DATA LINK LAYER & PHYSICAL LAYER

In data communication, physical layer deals with transmission of signals over different transmission medium. While sending data, the signals may get impaired due to the noise encountered during transmission. The data flow rate between the source and destination also should be kept under control. Therefore in order to achieve an efficient and reliable communication a data flow control mechanism needs to be implemented. Data link layer deals with frame formation, flow control, error control and addressing and ensures error free transfer of bits from one device to another. For the effective data communication data link layer needs to perform a number of specified functions**.**

Link layer services

- *flow control:*
  - pacing between adjacent sending and receiving nodes
- *error detection*:
  - errors caused by signal attenuation, noise.
  - receiver detects presence of errors:
    - signals sender for retransmission or drops frame
- *error correction:*
  - receiver identifies *and corrects* bit error(s) without resorting to retransmission
- *half-duplex and full-duplex*
  - with half duplex, nodes at both ends of link can transmit, but not at same time

Services provided to network layer: The main functionality of this layer is to transfer data from the network layer on source machine to the network layer on destination machine

Flow Control: The source machine should sent data at a rate faster than the destination machine can accept them

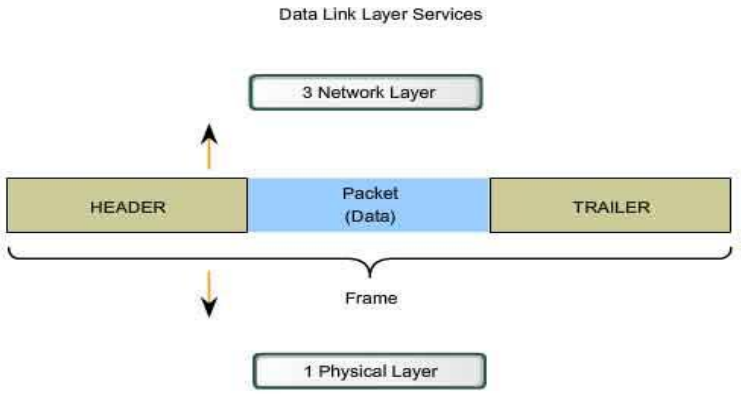
Framing: The bits to be transmitted are broken down into discrete frames. A frame contains user data and control fields.

Error Control: All the frames should be delivered from source to the destination. The errors made in bits during transmission must be detected and corrected

Addressing: On a multipoint line, such as many machines connected together, identity of individual machines must be specified while transmitting data frames

5.1FRAME

Frame is a data structure used in transmissions at DLL. The data link layer takes the packets it gets from the network layer and encapsulates them into frames for transmission. Each frame contains a frame header with fields for addressing and is located at the beginning of the frame, a payload field for holding the packet and a frame trailer. The trailer contains fields are used for error detection and mark the end of the frame.



(Fig 1.15) Frame structure

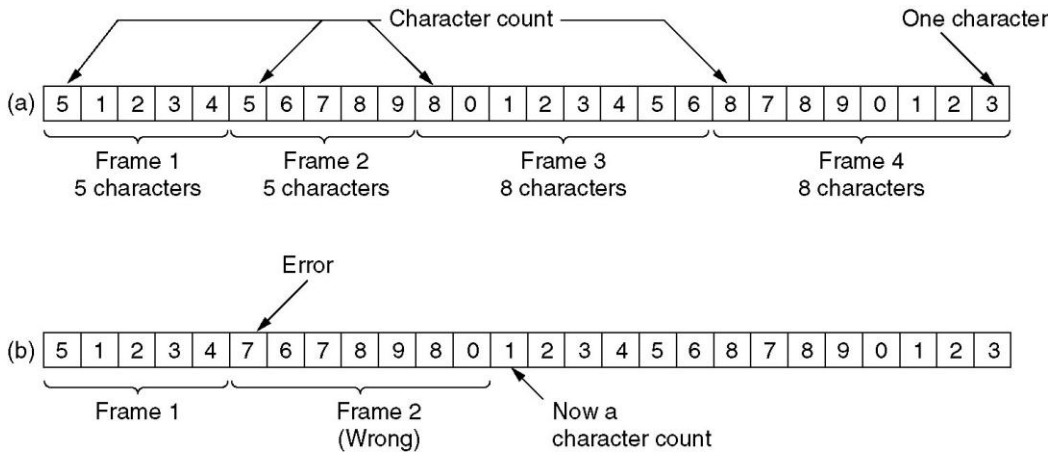
5.1.1 FRAME SYNCHRONIZATION

Frame synchronization or simply framing is the process of defining and locating frame boundaries (start and end of the frame) on a bit sequence. Converting the bit stream into frames is a tedious process. The frame format is designed in a way that enables the receiver to always locate the beginning of a frame and its various fields and should be able to separate the data field. To identify a frame and its various fields, field identifiers are incorporated. These are unique symbols that indicate by their presence the beginning and end of a frame. Four methods can be used to mark the start and end of each frame:

- **Character count**
- **Flag bytes with byte stuffing**
- **Starting and ending flags, with bit stuffing**
- **Physical layer coding violations**

Character Count

Character count, uses a header field to specify the number of characters in the frame. The Data Link Layer at the destination checks the header field to know the size of the frame and hence, the end of frame. The process is shown in the following figure for a four frame of size 5, 5,8 and 8 respectively.

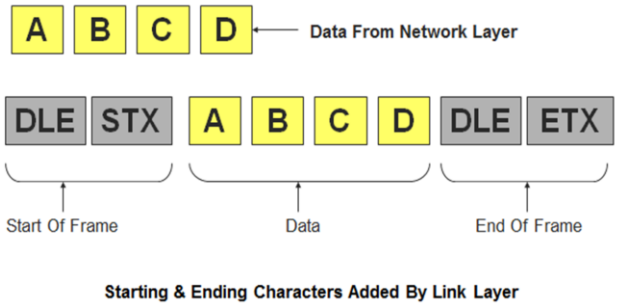


(Fig: 1.16) A byte stream. (a) Without errors. (b) With one error.

However, problems may arise due to changes in character count value during transmission. For example, in the second frame if the character count 5 changes to7, the destination will receive data out of synchronization and hence, it will not be able to identify the start of the next frame.

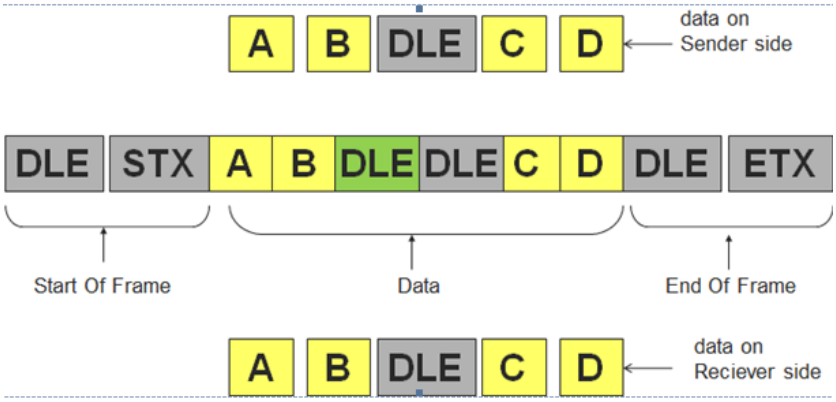
Flag Bytes With Byte Stuffing

Byte Stuffing also known as Character Stuffing is one of the earliest schemes adopted for delimiting packets containing character data. This method employees three special control characters in ASCII for the purpose of framing: **DLE** -Data Link Escape, **STX** - Start of Text and **ETX** -End of Text. The pattern DLE STX denotes the beginning of each frame and DLE ETX specifies the end of each frame.



(Fig 1.17) Byte Stuffing

However, there is a still a problem we have to solve. It may happen that the flag byte occurs in the data. For example, if a DLE occurs in the middle of the data and interferes with the data during framing then, sender stuffs an extra DLE into the data stream just before each occurrence of an "accidental" DLE in the data stream. The data link layer on the receiving end discards the first DLE and the second DLE is regarded as data.

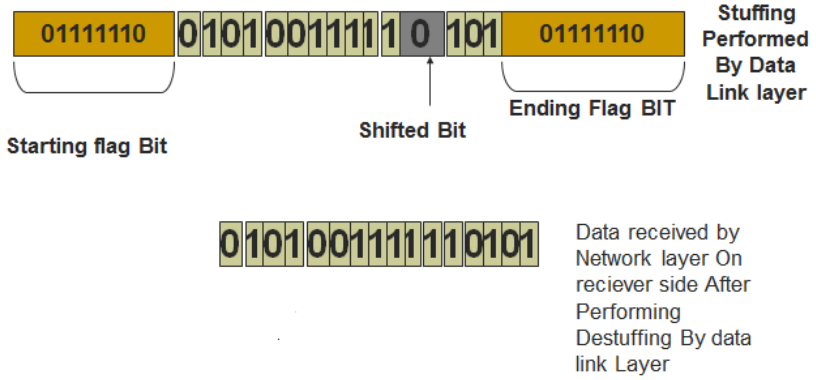


(Fig:1.18)Byte Stuffing: Flag Byte within Data

Bit Stuffing

Bit Stuffing is similar to the Byte Stuffing, except that, the method of bit stuffing allows insertion of bits instead of the entire character (8 bits). Here frames can contain an arbitrary number of bits made up of units of any size. Each frame begins and ends with a special bit pattern, 01111110 or 0x7E in hexadecimal. Whenever the sender's data link layer encounters five consecutive 1's in the data, it automatically stuffs a 0 bit into the outgoing bit stream. This bit stuffing is analogous to byte stuffing.

When the receiver sees five consecutive incoming 1 bits, followed by a 0 bit, it automatically removes the 0 bit. Just as byte stuffing is completely transparent to the network layer in both computers, so is bit stuffing. With bit stuffing, the boundary between two frames can be unambiguously recognized by the flag pattern. Thus, if the receiver loses track of where it is, all it has to do is scan the input for flag sequences, since they can only occur at frame boundaries and never within the data.



(Fig 1.19) Bit Stuffing

With both bit and byte stuffing, a side effect is that the length of a frame now depends on the contents of the data it carries. For instance, if there are no flag bytes in the data, 100 bytes might be carried in a frame of roughly 100 bytes. If, however, the data consists solely of flag bytes, each flag byte will be escaped and the frame will become roughly 200 bytes long. With bit stuffing, the increase would be roughly 12.5% as 1 bit is added to every byte.

5.2 FLOW CONTROL

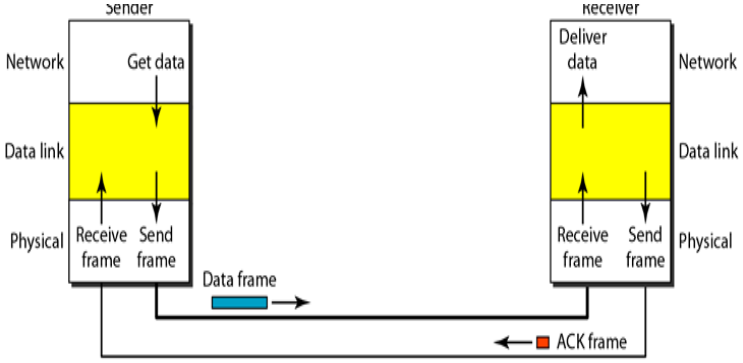
Another important issue for the data link layer is dealing with the situation which occurs when the sender transmits frames faster than the receiver can accept or process them. This situation can easily occur when the sender is running on a fast computer and the receiver is running on a slow machine. The sender keeps pumping the frames out at a high rate until the receiver is completely swamped. Even if the transmission is error free, at a certain point the receiver will simply be unable to handle the frames as they arrive and will start to lose some. To prevent this situation during transmission, an approach is introduced called the Flow Control.

Flow Control is a set of procedures that tells the sender how much data it can transmit before it must wait for an acknowledgment from the receiver. The flow of data should not be allowed to overwhelm the receiver. Receiver should also be able to inform the transmitter before its limits (this limit may be amount of memory used to store the incoming data or the processing power at the receiver end) are reached and the sender must send fewer frames. Hence, Flow control refers to the set of procedures used to restrict the amount of data the transmitter can send before waiting for acknowledgment.

There are two methods developed for flow control namely **Stop-and-wait** and **Sliding-window**. Stop-and-wait is also known as Request/reply sometimes. Request/reply (Stop-and-wait) flow control requires each data packet to be acknowledged by the remote host before the next packet is sent.. Sliding window permits multiple data packets to be in simultaneous transit, making more efficient use of network bandwidth

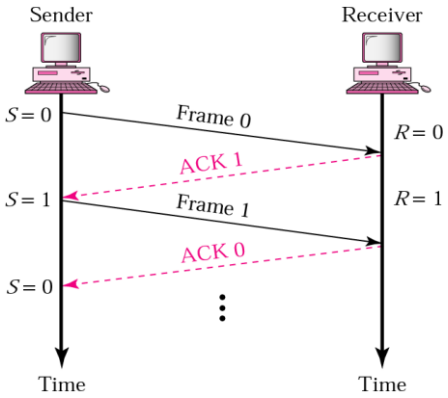
5.2.1 STOP AND WAIT

Stop-and-Wait Flow Control is the simplest form of flow control. The message is broken into multiple frames and only a single frame is send at a time. The Sender waits for an ACK (acknowledgement) after every frame for a specified time (called time out). It is sent to ensure that the receiver has received the frame correctly. It will then send the next frame only after the ACK has been received. Sender keeps a copy of the last frame until it receives an acknowledgement.



(Fig 1.20) Stop & Wait

For identification, both data frames and acknowledgements (ACK) frames are numbered alternatively 0 and 1. Sender has a control variable (S) that holds the number of the recently sent frame (0 or 1). Receiver has a control variable R that holds the number of the next frame expected (0 or 1)

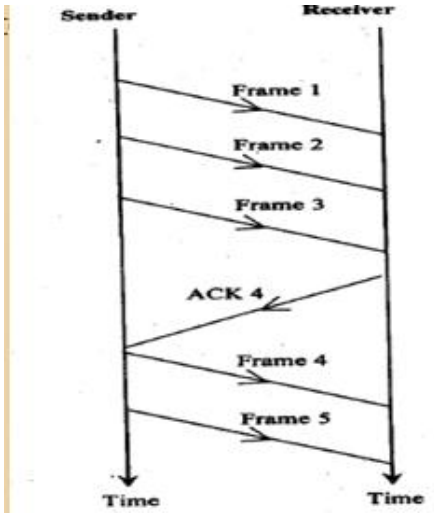


(Fig:1.21) Stop & Wait Operation

The problem with stop and wait is that at any point in time, there is only one frame that is sent and waiting to be acknowledged. Each frame must travel all the way to the receiver and an acknowledgment must travel all the way back before the next frame can be sent. Till we get the acknowledgement the sender cannot transmit any new packet. During this time both the sender and the channel are unutilized.

5.2.2 SLIDING WINDOW

With the use of multiple frames for a single message, the stop-and-wait protocol does not perform well. Only one frame at a time can be in transit .Sliding Window approach allows the sender to transmit multiple frames without an ACK. Each data frame carries a sequence number for its identification. Sequence number is a field in the frame that is of finite size. If k bits are reserved for the sequence number, then the values of sequence number ranges from 0 to 2^k–1.The receiver acknowledges the receipt of one or more data frames by sending back a numbered acknowledgment which specifies the sequence number of the next expected frame. All the previous data frames are assumed acknowledged on receipt of an acknowledgement. The sender sends the next n frames starting with the last received sequence number that has been transmitted by the receiver (ACK).

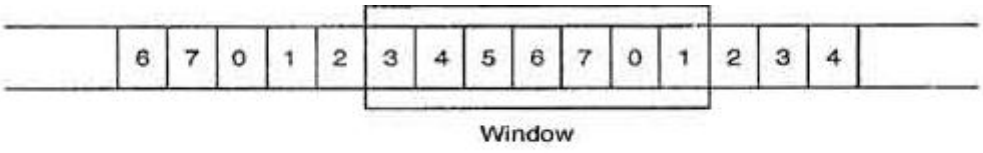


(Fig:1.22) Normal Flow diagram of a sliding window

The receiver receives frames 1,2 and 3. Once frame 3 arrives ACK4 is sent to the sender. This ACK4 acknowledge the receipt of frame 1, 2 and 3 and informs the sender that the next expected frame is frame 4. Therefore, the sender can send multiple back-to-back frames, making efficient use of the channel.

Operation Of A Sliding Window

The idea of sliding windows is to keep track of the acknowledgements. In this mechanism we maintain two types of windows (buffer) sending window and receiving window. The sender needs buffer because it needs to keep copies of all the sent frames for which acknowledgments are yet to be received. The receiver may request for the retransmission of a data frame that is received with errors. The receiver needs buffer to store the received data frames temporarily. The frames may be received out of sequence and it must put them in sequence before processing them for retrieval of user data. The size of the s window is at most $2^k - 1$.



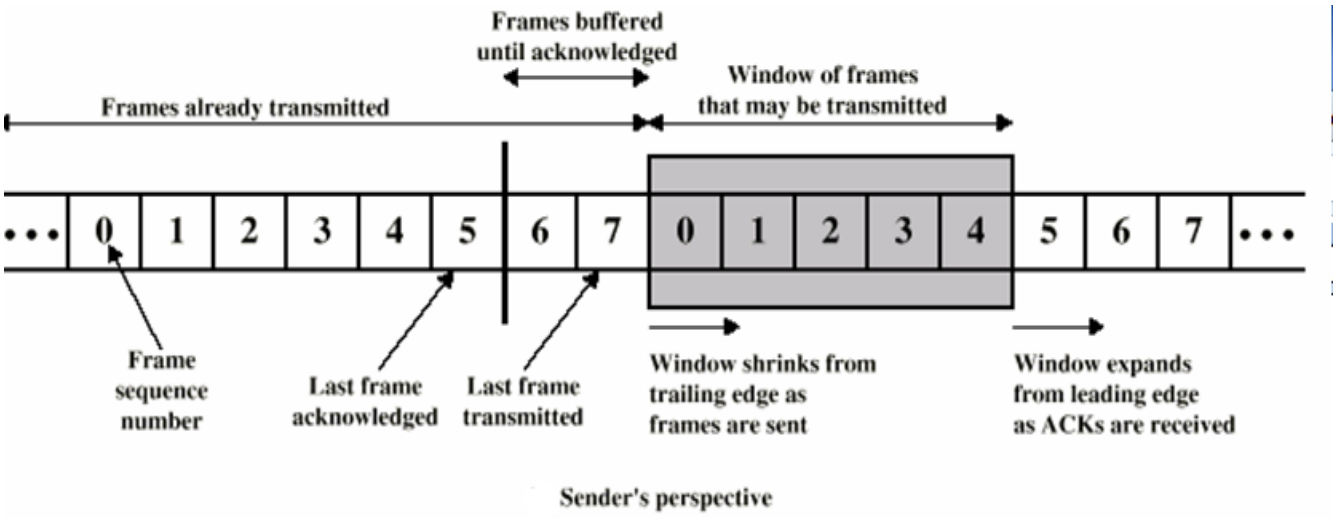
Sliding Window

(Fig 1.23)

Sending Window

The sending window contains the copies of those data frames that have been transmitted but their acknowledgments are yet to be received, and the data frames which are next to be transmitted. At the beginning of a transmission, the sender's window contains **n-l** frames.

As the frames are sent by source, the left boundary of the window moves inward, shrinking the size of window. This means if window size is **w**, if four frames are sent by source after the last acknowledgment, then the number of frames left in window is **w-4**. When the receiver sends an ACK, the source's window expand i.e. (right boundary moves outward) to allow in a number of new frames equal to the number of frames acknowledged by that ACK.

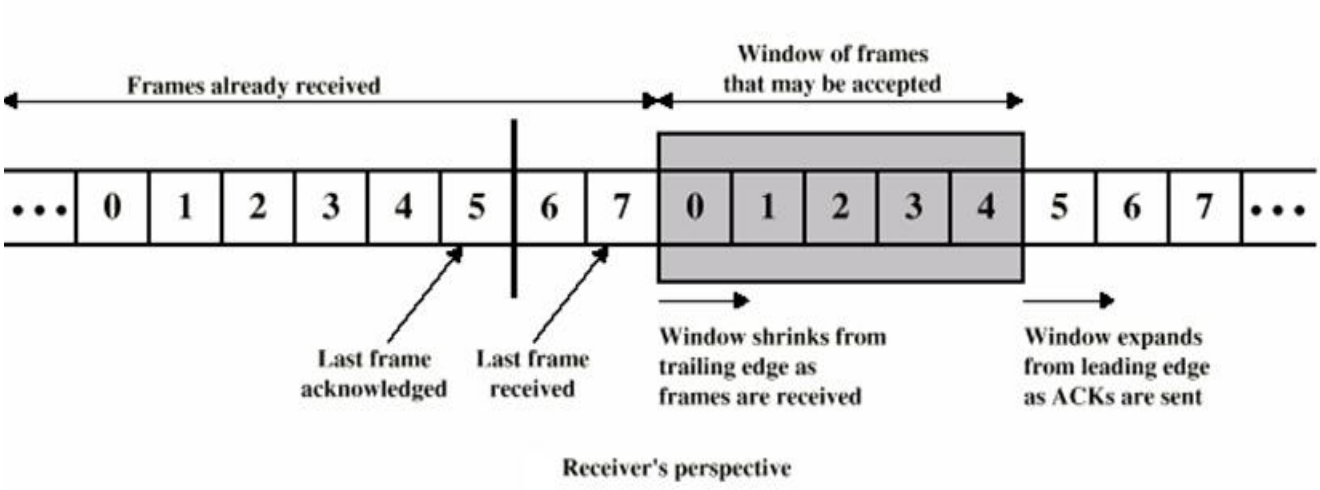


(Fig 1.24) Sliding Window Depiction: Sender's Perspective

For example, Let the window size is 7.If frames 0 through 3 have been sent and no acknowledgment has been received, then the sender's window contains three frames -4, 5, 6. Now, if an ACK numbered 3 is received by source, it means three frames (0, 1, 2) have been received by receiver and are undamaged. The sender's window will now expand to include the next three frames .At this point the sender's window will contain six frames (4, 5, 6, 7, 0, 1).

Receiving Window

At the receiving end, the window contains the sequence numbers of the data frames the receiver is ready to accept. As the new frames come in, the size of window shrinks. Therefore the receiver window represents not the number of frames received but the number of frames that may still be received without an acknowledgment ACK must be sent. Given a window of size w, if three frames are received without an ACK being returned, the number of spaces in a window is w-3.As soon as acknowledgment is sent, window expands to include the number of frames equal to the number of frames acknowledged.



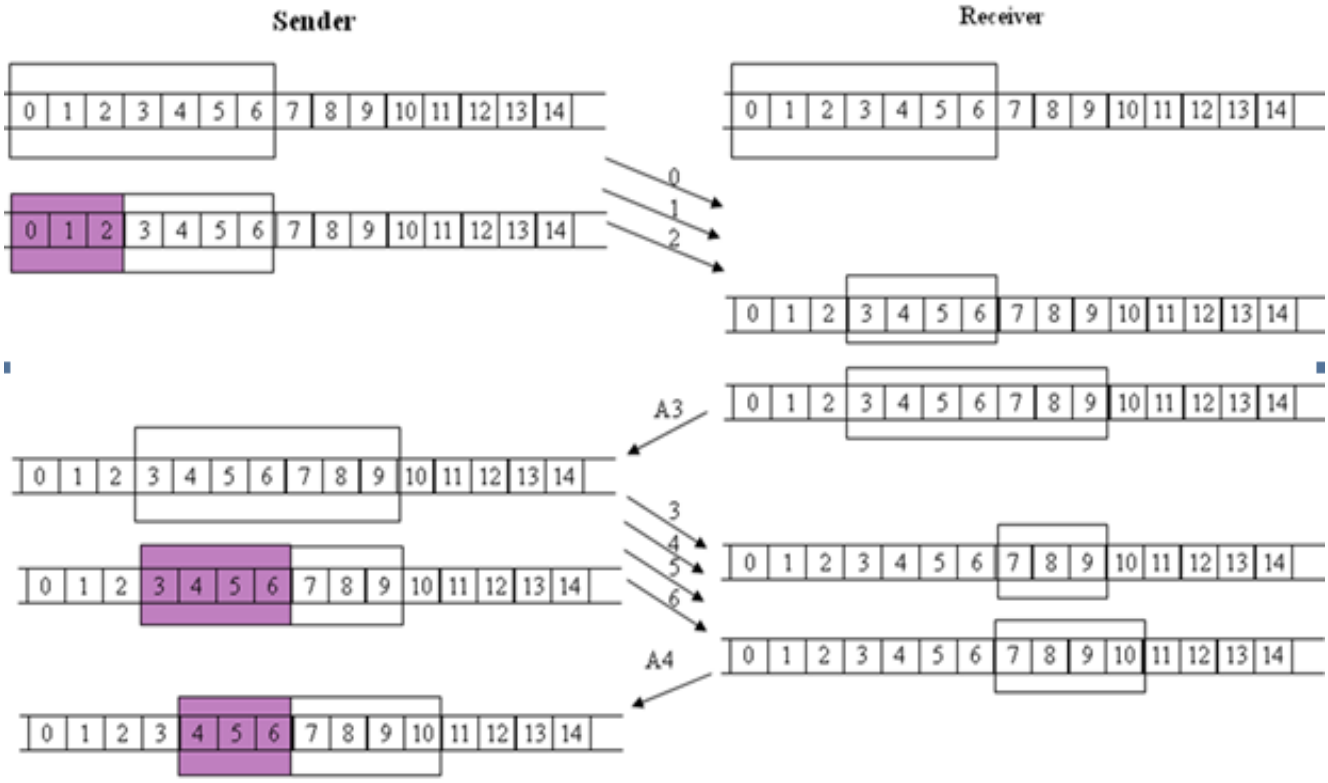
(Fig 1.24) Sliding Window Depiction: Reciever's Perspective

For example, let the size of receiver's window is 7. It means window contains spaces for 7 frames. With the arrival of the first frame, the receiving window shrinks, moving the boundary from space 0 to 1. Now, window has shrunk by one, so the receiver may accept six more frame before it is required to send an ACK. If frames 0 through 3 have arrived but have not been acknowledged, the window will contain three frame spaces. As receiver sends an ACK, the window of the receiver expands to include as many new placeholders as newly acknowledged frames. The window expands to include a number of new frame spaces equal to the number of the most recently acknowledged frame minus the number of previously acknowledged frame. For e.g., If window size is 7 and if prior ACK was for frame 2 & the current ACK is for frame 5 the window expands by three (5-2).

Therefore, the sliding window of sender shrinks from left when frames of data are sending and expands to right when acknowledgments are received. The sliding window of the receiver shrinks from left when frames of data are received and expands to the right when acknowledgement is sent.

In the following example, initially, both the sender and receiver windows have a size of 7.Sender transmits 3 frames numbered 0 through 2. The sliding window of sender shrinks 3 positions from left since 3 frames are transmitted. At the receiver side when these 3 frames are received the receiving window shrinks 3 positions from left moving the boundary from space 0 to 3.Now, window has shrunk by 3, so the receiver may accept 4 more frame before it is required to send an ACK. In the next step ACK3 is send specifying the sequence number of the next frame to send. As the receiver has send ACK3 acknowledging all the 3 frames which have been received, the window of the receiver expands to include as many new placeholders as newly acknowledged frame.ie, It expands by 3 places to right and the size of the receiver window is again back to 7.Once the acknowledgment ACK3 has reached the sender, it implies that three frames (0, 1, 2) have been received by receiver and are undamaged. The sender's window will now expand to include the next three frames (7, 8 & 9).

In the next step 4 frames, 3 through 6 have been transmitted by the sender shrinking the window size to 3.once these frames reaches the receiver, the receiving window shrinks by 4 places moving the boundary from 3 to 7.The receiver can accept only 3 more frames. The receiver sends ACK4 acknowledging the reception of frame no 3 and both the receiving window and sending window slides 1 position to right.



(Fig 1.25) Example of a Sliding-Window Protocol

5.3 ERROR CONTROL

The Network should ensure complete and accurate delivery of data from the source node to destination node. The end to end transfer of data from a transmitting application to a receiving application involves many steps, each subject to error. Error control refers to mechanisms to detect errors that occur in the transmission of frames and take corrective steps to make sure frames are received correctly.

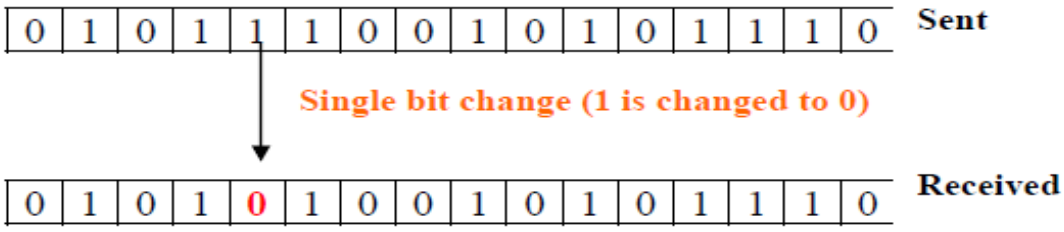
5.3.1 TYPES OF ERRORS

Several types of error may occur during transmission over the network:

- **1-bit error**
- **burst error**
- **lost message (frame)**

1-Bit Error

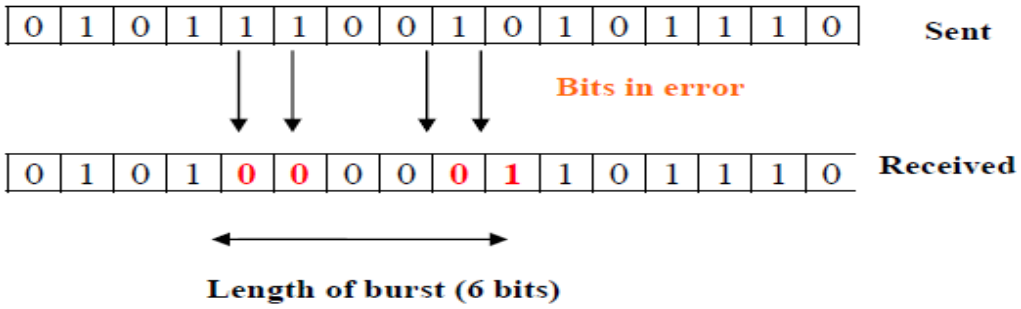
1-bit error/Single bit error means that only one bit is changed in the data during transmission from the source to the destination node i.e., either 0 is changed to 1 or 1 is changed to 0.



(Fig 1.26) Single Bit Error

Burst Error

The term burst error means that two or more bits in the data unit have changed from 0 to 1 or vice-versa. Note that burst error doesn't necessary means that error occurs in consecutive bits. The length of the burst error is measured from the first corrupted bit to the last corrupted bit. Some bits in between may not be corrupted.



(Fig 1.27) Burst Error

Lost Message (Frame)

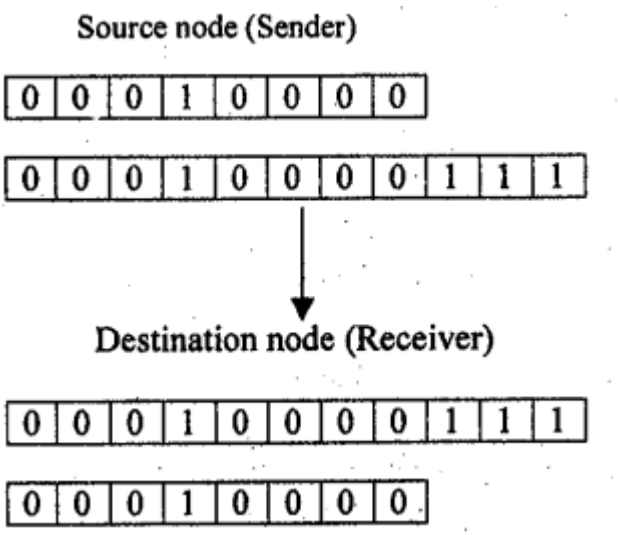
The sender has sent the frame but that is not received properly, this is known as loss of frame during transmission. To deal with this type of error, a retransmission of the sent frame is required by the sender.

5.4 ERROR DETECTION

Accurate delivery of data at the receiver's site is very important in a network application. This implies that the receivers should get the data that is error free. However, due to some factors if, the data gets corrupted, we need to correct it using various techniques. So, we require error detection methods first to detect the errors in the data before correcting it.

For error detection the sender can send every data unit twice and the receiver will do bit by bit comparison between the two sets of information. Any alteration found after the comparison will, indicate an error and a suitable method can be applied to correct the error. But, sending every data unit twice increases the transmission time as well as overhead in comparison. Hence, the basic strategy for dealing with errors is to include groups of bits as additional information in each transmitted frame, so that, the receiver can detect the presence of errors. This method is called Redundancy as extra bits appended in each frame are redundant. At the receiver end these extra bits will be discarded when the accuracy of data is confirmed.

EXAMPLE:



(Fig 1.28) Example: Redundant Bits

Three types of redundancy check methods are commonly used in data transmission:

- **Parity check**
- **CRC**
- **Checksum**

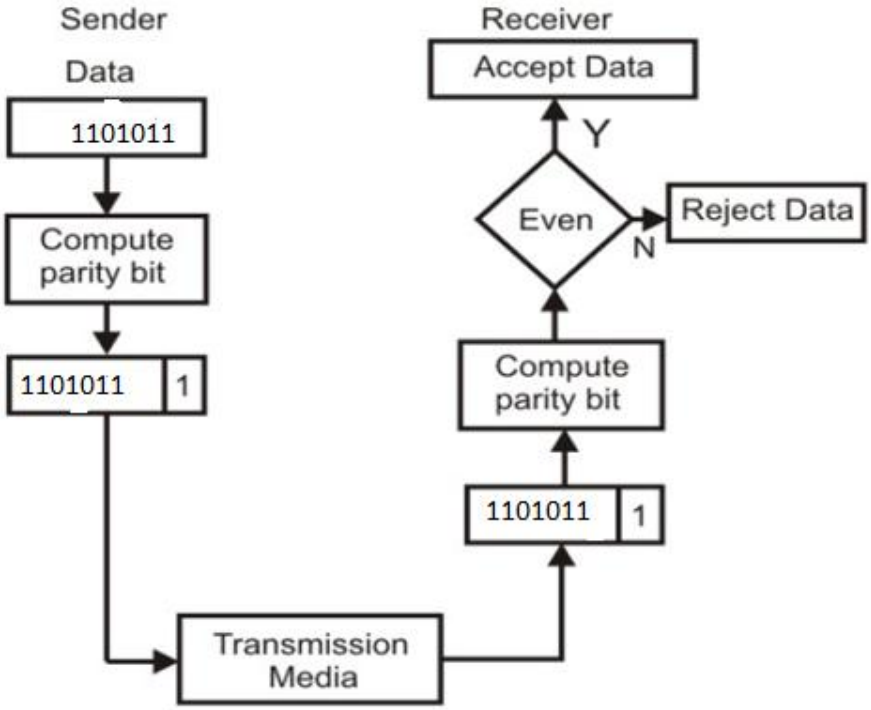
5.4.1 PARITY CHECK

This is the most common and least expensive mechanism for error detection. Parity check can be simple or two dimensional.

Simple Parity Checking or One-dimension Parity Check

This is the easiest method used for detecting errors when the number of bits in the data is small. A parity bit is an extra binary digit added to the group of data bits, so that, the total number of one's in the group is even or odd.. Data bits in each frame is inspected prior to transmission and an extra bit (the parity bit) is computed and appended to the bit string to ensure even or odd parity.

If odd parity is being used, the receiver expects to receive a block of data with an odd number of 1's. For even parity, the number of 1's should be even.



(Fig:1.29) Even Parity Checking Scheme

If a 7-bit ASCII character set is used, a parity is added as the eighth bit. Here the character 'k' which is **1101011** in binary is transmitted by applying even parity. In order to make the number of 1's even the binary digit 1 is appended to the unit and transmitted as follows 11010111.There is now an even number of 1's(six).If odd parity was used a 0 would have been added at the end, resulting in 11010110.

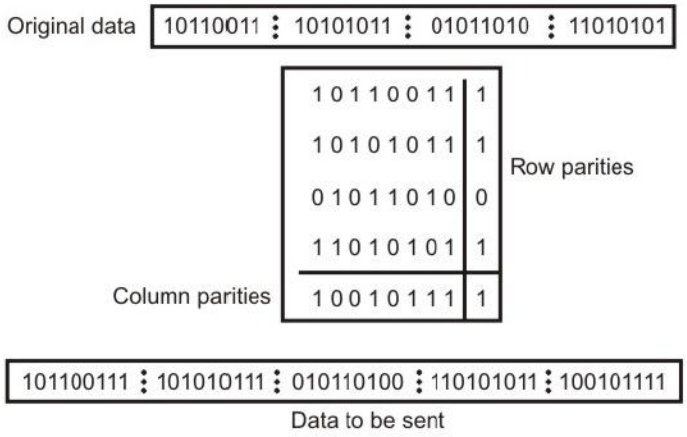
If the transmission error causes one of the bits to be flipped, at the receivers side the number of 1's received will be odd and know that there is an error. The main disadvantage of simple parity bit is that it will fail to detect any error patterns that introduce an even number of errors since the resulting code word will also have even parity. For example 11010111 is send with even parity and during the transmission 2 bits are corrupted.ie, 00010111 is received. Here the error will not be detected, because the number of 1's is still even. Simple parity can there for detect only odd number of erroneous bits per character. The simple parity produces relatively high ratios of check bits to data bits, while achieving only 50% error-detection.

Longitudinal Parity

Longitudinal Parity also known as Two-Dimension Parity tries to solve weakness of simple parity, ie. even numbers of errors are not detected. Parity check bits are calculated for each row, which is equivalent to a simple parity check bit. Parity check bits are also calculated for all columns then both are sent along with the data.

In other words, after sending a set of code words a row of parity bits is also sent. Each parity bit in this row is a parity check for all bits in the column above it. At the receiving end these are compared with the parity bits calculated on the received data.

If one bit is altered in Row1, parity bit for Row1 as well as the parity bit for corresponding column signals an error. If two bits in Row1 are flipped, the Row1 parity check will not signal an error, but two column parity checks will signal an error. This is how longitudinal parity is able to detect more errors than simple parity. However if two bits are flipped in Row1 and two bits are flipped in Row2, and errors occur in same column, no errors will be detected.



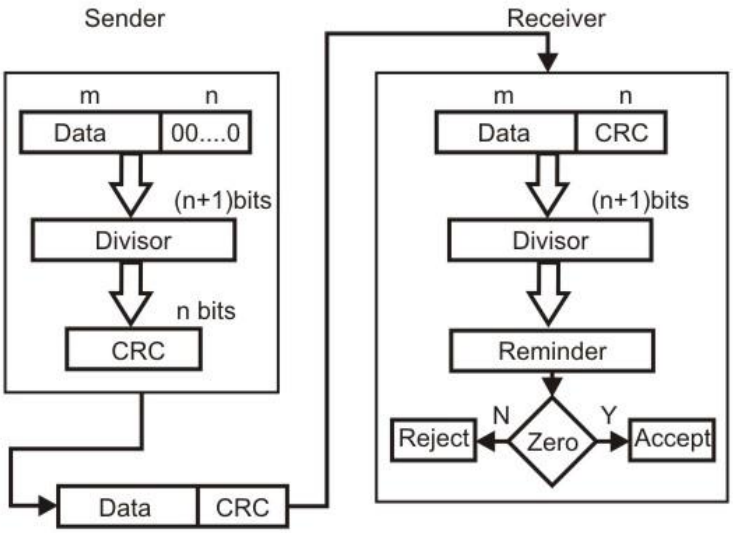
(Fig:1.30) Longitudinal Parity

Although, longitudinal parity provides an extra level of protection by using double parity check, this method like simple parity, also introduces a high number of check bits relative to data bits.

5.4.2 CYCLIC REDUNDANCY CHECK (CRC)

The Cyclic Redundancy Check is the most powerful and easy to implement error detection technique. The CRC is based on modulo arithmetic, where there are no carriers for addition and borrows for subtraction. In CRC sender divides frame (data string) by a predetermined Generator Polynomial and then appends the remainder (called checksum or CRC) onto the frame before starting the process of transmission. A generating polynomial is an industry approved bit string that is used to create cyclic checksum remainder. At the receiver end, the receiver divides the received frame by the same Generator polynomial. If the remainder obtained after the division is zero, it ensure that data received at the receiver's end is error free and accepted. A remainder indicates that the data unit has been damaged in transmit and therefore must be rejected.

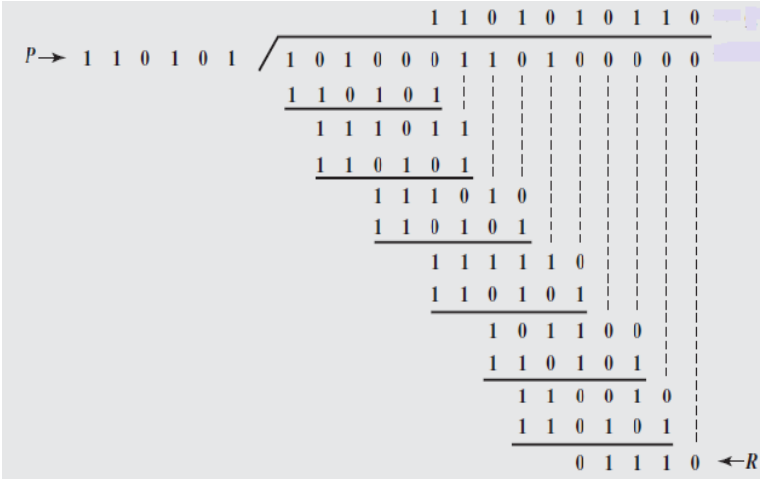
To be valid, a CRC must have two qualities: It must have exactly one less bit than the divisor, and appending it to the end of the data string must make the resulting bit sequence exactly divisible by the divisor. The following figure provides an outline of the basic steps.



(Fig:1.31)Cyclic Redundancy Check

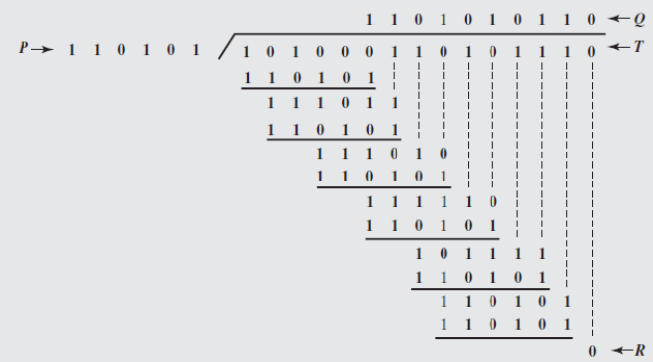
Suppose a m bit message is to be transmitted and we are using a generating polynomial of length n+1. Append the original message by n 0's and divide it by generating polynomial. The remainder of this division process generates the n-bit (CRC) remainder which will be appended to the m-bit message producing (m+n) bit frame for transmission. On receiving the packet, the receiver divides the (m+n) bit frame by the same generating polynomial and if it produces no remainder ,no error has occurred.

Let the data which needs to be send be D= 1010001101. Let the predetermined bit pattern be P=110101.Here m=10 and n+1=6.Multiply the value D by 2^5.The division process shown below:



(Fig:1.32) Cyclic Redundancy Check Step 1

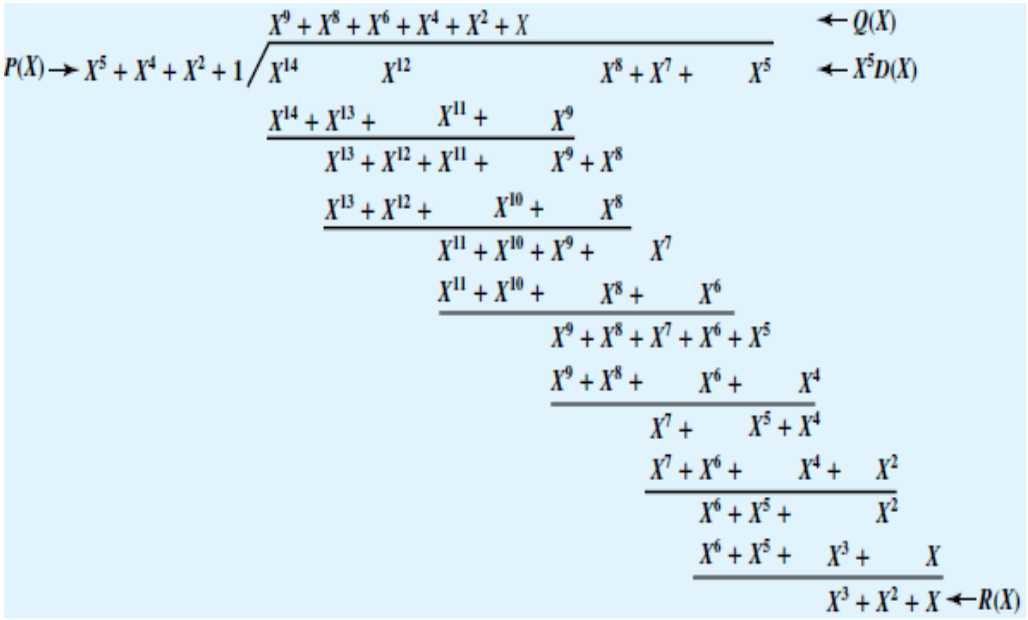
The remainder is added to D to give T = 10100011010111O, which is transmitted. If there are no errors, the receiver receives T intact. The received frame is divided by P and if there is no remainder, it is assumed that there have been no errors.



(Fig:1.33) Cyclic Redundancy Check Step 11

Second way of viewing the CRC process is to express all values as polynomials in a dummy variable X, with binary coefficients. The coefficients correspond to the bits in the binary number. The polynomial format is useful for two reasons: It is short, and it can be used to prove the concept mathematically.

Using the preceding example, for *D=1010001101* we have $D(X) = X^9 + X^7 + X^3 + X^2 + 1$, and for *P=110101* we have $P(X) = X^5 + X^4 + X^2 + 1$ and after division we obtain a remainder *R=01110* which corresponds to $R(X)=X^3+X^2+X$.The following figure shows the polynomial division that corresponds to the binary division in the preceding example:



(Fig:1.34) Example of Polynomial Division

Cabling media can be classified into three categories—twisted pair, coaxial and optical fiber.

- **Twisted-pair cables**

A twisted-pair consists of two individual insulated copper wires physically twisted together. The two wires are twisted together to minimize unwanted electromagnetic signals from interfering with or radiating from the pair. A wire pair acts as a single telecommunications path. Typically, a number of twisted-pairs are bundled into a cable by wrapping them in a protective sheath.

Coaxial cables

Coaxial cable is commonly referred to as coax. All coax consists of a central copper core surrounded by a layer of insulating material. This insulation is enveloped by a metallic wire mesh or, in some cases, a solid metallic sleeve. All of this is then protected by an outer layer of non-conducting material. Both the central core and the mesh or sleeve are capable of conducting electrical signals.

Optical fiber cables

Optical fiber cable contains glass fibers rather than copper wire. Signals are transmitted across these fibers in the form of light pulses rather than electrical pulses—as is the case with metallic cables (twisted-pair and coax). Optical fiber strands are thin filaments of glass consisting of an inner core and an outer cladding. Signals are transmitted as light pulses through the core of the optical fiber. When these light pulses strike the cladding they are reflected back to the core—because the glass used in the cladding has a lower refractive index than the core. This prevents the transmission signal from being lost. Although the majority of LANs connect devices using a physical cable, there are instances where it is difficult or impossible to install cable. In such cases, wireless transmission media is used to connect network devices. Wireless systems do not physically connect network devices since the links between the devices are invisible. They are either infrared light or radio links.

Infrared Links

Connecting devices using infrared light signals work essentially the same way that remote controls work with television sets. These systems consist of a base unit connected to the server and device connections for the stations. The base unit has two optical nodes—one to receive signals from the station and one to send signals to the station. Since the system depends on infrared light to transmit, a requirement is that the base unit and the station connections are in a direct line of sight to each other. Alternatively, some of these systems use a reflective surface positioned between the base unit and the station to redirect the signal.

Radio Links

This second type of wireless media uses radio waves to transmit information between the server and the stations. These systems also work with two components—a Control Module connected to the server and User Modules which connect to the network devices. The control module and the user module do not have to be in direct line-of-sight with each other. The radio signals are capable of passing through most office building doors and walls.

The primary differences between media are their cost and ease of installation; the bandwidth of the cable, which may or may not permit several transmission sessions to occur simultaneously; the maximum speed of communications permitted; and the geographical scope of the network that the medium supports.

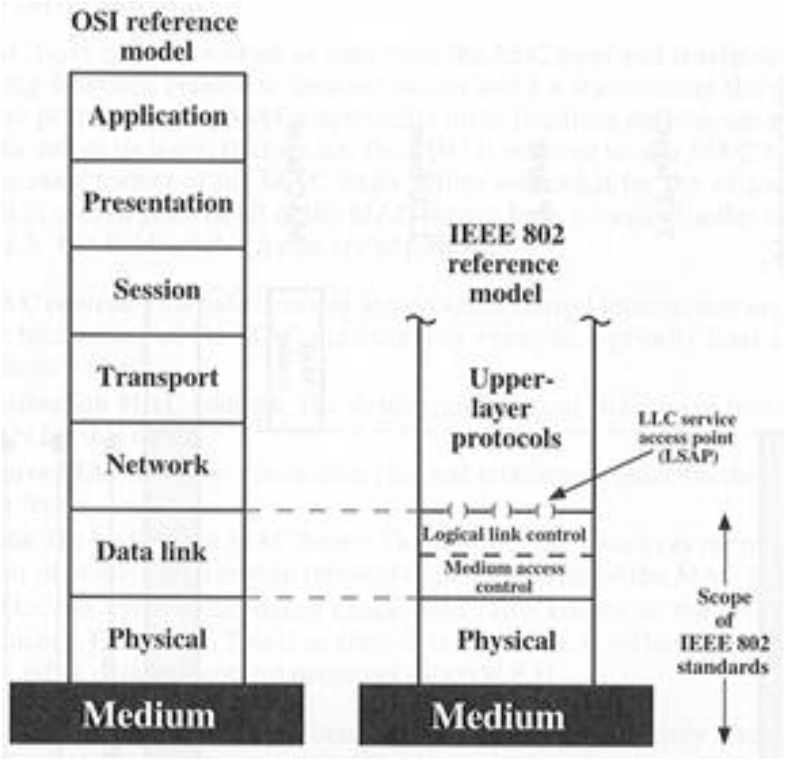
5.5 ACCESS METHODS

If the topology of a local area network can be compared to a data highway, then the access method might be viewed as the set of rules that enable data from one workstation to successfully reach the destination via the data highway. Without such rules, it is quite possible for two messages sent by two different workstations to collide, with the result that neither message reaches its destination. Two common access methods primarily employed in local area networks are carrier-sense multiple access/collision detection (CSMA/CD) and token passing. Each of these access methods is uniquely structured to address the previously mentioned collision and data destination problems.

5.5.1. IEEE 802 REFERENCE MODEL

The layers of OSI Reference Model can be classified as Network Support Layers (*Physical, Data link, Network*) and User support layers (*Session, Presentation, and Application*). LAN protocols are concerned with the network support layers, mainly Physical and Data Link Layer. This is because the functionality of these two layers are sufficient for the delivery of data within the frame work of standard LAN topologies: star, bus, tree etc. The higher layers of the OSI model are independent of network architecture and are applicable to LAN, MAN and WANs.

The relationship of the 802 Standard to the traditional OSI model is shown in figure below.



(Fig 2.5) IEEE 802 Protocol Layers Compared to OSI Model

Working from the bottom up, the lowest layer of the IEEE 802 reference model corresponds to the physical layer of the OSI model and includes such functions as:

- Encoding/decoding of signals
- Preamble generation/removal (for synchronization)
- Bit transmission/reception

In addition, the physical layer of the 802 model includes a specification of the transmission medium and the topology. The data link layer has been divided into two sub layers: logical link control (LLC) and media access control (MAC).The logical link control(LLC) sublayer is primary responsible for logical addressing and providing error control and flow control information. Media Access Control (MAC) defines the specific access method for each LAN. It also defines the layout and format of the data frame depending upon the type of the local area network.

5.5.2 Logical Link Control (LLC)

In IEEE Project 802, flow control, error control, and part of the framing duties are collected into one sub layer called the logical link control. The Logical Link Control is the upper sub layer of the Data Link layer. LLC masks the underlying network technology by hiding their differences hence providing a single interface to the network layer.

The LLC sub-layer is close to the network layer and it receives data frames from the Network layer and passes them to the MAC sub-layer. The functionality of the Network layer is to manage the communications between different networks (different LANs). The LLC sub-layer is the interface between the Network layer and the MAC sub-layer. LLC standard is common to all LAN's and specifies the mechanisms for addressing stations across the medium and for controlling the exchange of data between two users. It offers three types of services for the attached devices: unacknowledged connectionless, acknowledged connectionless and connection-oriented.

Unacknowledged connectionless service: This service is a datagram-style service. It is a very simple service that does not involve any of the flow- and error control mechanisms. Thus, the delivery of data is not guaranteed.

Connection-mode service: A logical connection is set up between two users exchanging data, and flow control and error control are provided.

Acknowledged connectionless service: This is a cross between the previous two services. It provides that datagrams are to be acknowledged, but no prior logical connection is set up

Higher-level data are passed down to LLC, which appends control information as a header, creating an LLC protocol data unit (PDU).This control information is used in the operation of the LLC protocol. The entire LLC PDU is then passed down to the MAC layer, which appends control information at the front and back of the packet, forming a MAC frame. Again, the control information in the frame is needed for the operation of the MAC protocol.

LLC defines a protocol data unit (PDU) which contains 4 fields: one byte DSAP, One byte SSAP, a one or 2 byte control field and a variable length information field. The Destination Service Access Point (DSAP), and the Source Service Access Point (SSAP) are address fields. Each of these is an 8-bit field and contains a 7-bit address, which specifies the destination and source users of LLC. One bit of the DSAP indicates whether the DSAP is an individual or group address. One bit of the SSAP indicates whether the PDU is a command or response PDU.

(Fig:2.6) LLC Protocol Data Unit

The control field of PDU describes type of PDU and includes other information such as sequencing and flow control information. LLC specifies 3 kinds of PDU: information, supervisory and unnumbered. An information PDU (I-PDU) is used to transfer data over a connection oriented service. A supervisory PDU(S-PDU) provides flow and error control but cannot carry any data. An unnumbered PDU (U-PDU) is used to transfer user data over a connectionless service or to transport management information over a connection oriented service.

5.5.3 Medium Access Control

MAC sub-layer is the lower sub-layer close to the physical layer. It has a significant amount of control over the physical layer and therefore the source of its name (Medium Access Control). All LANs and MANs consist of collections of devices that must share the network's transmission capacity. Some means of controlling access to the transmission medium is needed to provide for an orderly and efficient use of that capacity. This is the function of a medium access control.

The key parameters in any medium access control technique are ***where*** and ***how***. Where refers to whether control is exercised in a centralized or distributed fashion. In a centralized scheme, a controller is designated that has the authority to grant access to the network. A station wishing to transmit must wait until it receives permission from the controller. In a decentralized network, the stations collectively perform a medium access control function to determine dynamically the order in which stations transmit.

The second parameter, how, is constrained by the topology and is a tradeoff among competing factors, including cost, performance, and complexity. In general, we can categorize access control techniques as being either synchronous or asynchronous. With synchronous techniques, a specific capacity is dedicated to a connection. This is the same approach used in circuit switching, frequency division multiplexing (FDM), and synchronous time division multiplexing (TDM). Such techniques are generally not optimal in LANs and MANs because the needs of the stations are unpredictable. It is preferable to be able to allocate capacity in an asynchronous (dynamic) fashion, more or less in response to immediate demand. The asynchronous approach can be further subdivided into three categories: **round robin, reservation, and contention**.

Round Robin

With round robin, each station in turn is given the opportunity to transmit. During that opportunity, the station may decline to transmit or may transmit subject to a specified upper bound, usually expressed as a maximum amount of data transmitted or time for this opportunity. In any case, the station, when it is finished, relinquishes its turn, and the right to transmit passes to the next station in logical sequence.

When many stations have data to transmit over an extended period of time, round-robin techniques can be very efficient. If only a few stations have data to transmit over an extended period of time, then there is a considerable overhead in passing the turn from station to station, because most of the stations will not transmit but simply pass their turns. Under such circumstances other techniques may be preferable.

Reservation

For stream traffic, reservation techniques are well suited. Stream traffic is characterized by lengthy and fairly continuous transmissions; examples are voice communication, telemetry, and bulk file transfer. In general, for these techniques, time on the medium is divided into slots, much as with synchronous TDM. A station wishing to transmit reserves future slots for an extended or even an indefinite period

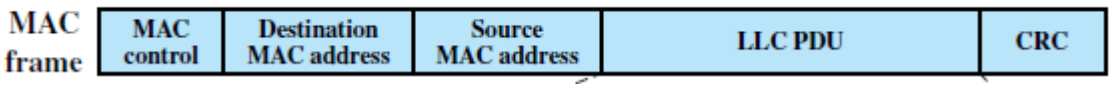
Contention

For bursty traffic, contention techniques are usually appropriate. Bursty traffic is characterized by short, sporadic transmissions. With these techniques, no control is exercised to determine whose turn it is; all stations contend for time in a way that can be, as we shall see, rather rough and tumble. These techniques are of necessity distributed in nature. Their principal advantage is that they are simple to implement and, under light to moderate load, efficient.

MAC Frame Format

The MAC layer receives a block of data from the LLC layer and is responsible for performing functions related to medium access and for transmitting the data. As with other protocol layers, MAC implements these functions making use of a protocol data unit at its layer. In this case, the PDU is referred to as a MAC frame. The fields of this frame are:

- **MAC Control:** This field contains any protocol control information needed for the functioning of the MAC protocol. For example, a priority level could be indicated here.
- **Destination MAC Address:** The destination physical attachment point on the LAN for this frame
- **Source MAC Address:** The source physical attachment point on the LAN for this frame.
- **LLC:** The LLC data from the next higher layer.
- **CRC:** The Cyclic Redundancy Check field



(Fig:2.7) MAC Frame Format

In most data link control protocols, the data link protocol entity is responsible not only for detecting errors using the CRC, but for recovering from those errors by retransmitting damaged frames. In the LAN protocol architecture, these two functions are split between the MAC and LLC layers. The MAC layer is responsible for detecting errors and discarding any frames that are in error. The LLC layer optionally keeps track of which frames have been successfully received and retransmits unsuccessful frames.

5.6 IEEE 802 STANDARDS

While connecting computers through networks we need to have set of rules/standards for the data to travel from one computer to other computer. The right example for this can be road traffic rules. It's self under stood, why we need traffic rules while driving, in same sense for the data packets to travel from one computer terminal to other terminal they should also follow set of rules and regulations. When local area networks (LANs) first began to emerge as potential business tools in the late 1970s, the IEEE (Institute of Electrical and Electronics Engineers, Inc.) realized that there was a need to define certain LAN standards. As there were different types of LAN's available on market it was a difficult task for the users and vendors to choose a particular LAN for their specific applications. The IEEE is the world's leading professional association for the advancement of technology. It's a non- profit organization offering its members immense benefits.

LAN standards proposed by the IEEE committee have the following goals in mind:

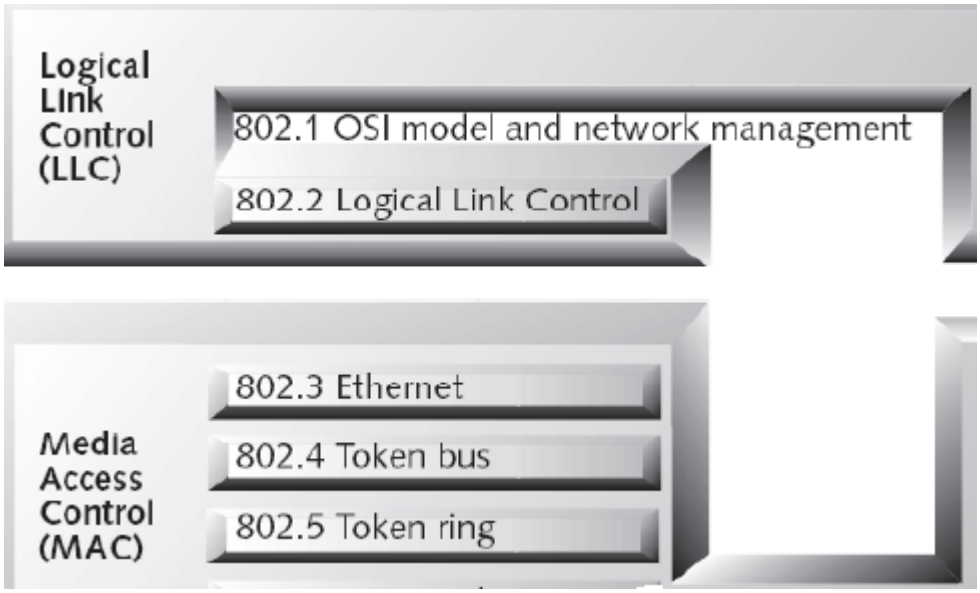
- To promote compatibility
- Implementation with minimum efforts
- Accommodate the need for diverse applications

For the fulfillment of the above mentioned goals, the committee came up with a bunch of LAN standards collectively known as IEEE 802.The various standards differ at the physical layer & MAC sub-layer but are compatible at the data link layer.In general, IEEE 802 standards define physical network interfaces such as network interface cards, bridges, routers, connectors, cables, and all the signaling and access methods associated with physical network connections. This architecture has been adopted by all organizations working on the specification of LAN standards. It is generally referred to as the IEEE 802 reference model. The 802 specification falls into distinct categories each of which has got its own number, as described in the following table:

Standard	Title	Description
802.1	High-level Interface	Specification of standards for LAN architecture, interconnection, management
802.2	Logical Link Control	Specification of standards for the LLC layer
802.3	Ethernet(CSMA/CD)	Specification of standards for CSMA/CD architectures
802.4	Token Bus	Specification of standards for token bus architectures
802.5	Token Ring	Specification of standards for token ring architectures
802.6	Metropolitan Area Networks	Specification of standards for MANs
802.7	Broadband Technical Advisory Group	Provision of guidance to other groups working on broadband LANs
802.8	Fiber Optic Technical Advisory Group	Provision of guidance to other groups working on fiber optic-based LANs
802.9	Integrated Data and Voice Networks	Specification of standards for interfaces to ISDN

(Table 2.1)

The following diagram maps IEEE 802 standards to IEEE Reference Model

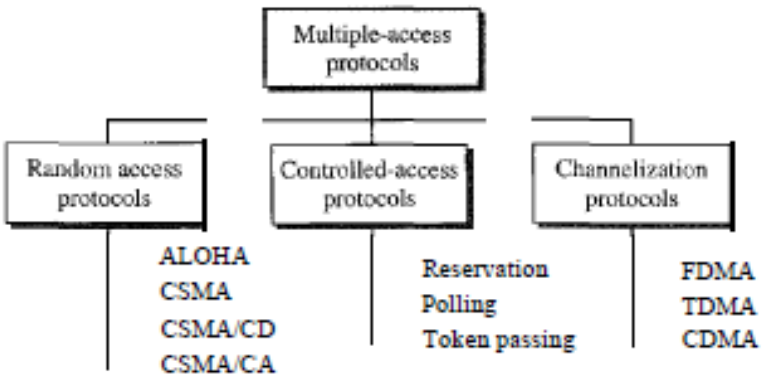


(Fig:)Mapping IEEE Reference Model & IEEE 802 standards

5.7. MULTIPLE ACCESS

We can consider the data link layer as two sub layers. The upper sub layer that is responsible for flow and error control is called the logical link control (LLC) layer; the lower sub layer that is mostly responsible for multiple access resolution is called the media access control (MAC) layer.

When nodes or stations are connected and use a common link, called a multipoint or broadcast link, we need a multiple-access protocol to coordinate access to the link. The problem of controlling the access to the medium is similar to the rules of speaking in an assembly. The procedures guarantee that the right to speak is upheld and ensure that two people do not speak at the same time, do not interrupt each other, do not monopolize the discussion, and so on.Many formal protocols have been devised to handle access to a shared link. We categorize them into three groups.



(Fig) Taxonomy of multiple-access protocols

5.8 RANDOM ACCESS

In random access or contention methods, no station is superior to another station and none is assigned the control over another. No station permits, or does not permit, another station to send. At each instance, a station that has data to send uses a procedure defined by the protocol to make a decision on whether or not to send. This decision depends on the state of the medium (idle or busy). In other words, each station can transmit when it desires on the condition that it follows the predefined procedure, including the testing of the state of the medium.

Two features give this method its name. First, there is no scheduled time for a station to transmit. Transmission is random among the stations. That is why these methods are called random access. Second, no rules specify which station should send next. Stations compete with one another to access the medium. That is why these methods are also called contention methods.

In a random access method, each station has the right to the medium without being controlled by any other station. However, if more than one station tries to send, there is an access conflict-collision-and the frames will be either destroyed or modified. To avoid access conflict or to resolve it when it happens, each station follows a procedure that answers the following questions:

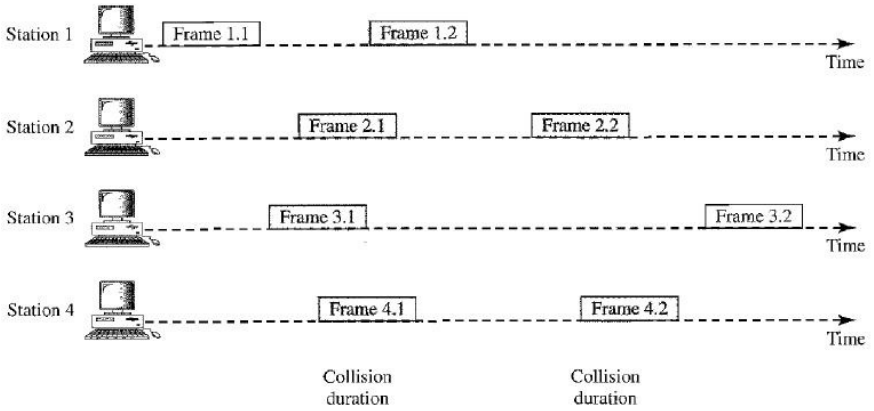
- When can the station access the medium?
- What can the station do if the medium is busy?
- How can the station determine the success or failure of the transmission?
- What can the station do if there is an access conflict?

5.8.1 ALOHA

Suppose when a user has to send data, the user simply transmits the frame. If no other user happened to have a frame to transmit then this frame would be successfully received. What if a collision occurs? The user could simply retransmit, but this would not help, other user involved in the collision will also retransmit, resulting in another collision. One way to avoid this is to wait a random amount of time before retransmitting. The above idea is the basis for the ALOHA protocol. It is the earliest random access method was developed at the University of Hawaii in early 1970. It was designed for a radio (wireless) LAN, but it can be used on any shared medium. There are two versions of ALOHA: **pure** and **slotted.**

5.8.2 PURE ALOHA

The idea is that each station sends a frame whenever it has a frame to send. However, since there is only one channel to share, there is the possibility of collision between frames from different stations. Following figure shows an example of frame collisions in pure ALOHA.

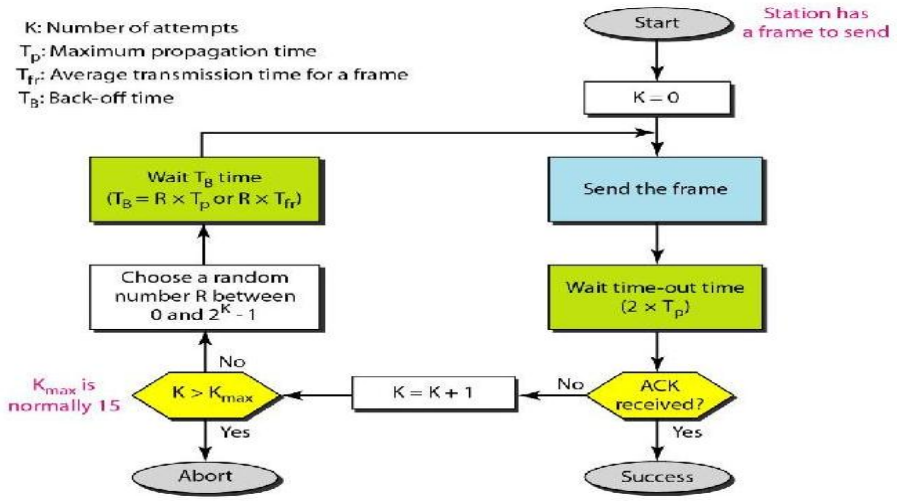


(Fig:) Frames in a pure ALOHA network

There are four stations (unrealistic assumption) that contend with one another for access to the shared channel. The figure shows that each station sends two frames; there are a total of eight frames on the shared medium. Some of these frames collide because multiple frames are in contention for the shared channel. Below figure shows that only two frames survive: frame 1.1 from station 1 and frame 3.2 from station 3. We need to mention that even if one bit of a frame coexists on the channel with one bit from another frame, there is a collision and both will be destroyed.

Procedure for Pure ALOHA Protocol

The pure ALOHA protocol relies on acknowledgments from the receiver. When a station sends a frame, it expects the receiver to send an acknowledgment. If the acknowledgment does not arrive after a time-out period, the station assumes that the frame (or the acknowledgment) has been destroyed and resends the frame. A collision involves two or more stations. If all these stations try to resend their frames after the time-out, the frames will collide again. Pure ALOHA dictates that when the time-out period passes, each station waits a random amount of time before resending its frame. The randomness will help avoid more collisions. We call this time the back-off time T_B. After a maximum number of retransmission attempts, **Kmax** station must give up and try later.



(Fig:2.11) Procedure for pure ALOHA protocol

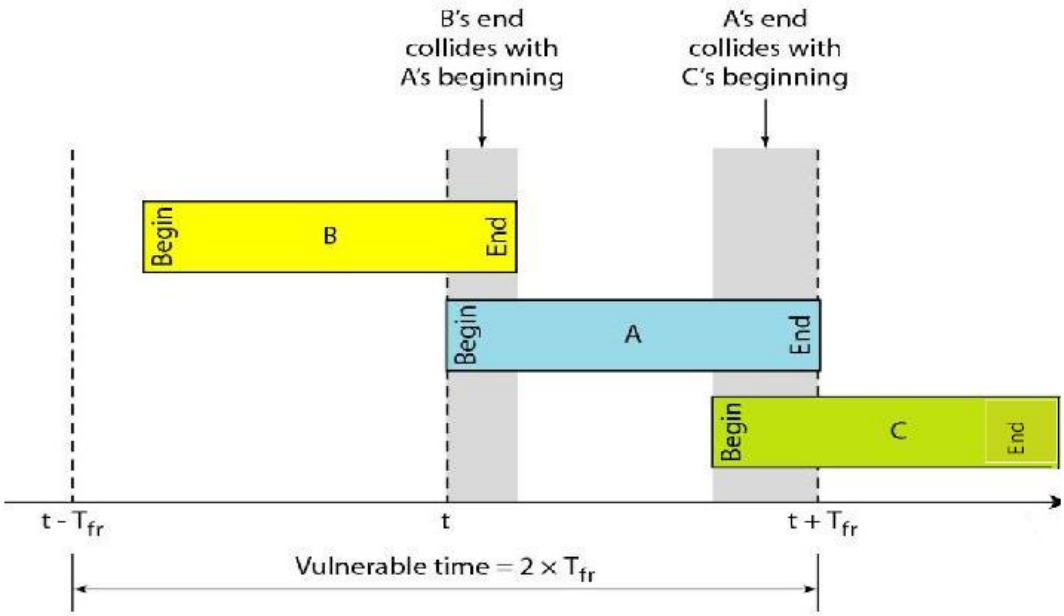
The time-out period is equal to the maximum possible round-trip propagation delay, which is twice the amount of time required to send a frame between the two most widely separated stations (2 x Tp). The back-off time TB is a random value that normally depends on K (the number of attempted unsuccessful transmissions)

Vulnerable Time

Vulnerable time is the time in which there is a possibility of collision. We assume that the stations send fixed length frames with each taking T_{fr} to send.

Station A sends a frame at time t. Now imagine station B has already sent a frame between t - Tfr and t. This leads to a collision between the frames from station A and station B. The end of B's frame collides with the beginning of A's frame. On the other hand, suppose that station C sends a frame between t and t + Tfr. Here, there is a collision between frames from station A and station C. The beginning of C's frame collides with the end of A's frame.

Following figure shows vulnerable time for station A.



(Fig:) Vulnerable time for pure ALOHA protocol

Here we can see that the vulnerable time, during which a collision may occur in pure ALOHA, is 2 times the frame transmission time.

Pure ALOHA vulnerable time = 2 x Tfr

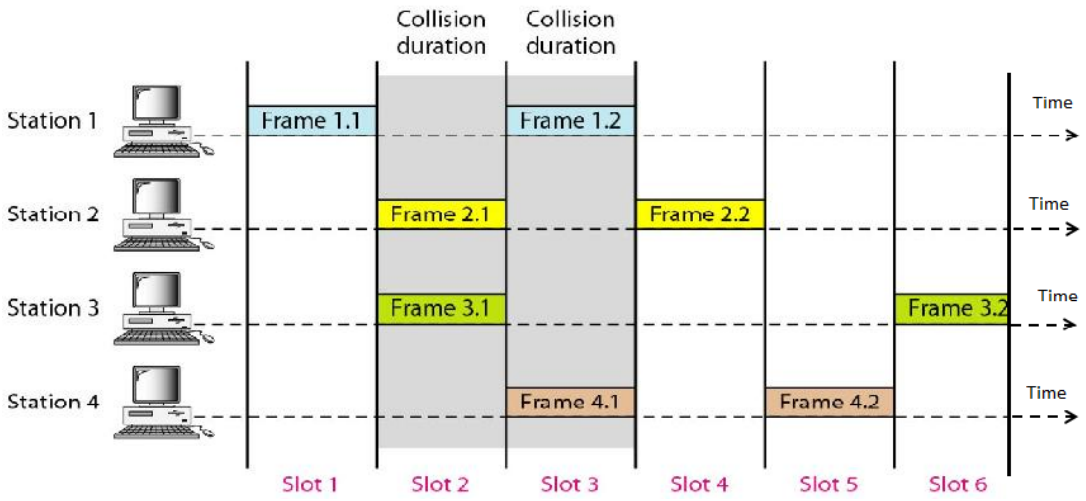
Throughput S of a channel is defined as average number of successful transmission of data frames on the channel per unit time. It is represented as percentage of carrying capacity of the channel.

Let us call G the average number of frames generated by the system during one frame transmission time. Then it can be proved that the average number of successful transmissions for pure ALOHA is S = G x e-2G. The maximum throughput Smax is 0.184, for G = .5. In other words, if one-half a frame is generated during one frame transmission time (in other words, one frame during two frame transmission times), then 18.4 percent of these frames reach their destination successfully. This is an expected result because the vulnerable time is 2 times the frame transmission time. Therefore, if a station generates only one frame in this vulnerable time (and no other stations generate a frame during this time), the frame will reach its destination successfully.

5.8.3 SLOTTED ALOHA

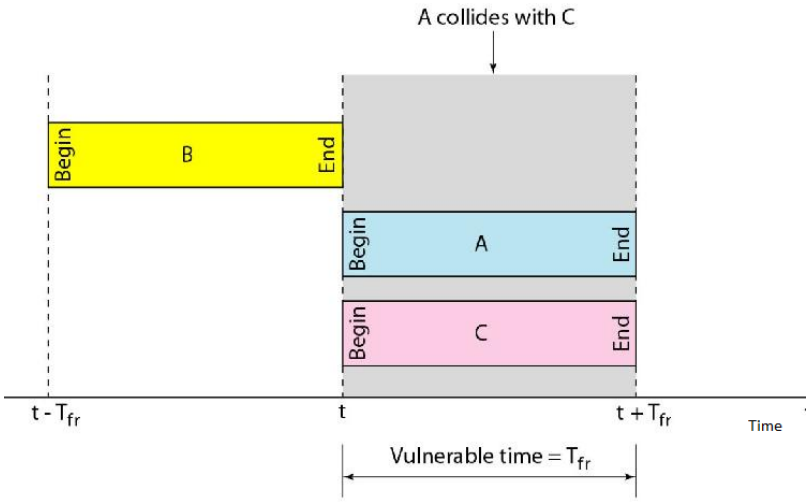
Pure ALOHA has a vulnerable time of **2 x Tfr**. This is so because there is no rule that defines when the station can send. A station may send soon after another station has started or soon before another station has finished. Slotted ALOHA was invented to improve the efficiency of pure ALOHA.

In slotted ALOHA we divide the time into slots of **Tfr** and force the station to send only at the beginning of the time slot.



(Fig:) Frames in a slotted ALOHA network

Because a station is allowed to send only at the beginning of the synchronized time slot, if a station misses this moment, it must wait until the beginning of the next time slot. This means that the station which started at the beginning of this slot has already finished sending its frame. Of course, there is still the possibility of collision if two stations try to send at the beginning of the same time slot. However, the vulnerable time is now reduced to one-half, equal to Tfr



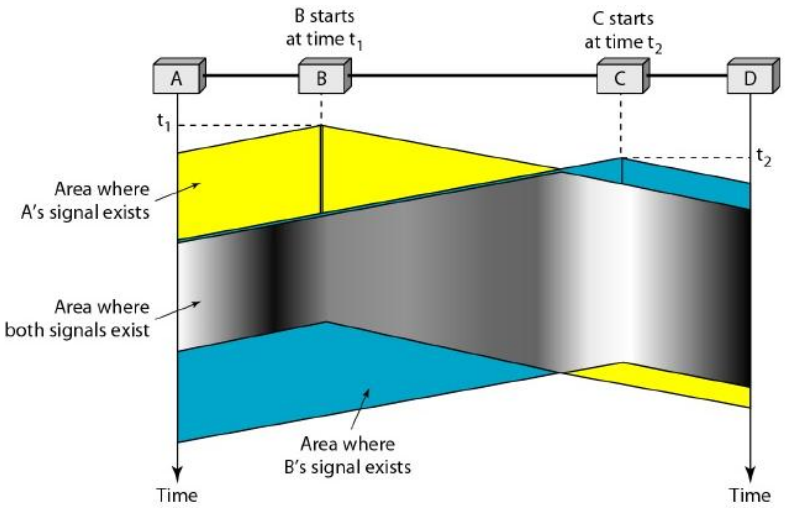
(Fig:) Vulnerable time for slotted ALOHA protocol

It can be proved that the average number of successful transmissions for slotted ALOHA is S = G x e-G. The maximum throughput Smax is 0.368, when G = 1. In other words, if a frame is generated during one frame transmission time, then 36.8 percent of these frames reach their destination successfully. This result can be expected because the vulnerable time is equal to the frame transmission time. Therefore, if a station generates only one frame in this vulnerable time (and no other station generates a frame during this time), the frame will reach its destination successfully.

5.8.4 CARRIER SENSE MULTIPLE ACCESS (CSMA)

The poor efficiency of the ALOHA scheme can be attributed to the fact that a node start transmission without paying any attention to what others are doing. In many multi-access channels it is possible for a node to detect when other nodes are transmitting after a small propagation delay. This is referred as carrier sensing. In situations where propagation delay of the signal between two nodes is small compared to the transmission time of a packet, all other nodes will know very quickly when a node starts transmission. This observation is the basis of the carrier-sense multiple-access (CSMA) protocol. The chance of collision can be reduced if a station senses the medium before trying to use it. Carrier sense multiple access (CSMA) requires that each station first listen to the medium (or check the state of the medium) before sending. In other words, CSMA is based on the principle "sense before transmit" or "listen before talk."

The first question that one might ask about CSMA is that if all stations perform carrier sensing, why do collisions occur in the first place? After all, a station will refrain from transmitting whenever it senses that another node is transmitting. The answer to the question can best be illustrated using space-time diagrams. Following figure shows a space-time diagram of four stations (A, B, C, D) attached to an linear broadcast bus. The horizontal axis shows the position of each station in space; the y-axis represents time.

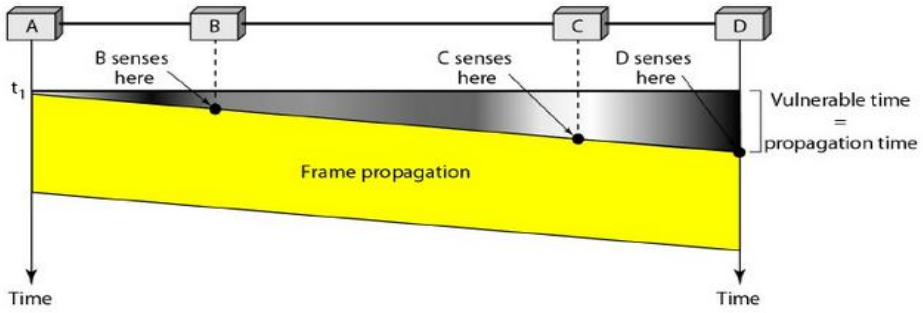


(Fig:) Space/time model of the collision in CSMA

At time t1, node B senses the channel is idle, as no other nodes are currently transmitting. Node B thus begins transmitting, and a non-zero amount of time is needed for B's bits to actually propagate (albeit at near the speed-of-light) along the broadcast medium. At time t2 (t2 > t1), node C has a frame to send. Although node B is currently transmitting at time t1, the bits being transmitted by B have yet to reach D, and thus C senses the channel idle at t1. In accordance with the CSMA protocol, C thus begins transmitting its frame. A short time later, B's transmission begins to interfere with C's transmission at C.

It is evident that the end-to-end channel propagation delay of a broadcast channel - the time it takes for a signal to propagate from one of the channel to another - will play a crucial role in determining its performance. The longer this propagation delay, the larger the chance that a carrier-sensing node is not yet able to sense a transmission that has already begun at another node in the network.

The vulnerable time for CSMA is the propagation time Tp. This is the time needed for a signal to propagate from one end of the medium to the other. When a station sends a frame, and any other station tries to send a frame during this time, a collision will result. But if the first bit of the frame reaches the end of the medium, every station will already have heard the bit and will refrain from sending



(Fig:) Vulnerable Time

CSMA have many variants available that are to be adapted according to the behaviour of the station that has frames to be transmitted when the channel is busy or that some transmission is going on. The following are some versions of CSMA protocols:

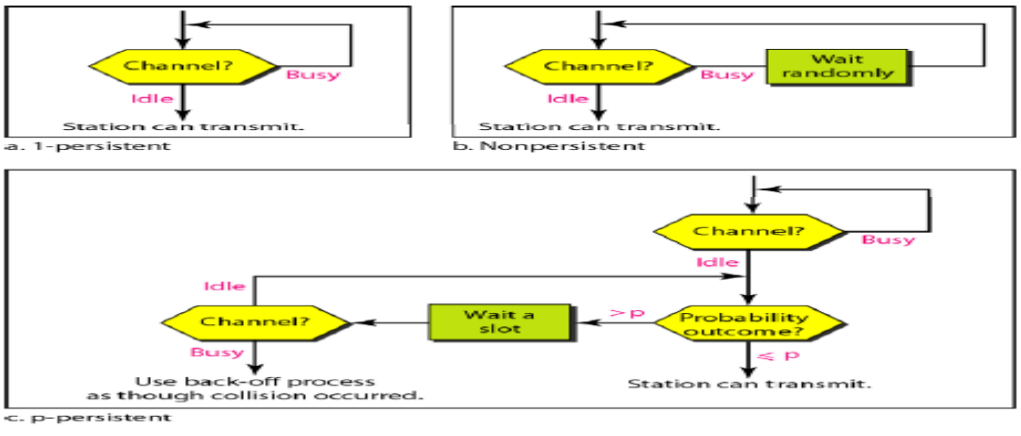
- **1-Persistent CSMA**
- **Non-Persistent CSMA**
- **p-Persistent CSMA**

I-Persistent

A station i.e., who wants to transmit some frame will sense the channel first, if it is found busy ie, some transmission is going on the medium, then, this station will continuously keep sensing that the channel. And as soon as this station finds that the channel has become idle it will transmit its frame. But if more than one station is in waiting state and keeps track of the channel then a collision will occur in the system because both waiting station will transmit their frames at the same time. The other possibility of collision can be if the frame has not reached any other station then, it indicates to the second station that the channel is free. So the second station also starts its transmission and that will lead to collision. Thus I-persistent CSMA a greedy protocol as to capture the channel as soon as it finds it idle. And, hence, it has a high frequency of collision in the system. In case of collision, the station senses the channel again after random delay.

Non-Persistent CSMA

To reduce the frequency of the occurrence of collision in the system then, another version of CSMA, that is non-persistent CSMA can be used. Here, the station who has frames to transmit first sense whether the channel is busy or free. If the station finds that channel to be free it simply transmits its frame. Otherwise, it will wait for a random amount of time and repeat the process after that time span is over. As it does not continuously senses the channel to be, it is less greedy in comparison of I-Persistent CSMA. It reduces the probability of the occurrence of collision as the waiting stations will not transmit their frames at the same time because the stations are waiting for a random amount of time, before restarting the process. Random time may be different for different stations so, the likelihood waiting station will start their transmission at the same time is reduced. But, it can lead to longer delays than the 1- Persistent CSMA.



(Fig) Flow diagram for three persistence methods

P-Persistent CSMA

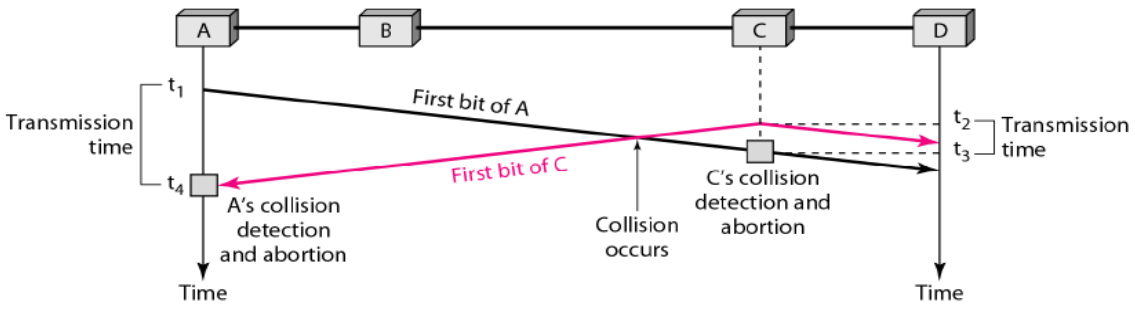
P-Persistent CSMA is used to reduce the probability of collision in 1-persistent, not all waiting stations are allowed to transmit immediately once the medium is free. This category of CSMA combines features of the above versions of CSMA that is 1-persistent CSMA and non-persistent CSMA. This version is applicable for the slotted channel .It reduces the chance of collision and improves efficiency. In this method, after the station finds the line idle it will select a random number and follows these steps.

1. With probability less than p, the station sends its frame.
2. With probability greater than p the station waits for the beginning of the next time slot and checks the line again.
    a. If the line is idle, it goes to step 1.
    b. If the line is busy, it acts as though a collision has occurred and uses the back off procedure. Here value of p is the controlling parameter.

5.8.5 CARRIER SENSE MULTIPLE ACCESS WITH COLLISION DETECTION (CSMA/CD)

CSMA/CD protocol can be considered as a refinement over the CSMA scheme. It has evolved to overcome one glaring inefficiency of CSMA. In CSMA scheme, a transmitting station continues to transmit its frame even though a collision occurs. The channel time unnecessarily wasted due to this. If the propagation time is small (which is usually the case) compared to the packet transmission time, wasted channel capacity can be considerable. This wastage of channel capacity can be reduced if the nodes continue to monitor the channel while transmitting a packet and immediately cease transmission when collision is detected. This refined scheme is known as Carrier Sensed Multiple Access with Collision Detection (CSMA/CD) or Listen-While-Talk.

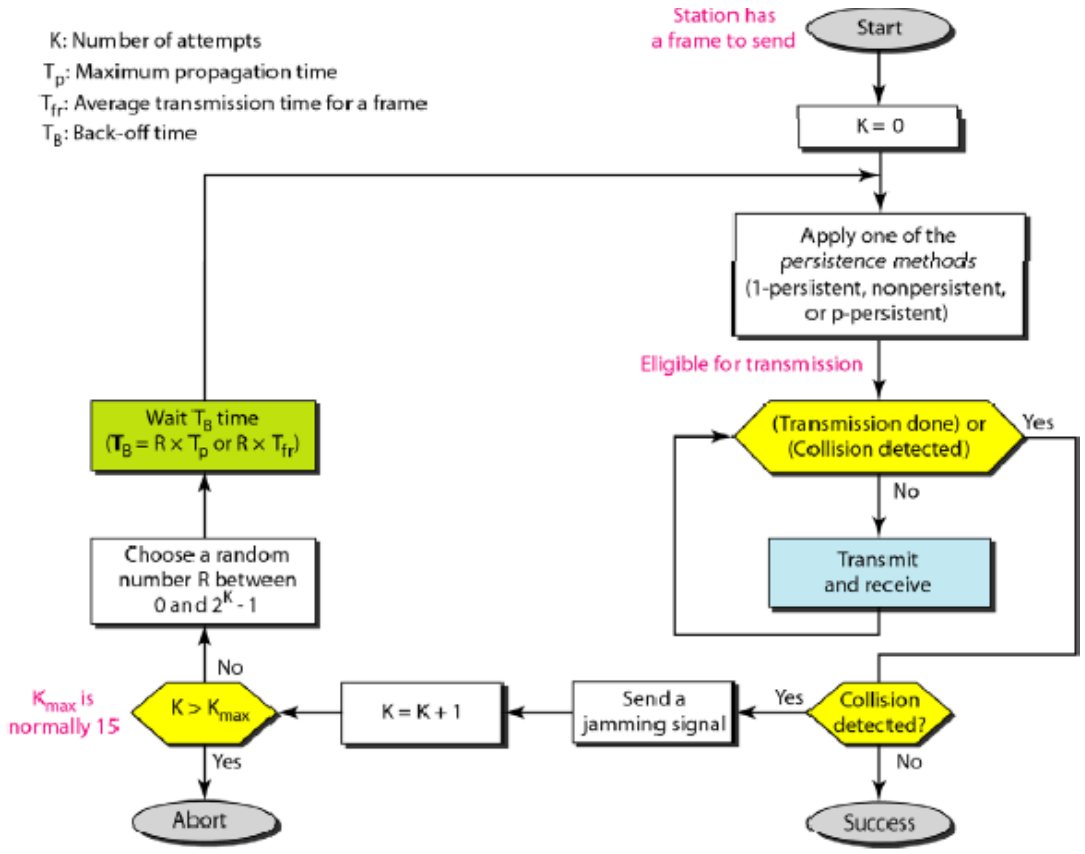
Carrier sense multiple access with collision detection (CSMA/CD) augments the algorithm to handle the collision. In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If a collision is detected rather than finish transmitting frames which are corrupted they should abruptly stop transmission. Quickly terminating damaged frames saves time and bandwidth.



(Fig:) Collision of the first bit in CSMA/CD

At time t1, station A has executed its persistence procedure and starts sending the Bits of its frame. At time t2, station C has not yet sensed the first bit sent by A. Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and to the right. The collision occurs sometime after time t2' Station C detects a collision at time t3 when it receives the first bit of A's frame. Station C immediately (or after a short time, but we assume immediately) aborts transmission. Station A detects collision at time t4 when it receives the first bit of C's frame; it also immediately aborts transmission. Looking at the figure, we see that A transmits for the duration t4 - tl; C transmits for the duration t3 - t2.

Before sending the last bit of the frame, the sending station must detect a collision, if any, and abort the transmission. This is so because the station, once the entire frame is sent, does not keep a copy of the frame and does not monitor the line for collision detection. Therefore, the frame transmission time Tfr must be at least two times the maximum propagation time Tp. To understand the reason, let us think about the worst-case scenario. If the two stations involved in a collision are the maximum distance apart, the signal from the first takes time Tp to reach the second and the effect of the collision takes another time Tp to reach the first. So the requirement is that the first station must still be transmitting after 2Tp.



(Fig) Flow diagram for the CSMA/CD

Let us look at the flow diagram for CSMA/CD It is similar to the one for the ALOHA protocol, but there are differences. The first difference is the addition of the persistence process. We need to sense the channel before we start sending the frame by using one of the persistence processes we discussed previously (non-persistent, I-persistent, or p-persistent).

The second difference is the frame transmission. In ALOHA, we first transmit the entire frame and then wait for an acknowledgment. In CSMA/CD, transmission and collision detection is a continuous process. We do not send the entire frame and then look for a collision. The station transmits and receives continuously and simultaneously (using two different ports). We use a loop to show that transmission is a continuous process. We constantly monitor in order to detect one of two conditions: either transmission is finished or a collision is detected. Either event stops transmission. When we come out of the loop, if a collision has not been detected, it means that transmission is complete; the entire frame is transmitted. Otherwise, a collision has occurred.

The third difference is sending a jamming signal. If a collision is detected during transmission of a packet, the node immediately ceases transmission and it transmits jamming signal for a brief duration to ensure that all stations know that collision has occurred.

5.8.6 EXPONENTIAL BACK OFF ALGORITHM

This algorithm is commonly used by a transmitting station to determine how long to wait following a collision before attempting to retransmit the frame. If all stations waited the same length of time before retransmission, then another collision would inevitably occur. This is avoided by having each station generate a random number which determines the length of time it must wait before testing the carrier.

The retransmission is delayed by an amount of time derived from the slot time and the number of attempts to retransmit. Slot time is the time required for a signal to traverse from one end of the network to another, plus the time required to send the jam signal in case of collision

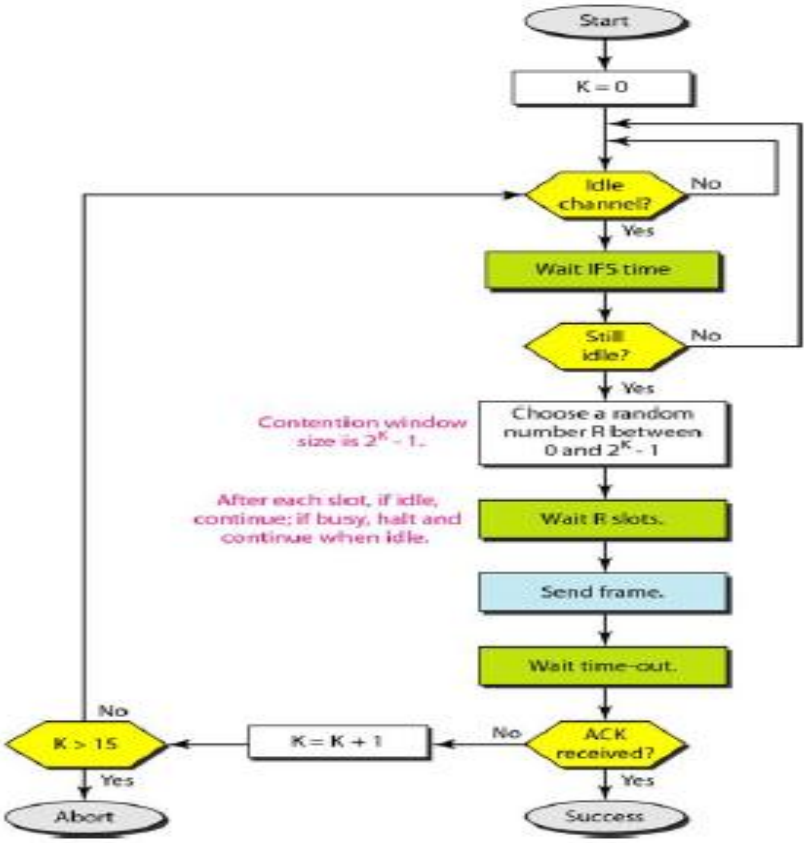
Following a collision, each station generates a random number that falls within a specified range of values. It then waits that number of slot times before attempting retransmission. The range of values increases exponentially after each failed retransmission. For the first attempt the range is 0 to 1; for the second attempt, 0 to 3; for the third, 0 to 7 and so on. After c collisions, the range is between 0 and 2^c - 1. If repeated collisions occur, the range continues to expand until after 10 attempts when it reaches 0 to 1023. After that the range of values stays fixed from 0 to 1023. If a station is unsuccessful in transmitting after 15 attempts, the MAC function reports an "excessive collision error". The frame being transmitted is then dropped, requiring that application software detect its loss and initiate a retransmission

5.8.7 CARRIER SENSE MULTIPLE ACCESS WITH COLLISION AVOIDANCE (CSMA/CA)

CSMA/CA is a modification of Carrier Sense Multiple Access. Collision avoidance is used to improve the performance of CSMA by attempting to be less greedy on the channel. If the channel is sensed busy before transmission then transmission is deferred for a random interval. This reduces the probability of collisions on the channel.

In CSMA a station wishing to transmit has to first listen to the channel for a predetermined amount of time so as to check for any activity on the channel. If the channel is sensed idle then the station is permitted to transmit. If the channel is sensed busy the station has to defer its transmission. This is the essence of both CSMA/CD and CSMA/CA. In CSMA/CA once the channel is clear, it again waits for an additional time period before performing the transmission

CSMA/CA is used where CSMA/CD cannot be implemented due to the nature of the channel. For example CSMA/CA is used in wireless LANs. One of the problems of wireless LAN's is that it is not possible to listen while sending. There fore collision detection is not possible. Another reason is hidden terminal problem, where by a node A, in the range of receiver R is not in the range of the sender S, and therefore cannot know that S is transmitting to R. Collisions are avoided through the use of CSMA/CA's three strategies: **the inter frame space**, **the contention window, and acknowledgments.**



(Fig:) Flow diagram for CSMA/CA

Inter Frame Space

Collisions are avoided by deferring transmission even if the channel is found idle. When an idle channel is found, the station does not send immediately. It waits for a period of time called the inter frame space or IFS. Even though the channel may appear idle when it is sensed, a distant station may have already started transmitting. The distant station's signal has not yet reached this station. The IFS time allows the front of the transmitted signal by the distant station to reach this station. If after the IFS time the channel is still idle, the station can send, but it still needs to wait a time equal to the contention time. The IFS variable can also be used to prioritize stations or frame types.

Contention Window

The contention window is an amount of time divided into slots. A station that is ready to send chooses a random number of slots as its wait time. The number of slots in the window changes according to the binary exponential back-off strategy. This means that it is set to one slot the first time and then doubles each time the station cannot detect an idle channel after the IFS time. This is very similar to the p-persistent method except that a random outcome defines the number of slots taken by the waiting station. One interesting point about the contention window is that the station needs to sense the channel after each time slot. However, if the station finds the channel busy, it does not restart the process; it just stops the timer and restarts it when the channel is sensed as idle. This gives priority to the station with the longest waiting time.

Acknowledgment

With all these precautions, there still may be a collision resulting in destroyed data. In addition, the data may be corrupted during the transmission. The positive acknowledgment and the time-out timer can help guarantee that the receiver has received the frame

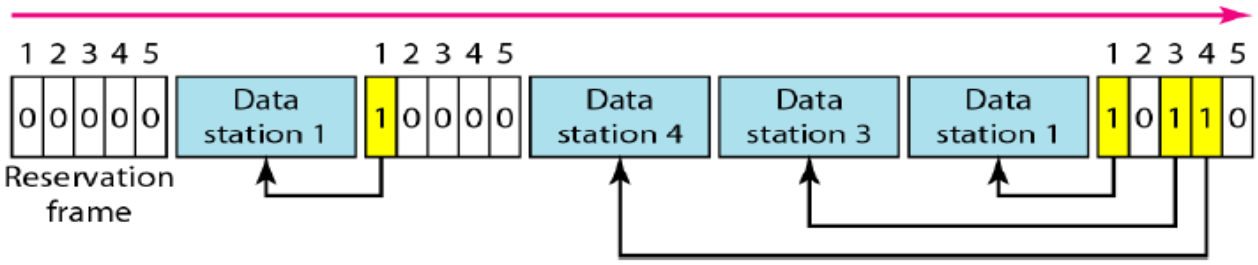
5.9 CONTROLLED ACCESS

In controlled access, the stations consult one another to find which station has the right to send. A station cannot send unless it has been authorized by other stations. Three popular control access methods are:

Reservation
Polling
Token Passing

5.9.1 RESERVATION

In the reservation method, a station needs to make a reservation before sending data. Time is divided into intervals. In each interval, a reservation frame precedes the data frames sent in that interval. If there are N stations in the system, there are exactly N reservation minislots in the reservation frame. Each minislot belongs to a station. When a station needs to send a data frame, it makes a reservation in its own minislot. The stations that have made reservations can send their data frames after the reservation frame.

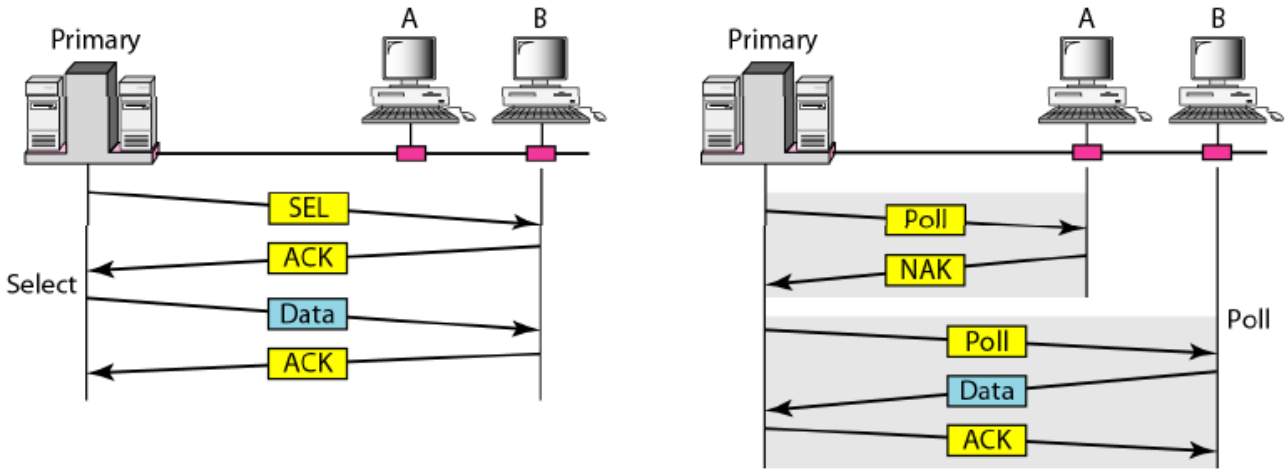


(Fig:) Reservation access method

The above figure shows a situation with five stations and a five-minislot reservation frame. In the first interval, only stations 1, 3, and 4 have made reservations. In the second interval, only station 1 has made a reservation.

5.9.2 POLLING

Polling works with topologies in which one device is designated as a primary station and the other devices are secondary stations. All data exchanges must be made through the primary device even when the ultimate destination is a secondary device. The primary device controls the link; the secondary devices follow its instructions. It is up to the primary device to determine which device is allowed to use the channel at a given time. The primary device, therefore, is always the initiator of a session. If the primary wants to receive data, it asks the secondaries if they have anything to send; this is called **poll function**. If the primary wants to send data, it tells the secondary to get ready to receive; this is called **select function.**



(Fig:) Select and poll functions in polling access method

The select function is used whenever the primary device has something to send. Remember that the primary controls the link. If the primary is neither sending nor receiving data, it knows the link is available. If it has something to send, the primary device sends it. What it does not know, however, is whether the target device is prepared to receive. So the primary must alert the secondary to the upcoming transmission and wait for an acknowledgment of the secondary's ready status. Before sending data, the primary creates and transmits a select (SEL) frame, one field of which includes the address of the intended secondary.

The poll function is used by the primary device to solicit transmissions from the secondary devices. When the primary is ready to receive data, it must ask (poll) each device in turn if it has anything to send. When the first secondary is approached, it responds either with a NAK frame if it has nothing to send or with data (in the form of a data frame) if it does. If the response is negative (a NAK frame), then the primary polls the next secondary in the same manner until it finds one with data to send. When the response is positive (a data frame), the primary reads the frame and returns an acknowledgment (ACK frame), verifying its receipt.

5.9.3 TOKEN PASSING

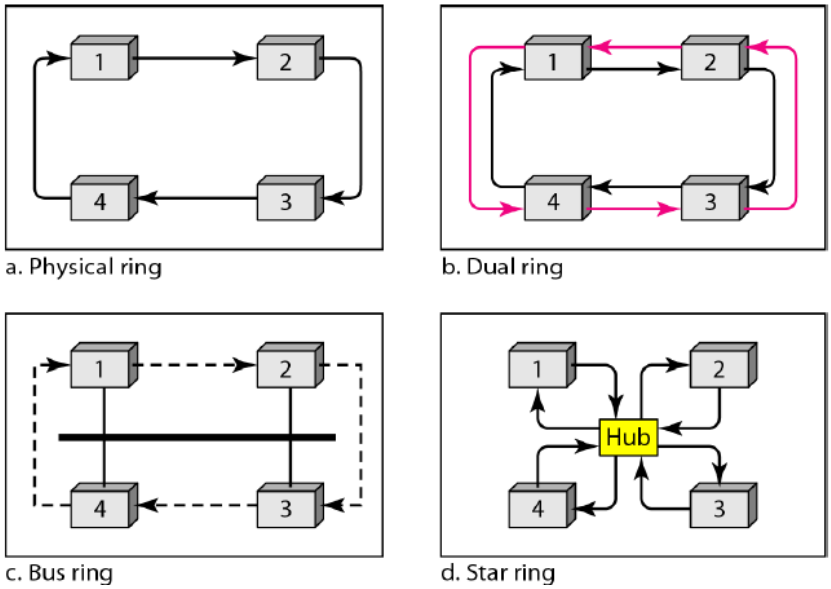
In the token-passing method, the stations in a network are organized in a logical ring. In other words, for each station, there is a predecessor and a successor. The predecessor is the station which is logically before the station in the ring; the successor is the station which is after the station in the ring. The current station is the one that is accessing the channel now. The right to this access has been passed from the predecessor to the current station. The right will be passed to the successor when the current station has no more data to send.

But how is the right to access the channel passed from one station to another? In this method, a special packet called a token circulates through the ring. The possession of the token gives the station the right to access the channel and send its data. When a station has some data to send, it waits until it receives the token from its predecessor. It then holds the token and sends its data. When the station has no more data to send, it releases the token, passing it to the next logical station in the ring. The station cannot send data until it receives the token again in the next round. In this process, when a station receives the token and has no data to send, it just passes the data to the next station.

Token management is needed for this access method. Stations must be limited in the time they can have possession of the token. The token must be monitored to ensure it has not been lost or destroyed. For example, if a station that is holding the token fails, the token will disappear from the network. Another function of token management is to assign priorities to the stations and to the types of data being transmitted. And finally, token management is needed to make low-priority stations release the token to high priority stations.

5.9.4 LOGICAL RING

In a token-passing network, stations do not have to be physically connected in a ring; the ring can be a logical one. Following figure shows four different physical topologies that can create a logical ring.



(Fig) Logical ring and physical topology in token-passing access method

In the physical ring topology, when a station sends the token to its successor, the token cannot be seen by other stations; the successor is the next one in line. This means that the token does not have to have the address of the next successor. The problem with this topology is that if one of the links-the medium between two adjacent stations fails, the whole system fails.

The dual ring topology uses a second (auxiliary) ring which operates in the reverse direction compared with the main ring. The second ring is for emergencies only (such as a spare tire for a car). If one of the links in the main ring fails, the system automatically combines the two rings to form a temporary ring. After the failed link is restored, the auxiliary ring becomes idle again. Note that for this topology to work, each station needs to have two transmitter ports and two receiver ports. The high-speed Token Ring networks called FDDI (Fiber Distributed Data Interface) and CDDI (Copper Distributed Data Interface) use this topology.

In the bus ring topology, also called a token bus, the stations are connected to a single cable called a bus. They, however, make a logical ring, because each station knows the address of its successor (and also predecessor for token management purposes). When a station has finished sending its data, it releases the token and inserts the address of its successor in the token. Only the station with the address matching the destination address of the token gets the token to access the shared media.In a star ring topology, the physical topology is a star. There is a hub, however, that acts as the connector. The wiring inside the hub makes the ring; the stations are connected to this ring through the two wire connections. This topology makes the network less prone to failure because if a link goes down, it will be bypassed by the hub and the rest of the stations can operate. Also adding and removing stations from the ring is easier.

5.10 ETHERNET (IEEE802.3)

Local area network (LAN) is a computer network that is designed for a limited geographic area such as a building or a campus. Although a LAN can be used as an isolated network to connect computers in an organization for the sole purpose of sharing resources, most LANs today are also linked to a wide area network (WAN) or the Internet.

Ethernet (the name commonly used for IEEE 802.3 CSMA/CD) is a dominant physical and data link layer technology for local area networks (LANs). It was initially designed by Bob Metcalfe in 1973, and through the efforts of Digital, Intel and Xerox (for which Metcalfe worked), "DIX" Ethernet became the standard model for LANs worldwide. Being the first network to provide Carrier Sense Multiple Access / Collision Detection (CSMA/CD), Ethernet is a fast and reliable network solution that is still widely used today.

Ethernet is a bus based broadcast network with decentralized control operating at 10 or 100 Mbps. Ethernet uses a control method called Carrier Sense Multiple Access/Collision Detection (CSMA/CD) to transmit data. Computers on an Ethernet can transmit whenever they want to; if two or more packets collide, each computer just waits a random time and tries again later. The IEEE 802.3 standard defines Ethernet protocols for (Open Systems Interconnect) OSI's Media Access Control (MAC) sub layer and physical layer network characteristics. The IEEE 802.2 standard defines protocols for the Logical Link Control (LLC) sub layer.

There are many reasons for Ethernet's success. First, Ethernet was the first widely-deployed high-speed LAN. Because it was deployed early, network administrators became intimately familiar with Ethernet and were reluctant to switch over to other LAN technologies when they came on the scene. Second, token ring, FDDI and ATM are more complex and expensive than Ethernet, which further discouraged network administrators from switching over. Third, the most compelling reason to switch to another LAN technology (such as FDDI or ATM) was usually the higher data rate of the new technology; however, Ethernet always fought back, producing versions that operated at equal data rates or higher. Switched Ethernet was also introduced in the early 1990s, which further increased its effective data rates.

The original Ethernet was created in 1976 at Xerox's Palo Alto Research Center (PARC).Since then, it has gone through different generations: Standard or Traditional Ethernet (10 Mbps), Fast Ethernet (100 Mbps) and Gigabit Ethernet (l Gbps).

5.10.1. TRADITIONAL ETHERNET

The traditional Ethernet is the oldest version of Ethernet created in 1976 which is designed to operate at 10Mbps.

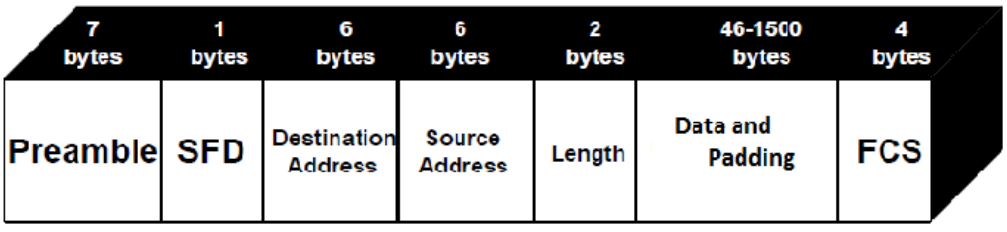
MAC sub layer

The MAC Sub layer governs the operations of the access method. It receives frames from the upper layer and passes them to the physical layer. Ethernet uses Carrier Sense Multiple Access/Collision Detection (CSMA/CD) when transmitting data. Carrier Sense allows a computer device to "sense" whether or not another transmission is being "carried" over the network. So, before a device sends data, it listens for a carrier (jam) signal. If a carrier signal is detected, it waits until that transmission is completed. Multiple Access means that all devices have equal access to the network. Since Ethernet is contention-based, equal access to the network for all is ensured. No device has priority over others, nor can it lock out any other device connected to the network. Information can be transmitted at any time by any device. Collision Detection means that a sending device can "detect" simultaneous transmission attempts. When two or more devices try to send data at the same time, the signals collide. When this happens, each device then transmits a jam signal, called a carrier, to alert all other devices that a collision has occurred. All devices then go into back off mode and wait a random amount of time before attempting to retransmit data. The random time provision prevents simultaneous retransmissions.

To better understand CSMA/CD, think about trying to make a telephone call. Many of us have more than one telephone in our homes (a telephone network). When you pick up the telephone to make a call, you "sense" a dial tone or someone else on the line. If there is a dial tone, you proceed with your call. If the telephone line is currently in use, you cannot make a call at this time and you try again later. This is similar to Ethernet Carrier Sense protocols.

All telephones in the house can be used at any time to make calls. All phones in the house have equal access to the telephone network. This is comparable to Multiple Access. Should two individuals in the house attempt to make a phone call simultaneously, both hear dial tone; neither party senses a carrier, (someone else on the line). However, like Ethernet technology, only one individual can use the line at any one time. Both parties must hang-up and wait a random amount of time before making a second attempt. This is how Ethernet's Collision Detection works.

Frame Format



(Fig:2.24) MAC Frame Format

Preamble: The first field of the 802.3 frame contains 7 bytes (56 bits) that repeats the 8-bit pattern 10101010 7 times. This patterns that alerts the receiving system about the incoming frame and enables it to synchronize its input timing.

Start frame delimiter (SFD): The second field (I byte: 10101011) signals the beginning of the frame. The SFD warns the station or stations that this is the last chance for synchronization. The last 2 bits is 11 and alerts the receiver that the next field is the destination address.

Destination address (DA): The DA field is 6 bytes and contains the physical address of the destination station or stations to receive the packet.

Source address (SA): The SA field is also 6 bytes and contains the physical address of the sender of the packet

Length or type: This field is defined as a type field or length field. The original Ethernet used this field as the type field to define the upper-layer protocol using the MAC frame. The IEEE standard used it as the length field to define the number of bytes in the data field. Both uses are common today.

Data: This field carries data encapsulated from the upper-layer protocols. It is a minimum of 46 and a maximum of 1500 bytes

FCS: This sequence contains a 4-byte cyclic redundancy check (CRC) value, which is created by the sending device

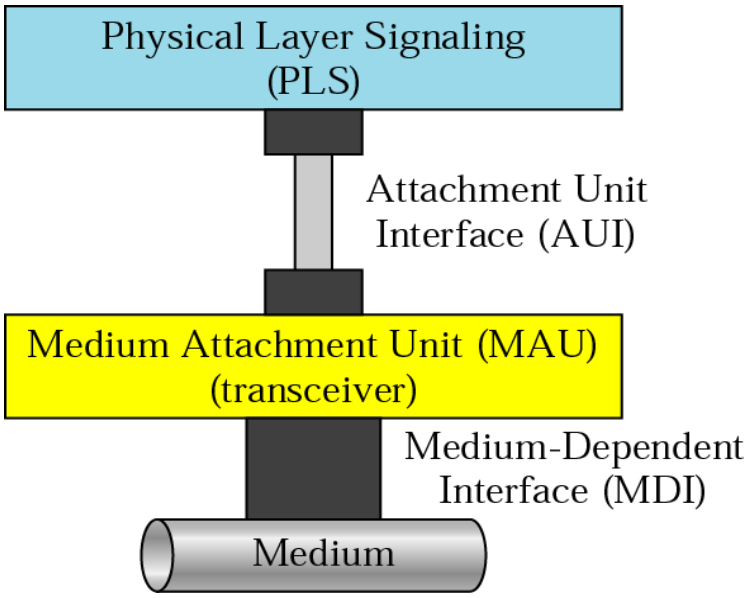
Addressing

There can be various types of stations connected on an Ethernet network such as a PC or Workstation or Printer. Each station on an Ethernet network (such as a PC, workstation, or printer) has its own network interface card (NIC). The NIC fits inside the station and provides the station with a 6-byte physical address. An example for Ethernet Address is: **06:01:02:01:2C:4B.**

A source address is always a unicast address-the frame comes from only one station. The destination address, however, can be unicast, multicast, or broadcast. If the least significant bit of the first byte in a destination address is 0, the address is unicast; otherwise, it is multicast. A unicast destination address defines only one recipient; the relationship between the sender and the receiver is one-to-one. A multicast destination address defines a group of addresses; the relationship between the sender and the receivers is one-to-many. The broadcast address is a special case of the multicast address; the recipients are all the stations on the LAN.

5.10.1.1. Physical Layer

The physical layer of Traditional Ethernet is made up of four sub layers: The physical layer signaling (PLS) sub layer, The attachment unit interface (AUI) sub layer, The medium attachment unit (MAU) sub layer, and The medium-dependent interface (MDI) sub layer. The PLS sub layer is common for all implementations. The AUI layer may or may not be present in some implementations. The MAU and MDI are specific for each medium type.



(Fig:) Traditional Ethernet: Physical Layer

Physical Layer Signaling (PLS): The PLS sub layer is used for encoding and decoding data. The type of encoding used in traditional Ethernet is the Manchester Encoding and the data rate is 10Mbps.The data from MAC sub layer is encoded using the Manchester Encoder. The encoded data is passed on to the destination. Here, the data is decoded by the Manchester decoder and applied to the MAC sub layer.

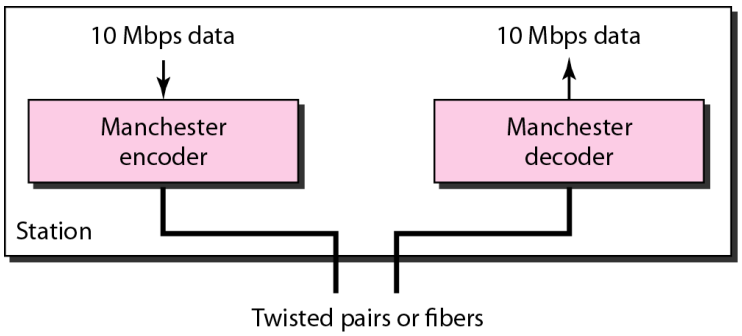
Attachment Unit Interface (AUI): It is a specification which is used for defining interface between PLS and MAU. The interface was designed for the first implementation of Ethernet, which used thick coaxial cable.

Medium Attachment Unit (Transceiver): The Medium Attachment Unit or transceiver is medium dependent. It creates appropriate signals for each type of medium. Different types of medium used in traditional Ethernet are coaxial cable, twisted pair cable and fiber optic cable. There is an MAU for each type of medium. The transceiver is a combination of transmitter and receiver. It performs the task of transmission as well as reception. It transmits signal over the medium; it receives signal over the medium and it also detects the collision. A transceiver can be internal or external. An external transceiver is installed close to the media and is connected via a AUI interface to the station. An internal transceiver is installed inside the station and does not need an AUI cable.

Medium Dependent Interface (MDI): To connect a transceiver to the medium, we need a Medium Dependent Interface (MDI).The MDI is just a piece of hardware for connecting a transceiver to the medium.

Encoding and Decoding

All standard implementations use digital signaling (baseband) at 10 Mbps. At the sender, data are converted to a digital signal using the Manchester scheme; at the receiver, the received signal is interpreted as Manchester and decoded into data.



(Fig:2.26) Encoding in a Standard Ethernet implementation

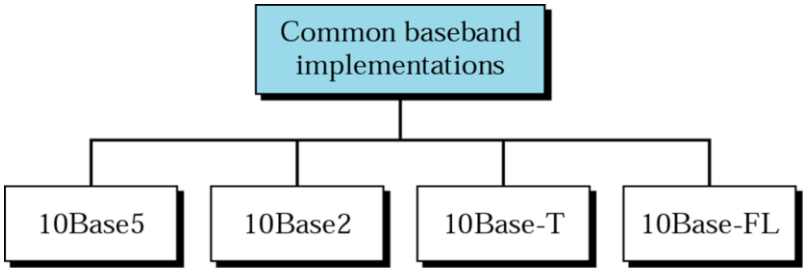
PHYSICAL LAYER IMPLEMENTATION

Ethernet is implemented in a variety of ways at the physical layer level. The choice is in terms of

- **Physical Topology**
- **Transmission media**
- **Bit Rate**

The IEEE standard has defined four implementations for Traditional Ethernet. To distinguish the various implementations that are available, the committee has developed a concise notation:

<data rate in Mbps><signaling method><maximum segment length in hundreds of meters>

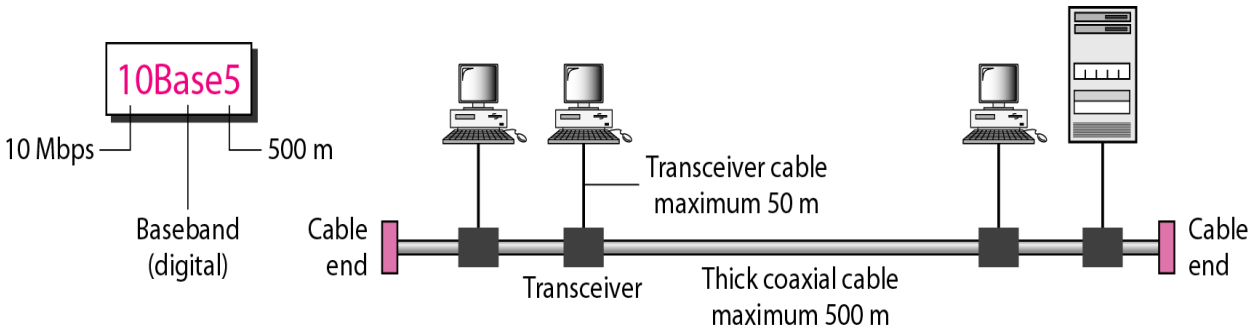


(Fig:2.27) Categories of Standard Ethernet

10Base5: Thick Ethernet

The first implementation is called 10Base5, thick Ethernet, or Thick net. The nickname derives from the size of the cable, which is roughly the size of a garden hose and too stiff to bend with your hands. 10Base5 was the first Ethernet specification to use a bus topology with an external transceiver (transmitter/receiver) connected via a tap to a thick coaxial cable. The term base is an abbreviation for base band signals using a Manchester Encoding. The 10 of 10Base5 represents a 10-Mbps transmission speed and 5 represents 500 meters maximum cable segment length.

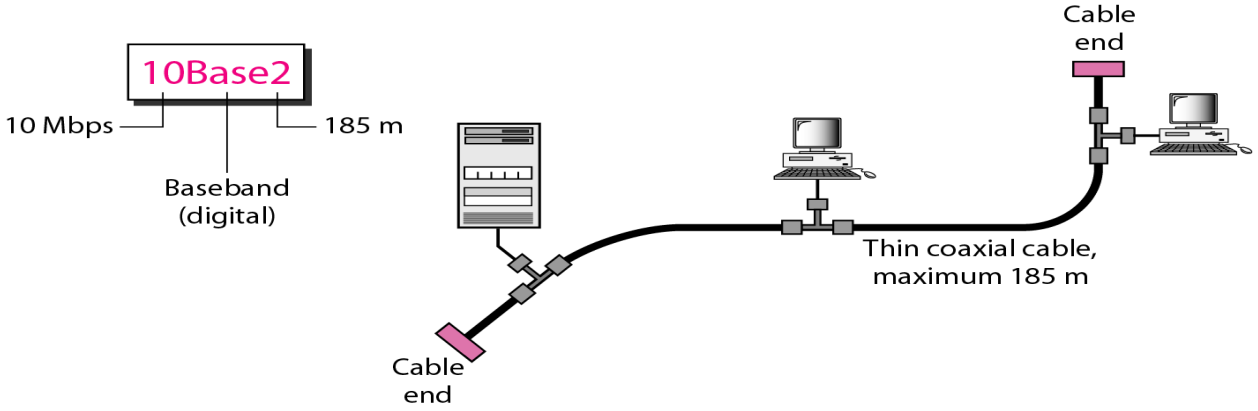
The transceiver is responsible for transmitting, receiving, and detecting collisions .The transceiver connects to the computer interface through cable up to 50 metres long via a connector called an attachment unit interface or AUI.



(Fig:2.28) IOBase5 implementation

10Base2: Thin Ethernet

I0Base2 also uses a bus topology, but the cable is much thinner and more flexible. It was designed to allow for a less expensive network by using less-expensive components. The cable can be bent to pass very close to the stations. In this case, the transceiver is normally part of the network interface card (NIC), which is installed inside the station. The 10Base2 network can transmit 10 Mbps digital signals over coaxial cable. The length of each segment cannot exceed 185 m (close to 200 m) due to the high level of attenuation in thin coaxial cable.

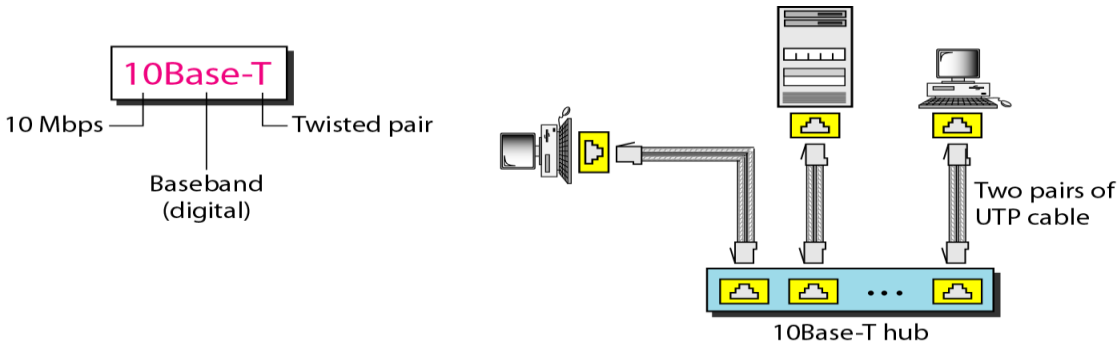


(Fig:) IOBase2 implementation

Note that the collision here occurs in the thin coaxial cable. This implementation is more cost effective than 10Base5 because thin coaxial cable is less expensive than thick coaxial and the tee connections are much cheaper than taps. Installation is simpler because the thin coaxial cable is very flexible.

10Base-T: Twisted-Pair Ethernet

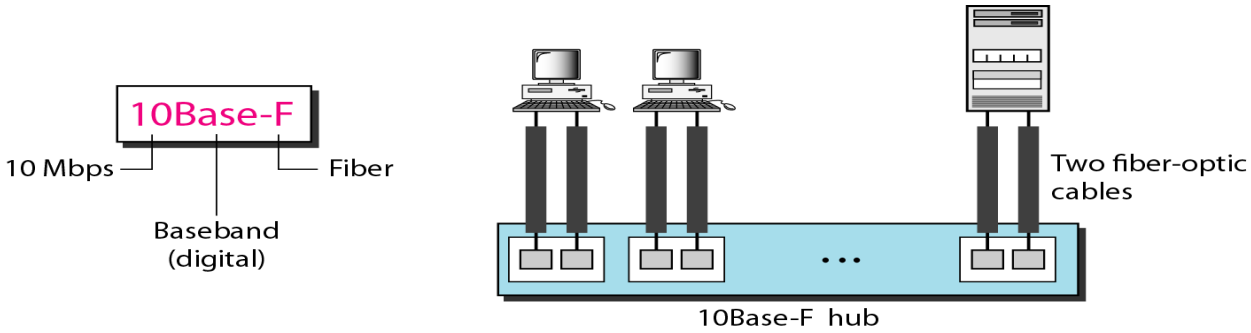
10Base-T uses a physical star topology. The stations are connected to a hub via two pairs of twisted cable. One pair of wires is used for transmitting data, and the other for receiving data. Any collision here happens in the hub. All 10Base-T connections are point-to-point. This means that a 10Base-T cable can only have two transceivers connected, one at each end. One end of the cable is typically attached to a hub, and the other directly to a computer or network device. Compared to l0Base5 or l0Base2, we can see that the hub actually replaces the coaxial cable as far as a collision is concerned. The maximum length of the twisted cable here is defined as 100 m, to minimize the effect of attenuation in the twisted cable.



(Fig:) IOBase-T implementation

10Base-F: Fiber Ethernet

Although there are several types of optical fiber l0-Mbps Ethernet, the most common is called 10Base-F.l0Base-F uses a star topology to connect stations to a hub. The stations are connected to the hub using two fiber-optic cables.



(Fig:) IOBase-F implementation

5.10.2 CHANGES IN THE STANDARD

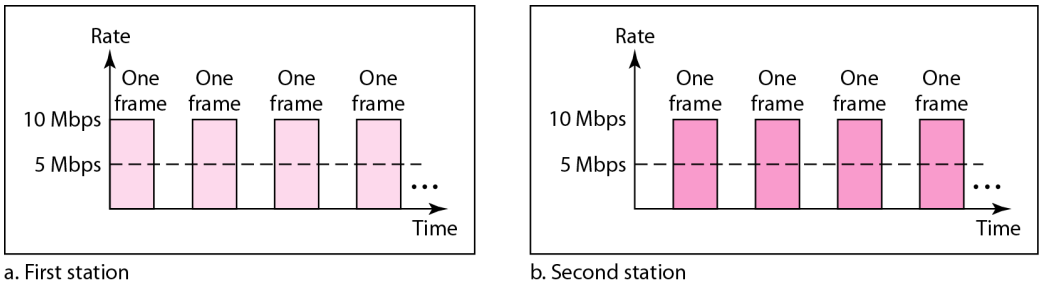
The 10-Mbps Standard Ethernet has gone through several changes before moving to the higher data rates. These changes actually opened the road to the evolution of the Ethernet to become compatible with other high-data-rate LANs.

Bridged Ethernet

The first step in the Ethernet evolution was the division of a LAN by bridges. Bridges have two effects on an Ethernet LAN: They raise the bandwidth and they separate collision domains.

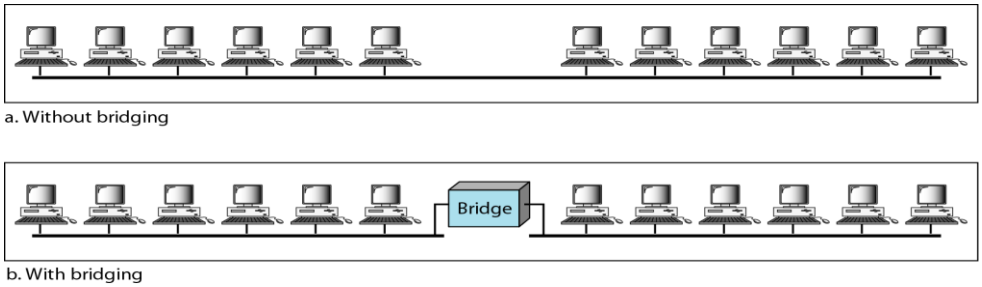
Raising the Bandwidth

In an unbridged Ethernet network, the total capacity (10 Mbps) is shared among all stations with a frame to send; the stations share the bandwidth of the network. If only one station has frames to send, it benefits from the total capacity (10 Mbps). But if more than one station needs to use the network, the capacity is shared. For example, if two stations have a lot of frames to send, they probably alternate in usage. When one station is sending, the other one refrains from sending. We can say that, in this case, each station on average sends at a rate of 5 Mbps.



(Fig) Sharing bandwidth

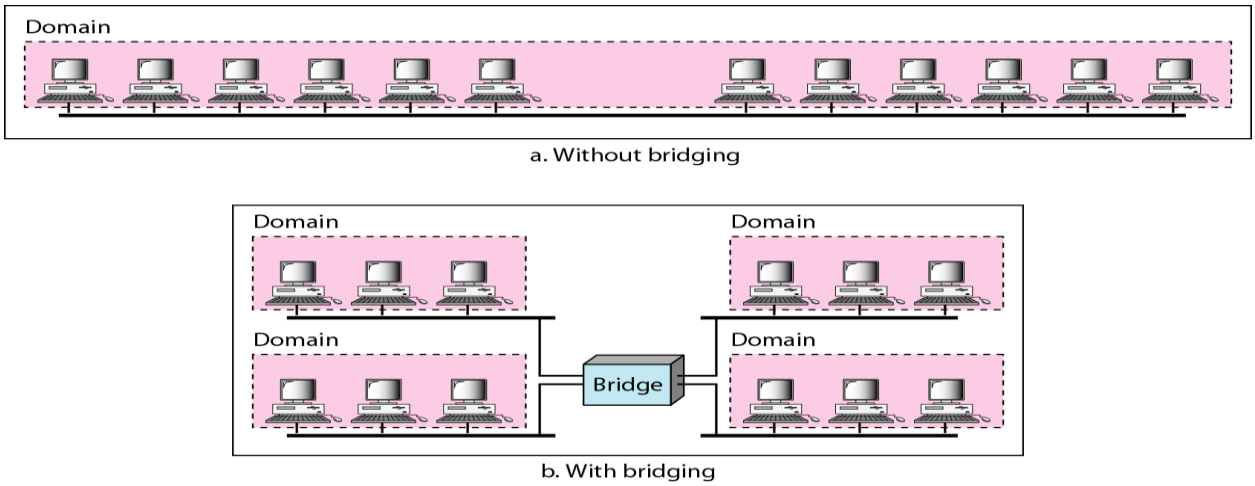
A bridge is used to divide the network into two or more networks. Bandwidth-wise, each network is independent. For example, a network with 12 stations is divided into two networks, each with 6 stations. Now each network has a capacity of 10 Mbps. The 10Mbps capacity in each segment is now shared between 6 stations (actually 7 because the bridge acts as a station in each segment), not 12 stations. In a network with a heavy load, each station theoretically is offered 10/6 Mbps instead of 10/12 Mbps.



(Fig:) A network with and without a bridge

Separating Collision Domains

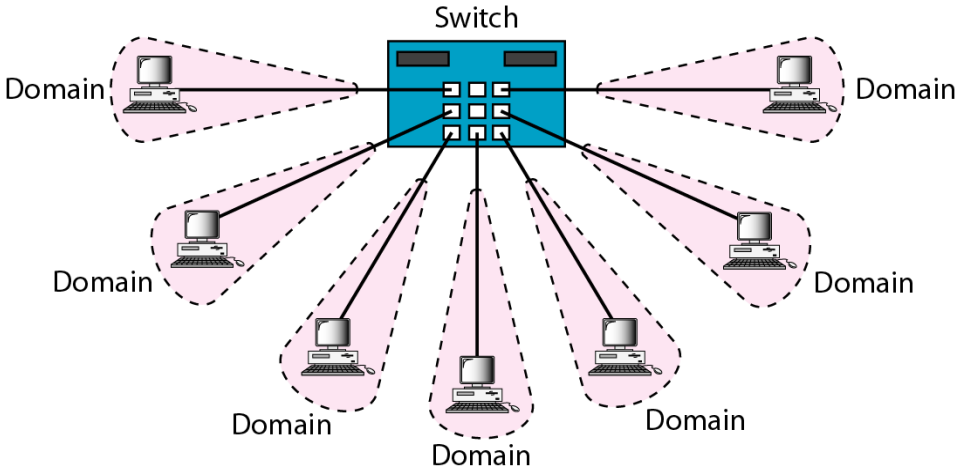
Another advantage of a bridge is the separation of the collision domain. The collision domain becomes much smaller and the probability of collision is reduced tremendously. As shown in the following figure, without bridging, 12 stations contend for access to the medium; with bridging only 3 stations contend for access to the medium.



(Fig:) Collision domains in an unbridged network and a bridged network

Switched Ethernet

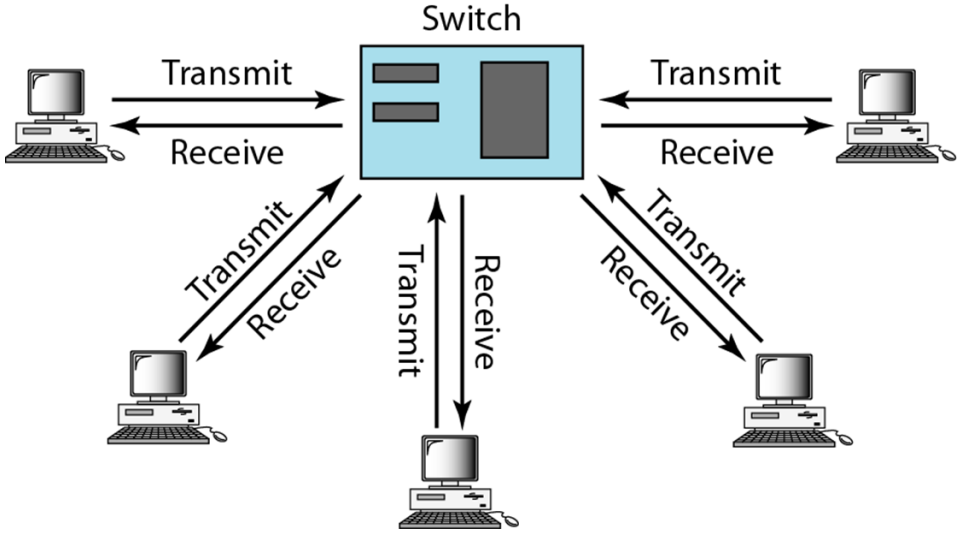
The idea of a bridged LAN can be extended to a switched LAN. Instead of having two to four networks, why not have N networks, where N is the number of stations on the LAN. In other words, if we can have a multiple-port bridge, why not have an N-port switch? In this way, the bandwidth is shared only between the station and the switch (5 Mbps each). In addition, the collision domain is divided into N domains. A layer 2 switch is an N-port bridge with additional sophistication that allows faster handling of the packets. Evolution from a bridged Ethernet to a switched Ethernet was a big step that opened the way to an even faster Ethernet.



(Fig:) Switched Ethernet

Full-Duplex Ethernet

One of the limitations of 10Base5 and l0Base2 is that communication is half-duplex (l0Base-T is always full-duplex); a station can either send or receive, but may not do both at the same time. The next step in the evolution was to move from switched Ethernet to full-duplex switched Ethernet. The full-duplex mode increases the capacity of each domain from 10 to 20 Mbps. Note that instead of using one link between the station and the switch, the configuration uses two links: one to transmit and one to receive.



(Fig:) Full-duplex switched Ethernet

In full-duplex switched Ethernet, there is no need for the CSMA/CD method. In a full duplex switched Ethernet, each station is connected to the switch via two separate links. Each station or switch can send and receive independently without worrying about collision. Each link is a point-to-point dedicated path between the station and the switch. There is no longer a need for carrier sensing; there is no longer a need for collision detection. The job of the MAC layer becomes much easier. The carrier sensing and collision detection functionalities of the MAC sub layer can be turned off.

FAST ETHERNET

Although Ethernet is the most popular method of linking computers on a network, its 10 Mbps speed is too slow for very data intensive or real-time applications. Fast Ethernet refers to a set of specifications developed by IEEE 802.3 committee to provide a low-cost Ethernet-compatible LAN. The fast Ethernet uses the same frame format, same CSMA/CD protocol and same interface as the 802.3, but uses a data transfer rate of 100 Mb/s instead of 10 Mb/s. However, fast Ethernet is based entirely on 10-BaseT, because of its advantages. The IEEE officially adopted the new IEEE 802.3u Fast Ethernet/100BASE-T specification in May 1995

MAC Sublayer

A main consideration in the evolution of Ethernet from 10 to 100 Mbps was to keep the MAC sub layer untouched. The Ethernet is designed in such a way that the speed can be increased if collision domain is decreased. The only two changes made in the MAC layer are the data rate and the collision domain. The data rate is increased by a factor of 10 and collision domain is decreased by a factor of 10.

A decision was made to drop the bus topologies and keep only the star topology. For the star topology, there are two choices, as we saw before: half duplex and full duplex. Half duplex means that a station is either transmitting or receiving, but not both, at the same time. That's because in CSMA/CD, a station has to listen to see whether the channel is available, and only if it is can a station start transmitting. When one station is transmitting data, all others have to listen. This method of operation was very efficient for the coaxial cable. Running a single coaxial cable throughout an entire office and providing everyone an access opportunity to the cable every few milliseconds was very efficient. Full-duplex, on the other hand, means that a station is simultaneously transmitting and receiving. The introduction of 10BASE-Twiring offered the capability for separate transmit and receive data paths. In Full Duplex, there is no need for CSMA/CD, because a cable pair would now be dedicated for both transmission and reception. However, the implementations keep CSMA/CD for backward compatibility with Standard Ethernet.

Auto Negotiation

A new feature added to Fast Ethernet is called auto negotiation. It allows a station or a hub a range of capabilities. Auto negotiation allows two devices to negotiate the mode or data rate of operation. It was designed particularly for the following purposes:

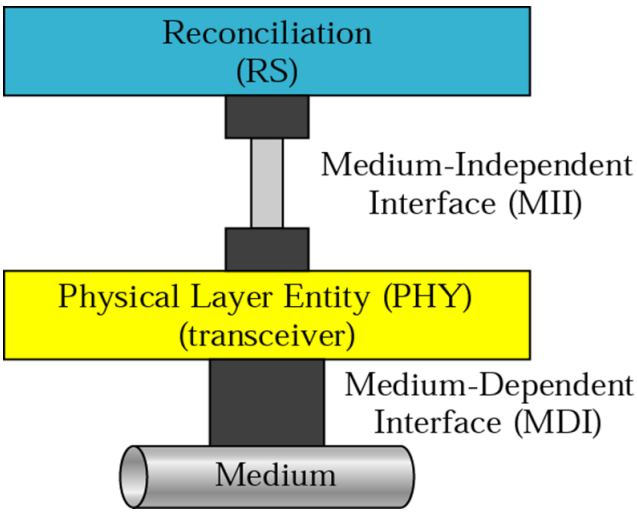
- To allow incompatible devices to connect to one another. For example, a device with a maximum capacity of 10 Mbps can communicate with a device with a 100 Mbps capacity (but can work at a lower rate).
- To allow one device to have multiple capabilities.
- To allow a station to check a hub's capabilities.

Physical Layer

The physical layer in Fast Ethernet is more complicated than the one in Standard Ethernet. The following figure shows the physical layer for Fast Ethernet

Reconciliation

The reconciliation sub layer in Fast Ethernet replaces the PLS sub layer in traditional Ethernet. It handles all the PLS sub layer tasks except for encoding and decoding and is responsible for the passing of data in 4-bit format to the MII.



(Fig:) Physical Layer: Fast Ethernet

Medium Independent Interface(MII)

In the design of Fast Ethernet, The AUI was replaced with the medium independent interface (MII).The MII is an improved interface that can be used with both a 10 and 100 Mbps data rate. It features a parallel data path (4 bits at a time) between the PHY sub layer and reconciliation sub layer.

PHY (transceiver)

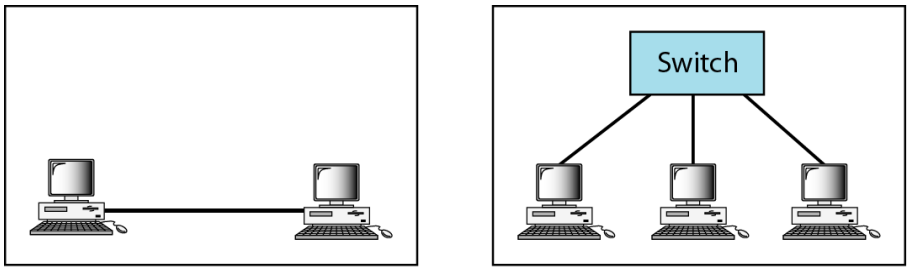
The transceiver in Fast Ethernet is called the PHY sub layer. Besides the regular functions mentioned in 10Mbps Ethernet, the transceiver in Fast Ethernet is also responsible for encoding and decoding. This function was moved from the PLS layer to the PHY sub layer. A transceiver can be external or internal. An external transceiver is installed close to the medium and is connected via an MII cable to the station. An internal transceiver is installed inside the station and does not need an MII cable.

Medium Dependent Interface (MDI)

To connect the transceiver to the medium, we need a medium dependent interface. The MDI is just a piece of hardware that is implementation specific.

Topology

Fast Ethernet is designed to connect two or more stations together. If there are only two stations, they can be connected point-to-point. Three or more stations need to be connected in a star topology with a hub or a switch at the center

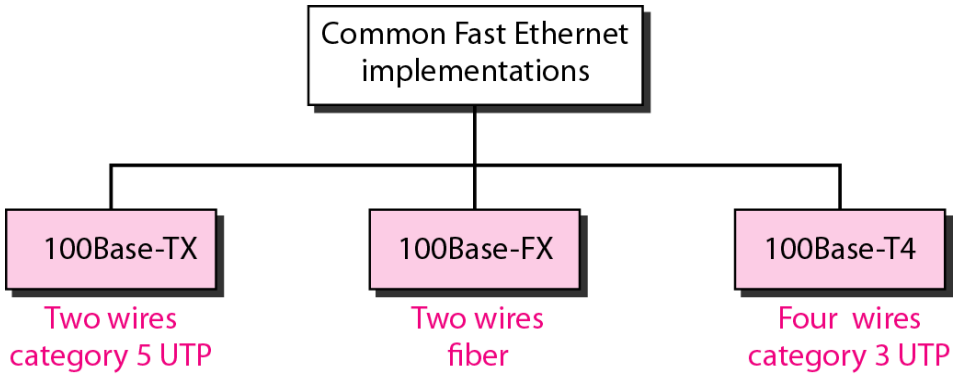


(Fig:2.38) Fast Ethernet Topology

IMPLEMENTATION

Fast Ethernet provides 3 versions depending upon the physical media

- 100Base-TX which uses 2 pairs of Category 5 UTP
- 100Base-FX which uses 4 pairs of wires Category 3,4 or 5 UTP
- 100Base-T4 which uses multi-mode or single-mode fiber optic cable



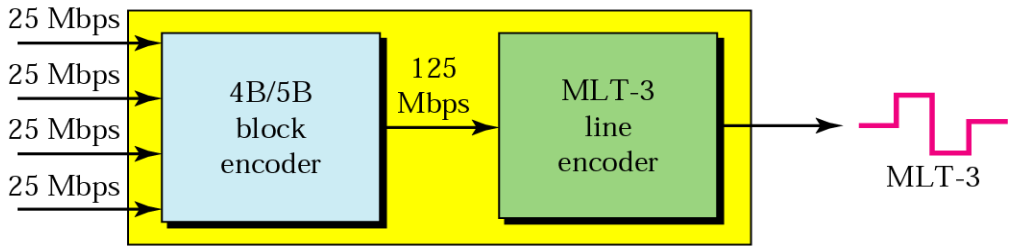
(Fig:) Fast Ethernet implementations

100Base-TX

100Base-TX uses two pairs of twisted-pair cable (either category 5 UTP or STP) in a physical star topology. The implementation allows either an external transceiver or an internal transceiver. The transceiver is responsible for transmitting, sending, detecting collisions, and encoding/decoding of data.

Encoding and Decoding

To achieve a 100Mbps data rate, encoding (and decoding) is implemented in two steps as show in figure. To maintain synchronization, the encoder first performs block encoding. The four parallel bits received from the NIC are encoded into 5 serial bits using 4B/5B encoder. This creates a data rate of 125 Mbps, which is fed into MLT-3 for encoding. It uses three levels of signals (+1,0 and -1).The signal transitions from one level to the next happens at the beginning of a 1 bit; there is no transition at the beginning of a 0 bit.

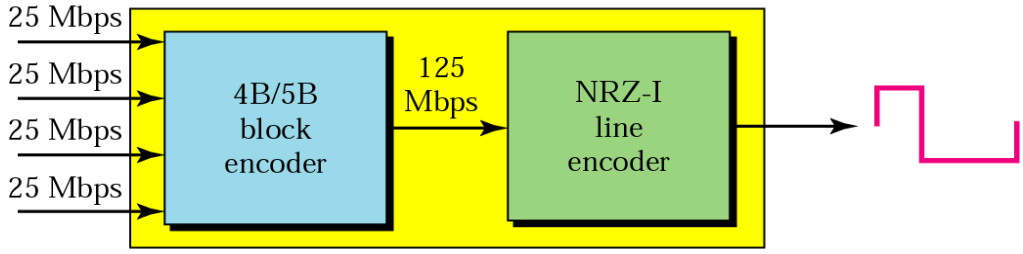


(Fig:) 100Base-TX Encoding and Decoding

100Base-FX

100Base-FX uses two pairs of fiber-optic cables in a physical star topology. The implementation allows either an external transceiver or an internal transceiver which is responsible for transmitting, sending, detecting the collision and encoding/decoding.

Encoding and Decoding



(Fig:) 100Base-TX Encoding and Decoding

100Base-FX uses two levels of encoding as shown in figure. To maintain synchronization, the encoder first performs block encoding. The four parallel bits received from the NIC are encoded into 5 serial bits using 4B/5B encoder. The data at the 125 Mbps rate is then encoded into a signal using the NRZ-I encoding scheme which transitions on bit 1.

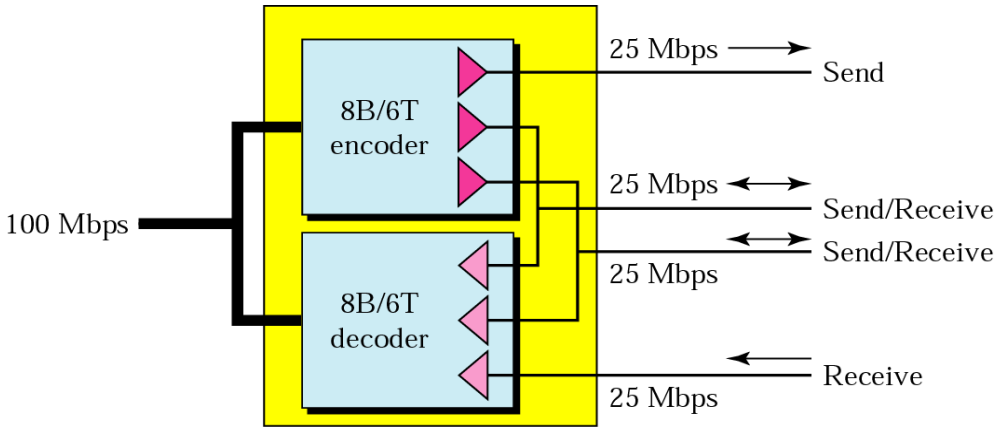
100Base-T4

A 100Base-TX network can provide a data rate of 100mbps,but it requires the use of category-5 UTP or STP cable. A new standard, called 100Base-T4 was designed to use category 3 or higher UTP. The implementation uses four pairs of UTP for transmitting 100Mbps.While the transceiver function in 100Base-T4 is similar to the other implementations, encoding and decoding is more complex.

Encoding and Decoding

To maintain synchronization and at the same time reduce the bandwidth, a three level line encoding method called 8B/6T (eight binary/six ternary) is used. This means that each block of 8-bit data is encoded as units of ternary signals(using 3 levels,+1,0 and -1).

As this implementation uses category 3 UTP, each twisted-pair cannot easily handle more than 25 Mbaud. In this design, one pair switches between sending and receiving. Three pairs of UTP category 3, however, can handle only 75 Mbaud (25 Mbaud) each. We need to use an encoding scheme that converts 100 Mbps to a 75 Mbaud signal.8B/6T satisfies this requirement. In 8B/6T, eight data elements are encoded as six signal elements. This means that 100 Mbps uses only (6/8) x 100 Mbps, or 75 Mbaud.



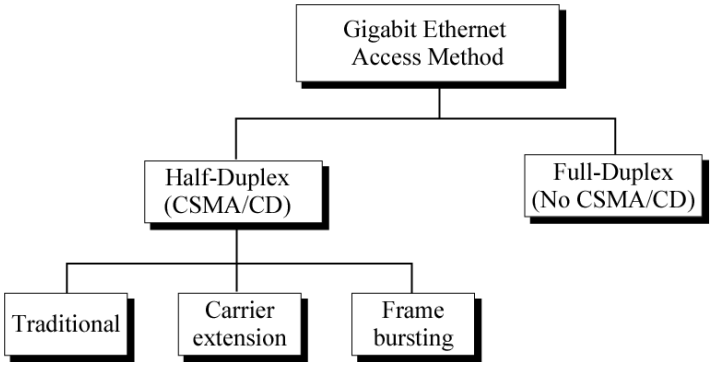
(Fig:) 100Base-T4 Encoding and Decoding

5.11. GIGABIT ETHERNET

The need for an even higher data rate resulted in the design of the Gigabit Ethernet (1000 Mbps).The IEEE 802.3z Gigabit Ethernet Standard was released in 1998. While defining a new medium and transmission specification, Gigabit Ethernet retains the CSMA/CD protocol and Ethernet format of its 10-Mbps and 100-Mbps predecessors. It is compatible with 100BASE-T and 10BASE-T, preserving a smooth migration path.

MAC Sub layer

A main consideration in the evolution of Ethernet was to keep the MAC sub layer untouched. However, to achieve a data rate 1 Gbps, this was no longer possible. Gigabit Ethernet uses the official 802.3 frame format, identical to that of 10Mbps and 100Mbps Ethernet. Gigabit Ethernet has two distinctive approaches for medium access: half-duplex and full-duplex.



(Fig:) Gigabit Ethernet Access methods

Full Duplex MAC

In full-duplex mode, there is a central switch connected to all computers or other switches. In this mode, each switch has buffers for each input port in which data are stored until they are transmitted. There is no collision in this mode and so CSMA/CD is not used.

Half Duplex MAC

Gigabit Ethernet can also be used in half-duplex mode, although it is rare. In this case, a switch can be replaced by a hub, which acts as the common cable in which a collision might occur. The half-duplex approach uses CSMA/CD. However, as we saw before, the maximum length of the network in this approach is totally dependent on the minimum frame size. Three methods have been defined: traditional, carrier extension, and frame bursting.

Traditional

In the traditional approach, we keep the minimum length of the frame as in traditional Ethernet (512 bits). However, because the length of a bit is 1/100 shorter in Gigabit Ethernet than in l0-Mbps Ethernet, the slot time (time taken for a station to send 512 bits) for Gigabit Ethernet is 512 bits x 1/1000μs, which is equal to 0.512μs. The reduced slot time means that collision is detected 100 times earlier.

There is a relation between slot time and maximum length of the network. In standard Ethernet, the maximum length of network is set to 2500.Here the slot time is reduced by a factor of 100 as the data transmission rate is 100 times faster. This means that the maximum length of the network is 25 m. This length may be suitable if all the stations are in one room, but it may not even be long enough to connect the computers in one single office.

Carrier Extension

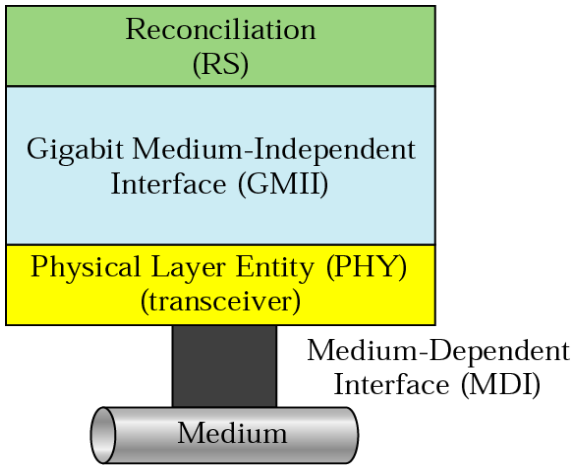
To allow for a longer network, we increase the minimum frame length. The carrier extension approach defines the minimum length of a frame as 512 bytes (4096 bits). This means that the minimum length is 8 times longer. This method forces a station to add extension bits (padding) to any frame that is less than 4096 bits. In this way, the maximum length of the network can be increased 8 times to a length of 200 m. This allows a length of 100 m from the hub to the station

Frame Bursting

Carrier extension is very inefficient if we have a series of short frames to send; each frame carries redundant data. To improve efficiency, frame bursting was proposed. Instead of adding an extension to each frame, multiple frames are sent. However, to make these multiple frames look like one frame, padding is added between the frames (the same as that used for the carrier extension method) so that the channel is not idle. In other words, the method deceives other stations into thinking that a very large frame has been transmitted.

Physical Layer

The physical layer is made up of four sub layers: Reconciliation, GMII,PHY and MDI. The reconciliation sub layer is common to all implementations. The PHY and MDI are medium dependent



(Fig:) Gigabit Ethernet: Physical Layer

Reconciliation

Reconciliation sub layer has the Responsibility for passing 8-bit parallel data to PHY sub layer via GMII interface

Gigabit Medium Independent Interface (GMII)

It is a specification that defines how the reconciliation sub layer is to be connected to the PHY sub layer. It is the counter part of MII in Fast Ethernet. It is primarily a logical rather than a physical interface

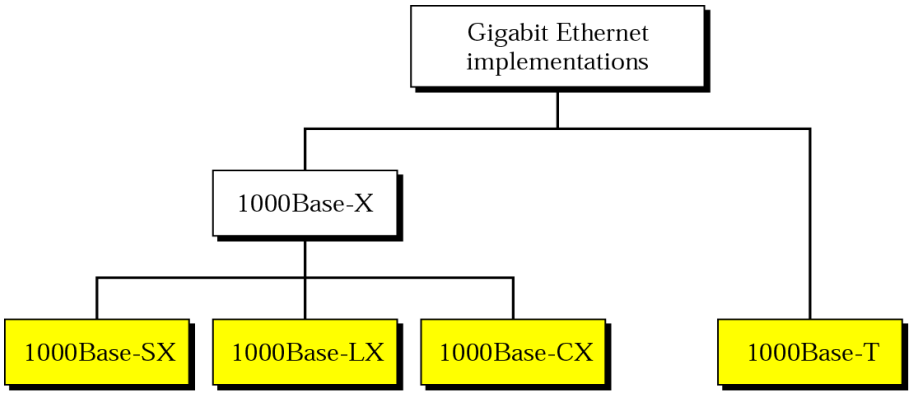
PHY(Transceiver)

Just as in Fast Ethernet, transceiver is medium dependent and also encodes and decodes. In gigabit Ethernet the transceiver can only be internal as there is no external GMII designed to provide connection

MDI: Just as in Fast Ethernet, the MDI connects the transceiver to the medium.

GIGABIT ETHERNET IMPLEMENTATION

Gigabit Ethernet can be categorized as either a two-wire or four-wire implementation. The two wire implementation is called 1000Base-X, which can use short-wave optical fiber (1000Base-SX),long wave optical Fiber(1000Base-LX) or copper wires(1000Base-CX).The four wire version uses twisted-pair cable(1000Base-T)



(Fig:) Gigabit Ethernet implementations

1000Base-X

Both 1000Base-SX and 1000Base-LX use two fiber- optic cables. The only difference is that former uses short-wave laser and the latter uses long-wave laser. All implementations are designed with an internal transceiver, so there is no external GMII cable or connector.1000Base-CX was designed to use STP cable, but it has never been implemented.

The transceiver in Giga-Bit Ethernet is internal. It functions the encoding/decoding, transmitting, receiving and collision detection. To achieve 1000Mbps data rate encoding and decoding are done in two steps:To maintain synchronization encoder first performs block encoding. The 8 parallel bits received are encoded into 10 serial bits using 8B/10B block encoder. The data obtained at 1.25Gps rate are encoded into a signal using NRZ encoding. In this implementation, one wire is used for sending and one for receiving.

1000Base-T

The 1000BASE -T employs full duplex transmission of 1000 Mbps over four pairs of Category 5 cable. The data rate of 1000 Mbps is achieved by transmission of 250 Mbps over each wire pair. The encoding scheme used is 4D-PAM5.It translates 8-bit data into a simultaneous transmission of four code symbols (4D), which are sent over the media, one on each pair

5.12.LAN CONNECTING DEVICES

Two or more devices connected for the purpose of sharing data and resources can form a network. Putting together a network is more complicated than simply plugging cable into a hub. A local area network may need to cover more distance than its media can handle effectively. Or the number of stations involved may be too great for efficient frame delivery or management of the network, and the network need to be sub divided. In the first case a device called a repeater or regenerator is inserted into the network to increase coverable distance. In the second, a device called a bridge is inserted for traffic management.

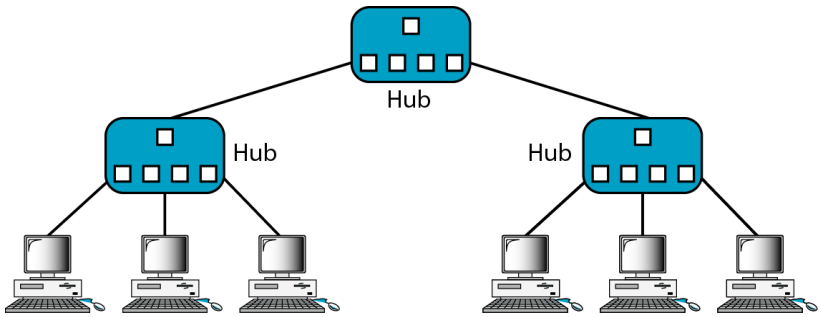
We discuss two kinds of connecting devices: repeaters, hubs and bridges (or two-layer switches), Repeaters and hubs operate in the first layer of the Internet model. Bridges and two-layer switches operate in the first two layers.

PASSIVE HUB

A passive hub is just a connector. It connects the wires coming from different branches. In a star-topology Ethernet LAN, a passive hub is just a point where the signals coming from different stations collide; the hub is the collision point. This type of a hub is part of the media; its location in the Internet model is below the physical layer.

ACTIVE HUB

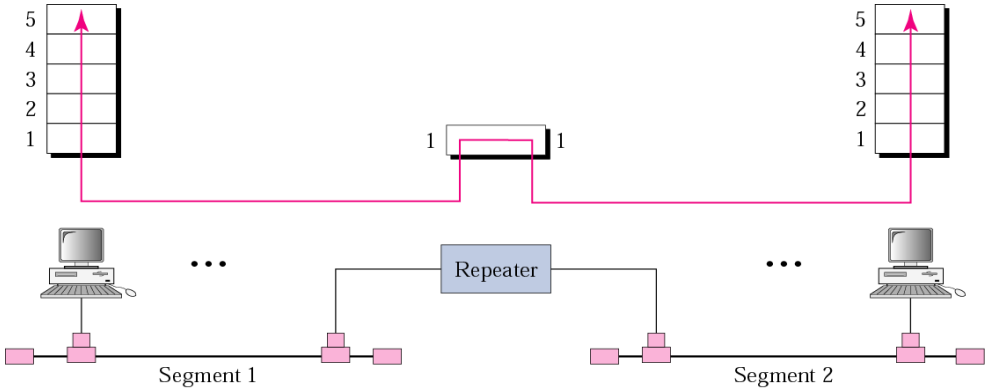
An active hub is actually a multipart repeater. It is normally used to create connections between stations in a physical star topology. We have seen examples of hubs in some Ethernet implementations (l0Base-T, for example). However, hubs can also be used to create multiple levels of hierarchy. The hierarchical use of hubs removes the length limitation of 10Base-T (100 m).



(Fig:) A hierarchy of hubs

REPEATERS

A repeater is a device that operates only in the physical layer. Signals that carry information within a network can travel a fixed distance before attenuation endangers the integrity of the data. A repeater receives a signal and, before it becomes too weak or corrupted, regenerates the original bit pattern. The repeater then sends the refreshed signal. A repeater can extend the physical length of a LAN



(Fig:) A repeater connecting two segments of a LAN

A repeater does not actually connect two LANs; it connects two segments of the same LAN. The segments connected are still part of one single LAN. A repeater is not a device that can connect two LANs of different protocols. A repeater can overcome the 10Base5 Ethernet length restriction. In this standard, the length of the cable is limited to 500 m.

To extend this length, we divide the cable into segments and install repeaters between segments. Note that the whole network is still considered one LAN, but the portions of the network separated by repeaters are
called segments.

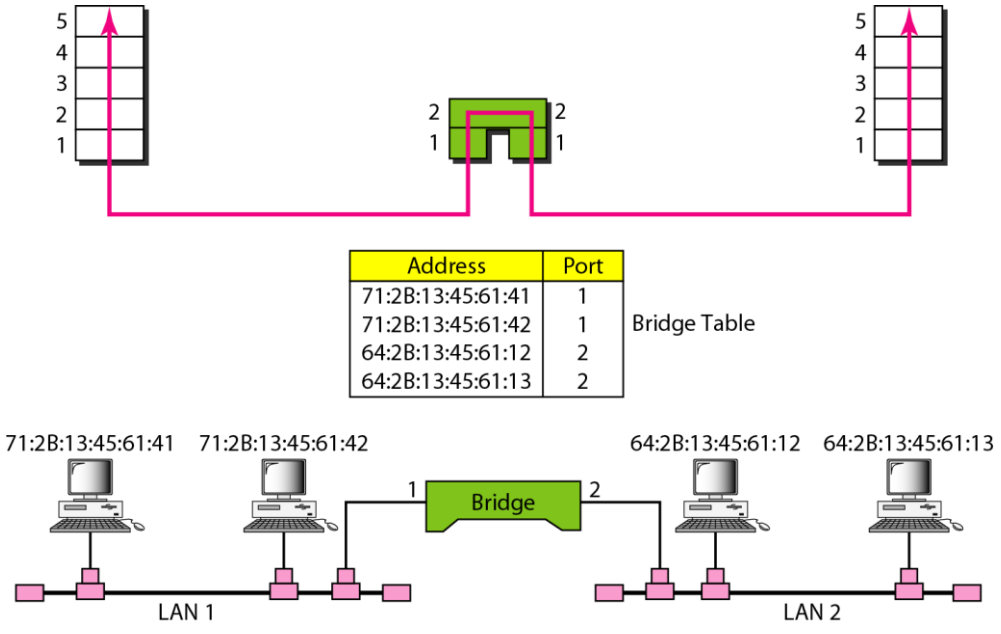
A repeater has no filtering capability. Every frame received will be regenerated (not amplified) and forwarded. The location of a repeater on a link is vital. A repeater must be placed so that a signal reaches it before any noise changes the meaning of any of its bits.

BRIDGES

Bridges are important in some network because the networks are geographically divided into many parts.It can be used in LAN to connect different segments of a LAN. A bridge operates in both the physical and the data link layer .It divides a large network into small segments As a physical layer device, it regenerates the signal it receives. Bridges are repeaters that are smart enough to relay a frame only to the side of the segment containing the intended recipient. In this way, they filter traffic, a fact that makes them useful for controlling congestion. A bridge operates at data link layer giving it access to the physical address of all stations connected to it. When a frame enters a bridge, it not only regenerates the signal, but checks the destination and forwards the new copy only to the segment to which address belongs

Filtering

A bridge has filtering capability. It can check the destination address of a frame and decide if the frame should be forwarded or dropped. If the frame is to be forwarded, the decision must specify the port. A bridge has a table that maps addresses to ports.



A bridge connecting two LANs

In the above example, two LANs are connected by a bridge. If a frame destined for station 71:2B:13:45:61:42 arrives at port 1, the bridge consults its table to find the departing port. According to its table, frames for 7l:2B:13:45:61:42 leave through port 1; therefore, there is no need for forwarding, and the frame is dropped. On the other hand, if a frame for 71:2B:13:45:61:41 arrives at port 2, the departing port is port 1 and the frame is forwarded. In the first case, LAN 2 remains free of traffic; in the second case, both LANs have traffic. In our example, we show a two-port bridge; in reality a bridge usually has more ports. Note also that a bridge does not change the physical addresses contained in the frame.

TRANSPARENT BRIDGES

A transparent bridge is a bridge in which the stations are completely unaware of the bridge's existence. If a bridge is added or deleted from the system, reconfiguration of the stations is unnecessary. According to the IEEE 802.1 d specification, a system equipped with transparent bridges must meet three criteria:

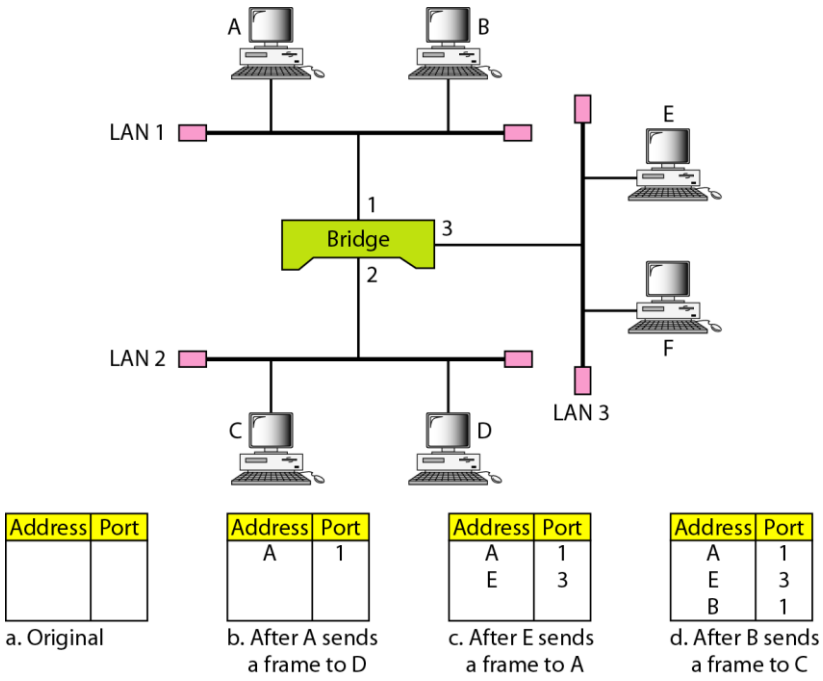
1. Frames must be forwarded from one station to another.
2. The forwarding table is automatically made by learning frame movements in the network.
3. Loops in the system must be prevented.

Forwarding: A transparent bridge must correctly forward the frames.

Learning: The earliest bridges had forwarding tables that were static. The systems administrator would manually enter each table entry during bridge setup. Although the process was simple, it was not practical. If a station was added or deleted, the table had to be modified manually. The same was true if a station's MAC address changed, which is not a rare event. For example, putting in a new network card means a new MAC address. A better solution to the static table is a dynamic table that maps addresses to ports automatically. To make a table dynamic, we need a bridge that gradually learns from the frame movements. To do this, the bridge inspects both the destination and the source addresses. The destination address is used for the forwarding decision (table lookup); the source address is used for adding entries to the table and for updating purposes.

1. When station A sends a frame to station D, the bridge does not have an entry for either D or A. The frame goes out from all three ports; the frame floods the network. However, by looking at the source address, the bridge learns that station A must be located on the LAN connected to port 1. This means that frames destined for A, in the future, must be sent out through port 1. The bridge adds this entry to its table. The table has its first entry now.

2. When station E sends a frame to station A, the bridge has an entry for A, so it forwards the frame only to port 1. There is no flooding. In addition, it uses the source address of the frame, E, to add a second entry to the table.



A learning bridge and the process of learning

3. When station B sends a frame to C, the bridge has no entry for C, so once again it floods the network and adds one more entry to the table.

4. The process of learning continues as the bridge forwards frames

TWO-LAYER SWITCHES

Two-layer switch performs at the physical and data link layers. A two-layer switch is a bridge, a bridge with many ports and a design that allows better (faster) performance. A bridge with a few ports can connect a few LANs together. A bridge with many ports may be able to allocate a unique port to each station, with each station on its own independent entity. A two-layer switch, as a bridge does, makes a filtering decision based on the MAC address of the frame it received. However, a two-layer switch can be more sophisticated. It can have a buffer to hold the frames for processing. It can have a switching factor that forwards the frames faster

There are two Types of Layer 2 Switches

- **Store-and-forward switch**
- **Cut-through switch**

Store-and-forward switch accepts frame on input line, buffers it briefly, then routes it to appropriate output line. There is delay between sender and receiver. Cut-through switch takes advantage of destination address appearing at beginning of frame. Switch begins repeating frame onto output line as soon as it recognizes destination address

5.13. Data Encoding

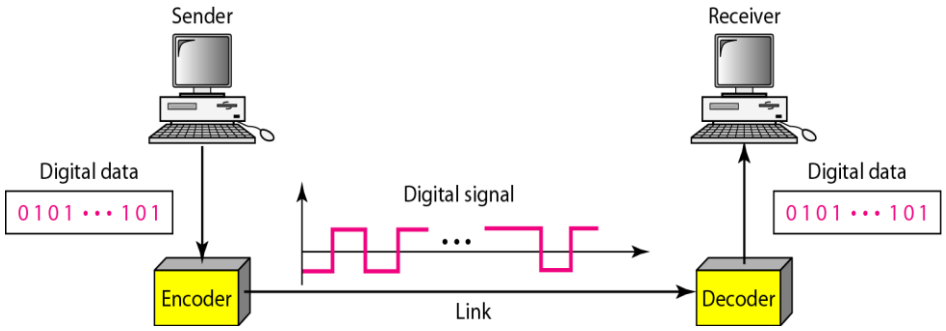
DIGITAL DATA, DIGITAL SIGNALS

A digital signal is a sequence of discrete, discontinuous voltage pulses, where each pulse is a signal element. Binary data are transmitted by encoding each data bit into signal elements. The conversion digital data into digital signals involves three techniques:

- line coding
- block coding
- scrambling

LINE CODING

Line coding converts a sequence of bits to a digital signal. At the sender, digital data are encoded into a digital signal; at the receiver, the digital data are recreated by decoding the digital signal. For example a high voltage level(+V) could represent a"1" and a low voltage level(0or-V)could representa"0".



Characteristics

Their common characteristics are:

Signal Element Versus Data Element

In data communications, our goal is to send data element, the smallest entity that can represent a piece of information. A signal element is the shortest unit (time wise) of a digital signal. Data elements are being carried; signal elements are the carriers

Mapping Data symbols onto Signal levels

A data symbol (or element) can consist of a number of data bits: ***1,0or 11, 10, 01,*** …… .A data symbol can be coded into a single signal element or multiple signal elements
1 -> +V, 0 -> -V
1 -> +V and -V, 0 -> -V and +V
The ratio „r" is the number of data elements carried by a signal element.

Data Rate versus Signal Rate

The data rate defines the number of data elements (bits) sent in 1s expressed in **bps.**The signal rate is the number of signal elements sent in Is. The unit is the **baud**.The data rate is sometimes called the **bit rate**; the signal rate is sometimes called the **pulse rate,** the **modulation rate**

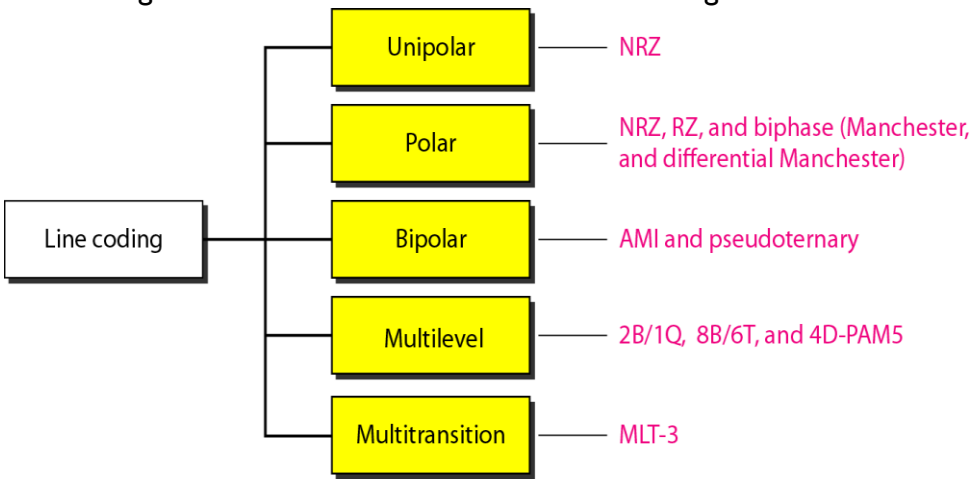
Data Rate versus Signal Rate
One goal in data communications is to increase the data rate while decreasing the signal rate. Increasing the data rate increases the speed of transmission; decreasing the signal rate decreases the bandwidth requirement. A formula for the relationship, b/w data rate and signal rate is:
S =c *N*1/r

Bandwidth

A digital signal which is non-periodic has a continuous band width with infinite range. Even if the bandwidth is theoretically infinite, many of the components have such a small amplitude that they can be ignored. So the effective bandwidth is finite. The baud rate, not the bitrate, determines the required bandwidth for a digital signal

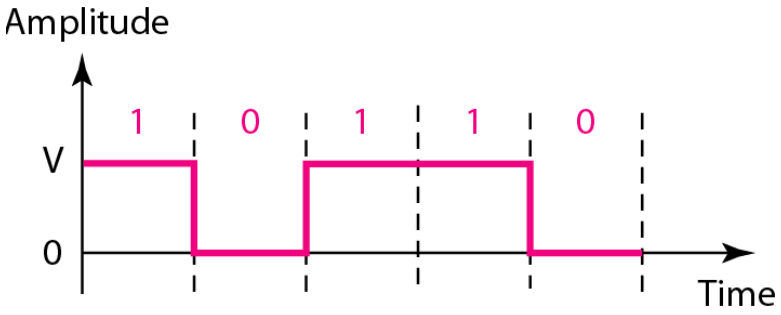
5.13.1LINE CODING SCHEMES

Line coding schemes are classified into five broad categories



UNIPOLAR

In a unipolar scheme, all the signal levels are on one side of the time axis, either above or below. **NRZ (Non-Return-to-Zero):**a unipolar scheme in which the positive voltage defines bit1 and the zero voltage defines bit0. It is called NRZ because the signal does not return to zero at the middle of the bit. Scheme is prone to baseline wandering and DC components. It has no synchronization or any error detection. It is simple but costly in power consumption .commonly used for digital magnetic recording.

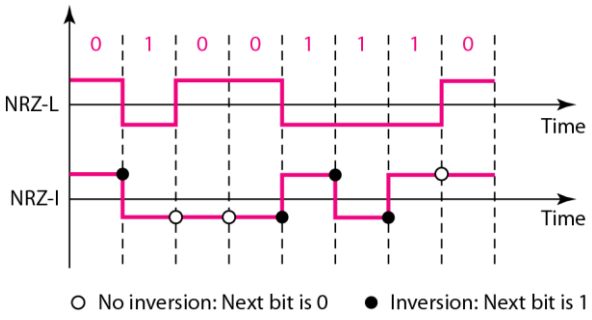


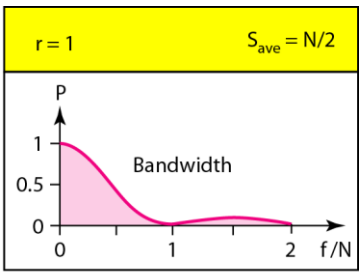
POLAR -NRZ

In polar schemes, the voltages are on the both sides of the time axis. In polar NRZ encoding, we use two levels of voltage amplitude. There are two versions:

☐**NRZ-L((NRZ-Level):** In **NRZ-L** the level of the voltage determines the value of the bit.
☐**NRZ-I (NRZ-Invert):** In **NRZ-I ,**the change or lack of change in the level of the voltage determines the value of the bit. If there is no change, the bit is 0; if there is a change, the bit is 1.

POLAR NRZ-L AND NRZ-I SCHEMES





POLAR NRZ-L AND NRZ-I SCHEMES

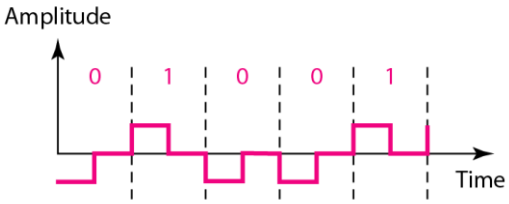
NRZ I is an example of **differential encoding.** In **differential encoding,** the information to be transmitted is represented in terms of the changes between successive signal elements rather than the signal elements themselves.
☐**NRZ-LandNRZ-I** both have an average signal rate of **N/2Bd**
☐**NRZ-LandNRZ-I** both have a DC component problem and baseline wandering, it is worse

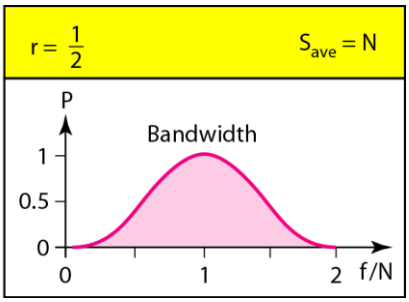
for NRZ-L.

☐Both have no self-synchronization & no error detection. Both are relatively simple to implement**.**

POLAR –RZ

The main problem with **NRZ** encoding occurs when the sender and receiver clocks are not synchronized. One solution is the **return-to-zero (RZ)** scheme, which uses three values: **positive**, **negative**, and **zero.** Each symbol has a transition in the middle. Either from high to zero or from low to zero .The main disadvantage of RZ encoding is that it requires two signal changes to encode a bit and therefore occupies greater bandwidth. No DC components or baseline wandering. Sudden change of polarity resulting in all 0"s interpreted as 1s and all 1s interpreted as 0"s.Self synchronization -transition indicates symbol value. More complex as it uses three voltage level. It has no error.





BIPHASE: MANCHESTER

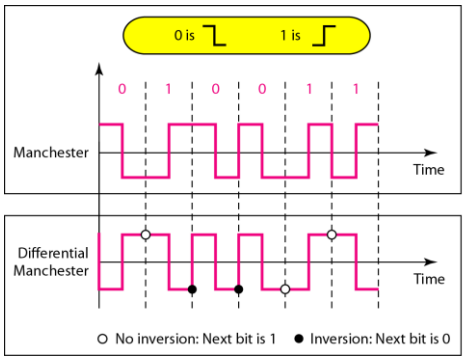
The idea of **RZ** and the idea of **NRZ-L** are combined into the **Manchester** scheme. Duration of the bit is divided into two halves and voltage remains at one level during the first half and moves to the other level in the second half. The transition at the middle of the bit provides synchronization.

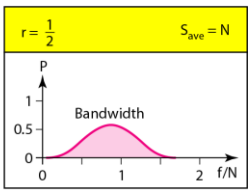
BIPHASE: DIFFERENTIAL MANCHESTER

Differential Manchester, combines the ideas of RZ and NRZ-I. There is always a transition at the middle of the bit, but the bit values are determined at the beginning of the bit. If the next bit is 0,there is a transition; if the next bit is1,there is none.

BIPHASE: PROPERTIES

The **Manchester** scheme overcomes several problems associated with **NRZ-L. Differential Manchester** overcomes several problems associated with **NRZ-I.** There is no **baseline wandering.** There is no **DC** component. The **signal rate** for Manchester and differential Manchester is double that for NRZ. None of these codes has error detection.

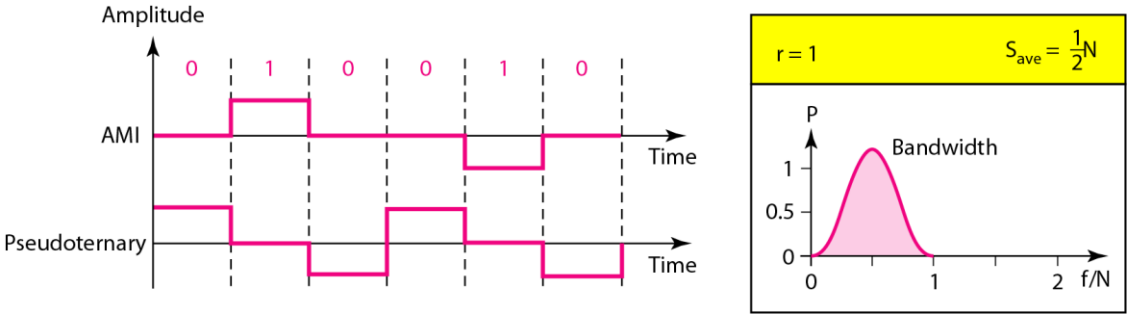




BIPOLAR SCHEMES

In bipolar encoding (sometimes called *multilevel binary), there are three voltage levels:* **positive, negative, and zero.** The voltage level for one data element is at zero, while the voltage level for the other element alternates between positive and negative. There are two variations of bipolar encoding:

- **AMI(*Alternate Mark Inversion):*** A neutral zero voltage represents binary 0. Binary 1"s are represented by alternating positive and negative voltages.
- **Pseudo ternary:** 1 bit is encoded as a zero voltage and the 0 bit is encoded as alternating positive and negative voltages.



Developed as an alternative to NRZ. The bipolar scheme the same signal rate as NRZ, but there is no DC component. The NRZ scheme has most of its energy concentrated near zero frequency, which makes it unsuitable for transmission over channels with poor performance around this frequency. The concentration of the energy in bipolar encoding is around frequency ***N/2.*** The pulse alternation property provides a simple means of error detection. Long string of 0s in the case of AMI or 1s in the case of pseudo ternary still presents a problem in synchronization.

Multilevel Schemes

In these schemes we increase the number of data bits per symbol there by increasing the bitrate. Since we are dealing with binary data we only have 2 types of data element a 1 or a 0.We can combine the 2 data elements into a pattern of "m" elements to create "2m" data patterns. If we have L signal levels, we can use "n" signal elements to create Ln signal patterns. Now we have 2m symbols and Ln signals.

- **If 2m> Ln** then we cannot represent the data elements, we don't have enough signals.

- **If 2m= Ln** then we have an exact mapping of one symbol on one signal.

- **If 2m< Ln** data patterns occupy only a subset of signal patterns. The subset can be carefully designed to prevent baseline wandering.

These type of coding is classified as ***mBnL.*** The first two letters define the data pattern, and the second two define the signal pattern. A letter is often used in place of L: *B* (binary) for *L =2, T (ternary) for L =3, and Q (quaternary) for L =4.* In **mBnL** schemes, a pattern of **m** data elements is encoded as a pattern of **n** signal elements in which **$2m \leq Ln$.**

Multilevel: 2B1Q scheme

Uses data patterns of size2 and encodes the 2-bitpatterns as one signal element belonging to a four-level signal. In this type of encoding *m=2,n=1,and L=4.*The average signal rate of 2BlQ is S=N/4.This means that using 2BlQ ,we can send data 2times faster than by using NRZ-L.There are no redundant signal patterns in this scheme because 22=41.

Multilevel: 8B6T scheme

The idea is to encode a pattern of 8 bits as a pattern of 6signal elements, where the signal has three levels. We can have 2-8=256 different data patterns and 3-6=478 different signal patterns. There are 478-256=222 redundant signal elements that provide synchronization and error detection. Part of the redundancy is also used to provide DC balance. Each signal pattern has a weight of 0 or +1 DC values. To make the whole stream Dc-balanced, the sender keeps track of the weight .If two groups of weight1 are encountered one after another, the first one is sent as is, while the next one is totally inverted to give a weight of -1, hence canceling the1st +1,for a DC balance. The average signal rate of the scheme is theoretically **6N/8.**

MultiLine Coding

Because of synchronization requirements we force transitions. This can result in very high bandwidth requirements. Codes can be created that are differential at the bit level forcing transitions at bit boundaries. This results in a bandwidth requirement that is equivalent to the bit rate. In some instances, the bandwidth requirement may even be lower, due to repetitive patterns resulting in a periodic signal.

MultiLine Coding-MLT-3

Similar to NRZ-I and differential Manchester in that transition define bits. Uses three voltage levels and three transition rules:

1.next bit is 0 **then no transition .**
2.next bit is 1 and current voltage is not zero **then next level is 0**
3.next bit is 1 and current voltage = 0 **then next level is the opposite of the** last non zero level

If there are a long sequence of 1s, the signal element pattern +VO –VO is repeated every 4 bits.
☐Anonperiodicsignalhaschangedtoaperiodicsignalwiththeperiodequalto4timesthebitduration.
☐It can be simulated as an analog signal with a frequency one-fourth of the bitrate.
☐In other words, the signal rate for MLT-3 is one-fourth the bitrate.

BLOCK CODING

For a code to be capable of **error detection,** we need to add redundancy, i.e., extra bits to the data bits. Synchronization also requires redundancy.Block coding can give us this redundancy and improve the performance of line coding.In general, block coding changes a block of ***m bits*** *into a* block of ***n bits****,* where **n**is larger than **m**.

- Referred to as an ***mB/nB*** **e**ncoding
- Block coding normally involves three steps:**division**,**substitution**,and**combination**.
- In the division step, a sequence of bits is divided into groups of *m bits*
- Substitute each***m-bit*** group with an***n-bit*** group**.**
- Finally, then-bit groups are combined together to form a stream

BLOCK CODING-4B/5B

The original bit sequence is divided into 4-bit groups. It substitute a 5-bit code for a 4-bit group. The 5-bit groups are combined to form the bit stream and 4B/5B Encoding used with NRZI avoids the problem of long sequences of **0's.**In4B/5B,the 5-bit output that replaces the4-bit input has no more than one leading zero(left bit)and no more than two trailing zeros(right bits).Sowhendifferentgroupsarecombinedtomakeanewsequence,therearenevermorethanthreeconsecutive*0's*.At the receiver, the NRZ-I encoded digital signal is first decoded into a stream of bits and then decoded to remove the redundancy.4B/5B encoding solves the problem of synchronization and overcomes one of the deficiencies of NRZ-I.But it increases the signal rate of NRZ-I.