UNIT – 2 : DATA LINK LAYER

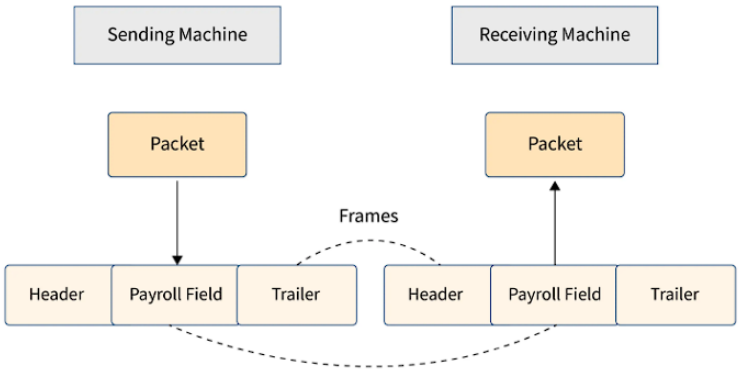
Introduction:

- The data link layer is the second layer of the seven-layer OSI model of computer networking. This is the protocol layer that transports data between network nodes in a wide area network (WAN) or nodes in the same local area network (LAN).
- The data link layer is one of the most complex layers, with numerous liabilities and functionalities. The data link layer conceals the underlying hardware features and represents itself as the communication medium to the upper layer.
- The data link layer connects two hosts that are in some ways directly connected. This direct link could be either point-to-point or broadcast.
- The data link layer is accountable for transforming data streams to signals bit by bit and transmitting them across the underlying hardware. At the receiving end, the data link layer collects data from hardware in the form of electrical signals, assembles it into a recognizable frame format, and passes it to the top layer.
- The data link layer is also responsible for delivering frames between devices on the same LAN. The primary function of the Data Link Layer is to transmit the datagram via an individual link. It performs this task by breaking up the input data into data frames (usually a few hundred or a few thousand bytes) and broadcasting the frames sequentially. If the service is reliable, the receiver acknowledges receipt of each frame by sending back an acknowledgment frame.
- Ethernet for local area networks (multi-node), Point-to-Point Protocol (PPP), HDLC, and ADCCP for point-to-point (dual-node) connections are examples of data link protocols.

Functions of Data Link Layer:

1) Framing and Link Access:

- The data-link layer takes packets from the network layer and encapsulates these packets into frames. A frame is a collection of a data field in which a network layer datagram is put along with a number of data fields. It defines the frame's structure and the channel access protocol that will be used to transmit the frame over the link. Then it sends each frame bit by bit on the hardware.
- Frames are the streams of bits received from the network layer that are converted into manageable data units. The Data Link Layer divides the bit stream. At the receiver's end, the data link layer collects signals from hardware and assembles them into frames.



2) Reliable Delivery:

The Data Link Layer provides a guaranteed delivery service, i.e., it transmits network layer datagrams without error. Transmissions and acknowledgments are used to provide a reliable delivery service. A data link layer typically offers a reliable delivery service over links since they have high error rates and can be repaired locally, rather than causing the data to be retransmitted.

3) Flow Control:

- Stations on the same link may have varying speeds or capacities. The data-link layer ensures flow control, allowing two machines to transmit data simultaneously.
- A receiving node can receive frames more quickly than it can process them. Without flow control, the receiver's buffer can overflow, which will result in frame loss. To address this issue, the data connection layer employs flow control to keep the sending node on one side of the network from overwhelming the receiving node.

4) Error Control:

Signals may experience problems during the transition, and the bits may get inverted. These errors are recognized and attempted to be recovered by the data link layer in order to retrieve actual data bits.

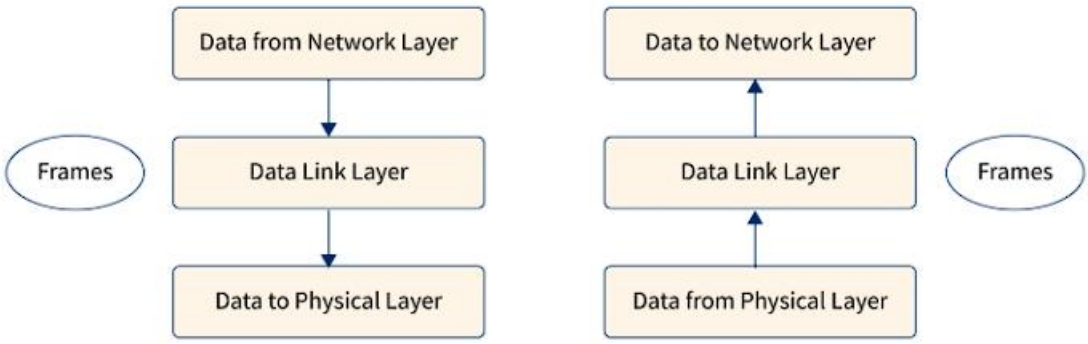
- **Error detection:** Signal attenuation and noise can both generate errors. The Data Link Layer protocol provides a mechanism for detecting one or more errors. This is accomplished by adding error detection bits at the end of each frame, which allows the receiving node to do an error check.
- **Error correction:** This is similar to error detection, in which the receiving node not only detects errors but also determines where the faults occurred in the frame.

5) Physical Addressing:

- The data-link layer provides a mechanism for layer-2 hardware addressing. If the frames are to be sent to multiple systems on the network, the Data Link layer adds a header to the frame to identify the physical address of the sender or receiver.
- Hardware addresses are expected to be unique on the link. It is encoded into devices throughout the manufacturing process.

6) Multi-Access:

When a host on the shared link tries to send data, there is a significant chance of a collision. The data-link layer provides mechanisms such as CSMA/CD that enable different systems to access shared media.



Sub-layers of the Data Link Layer:

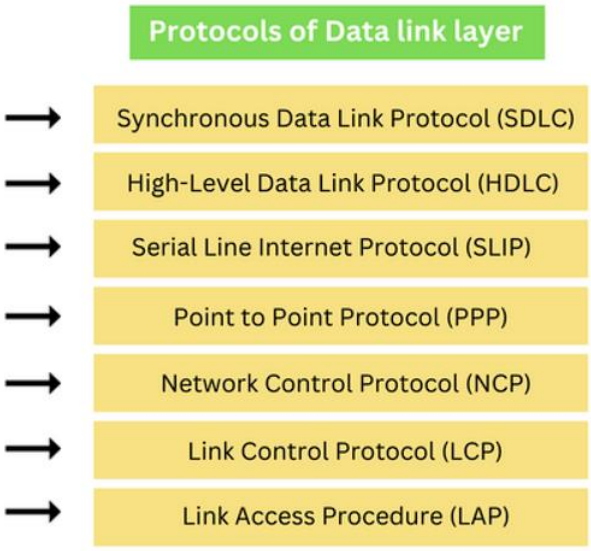
The data link layer is further divided into two sub-layers, which are as follows:

1. **Logical Link Control (LLC):** This sublayer of the data link layer deals with multiplexing, the flow of data among applications and other services, and LLC is responsible for providing error messages and acknowledgments as well.

2. **Media Access Control (MAC):** MAC sublayer manages the device's interaction, responsible for addressing frames, and also controls physical media access.

   The data link layer receives the information in the form of packets from the Network layer, it divides packets into frames and sends those frames bit-by-bit to the underlying physical layer.

Protocols of Data link layer:



- ***Synchronous Data Link Protocol (SDLC):*** It is the first bit-oriented protocol and is widely used. It is a subset of the High-Level Data Link Protocol. IBM developed this protocol in 1975. It manages synchronous serially transmitted bits over a data link layer.
- ***High-Level Data Link Protocol (HDLC):*** It is a bit-oriented protocol for conveying data on point-to-multipoint and point-to-point links. The International Organization for Standardization (ISO) developed this protocol in 1979. It is based on Synchronous Data Link Protocol. It provides connectionless and connection-oriented services. It provides two transmission modes: Asynchronous Balanced Mode (ABM) and Normal Feedback Mode (NRM).
- ***Serial Line Internet Protocol (SLIP):*** It is a simple internet protocol through which the user is allowed to access the internet with the help of a computer modem. Rick Adams developed this protocol in 1984. It works with TCP/IP for communication over the router and serial port.
- ***Point to Point Protocol (PPP):*** It is a character-oriented or byte-oriented protocol. PPP is a WAN protocol that runs over an Internet link. It is used in broadband communication. It is used to transmit multiprotocol data between point-to-point devices. It provides transmission encryption, loop connection authentication, and compression of data.
- ***Network Control Protocol (NCP):*** This layer was implemented by ARPANET. It allows transferring data between two devices. It is a part of the point-to-point protocol. This network layer will carry the data packets from the origin to the goal.
- ***Link Control Protocol (LCP):*** This layer is also a component of the point-to-point protocol. It is mainly used for establishing and maintaining the link before sending data.

- ***Link Access Procedure (LAP):*** It is derived from the high-level data link protocol. It is used for framing and data transmission over point-to-point links. It has several Link Access Protocols, such as Multilink Procedure (MLP), Link Access Procedure for Modems (LAPM), Link Access Procedure for Half-Duplex (LAPX), and Link Access Procedure for Frame Relay (LAPF).

Design Issues with Data Link Layer:

1. Network Layer Service Agreement:

The primary goal of this service is to provide services to the network layer. The main aim of the data link layer is to transmit data from the network layer on the source machine to the layer on the destination machine. The Data Link Control Protocol is used to communicate between the two data levels.

The Data Link layer provides the following essential services to the Network layer:

- **Unacknowledged connectionless services:** This is a connectionless service in which the sender transmits a message, and the receiver receives it without acknowledgment.
- **Acknowledged connectionless service:** The sender transmits the message to the receiver, and the receiver acknowledges receipt of the message to the sender using connectionless services.
- **Acknowledged-oriented service:** In this service, both the sender and the receiver use connection-oriented services, and communication between the two nodes is based on acknowledged base communication.

2. Framing:

The data is sent to the destination machine in the form of frames from the source machine. The starting and ending points of the frame should be highlighted so that the destination machine can clearly recognize the frame.

The data link layer divides the bitstream into layers and computes the checksum for each. The checksum is enumerated at the destination layer. **Framing** is the process of breaking up a bitstream by inserting spaces and time gaps.

Counting on time and marking the beginning and end of each frame is challenging and dangerous. Simple techniques used in framing are:

- Character Count
- Starting and ending character with character filling
- Starting and ending flags with little fillings.

3. **Flow Control:**

- Flow control is used to stop data flow at the receiver. The frames will be transferred to the receiver extremely fast by the transmitter. However, the sender operates on a lightly loaded system, and the receiver runs on a substantially loaded machine; therefore, the receiver will not be able to take them as rapidly as the sender sends them.
- It makes no difference if the transmission is flawless at some moment. The receiver will be unable to control the arriving frames.
- There is a mechanism that requests the transmitter to block the wrong signals in order to stop the broadcast.

4. **Error Control:**

- It is done to ensure that no frames are copied and that the frames arrive safely at their destination. Moreover, positive and negative acceptance of incoming frames is sent.
- As a result, if the sender receives positive acceptance, the frame appears safely, whereas a negative appearance indicates something is wrong with the frame and will be retransferred.
- The timer is set at both the receiver and sender ends. Additionally, the outgoing transmission is assigned a sequence number. So, the receiver will quickly recognize that it is a retransmitted frame. It is one of the main tasks of the data link layer.

5. **Physical Address of Frames:**

The data link layer appends a header to the frame that describes the physical address of the sender or receiver.

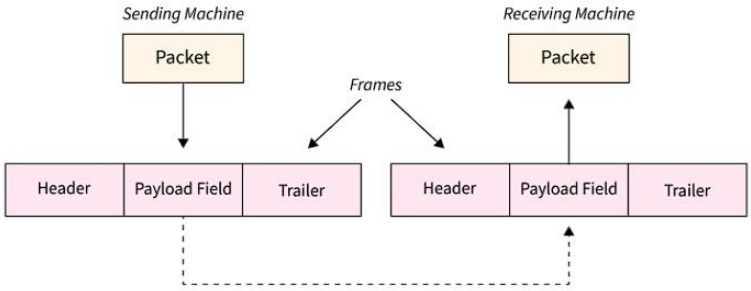
Framing:

Frames are the units of digital transmission, notably in Framing in Computer networks and telecommunications. Frames are akin to the packets of energy known as photons in the case of sunshine energy. The frame is unendingly utilized in the Time Division Multiplexing method.

Framing may be a point-to-point affiliation between 2 computers or devices consisting of a wire within which knowledge is transmitted as a stream of bits. However, these bits should be framed into discernible blocks of data. Framing may be an operation of the info link layer. It provides the simplest way for a sender to transmit a collection of bits that are important to the receiver. Ethernet, token ring, frame relay, and alternative electrical circuit layer technologies have their own frame structures. Frames have headers that contain data like error-checking codes.

Part of a Frame:

- **Header:** It consists of the frame's source and destination address. A frame header holds the destination address, the supply address, and 3 management fields kind, seq, and ack helping the subsequent purposes:
- **kind:** This field expresses whether the frame could be an information frame, or it's used for management functions like error and flow management or link management, etc.
- **Seq:** This holds the sequence variation of the frame for transcription of out–of–sequence frames and causing acknowledgments by the receiver.
- **Ack:** This contains the acknowledgment variety of some frames, significantly once piggybacking is employed.
- **Payload:** It contains the message to be delivered. It contains the particular message or data that the sender desires to transmit to the destination machine. It contains the particular message or data that the sender desires to transmit to the destination machine
- **Trailer:** It contains the error detection and correction bits.
- **Flag:** It contains the points to the starting and the ending of the frame.



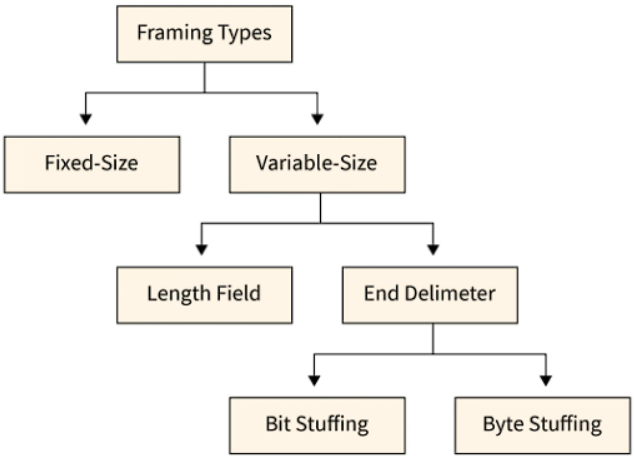
Problems in Framing:

- **Detecting start of the frame:** When a frame is transmitted, every station must be able to detect it. Station detects frames by looking out for a special sequence of bits that marks the beginning of the frame i.e. SFD (Starting Frame Delimiter).
- **How does the station detect a frame:** Every station listens to link for SFD pattern through a sequential circuit. If SFD is detected, sequential circuit alerts station. Station checks destination address to accept or reject frame.
- **Detecting end of frame:** When to stop reading the frame.
- **Handling errors:** Framing errors may occur due to noise or other transmission errors, which can cause a station to misinterpret the frame. Therefore, error detection and correction mechanisms, such as cyclic redundancy check (CRC), are used to ensure the integrity of the frame.

- **Framing overhead:** Every frame has a header and a trailer that contains control information such as source and destination address, error detection code, and other protocol-related information. This overhead reduces the available bandwidth for data transmission, especially for small-sized frames.
- **Framing incompatibility:** Different networking devices and protocols may use different framing methods, which can lead to framing incompatibility issues. For example, if a device using one framing method sends data to a device using a different framing method, the receiving device may not be able to correctly interpret the frame.
- **Framing synchronization:** Stations must be synchronized with each other to avoid collisions and ensure reliable communication. Synchronization requires that all stations agree on the frame boundaries and timing, which can be challenging in complex networks with many devices and varying traffic loads.
- **Framing efficiency:** Framing should be designed to minimize the amount of data overhead while maximizing the available bandwidth for data transmission. Inefficient framing methods can lead to lower network performance and higher latency.

Types of Framing:

There are two types of framing in the data link layer. The frame can be of fastened or variable size. founded on the size, the following are the types of framing in data link layers in computer networks –



1. Fixed Size Framing:

In this Fixed-size framing, the size of the frame is always fixed that's why frame length acts as the delimiter of the frame. And it also doesn't require additional boundaries to identify the start and end of the frame. For example- This kind of framing in the Data Link Layer is used in ATMs and wide area networks(WAN). They use frames of fastened size known as cells.

2. Variable Size Framing:

The size of the frame is variable during this form of framing. In variable-size framing, we are in need of a way to outline the tip of the frame and also the starting of the succeeding frame. This can be utilized in local area networks(LAN).

There are 2 different methods to define the frame boundaries, such as length field and finish decimeters.

Length field–To confirm the length of the field, a length field is used. It is utilized in Ethernet (1EEE 802.3).

End Delimeter–To confirm the size of the frame, a pattern is worn as a delimiter. This methodology is worn in the token ring. In short, it is referred to as ED. Two different methods are used to avoid this condition if the pattern happens within the message. Character Oriented Approach Bit Oriented Approach

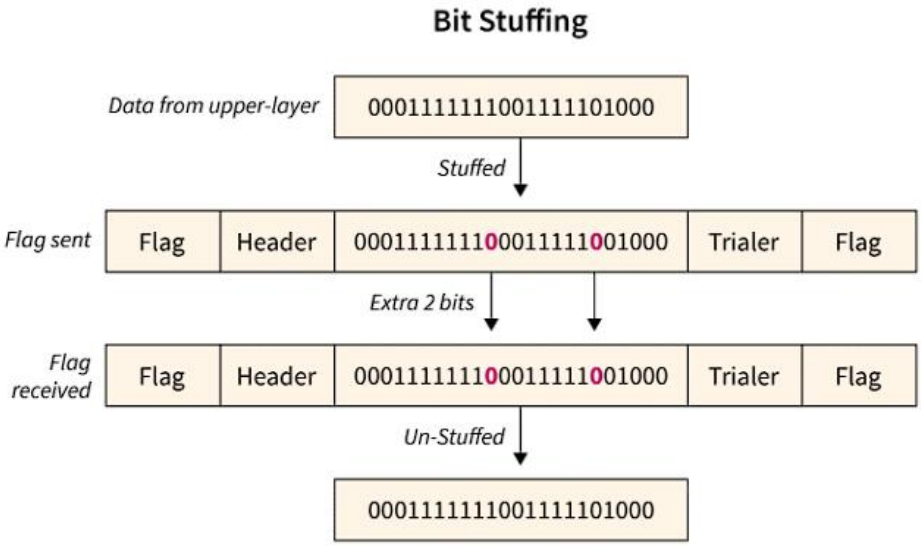
Framing Approaches in Computer Network:

Talking about Framing Approaches in computer networking, there are three 3-different kind of approaches to Framing in the Data link layer:

1. Bit-Oriented Framing:

Most protocols use a special 8-bit pattern flag 01111110 as a result of the delimiter to stipulate the beginning and so the end of the frame. Bit stuffing is completed at the sender end and bit removal at the receiver end.

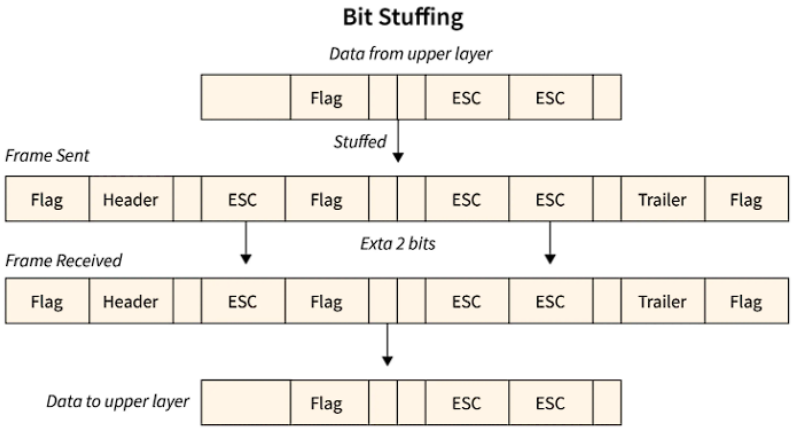
If we have a tendency to get a zero(0) after 5 1s. we have a tendency to tend to still stuff a zero(0). The receiver will remove the zero. Bit stuffing is in addition said as bit stuffing.



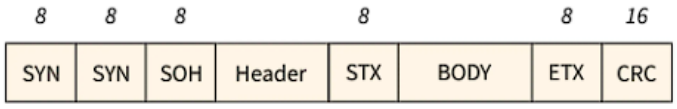
2. Byte-Oriented Framing:

Byte stuffing is one of the methods of adding an additional byte once there is a flag or escape character within the text. Take an illustration of byte stuffing as appears in the given diagram.

The sender sends the frame by adding three additional ESC bits and therefore the destination machine receives the frame and it removes the extra bits to convert the frame into an identical message.



- **Binary Synchronous Communication Protocol (BISYNC)** This is a lookout approach. The frame format outlined is given in the below figure, besides the bits.



SYN: The Special starting character (SYN),

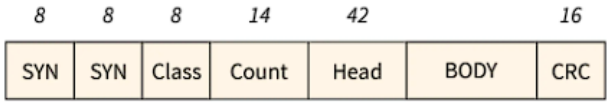
SOH: The Start of the Header (SOH),

STX: The start of text character (STX),

ETX: The end of text character (ETX)

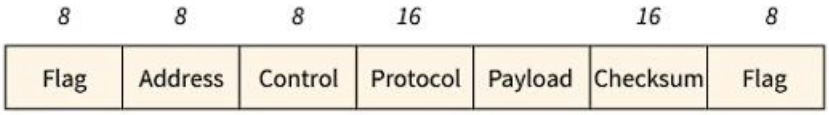
The STX and ETX look after the data part of the portion. To circumvent the framing in data link layer error problem in this approach, we have introduced Byte Stuffing. This can be used once the frames comprise characters. A byte is stuffed within the message to differentiate from the delimiter.

- **Digital Data Communication Message Protocol (DDCMP):** A new count field is introduced during this protocol. The frame format outlined is given in the below figure, in conjunction with the bits.



One risk with this approach is that if transmission error corrupts the count field, then the tip of the frame won't be detected by the receiver properly.

- **Point-to-Point Protocol (PPP):** It is an information link layer protocol. It's a large space network protocol that runs over network links. This protocol is especially employed in broadband communication that deals with high speed and significant hundreds. The frame format outlined is given within the below figure, at the side of the bits. The bit design of the flag is 01111110. The address field is set to about 11111111 just in case of broadcast. The control value is set to be a sustained value of 11000000. The protocol includes one or two bytes that outline the type of data within the payload section. The payload carries the data. The topmost length of the particular field is 1500 bytes. The checksum field is worn for error detection.



3. Clock Based Framing:

This framing in the data link layer is basically used for Optical Networks like SONET. In this approach, a sequence of monotonous pulses maintains an everlasting bit rate and keeps the digital bits oriented within the data stream.

Methods of Framing:

There are four different methods of framing as listed below –

1. Character Count:

In this given methodology is never worn perhaps is usually needed to count the total range of characters that are available in the frame. This process is often done by manipulating fields in the header. This Character count methodology makes sure that the Framing in the data link layer is at the receiver end regarding the total range of characters maintained, and wherever the frame ends. It also has its disadvantage conjointly of utilizing this methodology which is, if in any case, the character count is issued or bended by a miscalculation occurring throughout the transmission process, then at the receiver end it may drop synchronization. The receiver's strength is ineffective in finding or establishing the start of the next frame.

2. Flag Byte with Character Stuffing:

The Flag Byte with Character stuffing is additionally called byte stuffing or character-oriented framing which is identified as especially of bit stuffing however byte stuffing truly works on bytes because bit stuffing works on their bits. while in byte stuffing, a particular byte that is primarily called ESC (Escape Character) which has a fixed design is usually attached to the data section of the frame once there is communication that has a matching pattern, especially of the flag byte.

However, the receiver detaches the ESC along with the data which may cause difficulties.

3. Starting and Ending Flags, with Bit Stuffing:

Starting and Ending Flags, with Bit Stuffing, is additionally called a bit-oriented framing or bit-oriented approach. In bit stuffing, additional bits come about existence supplementary by framing in computer network protocol creators to data streaming. This is typically the lodging of additional bits within the transmission unit as an easy way to offer and provide communication data and information to the recipient and to stay away from or simply disregard the aspect of unwritten or inessential check sequences. Simply this is a kind of protocol control that is merely performed to interrupt the bit sample which ends up in communication undergoing out of synchronization.

4. Encoding Violations:

Encoding violation is a technique that is used just for framing in computer networks within which encoding on physical medium brings about some kind of repetition which is used in more than one graphical or visual structure to easily encode or represent one variable of information.

Advantages of Framing in Data Link Layer:

- Framing in Data links is used incessantly within the method of time-division multiplexing.
- Framing in the data link layer facilitates a type to the sender for transmission of a class of valid bits to a receiver.
- Frames are used continuously in the process of time-division multiplexing.
- It facilitates a form to the sender for transmitting a group of valid bits to a receiver.
- Frames also contain headers that include information such as error-checking codes.

- A-frame relay, token ring, ethernet, and other types of data link layer methods have their frame structures.
- Frames allow the data to be divided into multiple recoverable parts that can be inspected further for corrupt

Framing in the Data Link Layer also presents some challenges, which include:

Variable frame length: The length of frames can vary depending on the data being transmitted, which can lead to inefficiencies in transmission. To address this issue, protocols such as HDLC and PPP use a flag sequence to mark the start and end of each frame.

Bit stuffing: Bit stuffing is a technique used to prevent data from being interpreted as control characters by inserting extra bits into the data stream. However, bit stuffing can lead to issues with synchronization and increase the overhead of the transmission.

Synchronization: Synchronization is critical for ensuring that data frames are transmitted and received correctly. However, synchronization can be challenging, particularly in high-speed networks where frames are transmitted rapidly.

Error detection: Data Link Layer protocols use various techniques to detect errors in the transmitted data, such as checksums and CRCs. However, these techniques are not foolproof and can miss some types of errors.

Efficiency: Efficient use of available bandwidth is critical for ensuring that data is transmitted quickly and reliably. However, the overhead associated with framing and error detection can reduce the overall efficiency of the transmission.

Error Detection and Correction:

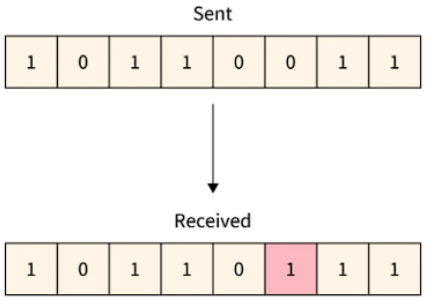
Errors in Communication:

When the information **received at the receiver** end does not match the sent data. At the time of transmission, errors are introduced into the binary data sent from the sender to the receiver due to noise during transmission. This means that a bit having a **0** value can change to **1** and a bit having a **1** value can change to **0**.

Types of Errors:

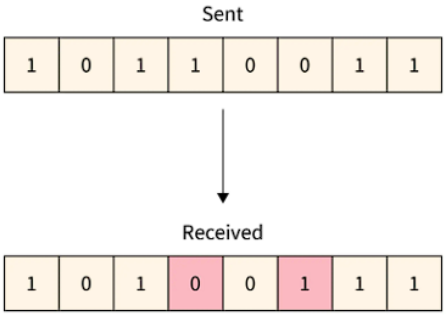
1. Single-bit Error:

A single-bit error refers to a type of data transmission error that occurs when one bit (i.e., a single binary digit) of a transmitted data unit is altered during transmission, resulting in an incorrect or corrupted data unit.



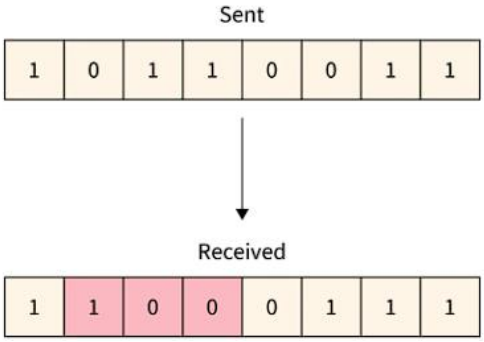
2. Multiple-bit Error:

A multiple-bit error is an error type that arises when more than one bit in a data transmission is affected. Although multiple-bit errors are relatively rare when compared to single-bit errors, they can still occur, particularly in high-noise or high-interference digital environments.



3. Burst Error:

When several consecutive bits are flipped mistakenly in digital transmission, it creates a burst error. This error causes a sequence of consecutive incorrect values.



Error Detection:

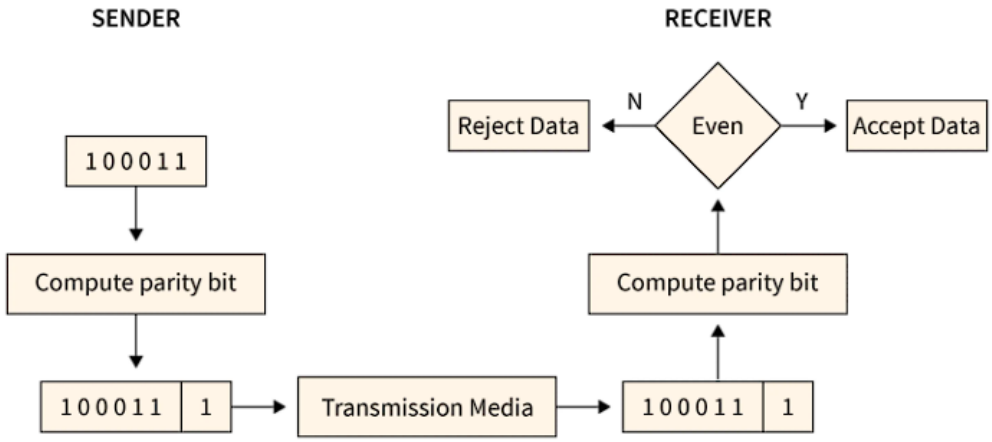
1. Simple Parity Check:

Data sent from the sender undergoes parity check:

- 1 is added as a parity bit to the data block if the data block has an **odd number of 1's**.
- 0 is added as a parity bit to the data block if the data block has an **even number of 1's**.

This procedure is used for making the **number of 1's even**. This is commonly known as even parity checking.

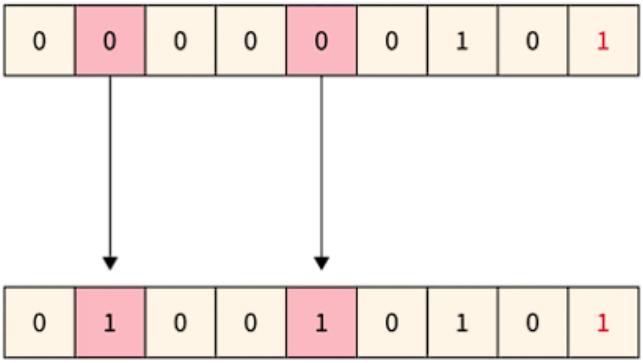
Refer to the below image for the simple parity-checking method:



Disadvantage:

- Only **single-bit error** is detected by this method, it fails in multi-bit error detection.
- It cannot detect an error in case of an error in **two bits**.

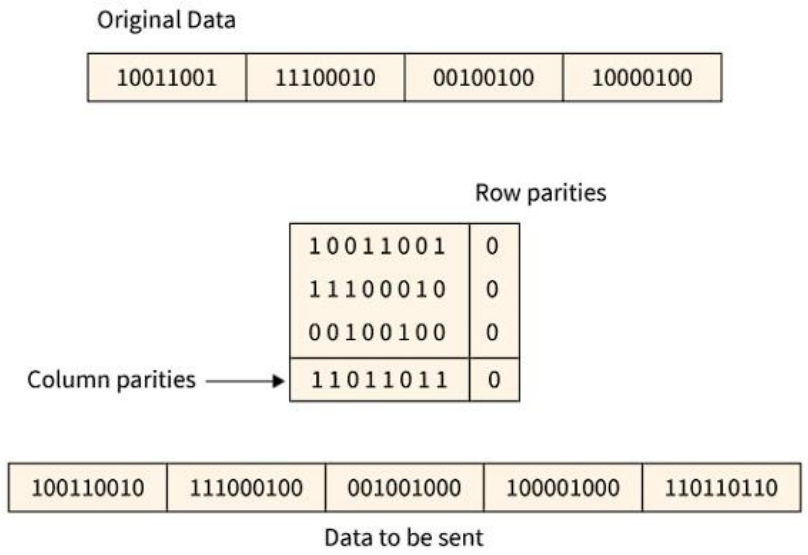
Refer to the below image for the disadvantage simple parity checking method



2. Two-Dimensional Parity Check:

For every **row and column**, parity check bits are calculated by a simple method of parity check. Parity for both rows and columns is transmitted with the data sent from sender to receiver. At the receiver's side, parity bits are compared with the calculated parity of the data received.

Refer to the below image for the two-dimensional parity checking method



Disadvantages:

- If **2 bits are corrupted in 1 data unit** and another data unit exactly at the same position is corrupted then this method is not able to detect the error.
- Sometimes this method is not used for **detecting 4-bit **errors or more than 4-bit errors.

3. Checksum:

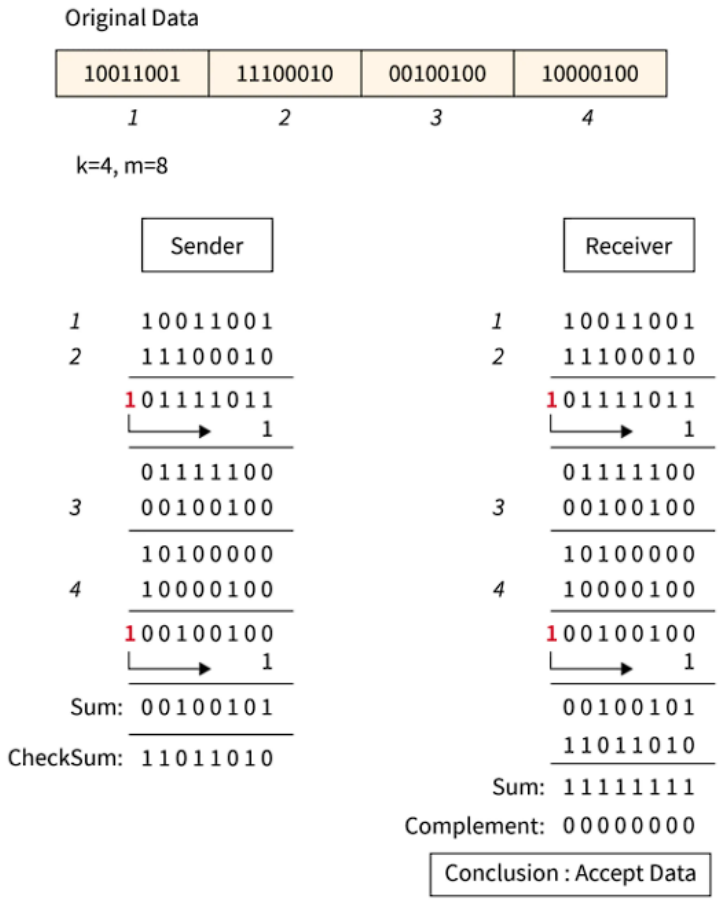
Checksum error detection is a method used to identify errors in transmitted data. The process involves dividing the data into equally sized segments and using a 1's complement to calculate the sum of these segments. The calculated sum is then sent along with the data to the receiver. At the receiver's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

Checksum – Operation at Sender's Side

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

Checksum – Operation at Receiver's Side

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



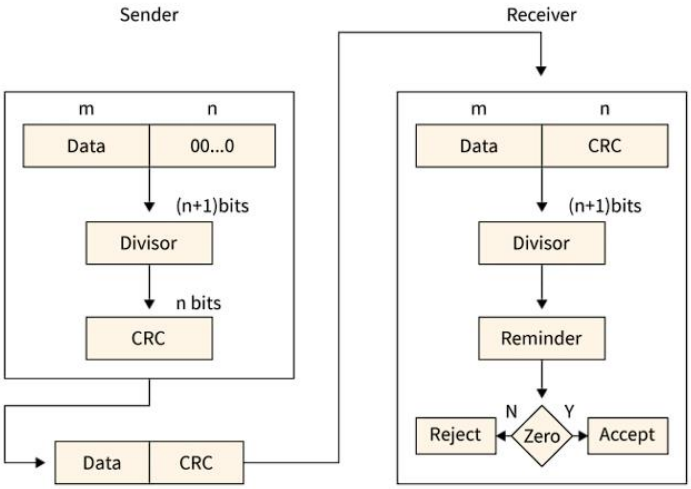
Disadvantages:

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

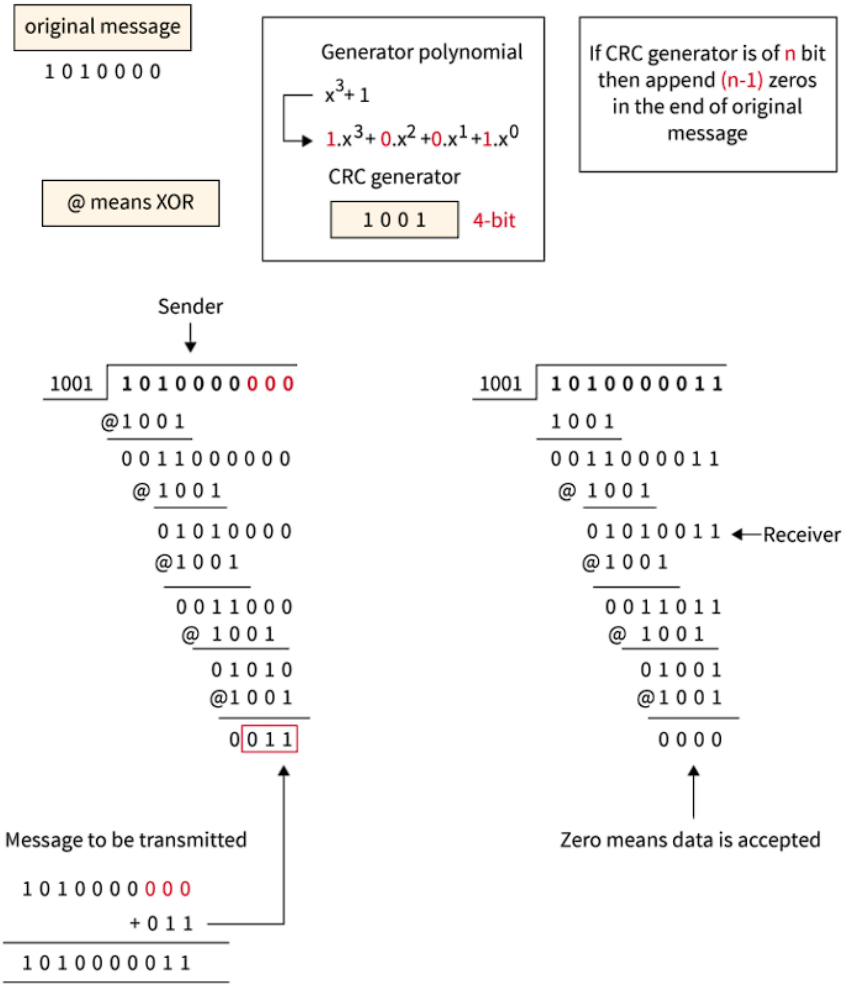
4. Cyclic Redundancy Check (CRC):

- Unlike the checksum scheme, which is based on addition, CRC is based on binary division.
- In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended to the end of the data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number.

- At the destination, the incoming data unit is divided by the same number. If at this step there is no remainder, the data unit is assumed to be correct and is therefore accepted.
- A remainder indicates that the data unit has been damaged in transit and therefore must be rejected.



Refer to the below image for the example of the cyclic redundancy checking method –



Advantages:

Increased Data Reliability: Error detection ensures that the data transmitted over the network is reliable, accurate, and free from errors. This ensures that the recipient receives the same data that was transmitted by the sender.

Improved Network Performance: Error detection mechanisms can help to identify and isolate network issues that are causing errors. This can help to improve the overall performance of the network and reduce downtime.

Enhanced Data Security: Error detection can also help to ensure that the data transmitted over the network is secure and has not been tampered with.

Disadvantages:

Overhead: Error detection requires additional resources and processing power, which can lead to increased overhead on the network. This can result in slower network performance and increased latency.

False Positives: Error detection mechanisms can sometimes generate false positives, which can result in unnecessary retransmission of data. This can further increase the overhead on the network.

Limited Error Correction: Error detection can only identify errors but cannot correct them. This means that the recipient must rely on the sender to retransmit the data, which can lead to further delays and increased network overhead.

Error Correction:

When the data is sent from the **sender side** to the receiver's side it needs to be detected and corrected. So, an error correction method is used for this purpose.

Following are the two ways through which error correction can be handled:

Backward Error Correction:

In this method, when any error is found in the data at the receiver's end. Then the request for resending the whole data unit is sent by the receiver.

Forward Error Correction:

In this method, an error-correcting code is used by the receiver that automatically corrects the errors.

Error Correction Techniques:

We can detect the error using a single additional bit but we cannot use this bit for the correction purpose. It is important to know the exact location of the error if we want to correct that error. For example, for finding out the **single-bit error**, the error detection code checks out that the error is actually in one of the seven bits. Let d represents the number of data bits and r represents the number of redundant bits. The formula below is used for finding the r **number of redundant bits**:

$$2^r >= d + r + 1$$

The above formula is used to find out the **value of r**. For example, suppose 4 will be the value of d, then **3** will be the only and **smallest value** that satisfies this particular relation. A **hamming code** is a technique developed by R.W Hamming for finding out the position of the error bit. This Hamming code is based on the relationship between the redundant bits and data units and its main advantage is that it can be applied to data units of any length.

Hamming Code:

Parity bits: The **parity bits** are the special type of bits that are added to the original data of binary bits to make the total 1s either even or odd.

Even parity: For checking the even parity, the following concept is used: The value of the even parity bit will be **0** if the total occurrence of 1s is even and the value of the parity bit can be 1 if the total occurrence of 1s is odd.

Odd Parity: For checking the even parity, the following concept is used: The value of the parity bit will be 1 if the total occurrence of 1s is even and the value of the parity bit can be 0 if the total occurrence of 1s is odd.

Algorithm of Hamming code:

1. Add the information in **'d'** bits to the redundant bits **'r'** to make data as **d+r**.
2. A decimal value will be assigned by the position of each **(d+r)** digit.
3. In positions **1,2…2k**, the **'r'** bits will be placed.
4. The parity bits are again calculated on the receiver's end. The position of an error defines the parity bit's decimal value.

Relationship b/w Error position & binary number:

Error Position	Binary Number
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

We take this example for a detailed explanation of the concept of hamming code. **1010** is supposed to be the original data that needs to be transferred.

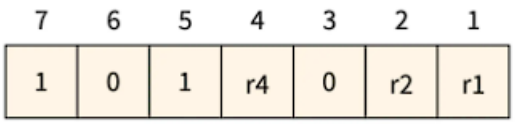
'd' total number of data bits = 4 **Redundant bits number =** $2^r >= d+r+1 = 2^r >= 4+r+1$

So, the above-given relation is completely satisfied when **3 will be the value of r**.

Determining the position of the redundant bits: 3 is the number of redundant bits. r1, r2, and r4 are used to show these three bits. We can find out the position of these bits for the raised power of 2. So according to this, their positions will be 1,21,221,21,22.

- r1 position is 1
- r2 position is 2
- r4 position is 4

Refer to the below image for the data with redundant bits

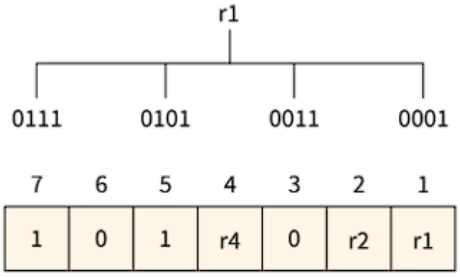


Determining the Parity Bits

Determining the r1 bit

The r1 bit value is calculated based on a parity check performed on the bits available at the position whose binary conversion contains **1** at the first position.

Refer to the below image for the **finding r1 bit value**

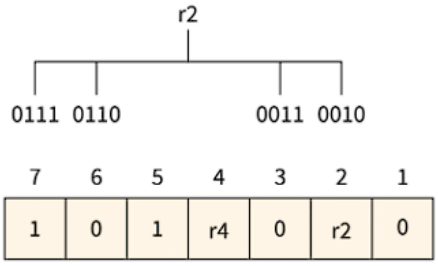


From the above figure, we can find the position having 1 as its first bit in binary representation is **1,3,5,6**. Now the even-parity check method is applied to these bit positions. The number of 1's at these bit positions 2 which is an even number so the 0 is the value of the r1 bit.

Determining r2 bit

The r2 bit value is calculated based on a parity check performed on the bits available at the position whose binary conversion contains **1 at the second position**.

Refer to the below image for the finding **r2 bit value**

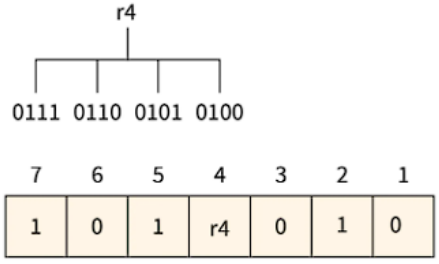


From the above figure, we can find the position having **1** as its second bit in binary representation is **2, 3, 6, 7**. Now the even-parity check method is applied to these bit positions. The number of 1's at these bit positions is 1 which is an odd number so 1 is the value of the **r2 bit**.

Determining r4 bit

The **r4 bit value** is calculated based on a parity check performed on the bits available at the position whose binary conversion contains **1** at the third position.

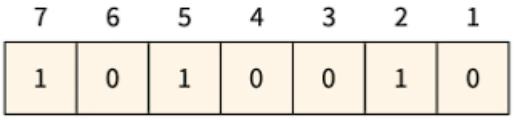
Refer to the below image for the finding **r4 bit value**



From the above figure, we can find the position having **1 as its third bit** in binary representation is **4, 5, 6, 7**. Now the even-parity check method is applied to these bit positions. The number of 1's at these bit positions 2 which is an even number so the **0 is the value of the r4 bit**.

Data to be transmitted is given below:

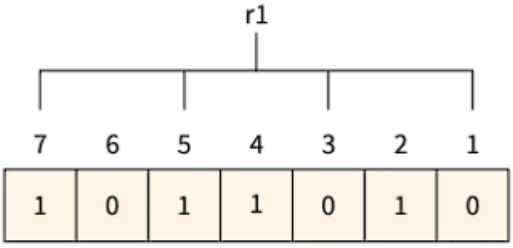
Refer to the below image for the data to be transmitted.



Suppose when the data is received at the receiver end the value of the **4th bit** is changed to **1 from 0**. Now parity bits are calculated again to find the position of the error.

R1 bit 1, 3, 5 and **7** are the positions of the bits for the r1 bit.

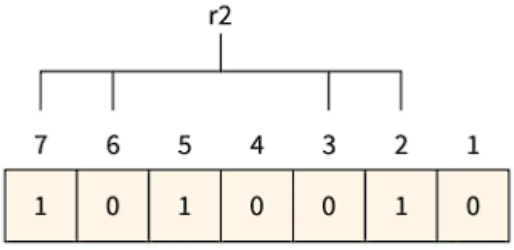
Refer to the below image for calculating the **r1 bit parity**



We can find from the above figure that the r1 binary representation is **1100**. After applying the even-parity check method on the bits appearing in the r4 bits we get an even number of totals 1's. So **0 is the value for the r1**.

R2 bit 2, 3, 6, and 7 are the positions of the bits for the r2 bit.

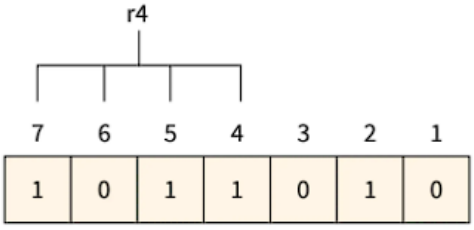
Refer to the below image for calculating the **r2 bit parity**



We can find from the above figure that the r2 binary representation is 1001. After applying the even-parity check method on the bits appearing in the r4 bits we get an even number of **totals 1's**. So, **0 is the value for the r2**.

R4 bit 4, 5, 6 and **7** are the positions of the bits for the **r4 bit**.

Refer to the below image for calculating the **r4-bit parity**



We can find from the above figure that the **r4** binary representation is 1011. After applying the even-parity check method on the bits appearing in the r4 bits we get an odd number of totals 1's. So, **1 is the value for the r4**.

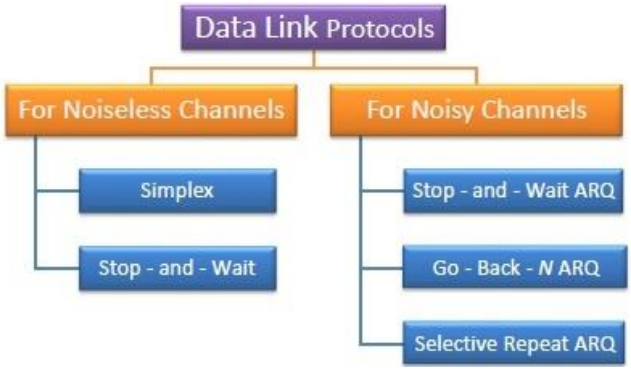
Redundant bits **r4r2r1** binary representation is 100 which is represented as **4** in decimal value. So, there is an error in the data at the 4th position and the value of the bit corresponding to the 4th position must be changed from **1 to 0** for error correction.

Elementary Data Link Protocols:

Data Link Protocols are a set of rules that determine how data is formatted, controlled, and transmitted over a network. These protocols ensure the safe and efficient transfer of data between connected devices. They mainly handle framing, error control, flow control, and the physical addressing of frames.

The study of protocols is divided into two categories: those that can be applied to channels with no noise or errors and those that can be applied to channels with noise or errors. Although the first group of protocols cannot be applied in real-world situations, they provide a foundation for protocols for noisy channels.

Types of Data Link Protocols:



Simplex/Simplest Protocol:

A simplex protocol for a noiseless channel would be one that involves the direct transfer of data from the source to the destination without any intermediate processing. In this scenario, the channel is assumed to be noise-free, which means that the data transmitted remains intact and does not get corrupted.

In a noiseless channel, a simplex protocol could consist of a straightforward method for transmitting data, such as sending one bit at a time, with no error correction or flow control mechanisms in place. This type of protocol is ideal for a noiseless channel as the lack of noise ensures that the data transmitted is received accurately, making the implementation of additional measures unnecessary.

The simplest protocol does not have any mechanisms for controlling the flow of data or detecting and correcting errors, as it is only used in noise-free channels. The protocol assumes that the receiver is always ready to process any data frames sent by the sender immediately. This type of protocol, which only allows data to flow from the sender to the receiver, is known as a one-way protocol. Since this protocol is unidirectional, there is no need for an acknowledgment or confirmation of receipt. Furthermore, because there is no data loss during transmission, there is no need to resend or retransmit the data.

In order to create the simplest protocol, the following presumptions have been made:

- There is no noise whatsoever in the broadcast channel (a channel with no duplications, lost frames, or corrupted frames).
- The assumption is that the transmission route is perfect, with no instances of data loss, duplication, or corruption.
- There is no system for controlling errors and flow.
- The sender's and receiver's end have an endless amount of buffer space for the frames.

Points to Remember:

- The processing time for this technique is quite brief. So, it may be disregarded.
- Each party is constantly prepared to send and receive data.
- Without giving the recipient any thought, the sender delivers a series of data frames.
- No data loss occurs, hence neither an ACK nor a NACK is sent.
- The data is transferred to the following layer right away by the DLL at the receiving end after the frame header has been deleted.

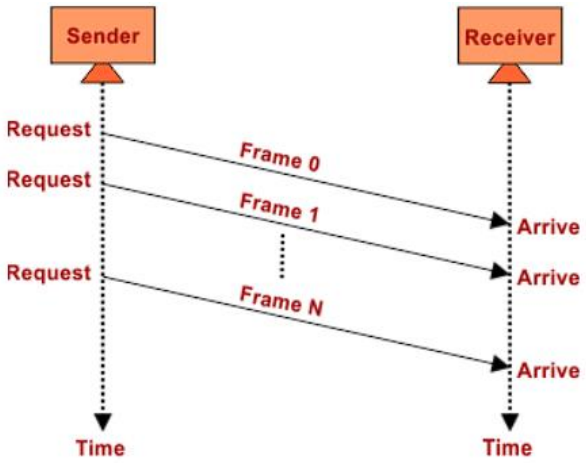
In the absence of an ACK or NACK, the sender must wait a pre-determined time before transmitting the message. To solve this issue, a different protocol known as the Sliding Window Protocol is used. The sliding window protocol is a data connection layer protocol that allows for the sequential and reliable transmission of data frames. This protocol allows the sender to transmit multiple frames at once by utilizing sequence numbers. The sequence numbers are assigned to each data frame by the sender to ensure proper ordering in the event that a frame needs to be retransmitted. This also enables the receiver to detect lost or damaged packets.

Note - An ACK (acknowledgment) is sent to the sender by the receiver after it has received the frame. The ACK informs the sender whether the recipient successfully received a specific frame. Go-Back-N ARQ and Selective Repeat ARQ are the two different kinds of sliding window techniques.

Flow Diagram in Simplest Protocol:

A data flow diagram (DFD) is a graphical representation of the flow of data in a system. In the context of the simplest protocol, a DFD can illustrate the movement of data between the sender and the receiver. The DFD would show how the sender sends the data frames to the receiver, how the receiver processes the data, and what happens if any errors occur during the transfer. It could also show the absence of flow control and error control mechanisms, which are typically included in more complex protocols. The DFD can help to clarify the basic functioning of the simplest protocol, making it easier to understand and implement.

An illustration of a Stop-and-wait protocol-based exchange is shown in this figure. It's still simple to understand. Following the transmission of a frame, the sender awaits the receiver's answer. Send the following frame after waiting for the acknowledgment, or ACK, from the receiving end. Keep in mind that every time there are two frames, the sender is involved in four events and the receivers are involved in two events.



Simplest Protocol

A Simplex Stop and Wait Protocol for an Error-Free Channel:

Stop and wait is a protocol that is used for reliable data transmission in a noiseless channel. In this protocol, the sender sends a single packet at a time and waits for an acknowledgment (ACK) from the receiver before sending the next packet. This way, the sender can ensure that each packet is received by the receiver and has been successfully processed. If the sender does not receive an ACK within a certain time frame, the packet is considered lost and must be retransmitted.

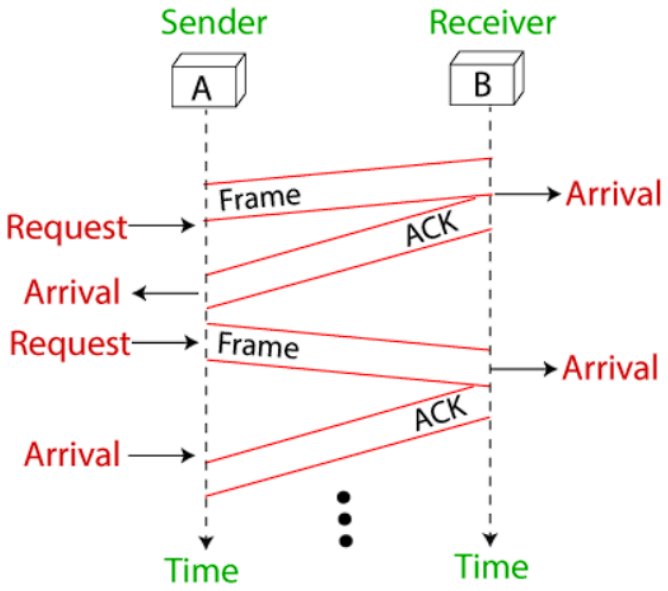
The stop and wait protocol is simple and efficient, but it has one major drawback. Because only one packet can be transmitted at a time, the overall data transmission rate is relatively slow. To overcome this limitation, the sliding window protocol was developed. In the sliding window protocol, multiple packets can be transmitted at the same time, allowing for faster data transmission. Despite this limitation, the stop and wait protocol is still widely used in many applications due to its simplicity and reliability.

- The flow of data frames at the receiver side may become too fast for it to be processed, causing the need for temporary storage.
- However, the limited storage space of the receiver may result in the loss or discarding of frames, or even denial of service.
- To prevent this, the sender must slow down their rate of transmission, which is achieved through the use of ACK messages from the receiver.
- The sender sends a single frame and waits for confirmation from the receiver before sending the next one, adding flow control to the previous protocol.
- The communication remains unidirectional for data frames, but ACK frames flow in the opposite direction.

Flow Diagram:

The flow diagram of the Stop-and-wait protocol in a noiseless channel involves the following steps:

1. The sender transmits a data frame to the receiver.
2. The sender waits for an acknowledgment (ACK) from the receiver.
3. The receiver processes the received data frame.
4. The receiver sends an ACK to the sender to confirm receipt of the data frame.
5. The sender continues to transmit the next data frame, repeating the process from step i.



In this protocol, the sender sends one frame at a time and stops until it receives an ACK from the receiver. This prevents the receiver from becoming overwhelmed with incoming frames and ensures reliable data transmission. Additionally, the ACK frames are used for flow control, allowing the sender to adjust the transmission rate based on the receiver's processing capacity.

Conclusion:

The main difference between the two protocols is that the Simplest Protocol has no flow control and error control mechanisms, while the Stop-and-Wait Protocol employs a flow control mechanism through the use of ACK frames.

In the Simplest Protocol, the recipient is always expected to be ready to receive any frames sent by the sender, so no acknowledgment is needed. In the Stop-and-Wait Protocol, the sender must wait for an acknowledgment from the receiver before sending the next frame.

Another difference between the two protocols is that the Simplest Protocol is unidirectional, while the Stop-and-Wait Protocol is bi-directional. In the Simplest Protocol, data frames can only move in one direction, from sender to receiver, while in the Stop-and-Wait Protocol, both data frames and ACK frames can travel in both directions.

In conclusion, the Simplest Protocol is a basic and straightforward protocol suitable for noiseless channels, while the Stop-and-Wait Protocol adds additional reliability through flow control mechanisms and is more suitable for noisy channels.

Noisy Channel Protocols:

A Noisy Channel Protocol is a type of communication protocol that is used in communication systems where the transmission channel may introduce errors into the transmitted data. This type of protocol is designed to deal with errors in the communication channel and ensure that the data being transmitted is received accurately at the receiver end. The main objective of Noisy Channel Protocols is to minimize the error rate in the transmitted data by using techniques such as error detection and correction, flow control, and retransmission of lost or corrupted data frames. Some examples of Noisy Channel Protocols include the Stop-and-Wait Protocol, the Sliding Window Protocol, and the Automatic Repeat Request (ARQ) Protocol.

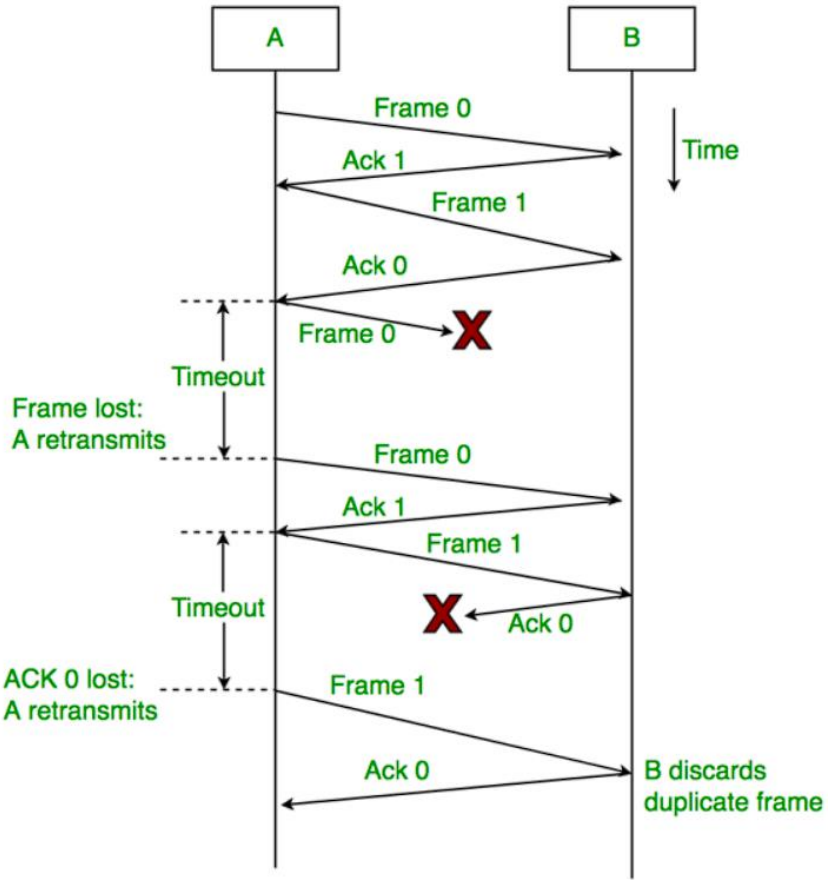
A Simplex Stop and Wait Protocol for Noisy Channel (STOP-AND-WAIT ARQ Protocol):

The Stop and Wait protocol is a protocol used for reliable data transmission over a noisy channel. In this protocol, the sender only sends one frame at a time and waits for an acknowledgment (ACK) from the receiver before sending the next frame. This helps to ensure that the receiver receives the data correctly and eliminates the need for retransmission in the case of errors caused by the noisy channel. The sender continuously monitors the channel for errors, and if an error is detected, it waits for the next ACK before resending the frame. This protocol adds error control to the basic unidirectional communication of data frames and ACK frames in the opposite direction.

Flow Diagram:

A data flow diagram in the Stop-and-Wait protocol in a noisy channel can be used to describe the flow of data between the sender and the receiver. This diagram generally includes the following components:

1. ***Sender*:** The sender sends data frames one at a time, and waits for a response (ACK or NACK) from the receiver before sending the next data frame.
2. ***Receiver*:** The receiver receives the data frames and processes them. If the frame is received correctly, the receiver sends an ACK signal to the sender. If the frame is not received correctly, the receiver sends a NACK signal to the sender.
3. ***Noisy Channel*:** The noisy channel is the medium through which the data frames are transmitted from the sender to the receiver. The channel can add noise to the data frames, resulting in errors and corruption of the data.
4. ***Error Detection*:** The receiver uses error detection techniques such as checksums to detect errors in the received data frames.
5. ***Error Correction*:** If an error is detected, the receiver sends a NACK signal to the sender, requesting a retransmission of the frame.



In this protocol, the sender only sends one data frame at a time and waits for a response from the receiver before sending the next frame. This ensures that the receiver has enough time to process each frame before receiving the next one. The Stop-and-Wait protocol is reliable, but has low throughput compared to other protocols.

Sliding Window Protocols:

Sliding window protocols are data link layer protocols for reliable and sequential delivery of data frames. The sliding window is also used in Transmission Control Protocol.

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.**

Working Principle of Sliding Window:

The sender's buffer is the sending window, and the receiver's buffer is the receiving window in these protocols.

Assignment of sequence numbers to frames is between the range 0 to 2n-1 if the frames' sequence number is an n-bit field. As a result, the sending window has a size of 2n-1. As a result, we choose an n-bit sequence number to accommodate a sending window size of 2n-1.

Modulo-n is used to number the sequence numbers. If the sending window size is set as 3, the sequence numbers will be 0, 1, 2, 0, 1, 2, 0, 1, 2 and so on.

The receiving window's size refers to the maximum number of frames the receiver can accept at one time. It establishes the maximum number of frames a sender can send before receiving an acknowledgement.

Sliding Window (Sender and Receiver side):

a. Sender Side:

The sequence number of the frame occupies a field in the frame. So, the sequence number should be kept to a minimum.

The sequence number ranges from 0 to 2k-1 if the frame header allows k bits.

The sender maintains a list of sequence numbers that are only allowed to be sent by the sender.

The sender window can only be 2k-1 in size.

For example, if the frame allows 4 bits, the window's size is 2 raised to the power 4 -1 16-1=15.

The sender has a buffer with the same size as the window.

b. Receiver Side:

On the receiver side, the size of the window is always 1.

The receiver acknowledges a frame by sending an ACK frame to the sender, along with the sequence number of the next expected frame.

The receiver declares explicitly that it is ready to receive N subsequent frames, starting with the specified number.

We use this scheme in order to acknowledge multiple frames.

The receiver's window can hold 2,3,4 frames, but the ACK frame will be held until frame 4 arrives. It will send the

ACK along with sequence number 5 after the arrival, with which the acknowledgment of 2,3,4 will be done one at a time.

The receiver requires a buffer size of one.

Types Of Sliding Window Protocols:

1. A 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 waits for the protocol. This protocol provides for full – duplex communications. Hence, the acknowledgment is attached along with the next data frame to be sent by piggybacking.

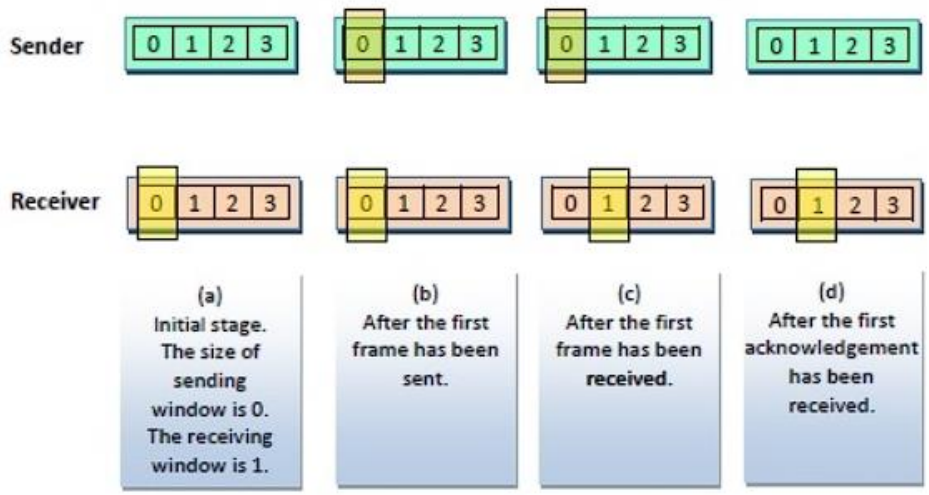
Working Principle:

The data frames to be transmitted additionally have an acknowledgment field, *ack* field that is of a few bits length. The *ack* field contains the sequence number of the last frame received without error. If this sequence number matches with the sequence number of the frame to be sent, then it is inferred that there is no error and the frame is transmitted. Otherwise, it is inferred that there is an error in the frame and the previous frame is retransmitted.

Since this is a bi-directional protocol, the same algorithm applies to both the communicating parties.

Illustrative Example:

The following diagram depicts a scenario with sequence numbers 0, 1, 2, 3, 0, 1, 2 and so on. It depicts the sliding windows in the sending and the receiving stations during frame transmission.



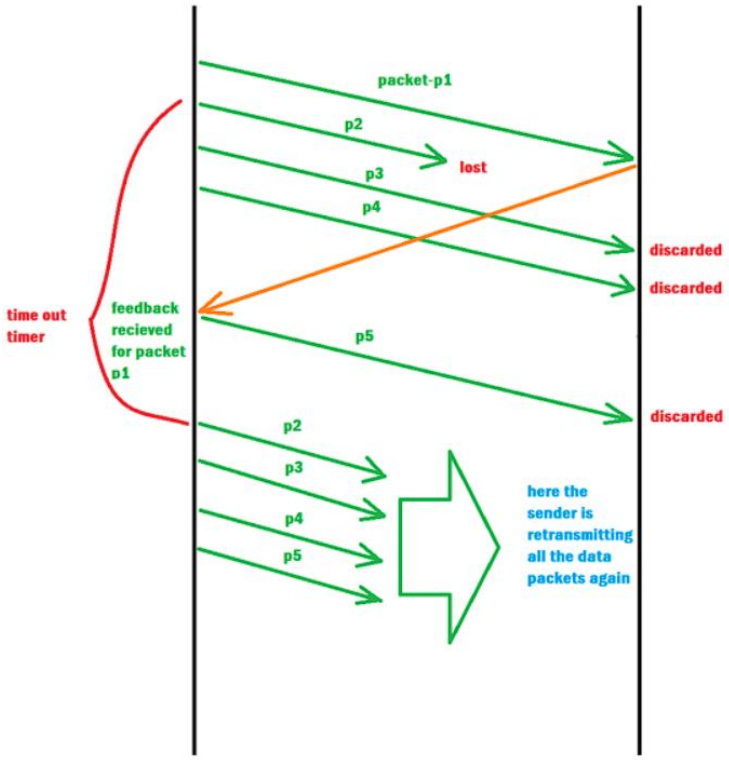
2. GO-BACK-N ARQ Protocol:

The Go-Back-N Automatic Repeat Request (ARQ) protocol is a type of error-control protocol used in data communication to ensure reliable delivery of data over a noisy channel. In a noisy channel, the probability of errors in the received packets is high, and hence, there is a need for a mechanism to detect and correct these errors.

The Go-Back-N ARQ protocol is a type of sliding window protocol where the sender transmits a window of packets to the receiver, and the receiver sends back an acknowledgment (ACK) to the sender indicating successful receipt of the packets. In case the sender does not receive an ACK within a specified timeout period, it retransmits the entire window of packets.

Flow Diagram:

The flow diagram that illustrates the operation of the Go-Back-N ARQ protocol in a noisy channel:



Sender Side:

a. The sender transmits a window of packets to the receiver, starting with sequence number i and ending with sequence number i + N - 1, where N is the window size.

b. The sender sets a timer for each packet in the window.

c. The sender waits for an acknowledgment (ACK) from the receiver.

Receiver Side:

a. The receiver receives the packets and checks for errors.

b. If a packet is received correctly, the receiver sends an ACK back to the sender with the sequence number of the next expected packet.

c. If a packet is received with errors, the receiver discards the packet and sends a negative acknowledgment (NAK) to the sender with the sequence number of the next expected packet.

Sender Side (in case of no ACK received):

1. If the sender does not receive an ACK before the timer for a packet expires, the sender retransmits the entire window of packets starting with the packet whose timer expired.
2. The sender resets the timer for each packet in the window.
3. The sender waits for an ACK from the receiver.

Sender Side (in case of NAK received):

a. If the sender receives a NAK from the receiver, the sender retransmits only the packets that were not correctly received by the receiver.

b. The sender resets the timer for each packet that was retransmitted.

c. The sender waits for an ACK from the receiver.

The above steps are repeated until all packets have been successfully received by the receiver. The Go-Back-N ARQ protocol provides a reliable mechanism for transmitting data over a noisy channel while minimizing the number of retransmissions required.

3. SELECTIVE REPEAT ARQ Protocol:

The Selective Repeat ARQ protocol is a type of error-control protocol used in data communication to ensure reliable delivery of data over a noisy channel. Unlike the Go-Back-N ARQ protocol which retransmits the entire window of packets, the Selective Repeat ARQ protocol retransmits only the packets that were not correctly received.

In the Selective Repeat ARQ protocol, the sender transmits a window of packets to the receiver, and the receiver sends back an acknowledgment (ACK) to the sender indicating successful receipt of the packets. If the receiver detects an error in a packet, it sends a negative acknowledgment (NAK) to the sender requesting retransmission of that packet.

In the Selective Repeat ARQ protocol, the sender maintains a timer for each packet in the window. If the sender does not receive an ACK for a packet before its timer expires, the sender retransmits only that packet.

At the receiver side, if a packet is received correctly, the receiver sends back an ACK with the sequence number of the next expected packet. However, if a packet is received with errors, the receiver discards the packet and sends back a NAK with the sequence number of the packet that needs to be retransmitted.

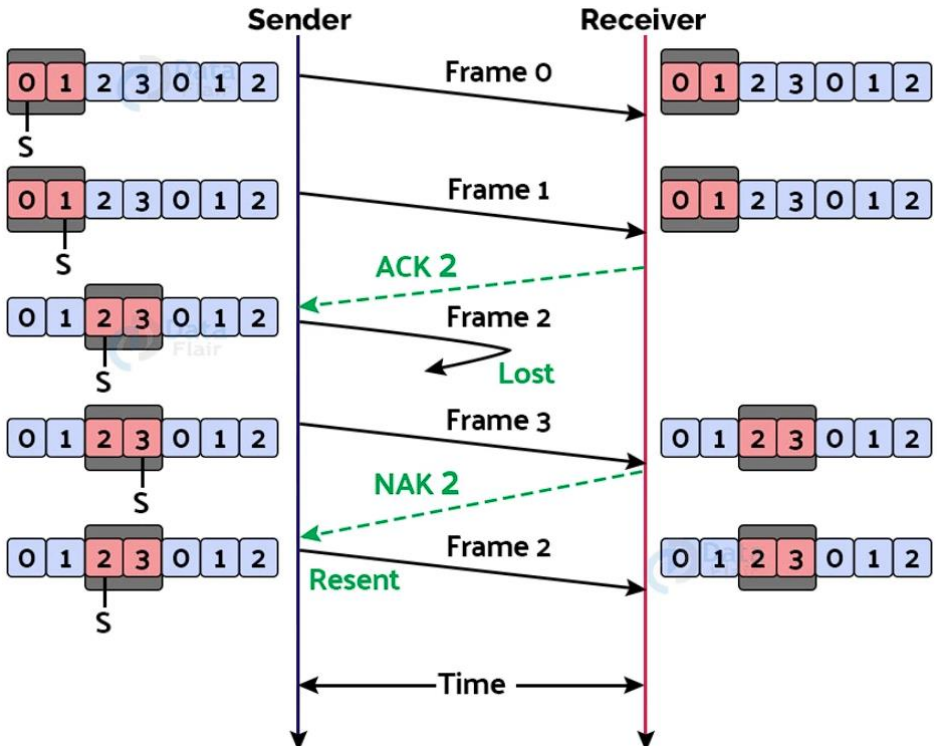
Unlike Go-Back-N ARQ, in Selective Repeat ARQ, the receiver buffer is maintained for all packets that are not in sequence. When a packet with a sequence number different from the expected sequence number arrives at the receiver, it is buffered, and the receiver sends an ACK for the last in-order packet it has received.

If a packet with a sequence number that the receiver has already buffered arrives, it is discarded, and the receiver sends an ACK for the last in-order packet it has received.

In summary, the Selective Repeat ARQ protocol provides a reliable mechanism for transmitting data over a noisy channel while minimizing the number of retransmissions required. It retransmits only the packets that were not correctly received and buffers packets that arrive out of order to reduce the number of retransmissions required.

Flow Diagram:

The flow diagram that illustrates the operation of the Selective Repeat ARQ protocol in a noisy channel:



Sender Side:

a. The sender transmits a window of packets to the receiver, starting with sequence number i and ending with sequence number i + N - 1, where N is the window size.

b. The sender sets a timer for each packet in the window.

c. The sender waits for an acknowledgment (ACK) from the receiver.

Receiver Side:

a. The receiver receives the packets and checks for errors.

b. If a packet is received correctly and is in order, the receiver sends an ACK back to the sender with the sequence number of the next expected packet.

c. If a packet is received with errors or is out of order, the receiver discards the packet and sends a negative acknowledgment (NAK) to the sender with the sequence number of the packet that needs to be retransmitted.

d. The receiver buffers out-of-order packets and sends an ACK for the last in-order packet it has received.

Sender Side (in case of no ACK received):

1. If the sender does not receive an ACK before the timer for a packet expires, the sender retransmits only that packet.
2. The sender resets the timer for the retransmitted packet.
3. The sender waits for an ACK from the receiver.

Sender Side (in case of NAK received):

1. If the sender receives a NAK from the receiver, the sender retransmits only the packets that were not correctly received.
2. The sender resets the timer for each packet that was retransmitted.
3. The sender waits for an ACK from the receiver.

The above steps are repeated until all packets have been successfully received by the receiver. The Selective Repeat ARQ protocol provides a reliable mechanism for transmitting data over a noisy channel while minimizing the number of retransmissions required. It retransmits only the packets that were not correctly received, and buffers out-of-order packets to reduce the number of retransmissions required.

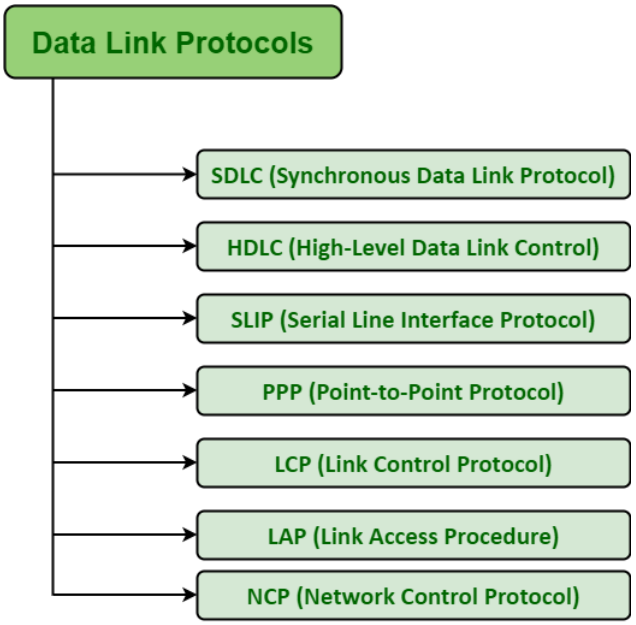
Example Data Link Protocols:

Data Link Layer protocols are generally responsible to simply ensure and confirm that the bits and bytes that are received are identical to the bits and bytes being transferred. It is basically a set of specifications that are used for implementation of data link layer just above the physical layer of the Open System Interconnections (OSI) Model.

Some Common Data Link Protocols:

There are various data link protocols that are required for Wide Area Network (WAN) and modem connections. Logical Link Control (LLC) is a data link protocol of Local Area Network (LAN).

Some of data link protocols are given below:



1. **Synchronous Data Link Protocol (SDLC) –** SDLC is basically a communication protocol of computer. It usually supports multipoint links even error recovery or error correction also. It is usually used to carry SNA (Systems Network Architecture) traffic and is present precursor to HDLC. It is also designed and developed by IBM in 1975. It is also used to connect all of the remote devices to mainframe computers at central locations may be in point-to-point (one-to-one) or point-to-multipoint (one-to-many) connections. It is also used to make sure that the data units should arrive correctly and with right flow from one network point to next network point.
2. **High-Level Data Link Protocol (HDLC) –** HDLC is basically a protocol that is now assumed to be an umbrella under which many Wide Area protocols sit. It is also adopted as a part of X.25 network. It was originally created and developed by ISO in 1979. This protocol is generally based on SDLC. It also provides best-effort unreliable service and also reliable service. HDLC is a bit-oriented protocol that is applicable for point-to-point and multipoint communications both.

3. **Serial Line Interface Protocol (SLIP) –** SLIP is generally an older protocol that is just used to add a framing byte at end of IP packet. It is basically a data link control facility that is required for transferring IP packets usually among Internet Service Providers (ISP) and a home user over a dial-up link. It is an encapsulation of the TCP/IP especially designed to work with over serial ports and several router connections simply for communication. It is some limitations like it does not provide mechanisms such as error correction or error detection.

4. **Point to Point Protocol (PPP) –** PPP is a protocol that is basically used to provide same functionality as SLIP. It is most robust protocol that is used to transport other types of packets also along with IP Packets. It can also be required for dial-up and leased router-router lines. It basically provides framing method to describe frames. It is a character-oriented protocol that is also used for error detection. It is also used to provides two protocols i.e. NCP and LCP. LCP is used for bringing lines up, negotiation of options, bringing them down whereas NCP is used for negotiating network-layer protocols. It is required for same serial interfaces like that of HDLC.

5. **Link Control Protocol (LCP) –** It was originally developed and created by IEEE 802.2. It is also used to provide HDLC style services on LAN (Local Area Network). LCP is basically a PPP protocol that is used for establishing, configuring, testing, maintenance, and ending or terminating links for transmission of data frames.

6. **Link Access Procedure (LAP) –** LAP protocols are basically a data link layer protocols that are required for framing and transferring data across point-to-point links. It also includes some reliability service features. There are basically three types of LAP i.e. LAPB (Link Access Procedure Balanced), LAPD (Link Access Procedure D-Channel), and LAPF (Link Access Procedure Frame-Mode Bearer Services). It is actually originated from IBM SDLC, which is being submitted by IBM to the ISP simply for standardization.

7. **Network Control Protocol (NCP) –** NCP was also an older protocol that was implemented by ARPANET. It basically allows users to have access to use computers and some of the devices at remote locations and also to transfer files among two or more computers. It is generally a set of protocols that is forming a part of PPP. NCP is always available for each and every higher-layer protocol that is supported by PPP. NCP was replaced by TCP/IP in the 1980s.

Medium Access Sub Layer:

The **medium access control** (MAC) is a sublayer of the data link layer of the open system interconnections (OSI) reference model for data transmission. It is responsible for flow control and multiplexing for transmission medium. It controls the transmission of data packets via remotely shared channels. It sends data over the network interface card.

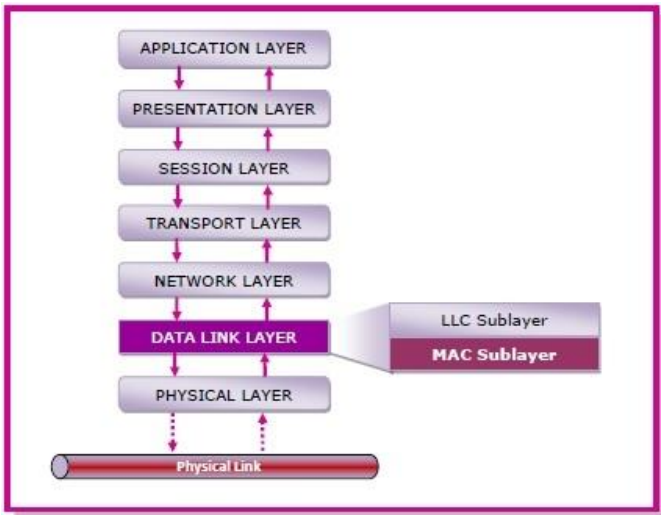
MAC Layer in the OSI Model:

The Open System Interconnections (OSI) model is a layered networking framework that conceptualizes how communications should be done between heterogeneous systems. The data link layer is the second lowest layer.

It is divided into two sublayers −

- The logical link control (LLC) sublayer
- The medium access control (MAC) sublayer

The following diagram depicts the position of the MAC layer −



Functions of MAC Layer:

- It provides an abstraction of the physical layer to the LLC and upper layers of the OSI network.
- It is responsible for encapsulating frames so that they are suitable for transmission via the physical medium.
- It resolves the addressing of source station as well as the destination station, or groups of destination stations.
- It performs multiple access resolutions when more than one data frame is to be transmitted. It determines the channel access methods for transmission.

- It also performs collision resolution and initiating retransmission in case of collisions.
- It generates the frame check sequences and thus contributes to protection against transmission errors.

MAC Addresses:

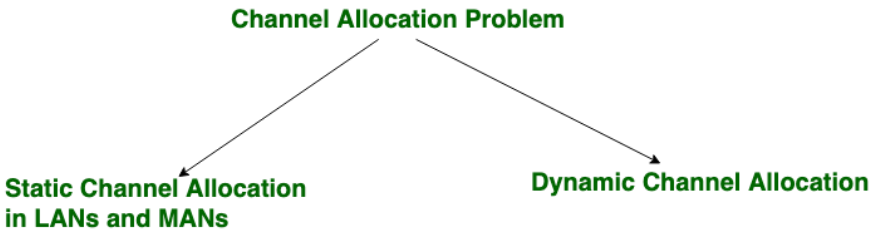
MAC address or media access control address is a unique identifier allotted to a network interface controller (NIC) of a device. It is used as a network address for data transmission within a network segment like Ethernet, Wi-Fi, and Bluetooth.

MAC address is assigned to a network adapter at the time of manufacturing. It is hardwired or hard-coded in the network interface card (NIC). A MAC address comprises of six groups of two hexadecimal digits, separated by hyphens, colons, or no separators. An example of a MAC address is 00:0A:89:5B:F0:11.

The Channel Allocation Problem:

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, then 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.



These are explained as following below.

1. Static Channel Allocation in LANs and MANs:

It is the classical or traditional approach of allocating a single channel among multiple competing users using Frequency Division Multiplexing (FDM). if there are N users, the frequency channel is divided into N equal sized portions (bandwidth), each user being assigned one portion. since each user has a private frequency band, there is no interference between users.

However, it is not suitable in case of a large number of users with variable bandwidth requirements.
It is not efficient to divide into fixed number of chunks.

However, when the number of senders is large and continuously varying or the traffic is bursty, FDM presents some problems. If the spectrum is cut up into N regions and fewer than N users are currently interested in communicating, a large piece of valuable spectrum will be wasted. If more than N users want to communicate, some of them will be denied permission for lack of bandwidth, even if some of the users who have been assigned a frequency band hardly ever transmit or receive anything.

However, even assuming that the number of users could somehow be held constant at N, dividing the single available channel into static subchannels is inherently inefficient. The basic problem is that when some users are quiescent, their bandwidth is simply lost. They are not using it, and no one else is allowed to use it either. Furthermore, in most computer systems, data traffic is extremely bursty (peak traffic to mean traffic ratios of 1000:1 are common). Consequently, most of the channels will be idle most of the time.

The poor performance of static FDM can easily be seen from a simple queueing theory calculation. Let us start with the mean time delay, T, for a channel of capacity C bps, with an arrival rate of λ frames/sec, each frame having a length drawn from an exponential probability density function with mean 1/μ bits/frame. With these parameters the arrival rate is λ frames/sec and the service rate is μC frames/sec. From queueing theory it can be shown that for Poisson arrival and service times,

$$T = \frac{1}{\mu C - \lambda}$$

For example, if C is 100 Mbps, the mean frame length, 1/μ, is 10,000 bits, and the frame arrival rate, l, is 5000 frames/sec, then T = 200 μsec. Note that if we ignored the queueing delay and just asked how long it takes to send a 10,000 bit frame on a 100-Mbps network, we would get the (incorrect) answer of 100 μsec. That result only holds when there is no contention for the channel. Now let us divide the single channel into N independent subchannels, each with capacity C/N bps. The mean input rate on each of the subchannels will now be l/N. Recomputing T we get

$$T_{\text{FDM}} = \frac{1}{\mu(C/N) - (\lambda/N)} = \frac{N}{\mu C - \lambda} = NT$$

Where,

T = mean time delay, **C** = capacity of channel,

L = arrival rate of frames, **N** = number of sub channels,

1/μ = bits/frame,

T(FDM) = Frequency Division Multiplexing Time

The mean delay using FDM is N times worse than if all the frames were somehow magically arranged orderly in a big central queue.

Precisely the same arguments that apply to FDM also apply to time division multiplexing (TDM). Each user is statically allocated every Nth time slot. If a user does not use the allocated slot, it just lies fallow. The same holds if we split up the networks physically. Using our previous example again, if we were to replace the 100-Mbps network with 10 networks of 10 Mbps each and statically allocate each user to one of them, the mean delay would jump from 200 μsec to 2 msec. Since none of the traditional static channel allocation methods work well with bursty traffic, we will now explore dynamic methods.

2. 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 optimises bandwidth usage and results is faster transmissions.

Dynamic channel allocation is further divided into:

1. **Centralised Allocation**
2. **Distributed Allocation**

Possible assumptions include:

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.

Single Channel Assumption: In this allocation all stations are equivalent and can send and receive on that channel.

Collision Assumption: If two frames overlap in time-wise, then that's collision. Any collision is an error, and both frames must re transmitted. Collisions are only possible error.

Time can be divided into Slotted or Continuous.

Stations can sense a channel is busy before they try it.

Protocol Assumption:

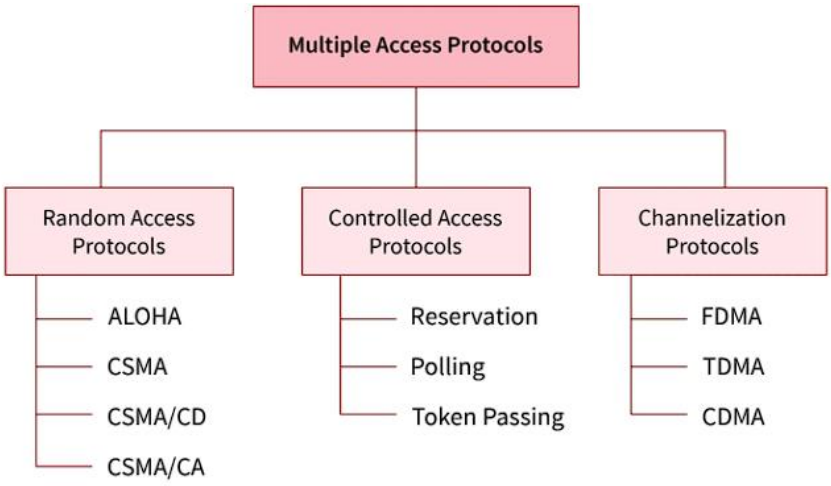
- N independent stations.
- A station is blocked until its generated frame is transmitted.
- probability of a frame being generated in a period of length Dt is IDt where I is the arrival rate of frames.
- Only a single Channel available.
- Time can be either: Continuous or slotted.
- **Carrier Sense:** A station can sense if a channel is already busy before transmission.
- **No Carrier Sense:** Time out used to sense loss data.

Multiple Access Protocols:

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.

Types of Multiple Access Protocols:



1. Random Access Protocol: In this, all stations have same superiority that is no station has more priority than another station. Any station can send data depending on medium's state (idle or busy). It has two features:

1. There is no fixed time for sending data
2. There is no fixed sequence of stations sending data

The Random access protocols are further subdivided as:

1) ALOHA:

The ALOHA protocol or conjointly referred to as the ALOHA methodology could be an easy communication medium during which each transmittal station or supply in an exceeding network can send the data or the information whenever a frame is accessible for transmission. If we tend to succeed and therefore the frame reaches its destination, then the successive frame is lined up for transmission. however, keep in mind, that if the data or the information frame isn't received by the receiver which may be due to the collision then the frame is shipped once more till it reaches the receiver's end successfully.

There are two different versions of Aloha:

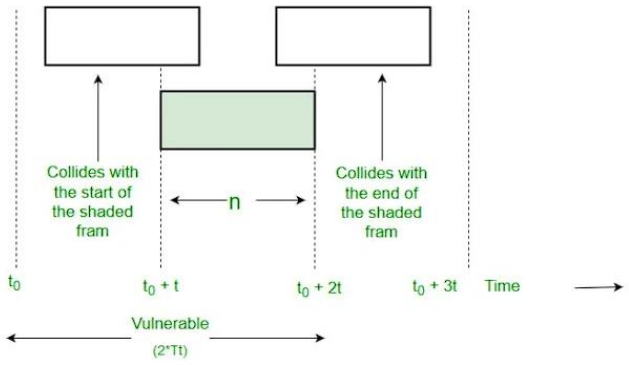
1. Pure Aloha:

Pure Aloha is an un-slotted, decentralized, and simple-to-implement protocol. In pure Aloha, the stations simply transmit frames whenever they want data to send. It does not check whether the channel is busy or not before transmitting. In the event that two or more stations transmit simultaneously, a collision occurs and frames are destroyed. Whenever any station transmits a frame, it expects acknowledgment from the receiver. If it is not received within a specified time, the station assumes that the frame or acknowledgment has been destroyed. Then, the station waits for a random amount of time and sends the frame again. This randomness helps in avoiding more collisions. This scheme works well in small networks where the load is not much. But in largely loaded networks, this scheme fails poorly. This led to the development of Slotted Aloha.

To assure pure aloha: Its throughput and rate of transmission of the frame are to be predicted.

For that, let's make some assumptions:

- All the frames should be the same length.
- Stations cannot generate frames while transmitting or trying to transmit frames.
- The population of stations attempts to transmit (both new frames and old frames that collided) according to a Poisson distribution.



Vulnerable Time = 2 * Tt

The efficiency of Pure ALOHA:

***Pure Aloha*= G * e^-2G**

where G is number of stations wants to transmit in Tt slot.

Maximum Efficiency:

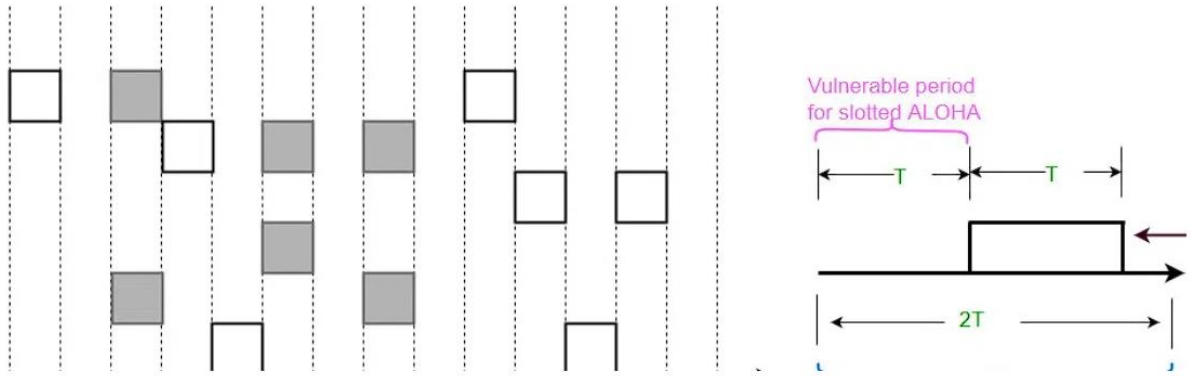
Maximum Efficiency will be obtained when G=1/2

(pure aloha)*max* = 1/2 * e^-1 = 0.184

*Which means, in Pure ALOHA, only about **18.4%** of the time is used for successful transmissions.*

2. Slotted Aloha:

This is quite similar to Pure Aloha, differing only in the way transmissions take place. Instead of transmitting right at the 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.



Collision is possible only in the current slot. Therefore, Vulnerable Time is **Tt**.

The efficiency of Slotted Aloha:

slotted Aloha= G * e^-G

Maximum Efficiency:

(slotted)*max* = 1 * e^-1 = 1/e = 0.368

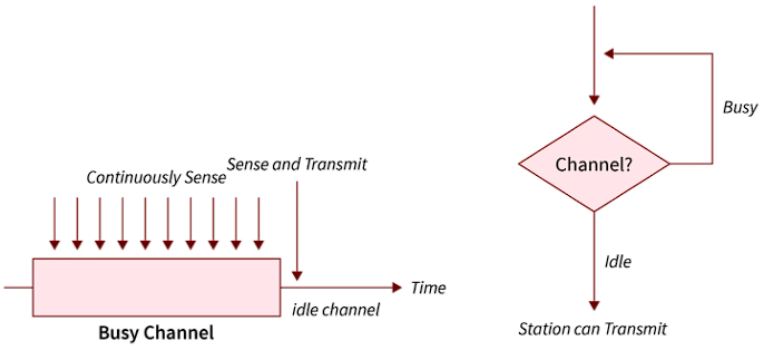
Maximum Efficiency, in Slotted ALOHA, is **36.8%.**

2) CSMA (Carrier Sense Multiple Access):

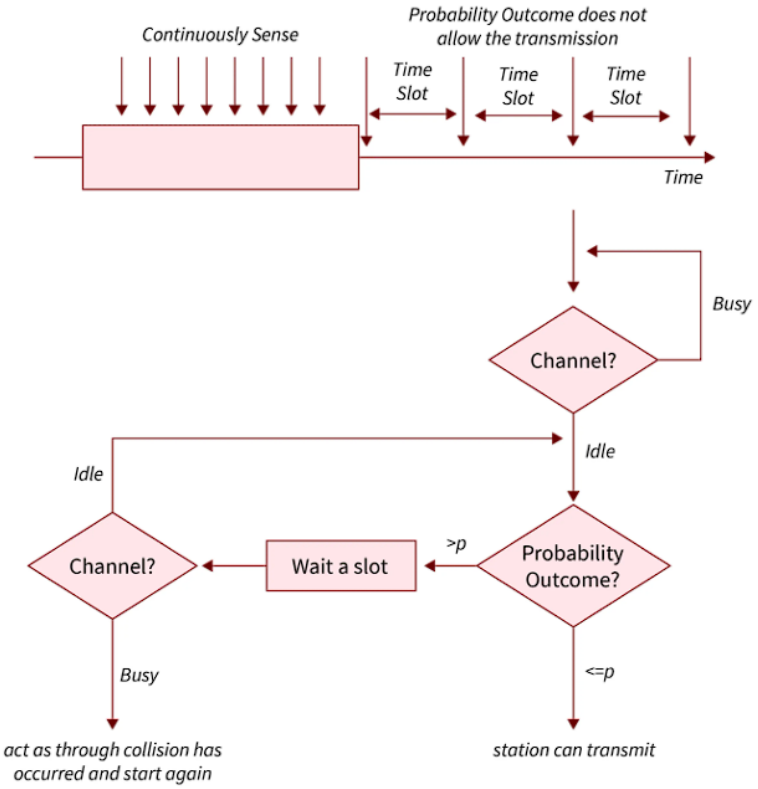
Carrier Sense Multiple Access ensures fewer collisions as the station is required to first sense the medium (for idle or busy) before transmitting data. If it is idle then it sends data, otherwise it waits till the channel becomes idle. However, there is still chance of collision in CSMA due to propagation delay. For example, if station A wants to send data, it will first sense the medium. If it finds the channel idle, it will start sending data. However, by the time the first bit of data is transmitted (delayed due to propagation delay) from station A, if station B requests to send data and senses the medium it will also find it idle and will also send data. This will result in collision of data from station A and B.

CSMA is having 4 different access protocol which is listed below:

- **1-persistent:** In this, initially, the node checks the channel, if the channel is passive then the node or station transmits information, it keeps on waiting and whenever the channel is passive, the stations transmit the data frame.



- **Non-persistent:** In this non-persistent, the station checks the channel equally as 1-persistent mode, however, the sole distinction is that once the channel is busy it checks it once more after a random quantity of your time, not like the 1-persistent mode wherever the stations keep it up checking incessantly.
- **P-persistent:** In P-persistent, the station checks the channel and if found passive then it transmits the information/data frame with the likelihood of P and if the data isn't transmitted (1-P) then the station waits for a random quantity of your time and another time transmits the data with the likelihood P and this cycle goes on endlessly till the data-frame is sent successfully.



- **O-persistent:** In O-persistent, the transmission supports the prevalence of stations which is set beforehand and transmission happens in this given order. If the channel is passive, then the station waits for its address to send the data frame.

Variations of CSMA Protocol:

1. Carrier Sense Multiple Access with Collision Detection (CSMA/CD):

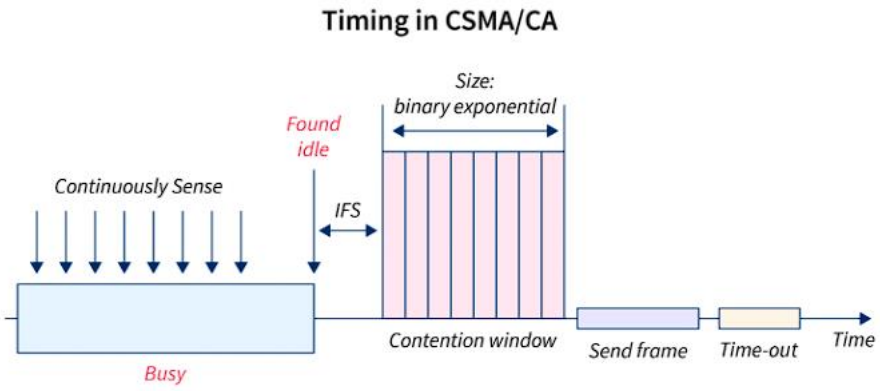
CSMA/CD basically means CSMA with Collision Detection. In CSMA/CD, whenever the station transmits data-frame it then detects the channel or the medium to admit the state of the transmission that is profitably transmitted or failed. If the transmission succeeds, then it produces the successive frame otherwise it resends the formerly failed data frame. The purpose to recollect here is, that the frame coordinate universal time ought to be at minimum twice the most propagation time, which might be deduced once the gap between the 2 stations concerned in a collision is most.

2. Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA):

CSMA/CA is generally known as CSMA with collision avoidance. To sight the potential collisions, the sender receives the acknowledgment, and if there is just one acknowledgment gift of its own. then this suggests that the data frame has been sent profitably. But if there are two or more additional acknowledgment signals then this means that the collision has occurred.

To avoid collision following methods are used in general -

- **Interframe Space:** In the Interface space case, assume that your station waits for the channel to become passive and locate that the channel is passive, then it will not convey the data-frame Straight away to keep away from the collision due to propagation hold on. It rather waits for a few moments called interframe space or IFS, and when the time is up the station once again checks the medium for being passive. However, it ought to have remained in mind that the IFS length depends on the first concern of the station.
- **Contention Window:** In this contention window, the time is split into slots. Say, if the sender is prepared for transmission of the data or the in, it then chooses a random variety of slots as waiting time that doubles every time if the channel is busy. But, if the channel is not passive at that moment, then it does not restart the whole method; however, it restarts the timer once when the channel is found passive again.
- **Acknowledgement:** We have a tendency to mention on top of the sender station that will re-transmit the data if acknowledgment is not received prior to the timer runouts.



2. Controlled Access Protocols:

In a controlled access protocol, the stations obtain info from each other to seek out what the station has the authority to send. It permits just one node to send at a time, to control the collision of messages on a shared medium. The 3 controlled-access methods are given below:

- **Reservation:**

In the reservation methodology, a station has to create a reservation prior to data.

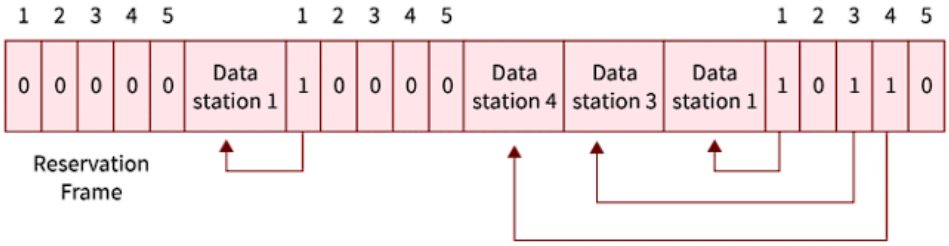
The schedule has 2 kinds of periods:

1. Reservation interval of agreed time length
2. Data transmission duration of variable frames.

If there are M stations, the reservation interval is split into M slots, and every station has only one slot.

Assume that station one encompasses a frame to send, it transmits one bit throughout slot one. No alternative station is permitted to transmit throughout this slot.

Commonly, i the station could announce that it's a frame to send by inserting one bit into i the slot. Finally, all N slots have been checked, every station is aware of which stations want to transmit.



In this figure a condition with 5 stations and 5 slot reservations available. However, in this given first interval only 1,3 and 4 are able to make the reservation and in another second interval, only 1 could make it.

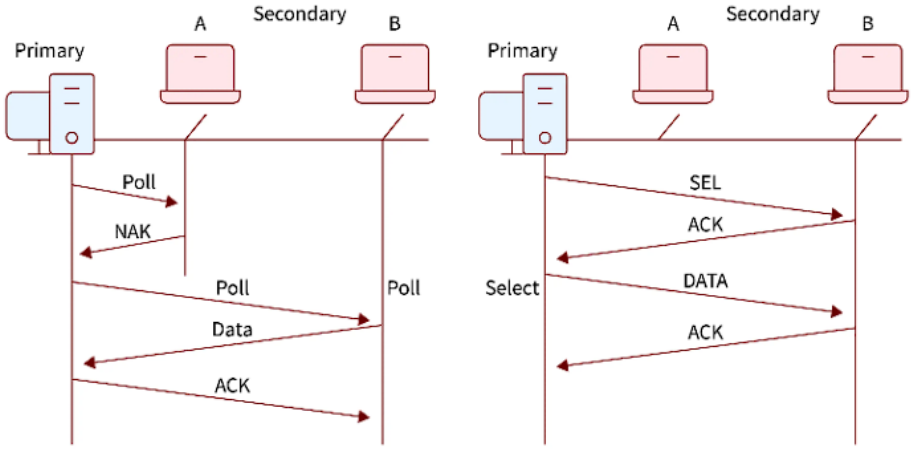
- **Polling:**

The polling method is analogous to the roll call performed in school. rather like the teacher, a controller sends a message to every node successively.

In this, one acts as a primary station(controller) and therefore the others are secondary stations. All data exchanges should be created through the controller.

The message sent by the controller holds the address of the node being hand-picked for granting access.

Although all nodes receive the message however the addressed one responds to that and sends data if any. If no data is available, normally a "poll rejection" (NAK) message is dispatched back.



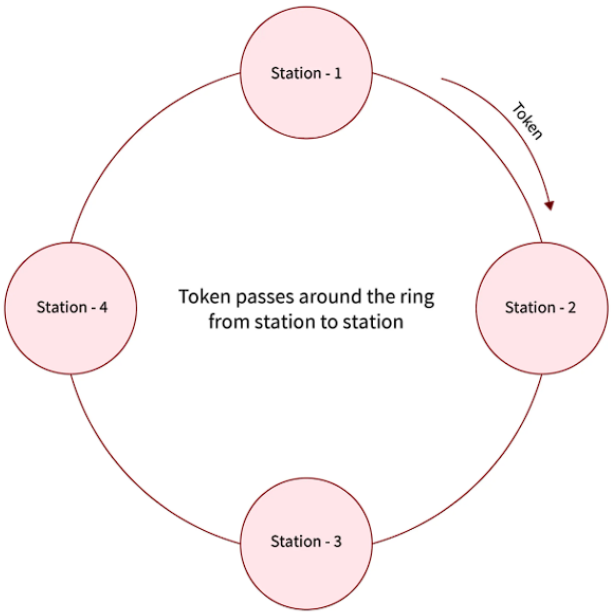
- **Token Passing:**

In the token-passing process, the stations are linked logically to every alternative in the form of a ring, and access to stations is ruled by tokens.

A token could be a special bit pattern or a little message, that flows from 1 station to the consecutive in some predefined order.

In a Token ring, the token is passed from 1 station to a different adjacent station within the ring whereas, in the case of a Token bus, every station uses the bus to send the token to the consecutive station in some predefined order.

Basically, in both the given cases, the token represents permission to send. If a station incorporates a frame queued for transmission once it receives the token, it will redirect that frame before it passes the token to the consecutive station. If it has no queued frame, it passes the token merely.



3. Channelization Protocols:

In the channelization protocol, the accessible bandwidth of the link is split in time, frequency, and code to numerous stations to access the channel at the same time.

- **Frequency Division Multiple Access (FDMA):**

The convenient bandwidth is split into equal bands for every station that will be allotted its band. Guard bands are attached in such an order that no 2 bands overlap to avoid debate and noise.

- **Time Division Multiple Access (TDMA):**

In this Time Division of Multiple Access, the bandwidth is divided between multiple stations. To circumvent collision time is split into slots and stations are allocated these slots to transmit data. but there is an overhead of synchronization as every station must grasp its time slot. This is often resolved by adding synchronization bits to every slot. Another issue with TDMA is propagation delay which is determined by the supplementary guard bands.

- **Code Division Multiple Access (CDMA):**

One channel carries every transmission at the same time. There is neither a splitting of bandwidth nor a splitting of time. Let's Assume, that there are plenty of people in a hall all speaking at an identical time, then in addition to that excellent reception of data is feasible if only 2 people speak an identical language. Similarly, data from completely different stations will be transmitted at the same time in numerous code languages.

Collision Free Protocols:

Almost all collisions can be avoided in **CSMA/CD** but they can still occur during the contention period. The collision during the contention period adversely affects the system performance, this happens when the cable is long and length of packet are short. This problem becomes serious as fiber optics network came into use.

Here we shall discuss some protocols that resolve the collision during the contention period -

- Bit-map Protocol
- Binary Countdown
- Limited Contention Protocols
- The Adaptive Tree Walk Protocol

Pure and slotted Aloha, CSMA and CSMA/CD are **Contention based Protocols:**

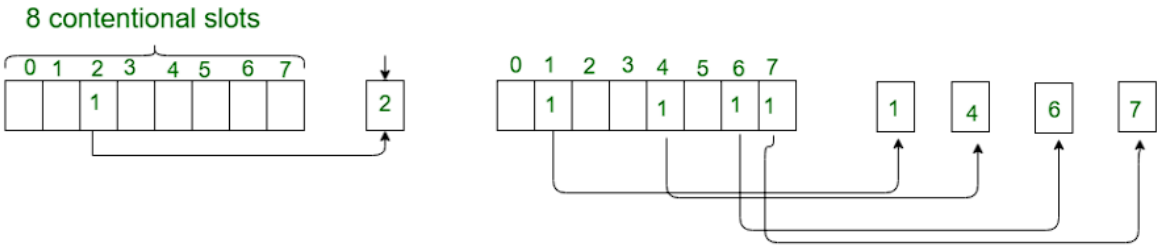
- Try-if collide-Retry
- No guarantee of performance
- What happen if the network load is high?

Collision Free Protocols:

- Pay constant overhead to achieve performance guarantee
- Good when network load is high

1. Bit-map Protocol: Bit map protocol is collision free Protocol. In bitmap protocol method, each contention period consists of exactly N slots. If any station has to send frame, then it transmits a 1 bit in the corresponding slot. For example, if station 2 has a frame to send, it transmits a 1 bit to the 2nd slot.

In general, Station 1 Announce the fact that it has a frame question by inserting a 1 bit into slot 1. In this way, each station has complete knowledge of which station wishes to transmit. There will never be any collisions because everyone agrees on who goes next. Protocols like this in which the desire to transmit is broadcasting for the actual transmission are called Reservation Protocols.



Bit Map Protocol

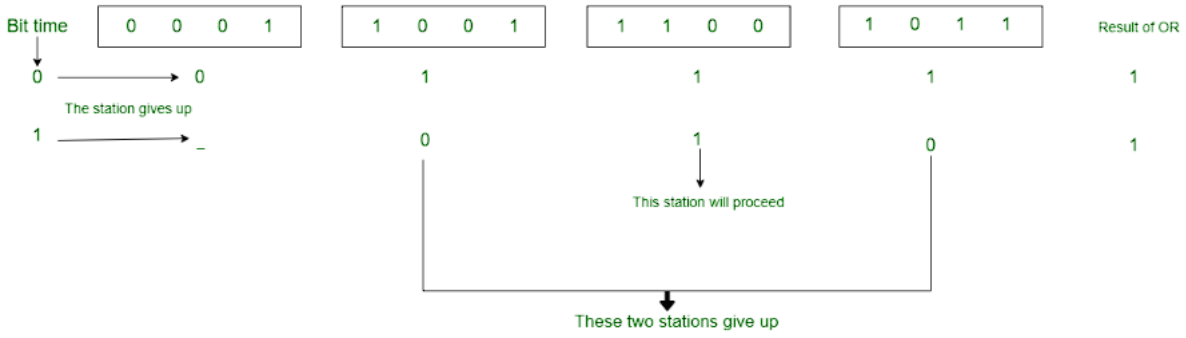
For analyzing the performance of this protocol, we will measure time in units of the contention bits slot, with a data frame consisting of d time units. Under low load conditions, the bitmap will simply be repeated over and over, for lack of data frames. All the stations have something to send all the time at high load, the N bit contention period is prorated over N frames, yielding an overhead of only 1 bit per frame.

Generally, high numbered stations have to wait for half a scan before starting to transmit low numbered stations have to wait for half a scan (N/2 bit slots) before starting to transmit, low numbered stations have to wait on an average 1.5 N slots.

2. Binary Countdown: Binary countdown protocol is used to overcome the overhead 1 bit per binary station. In binary countdown, binary station addresses are used. A station wanting to use the channel broadcast its address as binary bit string starting with the high order bit. All addresses are assumed of the same length. Here, we will see the example to illustrate the working of the binary countdown.

In this method, different station addresses are read together who decide the priority of transmitting. If these stations 0001, 1001, 1100, 1011 all are trying to seize the channel for transmission. All the station at first broadcast their most significant address bit that is 0, 1, 1, 1 respectively. The most significant bits are read together. Station 0001 see the 1 MSB in another station address and knows that a higher numbered station is competing for the channel, so it gives up for the current round.

Other three stations 1001, 1100, 1011 continue. The next station at which next bit is 1 is at station 1100, so station 1011 and 1001 give up because there 2nd bit is 0. Then station 1100 starts transmitting a frame, after which another bidding cycle starts.



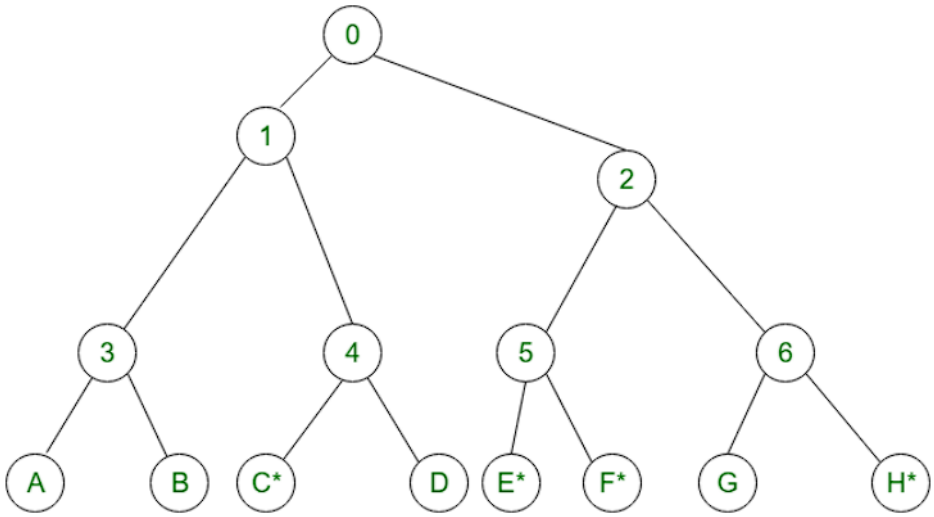
Binary countdown

3. Limited Contention Protocols:

- Collision based protocols (pure and slotted ALOHA, CSMA/CD) are good when the network load is low.
- Collision free protocols (bitmap, binary Countdown) are good when load is high.
- How about combining their advantages:
  1. Behave like the ALOHA scheme under light load
  2. Behave like the bitmap scheme under heavy load.

4. Adaptive Tree Walk Protocol:

- partition the group of station and limit the contention for each slot.
- Under light load, everyone can try for each slot like aloha
- Under heavy load, only a group can try for each slot
- How do we do it:
  1. treat every station as the leaf of a binary tree
  2. first slot (after successful transmission), all stations can try to get the slot (under the root node).
  3. If no conflict, fine.
  4. Else, in case of conflict, only nodes under a subtree get to try for the next one. (depth first search)



Adaptive Tree Walk Protocol fig (1.3)

Slot-0: C*, E*, F*, H* (all nodes under node 0 can try which are going to send), conflict

Slot-1: C* (all nodes under node 1 can try}, C sends

Slot-2: E*, F*, H*(all nodes under node 2 can try}, conflict

Slot-3: E*, F* (all nodes under node 5 can try to send), conflict

Slot-4: E* (all nodes under E can try), E sends

Slot-5: F* (all nodes under F can try), F sends

Slot-6: H* (all nodes under node 6 can try to send), H sends.

Wireless LANs (WLANs):

WLAN stands for **Wireless Local Area Network.** WLAN is a local area network that uses radio communication to provide mobility to the network users while maintaining the connectivity to the wired network. A WLAN basically, extends a wired local area network. WLANs are built by attaching a device called the access point (AP) to the edge of the wired network. Clients communicate with the AP using a wireless network adapter which is similar in function to an ethernet adapter. It is also called a LAWN is a Local area wireless network.

The performance of WLAN is high compared to other wireless networks. The coverage of WLAN is within a campus or building or that tech park. It is used in the mobile propagation of wired networks. The standards of WLAN are HiperLAN, Wi-Fi, and IEEE 802.11. It offers service to the desktop laptop, mobile application, and all the devices that work on the Internet. WLAN is an affordable method and can be set up in 24 hours. WLAN gives users the mobility to move around within a local coverage area and still be connected to the network. Most latest brands are based on IEE 802.11 standards, which are the WI-FI brand name.

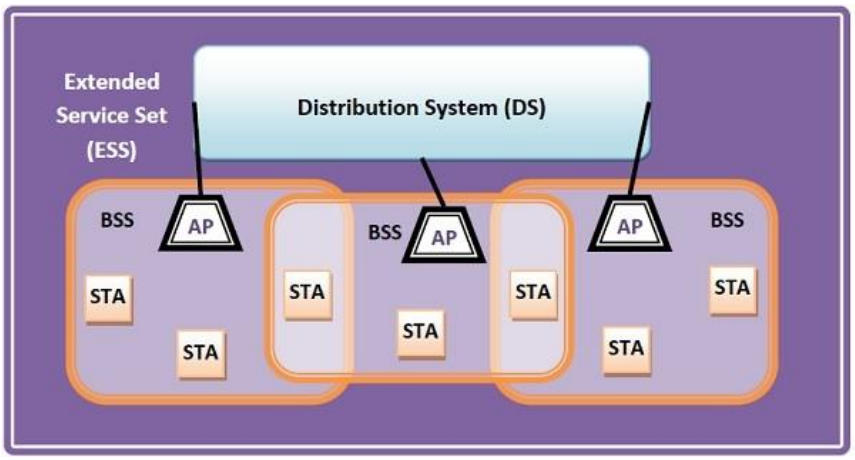
History of WLAN:

A professor at the University of Hawaii whose name was Norman Abramson, developed the world's first wireless computer communication network. In 1979, Gfeller and u. Bapst published a paper in the IEE proceedings reporting an experimental wireless local area network using diffused infrared communications. The first of the IEEE workshops on Wireless LAN was held in 1991.

WLAN Architecture:

The components of WLAN architecture as laid down in IEEE 802.11 are −

- **Stations (STA)** − Stations comprises of all devices and equipment that are connected to the wireless LAN. Each station has a wireless network interface controller. A station can be of two types −
    - Wireless Access Point (WAP or AP)
    - Client
- **Basic Service Set (BSS)** − A basic service set is a group of stations communicating at the physical layer level. BSS can be of two categories −
    - Infrastructure BSS
    - Independent BSS
- **Extended Service Set (ESS)** − It is a set of all connected BSS.
- **Distribution System (DS)** − It connects access points in ESS.



Types of WLANs:

As per IEEE standard WLAN is categorized into two basic modes, which are as follows:

1. **Infrastructure:** In Infrastructure mode, all the endpoints are connected to a base station and communicate through that; and this can also enable internet access. A WLAN infrastructure can be set up with: a wireless router (base station) and an endpoint (computer, mobile phone, etc). An office or home WiFi connection is an example of Infrastructure mode.
2. **Ad Hoc:** In Ad Hoc mode WLAN connects devices without a base station, like a computer workstation. An Ad Hoc WLAN is easy to set up it provides peer-to-peer communication. It requires two or more endpoints with built-in radio transmission.

Working of WLAN:

WLAN transmits data over radio signals and the data is sent in the form of a packet. Each packet consists of layers, labels, and instructions with unique MAC addresses assigned to endpoints. This enables routing data packets to correct locations.

Characteristics of WLAN:

1. Seamless operation.
2. Low power for battery use.
3. Simple management, easy to use for everyone.
4. Protection of investment in wired networks.
5. Robust transmission technology.

Is a WLAN secure?

A WLAN is more vulnerable to being breached than a physical network. With a wired network, a bad actor must gain physical access to an internal network or breach an external firewall. To access a WLAN, a bad actor must simply be within range of the network.

The most basic method of securing a WLAN is to use MAC addresses to disallow unauthorized stations. However, determined adversaries may be able to join networks by spoofing an authorized address.

The most common security method for a WLAN is encryption, including Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access (WPA), with WPA2 as the standard authentication method.

WLAN Applications:

The wireless LAN network has various applications, depending on the place it is being used. Such as, if you are planning to use Wireless LAN at home, it will allow you to connect devices such as laptops, smartphones, and smart TVs to the internet.

In office places, the Wireless LAN network will allow your employees to connect their devices to the internet and share resources such as printers and files with other employees.

The Wireless LAN network will allow customers to access the internet in public places, such as coffee shops. Wireless LAN technology is also being used in IoT devices such as smart thermostats and security cameras in order to improve flexibility.

Advantages of WLAN:

1. Installation speed and simplicity.
2. Installation flexibility.
3. Reduced cost of ownership.
4. Reliability.
5. Mobility.
6. Robustness.

Disadvantages of WLAN:

1. Slower bandwidth.
2. Security for wireless LANs is the prime concern.
3. Less capacity.
4. Wireless networks cost four times more than wired network cards.
5. Wireless devices emit low levels of RF which can be harmful to our health.

What are WLAN 'hotspots'?

WLAN 'hotspots' are wireless-enabled areas offering customers access to a broadband internet connection, usually for a usage fee. Such services are becoming common in public areas such as airports, stations, cafes and hotels so that workers can be in regular communication with their business while travelling.

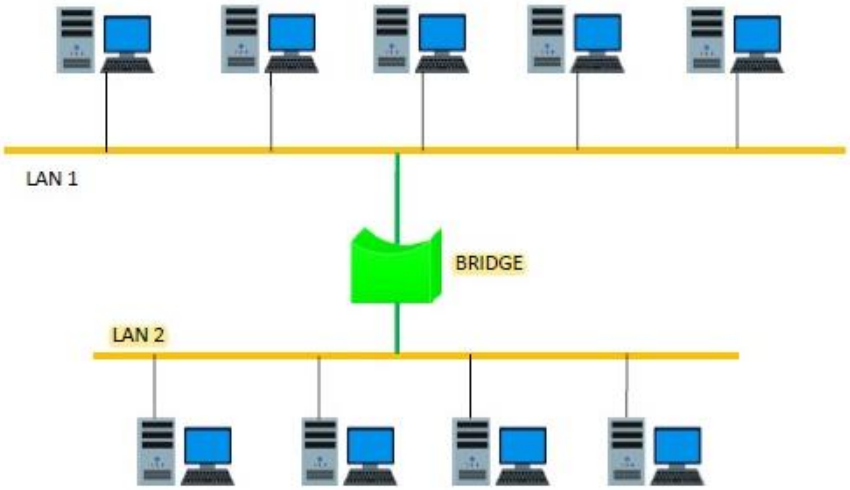
Data Link Layer Switching:

Data Link Layer switching involves forwarding frames (or packets at Layer 2) within the same network segment or LAN (Local Area Network). Switches are the primary devices that operate at this layer and are used to make forwarding decisions based on the MAC (Media Access Control) addresses of devices connected to the network. Switching in data link layer is done by network devices called **bridges**.

Bridges:

A data link layer bridge connects multiple LANs (local area networks) together to form a larger LAN. This process of aggregating networks is called network bridging. A bridge connects the different components so that they appear as parts of a single network.

The following diagram shows connection by a bridge −



Switching by Bridges:

When a data frame arrives at a particular port of a bridge, the bridge examines the frame's data link address, or more specifically, the MAC address. If the destination address as well as the required switching is valid, the bridge sends the frame to the destined port. Otherwise, the frame is discarded.

The bridge is not responsible for end-to-end data transfer. It is concerned with transmitting the data frame from one hop to the next. Hence, they do not examine the payload field of the frame. Due to this, they can help in switching any kind of packets from the network layer above.

Bridges also connect virtual LANs (VLANs) to make a larger VLAN.

If any segment of the bridged network is wireless, a wireless bridge is used to perform the switching.

There are three main ways for bridging −

1. Simple Bridging: Simple bridging, as the name suggests, is the most basic form of bridging. It involves connecting two network segments together to extend the network. In simple bridging:

- The bridge has two ports, each connecting to a separate network segment.
- Frames received on one port are simply forwarded to the other port.
- There is no intelligence to learn MAC addresses or make decisions based on them.
- This type of bridging is often used in scenarios where a direct connection between two LANs or network segments is needed.

2. Multi-Port Bridging: Multi-port bridging extends the concept of simple bridging to connect more than two network segments. In multi-port bridging:

- The bridge has multiple ports, each connecting to a different network segment.
- Frames received on one port are forwarded to all other ports except the one it was received on.
- This type of bridging creates a larger network by interconnecting multiple LANs or segments.
- It helps in reducing collisions by segmenting the network.

3. Learning or Transparent Bridging: Learning or transparent bridging is the most common and widely used form of bridging in modern networks. This type of bridging adds intelligence to the bridge by learning MAC addresses and making forwarding decisions based on this information. Here's how it works:

- When a frame enters the bridge, it reads the source MAC address and associates it with the port on which the frame arrived. This is the process of learning.
- The bridge builds a MAC address table (also known as a bridge table or forwarding table) that maps MAC addresses to the ports they are connected to.
- When a frame with a destination MAC address arrives, the bridge looks up the MAC address in its table:
    - If the MAC address is found, the frame is forwarded only to the port where the destination device is located.
    - If the MAC address is not found, the bridge behaves similar to multi-port bridging, forwarding the frame to all ports except the one it was received on.
- This type of bridging improves network efficiency by only sending frames where they need to go, rather than flooding all ports.

What are the advantages of using Data Link Switching?

Some advantages of using Data Link Switching include the ability to connect geographically distant networks, simplifying network configuration and management, and providing reliable and efficient transport of legacy protocols over modern IP-based networks.

What are the disadvantages of using Data Link Switching?

Disadvantages of using Data Link Switching include potential performance issues due to the additional overhead of encapsulation, limited support from modern network devices, and the necessity for a specialized skill set to troubleshoot and manage the DLSw environment.

What are some alternatives to Data Link Switching?

Some alternatives to Data Link Switching include using native Layer 3 protocols (such as IP or IPX) to replace the need for encapsulating Layer 2 protocols, implementing MPLS-TP (Multi-Protocol Label Switching – Transport Profile) for layer 2 switching over a layer 3 network, or using GRE (Generic Routing Encapsulation) tunneling for transparent transport of layer 2 frames.

Ethernet Bridging:

Ethernet bridging is a technique used to connect multiple network segments together into a single logical network. It is achieved using bridges or more commonly known today as Ethernet switches. Ethernet bridging operates at the Data Link Layer (Layer 2) of the OSI model and allows for the forwarding of Ethernet frames between different network segments based on their MAC addresses. This helps to extend the reach of a local network and improve its efficiency. Here's an overview of Ethernet bridging:

How Ethernet Bridging Works:

1. **MAC Address Learning**:
   - When an Ethernet frame enters an Ethernet bridge, the bridge reads the source MAC address from the frame.
   - It then associates this MAC address with the port on which the frame was received.
   - This information is stored in a MAC address table (also called a forwarding table or bridge table).

2. **Forwarding Decisions**:
    - When an Ethernet frame with a destination MAC address arrives at the bridge, the bridge looks up the destination MAC address in its MAC address table.
    - If the MAC address is found in the table, the bridge forwards the frame out of the appropriate port.
    - If the MAC address is not found, the bridge forwards the frame out of all ports except the one it was received on, similar to a broadcast.

3. **Loop Prevention**:
    - Ethernet bridges use protocols like Spanning Tree Protocol (STP) to prevent loops in the network topology.
    - STP identifies and blocks redundant paths, ensuring there are no loops that could cause broadcast storms or other issues.

Advantages of Ethernet Bridging:

- **Network Extension**: Bridges allow different physical segments of a network to be connected, extending the size of the network.
- **Traffic Isolation**: Different segments can be isolated from each other, preventing unnecessary traffic from crossing over.
- **Efficiency**: Ethernet bridging can improve network efficiency by segmenting traffic and only forwarding frames where they need to go.

Ethernet Bridging vs. Ethernet Switching:

Ethernet bridging and Ethernet switching are often used interchangeably. However, there is a subtle difference:

- **Ethernet Bridging**: This term is often used in the context of connecting two separate networks or segments together. It refers to the process of forwarding Ethernet frames based on MAC addresses to bridge these segments.
- **Ethernet Switching**: This term is more commonly used today to describe the function of modern network switches. Ethernet switches are essentially multi-port bridges that have more features and capabilities. They are responsible for forwarding Ethernet frames within a single network segment based on MAC addresses.

Example Scenario:

Imagine a company with two separate office buildings connected by a bridge. Each building has its own LAN (Local Area Network). When a computer in Building A wants to communicate with a computer in Building B:

- The Ethernet bridge in Building A receives the frame from the sending computer.
- It reads the source MAC address and associates it with the port the frame arrived on.
- The frame is then forwarded across the bridge to Building B based on the destination MAC address.
- The Ethernet bridge in Building B receives the frame and forwards it to the destination computer based on its MAC address.