peak around 1400nm which is due to OH absorption. The low-loss window therefore is divided into sub-windows, one around 1300nm and other around 1550nm. In fact these are the windows which are the II and III generation windows of optical communication.

MICRO-BENDING LOSSES

While commissioning the optical fiber is subjected to micro-bending as shown in Fig.



Power loss from higher-order modes



Power coupling to higher-order modes

The analysis of micro-bends is a rather complex task. However, just for basic understanding of how the loss takes place due to micro-bending, we use following arguments.

In a fiber without micro-bends the light is guided by total internal reflection (ITR) at the core-cladding boundary. The rays which are guided inside the fiber has incident angle greater than the critical angle at the core-cladding interface. In the presence of micro-bends however, the direction of the local normal to the core-cladding interface deviates and therefore the rays may not have angle of incidence greater than the critical angle and consequently will be leaked out.

A part of the propagating optical energy therefore leaks out due to micro-bends.

Depending upon the roughness of the surface through which the fiber passes, the micro-bending loss varies.

Typically the micro-bends increase the fiber loss by 0.1-0.2 dB/Km.

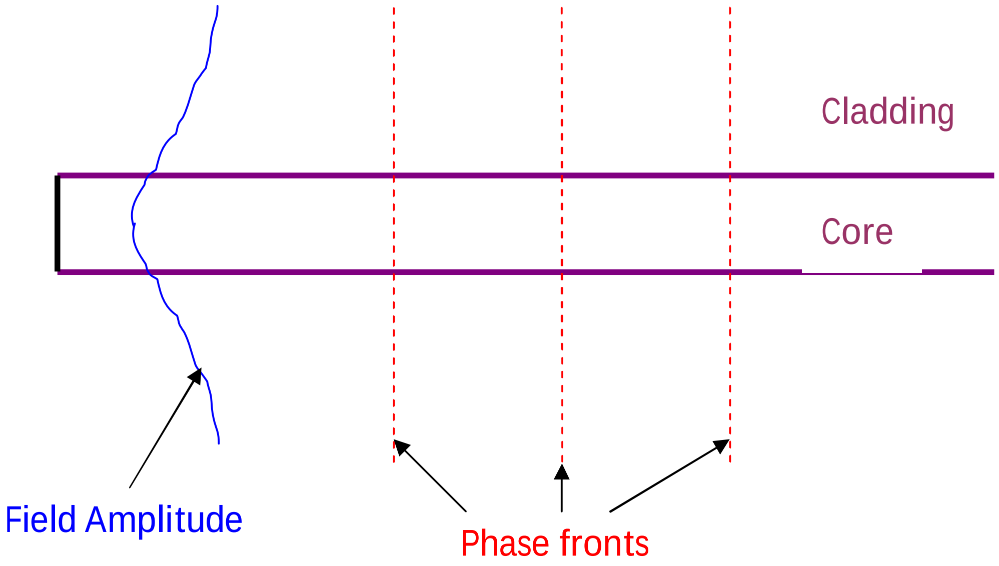
RADIATION OR BENDING LOSS

While laying the fiber the fiber may undergo a slow bend. In micro-bend the bending is on micron scale, whereas in a slow bend the bending is on cm scale. A typical example of a slow bend is a formation of optical fiber loop.

The loss mechanism due to bending loss can be well understood using modal propagation model.

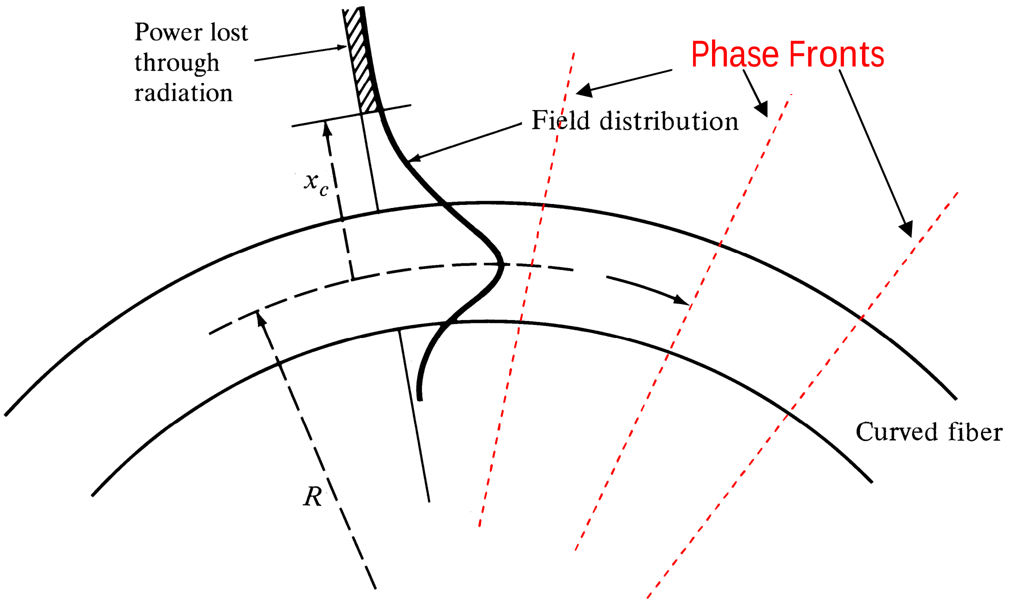
As we have seen, the light inside a fiber propagates in the form of modes. The modal fields decay inside the cladding away from the core cladding interface. Theoretically the field in the cladding is finite no matter how far away we are from the core-cladding interface. Now look at the amplitude and phase distribution for the fibers which are straight and which are bent over an circular arc as shown in Fig.

Phase Fronts in a Straight Fiber



It can be noted that for the straight the phase fronts are parallel and each point on the phase front travels with the same phase velocity.

Phase Fronts for a Bent Fiber



However, as soon the fiber is bent (no matter how gently) the phase fronts are no more parallel. The phase fronts move like a fan pivoted to the center of curvature of the bent fiber (see Fig.). Every point on the phase front consequently does not move with same velocity. The velocity increases as we move radially outwards the velocity of the phase front increases. Very quickly we reach to a distance x_c from the fiber where the velocity tries to become greater than the velocity of light in the cladding medium.

Since the velocity of energy can not be greater than velocity of light, the energy associated with the modal field beyond x_c gets detached from the mode and radiates away. This is called the bending or the radiation loss.

Following important things can be noted about the bending loss.

1. The radiation loss is present in every bent fiber no matter how gentle the bend is.
2. Radiation loss depends upon how much is the energy beyond x_c.
3. For a given modal field distribution if x_c reduces, the radiation loss increases. The x_c reduces as the radius of curvature of the bent fiber reduces, that is the fiber is sharply bent.
4. The number of modes therefore reduces in a multimode fiber in presence of bends.

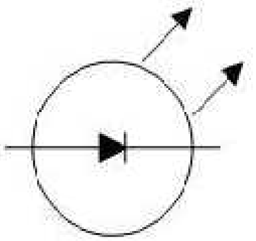
Light Emitting Diodes:-

INTRODUCTION

Over the past 25 years the light-emitting diode (LED) has grown from a laboratory curiosity to a broadly used light source for signaling applications . In 1992 LED production reached a level of approximately 25 billion chips , and $2 . 5 billion worth of LED-based components were shipped to original equipment manufacturers .

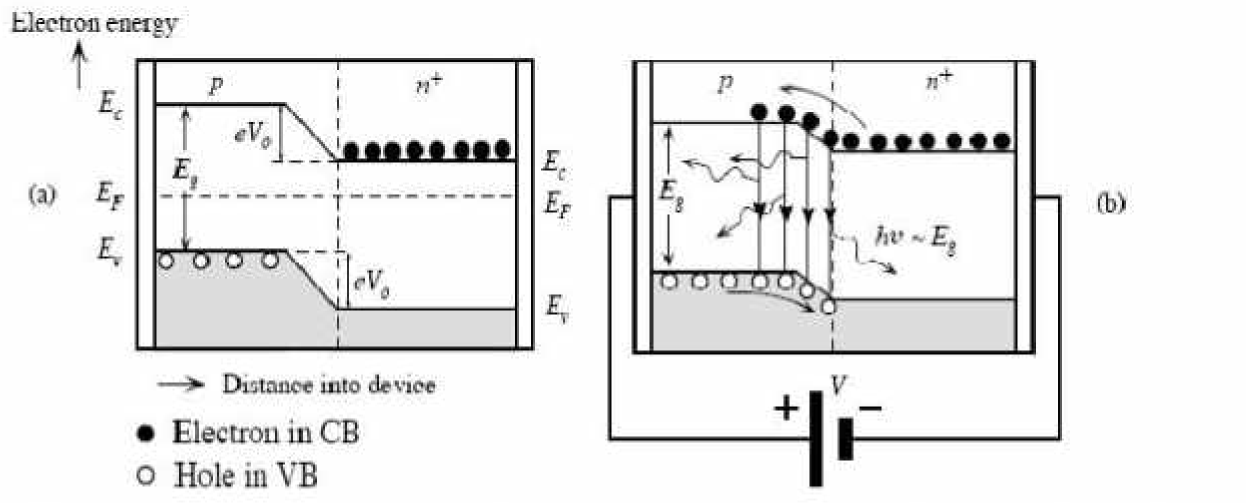
This article covers light-emitting diodes from the basic light-generation processes todescriptions of LED products . First , we will deal with light-generation mechanisms and light extraction . Four major types of device structures—from simple grown or dif fused homojunctions to complex double heterojunction devices are discussed next , followed by a description of the commercially important semiconductors used for LEDs , from the pioneering GaAsP system to the AlGaInP system that is currently revolutionizing LED technology . Then processes used to fabricate LED chips are explained—the growth of GaAs and GaP substrates ; the major techniques used for growing the epitixal material in which the light-generation processes occur ; and the steps required to create LED chips up to the point of assembly . Next the important topics of quality and reliability—in particular , chip degradation and package-related failure mechanisms—will be addressed . Finally , LED-based products , such as indicator lamps , numeric and alphanumeric displays , optocouplers , fiber-optic transmitters , and sensors , are described . This article covers the mainstream structures , materials , processes , and applications in use today . It does not cover certain advanced structures , such as quantum well or strained layer devices , The reader is also referred to for current information on edge-emitting LEDs , whose fabrication and use are similar to lasers .

Schematic:



Theory:

A Light emitting diode (LED) is essentially a pn junction diode. When carriers are injected across a forward-biased junction, it emits incoherent light. Most of the commercial LEDs are realized using a highly doped n and a p Junction.



(a) The energy band diagram of a pn^+ (heavily n-type doped) junction without any bias. Built-in potential V_o prevents electrons from diffusing from n^+ to p side. (b) The applied bias reduces V_o and thereby allows electrons to diffuse or be injected into the p-side. Recombination around the junction and within the diffusion length of the electrons in the p-side leads to photon emission.

Figure 1: p-n+ Junction under Unbiased and biased conditions

To understand the principle, let’s consider an unbiased pn+ junction (Figure1 shows the pn+ energy band diagram). The depletion region extends mainly into the p-side. There is a potential barrier from Ec on the n-side to the Ec on the p-side, called the built-in voltage, V0. This potential barrier prevents the excess free electrons on the n+ side from diffusing into the p side. When a Voltage V is applied across the junction, the built-in potential is reduced from V0 to V0 – V. This allows the electrons from the n+ side to get injected into the p-side. Since electrons are the minority carriers in the p-side, this process is called minority carrier injection. But the hole injection from the p side to n+ side is very less and so the current is primarily due to the flow of electrons into the p-side. These electrons injected into the p-side recombine with the holes. This recombination results in spontaneous emission of photons (light). This effect is called injection electroluminescence. These photons should be allowed to escape from the device without being reabsorbed.

The recombination can be classified into the following two kinds

- Direct recombination
- Indirect recombination

Direct Recombination:

In direct band gap materials, the minimum energy of the conduction band lies directly above the maximum energy of the valence band in momentum space energy . In this material, free electrons at the bottom of the conduction band can recombine directly with free holes at the top of the valence band, as the momentum of the two particles is the same. This transition from conduction band to valence band involves photon emission (takes care of the principle of energy conservation). This is known as direct recombination. Direct recombination occurs spontaneously. GaAs is an example of a direct band-gap material.

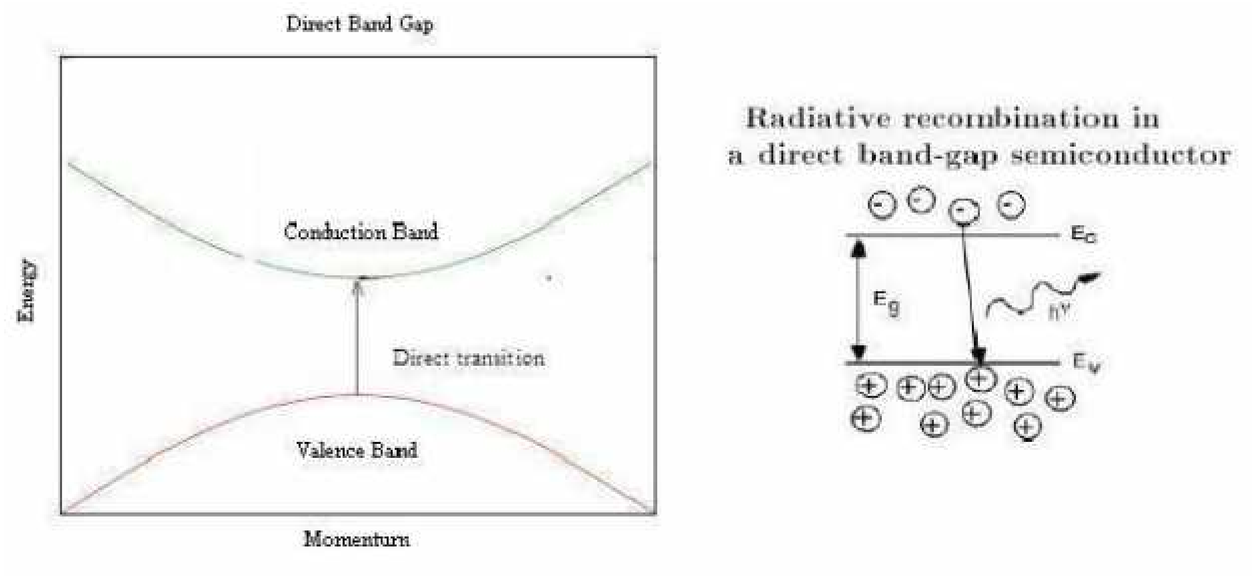


Figure 2: Direct Bandgap and Direct Recombination

Indirect Recombination:

In the indirect band gap materials, the minimum energy in the conduction band is shifted by a k-vector relative to the valence band. The k-vector difference represents a difference in momentum. Due to this difference in momentum, the probability of direct electronhole recombination is less.

In these materials, additional dopants(impurities) are added which form very shallow donor states. These donor states capture the free electrons locally; provides the necessary momentum shift for recombination. These donor states serve as the recombination centers. This is called Indirect (non-radiative) Recombination.

Nitrogen serves as a recombination center in GaAsP. In this case it creates a donor state, when SiC is doped with Al, it recombination takes place through an acceptor level. when SiC is doped with Al, it recombination takes place through an acceptor level.

The indirect recombination should satisfy both conservation energy, and momentum.

Thus besides a photon emission, phonon emission or absorption has to take place.

GaP is an example of an indirect band-gap material.

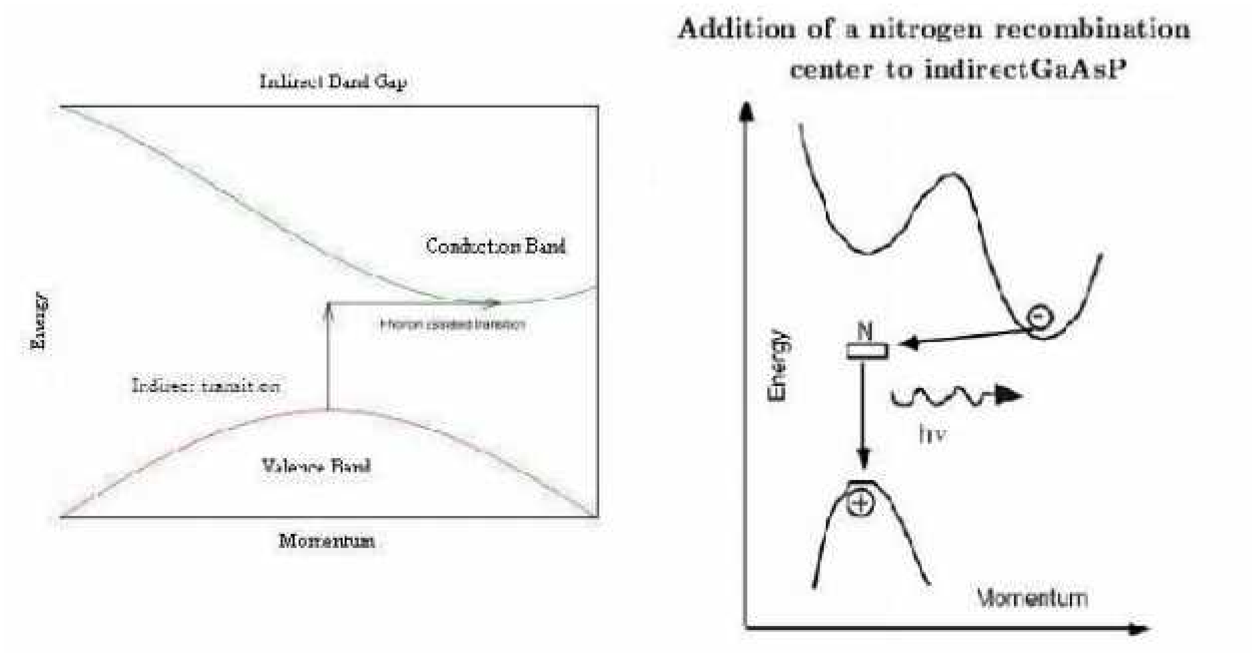


Figure 3: Indirect Bandgap and NonRadiative recombination

The wavelength of the light emitted, and hence the color, depends on the band gap energy of the materials forming the p-n junction.

The emitted photon energy is approximately equal to the band gap energy of the semiconductor. The following equation relates the wavelength and the energy band gap.

$$h\nu = Eg$$
$$hc/\lambda = Eg$$
$$\lambda = hc/ Eg$$

Where h is Plank’s constant, c is the speed of the light and Eg is the energy band gap Thus, a semiconductor with a 2 eV band-gap emits light at about 620 nm, in the red. A 3 eV band-gap material would emit at 414 nm, in the violet.

LED Materials:

An important class of commercial LEDs that cover the visible spectrum are the III-V. ternary alloys based on alloying GaAs and GaP which are denoted by GaAs1-yPy. InGaAlP is an example of a quarternary (four element) III-V alloy with a direct band gap.

The LEDs realized using two differently doped semiconductors that are the same material is called a homojunction. When they are realized using different bandgap materials they are called a heterostructure device. A heterostructure LED is brighter than a homoJunction LED.

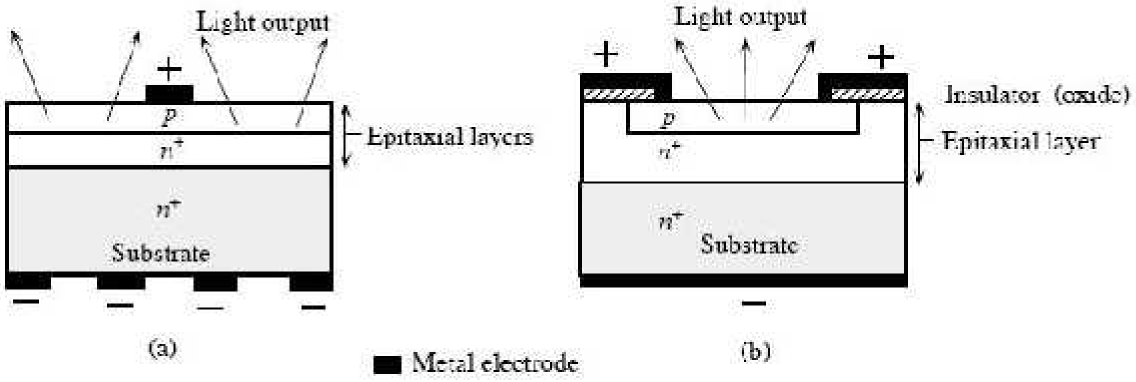
LED Structure:

The LED structure plays a crucial role in emitting light from the LED surface. The LEDs are structured to ensure most of the recombinations takes place on the surface by the following two ways.

• By increasing the doping concentration of the substrate, so that additional free minority charge carriers electrons move to the top, recombine and emit light at the surface.

• By increasing the diffusion length $L = \sqrt{D\tau}$, where D is the diffusion coefficient and τ is the carrier life time. But when increased beyond a critical length there is a chance of re-absorption of the photons into the device.

The LED has to be structured so that the photons generated from the device are emitted without being reabsorbed. One solution is to make the p layer on the top thin, enough to create a depletion layer. Following picture shows the layered structure. There are different ways to structure the dome for efficient emitting.



A schematic illustration of typical planar surface emitting LED devices. (a) *p*-layer grown epitaxially on an n^+ substrate. (b) First n^+ is epitaxially grown and then *p* region is formed by dopant diffusion into the epitaxial layer.

LED structure

LEDs are usually built on an n-type substrate, with an electrode attached to the p-typelayer deposited on its surface. P-type substrates, while less common, occur as well. Manycommercial LEDs, especially GaN/InGaN, also use sapphire substrate.

LED efficiency:

A very important metric of an LED is the external quantum efficiency η_{ext}. It quantifies the efficeincy of the conversion of electrical energy into emitted optical energy. It is defined as the light output divided by the electrical input power. It is also defined as the product of Internal radiative efficiency and Extraction efficiency.

η_{ext} = Pout(optical) / IV

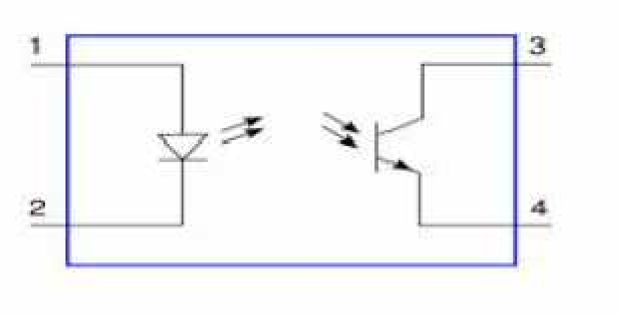
For indirect bandgap semiconductors η_{ext} is generally less than 1%, where as for a direct band gap material it could be substantial.

η_{int} = rate of radiation recombination/ Total recombination

The internal efficiency is a function of the quality of the material and the structure and composition of the layer.

Applications: LED have a lot of applications. Following are few examples.

• Devices, medical applications, clothing, toys
• Remote Controls (TVs, VCRs)
• Lighting
• Indicators and signs
• Optoisolators and optocouplers
• Swimming pool lighting



Optocoupler schematic showing LED and phototransistor

Advantages of using LEDs:

• LEDs produce more light per watt than incandescent bulbs; this is useful inbattery powered or energy-saving devices.

• LEDs can emit light of an intended color without the use of color filters that traditional lighting methods require. This is more efficient and can lower initial costs.

• The solid package of the LED can be designed to focus its light. Incandescent and fluorescent sources often require an external reflector to collect light and direct itin a usable manner.

• When used in applications where dimming is required, LEDs do not change their color tint as the current passing through them is lowered, unlike incandescent lamps, which turn yellow.

• LEDs are ideal for use in applications that are subject to frequent on-off cycling, unlike fluorescent lamps that burn out more quickly when cycled frequently, or High Intensity Discharge (HID) lamps that require a long time before restarting.

• LEDs, being solid state components, are difficult to damage with external shock.Fluorescent and incandescent bulbs are easily broken if dropped on the ground.

• LEDs can have a relatively long useful life. A Philips LUXEON k2 LED has a life time of about 50,000 hours, whereas Fluorescent tubes typically are rated at about 30,000 hours, and incandescent light bulbs at 1,000–2,000 hours.

• LEDs mostly fail by dimming over time, rather than the abrupt burn-out of incandescent bulbs.

• LEDs light up very quickly. A typical red indicator LED will achieve full brightness in microseconds; Philips Lumileds technical datasheet DS23 for the Luxeon Star states "less than 100ns." LEDs used in communications devices can have even faster response times.

• LEDs can be very small and are easily populated onto printed circuit boards.

• LEDs do not contain mercury, unlike compact fluorescent lamps.

Disadvantages:

• LEDs are currently more expensive, price per lumen, on an initial capital cost basis, than more conventional lighting technologies. The additional expense partially stems from the relatively low lumen output and the drive circuitry and power supplies needed. However, when considering the total cost of ownership (including energy and maintenance costs), LEDs far surpass incandescent or halogen sources and begin to threaten the future existence of compact fluorescent lamps.

• LED performance largely depends on the ambient temperature of the operating environment. Over-driving the LED in high ambient temperatures may result in overheating of the LED package, eventually leading to device failure. Adequate heat-sinking is required to maintain long life .

• LEDs must be supplied with the correct current. This can involve series resistors or current-regulated power supplies.

• LEDs do not approximate a "point source" of light, so they cannot be used in applications needing a highly collimated beam. LEDs are not capable of providing divergence below a few degrees. This is contrasted with commercial ruby lasers with divergences of 0.2 degrees or less. However this can be corrected by using lenses and other optical devices.

Laser diodes:-

Laser diodes (also called .injection lasers.) are in effect anspecialised form of LED. Just like a LED, they.re a form of P-N junction diode with a thin depletion layer where electrons and holes collide to create light photons, when the diode is forward biased.

The difference is that in this case the .active. part of the depletion layer (i.e., where most of the current flows) is made quite narrow, to concentrate the carriers. The endsof this narrow active region are also highly polished, or coated with multiple very thin reflective layers to act as mirrors, so it forms a resonant optical cavity.

The forward current level is also increased, to the point where the current density reaches a critical level where carrier population inversion. occurs. This means there are more holes than electrons in the conduction band, and more electrons than holes in the valence band . or in other words, a very large excess population of electrons and holes which can potentially combine to release photons. And when this happens, the creation of new photons can be triggered not just by random collisions of electrons and holes, but lso by the influence of passing photons. Passing photons are then able to stimulate the production of more photons, without themselves being absorbed. So laser action is able to occur: Light Amplification by Stimulated Emission of Radiation. And the important thing to realise is that the photons that are triggered by other passing photons have the same wavelength, and arealso in phase with them. In other words, they end up .in sync. and forming continuous-wave coherent radiation.

Because of the resonant cavity, photons are thus able to travel back and forth from one end of the active region to the other, triggering the production of more and more photons in sync with themselves. So quite a lot of coherent light energy is generated.

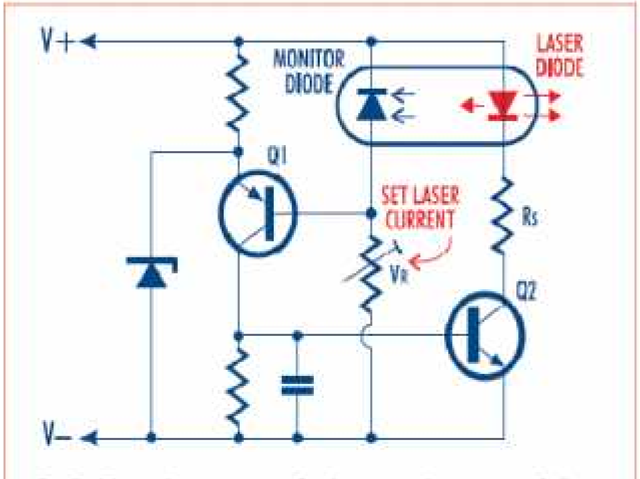


Fig.7: How the monitor diode is used to control the laser operating current and optical output, in a simple feedback circuit.

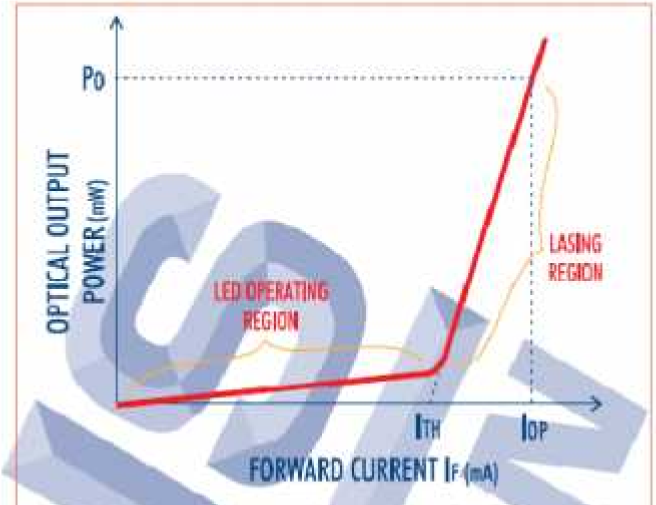


Fig.8: Lasing occurs only at current levels above a critical value, ITH.

And as the ends of the cavity are not totally reflective (typically about 90-95%), some of this coherent light can leave the laser chip . to form its output beam.

Because a laser.s light output is coherent, it is very low in noise and also more suitable for use as a .carrier. for data communications. The bandwidth also tends to be narrower and better defined than LEDs, making them more suitable for optical systems where light beams need to be separated or manipulated on the basis of wavelength.

The very compact size of laser diodes makes them very suitable for use in equipment like CD, DVD and MiniDisc players and recorders. As their light is reasonably well collimated (although not as well as gas lasers) and easily focussed, they.re also used in optical levels, compact handheld laser pointers, barcode scanners etc. There are two main forms of laser diode: the horizontal type, which emits light from the polished ends of the chip, and the vertical or .surface emitting. type. They both operate in the way just described, differing mainly in terms of the way the active light generating region and resonant cavity are formed inside the chip. Because laser diodes have to be operated at such a high current density, and have a very low forward resistance when lasing action occurs, they are at risk of destroying themselves due to thermal runaway. Their operating light density can also rise to a level where the end mirrors can begin melting. As a result their electrical operation must be much more carefully controlled than a LED. This means that not only must a laser diode.s current be regulated by a .constant current. circuit rather than a simple series resistor, but optical negative feedback must generally be used as well . to ensure that the optical output is held to a constant safe level.

To make this optical feedback easier, most laser diodes have a silicon PIN photodiode built right into the package, arranged so that it automatically receives a fixed proportion of the laser.s output. The output of this monitor diode can then be used to control the current fed through the laser by the constant current circuit, for stable and reliable operation. Fig.6 shows a typical .horizontal. type laser chip mounted in its package, with the monitor photodiode mounted on the base flange below it so the diode receives the light output from the .rear. of the laser chip.

Fig.7 (page 3) shows a simple current regulator circuit used to operate a small laser diode, and you can see how the monitor photodiode is connected. The monitor diode is shunting the base forward bias for transistor Q1, which has

its emitter voltage fixed by the zener diode. So as the laseroutput rises, the monitor diode current increases, reducing the conduction of Q1 and hence that of transistor Q2, which controls the laser current. As a result, the laser current is automatically stabilised to a level set by adjustable resistor VR.

Laser diode parameters

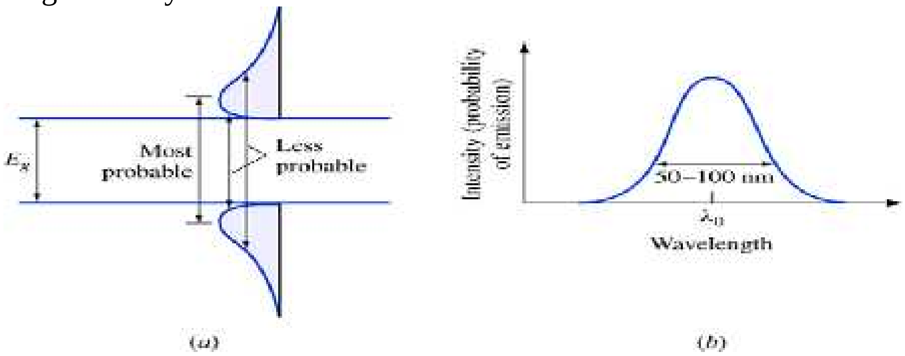
Perhaps the key parameter for a laser diode is the threshold current (ITH), which is the forward current level where lasing actually begins to occur. Below that current level the device delivers some light output, but it operates only as a LED rather than a laser. So the light it does produce in this mode is incoherent. Another important parameter is the rated light output (Po), which is the highest recommended light output level (in milliwatts) for reliable continuous operation. Not surprisingly there.s an operating current level (IOP) which corresponds to this rated light output (Fig.8). There.s also the corresponding current output from the feedback photodiode, known as the monitor current level (Im). Other parameters usually given for a laser diode are its peak lasing wavelength, using given in nanometres (nm); and its beam divergence angles (defined as the angle away from the beam axis before the light intensity drops to 50%), in the X and Y directions (parallel to, and normal to the chip plane).

Laser safety

Although most of the laser diodes used in electronic equipment have quite low optical output levels . typically less than 5mW (milliwatts) . their output is generally concentrated in a relatively narrow beam. This means that it is still capable of causing damage to a human or animal eye, and particularly to its light-sensitive retina.

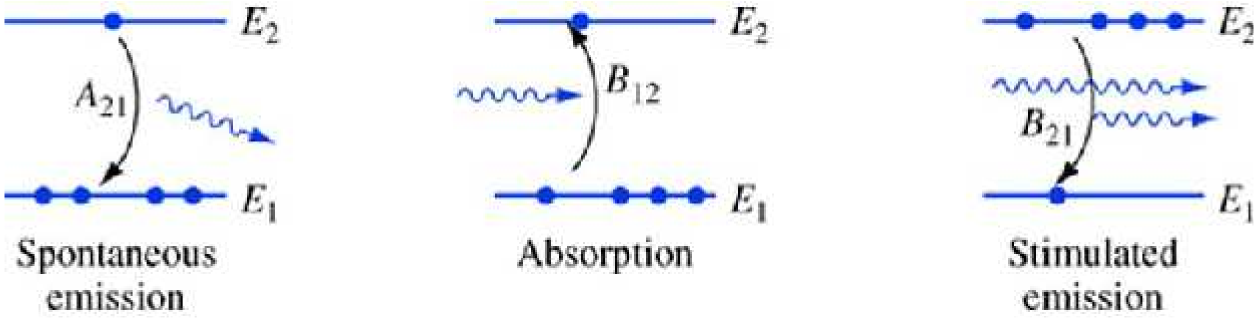
Infra-red (IR) lasers are especially capable of causing eye damage, because their light is not visible. This prevents the eye.s usual protective reflex mechanisms (iris contraction, eyelid closure) from operating. So always take special care when using devices like laser pointers, and especially when working on equipment which includes IR lasers, to make sure that the laser beam cannot enter either your own, or anyone else.s eyes. If you need to observe the output from a laser, either use protective filter goggles or use an IR-sensitive CCD type video camera. Remember that eye damage is often irreversible, especially when it.s damage to the retina.

•Light Emitting Diode
•Light is mostly monochromatic (narrow energy spread comparable to the distribution of electrons/hole populations in the band edges)
•Light is from spontaneous emission (random events in time and thus phase).
•Light diverges significantly



LASER
•Light is essentially single wavelength (highly monochromatic)
•Light is from "stimulated emission" (timed to be in phase with other photons
•Light has significantly lower divergence (Semiconductor versions have more than gas lasers though).

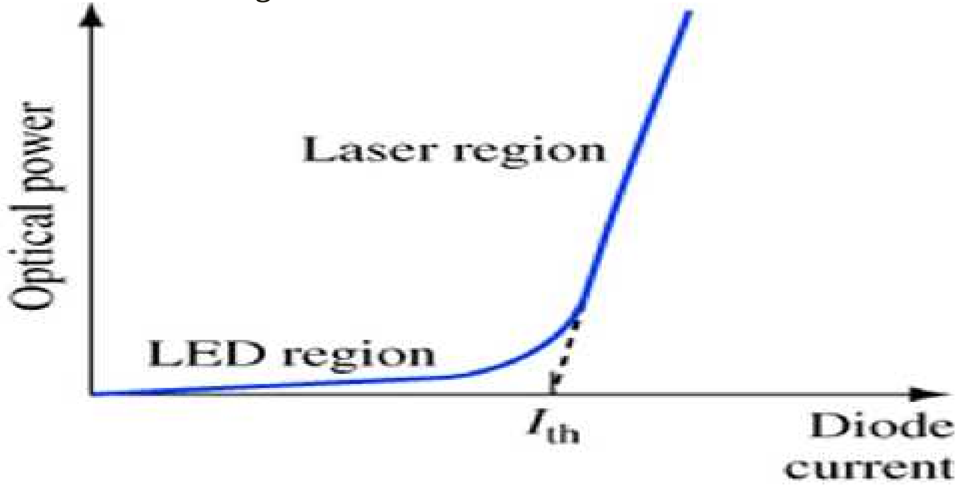
Spontaneous Light Emission



• We can add to our understanding of absorption and spontaneous radiation due to random recombination another form of radiation – Stimulated emission.
• Stimulated emission can occur when we have a "population inversion", i.e. when we have injected so many minority carriers that in some regions there are more "excited carriers" (electrons) than "ground state" carriers (holes).
• Given an incident photon of the band gap energy, a second photon will be "stimulated" by the first photon resulting in two photons with the same energy (wavelength) and phase.

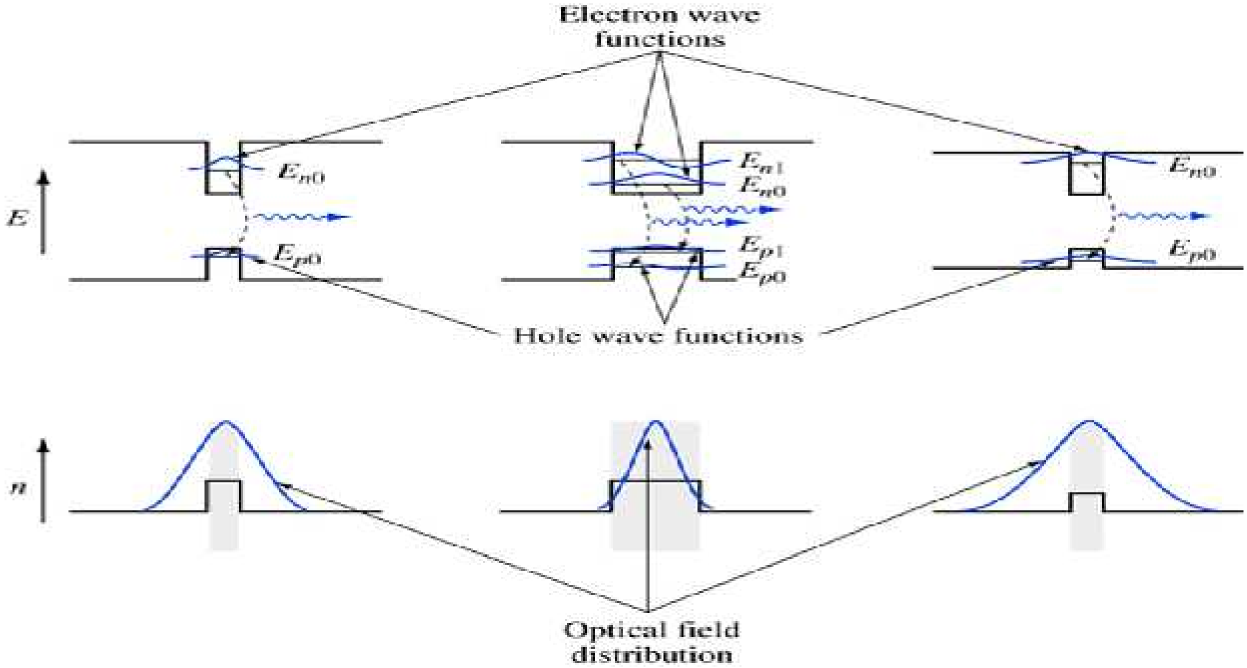
• This phase coherence results in minimal divergence of the optical beam resulting in a directed light source.

Spontaneous vs Stimulated Light Emission:



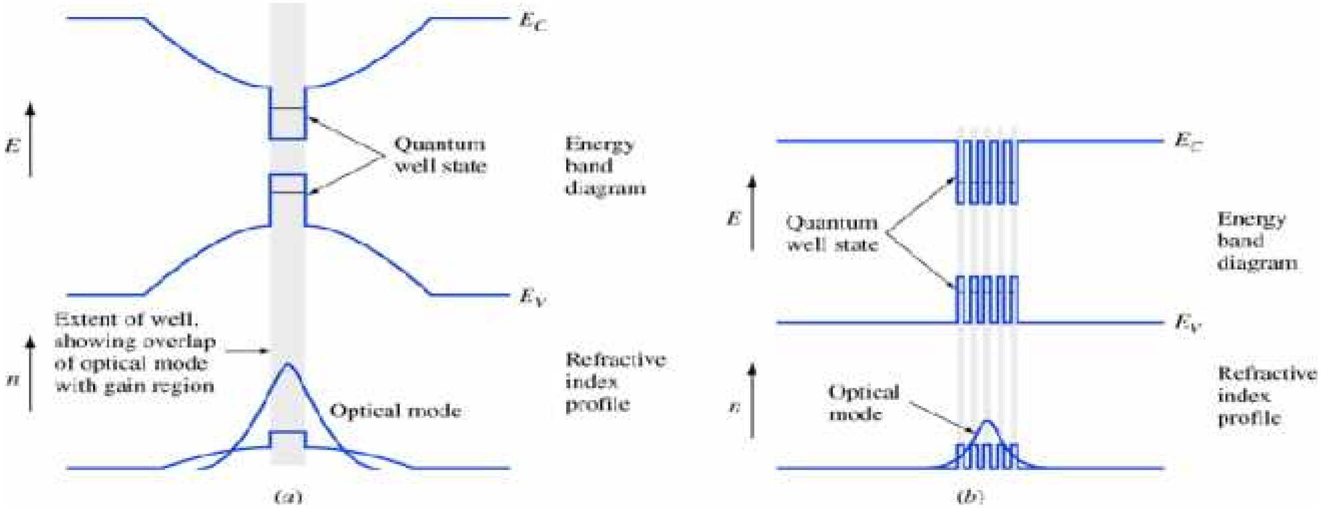
The power-current curve of a laser diode. Below threshold, the diode is an LED. Above threshold, the population is inverted and the light output increases rapidly.

LASER Wavelength Design:



Adjusting the depth and width of quantum wells to select the wavelength of emission is one form of band-gap engineering. The shaded areas indicate the width of the well to illustrate the degree of confinement of the mode.

Advanced LASER Wavelength Design:



(a) A GRINSCH structure helps funnel the carriers into the wells to improve the probability of recombination. Additionally, the graded refractive index helps confine the optical mode in the nearwell region. Requires very precise control over layers due to grading. Almost always implemented via MBE
(b) A multiple quantum well structure has improves carrier capture.
Sometimes the two are combined to give a "digitally graded" device where only two compositions are used but the well thicknesses are varied to implement an effective "index grade"

Photodetectors:-

These are **Opto-electric devices** i.e. to convert the optical signal back into electrical impulses.
The light detectors are commonly made up of semiconductor material.
When the light strikes the light detector a current is produced in the external circuit proportional to the intensity of the incident light.
Optical signal generally is **weakened** and distorted when it emerges from the end of the fiber,
the photodetector must meet following strict performance requirements.
A **high sensitivity** to the emission wavelength range of the received light signal.
A **minimum** addition of **noise** to the signal.
A **fast response** speed to handle the desired data rate.
Be **insensitive** to **temperature** variations.
Be **compatible** with the physical dimensions of the **fiber.**
Have a **Reasonable cost** compared to other system components.
Have a long **operating lifetime.**

Some important parameters while discussing photodetectors:
Quantum Efficiency
It is the ratio of primary electron-hole pairs created by incident photon to the photon incident on the diode material.
Detector Responsivity

This is the ratio of output current to input optical power. Hence this is the efficiency of the device.

Spectral Response Range

This is the range of wavelengths over which the device will operate.

Types of Light Detectors

_ PIN Photodiode

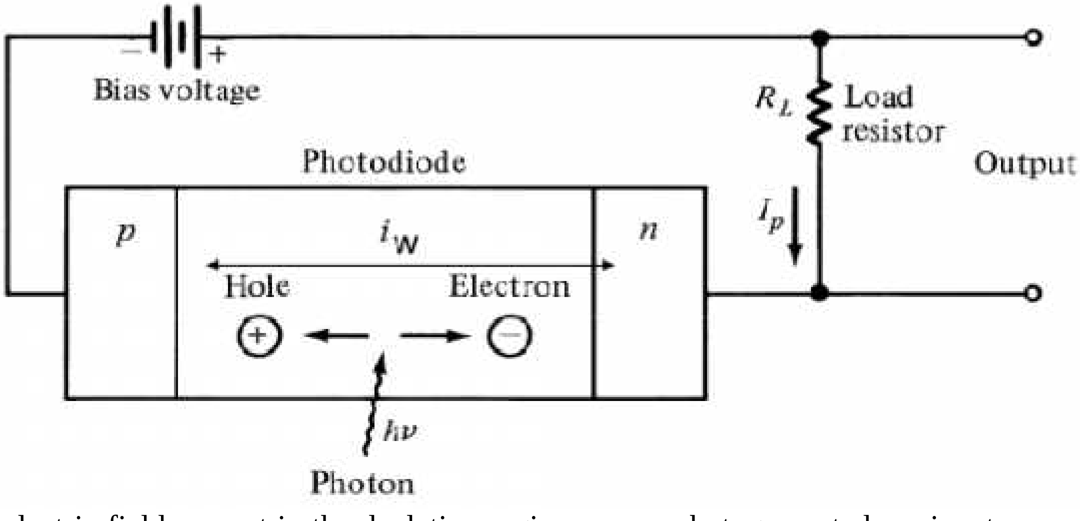
_ Avalanche Photodiode

The Pin Photodetector:-

The **device structure** consists of **p and n** semiconductor regions separated by a very **lightly n-doped intrinsic (i) region.**

In **normal operation** a reverse-bias voltage is applied across the device so that **no free electrons or holes** exist in the **intrinsic region**.

Incident photon having energy **greater than or equal** to the **bandgap energy** of the semiconductor material, **give up itsenergy** and **excite an electron** from the valence band to the conduction band.



The high electric field present in the depletion region causes photogenerated carriers to separate and be collected across the reverse – biased junction. This gives rise to a current flow in an external circuit, known as **photocurrent**.

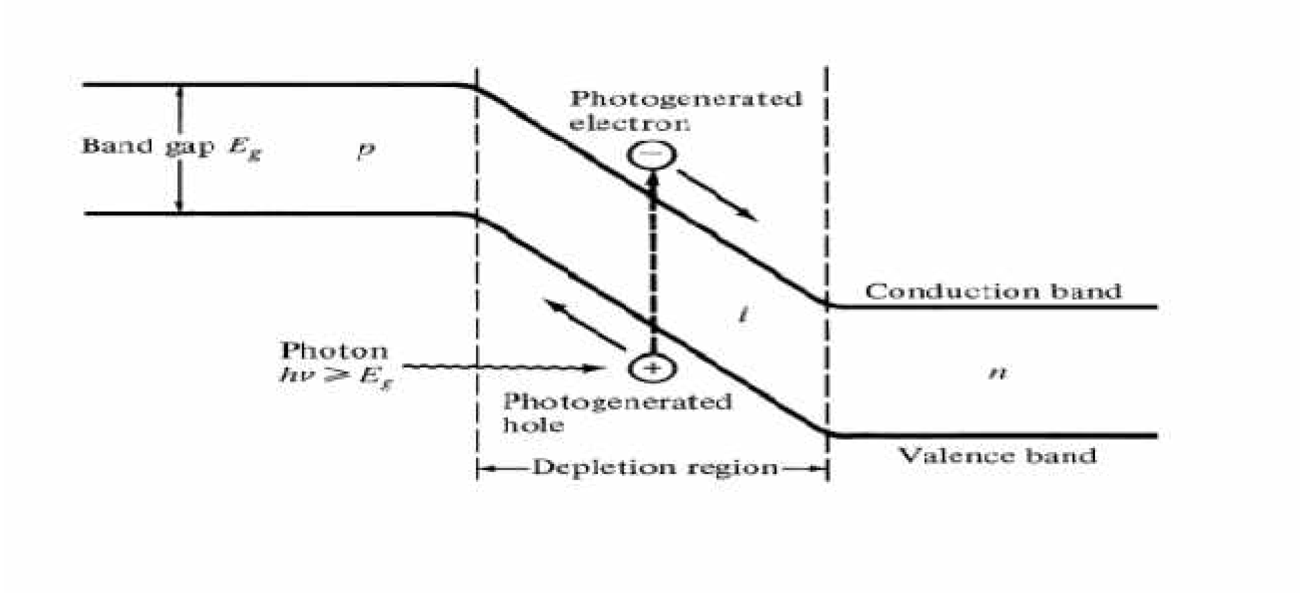
Photocarriers:

Incident photon, generates free (mobile) **electron-hole pairs in the intrinsic region**. These charge carriers are known as **photocarriers,** since they are generated by a photon.

Photocurrent:

The electric field across the device causes the **photocarriers to be swept out of the intrinsic region**, thereby giving rise to a **current flow in an external circuit**. This current flow is known as the **photocurrent.**

Energy-Band diagram for a *pin* photodiode:



An incident photon is able to boost an electron to the conduction band only if it has an energy that is greater than or equal to the bandgap energy

Thus, a particular semiconductor material can be used only over a limited wavelength range.

$$\lambda_c = \frac{hc}{E_g}$$

As the charge carriers flow through the material some of them recombine and disappear.

The charge carriers move a distance Ln or Lp for electrons and holes before recombining. This distance is known as diffusion length

The time it take to recombine is its life time _n or _p respectively.

$$L_n = (D_n \tau_n)^{1/2} \quad \text{and} \quad L_p = (D_p \tau_p)^{1/2}$$

Where Dn and Dp are the diffusion coefficients for electrons and holes respectively.

Photocurrent:-

As a photon flux penetrates through the semiconductor, it will be absorbed.

If *P*in is the optical power falling on the photo detector at *x*=0 and *P*(*x*) is the power level at a distance *x* into the material then the incremental change be given as

$$dP(x) = -\alpha_s(\lambda)P(x)dx$$

where _s(_) is the photon absorption coefficient at a wavelength _. So that

$$P(x) = P_{in}\exp(-\alpha_s x)$$

Optical power absorbed, *P*(*x*), in the depletion region can be written in terms of incident optical power, *Pin* :

$$P(x) = P_{in}(1 - e^{-\alpha_s(\lambda)x})$$

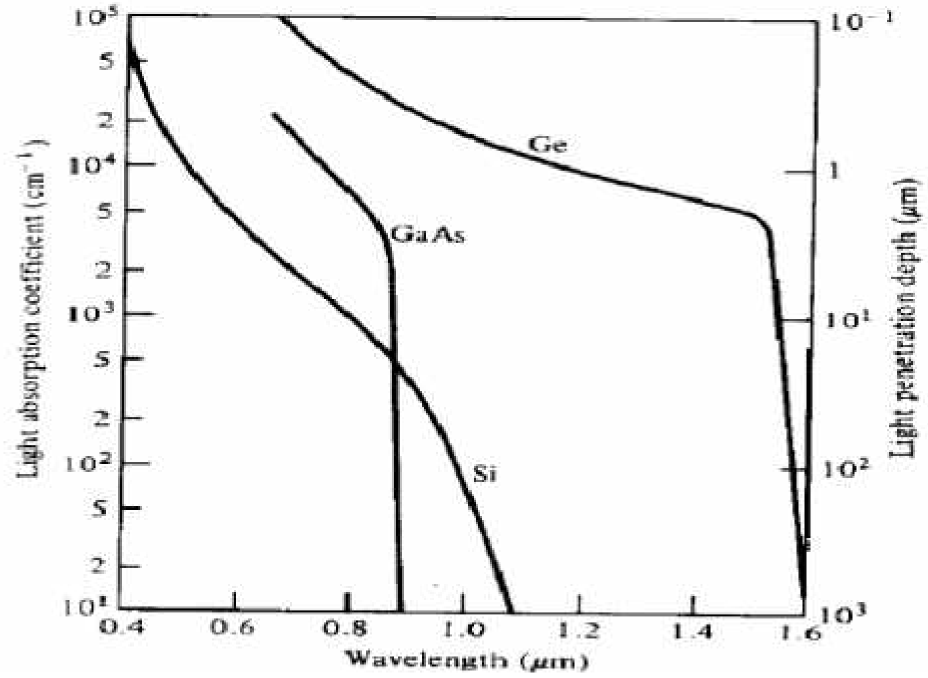
Absorption coefficient a*s* (l) strongly depends on wavelength. The upper wavelength cutoff for any semiconductor can be

$$\lambda_c(\mu\text{m}) = \frac{1.24}{E_g(\text{eV})}$$

Taking entrance face reflectivity into consideration, the absorbed power in the width of depletion region, *w*, becomes:

$$(1-R_f)P(w) = P_{in}(1-e^{-\alpha_s(\lambda)w})(1-R_f)$$

Optical Absorption Coefficient



The primary photocurrent resulting from absorption is:

$$\boxed{I_p = \frac{q}{h\nu} P_{in}(1 - e^{-\alpha_s(\lambda)w})(1 - R_f)}$$

Quantum Efficiency:

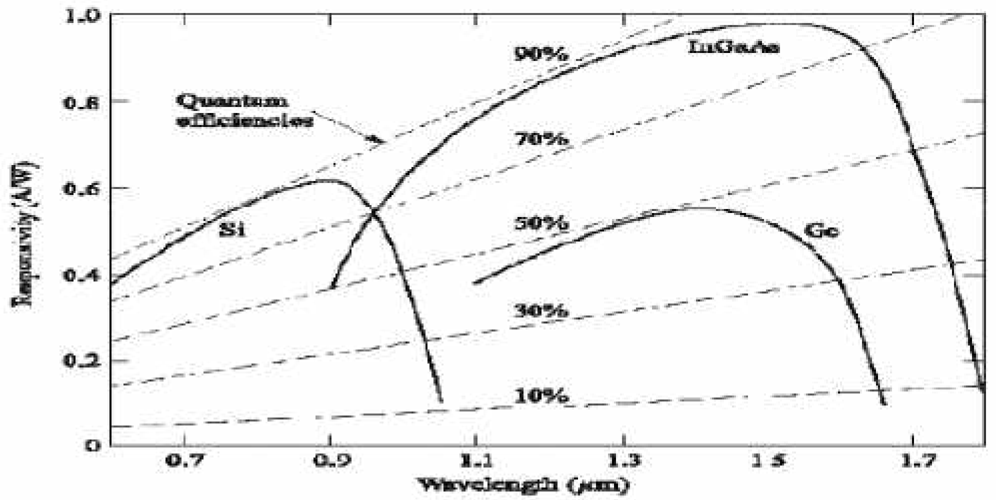
$$\eta = \frac{\text{\# of electron - hole photogener ated pairs}}{\text{\# of incident photons}}$$

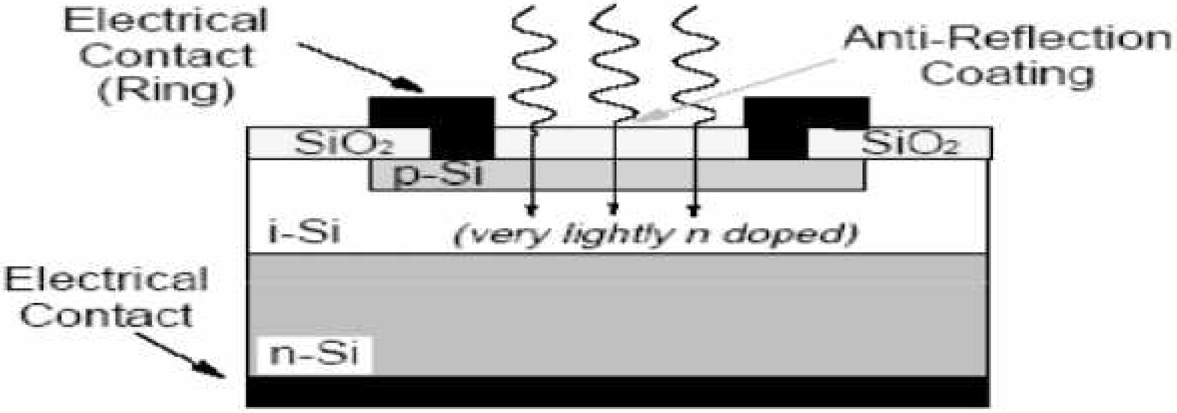
$$\boxed{\eta = \frac{I_p / q}{P_{in} / h\nu}}$$

Responsivity:

$$\boxed{\Re = \frac{I_p}{P_{in}} = \frac{\eta q}{h\nu} \quad [\text{A/W}]}$$

Responsivity vs. wavelength

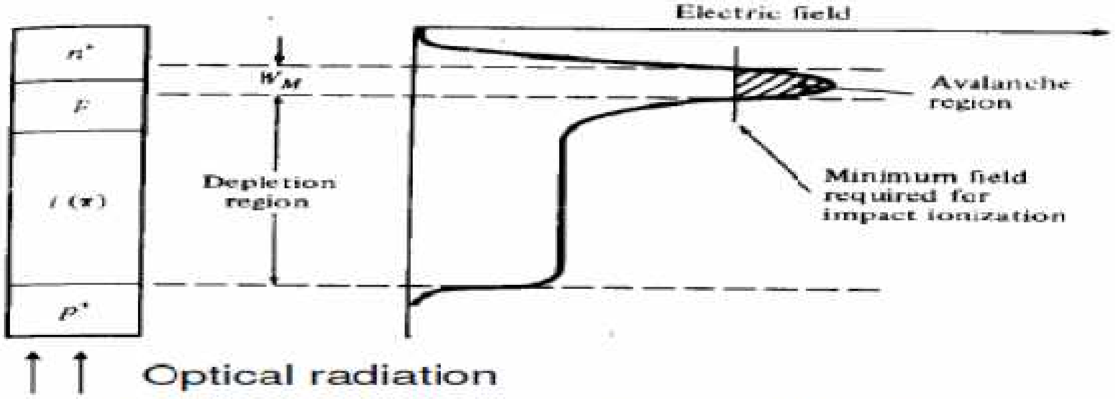




Typical Silicon P-I-N Diode Schematic

Avalanche Photodiode (APD):

APDs internally multiply the primary photocurrent before it enters to following circuitry. In order to carrier multiplication take place, the photogenerated carriers must traverse along a high field region. In this region, photogenerated electrons and holes gain enough energy toionize bound electrons in VB upon colliding with them. This multiplication is known as impact ionization. The newly created carriers in the presence of high electric field result in more ionization called avalanche effect.



Reach-Through APD structure (RAPD) showing the electric fields in depletion region and multiplication region.

Responsivity of APD:

The multiplication factor (current gain) *M* for all carriers generated in the photodiode is defined as:

$$M = \frac{I_M}{I_P}$$

where *IM* is the average value of the total multiplied output current & *Ip* is the primary photocurrent.

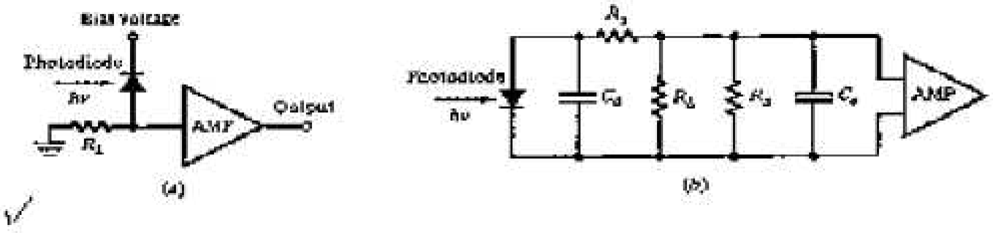
The responsivity of APD can be calculated by considering the current gain as:

$$\boxed{\Re_{\text{APD}} = \frac{\eta q}{h\nu} M = \Re_0 M}$$

Photodetector Noise & S/N:-

Detection of weak optical signal requires that the photodetector and its following amplification circuitry be optimized for a desired signal-to-noise ratio.

It is the noise current which determines the minimum optical power level that can be detected. This minimum detectable optical power defines the **sensitivity** of photodetector. That is the optical power that generates a photocurrent with the amplitude equal to that of the total noise current (*S/N=1*)



$$\frac{S}{N} = \frac{\text{signal power from photocurre nt}}{\text{photodetec tor noise power + amplifier noise power}}$$

$$\boxed{\frac{S}{N} = \frac{\langle i_P{}^2 \rangle M^2}{2q(I_P + I_D)BM^2 F(M) + 2qI_L B + 4k_B TB / R_L}}$$

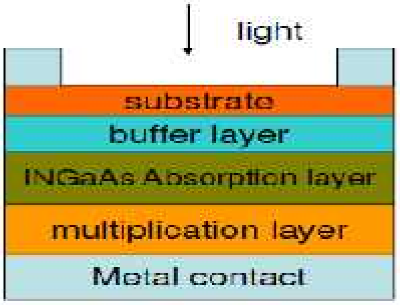
Since the noise figure $F(M)$ increases with M, there always exists an optimum value of M that maximizes the S/N. For sinusoidally modulated signal with $m=1$ and

$$F(M) \approx M^x$$

$$M_{\text{opt}}^{x+2} = \frac{2qI_L + 4k_B T / R_L}{xq(I_P + I_D)}$$

Structures for InGaAs APDs:-

Separate-absorption-and multiplication (SAM) APD



InGaAs APD superlattice structure (The multiplication region is composed of several layers of InAlGaAs quantum wells separated by InAlAs barrier layers.

Fiber Optic System Design:-

There are many factors that must be considered to ensure that enough light reaches the receiver. Without the right amount of light, the entire system will not operate properly.

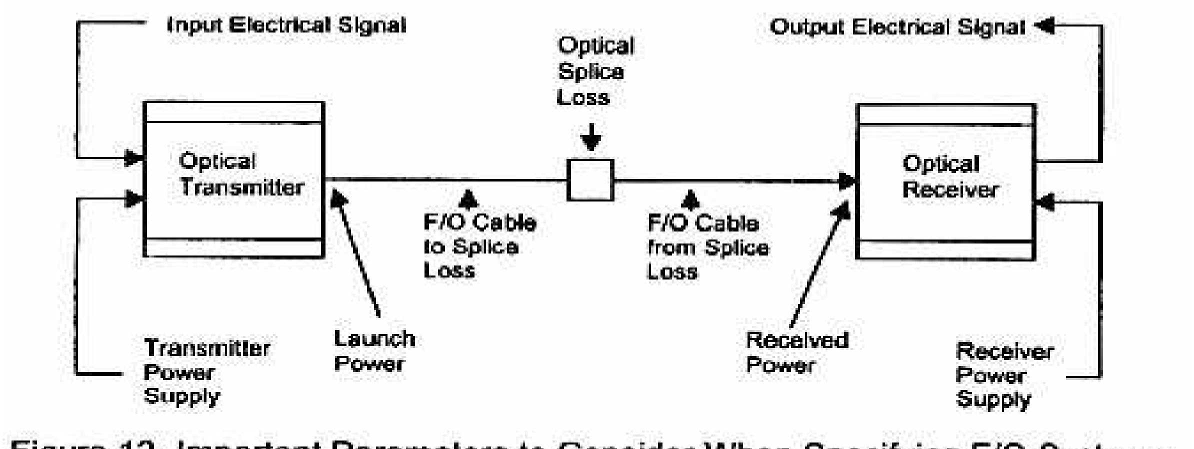


Figure 12, Important Parameters to Consider When Specifying F/O Systems

Fiber Optic System Design- Step-by-Step:-

Select the most appropriate optical transmitter and receiver combination based upon the signal to be transmitted

Determine the operating power available (AC, DC, etc.).

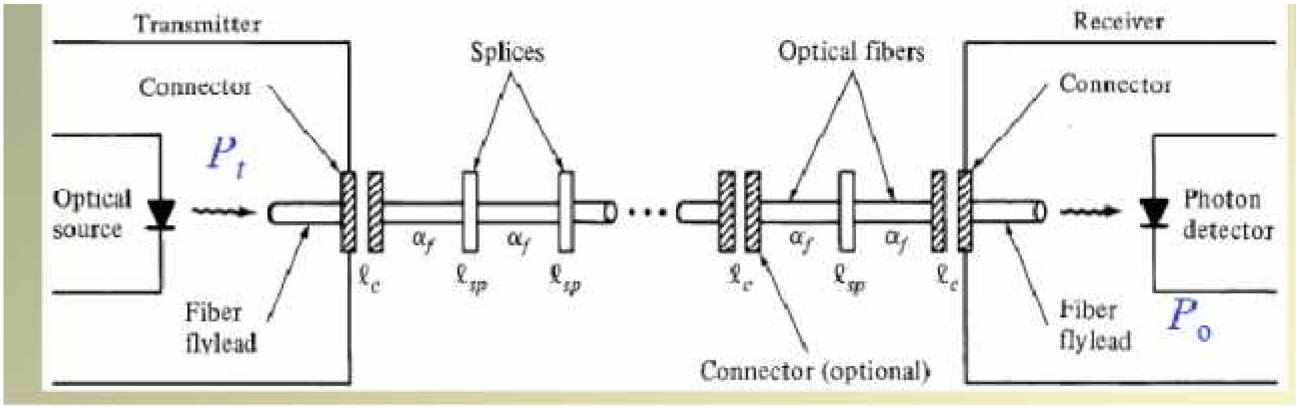
Determine the special modifications (if any) necessary (Impedances, bandwidths, connectors, fiber size, etc.).

Carry out system link power budget.

Carry out system rise time budget (I.e. bandwidth budget).

If it is discovered that the fiber bandwidth is inadequate for transmitting the required signal over the necessary distance, then either select a different transmitter/receiver (wavelength) combination, or consider the use of a lower loss premium fiber

Link Power Budget:-



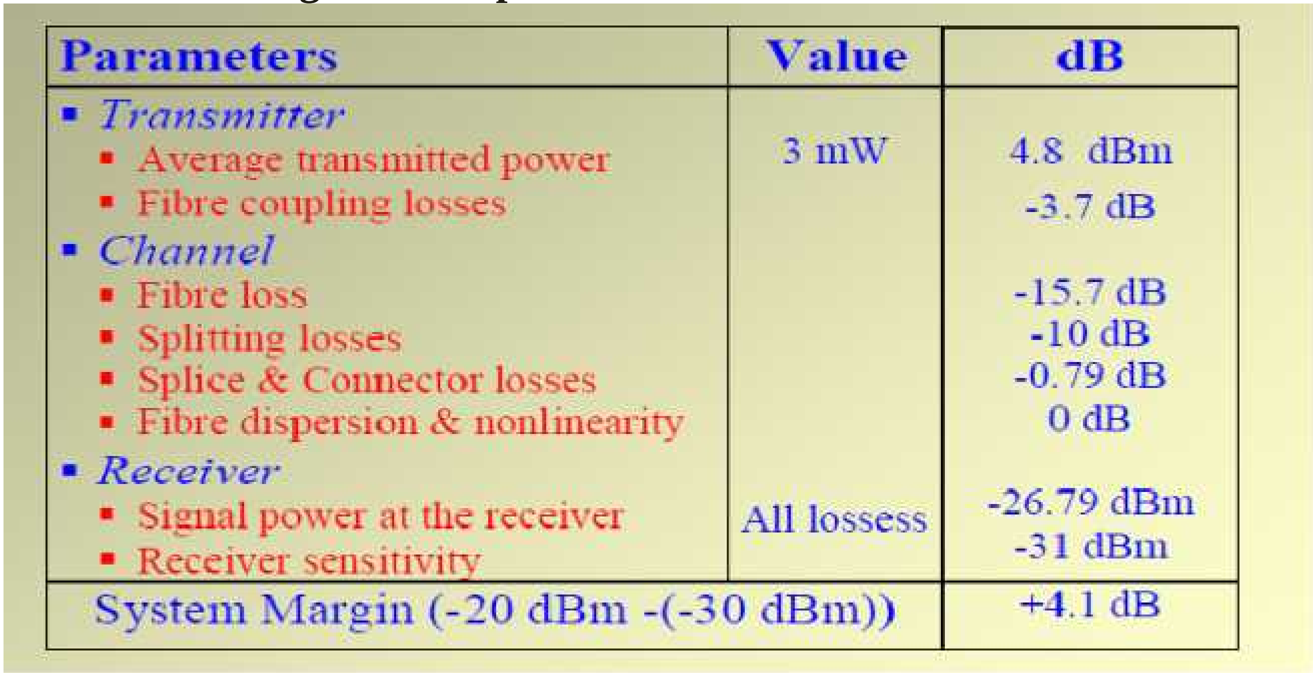
Total loss $LT = \alpha f\ L + lc + lsp$

$$Pt - Po = LT + SM$$

Po = Receiver sensitivity (i.e. minimum power requirement)

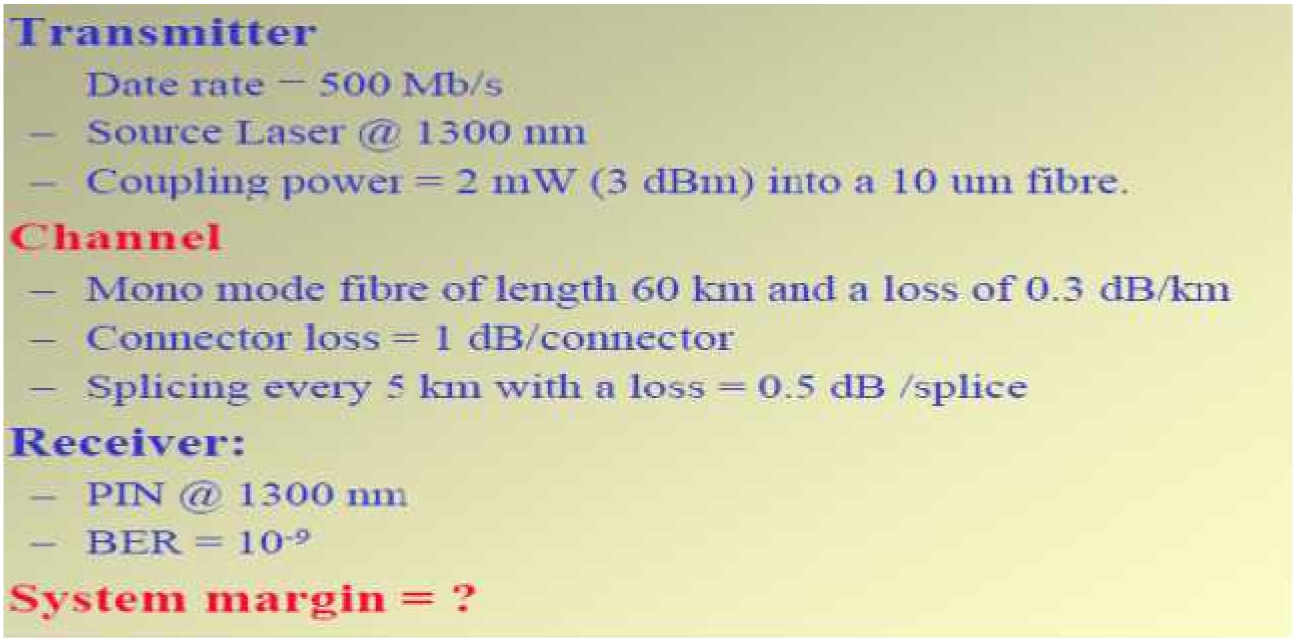
SM = System margin (to ensure that small variation the system operating parameters do not result in an unacceptable decrease in system performance)

Link Power Budget - Example 1:-



Parameters	Value	dB
▪ *Transmitter*		
▪ Average transmitted power	3 mW	4.8 dBm
▪ Fibre coupling losses		-3.7 dB
▪ *Channel*		
▪ Fibre loss		-15.7 dB
▪ Splitting losses		-10 dB
▪ Splice & Connector losses		-0.79 dB
▪ Fibre dispersion & nonlinearity		0 dB
▪ *Receiver*		
▪ Signal power at the receiver	All lossess	-26.79 dBm
▪ Receiver sensitivity		-31 dBm
System Margin (-20 dBm -(-30 dBm))		+4.1 dB

Link Power Budget - Example 2:-



Transmitter

Date rate – 500 Mb/s

– Source Laser @ 1300 nm

– Coupling power = 2 mW (3 dBm) into a 10 um fibre.

Channel

– Mono mode fibre of length 60 km and a loss of 0.3 dB/km

– Connector loss = 1 dB/connector

– Splicing every 5 km with a loss = 0.5 dB /splice

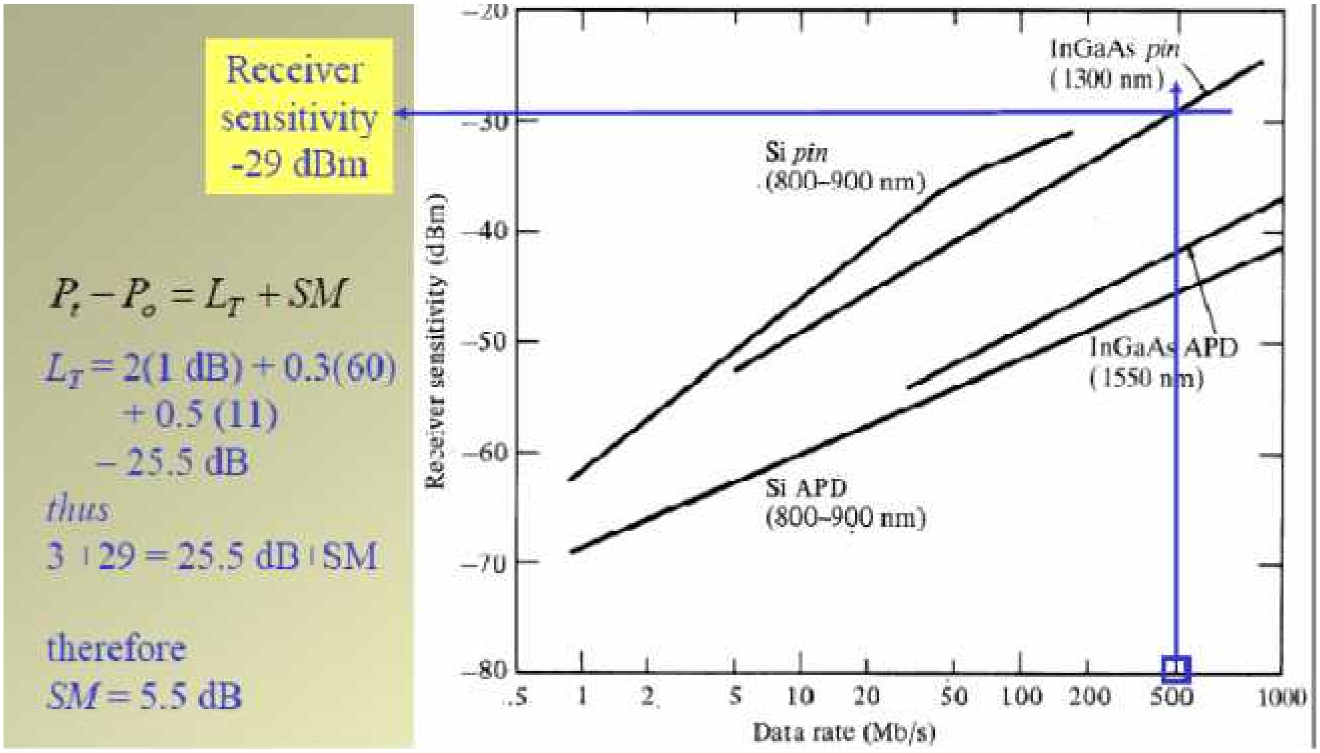
Receiver:

– PIN @ 1300 nm

– BER = 10^{-9}

System margin = ?

Link Power Budget - Example 2 *contd.:-*



Receiver sensitivity
-29 dBm

$$P_t - P_o = L_T + SM$$

$L_T = 2(1\text{ dB}) + 0.3(60)$
$+ 0.5\ (11)$
$- 25.5\text{ dB}$

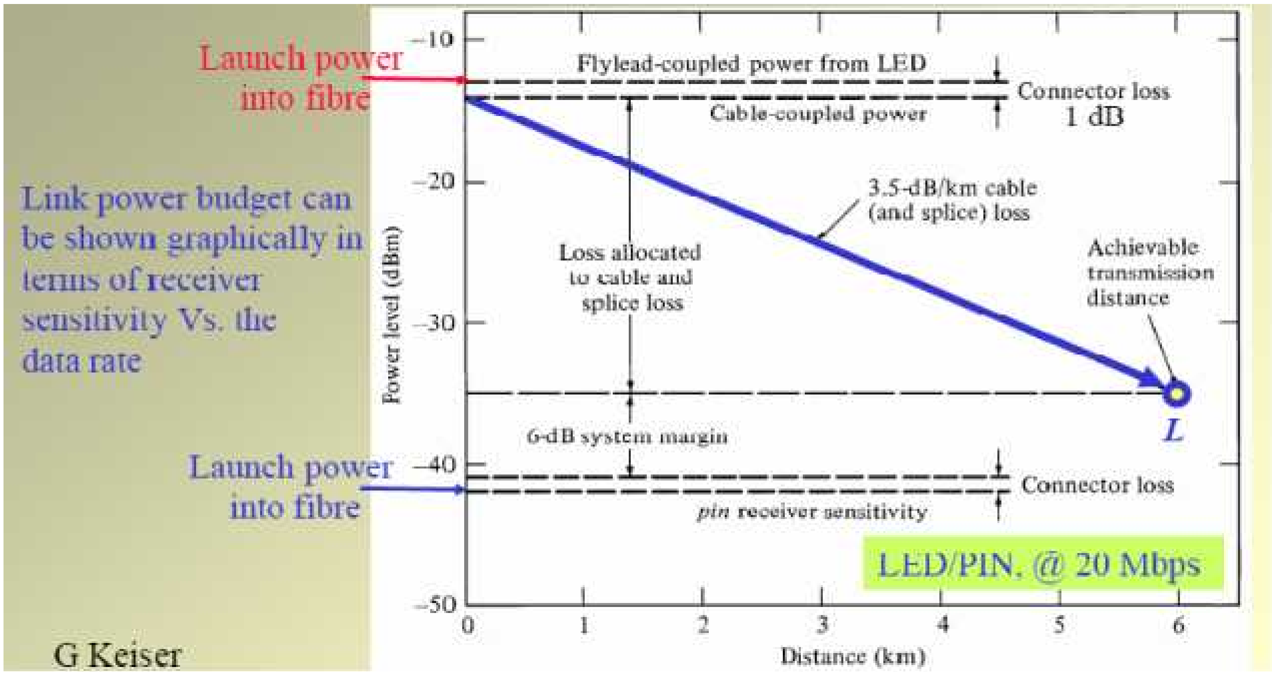
thus

3 ı 29 = 25.5 dB ı SM

therefore

$SM = 5.5$ dB

Link-Power Budget - Example 3:-



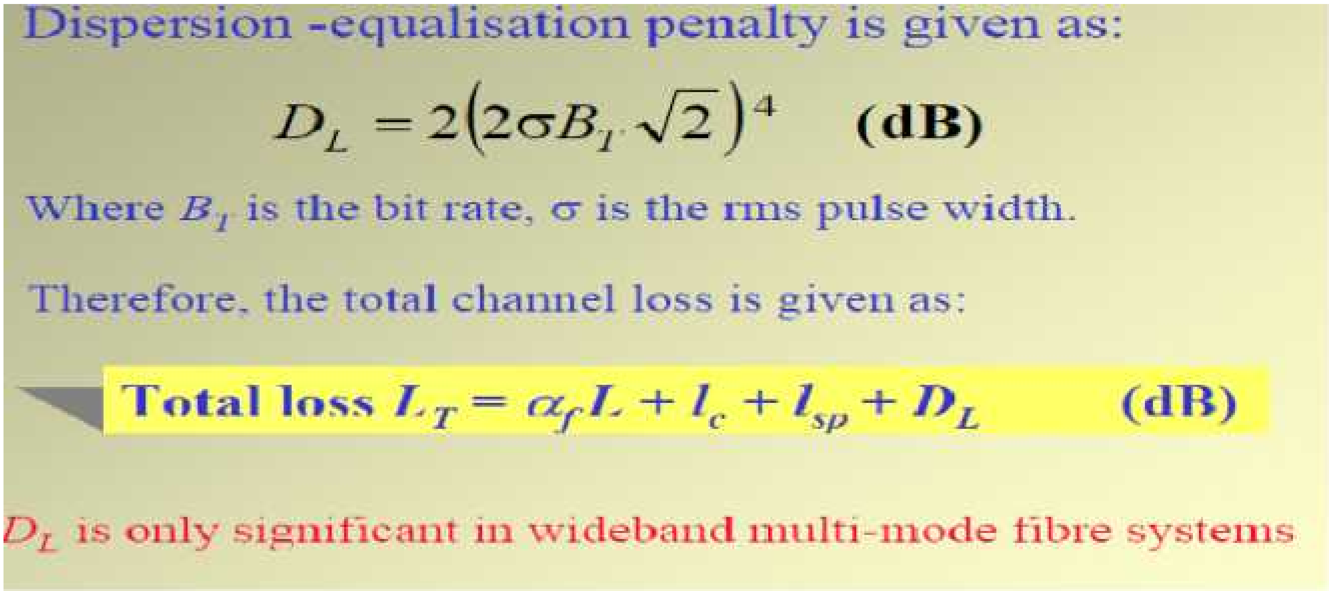
Launch power
into fibre

Link power budget can be shown graphically in terms of receiver sensitivity Vs. the data rate

Launch power
into fibre

LED/PIN, @ 20 Mbps

G Keiser



Dispersion -equalisation penalty is given as:

$$D_L = 2\left(2\sigma B_1 \sqrt{2}\right)^4 \quad \textbf{(dB)}$$

Where B_1 is the bit rate, σ is the rms pulse width.

Therefore, the total channel loss is given as:

Total loss $L_T = \alpha_f L + l_c + l_{sp} + D_L$ **(dB)**

D_L is only significant in wideband multi-mode fibre systems

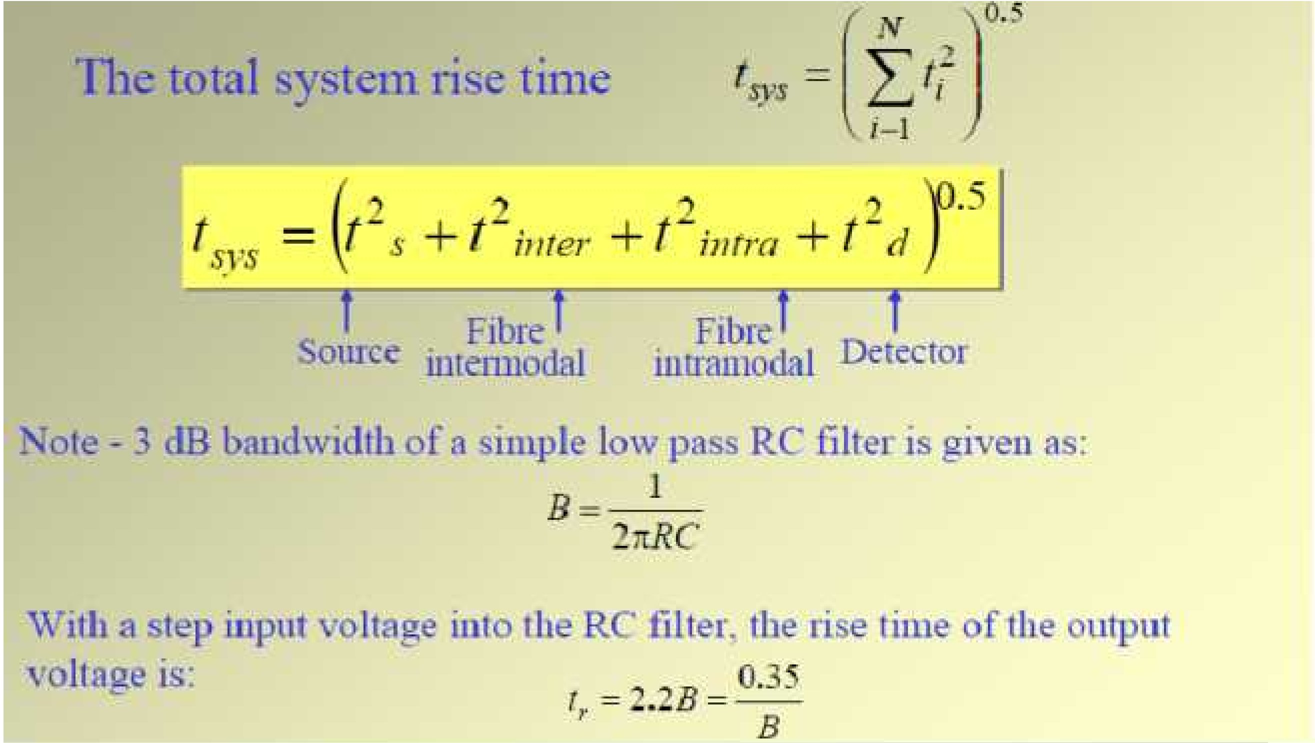
Rise Time Budget:-

The system design must also take into account the temporal response of the system components.

The total loss LT (given in the power budget section) is determined in the absence of the any pulse broadening due to dispersion.

Finite bandwidth of the system (transmitter, channel, receiver) may results in pulse spreading (i.e. intersymbol interference), giving a **reduction in the receiver sencitivity**. I.e. worsening of BER or SNR

The additional loss penalty is known as **dispersion equalisation or ISI penalty.**



The total system rise time $t_{sys} = \left(\sum_{i-1}^{N} t_i^2 \right)^{0.5}$

$$t_{sys} = \left(t^2{}_s + t^2{}_{inter} + t^2{}_{intra} + t^2{}_d \right)^{0.5}$$

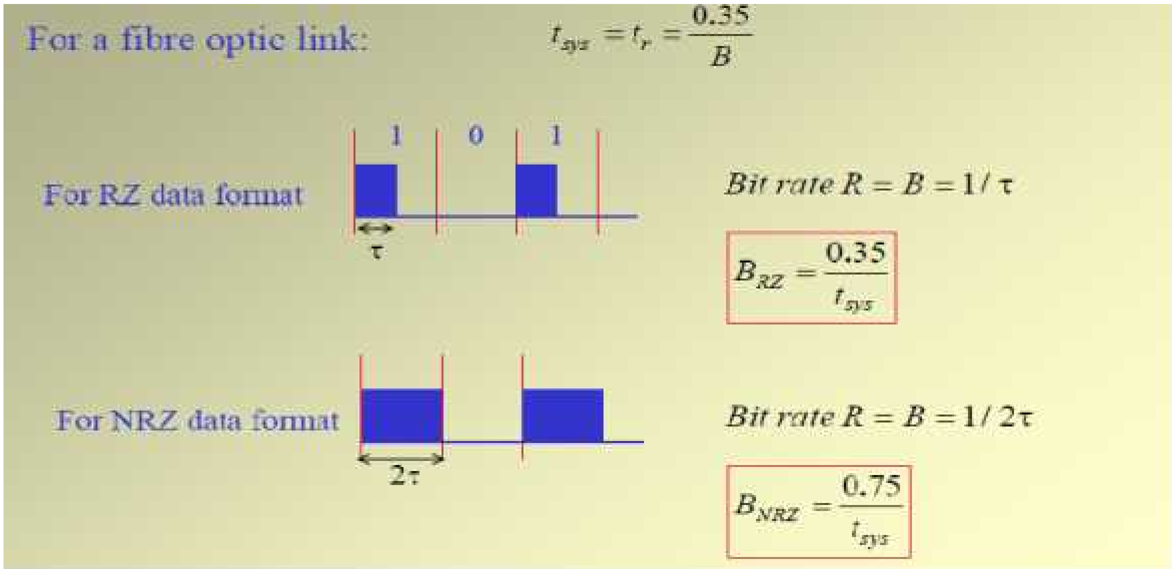
Source
Fibre intermodal
Fibre intramodal
Detector

Note - 3 dB bandwidth of a simple low pass RC filter is given as:

$$B = \frac{1}{2\pi RC}$$

With a step input voltage into the RC filter, the rise time of the output voltage is:

$$t_r = 2.2B = \frac{0.35}{B}$$



For a fibre optic link: $t_{sys} = t_r = \frac{0.35}{B}$

For RZ data format

$Bit\ rate\ R = B = 1/\tau$

$$\boxed{B_{RZ} = \frac{0.35}{t_{sys}}}$$

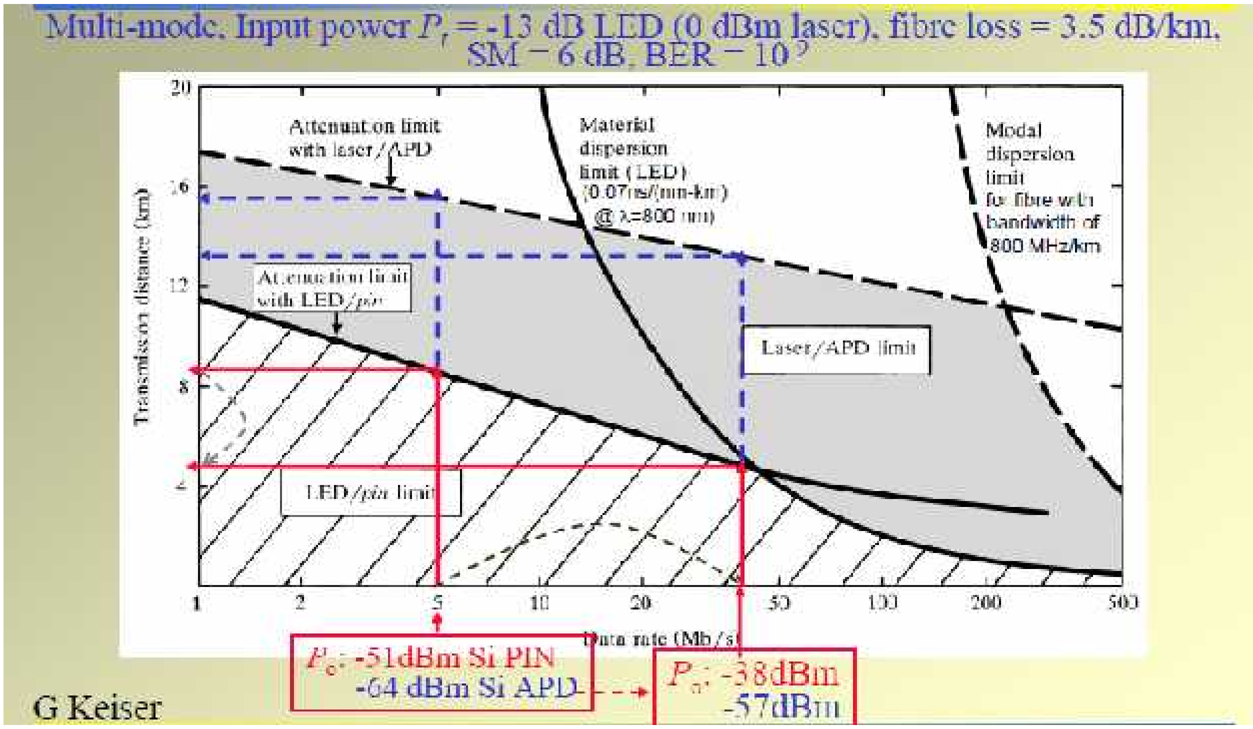
For NRZ data format

$Bit\ rate\ R = B = 1/2\tau$

$$\boxed{B_{NRZ} = \frac{0.75}{t_{sys}}}$$

Transmission Distance -1st window:-

Multi-mode, Input power P_t = -13 dB LED (0 dBm laser), fibre loss = 3.5 dB/km, SM − 6 dB, BER − 10^{-9}

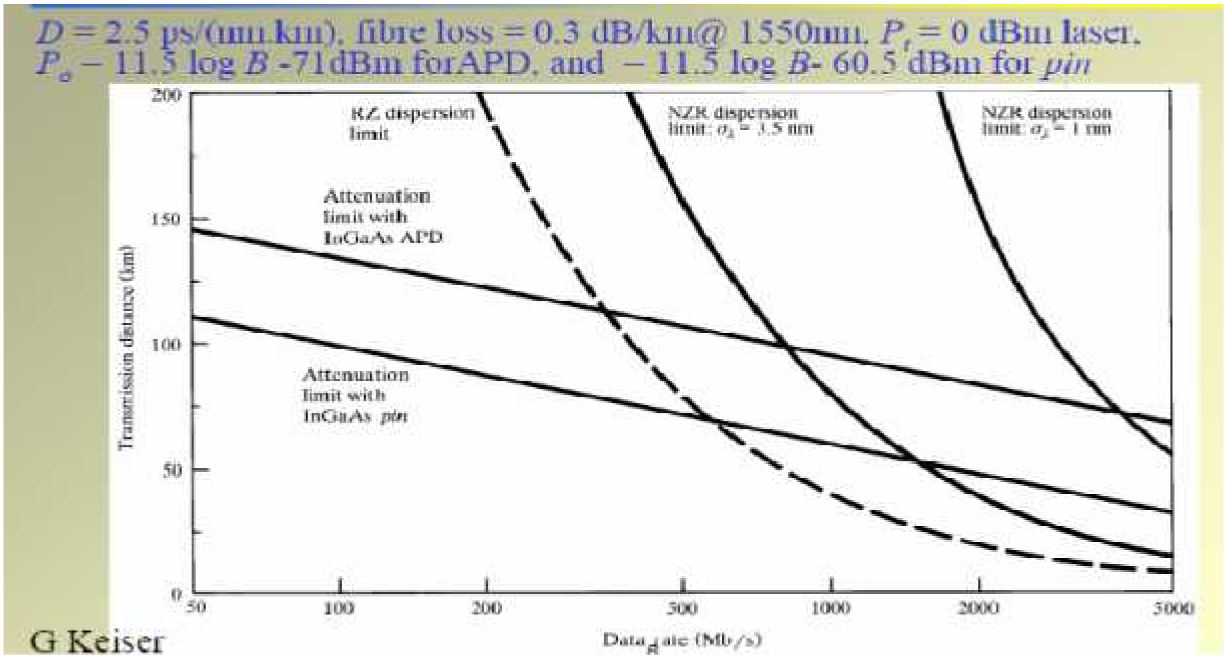


P_o: -51dBm Si PIN
-64 dBm Si APD

P_o: -38dBm
-57dBm

G Keiser

Transmission Distance -3rd window:-



$D = 2.5$ ps/(nm km), fibre loss = 0.3 dB/km@ 1550nm, $P_t = 0$ dBm laser, P_o – 11.5 log B -71dBm forAPD, and – 11.5 log B- 60.5 dBm for *pin*

G Keiser

Analogue System:-

The system must have sufficient bandwidth to pass the HIGEST FREQUENCIES. Link Power budget is the same as in digital systems Rise Time budget is also the same, except for the system bandwidth which is defined as:

$$B_{sys} = \frac{0.35}{t_{sys}}$$