# Chapter-3
# Data link layer



Compiled by :- Er. Nagendra Karn

# Data link layer :-

- Data link layer works between two hosts which are directly connected in some sense. This direct connection could be point to point or broadcast.
- Data link layer is responsible for converting data stream to signals bit by bit and to send that over the underlying hardware.
- At the receiving end, Data link layer picks up data from hardware which are in the form of electrical signals, assembles them in a recognizable frame format, and hands over to upper layer.
- Data link layer has two sub-layers :
  - Logical link control :- It deals with protocols, flow-control, and error control.
  - Media Access control :- It deals with actual control of media.

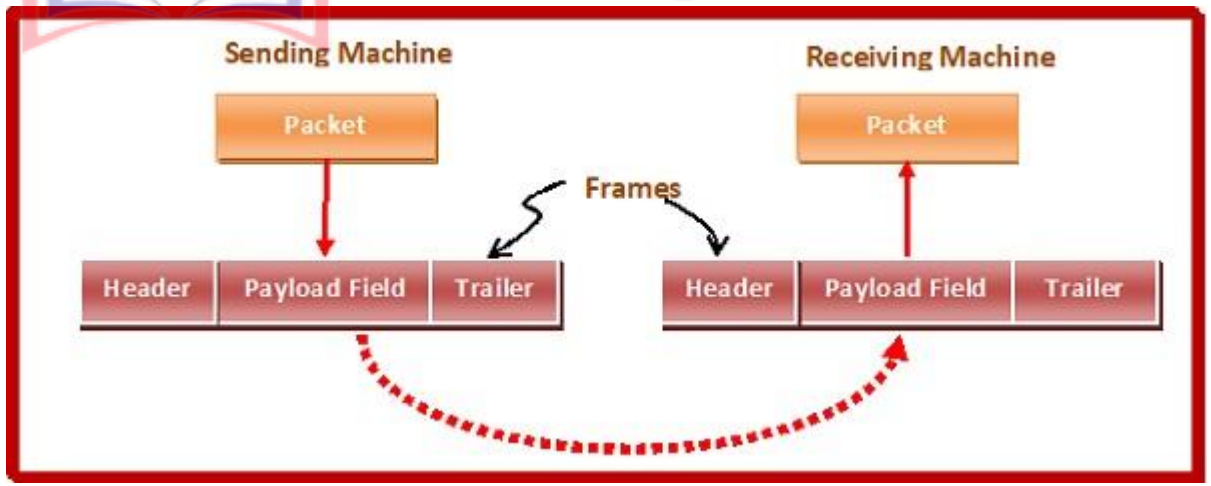
# Functionality of Data-link layer

- **Framing :-**
  - Data-link layer takes packets from Network Layer and encapsulates them into Frames.
  - Then, it sends each frame bit-by-bit on the hardware. At receiver' end, data link layer picks up signals from hardware and assembles them into frames.
- **Addressing :-**
  - Data-link layer provides layer-2 hardware addressing mechanism.
  - Hardware address is assumed to be unique on the link. It is encoded into hardware at the time of manufacturing.
- **Synchronization :-**
  - When data frames are sent on the link, both machines must be synchronized in order to transfer to take place.
- **Error Control :-**
  - Sometimes signals may have encountered problem in transition and the bits are flipped.
  - These errors are detected and attempted to recover actual data bits. It also provides error reporting mechanism to the sender.
- **Flow Control :-**
  - Stations on same link may have different speed or capacity.
  - Data-link layer ensures flow control that enables both machine to exchange data on same speed.
- **Multi-Access :-**
  - When host on the shared link tries to transfer the data, it has a high probability of collision. Data-link layer provides mechanism such as CSMA/CD to equip capability of accessing a shared media among multiple Systems.

# Framing :-

- In the physical layer, data transmission involves synchronized transmission of bits from the source to the destination. The data link layer packs these bits into frames.
- Data-link layer takes the packets from the Network Layer and encapsulates them into frames.
- If the frame size becomes too large, then the packet may be divided into small sized frames. Smaller sized frames makes flow control and error control more efficient.
- It sends each frame bit-by-bit on the hardware. At receiver's end, data link layer picks up signals from hardware and assembles them into frames.





# Parts of Frame :-

- A frame has following parts –
  - Frame header - It contains the source and the destination addresses of the frame.
  - Payload field – It contains the message to be delivered.
  - Trailer – It contains the error detection and error correction bits.
  - Flag – It marks the beginning and end of the frame.



# Types of Framing

- Framing can be of two types, fixed sized framing and variable sized framing.
- **Fixed-sized Framing**
  - Here the size of the frame is fixed and so the frame length acts as delimiter of the frame. Consequently, it does not require additional boundary bits to identify the start and end of the frame.
  - It suffers from internal fragmentation if data size is less than frame size. To overcome this problem we use padding.
  - Example ATM cells.
- **Variable- Sized Framing**
  - Here, the size of each frame to be transmitted may be different. So additional mechanisms are kept to mark the end of one frame and the beginning of the next frame.
  - It is used in local area network.

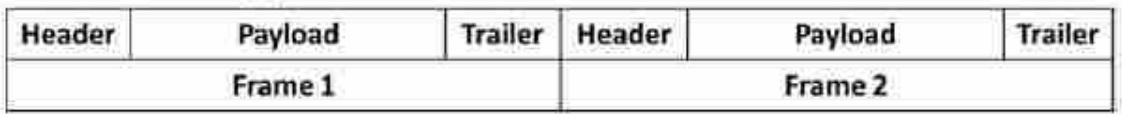
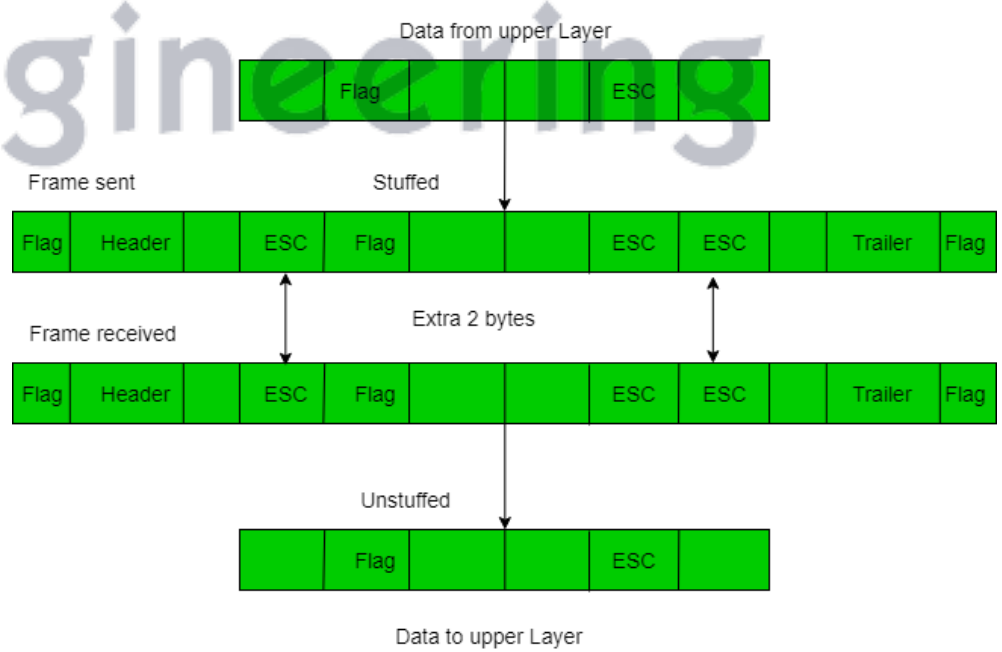



*Figure : Frame Format in Variable Size frame*

# Variable- Sized Framing

- This can be done in two ways :-
- **Length Field** - Here, a length field is used that determines the size of the frame. It is used in Ethernet (IEEE 802.3).
- **End Delimiter** – Here, a pattern is used as a delimiter to determine the size of frame. It is used in Token Rings. If the pattern occurs in the message, then two approaches are used to avoid the situation –
  - **Byte – Stuffing** – A byte is stuffed in the message to differentiate from the delimiter. This is also called character-oriented framing.
  - It is very costly and obsolete method.

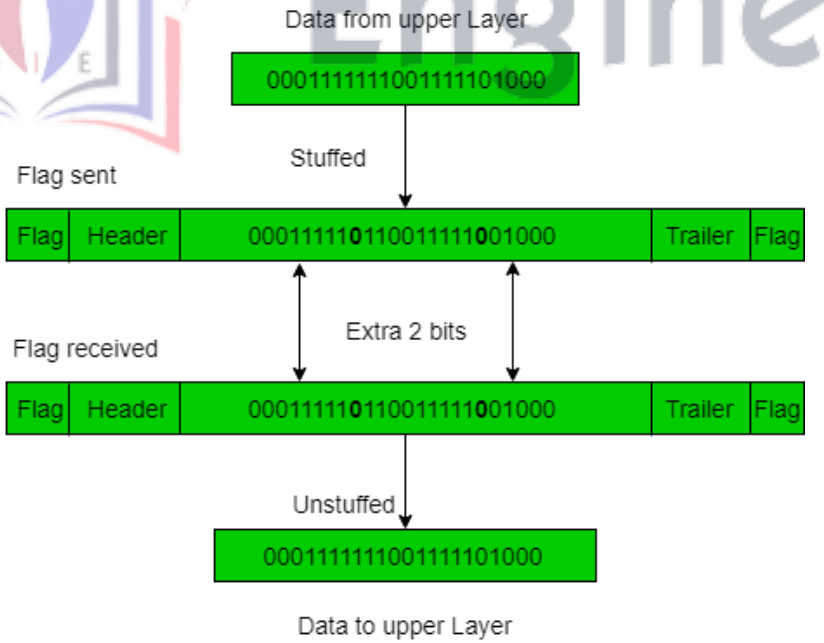




# Framing

- **Bit – Stuffing** − A pattern of bits of arbitrary length is stuffed in the message to differentiate from the delimiter. This is also called bit – oriented framing.
- Let ED = 01111 and if data = 01111
- Sender stuffs a bit to break the pattern i.e. here appends a 0 in data = 011101
- Receiver receives the frame.
- If data contains 011101, receiver removes the 0 and reads the data.





# Flow and Error Control

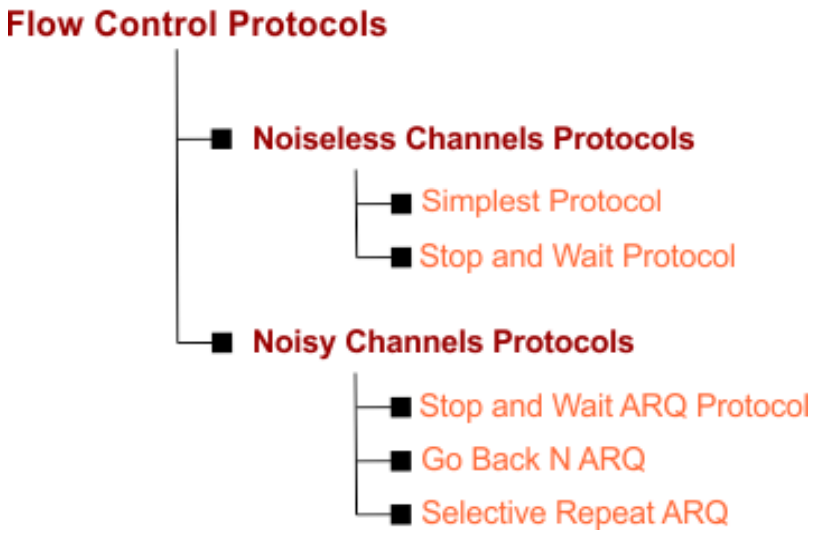
- **Data link control = Flow control + Error control**.
- Flow control is basically a technique that gives permission to two of stations that are working and processing at different speeds to just communicate with one another.
- Flow control in Data Link Layer simply restricts and coordinates number of frames or amount of data sender can send just before it waits for an acknowledgement from receiver.
- Flow control is actually set of procedures that explains sender about how much data or frames it can transfer or transmit before data overwhelms receiver.
- The receiving device also contains only limited amount of speed and memory to store data. This is why receiving device should be able to tell or inform the sender about stopping the transmission or transferring of data on temporary basis before it reaches limit.
- Two approaches :-

❖Feedback-based flow control

- ❖ Feedback to the sender telling how receiver is doing.

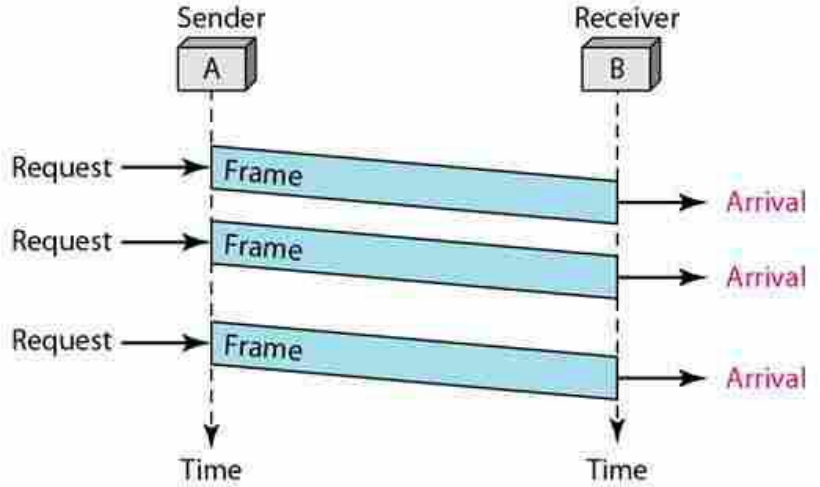
❖Rate based flow control

- ❖ Transfer rate is fixed by sender
- ❖ Not used often in DLL.



# Simplest :-

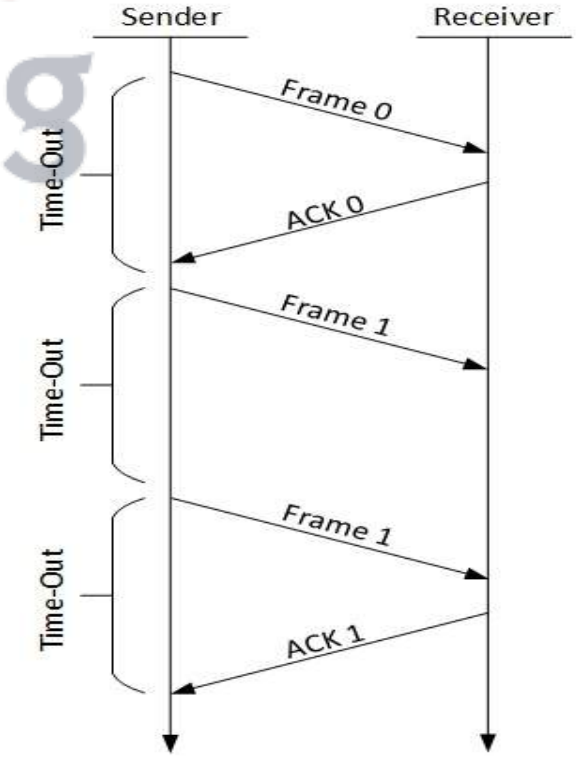
- Simplest Protocol is one that has no flow or error control and it is a unidirectional protocol in which data frames are traveling in only one direction-from the sender to receiver.
- We assume that the receiver can immediately handle any frame it receives with a processing time that is small enough to be negligible.
- The data link layer of the receiver immediately removes the header from the frame and hands the data packet to its network layer, which can also accept the packet immediately.
- The following figure shows an example of communication using this protocol. It is very simple.
- The sender sends a sequence of frames without even thinking about the receiver.
- To send three frames, three events occur at the sender site and three events at the receiver site.
- The height of the box defines the transmission time difference between the first bit and the last bit in the frame.



# Simplex stop and wait  protocol

- Error correction in Stop-and-Wait ARQ is done by keeping a copy of the sent frame and retransmitting of the frame when the timer expires.
- The sender maintains a timeout counter.
- When a frame is sent, the sender starts the timeout counter.
- If acknowledgement of frame comes in time, the sender transmits the next frame in queue.
- If acknowledgement does not come in time, the sender assumes that either the frame or its acknowledgement is lost in transit. Sender retransmits the frame and starts the timeout counter.
- If a negative acknowledgement is received, the sender retransmits the frame





# Sliding window protocol

- In this flow control mechanism, both sender and receiver agree on the number of data-frames after which the acknowledgement should be sent.
- As we learnt, stop and wait flow control mechanism wastes resources, this protocol tries to make use of underlying resources as much as possible.
- In this protocol, multiple frames can be sent by a sender at a time before receiving an acknowledgment from the receiver.
- The term sliding window refers to the imaginary boxes to hold frames. Sliding window method is also known as windowing.
- In these protocols, the sender has a buffer called the **sending window** and the receiver has buffer called the **receiving window.** The size of the sending window determines the sequence number of the outbound frames. The size of the receiving window is the maximum number of frames that the receiver can accept at a time.
- It determines the maximum number of frames that the sender can send before receiving acknowledgment.
- The types of sliding window protocol include:
  - A one bit sliding window protocol
  - A protocol using Go back N
  - A protocol using Selective Repeat

# One bit Sliding window protocol :

- In one – bit sliding window protocol, the size of the window is 1. So, the sender transmits a frame, waits for its acknowledgment, then transmits the next frame.
- Thus, it uses the concept of stop and wait protocol.
- This protocol provides for full – duplex communications. Hence, the acknowledgment is attached along with the next data frame to be sent called piggybacking.
- So, it is better compared to stop and wait due to full duplex communications.

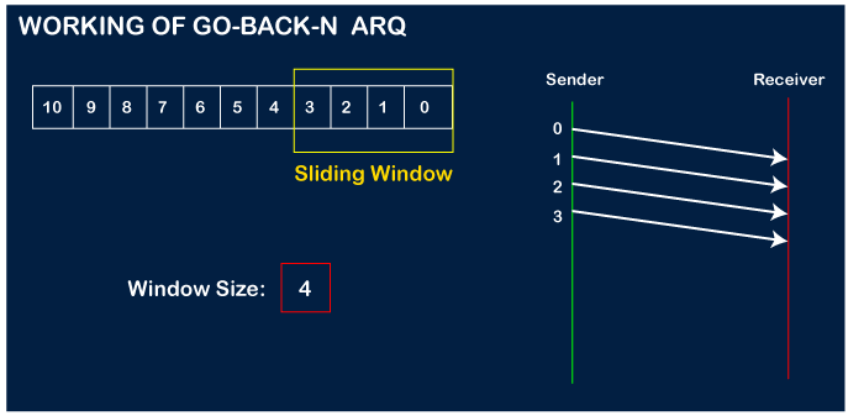
# Go-Back-N ARQ :-

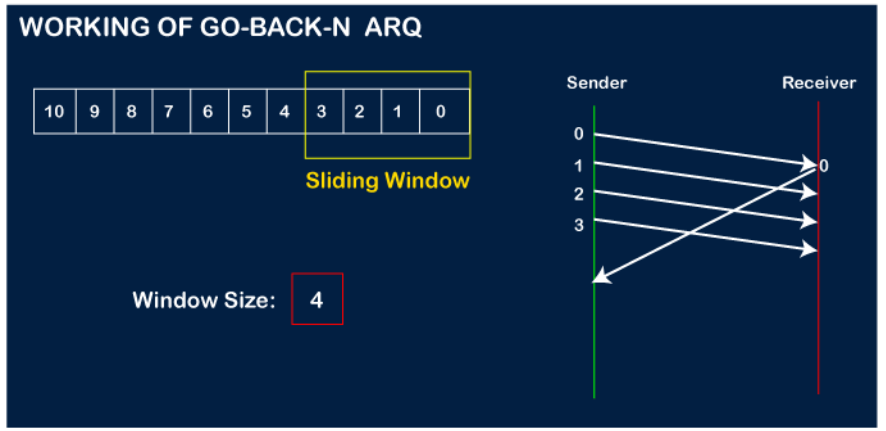
- In this protocol, we can send several frames before receiving acknowledgements; we keep a copy of these frames until the acknowledgements arrive.
- Stop and wait mechanism does not utilize the resources at their best. When the acknowledgement is received, the sender sits idle and does nothing.
- In Go-Back-N method, both sender and receiver maintain a window.
- In Go-Back-N ARQ, **N** is the sender's window size. Suppose we say that Go-Back-3, which means that the three frames can be sent at a time before expecting the acknowledgment from the receiver.

# Working of Go-Back-N ARQ

- Suppose there are a sender and a receiver, and let's assume that there are 11 frames to be sent.
- These frames are represented as 0,1,2,3,4,5,6,7,8,9,10, and these are the sequence numbers of the frames. Mainly, the sequence number is decided by the sender's window size. But, for the better understanding, we took the running sequence numbers, i.e., 0,1,2,3,4,5,6,7,8,9,10.
- Let's consider the window size as 4, which means that the four frames can be sent at a time before expecting the acknowledgment of the first frame.
- Let's assume that the receiver has sent the acknowledgment for the 0 frame, and the receiver has successfully received it.

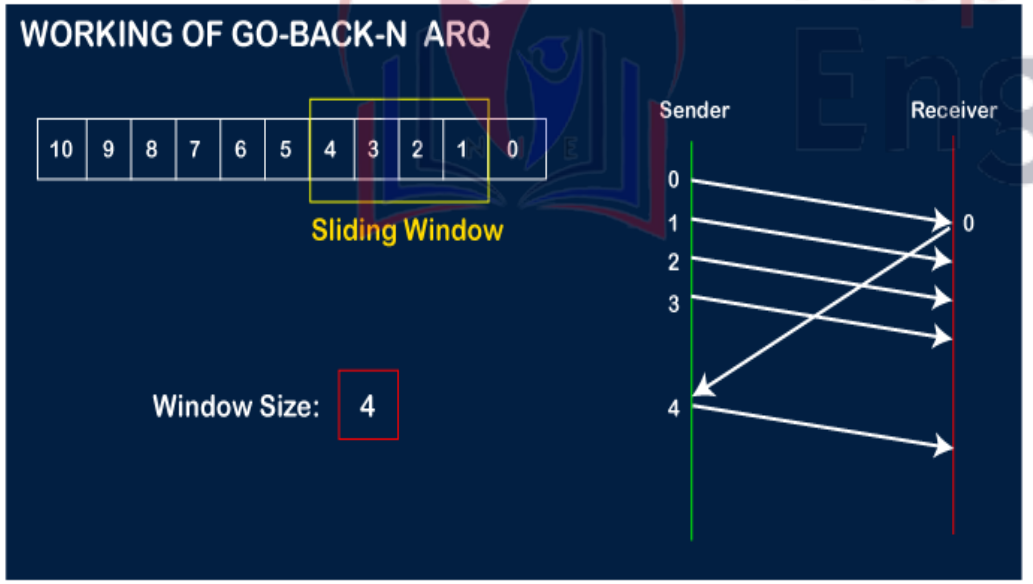


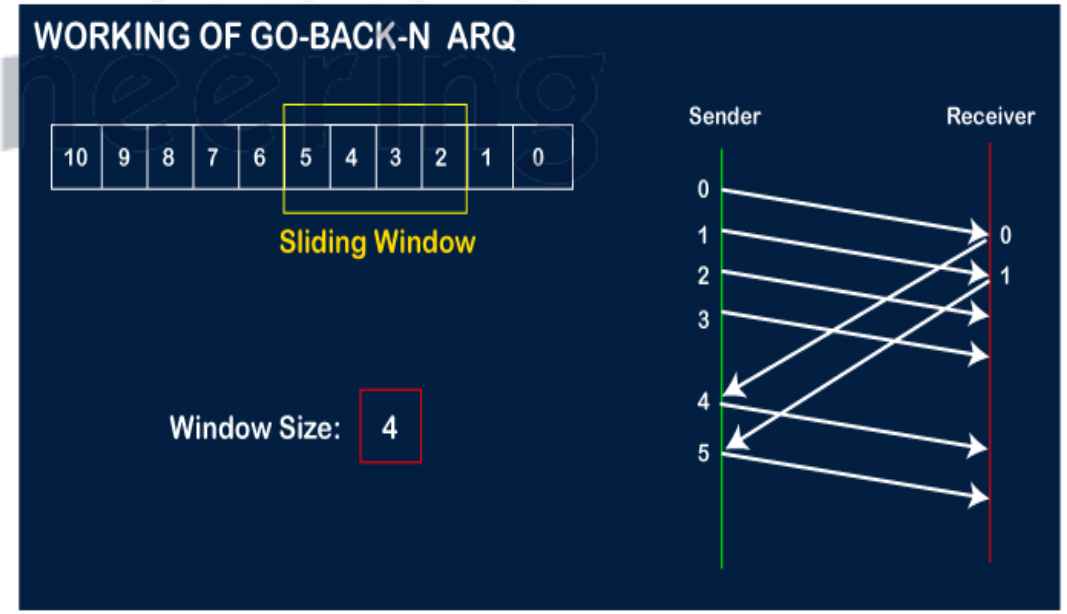


# Working of Go-Back-N ARQ (cont…)

- The sender will then send the next frame, i.e., 4, and the window slides containing four frames (1,2,3,4).
- The receiver will then send the acknowledgment for the frame no 1. After receiving the acknowledgment, the sender will send the next frame, i.e., frame no 5, and the window will slide having four frames (2,3,4,5).

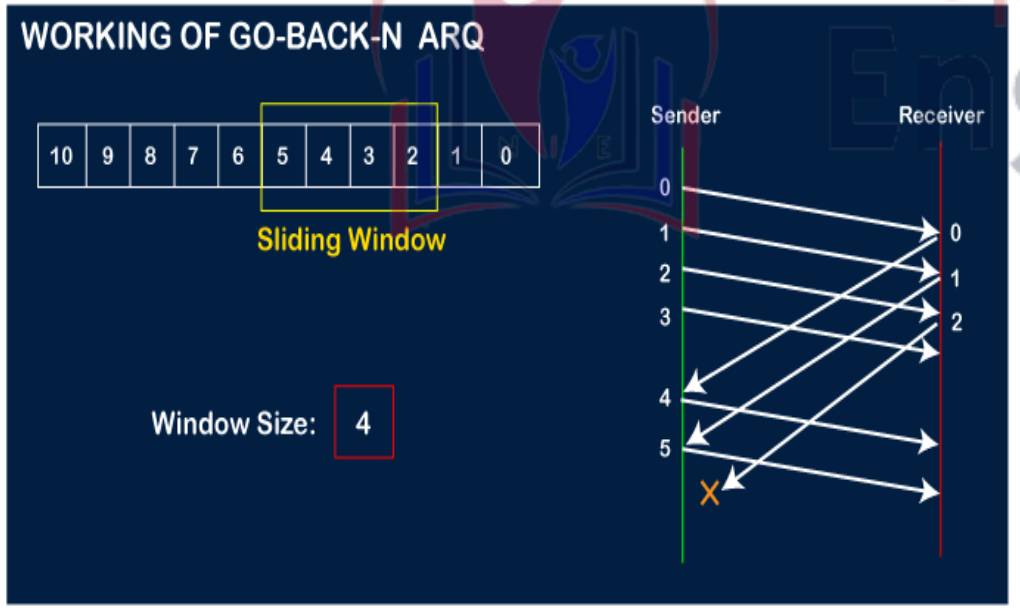


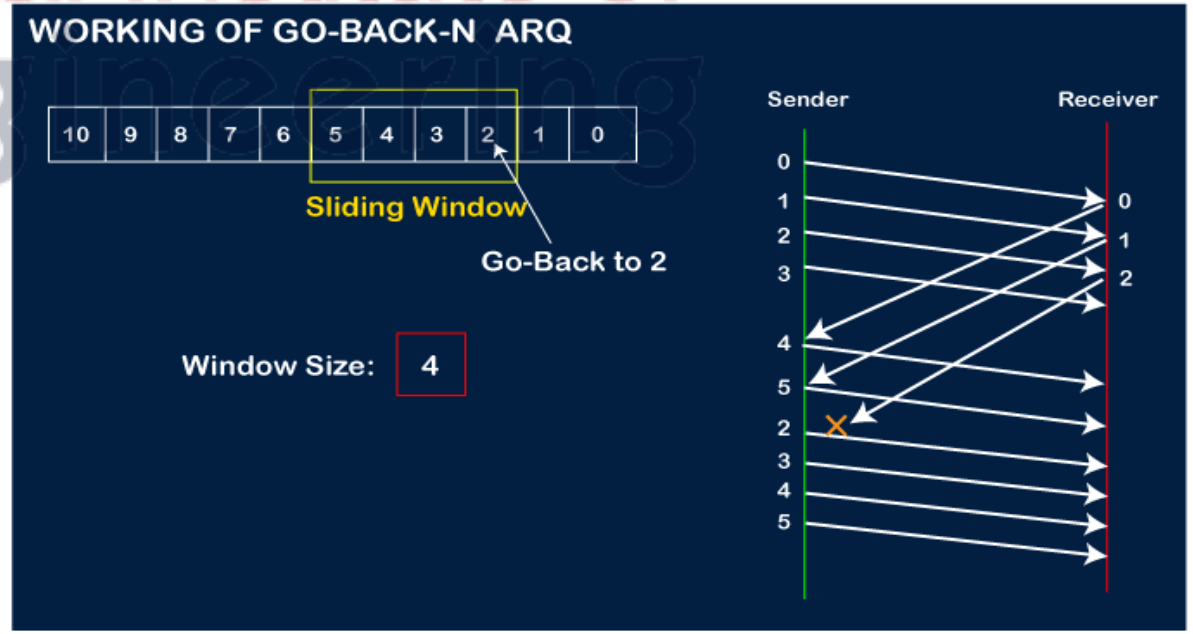



# Working of Go-Back-N ARQ (cont...)

- Now, let's assume that the receiver is not acknowledging the frame no 2, either the frame is lost, or the acknowledgment is lost. Instead of sending the frame no 6, the sender Go-Back to 2, which is the first frame of the current window, retransmits all the frames in the current window, i.e., 2,3,4,5.





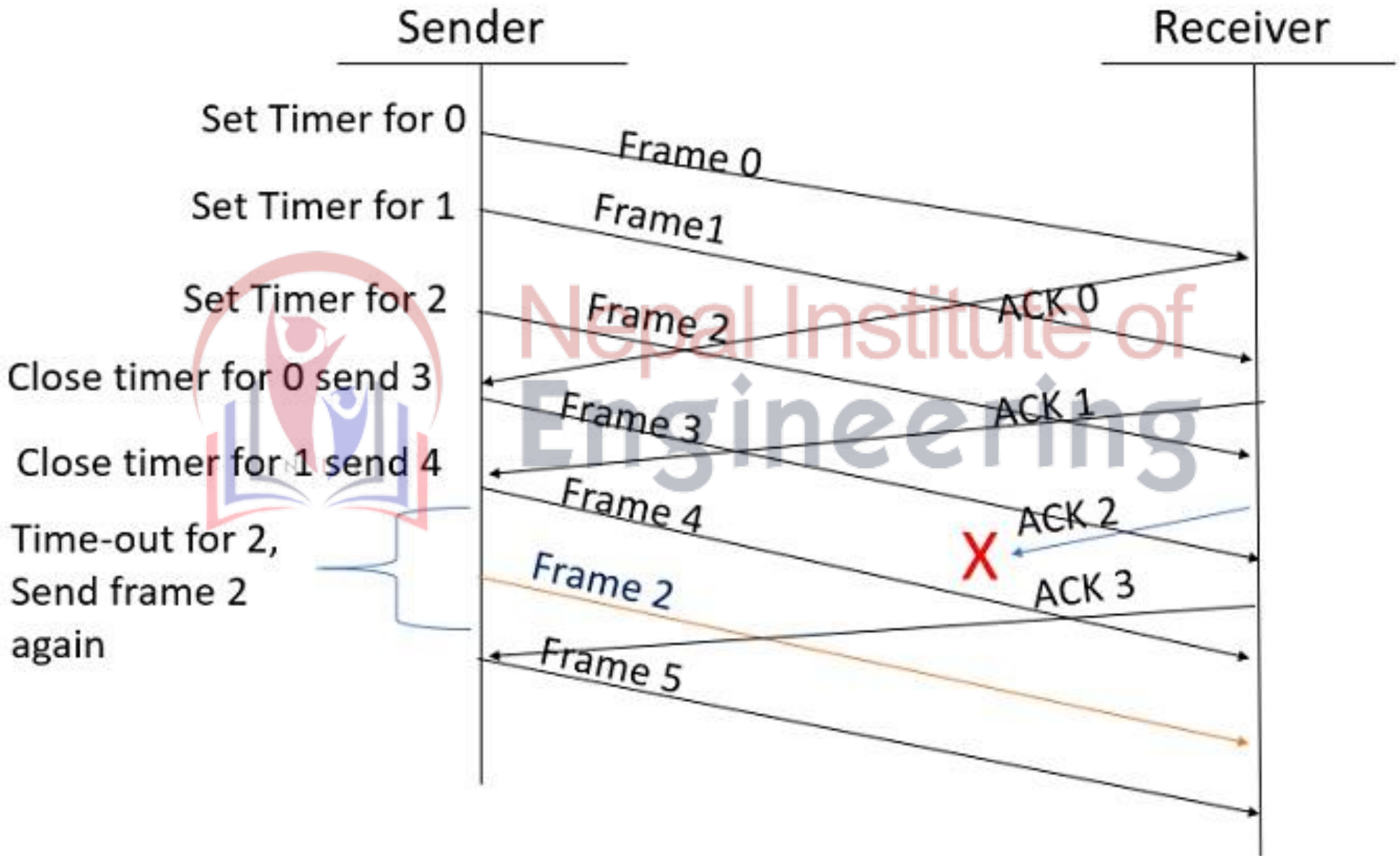



# Important points related to Go-Back-N ARQ :

- In Go-Back-N, N determines the sender's window size, and the size of the receiver's window is always 1.
- It does not consider the corrupted frames and simply discards them.
- It does not accept the frames which are out of order and discards them.
- If the sender does not receive the acknowledgment, it leads to the retransmission of all the current window frames.

# Selective Repeat ARQ

- Go-Back-N ARQ simplifies the process at the receiver site. The receiver keeps track of only one variable, and there is no need to buffer out-of-order frames; they are simply discarded. However, this protocol is very inefficient for a noisy link.
- In a noisy link a frame has a higher probability of damage, which means the resending of multiple frames. This resending uses up the bandwidth and slows down the transmission.
- For noisy links, there is another mechanism that does not resend N frames when just one frame is damaged; only the damaged frame is resent. This mechanism is called Selective Repeat ARQ.
- It is more efficient for noisy links, but the processing at the receiver is more complex.
- The Selective Repeat Protocol also uses two windows: a send window and a receive window.
- However, there are differences between the windows in this protocol and the ones in Go-Back-N. First, the size of the send window is much smaller; it is 2m- 1.



# Explanation

- **Step 1** − Frame 0 sends from sender to receiver and set timer.
- **Step 2** − Without waiting for acknowledgement from the receiver another frame, Frame1 is sent by sender by setting the timer for it.
- **Step 3** − In the same way frame2 is also sent to the receiver by setting the timer without waiting for previous acknowledgement.
- **Step 4** − Whenever sender receives the ACK0 from receiver, within the frame 0 timer then it is closed and sent to the next frame, frame 3.
- **Step 5** − whenever the sender receives the ACK1 from the receiver, within the frame 1 timer then it is closed and sent to the next frame, frame 4.
- **Step 6** − If the sender doesn't receive the ACK2 from the receiver within the time slot, it declares timeout for frame 2 and resends the frame 2 again, because it thought the frame2 may be lost or damaged.

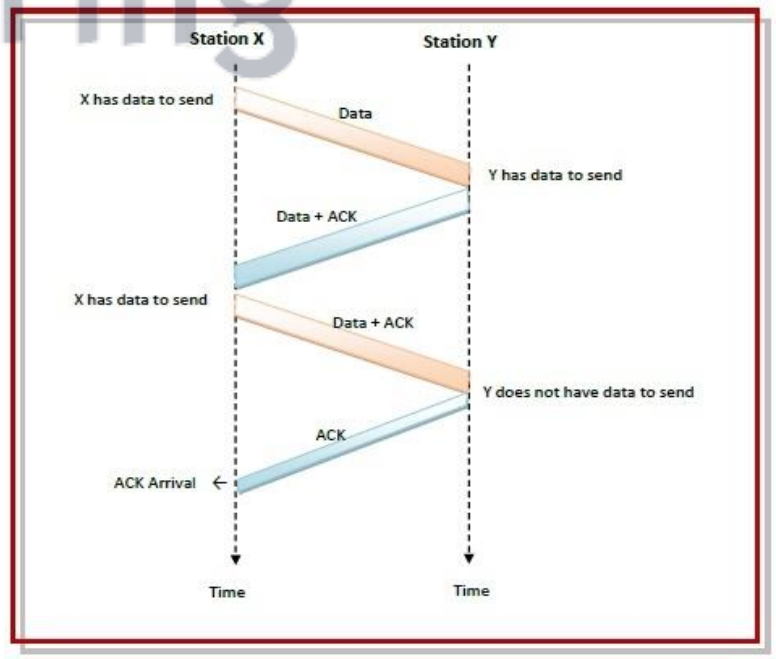
# Difference between the Go-Back-N ARQ and Selective Repeat ARQ?

| Go-Back-N ARQ | Selective Repeat ARQ |
|---|---|
| If a frame is corrupted or lost in it,all subsequent frames have to be sent again. | In this, only the frame is sent again, which is corrupted or lost. |
| If it has a high error rate,it wastes a lot of bandwidth. | There is a loss of low bandwidth. |
| It is less complex. | It is more complex because it has to do sorting and searching as well. And it also requires more storage. |
| It does not require sorting. | In this, sorting is done to get the frames in the correct order. |
| It does not require searching. | The search operation is performed in it. |
| It is used more. | It is used less because it is more complex. |

# Piggybacking Protocol

- Communications are mostly full – duplex in nature, i.e. data transmission occurs in both directions.
- A method to achieve full – duplex communication is to consider both the communication as a pair of simplex communication.
- Each link comprises a forward channel for sending data and a reverse channel for sending acknowledgments.
- However, in the above arrangement, traffic load doubles for each data unit that is transmitted. Half of all data transmission comprise of transmission of acknowledgments.
- So, a solution that provides better utilization of bandwidth is piggybacking. Here, sending of acknowledgment is delayed until the next data frame is available for transmission.
- The acknowledgment is then hooked onto the outgoing data frame. The data frame consists of an *ack* field. The size of the *ack* field is only a few bits, while an acknowledgment frame comprises of several bytes. Thus, a substantial gain is obtained in reducing bandwidth requirement.

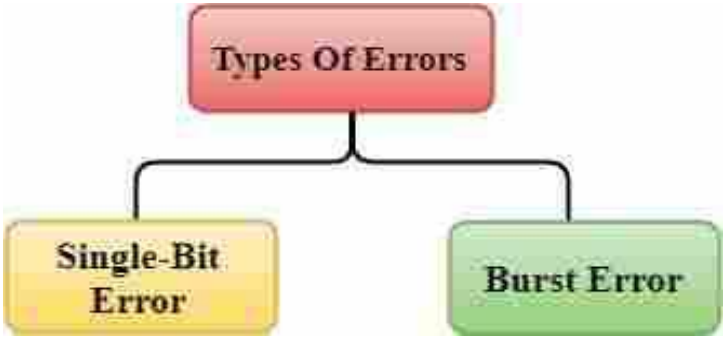




# Error Control -

- When bits are transmitted over the computer network, they are subject to get corrupted due to interference and network problems. The corrupted bits leads to spurious data being received by the receiver and are **called errors.**
- Error detection techniques are responsible for checking whether any error has occurred or not in the frame that has been transmitted via network.
- It does not take into account the number of error bits and the type of error.
- **When sender transmits data to the receiver, the data might get scrambled by noise or data might get corrupted during the transmission.**

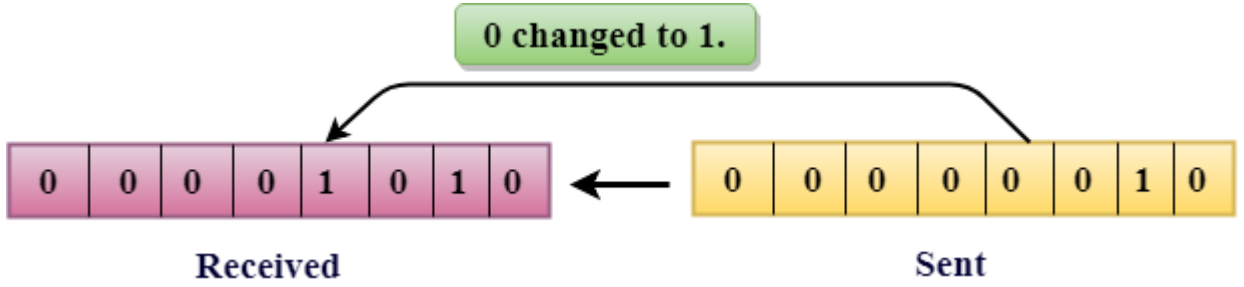




# Single-Bit Error :

- The only one bit of a given data unit is changed from 1 to 0 or from 0 to 1.
- **Single-Bit Error** does not appear more likely in Serial Data Transmission.
- For example, Sender sends the data at 10 Mbps, this means that the bit lasts only for 1s and for a single-bit error to occurred, a noise must be more than 1s.
- Single-Bit Error mainly occurs in Parallel Data Transmission. For example, if eight wires are used to send the eight bits of a byte, if one of the wire is noisy, then single-bit is corrupted per byte.

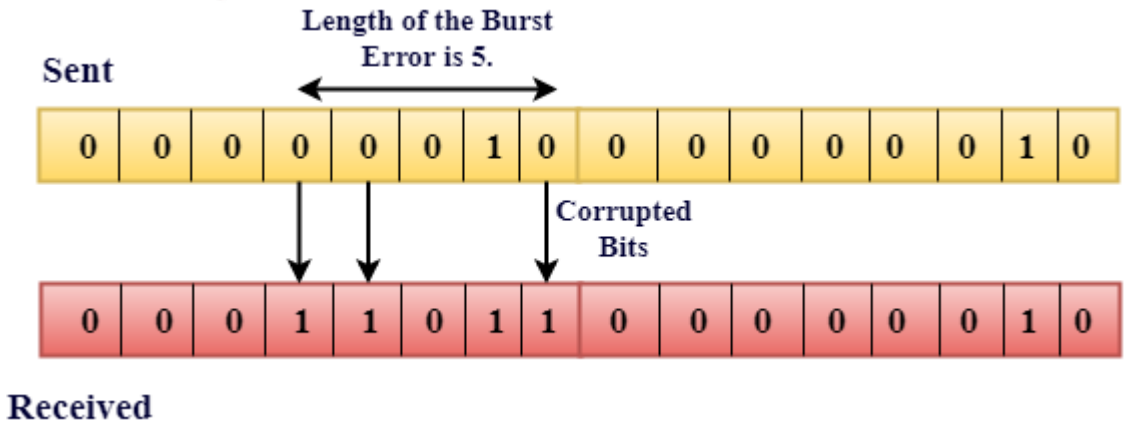




# Burst Error :

- The two or more bits are changed from 0 to 1 or from 1 to 0 is known as Burst Error.
- The Burst Error is determined from the first corrupted bit to the last corrupted bit.
- The duration of noise in Burst Error is more than the duration of noise in Single-Bit.
- Burst Errors are most likely to occurr in Serial Data Transmission.
- The number of affected bits depends on the duration of the noise and data rate.





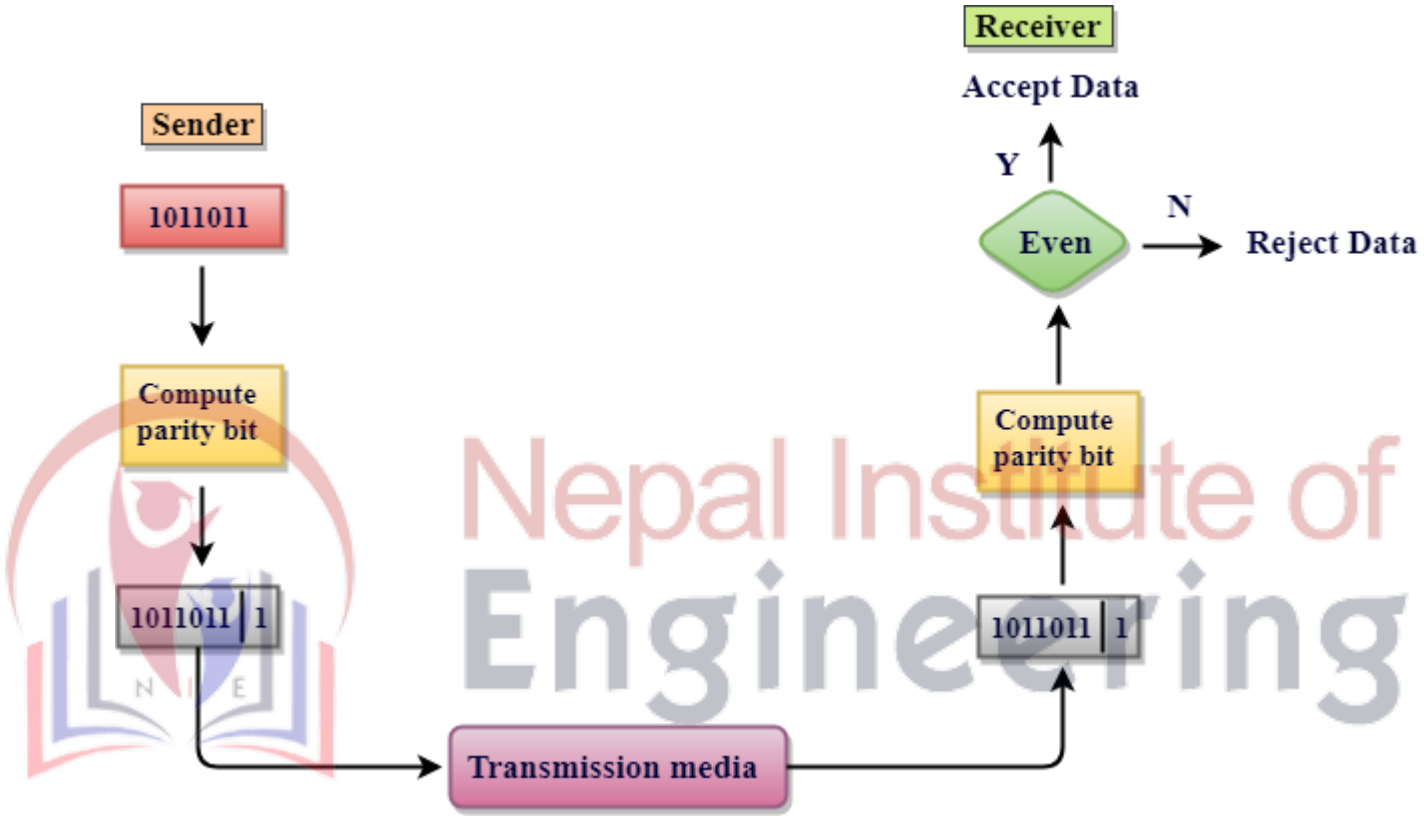
# Error Detecting Techniques :-

- The most popular Error Detecting Techniques are:
  - Single parity check
  - Two-dimensional parity check
  - Checksum
  - Cyclic redundancy check



- **Simple Parity check :-**

- Single Parity checking is the simple mechanism and inexpensive to detect the errors.
- In this technique, a redundant bit is also known as a parity bit which is appended at the end of the data unit so that the number of 1s becomes even. Therefore, the total number of transmitted bits would be 9 bits.
- If the number of 1s bits is odd, then parity bit 1 is appended and if the number of 1s bits is even, then parity bit 0 is appended at the end of the data unit.
- At the receiving end, the parity bit is calculated from the received data bits and compared with the received parity bit.
- This technique generates the total number of 1s even, so it is known as even-parity checking.






- **Drawback of single parity checking :-**
  - It can only detect single-bit errors which are very rare.
  - If two bits are interchanged, then it cannot detect the errors.

# Checksum :-

- A Checksum is an error detection technique based on the concept of redundancy.
- In checksum error detection scheme, the data is divided into k segments each of m bits.
- In the sender's end the segments are added using 1's complement arithmetic to get the sum. The sum is complemented to get the checksum.
- The checksum segment is sent along with the data segments.
- At the receiver's end, all received segments are added using 1's complement arithmetic to get the sum.
- The sum is complemented.
- If the result is zero, the received data is accepted; otherwise discarded.
- **It is divided into two parts :**
- **Checksum Generator :-**
  - A Checksum is generated at the sending side. Checksum generator subdivides the data into equal segments of n bits each, and all these segments are added together by using one's complement arithmetic. The sum is complemented and appended to the original data, known as checksum field. The extended data is transmitted across the network.
- **Checksum checker :-**
  - A Checksum is verified at the receiving side. The receiver subdivides the incoming data into equal segments of n bits each, and all these segments are added together, and then this sum is complemented. If the complement of the sum is zero, then the data is accepted otherwise data is rejected.

# Example

- If the data unit to be transmitted is 10101001 00111001, the following procedure is used at Sender site and Receiver site.
- Sender Site :
- 10101001 subunit 1
- 00111001 subunit 2
- 11100010 sum (using 1s complement)
- 00011101 checksum (complement of sum)
- Data transmitted to Receiver is –

- Receiver Site :
- 10101001 subunit 1
- 00111001 subunit 2
- 00011101 checksum
- 11111111 sum
- 00000000 sum's complement
- Result is zero, it means no error.

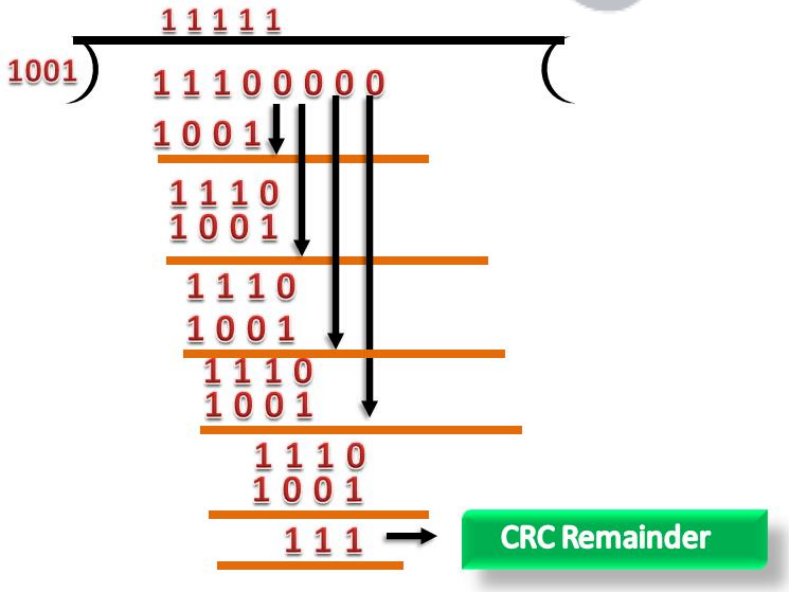
# Cyclic redundancy check (CRC)

- Unlike checksum scheme, which is based on addition, CRC is based on binary division.
- In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended to the end of data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number.
- At the destination, the incoming data unit is divided by the same number. If at this step there is no remainder, the data unit is assumed to be correct and is therefore accepted.
- A remainder indicates that the data unit has been damaged in transit and therefore must be rejected.
- **Following are the steps used in CRC for error detection :-**
- In CRC technique, a string of n 0s is appended to the data unit, and this n number is less than the number of bits in a predetermined number, known as division which is n+1 bits.
- Secondly, the newly extended data is divided by a divisor using a process is known as binary division. The remainder generated from this division is known as CRC remainder.
- Thirdly, the CRC remainder replaces the appended 0s at the end of the original data. This newly generated unit is sent to the receiver.
- The receiver receives the data followed by the CRC remainder. The receiver will treat this whole unit as a single unit, and it is divided by the same divisor that was used to find the CRC remainder.
- If the resultant of this division is zero which means that it has no error, and the data is accepted.

# Cyclic redundancy check (CRC)

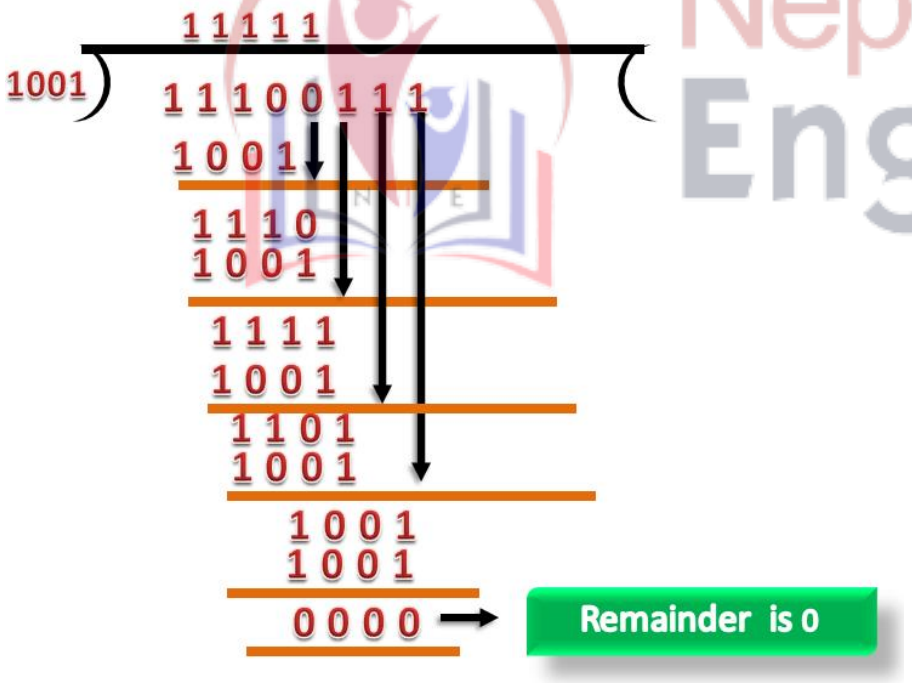
- **Suppose the original data is 11100 and divisor is 1001.**
- CRC Generator :-
- A CRC generator uses a modulo-2 division. Firstly, three zeroes are appended at the end of the data as the length of the divisor is 4 and we know that the length of the string 0s to be appended is always one less than the length of the divisor.
- Now, the string becomes 11100000, and the resultant string is divided by the divisor 1001.
- The remainder generated from the binary division is known as CRC remainder. The generated value of the CRC remainder is 111.
- CRC remainder replaces the appended string of 0s at the end of the data unit, and the final string would be 11100111 which is sent across the network.

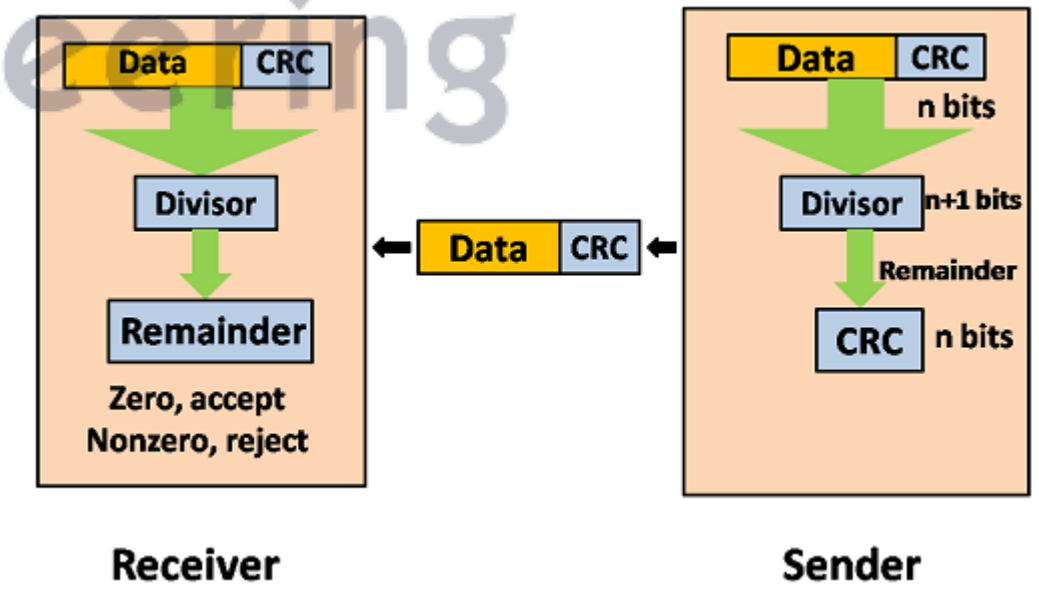




# CRC Checker

- The functionality of the CRC checker is similar to the CRC generator.
- When the string 11100111 is received at the receiving end, then CRC checker performs the modulo-2 division.
- A string is divided by the same divisor, i.e., 1001.
- In this case, CRC checker generates the remainder of zero. Therefore, the data is accepted.



# Error Detection and correction  Codes :-

- Error detection and correction code plays an important role in the transmission of data from one source to another.
- The noise also gets added into the data when it transmits from one system to another, which causes errors in the received binary data at other systems.
- The bits of the data may change(either 0 to 1 or 1 to 0) during transmission.
- It is impossible to avoid the interference of noise, but it is possible to get back the original data. For this purpose, we first need to detect either an error **z** is present or not using error detection codes. If the error is present in the code, then we will correct it with the help of error correction codes.
- **Error detection code**
- The error detection codes are the code used for detecting the error in the received data **bitstream**. In these codes, some bits are included appended to the original bitstream.
- Error detecting codes encode the message before sending it over the noisy channels. The encoding scheme is performed in such a way that the decoder at the receiving can find the errors easily in the receiving data with a higher chance of success.

## Parity Check :-

- In error-correcting codes, parity check has a simple way to detect errors along with a sophisticated mechanism to determine the corrupt bit location. Once the corrupt bit is located, its value is reverted (from 0 to 1 or 1 to 0) to get the original message.
- **How to Detect and Correct Errors?**
- To detect and correct the errors, additional bits are added to the data bits at the time of transmission.
  - The additional bits are called **parity bits**. They allow detection or correction of the errors.
  - The data bits along with the parity bits form a **code word**.

# Parity Checking of Error Detection

- It is the simplest technique for detecting and correcting errors. The MSB of an 8-bits word is used as the parity bit and the remaining 7 bits are used as data or message bits.
- The parity of 8-bits transmitted word can be either even parity or odd parity.




- **Even parity** -- Even parity means the number of 1's in the given word including the parity bit should be even (2,4,6,....).
- **Odd parity** -- Odd parity means the number of 1's in the given word including the parity bit should be odd (1,3,5,....).

# Use of Parity Bit

- The parity bit can be set to 0 and 1 depending on the type of the parity required.
- For even parity, this bit is set to 1 or 0 such that the no. of "1 bits" in the entire word is even. Shown in fig. (a).
- For odd parity, this bit is set to 1 or 0 such that the no. of "1 bits" in the entire word is odd. Shown in fig. (b).



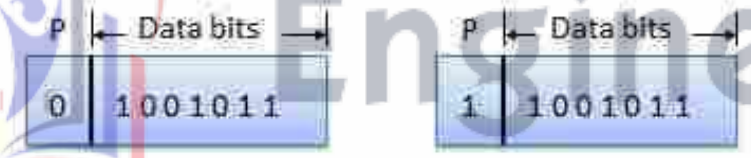



Fig. (a)

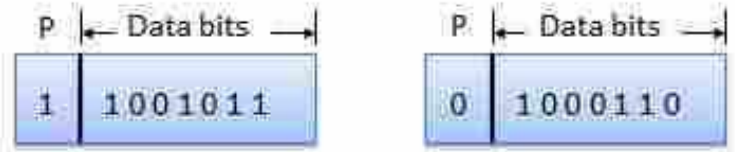
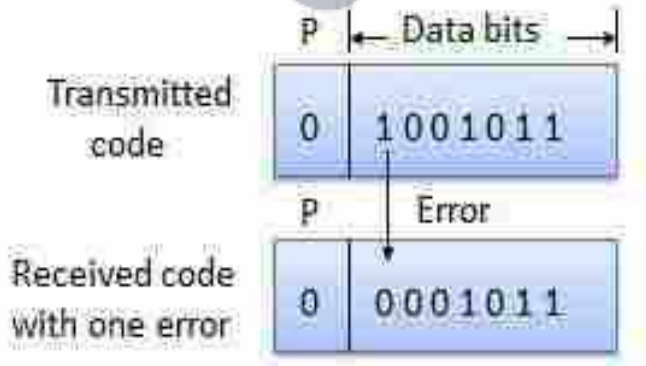



Fig. (b)

# How Does Error Detection Take Place?

- Parity checking at the receiver can detect the presence of an error if the parity of the receiver signal is different from the expected parity.
- That means, if it is known that the parity of the transmitted signal is always going to be "even" and if the received signal has an odd parity, then the receiver can conclude that the received signal is not correct. If an error is detected, then the receiver will ignore the received byte and request for retransmission of the same byte to the transmitter.





# Hamming Code

- Hamming code is a block code that is capable of detecting up to two simultaneous bit errors and correcting single-bit errors. It was developed by R.W. Hamming for error correction.
- In this coding method, the source encodes the message by inserting redundant bits within the message.
- These redundant bits are extra bits that are generated and inserted at specific positions in the message itself to enable error detection and correction.
- When the destination receives this message, it performs recalculations to detect errors and find the bit position that has error.



## • Hamming Code for Single Error Correction

- The procedure for single error correction by Hamming Code includes two parts, encoding at the sender's end and decoding at receiver's end.

# Encoding a message by Hamming Code

- The procedure used by the sender to encode the message encompasses the following steps −
  - Step 1 − Calculation of the number of redundant bits.
  - Step 2 − Positioning the redundant bits.
  - Step 3 − Calculating the values of each redundant bit.
- Once the redundant bits are embedded within the message, this is sent to the destination.
- **Step 1 − Calculation of the number of redundant bits.**
- If the message contains *m* number of data bits, *r* number of redundant bits are added to it so that is able to indicate at least *(m + r + 1)* different states.
- Here, *(m + r)* indicates location of an error in each of bit positions and one additional state indicates no error.
- Since, *r* bits can indicate $2^r$ states, $2^r$ must be at least equal to *(m + r + 1)*. Thus the following equation should hold −

$$2^r \geq m + r + 1$$

- **Example 1 −**
  - If the data is of 7 bits, i.e. $m = 7$, the minimum value of $r$ that will satisfy the above equation is 4, $(2^4 \geq 7 + 4 + 1)$. The total number of bits in the encoded message, $(m + r) = 11$. This is referred as (11,4) code.

- **Step 2 − Positioning the redundant bits.**
- The $r$ redundant bits placed at bit positions of powers of 2, i.e. 1, 2, 4, 8, 16 etc.
- They are referred in the rest of this text as $r_1$ (at position 1), $r_2$ (at position 2), $r_3$ (at position 4), $r_4$ (at position 8) and so on.
- **Example 2** − If, $m = 7$ comes to 4, the positions of the redundant bits are as follows −

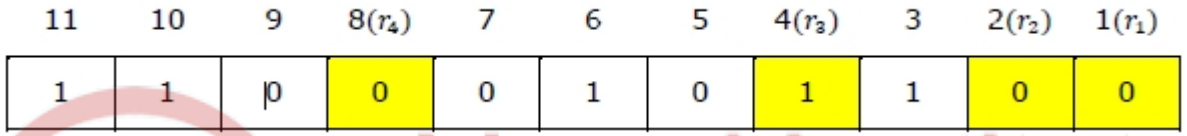


| 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|
| $d$ | $d$ | $d$ | $r_4$ | $d$ | $d$ | $d$ | $r_3$ | $d$ | $r_2$ | $r_1$ |

# Step 3 – Calculating the values of each redundant bit.

- The redundant bits are parity bits. A parity bit is an extra bit that makes the number of 1s either even or odd. The two types of parity are −
- **Even Parity** − Here the total number of bits in the message is made even.
- **Odd Parity** − Here the total number of bits in the message is made odd.
- Each redundant bit, $r_i$, is calculated as the parity, generally even parity, based upon its bit position. It covers all bit positions whose binary representation includes a 1 in the $i^{th}$ position except the position of $r_i$. Thus −
- $r_1$ is the parity bit for all data bits in positions whose binary representation includes a 1 in the least significant position excluding 1 (3, 5, 7, 9, 11 and so on)
- $r_2$ is the parity bit for all data bits in positions whose binary representation includes a 1 in the position 2 from right except 2 (3, 6, 7, 10, 11 and so on)
- $r_3$ is the parity bit for all data bits in positions whose binary representation includes a 1 in the position 3 from right except 4 (5-7, 12-15, 20-23 and so on)

- **Example 3** –
- Suppose that the message 1100101 needs to be encoded using even parity Hamming code. Here, *m = 7* and *r* comes to 4. The values of redundant bits will be as follows −



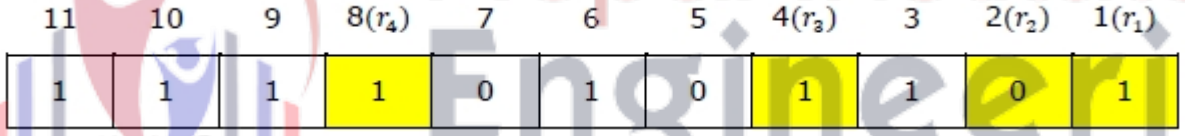

- Hence, the message sent will be 11000101100.
- **Decoding a message in Hamming Code**
- Once the receiver gets an incoming message, it performs recalculations to detect errors and correct them. The steps for recalculation are −
- Step 1 − Calculation of the number of redundant bits.
- Step 2 − Positioning the redundant bits.
- Step 3 − Parity checking.
- Step 4 − Error detection and correction

- **Step 1) Calculation of the number of redundant bits**
- Using the same formula as in encoding, the number of redundant bits are ascertained.

$$2^r \geq m + r + 1$$

- where *m* is the number of data bits and *r* is the number of redundant bits.
- **Step 2) Positioning the redundant bits**
- The *r* redundant bits placed at bit positions of powers of 2, i.e. 1, 2, 4, 8, 16 etc.
- **Step 3) Parity checking**
- Parity bits are calculated based upon the data bits and the redundant bits using the same rule as during generation of $c_1$, $c_2$, $c_3$, $c_4$ etc. Thus
- $c_1$ = parity(1, 3, 5, 7, 9, 11 and so on)
- $c_2$ = parity(2, 3, 6, 7, 10, 11 and so on)
- $c_3$ = parity(4-7, 12-15, 20-23 and so on)

- **Step 4) Error detection and correction**
- The decimal equivalent of the parity bits binary values is calculated. If it is 0, there is no error. Otherwise, the decimal value gives the bit position which has error. For example, if $c_1c_2c_3c_4$ = 1001, it implies that the data bit at position 9, decimal equivalent of 1001, has error. The bit is flipped (converted from 0 to 1 or vice versa) to get the correct message.
- **Example 4** − Suppose that an incoming message 11110101101 is received.
- **Step 1** − At first the number of redundant bits are calculated using the formula $2^r \geq m + r + 1$. Here, m + r + 1 = 11 + 1 = 12. The minimum value of *r* such that $2^r \geq 12$ is 4.
- **Step 2** − The redundant bits are positioned as below −



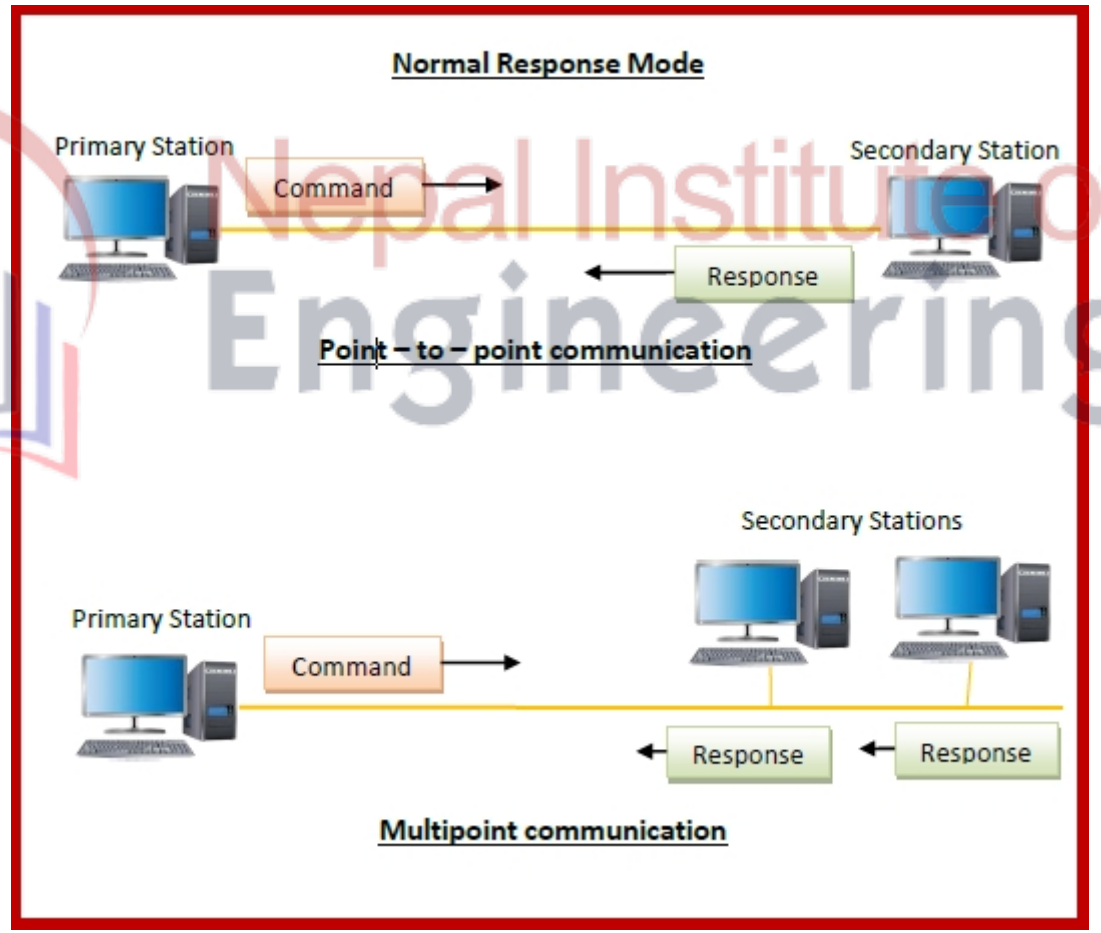

- **Step 3 − Even parity checking is done −**
- $c_1$ = even_parity(1, 3, 5, 7, 9, 11) = 0
- $c_2$ = even_parity(2, 3, 6, 7, 10, 11) = 0
- $c_3$ = even_parity (4, 5, 6, 7) = 0
- $c_4$ = even_parity (8, 9, 10, 11) = 0
- **Step 4 -** Since the value of the check bits $c_1c_2c_3c_4$ = 0000 = 0, there are no errors in this message.

# High-level Data Link Control (HDLC)

- High-level Data Link Control (HDLC) is a group of communication protocols of the data link layer for transmitting data between network points or nodes.
- Since it is a data link protocol, data is organized into frames. A frame is transmitted via the network to the destination that verifies its successful arrival.
- It is a bit - oriented protocol that is applicable for both point - to - point and multipoint communications.
- Transfer Modes
- HDLC supports two types of transfer modes, normal response mode and asynchronous balanced mode.

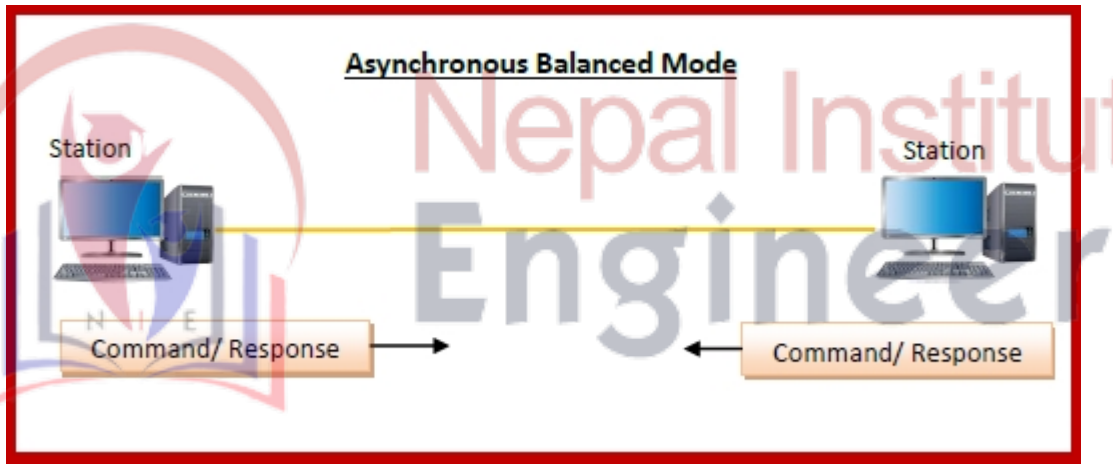
# Normal Response Mode (NRM)

- Here, two types of stations are there, a primary station that send commands and secondary station that can respond to received commands. It is used for both point - to - point and multipoint communications.



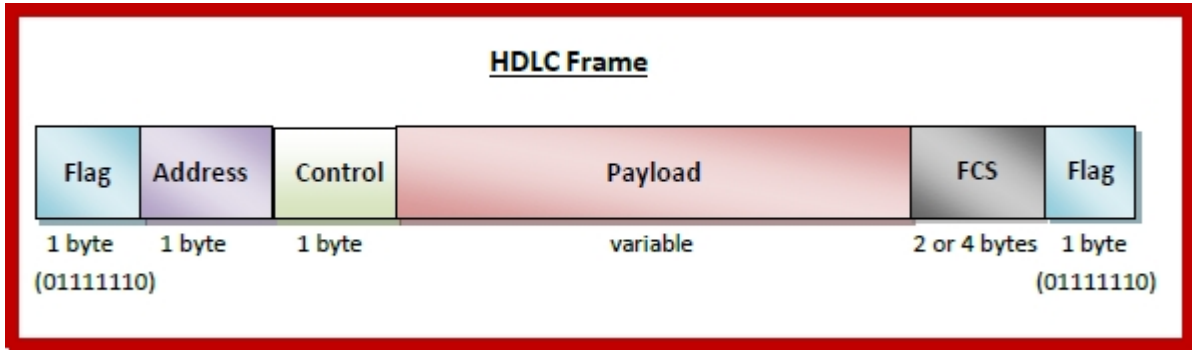
# Asynchronous Balanced Mode (ABM)

- Here, the configuration is balanced, i.e. each station can both send commands and respond to commands. It is used for only point - to - point communications.



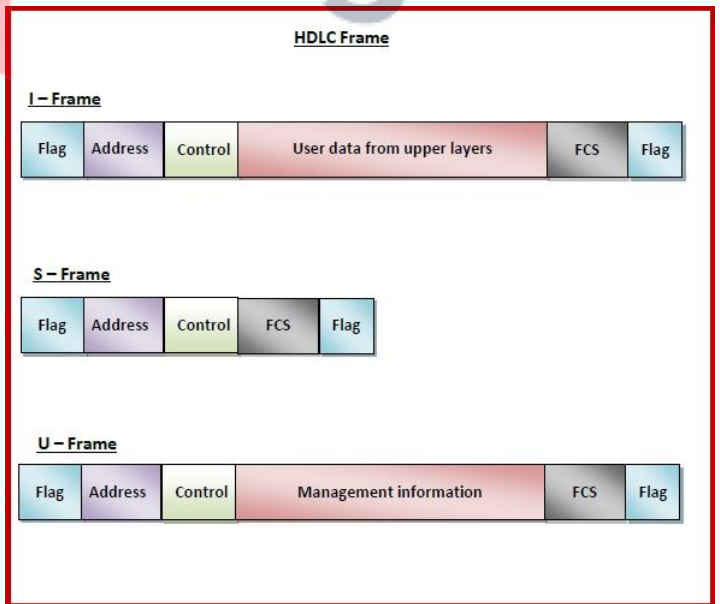
# HDCL Frame

- HDLC is a bit - oriented protocol where each frame contains up to six fields. The structure varies according to the type of frame. The fields of a HDLC frame are −
- **Flag** − It is an 8-bit sequence that marks the beginning and the end of the frame. The bit pattern of the flag is 01111110.
- **Address** − It contains the address of the receiver. If the frame is sent by the primary station, it contains the address(es) of the secondary station(s). If it is sent by the secondary station, it contains the address of the primary station. The address field may be from 1 byte to several bytes.
- **Control** − It is 1 or 2 bytes containing flow and error control information.
- **Payload** − This carries the data from the network layer. Its length may vary from one network to another.
- **FCS** − It is a 2 byte or 4 bytes frame check sequence for error detection. The standard code used is CRC (cyclic redundancy code)



# Types of HDLC Frames

- There are three types of HDLC frames. The type of frame is determined by the control field of the frame −
- **I-frame** − I-frames or Information frames carry user data from the network layer. They also include flow and error control information that is piggybacked on user data. The first bit of control field of I-frame is 0.
- **S-frame** − S-frames or Supervisory frames do not contain information field. They are used for flow and error control when piggybacking is not required. The first two bits of control field of S-frame is 10.
- **U-frame** − U-frames or Un-numbered frames are used for myriad miscellaneous functions, like link management. It may contain an information field, if required. The first two bits of control field of U-frame is 11.



# Point to Point Protocol (PPP)

- Point - to - Point Protocol (PPP) is a communication protocol of the data link layer that is used to transmit multiprotocol data between two directly connected (point-to-point) computers.
- It is a byte - oriented protocol that is widely used in broadband communications having heavy loads and high speeds.
- Since it is a data link layer protocol, data is transmitted in frames. It is also known as RFC 1661.
- **Services Provided by PPP**
- The main services provided by Point - to - Point Protocol are –
  - Defining the frame format of the data to be transmitted.
  - Defining the procedure of establishing link between two points and exchange of data.
  - Stating the method of encapsulation of network layer data in the frame.
  - Stating authentication rules of the communicating devices.
  - Providing address for network communication.
  - Providing connections over multiple links.
  - Supporting a variety of network layer protocols by providing a range os services.

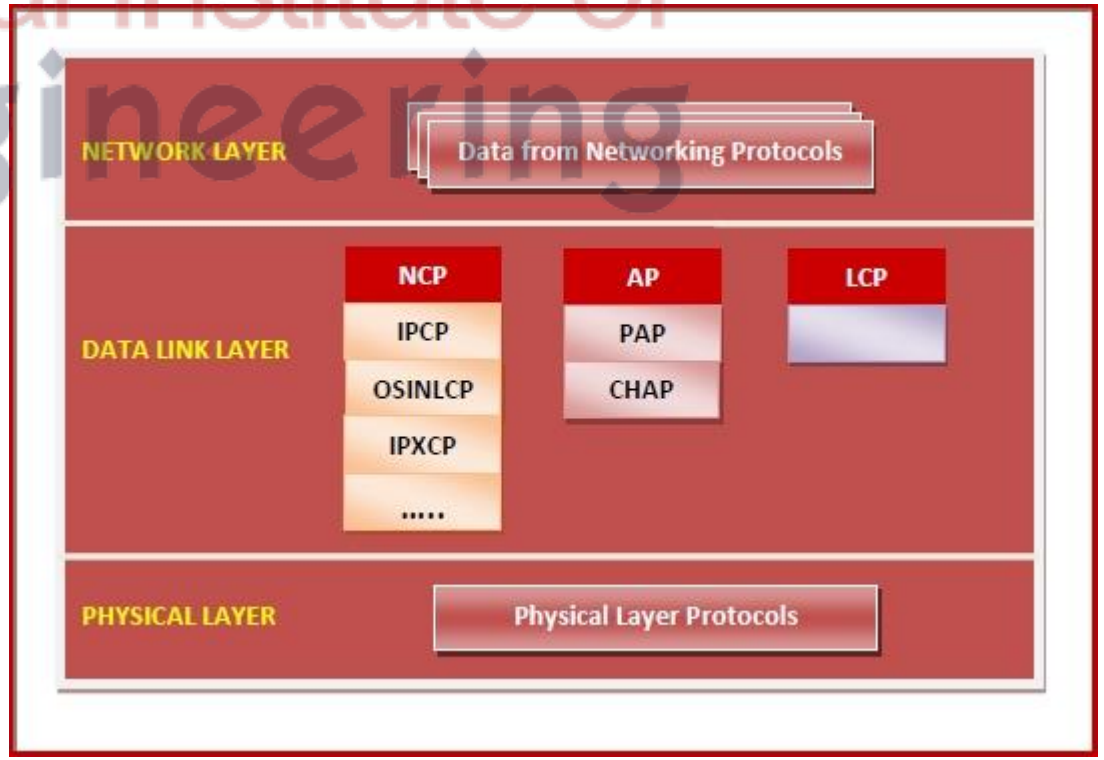
# Components of PPP

- Point - to - Point Protocol is a layered protocol having three components –
- **Encapsulation Component** – It encapsulates the datagram so that it can be transmitted over the specified physical layer.
- **Link Control Protocol (LCP)** – It is responsible for establishing, configuring, testing, maintaining and terminating links for transmission. It also imparts negotiation for set up of options and use of features by the two endpoints of the links.
- **Authentication Protocols (AP)** – These protocols authenticate endpoints for use of services. The two authentication protocols of PPP are –
  - Password Authentication Protocol (PAP)
  - Challenge Handshake Authentication Protocol (CHAP)

# Network Control Protocols (NCPs)

- These protocols are used for negotiating the parameters and facilities for the network layer. For every higher-layer protocol supported by PPP, one NCP is there. Some of the NCPs of PPP are −
- Internet Protocol Control Protocol (IPCP)
- OSI Network Layer Control Protocol (OSINLCP)
- Internetwork Packet Exchange Control Protocol (IPXCP)
- DECnet Phase IV Control Protocol (DNCP)
- NetBIOS Frames Control Protocol (NBFCP)
- IPv6 Control Protocol (IPV6CP)

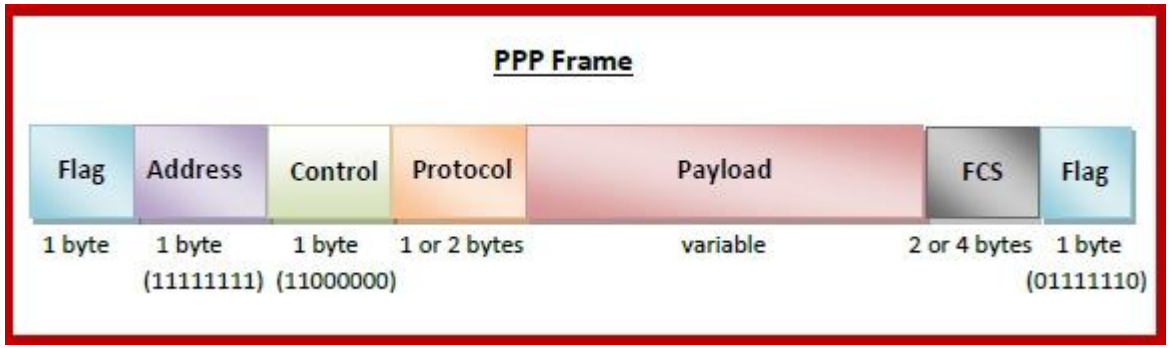




# PPP Frame

- PPP is a byte - oriented protocol where each field of the frame is composed of one or more bytes. The fields of a PPP frame are −
- **Flag** − 1 byte that marks the beginning and the end of the frame. The bit pattern of the flag is 01111110.
- **Address** − 1 byte which is set to 11111111 in case of broadcast.
- **Control** − 1 byte set to a constant value of 11000000.
- **Protocol** − 1 or 2 bytes that define the type of data contained in the payload field.
- **Payload** − This carries the data from the network layer. The maximum length of the payload field is 1500 bytes. However, this may be negotiated between the endpoints of communication.
- **FCS** − It is a 2 byte or 4 bytes frame check sequence for error detection. The standard code used is CRC (cyclic redundancy code)





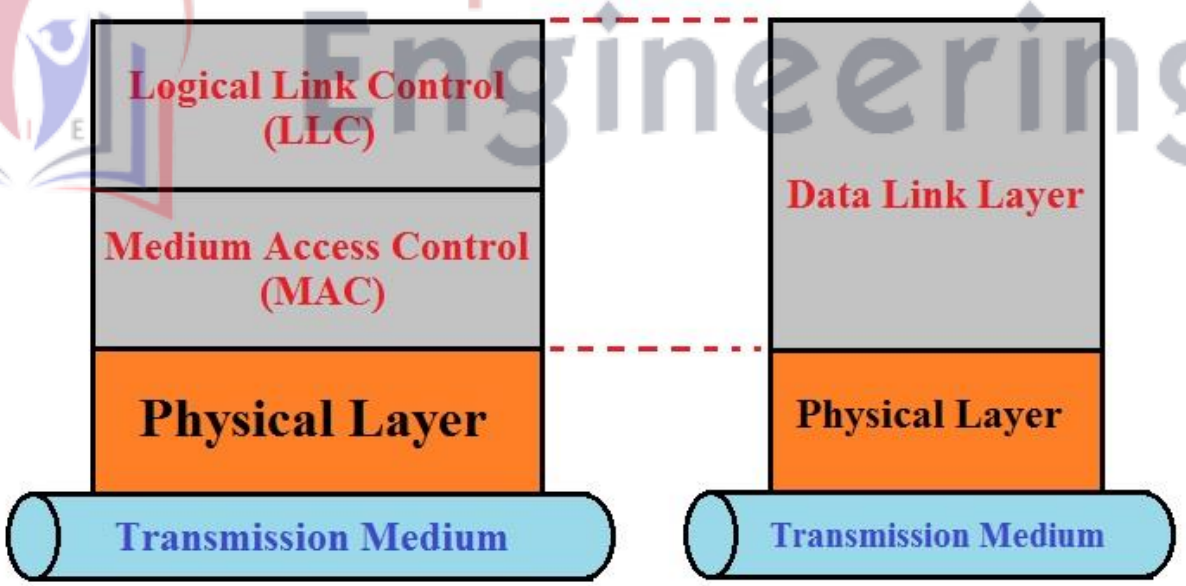
# Medium Access Control Sub-layer

- Networks can be divided into two categories :
  - Those using point to point connections and
  - Those using broadcast chnnels.
- This medium access sub-layer deals with broadcast networks and their protocols.
- In any broadcast network, the key issue is how to determine who gets to use the channel when there is competition for it. To make this point clearer, consider a conference call in which six people, on six different telephones, are all connected so that each one can hear and talk to all the others.
- It is very likely that when one of them stops speaking, two or more will start talking at once, leading to chaos.
- In a face to face meeting, chaos is avoidedby external means, for example at a meeting, people raise their hands to request permission to speak. When only a single channel is available, determining who should go next is much harder.

# Medium Access Control Sub-layer

- The protocols used to determine who goes next on a multi-access channel belong to a sub-layer of the data link layer called the MAC (medium access control) sub-layer.
- The mac sub-layer is especially important in LANs, many of which use a multi-access channel as the basis for communication.





# Channel Allocation Problem

- When there are more than one user who desire to access a shared network channel, an algorithm is deployed for channel allocation among the competing users.
- The network channel may be a single cable or optical fiber connecting multiple nodes, or a portion of the wireless spectrum.
- Channel allocation algorithms allocate the wired channels and bandwidths to the users, who may be base stations, access points or terminal equipment.
- Channel allocation is a process in which a single channel is divided and allotted to multiple users in order to carry user specific tasks. There are user's quantity may vary every time the process takes place.
- If there are N number of users and channel is divided into N equal-sized sub channels, Each user is assigned one portion. If the number of users are small and don't vary at times, than Frequency Division Multiplexing can be used as it is a simple and efficient channel bandwidth allocating technique.
- Channel allocation problem can be solved by two schemes:
  - Static Channel Allocation in LANs and MANs, and
  - Dynamic Channel Allocation.

# Static Channel Allocation

- In static channel allocation scheme, a fixed portion of the frequency channel is allotted to each user. For N competing users, the bandwidth is divided into N channels using frequency division multiplexing (FDM), and each portion is assigned to one user.
- This scheme is also referred as fixed channel allocation or fixed channel assignment.
- It is not efficient to divide into fixed number of chunks.
- In this allocation scheme, there is no interference between the users since each user is assigned a fixed channel. However, it is not suitable in case of a large number of users with variable bandwidth requirements.

# Dynamic Channel Allocation

- In dynamic channel allocation scheme, frequency bands are not permanently assigned to the users. Instead channels are allotted to users dynamically as needed, from a central pool. The allocation is done considering a number of parameters so that transmission interference is minimized.
- This allocation scheme optimizes bandwidth usage and results is faster transmissions.
- Dynamic channel allocation is further divided into centralized and distributed allocation.
- **Possible assumptions include :**

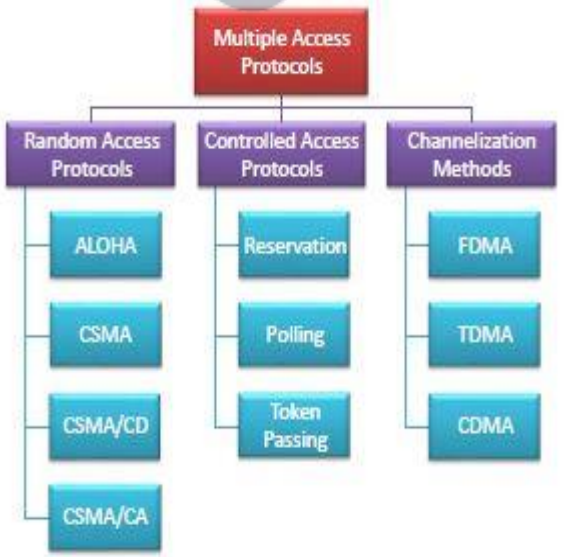
1. **Station model :-** Assumes that each of N stations independently produce frames. The probability of producing a packet in the interval IDt where I is the constant arrival rate of new frames.
2. **Single Channel Assumption :-** In this allocation all stations are equivalent and can send and receive on that channel.
3. **Collision Assumption :-** If two frames overlap in time-wise, then that's collision. Any collision is an error, and both frames must be transmitted. Collisions are only possible error.
4. **Time :- Time** can be divided into Slotted or Continuous.
5. **Stations** can sense a channel is busy before they try it.

# Media Access (Multiple Access )

- Multiple access protocols are a set of protocols operating in the Medium Access Control sublayer (MAC sublayer) of the Open Systems Interconnection (OSI) model.
- These protocols allow a number of nodes or users to access a shared network channel. Several data streams originating from several nodes are transferred through the multi-point transmission channel.
- The objectives of multiple access protocols are optimization of transmission time, minimization of collisions and avoidance of crosstalks.

## • Categories of Multiple Access Protocols

- Multiple access protocols can be broadly classified into three categories - random access protocols, controlled access protocols and channelization protocols.



# Random Access Protocols

- Random access protocols assign uniform priority to all connected nodes. Any node can send data if the transmission channel is idle. No fixed time or fixed sequence is given for data transmission.
- The four random access protocols are−
  - ALOHA
  - Carrier sense multiple access (CMSA)
  - Carrier sense multiple access with collision detection (CMSA/CD)
  - Carrier sense multiple access with collision avoidance (CMSA/CA)

## • Controlled Access Protocols

- Controlled access protocols allow only one node to send data at a given time. Before initiating transmission, a node seeks information from other nodes to determine which station has the right to send. This avoids collision of messages on the shared channel.
- The station can be assigned the right to send by the following three methods−
  - Reservation
  - Polling
  - Token Passing

# ALOHA

- Aloha is a packet switching system. The time interval required to transmit one packet is called a slot. Aloha is a random access technique.
- There are two ALOHA protocols as follows −
    - Pure ALOHA
    - Slotted ALOHA
- **Pure ALOHA**
- The mode of random access in which users can transmit at any time is called pure Aloha. This technique is explained below in a stepwise manner.
- **Step 1** − In pure ALOHA, the nodes transmit frames whenever there is data to send.
- **Step 2** − When two or more nodes transmit data simultaneously, then there is a chance of collision and the frames are destroyed.
- **Step 3** − In pure ALOHA, the sender will expect acknowledgement from the receiver.
- **Step 4** − If acknowledgement is not received within specified time, the sender node assumes that the frame has been destroyed.
- **Step 5** − If the frame is destroyed by collision the node waits for a random amount of time and sends it again. This waiting time may be random otherwise the same frames will collide multiple times.
- **Step 6** − Therefore, pure ALOHA says that when the time-out period passes, each station must wait for a random amount of time before re-sending its frame. This randomness will help avoid more collisions.

# ALOHA (Contd…)

- In PURE ALOHA, the vulnerable period is two slot times.
- Vulnerable period is the maximum interval over which two packets can overlap and destroy each other. This phenomenon is shown the below figure −
- The throughput of pure ALOHA is given by the following −
- **Probability of success ($P_{Success}$) is equal to probability of no other packet transmission occurring within the vulnerable period.**
- Therefore, Throughput(s) is defined as the successfully transmitted traffic load and 'G' is the total offered channel traffic load.
- Assume that traffic generated for transmission obeys a Poisson distribution,
- P(no other packet transmission occurs) = exp(-TG)
- Where T is vulnerable period,

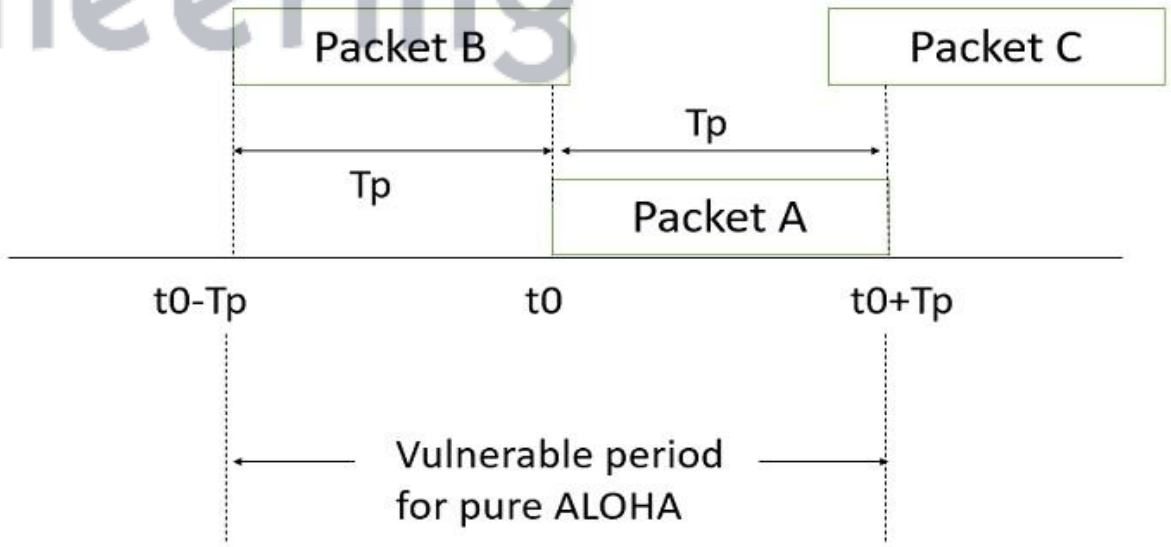
$$P_{success} = \exp(-TG)$$

$$= S/G$$

$$S = Ge^{-TG}$$

If vulnerable time is '2' $S_{max} = \frac{1}{2e}$

$$= 0.184$$

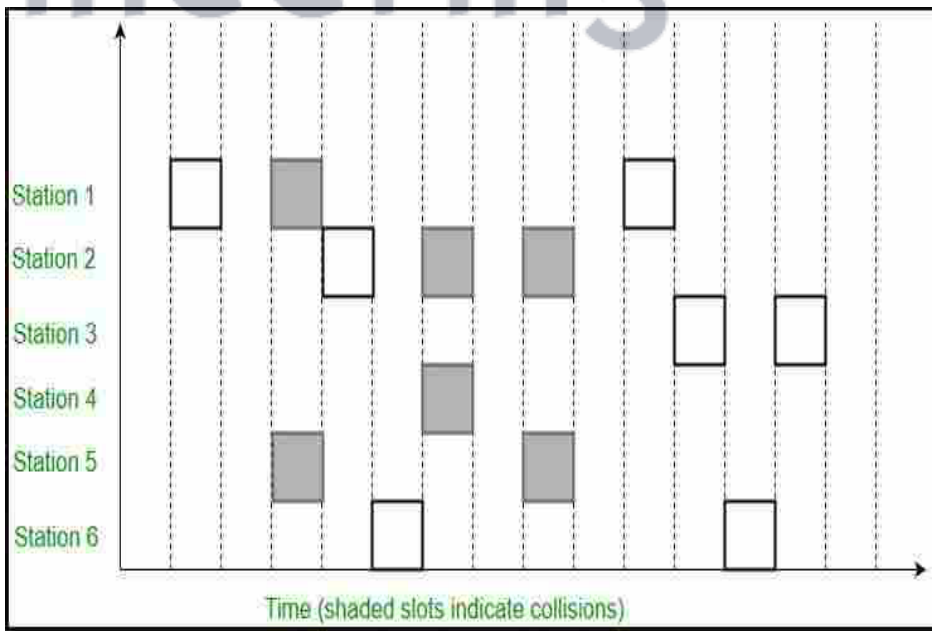



- It means, in Pure ALOHA the channel utilization is 18 percent.

# Slotted ALOHA

- This is quite similar to Pure Aloha, differing only in the way transmissions take place.
- Instead of transmitting right at demand time, the sender waits for some time.
- In slotted ALOHA, the time of the shared channel is divided into discrete intervals called *Slots*.
- The stations are eligible to send a frame only at the beginning of the slot and only one frame per slot is sent.
- If any station is not able to place the frame onto the channel at the beginning of the slot, it has to wait until the beginning of the next time slot.
- There is still a possibility of collision if two stations try to send at the beginning of the same time slot.
- But still the number of collisions that can possibly take place is reduced by a large margin and the performance becomes much well compared to Pure Aloha.

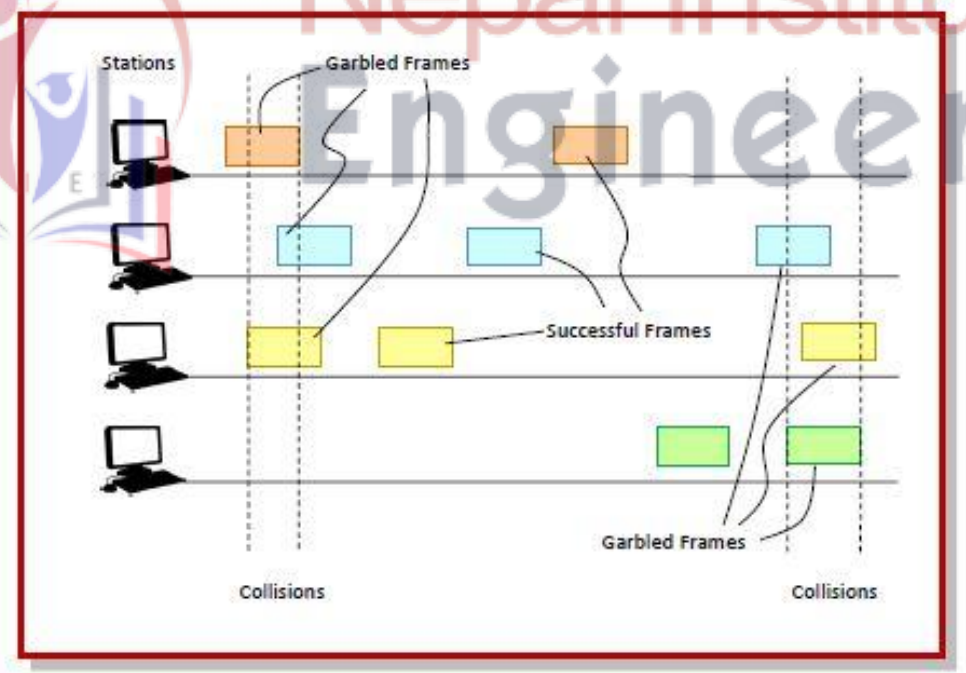




# Working Principle

- The communicating stations agree upon the slot boundaries. Any station can send only one frame at each slot. Also, the stations cannot transmit at any time whenever a frame is available. They should wait for the beginning of the next slot.
- However, there still can be collisions. If more than one frame transmits at the beginning of a slot, collisions occur. The collision duration is 1 slot. The situation is depicted in the following diagram−





# Throughput of Slotted ALOHA

- Let T be the frame time, i.e. the time required for 1 frame to be transmitted.
- Let G be the number of transmission attempts per frame time.
- The probability that $k$ frames are generated during the frame time is given by the Poisson distribution−

$$P(k) = \frac{G^k e^{-G}}{k!}$$

- In case of slotted ALOHA, the vulnerable time period for collision between two frames is equal to time duration of 1 slot, which is equal to 1 frame time, i.e. $T$.In $T$ time, average number of transmission attempts is G.
- The probability that 0 frames are initiated in the vulnerable time period will be – $P(0) = e^{-G}$
- The throughput, S is calculated as the number of transmission attempts per frame time G, multiplied by the probability of success , pP(0).

$$S = G.P(0)$$

- The maximum throughput occurs when G = 1.
- The maximum throughput is 36.8% in slotted ALOHA, which is an improvement over maximum throughput of 18.4% in pure ALOHA.

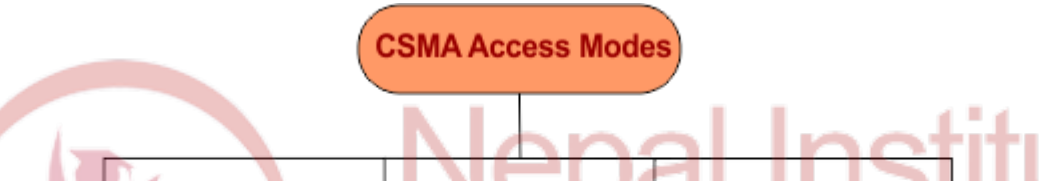
- Advantage, Disadvantage self study

# Carrier Sense Multiple Access (CSMA)

- Carrier Sense Multiple Access (CSMA) is a network protocol for carrier transmission that operates in the Medium Access Control (MAC) layer.
- It senses or listens whether the shared channel for transmission is busy or not, and transmits if the channel is not busy.
- Using CMSA protocols, more than one users or nodes send and receive data through a shared medium that may be a single cable or optical fiber connecting multiple nodes, or a portion of the wireless spectrum.
- **Working Principle**
- When a station has frames to transmit, it attempts to detect presence of the carrier signal from the other nodes connected to the shared channel. If a carrier signal is detected, it implies that a transmission is in progress. The station waits till the ongoing transmission executes to completion, and then initiates its own transmission.
- Generally, transmissions by the node are received by all other nodes connected to the channel.
- Since, the nodes detect for a transmission before sending their own frames, collision of frames is reduced.
- However, if two nodes detect an idle channel at the same time, they may simultaneously initiate transmission. This would cause the frames to garble resulting in a collision.

# CSMA Access Modes

- The versions of CSMA access modes are-



# Persistent CSMA (or 1-Persistent CSMA)

- When a station wants to send data, it listens to channel.
- If channel is busy, then it continually sense and waits for channel to be free.
- Sends the frame as soon as channel becomes idle.
- Collision may occur even if propagation delay is 0. It's because two stations may be waiting for channel to be idle and may start transmitting at exactly same time.
- If collision occurs, stations waits for random amount of time and starts over again.

# Non-Persistent CSMA

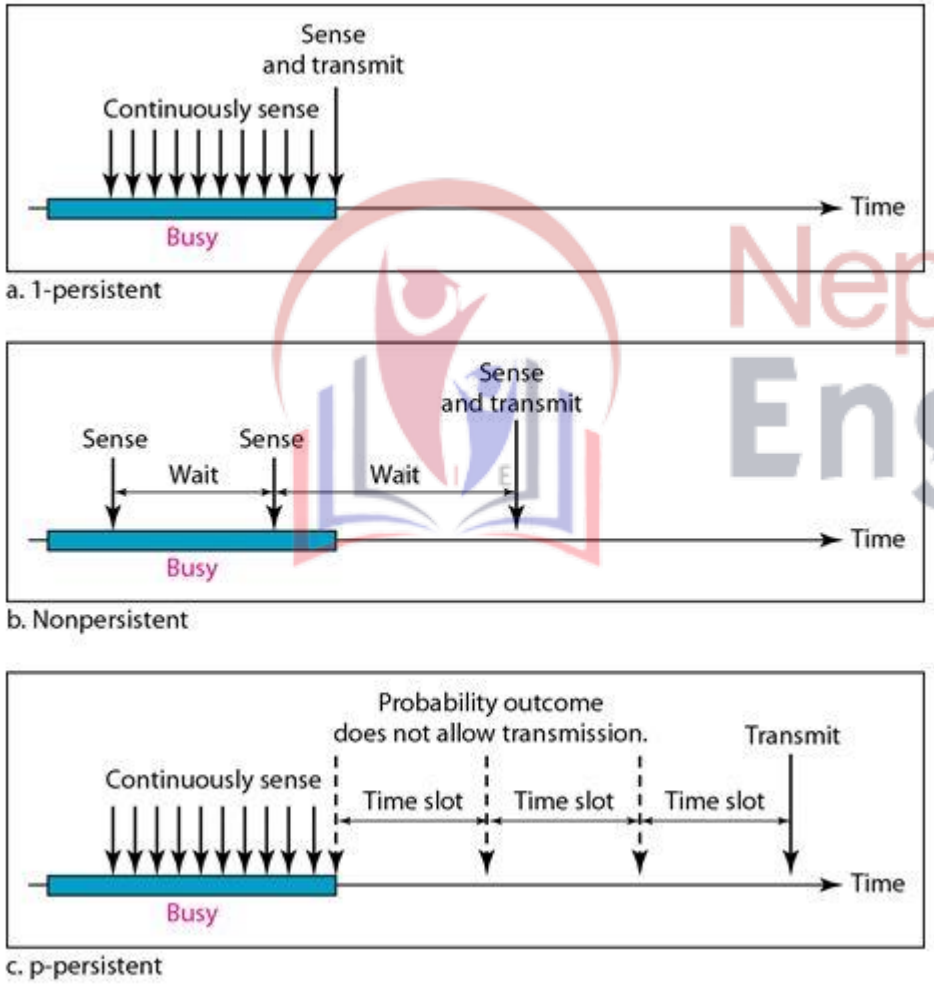
- Non-Persistent CSMA
  - Less Greedy than the persistent CSMA
  - Before Sending, station Sense the Channel.
  - If no one else is sending, it begins its transmission.
  - However, if channel is in use, it waits for random amount of time rather than continually sensing and waiting the channel to be free.
  - It leads to better channel utilization but longer delays than 1-Persistent CSMA.

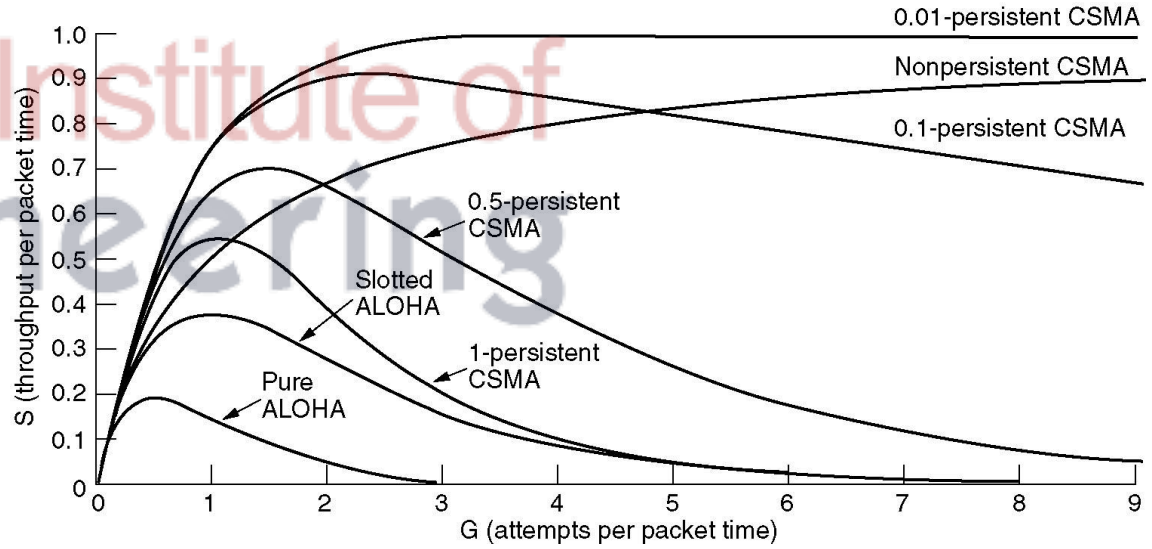
# p-Persistent CSMA

- p-Persistent CSMA
  - Applies to slotted channels
  - If channel is busy, waits for it to be idle.
  - If channel is idle, station transmits with probability p (ie. It defers untill next slot with probability q = 1-p)
  - If next slot is also idle, it transmits or defers with probabilities p and q.
  - Same process is repeated untill either frame has been transmmitted or another station starts transmission.
  - If another station begun transmitting, the unlucky station acts as if there has been a collision

    (ie. It waits for random amount of time and starts over again)
  - Better trade-off between non-persistent and 1-persistent CSMA

# Behavior analysis



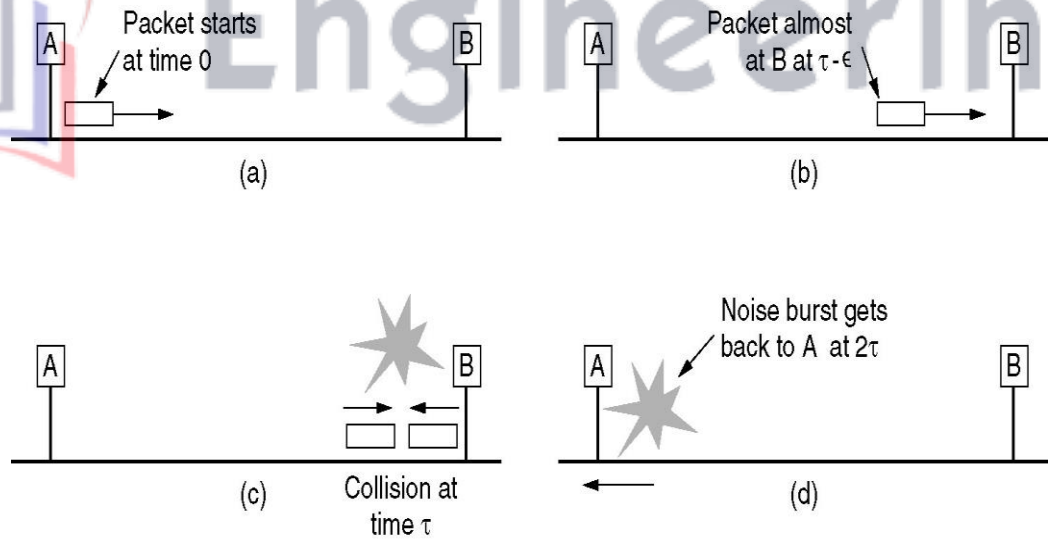




Comparison of the channel utilization versus load for various random access protocols.

# CSMA with Collision Detection (CSMA/CD)

- Widely Used on LANs in MAC Sub-layer
- Basis of Popular Ethernet LANs
- Detects Collision within short period and quickly aborts transmission in case of collision, thus reducing channel wastage.
- Detects collision by comparing transmitted and received signal.
- Easy in wired LANs but difficult to implement in wireless LANs(as in WLANs receiver shuts off while transmitting).
- Let, $\tau$ be time for signal to propagate between two farthest stations.



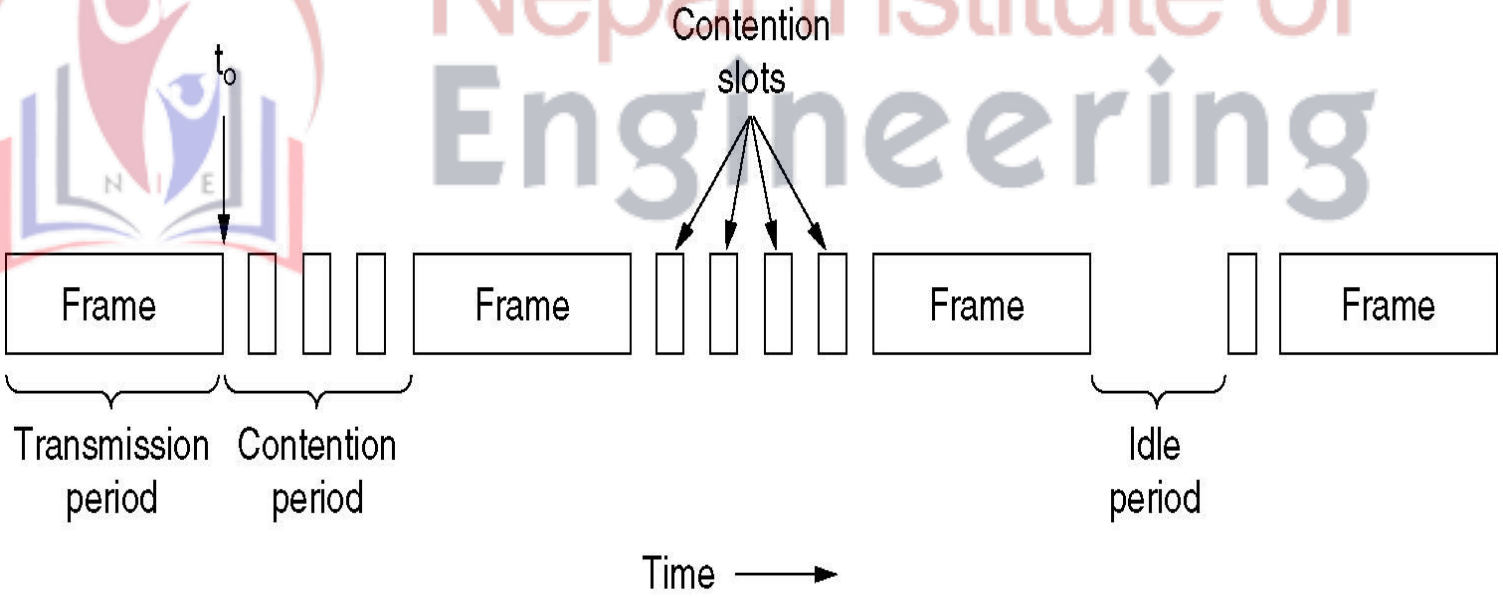



Collision Detection can take as long as $2\tau$.

# CSMA with Collision Detection (CSMA/CD)...

- CSMA/CD can be in one of the three states :
  - Transmission : Frame being transmitted
  - Contention :  Failed Transmission due to Collision
  - Idle : Free Channel

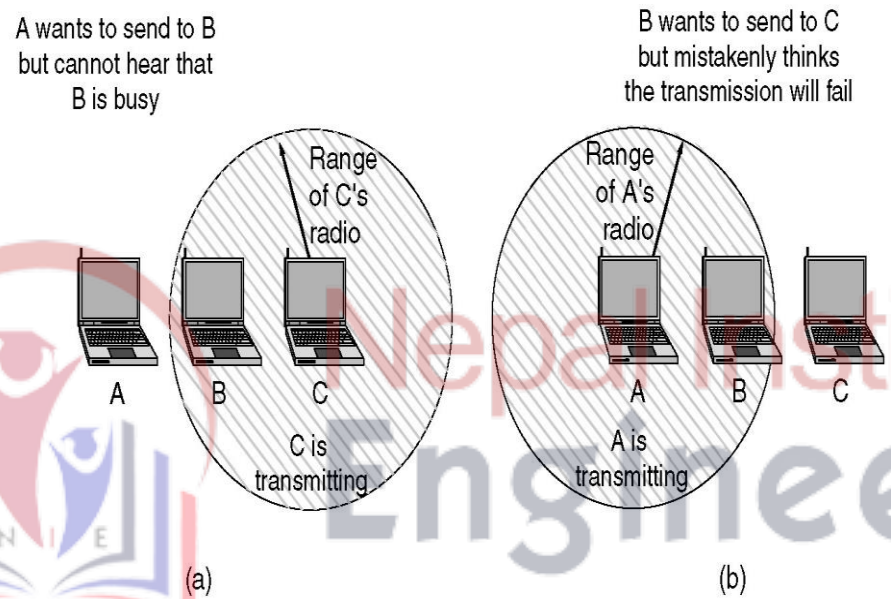




# CSMA with Collision Avoidance (CSMA/CA)

- Used for Wireless LANs where Collision Detection is not possible (as Receiver shuts off while transmitting).
- Device wanting to transmit senses the medium (Air)
- If medium is busy – defers for random time
- If medium is free for certain period (DIFS) - *transmits frame*
- Problems:
  - Hidden Station Problem
  - Exposed Station Problem

# CSMA with Collision Avoidance (CSMA/CA)




a)The hidden station problem.

b)The exposed station problem.

# Controlled Access Protocols in Computer Network

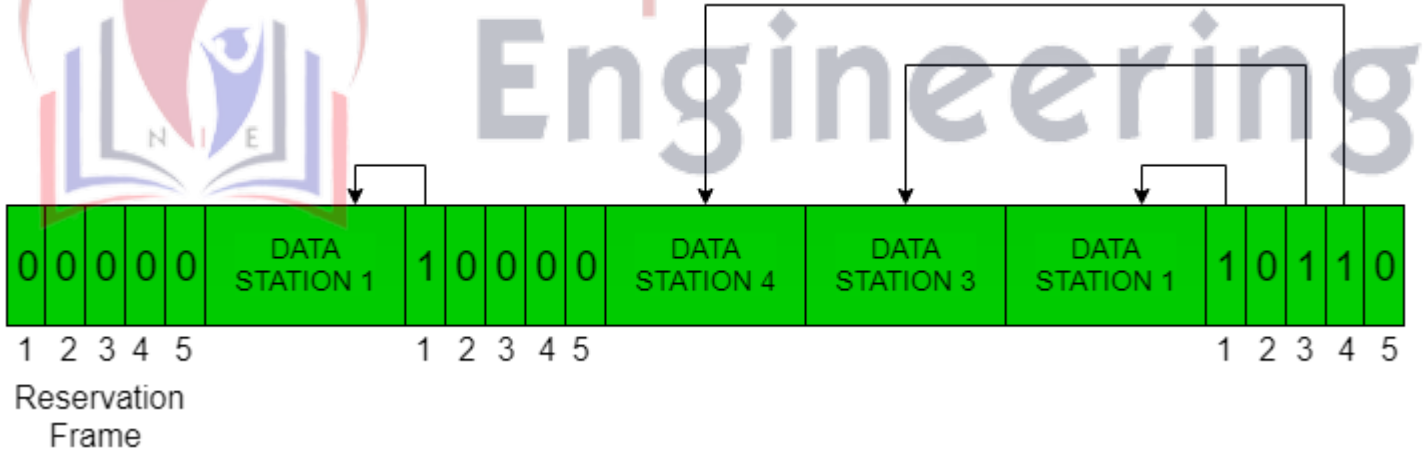
- In controlled access, the stations seek information from one another to find which station has the right to send. It allows only one node to send at a time, to avoid collision of messages on shared medium.
- The three controlled-access methods are:
    1. Reservation
    2. Polling
    3. Token Passing

# Reservation

- In the reservation method, a station needs to make a reservation before sending data.
- The time line has two kinds of periods:
  - Reservation interval of fixed time length
  - Data transmission period of variable frames.
- If there are M stations, the reservation interval is divided into M slots, and each station has one slot.
- Suppose if station 1 has a frame to send, it transmits 1 bit during the slot 1. No other station is allowed to transmit during this slot.
- In general, i $^{th}$ station may announce that it has a frame to send by inserting a 1 bit into i $^{th}$ slot. After all N slots have been checked, each station knows which stations wish to transmit.
- The stations which have reserved their slots transfer their frames in that order.
- After data transmission period, next reservation interval begins.
- Since everyone agrees on who goes next, there will never be any collisions.

# Reservation (contd…)

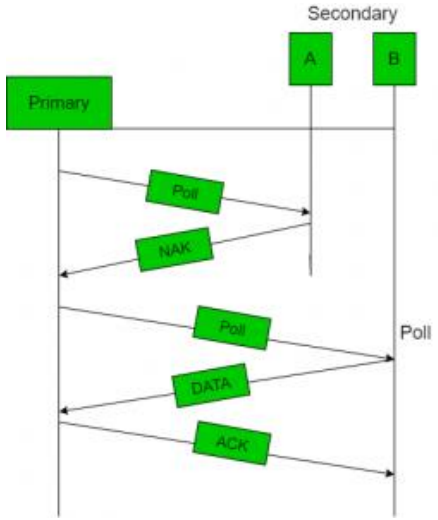
- The following figure shows a situation with five stations and a five-slot reservation frame. In the first interval, only stations 1, 3, and 4 have made reservations. In the second interval, only station 1 has made a reservation.

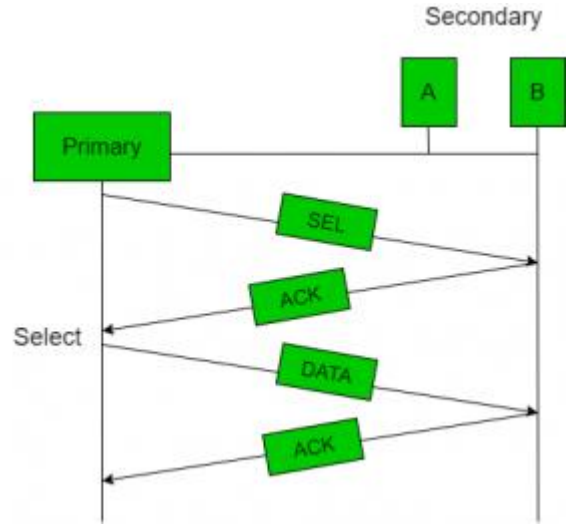


# Polling

- Polling process is similar to the roll-call performed in class. Just like the teacher, a controller sends a message to each node in turn.
- In this, one acts as a primary station(controller) and the others are secondary stations. All data exchanges must be made through the controller.
- The message sent by the controller contains the address of the node being selected for granting access.
- Although all nodes receive the message but the addressed one responds to it and sends data, if any. If there is no data, usually a “poll reject”(NAK) message is sent back.
- Problems include high overhead of the polling messages and high dependence on the reliability of the controller.
- **Efficiency** Let $T_{poll}$ be the time for polling and $T_t$ be the time required for transmission of data. Then,

$$Efficiecy = \frac{T_t}{T_t + T_{Poll}}$$



# Token Passing

- In token passing scheme, the stations are connected logically to each other in form of ring and access to stations is governed by tokens.
- A token is a special bit pattern or a small message, which circulate from one station to the next in some predefined order.
- In Token ring, token is passed from one station to another adjacent station in the ring whereas incase of Token bus, each station uses the bus to send the token to the next station in some predefined order.
- In both cases, token represents permission to send. If a station has a frame queued for transmission when it receives the token, it can send that frame before it passes the token to the next station. If it has no queued frame, it passes the token simply.
- After sending a frame, each station must wait for all N stations (including itself) to send the token to their neighbours and the other N – 1 stations to send a frame, if they have one.
- There exists problems like duplication of token or token is lost or insertion of new station, removal of a station, which need be tackled for correct and reliable operation of this scheme.

# Token Passing

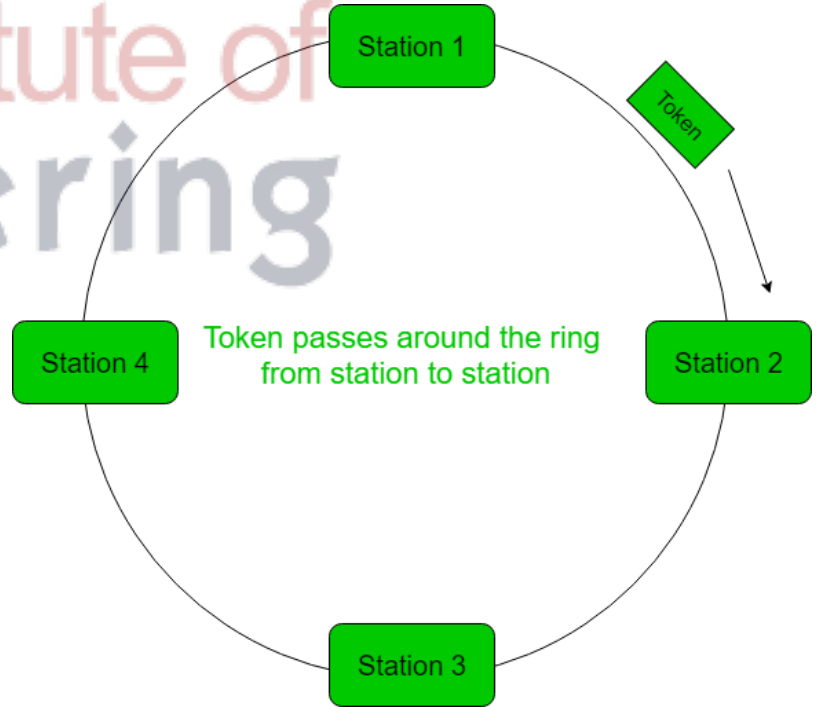
- **Performance** Performance of token ring can be concluded by 2 parameters:-
- **Delay**, which is a measure of time between when a packet is ready and when it is delivered. So, the average time (delay) required to send a token to the next station = a/N.
- **Throughput**, which is a measure of the successful traffic.
- Throughput, S = 1/(1+a/N) and

  S = 1/{a(1+1/N)} for a>1

  a= Tp/Tt

  ( Tp= propagation delay and Tt= transmission delay)





# Channelization :

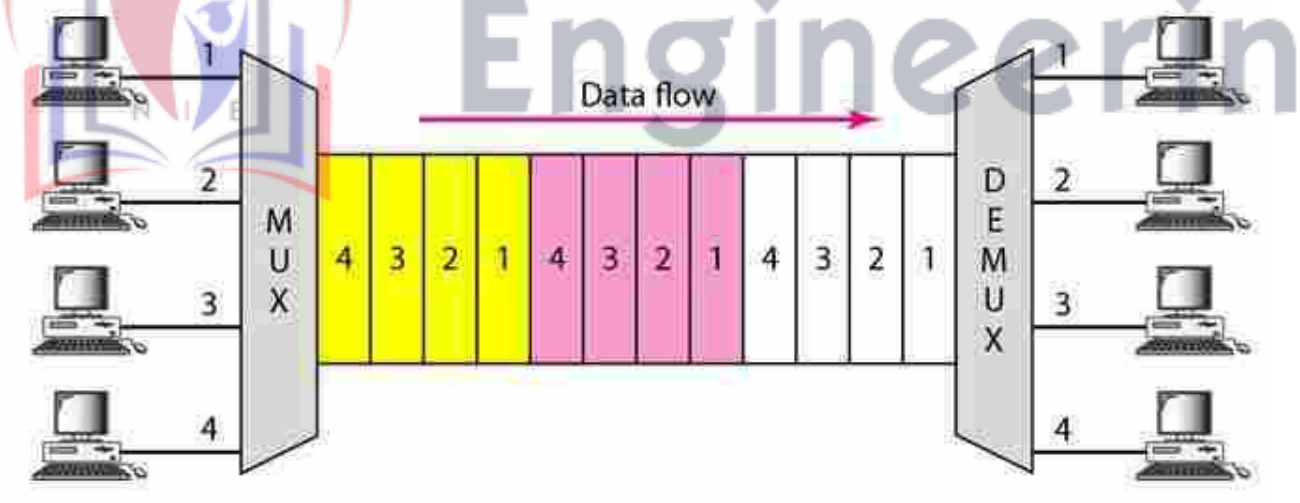
- In this, the available bandwidth of the link is shared in time, frequency and code to multiple stations to access channel simultaneously.
- **Frequency Division Multiple Access (FDMA) –** The available bandwidth is divided into equal bands so that each station can be allocated its own band. Guard bands are also added so that no to bands overlap to avoid crosstalk and noise.
- Frequency-division multiplexing (FDM) is an analog technique that can be applied when the bandwidth of a link (in hertz) is greater than the combined bandwidths of the signals to be transmitted.
- In this illustration, the transmission path is divided into three parts, each representing a channel that carries one transmission.

# Frequency Division Multiple Access (FDMA)

- **Multiplexing Process:**
  - The following figure is a conceptual illustration of the multiplexing process. Each source generates a signal of a similar frequency range.
  - Inside the multiplexer, these similar signals modulates different carrier frequencies (f1,f2 and f3).
  - The resulting modulated signals are then combined into a single composite signal that is sent out over a media link that has enough bandwidth to accommodate it.
- **Demultiplexing Process:**
  - The demultiplexer uses a series of filters to decompose the multiplexed signal into its constituent component signals.
  - The individual signals are then passed to a demodulator that separates them from their carriers and passes them to the output lines.

# Time Division Multiple Access (TDMA)

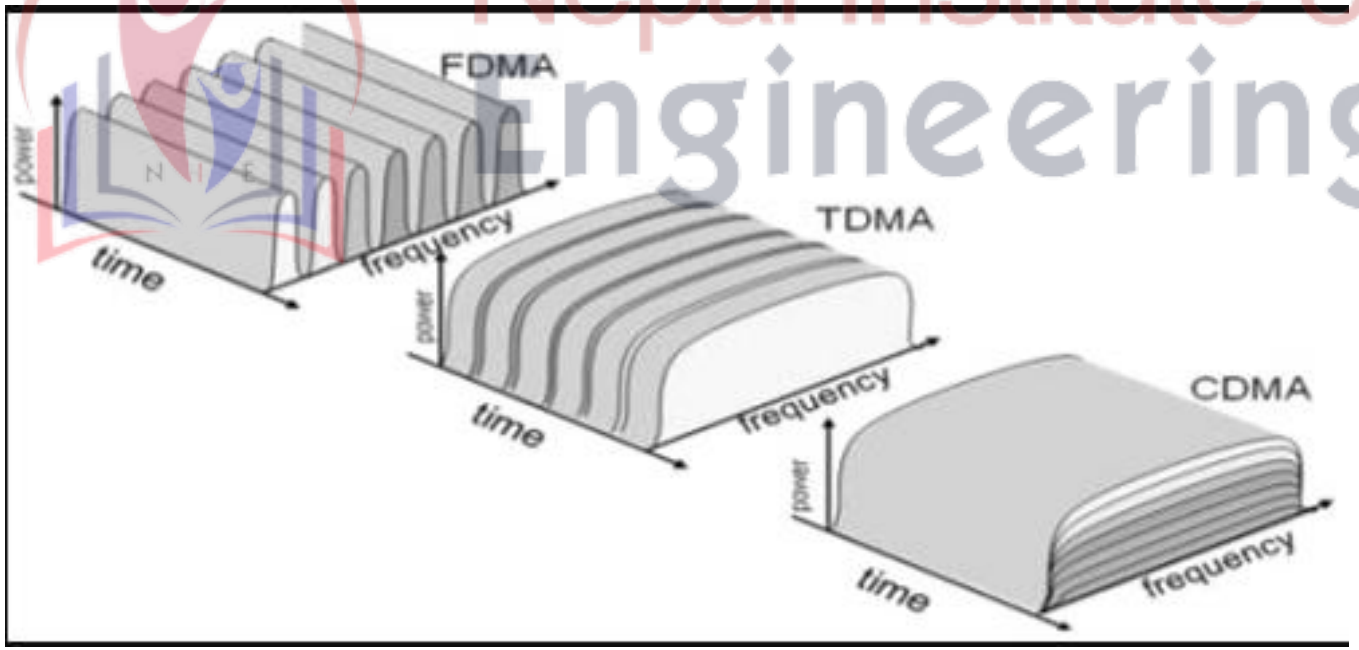
- In this, the bandwidth is shared between multiple stations. To avoid collision time is divided into slots and stations are allotted these slots to transmit data.
- However there is a overhead of synchronization as each station needs to know its time slot. This is resolved by adding synchronization bits to each slot.
- Another issue with TDMA is propagation delay which is resolved by addition of guard bands.



# Code Division multiple Access (CDMA)

- One channel carries all transmissions simultaneously. There is neither division of bandwidth nor division of time.
- For example, if there are many people in a room all speaking at the same time, then also perfect reception of data is possible if only two person speak the same language. Similarly data from different stations can be transmitted simultaneously in different code languages.





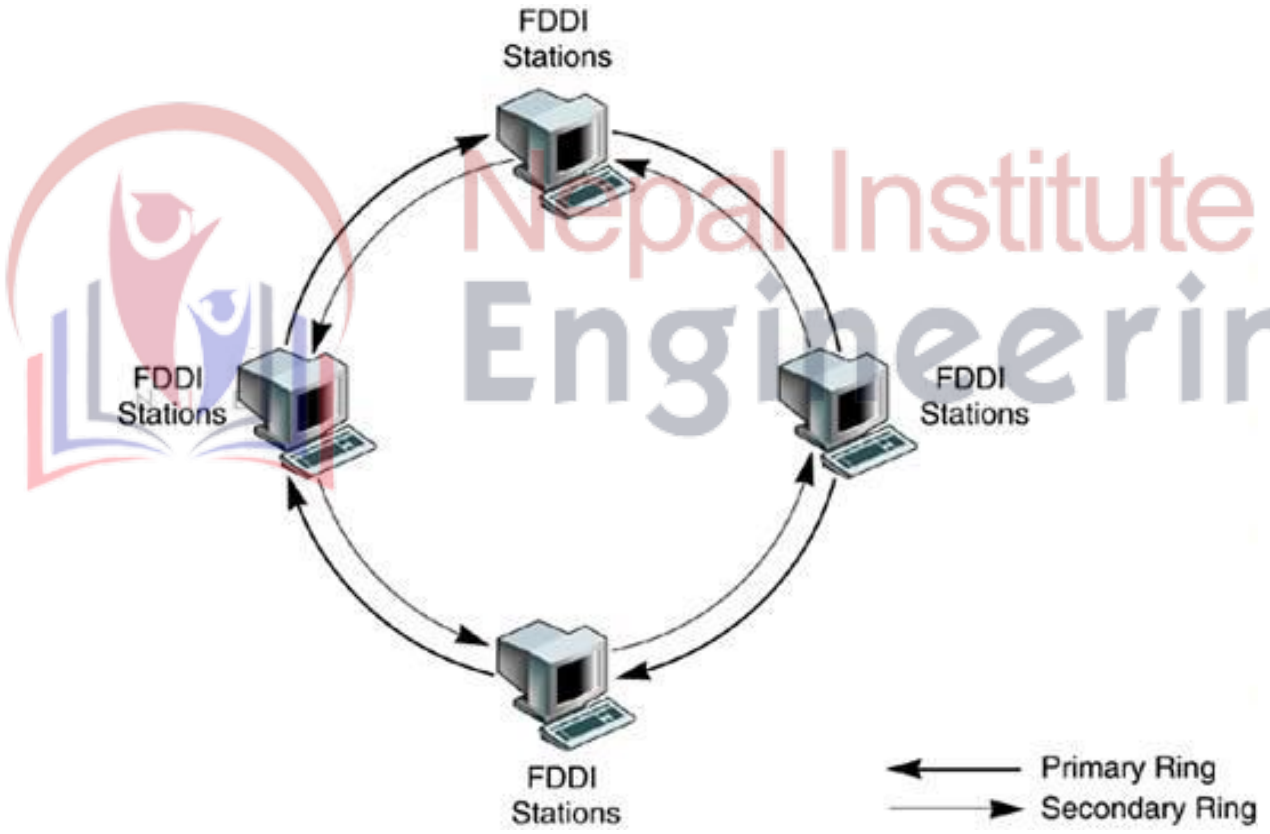
# IEEE LAN standards

- IEEE 802.3 Ethernet (CSMA/CD)
- IEEE 802.4 Token Bus
- IEEE 802.5 Token Ring
- IEEE 802.6 Metropolitan Area Networks
- IEEE 802.7 Broadband LANs
- IEEE 802.8 Fiber Optic LANs
- IEEE 802.9 Integrated Data and Voice Networks
- IEEE 802.11 Wireless Networks
- IEEE 802.14 Cable TV

# Fiber Distributed Data Interface (FDDI)

- FDDI specifies a **100-MBPS token passing**, **dual ring** LAN **using fiber-optic cable.**
- FDDI is frequently used as high-speed backbone technology because of its support for **high bandwidth and greater distances** than copper.
- FDDI supports extensions **up to 100Km**.
- A total of **1000 stations** can be connected with **a maximum separation of 2 km**.
- **Node-to-node distance** : **2 km** using multi-mode and **40 km** using single mode fiber.
- FDDI uses **dual counter-rotating** ring. Data normally travels on the primary ring. Stations can be attached to the primary ring as **single attachment stations** (SAS) or both rings as **dual attachment stations** (DAS).
- An **important feature** of FDDI is **its ability to handle a breaks** in the network by forming a single temporary ring out of the pieces of the primary and secondary rings.
- Once the stations **detect the break**, **traffic is rerouted through a new ring** formed out of the parts of the primary and secondary rings not affected by the break.

# FDDI's dual-ring environment

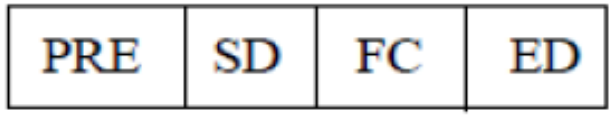


# Fiber Distributed Data Interface (FDDI)

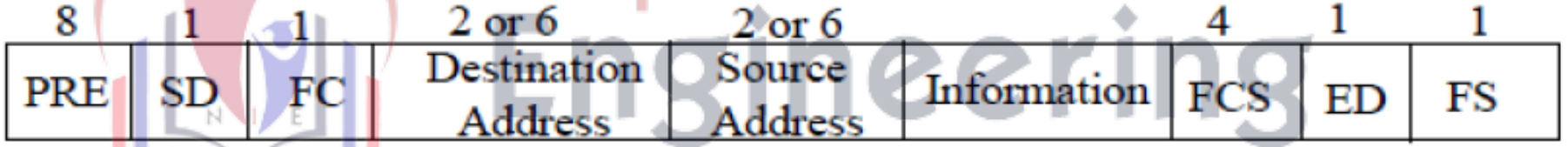
- **Media Access Control**
  - FDDI uses a token passing system. Computers wanting to send pckets wait to receive a token before transmitting.
  - Multiple packets can be attached to the token as it moves around the network.
  - When a station receives the token, it looks for attached packets addressed to it and removes them from the incoming packet.
  - If the station wants to send a packet it attaches it to the token and sends the token with its attached packets to the next station.
  - The controlled access technique provides a higher performance level at high traffic levels compared to a contention-based technique like Ethernet.

# FDDI Frame structure

Token Frame Format




Data Frame Format




Preamble

Frame Control CLFFZZZZ

C = Synch/Asynch
L = Address length (16 or 48 bits)
FF = LLC/MAC control/reserved frame type

# FDDI: Frame Layout

- **FDDI** frame can be as long as 4500 bytes.
- **Preamble** : Unique sequence that prepares each station for an upcoming frame.
- **Frame Control** : Indicates size of address field and whether the frame contains synchronous or asynchronous data, among other control information.
- **Destination Address**: contains unicast, multicast or broadcast address.
- **Source Address** : 6 byte address source address.
- **Data** : contains either information destined for upper layers or control information.
- **Frame Check sequence** : for error detection.
- **End Delimiter** : End of frame.
- **Frame Status** : Allows the source station to determine whether an error occurred; identifies whether the frame was recognized and copied by a receiving station.

# BLUETOOTH

- Bluetooth is a wireless technology standard for exchanging data over short distances (using short-wavelength radio transmissions in the band from 2400–2480 MHz) from fixed and mobile devices, creating personal area networks (PANs) with high levels of security.
- The Bluetooth technology relies on short-range radio frequency, and any device that incorporates the technology can communicate as long as it is within the required distance.
- The technology is often used to allow two different types of devices to communicate with each other.
- For example, you may be able to operate your computer with a wireless keyboard, use a wireless headset to talk on your mobile phone, or add an appointment to your friend's PDA calendar from your own PDA.
- Bluetooth can connect up to eight devices simultaneously.
- With all of those devices in the same 10-meter (32-foot) radius, you might think they'd interfere with one another, but it's unlikely.
- Bluetooth uses a technique called spread-spectrum frequency hopping that makes it rare for more than one device to be transmitting on the same frequency at the same time.

# Bluetoth (contd…)

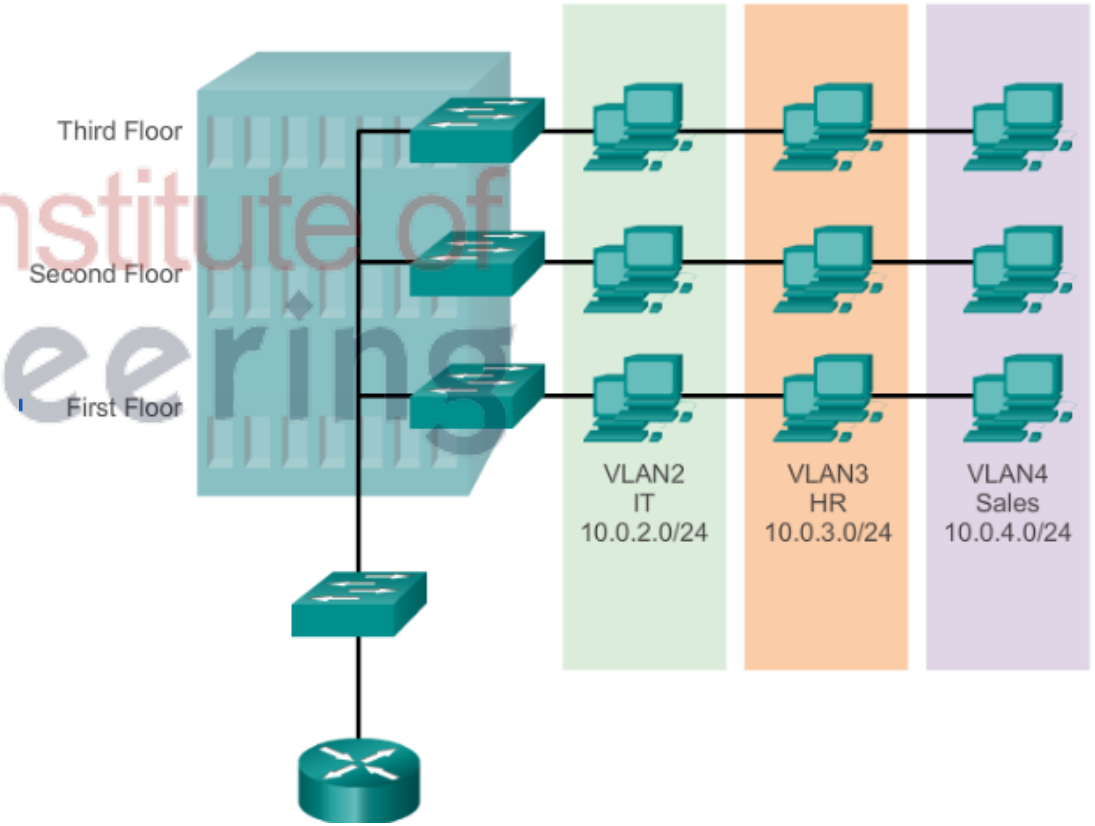
- In this technique, a device will use 79 individual, randomly chosen frequencies within a designated range, changing from one to another on a regular basis.
- In the case of Bluetooth, the transmitters change frequencies 1,600 times every second, meaning that more devices can make full use of a limited slice of the radio spectrum.
- Since every Bluetooth transmitter uses spread-spectrum transmitting automatically, it’s unlikely that two transmitters will be on the same frequency at the same time.
- This same technique minimizes the risk that portable phones or other devices will disrupt Bluetooth devices, since any interference on a particular frequency will last only a tiny fraction of a second.
- When Bluetooth-capable devices come within range of one another, an electronic conversation takes place to determine whether they have data to share or whether one needs to control the other.
- The user doesn't have to press a button or give a command -- the electronic conversation happens automatically.
- Once the conversation has occurred, the devices -- whether they're part of a computer system or a stereo -- form a network.
- Bluetooth systems create a personal-area network (PAN), or **piconet**, that may fill a room or may encompass no more distance than that between the cell phone on a belt-clip and the headset on your head.
- Once a piconet is established, the members randomly hop frequencies in unison so they stay in touch with one another and avoid other piconets that may be operating in the same room.

# MAC Address

- Media Access Control – MAC
- A unique 48-bit address assigned to each network card
- IEEE Standard also allow 16 bit MAC address but are rarely used.
- Encoded into the firmware of cards during manufacture.
- All 48 bits set to 1 indicates the broadcast address. A broadcasted frame is received by all devices in network.
- Eg MAC Address : 2A-4B-B3-45-C3-B4

# VLAN Definitions :-

- A VLAN is a logical partition of a Layer 2 network.
- Multiple partitions can be created, allowing for multiple VLANs to co-exist.
- Each VLAN is a broadcast domain, usually with its own IP network.
- VLANs are mutually isolated and packets can only pass between them via a router.
- The partitioning of the Layer 2 network takes place inside a Layer 2 device, usually via a switch.
- The hosts grouped within a VLAN are unaware of the VLAN's existence.



# Benefits of VLANs

- Security
- Cost reduction
- Better Performance
- Shrink Broadcast domains
- Improved IT staff efficiency
- Simpler project and application management.

# Types of VLANs

- Data VLAN
- Default VLAN
- Native VLAN
- Management VLAN.

# VLAN Types :-

- **Static VLAN**
  - Static vlan are called port-based and port-centric membership VLANs.
  - Ports on a switch are manually assigned to a VLAN.
  - This is the most common method of assigning ports to VLANs.
  - As a device enters the network, it automatically assumes the VLAN membership of the port to which it is attached.
- **Dynamic VLAN**
  - Dynamic VLAN allow membership based on the MAC address of the device connected to the switch port.
  - As a device enters the network, it queries a database within the switch for VLAN membership.
  - Membership is configured using a special server called a VLAN membership Policy Server (VMPS).

# THANK YOU

???