Data Communications:

Data communications are the exchange of data between two devices via some form of transmission medium such as a wire cable as shown in **Fig1**.

Components:

A data communications system has five components:

1. Message. The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.
2. Sender. The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.
3. Receiver. The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.
4. Transmission medium. The transmission medium is the physical path by which a message travels from sender to receiver. Some examples of transmission media include twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves.
5. Protocol. A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating, just as a person speaking French cannot be understood by a person who speaks only Japanese.

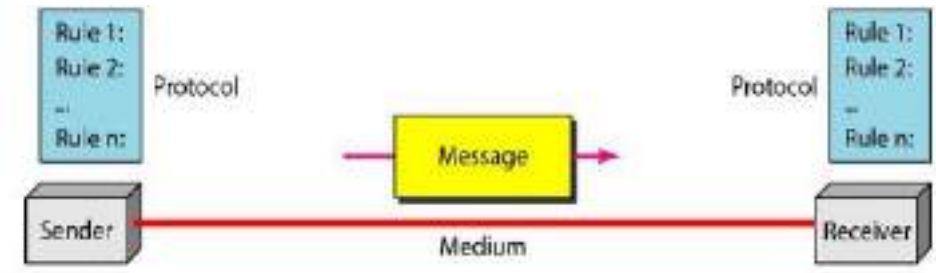


Fig 1: Components of Data Communication.

Direction of Data flow:

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

Simplex: The communication is unidirectional, as on a one-way street.

Only one of the two devices on a link can transmit; the other can only receive.

Keyboards and traditional monitors are examples of simplex devices.

The simplex mode can use the entire capacity of the channel to send data in one direction as shown in **Fig 2.a**

Half-Duplex: Each station can both transmit and receive, but not at the same time.

The half-duplex mode is like a one-lane road with traffic allowed in both directions.

The entire capacity of a channel is taken over by whichever of the two devices is transmitting at the time as shown in **Fig 2.b**

Walkie-talkies and CB (citizens band) radios are both half-duplex systems.

Full-Duplex:

Both stations can transmit and receive simultaneously.

Signals going in one direction share the capacity of the link with signals going in the other direction as shown in **Fig 2.c.**

One common example of full-duplex communication is the telephone network.

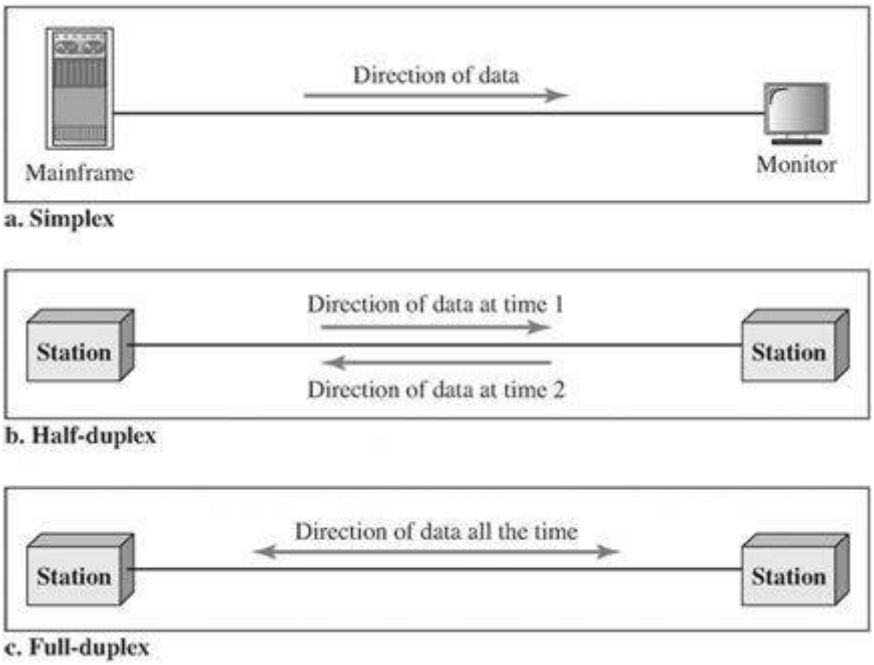


Fig 2: Data Flow

NETWORKS:

A network is a set of devices (often referred to as nodes) connected by communication links. A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.

Components:

Computer networks components comprise both physical parts as well as the software required for installing computer networks, both at organizations and at home. The hardware

components are the server, client, peer, transmission medium, and connecting devices. The software components are operating system and protocols.

The following figure shows a network along with its components −

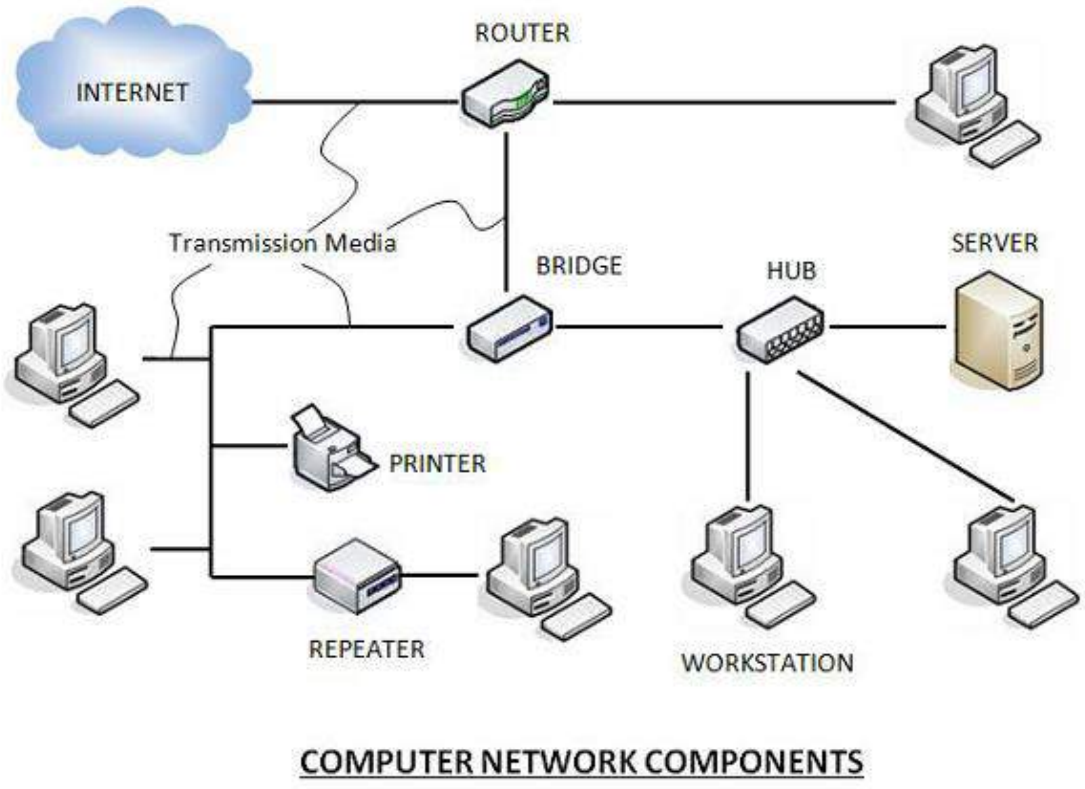


Fig 3: Network Components

Hardware Components

- **Servers** −Servers are high-configuration computers that manage the resources of the network. The network operating system is typically installed in the server and so they give user accesses to the network resources. Servers can be of various kinds: file servers, database servers, print servers etc.
- **Clients** − Clients are computers that request and receive service from the servers to access and use the network resources.
- **Peers** − Peers are computers that provide as well as receive services from other peers in a workgroup network.
- **Transmission Media** − Transmission media are the channels through which data is transferred from one device to another in a network. Transmission media may be guided media like coaxial cable, fibre optic cables etc; or maybe unguided media like microwaves, infra-red waves etc.

- **Connecting Devices** − Connecting devices act as middleware between networks or computers, by binding the network media together. Some of the common connecting devices are:

a. Routers

b. Bridges

c. Hubs

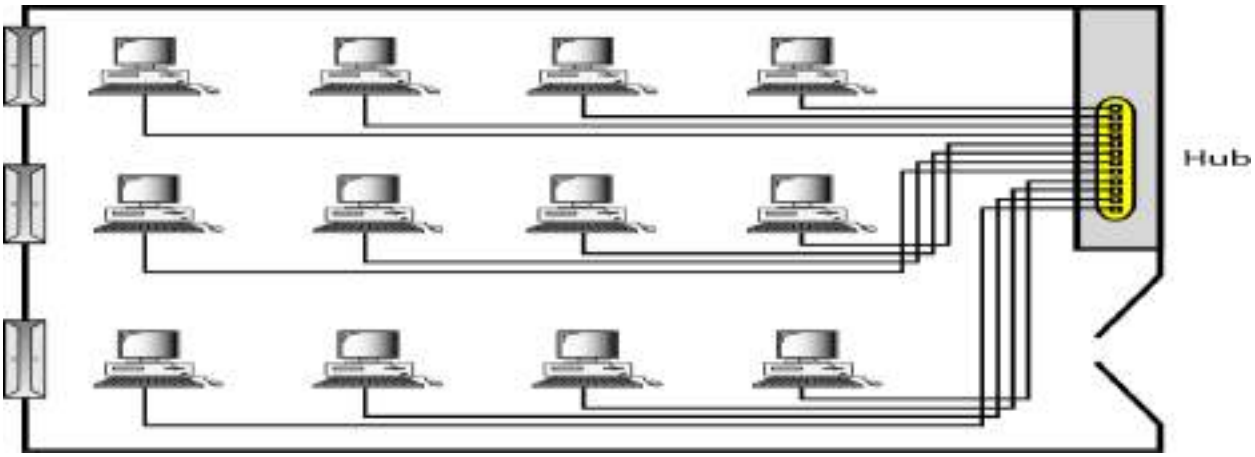
d. Repeaters

e. Gateways

f. Switches

Network Categories:

- A Local Area Network (LAN) provides short-distance transmission of data over small geographic areas that may comprise a single office, building, or campus.
- **Size:** LAN size is limited to a few kilometers.
- **Speed:** Early LANs had data rates in the 4 to 16 megabits per second (Mbps) range but now speeds are increased to 100 or 1000Mbps.
- LANs are designed to allow resources to be shared between personal computers or workstations.
- The resources to be shared can include hardware (e.g., a printer), software (e.g., an application program), or data.
- A local area network (LAN) is usually privately owned.
- LAN will use only one type of transmission medium.
- The most common LAN topologies are bus, ring, and star.

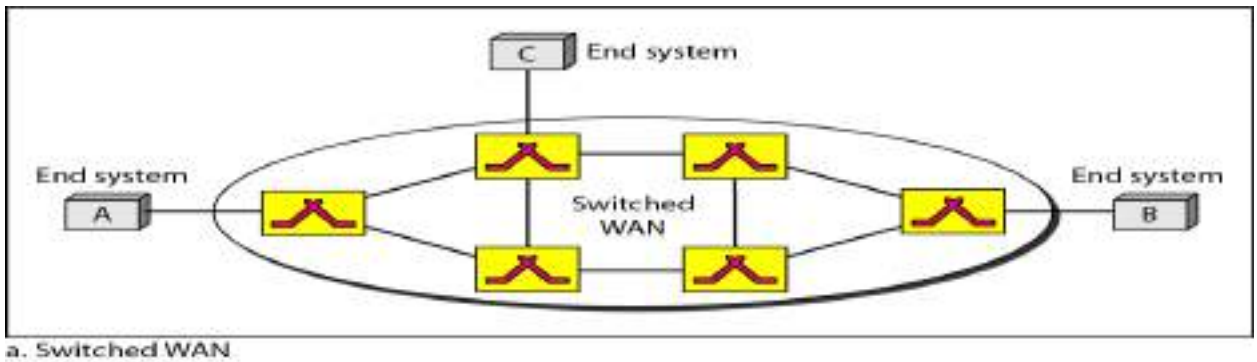


Wide Area Network

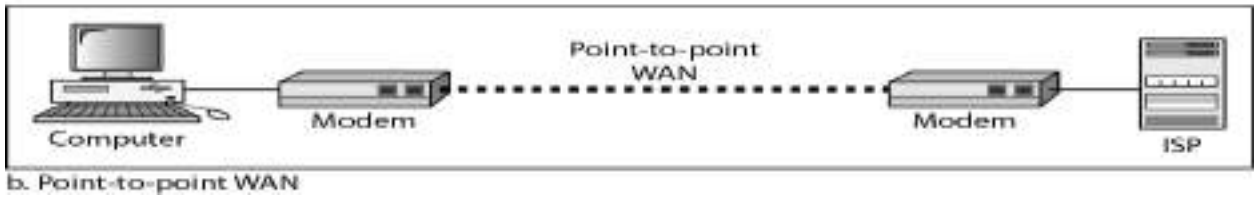
A Wide Area Network (WAN) provides long-distance transmission of data, image, audio, and video information over large geographic areas that may comprise a country, a continent, or even the whole world.

The switched WAN connects the end systems, which usually comprise a router (inter- networking connecting device) that connects to another LAN or WAN.

The point-to-point WAN is often used to provide Internet access. A line leased from a telephone provider that connects a home computer or a s mall LAN to an Internet service provider (lSP).



a. Switched WAN



b. Point-to-point WAN

Metropolitan Area Networks

A Metropolitan Area Network (MAN) is a network with a size between a LAN and a WAN. It normally covers the area inside a town or a city.

It is designed for customers who need a high-speed connectivity to the Internet, and have endpoints spread over a city or part of city.

Example of a MAN is the part of the telephone company network that can provide a high- speed DSL line to the customer.

Type of Connection:

A network is two or more devices connected through links.

A link is a communications pathway that transfers data from one device to another.

There are two possible types of connections: point-to-point and multipoint.

1. **Point-to-Point**: A point-to-point connection provides a dedicated link between two devices. The entire capacity of the link is reserved for transmission between those two devices as shown in **Fig 3.a**.
   When you change television channels by infrared remote control, you are establishing a point-to-point connection between the remote control and the television's control system.
2. **Multipoint**: A multipoint (also called multidrop) connection is one in which more than two specific devices share a single link as shown in **Fig 3.b**.
   In a multipoint environment, the capacity of the channel is shared, either spatially or temporally. If several devices can use the link simultaneously, it is a spatially shared connection. If users must take turns, it is a timeshared connection.

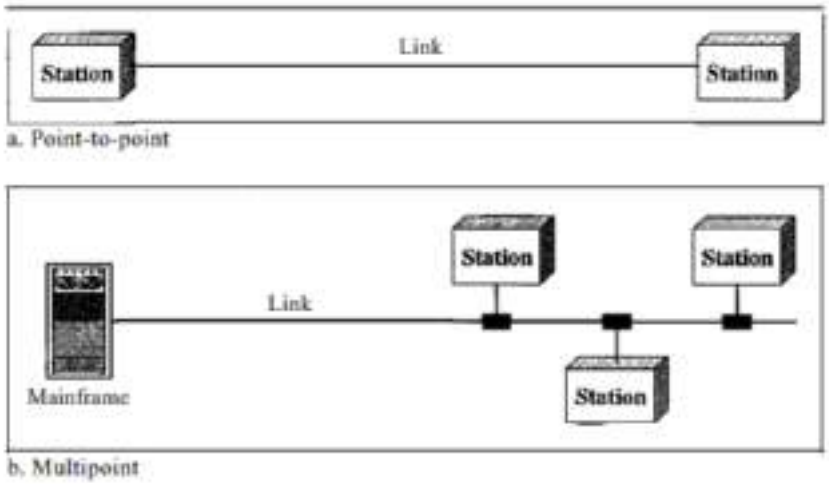


Fig 1: Type of Connection

Physical Topology: The term physical topology refers to the way in which a network is laid out physically. Two or more devices connect to a link; two or more links form a topology.

The topology of a network is the geometric representation of the relationship of all the links and linking devices (usually called nodes) to one another.

There are four basic topologies possible: mesh, star, bus, and ring.

1. **Mesh**:

In a mesh topology, every device has a dedicated point-to-point link to every other device. The term dedicated means that the link carries traffic only between the two devices it connects as shown in **Fig 4**.

In a mesh topology, we need n(n -1) /2 duplex-mode links.

To accommodate that many links, every device on the network must have n - 1 input/output ports to be connected to the other n - 1 stations.

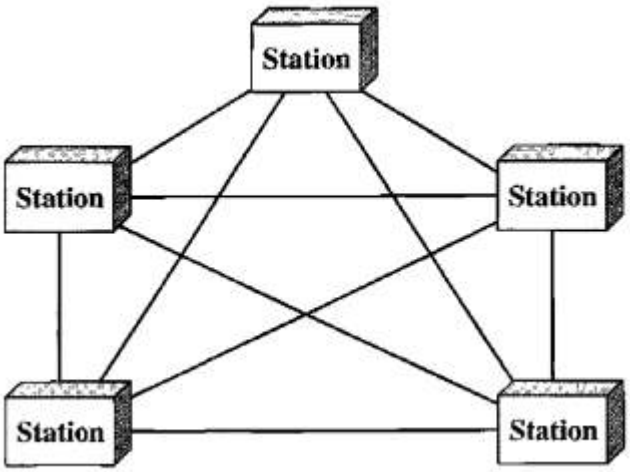


Fig 2: Mesh Topology

Advantages:

1. Eliminating the traffic problems.
2. Robust.
3. Privacy or security.
4. Fault Isolation is easy.
5. High Performance.

Disadvantages:

1. Complexity
2. High Cost
3. Limited scalability
4. Redundancy
5. Bandwidth Issues

2. **Star**:

In a star topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub.

The devices are not directly linked to one another as shown in **Fig 5.**

The controller acts as an exchange: If one device wants to send data to another, it sends the data to the controller, which then relays the data to the other connected device.

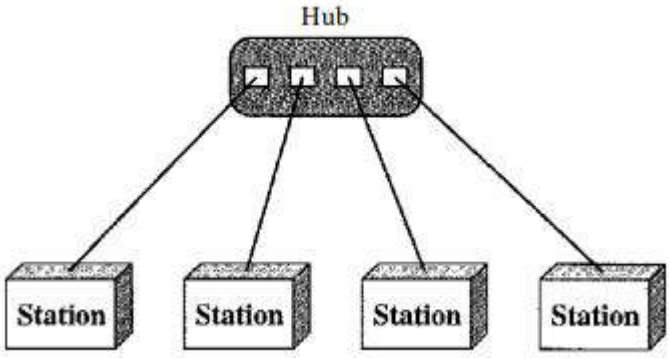


Fig 3: Star Topology

Advantages:

1. Less expensive
2. Easy to install and reconfigure.
3. Less cabling
4. Robustness.
5. Easy fault identification and fault isolation.

Disadvantages:

1. Dependency of the whole topology on one single point, the hub. If the hub goes down, the whole system is dead.

3. **Bus**:

A bus topology, on the other hand, is multipoint. One long cable acts as a backbone to link all the devices in a network.

Nodes are connected to the bus cable by drop lines and taps as shown in **Fig 6.**

A drop line is a connection running between the device and the main cable.

A tap is a connector that either splices into the main cable or punctures the sheathing of a cable to create a contact with the metallic core.

As a signal travels along the backbone, some of its energy is transformed into heat. Therefore, it becomes weaker and weaker as it travels farther and farther.

For this reason there is a limit on the number of taps a bus can support and on the distance between those taps.

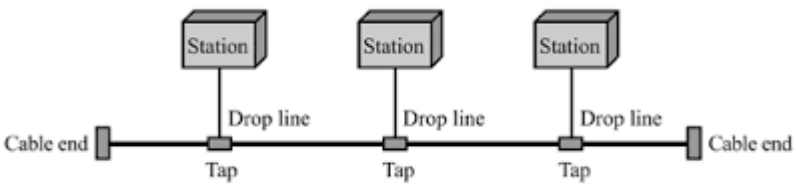


Fig 4: Bus Topology

Advantages:

1. Ease of installation.

Disadvantages:

1. Difficult reconnection and fault isolation.
2. A fault or break in the bus cable stops all transmission.

3.Ring Topology: In a ring topology, each device has a dedicated point-to-point connection with only the two devices on either side of it.

A signal is passed along the ring in one direction, from device to device, until it reaches its destination.

Each device in the ring incorporates a repeater.

When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along as shown in **Fig 7.**

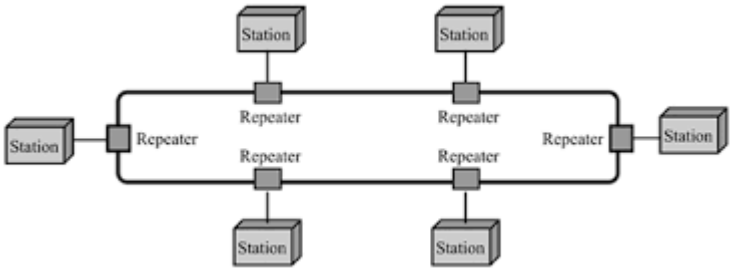


Fig 5: Ring Topology

Advantages:

1. Easy to install and reconfigure.
2. fault isolation is simplified.

Disadvantages:

1. Unidirectional traffic.

4.Hybrid Topology: A network can be hybrid. For example, we can have a main star topology with each branch connecting several stations in a bus topology as shown in **Fig 8**.

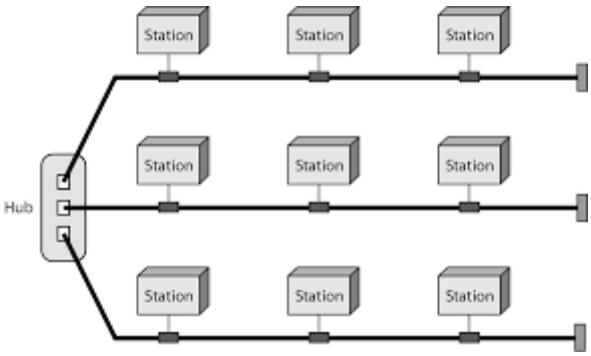


Fig 6: Hybrid Topology

Protocols and Standards:

Protocols: A protocol is a set of rules that govern data communications. A protocol defines what is communicated, how it is communicated, and when it is communicated. For communication to occur, the entities must agree on a protocol.

The key elements of a protocol are: Syntax, Semantics, Timing.

- Syntax refers to the structure or format of the data, meaning the order in which they are presented.
- Semantics refers to the meaning of each section of bits. How are a particular pattern to be interpreted, and what action is to be taken based on that interpretation?
- Timing refers to two characteristics: when data should be sent and how fast they can be sent.

Standards: Standards are essential in creating and maintaining an open and competitive market for equipment manufacturers and in guaranteeing national and international interoperability of data and telecommunications technology and processes.

Data communication standards fall into two categories:

1) de facto (meaning "by fact" or "by convention") and

2) de jure (meaning "by law" or "by regulation").

Some Data Communication Standards are:

1. International Organization for Standardization (ISO).
2. Telecommunication Union-Telecommunication Standards Sector (ITU-T).
3. Committee for International Telegraphy and Telephony (CCITT).
4. American National Standards Institute (ANSI).
5. Institute of Electrical and Electronics Engineers (IEEE).
6. Electronic Industries Association (EIA).

ISO/OSI MODEL:

This model is based on a proposal developed by the International Standards Organization (ISO) as a first step toward international standardization of the protocols used in the various layers

The model is called the ISO OSI (Open Systems Interconnection) Reference Model because it deals with connecting open systems—that is, systems that are open for communication with other systems.

The OSI model has seven layers.

The **principles** that were applied to arrive at the seven layers can be summarized as follows:

1. A layer should be created where a different abstraction is needed.

2. Each layer should perform a well-defined function.

3. The function of each layer should be chosen with an eye toward defining internationally standardized protocols.

4. The layer boundaries should be chosen to minimize the information flow across the interfaces.

5. The number of layers should be large enough that distinct functions need not be thrown together in the same layer out of necessity and small enough that the architecture does not become unwieldy.

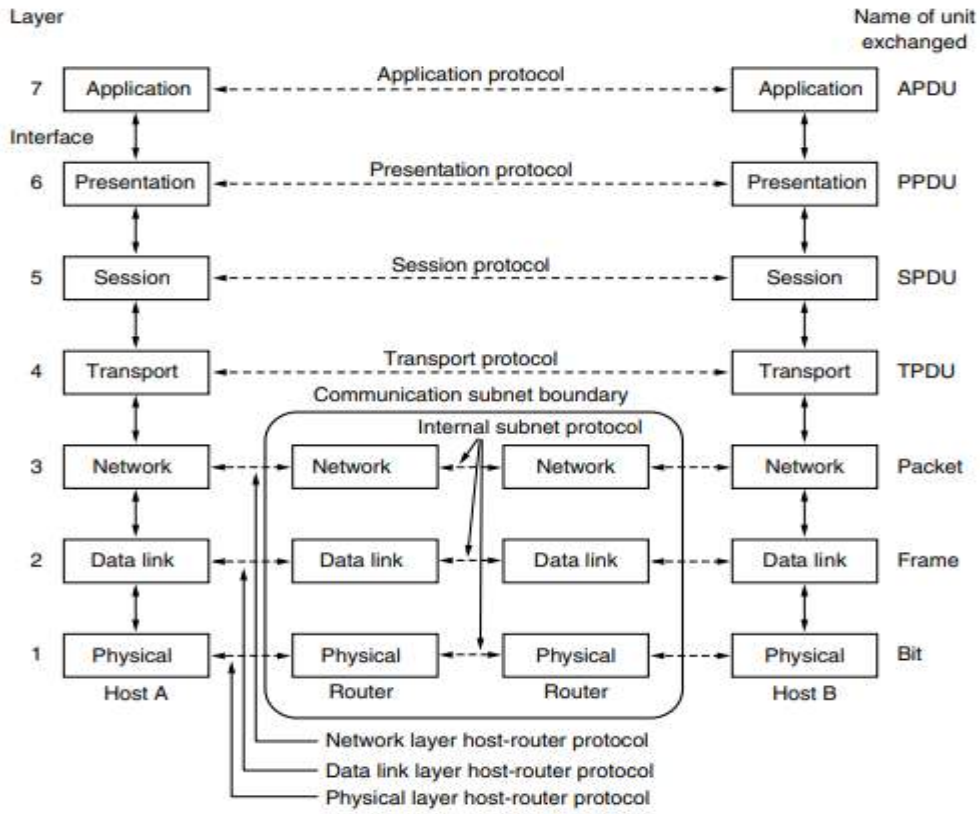


Fig 1: OSI Model

NOTE:

- Layers 1-4 relate to communications technology.
- Layers 5-7 relate to user applications.

Layer 1: Physical Layer

- Transmits bits from one computer to another
- Regulates the transmission of a stream of bits over a physical medium.
- Defines how the cable is attached to the network adapter and what transmission technique is used to send data over the cable. Deals with issues like
  - The definition of 0 and 1, e.g. how many volts represents a 1, and how long a bit lasts?
  - Whether the channel is simplex or duplex?
  - How many pins a connector has, and what the function of each pin is?

Layer 2: Data Link Layer

- Packages raw bits from the Physical layer into frames (logical, structured packets for data).
- Provides reliable transmission of frames
  - It waits for an acknowledgment from the receiving computer.
  - Retransmits frames for which acknowledgement not received

Layer 3: Network Layer

- Manages addressing/routing of data within the subnet
  - Addresses messages and translates logical addresses and names into physical addresses.
  - Determines the route from the source to the destination computer
  - Manages traffic problems, such as switching, routing, and controlling the congestion of data packets.
- Routing can be:
  - Based on static tables
  - determined at start of each session

- Individually determined for each packet, reflecting the current network load.

Layer 4: Transport Layer

- Manages transmission packets
  - Repackages long messages when necessary into small packets for transmission
  - Reassembles packets in correct order to get the original message.
- Handles error recognition and recovery.
  - Transport layer at receiving acknowledges packet delivery.

  Resends missing packets

Layer 5 : Session Layer

- Allows two applications on different computers to establish, use, and end a session.
  - e.g. file transfer, remote login
- Establishes dialog control
  - Regulates which side transmits, plus when and how long it transmits.
- Performs *token management* and *synchronization*.

Layer 6: Presentation Layer

- Related to representation of transmitted data
  - Translates different data representations from the Application layer into uniform standard format
- Providing services for secure efficient data transmission
  - e.g., data encryption, and data compression.

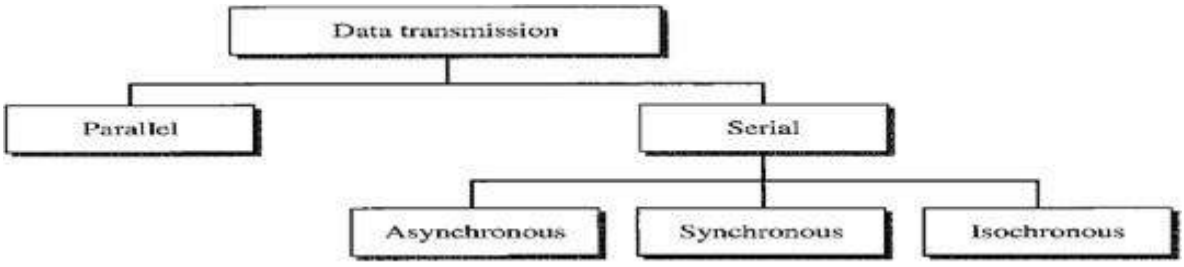
Layer 7: Application Layer

- Level at which applications access network services.
  - Represents services that directly support software applications for file transfers, database access, and electronic mail etc.

TRANSMISSION MODES

Transmission modes are two types:

1. Parallel Transmission
2. Serial Transmission



Parallel Transmission

Parallel Transmission is defined as sending n bits of data at a time instead of transmitting one bit at a time.

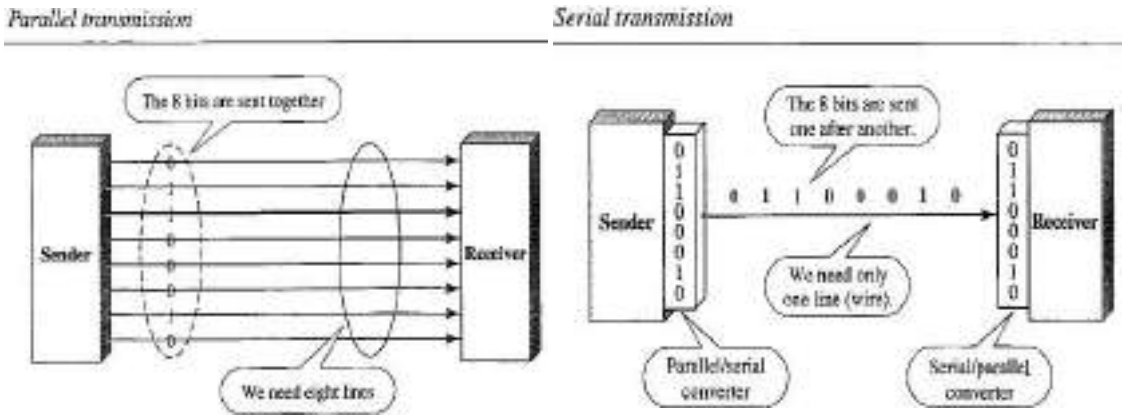
The mechanism for parallel transmission is a conceptually simple one: Use **n-wires** to send **n-bits** at one time.

Advantage: Speed of the transmission is increased.

Disadvantage : Cost of equipment is increased for this reason parallel transmission is usually limited to short distances.

Serial Transmission

In serial transmission one bit follows another, so we need only one communication channel rather than **n channels** to transmit data between two communicating devices



Advantage: Reduces the cost transmission equipment because we need only one communication channel.

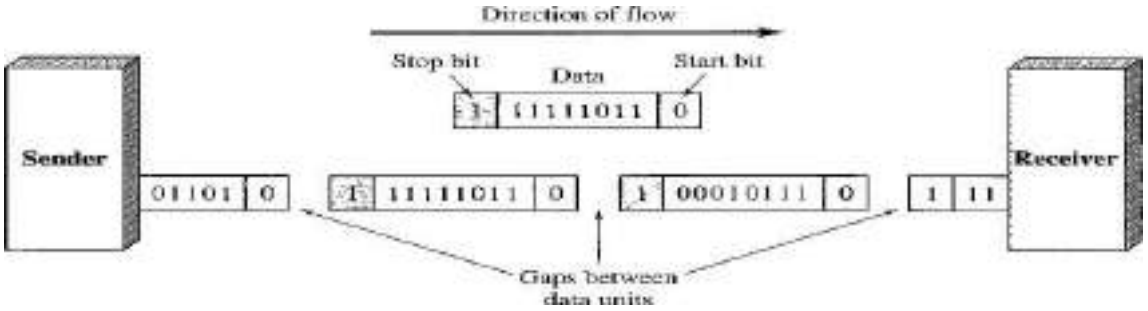
Since communication within devices is parallel, conversion devices are required at the interface between the sender and the line (parallel- to-serial) and between the line and the receiver (serial-to-parallel).

Serial transmission categorized into 3 types:

1. Asynchronous Transmission
2. Synchronous Transmission
3. Isochronous Transmission

Asynchronous Transmission

- The timing of signal is not important in Asynchronous transmission. Information is received and translated by agreed upon patterns.
- As long as those patterns are followed, the receiving device can retrieve the information without regard to the order in which it is sent.
- Patterns are based on grouping the bit stream into bytes. Each group contains 8 bits is sent along the link as a unit.



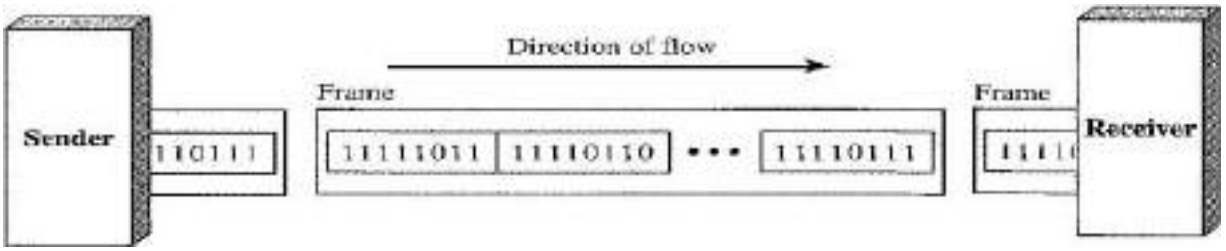
- In asynchronous transmission, we send one start bit (0) at the beginning and one or more stop bits (1's) at the end of each byte. There may be a gap between each byte.
- The start and stop bits are used because the sending system handles each group independently whenever the group is ready it will transmitted through the link.
- Without synchronization, the receiver cannot use timing to predict when the next group will arrive.
- To alert the receiver to the arrival of a new group the extra bits 0 and 1 are added.
- At the receiver side when the receiver detects a start bit, it sets a timer and begins counting bits as they come in. After *n* bits, the receiver looks for a stop bit. As soon as it detects the stop bit, it waits until it detects the next start bit
- **Start** and **Stop** bits and the **Gap** alert the receiver to the beginning and end of each byte and allow it to synchronize with data stream. This mechanism is called **Asynchronous.**
- The transmission is slow because of addition of start, stops and gaps between bit streams. Hence it is used for low speed communications.
- Example: The connection to the keyboard to the computer is application of Asynchronous transmission.
- Apart from slower transmission Asynchronous transmission is cheap and effective.

Synchronous Transmission

In synchronous transmission, we send bits one after another without start or stop bits or gaps. It is the responsibility of the receiver to group the bits.

That means:

- The bit stream is combined into longer "**Frames**," which may contain multiple bytes.
- Each byte is introduced onto the transmission link without a gap between the byte and the next byte.
- It is left to the receiver to separate the bit stream into bytes for decoding purposes.
- Data are transmitted as an unbroken string of 1s and 0's, and the receiver separates that string into the bytes, or characters and receiver needs to reconstruct the information.



In synchronous transmission **Timing** plays very crucial role. When the information comes from sender, the receiving device **accurately count the bits** and group them into 8 bits because we don't have any extra bits to identify starting and ending of byte. This process is called **Byte Synchronization**.

Advantage: Speed of the transmission is increased as compared to Asynchronous transmission because there are no extra bits to be add or remove at the sender side and receiver side respectively.

It is useful for **High Speed Application** such as transmission of data from one computer to another computer.

Note:

1. Byte Synchronization is accomplished at Receiver side.
2. Although there is no gap between characters in synchronous serial transmission, there may be uneven gaps between frames.

Isochronous Transmission

- The isochronous transmission guarantees that the data arrive at a fixed rate.
- In real- time audio and video, in which synchronous transmission fails such as uneven delays between frames, are not acceptable.
- **For example**, TV images are broadcast at the rate of 30 images per second; they must be viewed at the same rate. If each image is sent by using one or more frames, there should be no delays between frames.
- For this type of application, synchronization between characters is not enough; the entire stream of bits must be synchronized.

Multiplexing is the set of techniques that allows the simultaneous transmission of multiple signals across a single data link.

As data and telecommunications use increases the data traffic is also increases.

We can accommodate this increase by continuously adding the individual links each time a new channel is needed, or we can install higher-bandwidth links and use each to carry multiple signals.

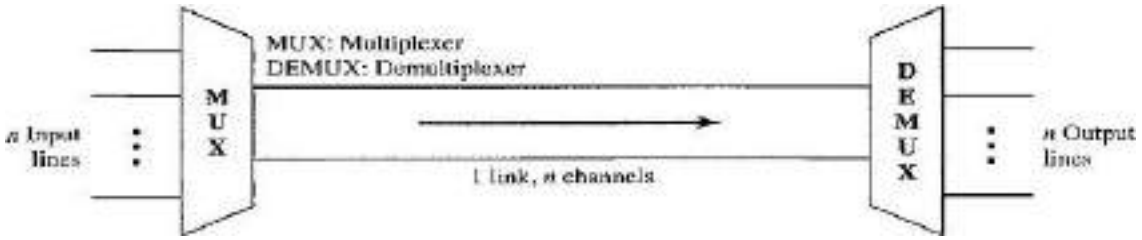


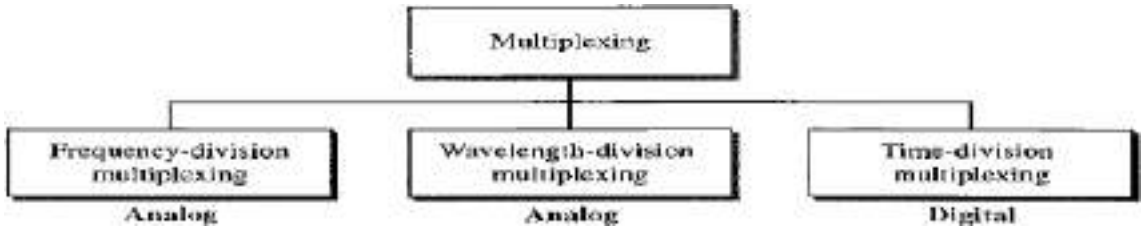
Fig: Dividing the link into channels

In a multiplexed system, *n* lines share the bandwidth of one link.

- **Link** refers to the physical path.
- **Channel** refers to the portion of a link that carries a transmission between a given pair of lines.
- The lines on the left direct their transmission streams to a **Multiplexer (MUX)**, which combines them into a single stream (many-to-one).
- At the receiving end, that stream is fed into a **Demultiplexer (DEMUX)**, which separates the stream back into its component transmissions (one-to- many) and directs them to their corresponding lines.

Multiplexing is categorized into 3 types:

1. Frequency Division Multiplexing
2. Wavelength Division Multiplexing
3. Time Division Multiplexing



Frequency Division Multiplexing(FDM)

FDM is an analog multiplexing technique that combines analog signals.

That means:

- FDM is an analog technique that can be applied when:
  Bandwidth of link (in Hz) >= Combined bandwidth of the signal to be transmitted.
- In FDM, signals generated by each sending device modulate different carrier frequencies.
- These modulated signals are then combined into a single composite signal that can be transported by the link.
- **Carrier frequencies** are separated by sufficient bandwidth to accommodate the modulated signal.
- These bandwidth ranges are the channels through which the various signals travel.
- **Channels** can be separated by strips of unused bandwidth called **Guard Bands.**
- **Guard bands** are used to prevent signals from overlapping.
- In addition, carrier frequencies must not interfere with the original data frequencies.

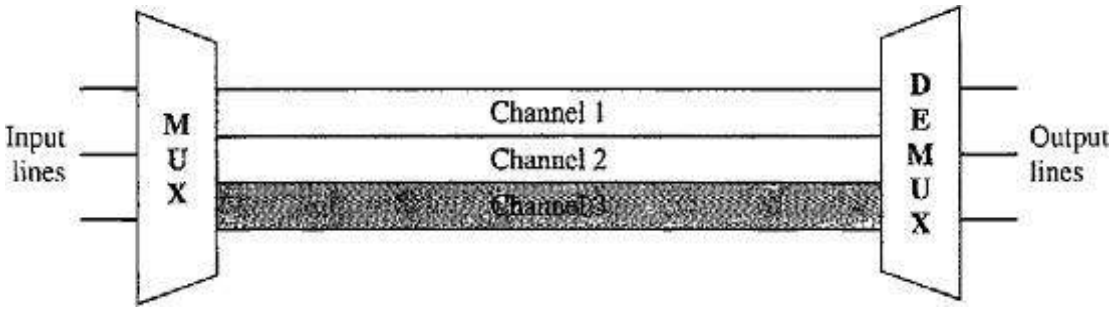
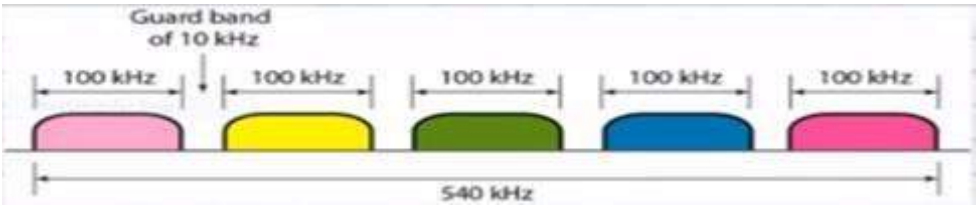


Fig: Frequency Division Multiplexing



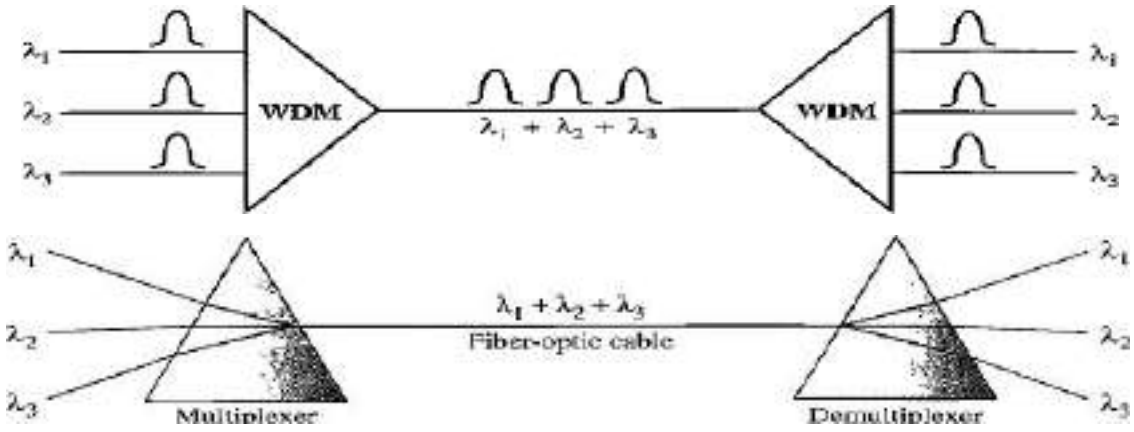
In the above figure, the transmission path is divided into three parts, each representing a channel that carries one transmission.

Multiplexing Process

- Each source generates a signal of a similar frequency range.
- Inside the multiplexer, these similar signals modulate different carrier frequencies (f1, f2, f3).
- The resulting modulated signals are then combined into a single composite signal that is sent out over a media link that has enough bandwidth to accommodate it.

Wavelength-Division Multiplexing (WDM)

- WDM is an analog multiplexing technique to combine optical signals. WDM is designed to use the high-data-rate capability of fiber-optic cable.
- The optical fiber data rate **higher than** the data rate of metallic transmission cable
- Using a fiber-optic cable for one single line wastes the available bandwidth. Multiplexing allows us to combine several lines into one.



- Very narrow bands of light from different sources are combined to make a wider band of light. At the receiver, the signals are separated by the

demultiplexer.

- The combining and splitting of light sources are easily handled by a **Prism**. A Prism bends a beam of light based on the angle of incidence and the frequency.
- A multiplexer can be made to combine several input beams of light, each containing a narrow band of frequencies, into one output beam of a wider band of frequencies.
- A demultiplexer can be made to divide wider band of frequencies by decomposing the light beams into narrow band frequencies.

Advantages: High Speed and High frequency, uses narrow bands of light sources.

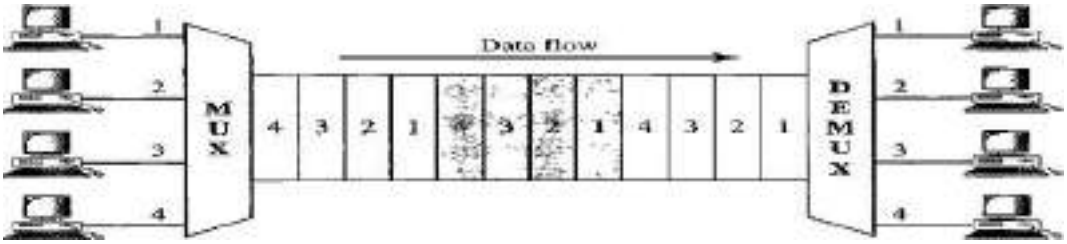
Disadvantages: Expensive than FDM.

Time-Division Multiplexing (TDM)

TDM is a digital multiplexing technique for combining several low-rate channels into one high-rate channel. Digital data from different sources are combined into one timeshared link

(i.e.) The **data rate** capacity of transmission medium >= The **data rate** required by sending and receiving devices.

TDM is a digital process that allows several connections to share the high bandwidth of a link. Each connection occupies a portion of time in the link.



In the above figure all the data in a message from source 1 always go to one specific destination either of 1, 2, 3, or 4. The delivery is fixed and unvarying.

Data Link Layer

The responsibility of the **Physical Layer** is to transmit the unstructured raw bit stream over a physical medium.

The responsibility of **Data-Link Layer** is to transforming raw transmission facility into a **Link** responsible for node-to-node communication (hop-to-hop communication).

Responsibilities of the Data Link Layer include:

1. Framing
2. Physical Addressing
3. Flow control
4. Error control
5. Media Access Control.

Framing

The data link layer divides the stream of bits received from the network layer into manageable data units called frames. In simple terms data link layer is responsible for moving frames from one node to another node.

Physical Addressing

The data link layer adds a header to the frame to define the addresses of the sender and receiver of the frame.

Flow Control

If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.

Error Control

The data link layer also adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged, duplicate, or lost frames.

Media Access Control

When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time.

Framing

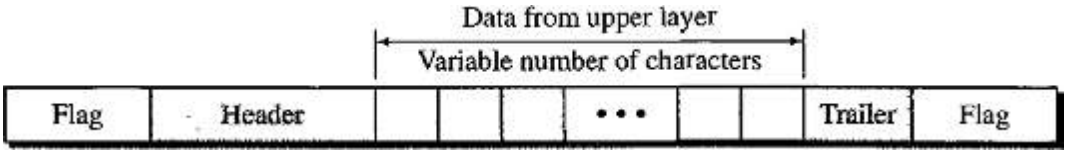
- Framing in the data link layer is breaking up the bit stream into frames.
- Framing can be done in 2ways:
- **Fixed size framing:** The size of the frame is fixed for all the frames. There is no need to define the boundaries of a frame.
- **Variable size framing:** In variable-size framing, we need a way to define the end of the frame and the beginning of the next frame.

There are 2 approaches are used for variable size framing:

1. Character Stuffing(A Character-Oriented Approach)
2. Bit Stuffing(A Bit-Oriented Approach)

Character Stuffing/Byte Stuffing

In a character stuffing, data to be carried are 8-bit characters from a coding system such as ASCII. The Frame format in Character Stuffing is given below:



Character Stuffing uses: Header, Trailer and a Flag.

- **Header** carries the source and destination addresses and other control information.
- **Trailer** carries error detection or error correction redundant bits, are also multiples of 8 bits.
- To separate one frame from the next, an 8-bit (I-byte) flag is added at the beginning and the end of a frame. The flag signals receiver either start or end of a frame.

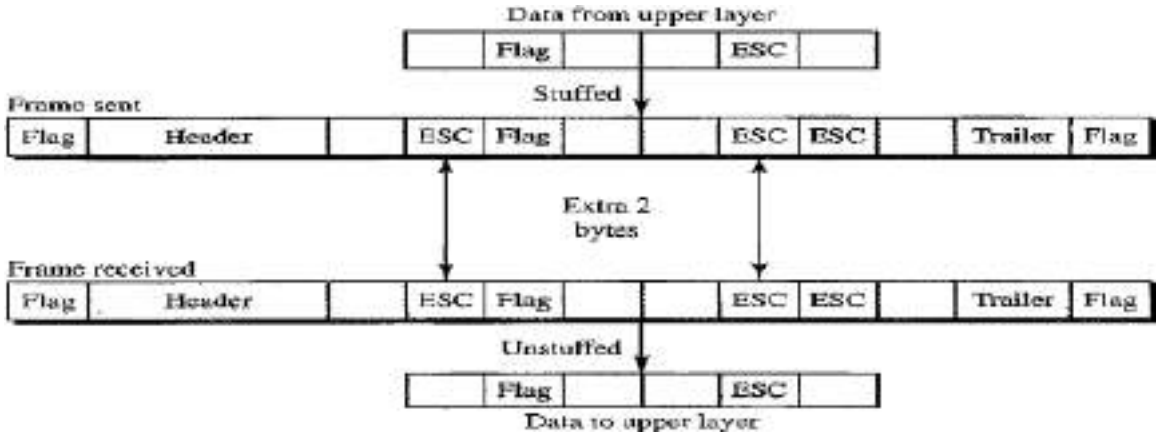
Disadvantages of Character Stuffing

- Character oriented framing is useful for text transfer not useful for audio video etc.
- Any pattern used for the flag could also be part of the information.
- If this happens, the receiver, when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame and the treats then exit bit as new frame.

To fix this problem a **Byte Stuffing** strategy is introduced.

- In byte stuffing a special byte is added to the data section of the frame when there is a character with the same pattern as the flag.
- The data section is stuffed with an extra byte called Escape character (ESC).
- Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not a delimiting flag.

Figure shows the byte stuffing and Unstuffing:



Problems with Byte Stuffing

- If the text contains one or more escape characters followed by a flag, the receiver removes the escape character, but keeps the flag, which is incorrectly interpreted as the end of the

frame.

Solution

- To solve this problem, the escape characters that are part of the text must also be marked by another escape character.

Disadvantages of character/Byte stuffing Procedure

- The universal coding systems (Unicode) in use today have16-bit and 32-bit characters that conflict with 8-bit characters.
- Character stuffing deals with 8-bit characters but today's systems using 16 bits, 32 bits and 64 bit characters hence there will be conflict.

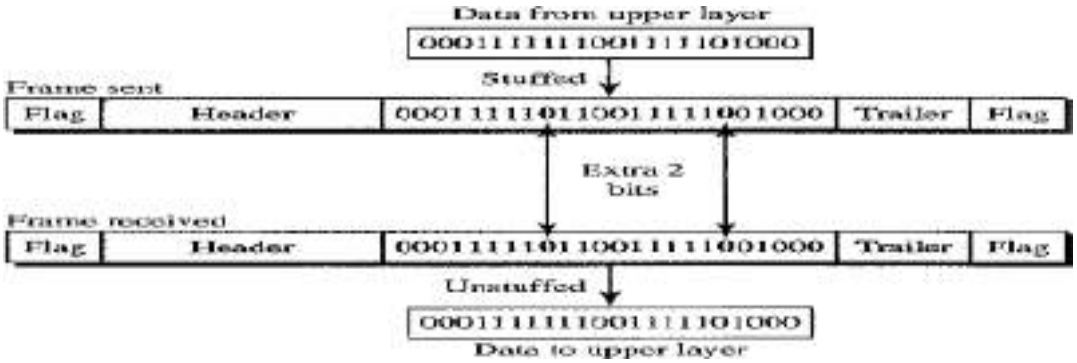
The solution for this problem is using ***Bit Oriented Approach.***

Bit stuffing

- In a bit-oriented protocol, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on.
- In addition to headers (and possible trailers), we still need a delimiter to separate one frame from the other.
- Most protocols use a special 8-bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame is give below figure:

- In bit stuffing, if a 0 and five consecutive1-bits are encountered, an extra0 is added.
- This extra stuffed bit is eventually removed from the data by the receiver.

Note: the extra bit is added after one 0 followed by five 1s regardless of the value of the next bit. (i.e.) when 0**11111**00 is a part of the data, then also we have to add "0" after five 1"s .Hence the data will be 0**11111**000



Advantages of Bit Stuffing

If the flag like pattern 01111110 appears in the data, it will change to 011111010 (stuffed) and is not mistaken as a flag by the receiver. The real flag 01111110 is not stuffed by the sender and is recognized by the receiver.

Error is a condition when the receiver's information does not match the sender's information. During transmission, digital signals suffer from noise that can introduce errors in the binary bits traveling from sender to receiver. That means a 0 bit may change to 1 or a 1 bit may change to 0.

Error Detection:
To prevent such errors, error-detection codes are added as extra data to digital messages. This helps in detecting any errors that may have occurred during message transmission.

1. **Checksum**

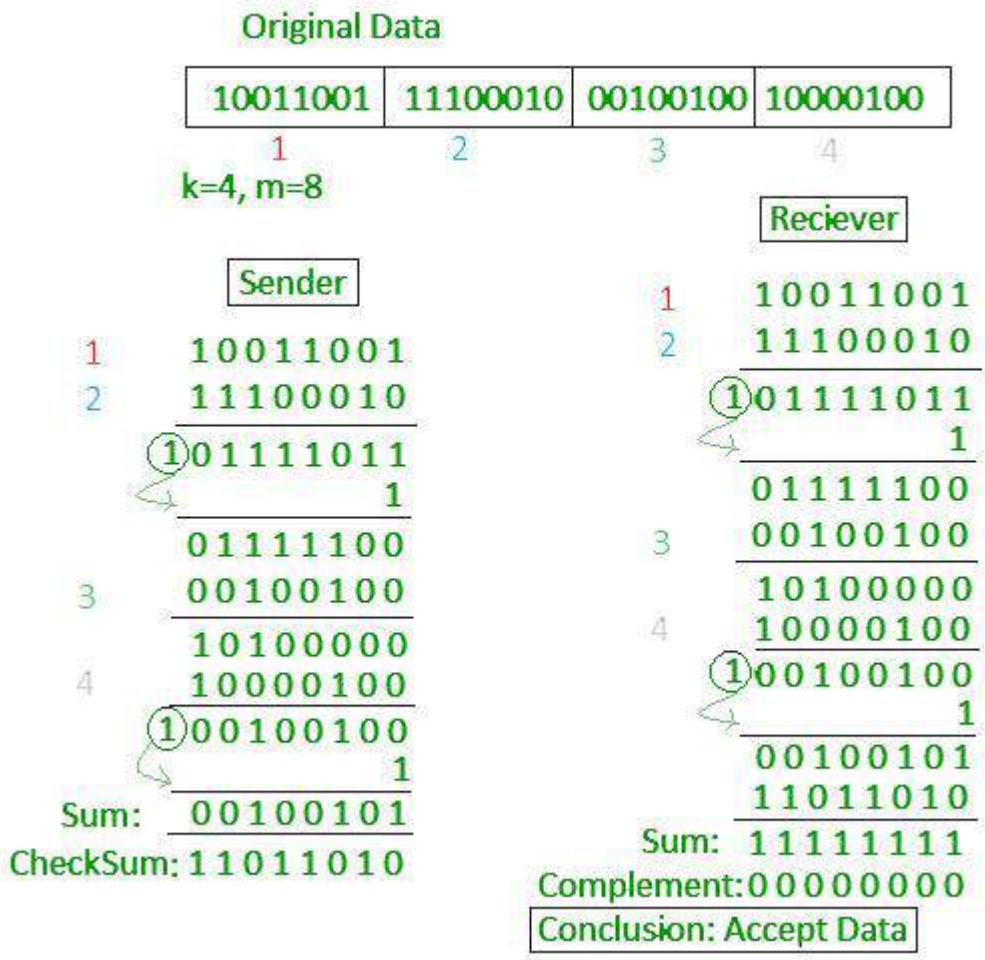
Checksum error detection is a method used to identify errors in transmitted data. The process involves dividing the data into equally sized segments and using a 1's complement to calculate the sum of these segments. The calculated sum is then sent along with the data to the receiver. At the receiver's end, the same process is repeated and if all zeroes are obtained in the sum, it means that the data is correct.

Checksum – Operation at Sender's Side

- Firstly, the data is divided into k segments each of m bits.
- On the sender's end, the segments are added using 1's complement arithmetic to get the sum. The sum is complemented to get the checksum.
- The checksum segment is sent along with the data segments.

Checksum – Operation at Receiver's Side

- At the receiver's end, all received segments are added using 1's complement arithmetic to get the sum. The sum is complemented.
- If the result is zero, the received data is accepted; otherwise discarded.

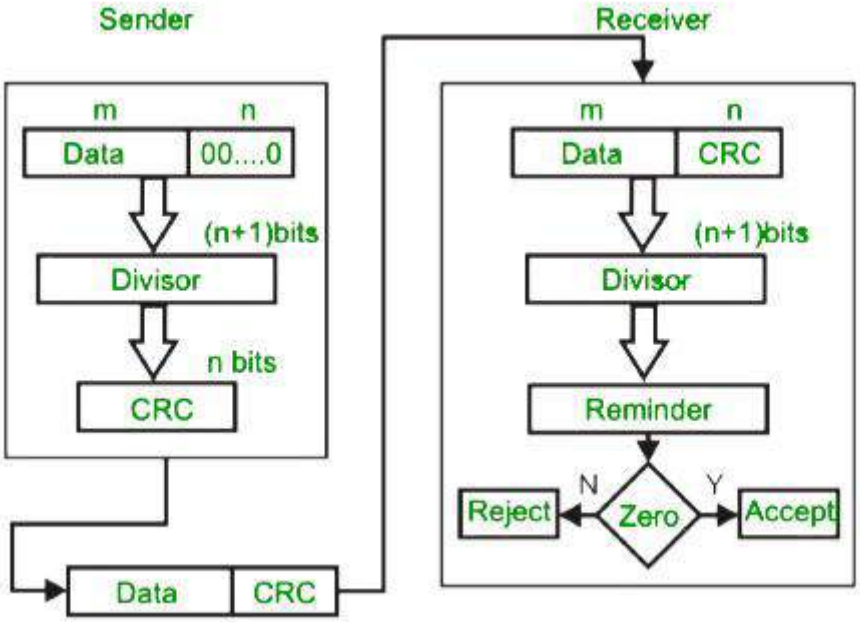


Disadvantages

- If one or more bits of a segment are damaged and the corresponding bit or bits of opposite value in a second segment are also damaged.

2. Cyclic Redundancy Check (CRC)

- Unlike the checksum scheme, which is based on addition, CRC is based on binary division.
- In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended to the end of the data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number.
- At the destination, the incoming data unit is divided by the same number. If at this step there is no remainder, the data unit is assumed to be correct and is therefore accepted.
- A remainder indicates that the data unit has been damaged in transit and therefore must be rejected.



original message
1010000

@ means X-OR

Generator polynomial
x^3+1
$1\,x^3+0\,x^2+0\,x^1+1\,x^0$
CRC generator
1001 4-bit

If CRC generator is of n bit then append (n-1) zeros in the end of original message

Sender

```
1001 | 1010000000
     @1001
     ----
     0011000000
      @1001
      ----
      01010000
       @1001
       ----
       0011000
        @1001
        ----
        01010
         @1001
         ----
         0011
```

Message to be transmitted

```
1010000000
    +011
----------
1010000011
```

```
1001 | 1010000011
     @1001
     ----
     0011000011
      @1001
      ----
      01010011      <- Receiver
       @1001
       ----
       0011011
        @1001
        ----
        01001
         @1001
         ----
         0000
```

Zero means data is accepted

Error Correction:

Hamming code is a set of error-correction codes that can be used to **detect and correct the errors** that can occur when the data is moved or stored from the sender to the receiver.

Redundant bits – Redundant bits are extra binary bits that are generated and added to the information-carrying bits of data transfer to ensure that no bits were lost during the data transfer. The number of redundant bits can be calculated using the following formula:

```
2^r ≥ m + r + 1

 where, r = redundant bit, m = data bit
```

m=7

2^4 ≥ 7 + 4 + 1 Thus, the number of redundant bits= 4 **Parity bits.**

There are two types of parity bits:

1. **Even parity bit:** In the case of even parity, for a given set of bits, the number of 1's are counted. If that count is odd, the parity bit value is set to 1, making the total count of occurrences of 1's an even number. If the total number of 1's in a given set of bits is already even, the parity bit's value is 0.
2. **Odd Parity bit –** In the case of odd parity, for a given set of bits, the number of 1's are counted. If that count is even, the parity bit value is set to 1, making the total count of occurrences of 1's an odd number. If the total number of 1's in a given set of bits is already odd, the parity bit's value is 0.

General Algorithm of Hamming code: Hamming Code is simply the use of extra parity bits to allow the identification of an error.

1. Write the bit positions starting from 1 in binary form (1, 10, 11, 100, etc).
2. All the bit positions that are a power of 2 are marked as parity bits (1, 2, 4, 8, etc).
3. All the other bit positions are marked as data bits.
4. Each data bit is included in a unique set of parity bits, as determined its bit position in binary form. **a.** Parity bit 1 covers all the bits positions whose binary representation includes a 1 in the least significant position (1, 3, 5, 7, 9, 11, etc). **b.** Parity bit 2 covers all the bits positions whose binary representation includes a 1 in the second position from the least significant bit (2, 3, 6, 7, 10, 11, etc). **c.** Parity bit 4 covers all the bits positions whose binary representation includes a 1 in the third position from the least significant bit (4–7, 12–15, 20–23, etc). **d.** Parity bit 8 covers all the bits positions whose binary representation includes a 1 in the fourth position from the least significant bit bits (8–15, 24–31, 40–47,

etc). **e.** In general, each parity bit covers all bits where the bitwise AND of the parity position and the bit position is non-zero.

5. Since we check for even parity set a parity bit to 1 if the total number of ones in the positions it checks is odd.
6. Set a parity bit to 0 if the total number of ones in the positions it checks is even.

Position	R8	R4	R2	R1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1

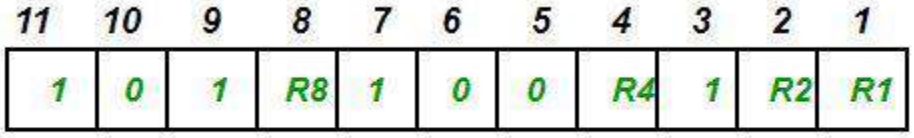
R1 -> 1,3,5,7,9,11
R2 -> 2,3,6,7,10,11
R3 -> 4,5,6,7
R4 -> 8,9,10,11

Determining the position of redundant bits – These redundancy bits are placed at positions that correspond to the power of 2.
As in the above example:

- The number of data bits = 7

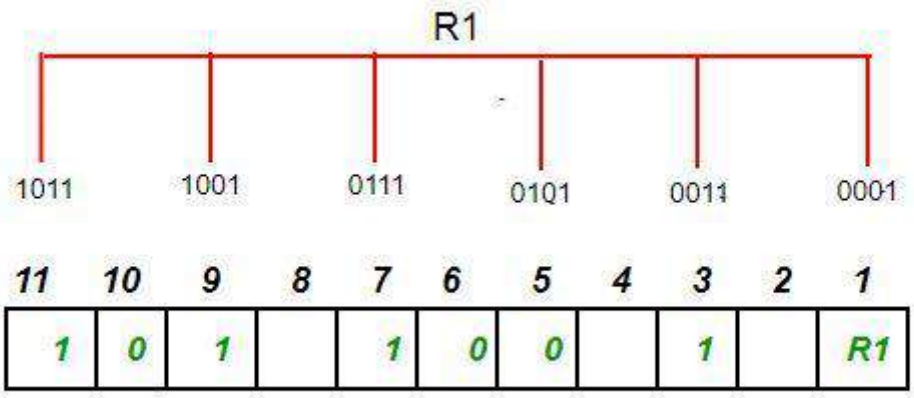
- The number of redundant bits = 4
- The total number of bits = 11
- The redundant bits are placed at positions corresponding to power of 2- 1, 2, 4, and 8

- Suppose the data to be transmitted is 1011001, the bits will be placed as follows:

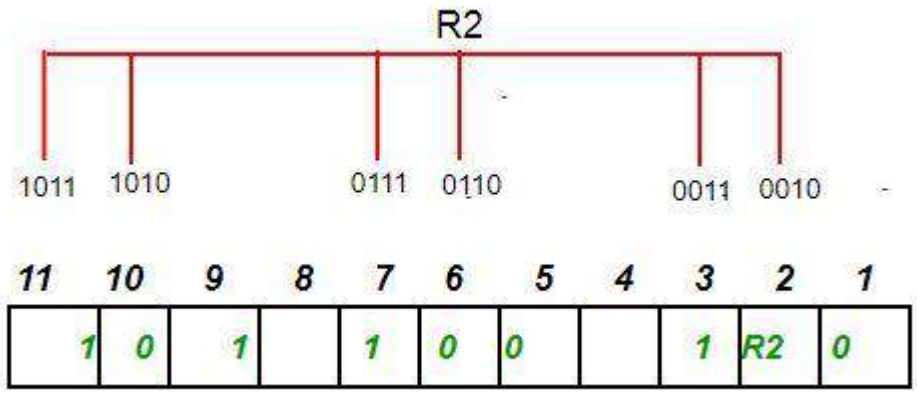


Determining the Parity bits:

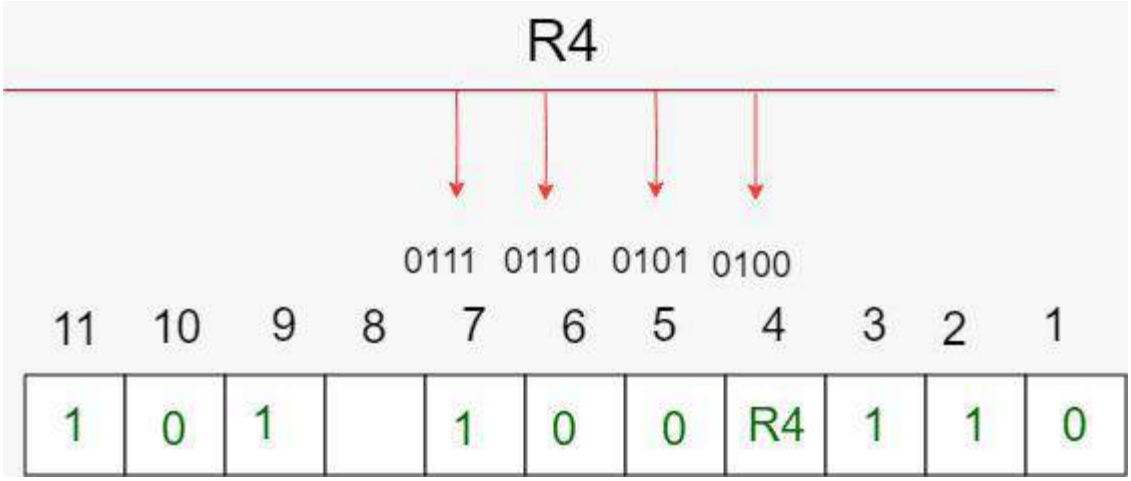
- R1 bit is calculated using parity check at all the bits positions whose binary representation includes a 1 in the least significant position. R1: bits 1, 3, 5, 7, 9, 11



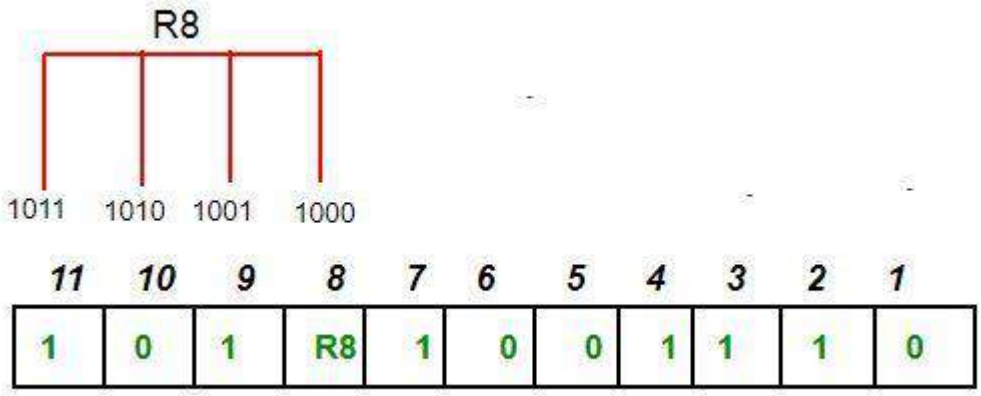
- To find the redundant bit R1, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R1 is an even number the value of R1 (parity bit's value) = 0
- R2 bit is calculated using parity check at all the bits positions whose binary representation includes a 1 in the second position from the least significant bit. R2: bits 2,3,6,7,10,11



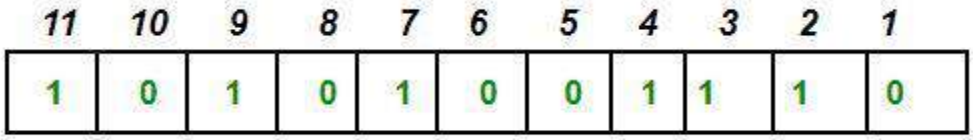
- To find the redundant bit R2, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R2 is odd the value of R2(parity bit's value)=1
- R4 bit is calculated using parity check at all the bits positions whose binary representation includes a 1 in the third position from the least significant bit. R4: bits 4, 5, 6, 7



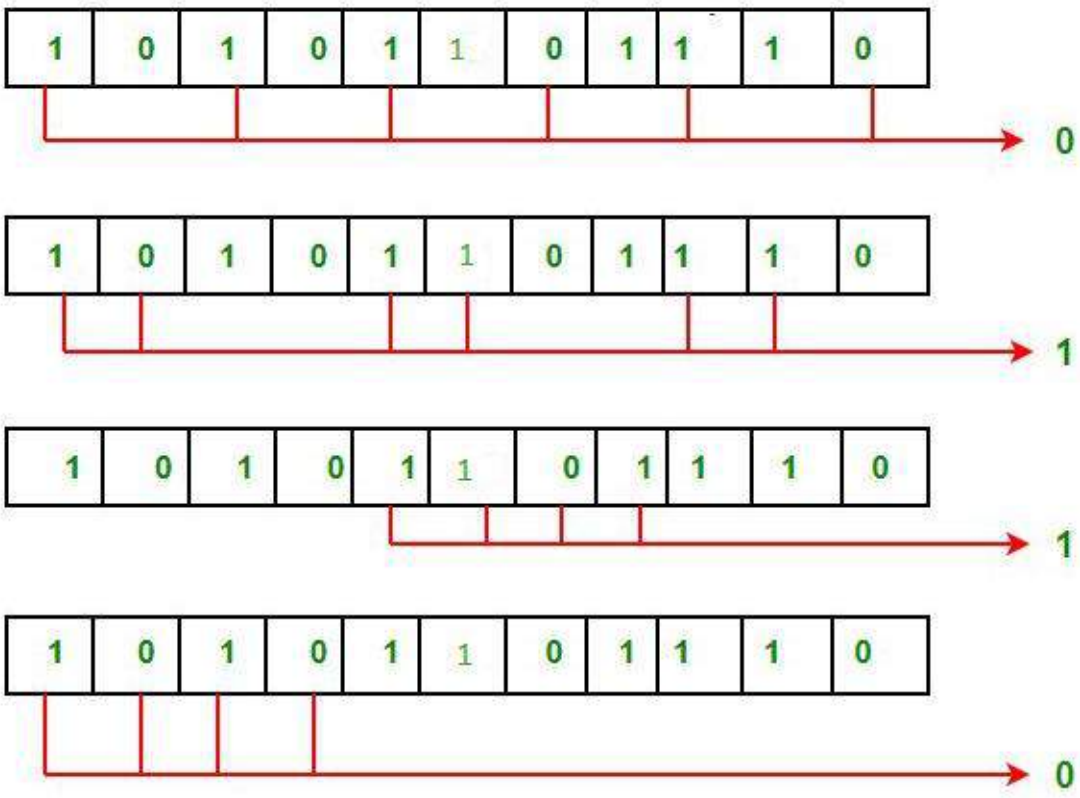
1. To find the redundant bit R4, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R4 is odd the value of R4(parity bit's value) = 1
2. R8 bit is calculated using parity check at all the bits positions whose binary representation includes a 1 in the fourth position from the least significant bit. R8: bit
   8,9,10,11



- To find the redundant bit R8, we check for even parity. Since the total number of 1's in all the bit positions corresponding to R8 is an even number the value of R8(parity bit's value)=0. Thus, the data transferred is:



Error detection and correction: Suppose in the above example the 6th bit is changed from 0 to 1 during data transmission, then it gives new parity values in the binary number:



For all the parity bits we will check the number of 1's in their respective bit positions.

For R1: bits 1, 3, 5, 7, 9, 11. We can see that the number of 1's in these bit positions are 4 and that's even so we get a 0 for this.

For R2: bits 2,3,6,7,10,11 . We can see that the number of 1's in these bit positions are 5 and that's odd so we get a 1 for this.

For R4: bits 4, 5, 6, 7 . We can see that the number of 1's in these bit positions are 3 and that's odd so we get a 1 for this.

For R8: bit 8,9,10,11 . We can see that the number of 1's in these bit positions are 2 and that's even so we get a 0 for this.

The bits give the binary number 0110 whose decimal representation is 6. Thus, bit 6 contains an error. To correct the error the 6th bit is changed from 1 to 0.