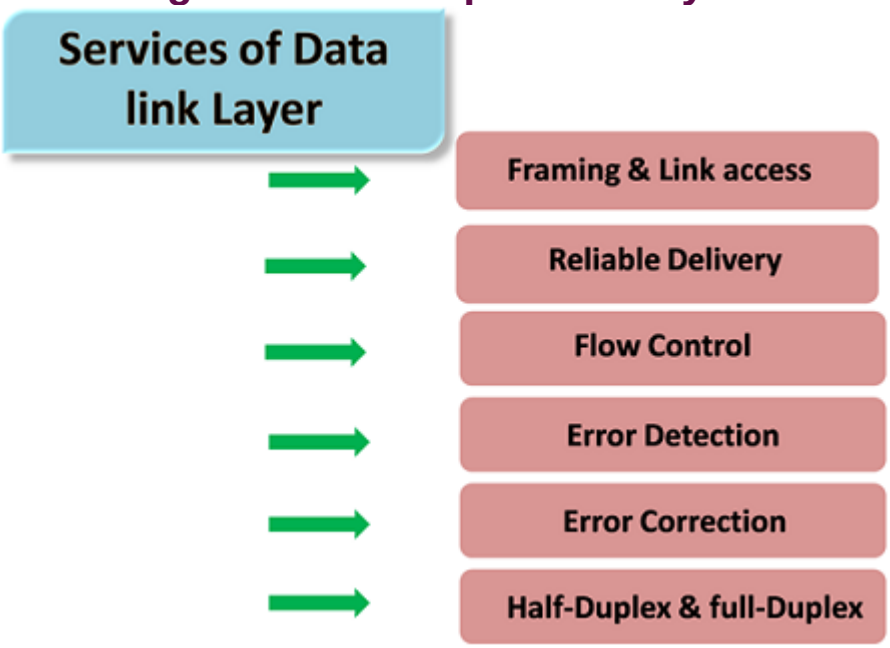
Data Link Layer

- In the OSI model, the data link layer is a 2nd layer from the bottom.
- The communication channel that connects the adjacent nodes is known as links, and in order to move the datagram from source to the destination, the datagram must be moved across an individual link.
- The main responsibility of the Data Link Layer is to transfer the datagram across an individual link.
- The Data link layer protocol defines the format of the packet exchanged across the nodes as well as the actions such as Error detection, retransmission, flow control, and random access.

Following services are provided by the Data Link Layer:



- **Framing & Link access:** Data Link Layer protocols encapsulate each network frame within a Link layer frame before the transmission across the link. A frame consists of a data field in which network layer datagram is inserted and a number of data fields. It specifies the structure of the frame as well as a channel access protocol by which frame is to be transmitted over the link.
- **Reliable delivery:** Data Link Layer provides a reliable delivery service, i.e., transmits the network layer datagram without any error. A reliable delivery service is accomplished with transmissions and acknowledgements. A data link layer mainly provides the reliable delivery service over the links as they have higher error rates and they can be

corrected locally, link at which an error occurs rather than forcing to retransmit the data.

- **Flow control:** A receiving node can receive the frames at a faster rate than it can process the frame. Without flow control, the receiver's buffer can overflow, and frames can get lost. To overcome this problem, the data link layer uses the flow control to prevent the sending node on one side of the link from overwhelming the receiving node on another side of the link.
- **Error detection:** Errors can be introduced by signal attenuation and noise. Data Link Layer protocol provides a mechanism to detect one or more errors. This is achieved by adding error detection bits in the frame and then receiving node can perform an error check.
- **Error correction:** Error correction is similar to the Error detection, except that receiving node not only detect the errors but also determine where the errors have occurred in the frame.
- **Half-Duplex & Full-Duplex:** In a Full-Duplex mode, both the nodes can transmit the data at the same time. In a Half-Duplex mode, only one node can transmit the data at the same time.

Design issues with data link layer are :

Services provided to the network layer – The data link layer act as a service interface to the network layer. The principle service is transferring data from network layer on sending machine to the network layer on destination machine. This transfer also takes place via DLL (Data link-layer).

Frame synchronization – The source machine sends data in the form of blocks called frames to the destination machine. The starting and ending of each frame should be identified so that the frame can be recognized by the destination machine.

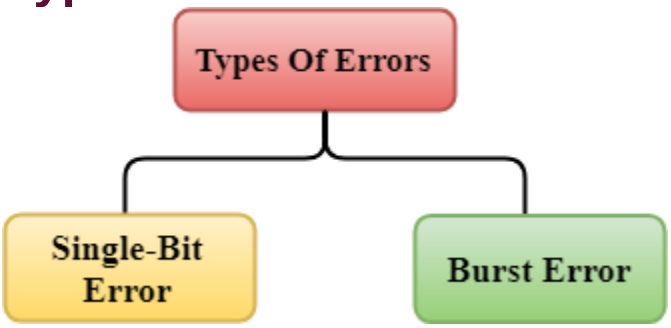
Flow control – Flow control is done to prevent the flow of data frame at the receiver end. The source machine must not send data frames at a rate faster than the capacity of destination machine to accept them.

Error control – Error control is done to prevent duplication of frames. The errors introduced during transmission from source to destination machines must be detected and corrected at the destination machine.

Error Detection

When data is transmitted from one device to another device, the system does not guarantee whether the data received by the device is identical to the data transmitted by another device. An Error is a situation when the message received at the receiver end is not identical to the message transmitted.

Types Of Errors

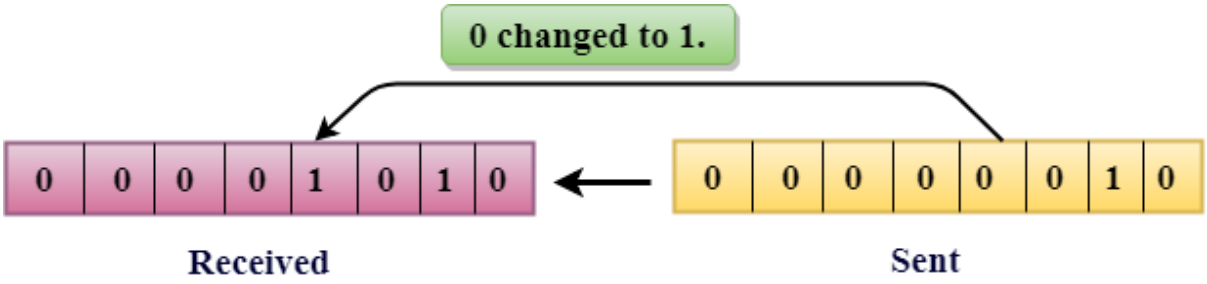


Errors can be classified into two categories:

- Single-Bit Error
- Burst Error

Single-Bit Error:

The only one bit of a given data unit is changed from 1 to 0 or from 0 to 1.



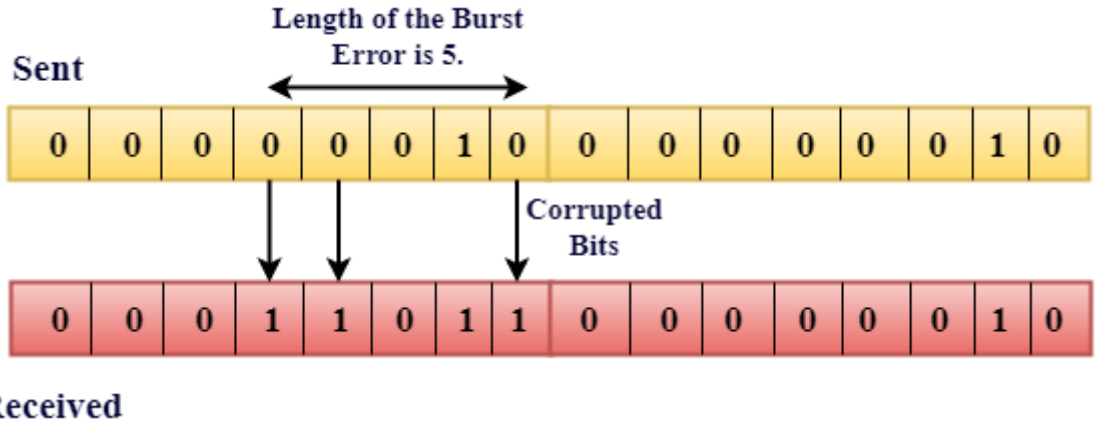
In the above figure, the message which is sent is corrupted as single-bit, i.e., 0 bit is changed to 1.

Single-Bit Error does not appear more likely in Serial Data Transmission. Single-Bit Error mainly occurs in Parallel Data Transmission. For example, if eight wires are used to send the eight bits of a byte, if one of the wire is noisy, then single-bit is corrupted per byte.

Burst Error:

The two or more bits are changed from 0 to 1 or from 1 to 0 is known as Burst Error.

The Burst Error is determined from the first corrupted bit to the last corrupted bit.



The duration of noise in Burst Error is more than the duration of noise in Single-Bit.

Burst Errors are most likely to occurr in Serial Data Transmission.

The number of affected bits depends on the duration of the noise and data rate.

Error Detecting Techniques:

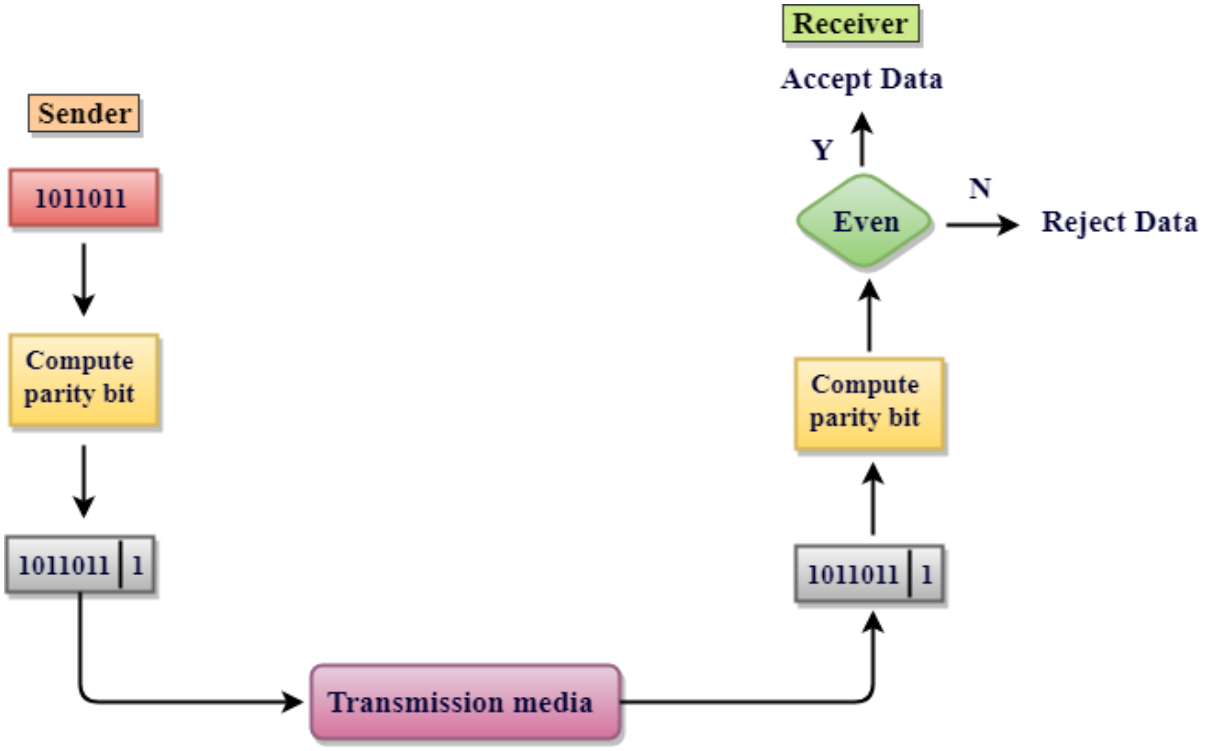
The most popular Error Detecting Techniques are:

- Single parity check
- Two-dimensional parity check
- Checksum
- Cyclic redundancy check

Single Parity Check

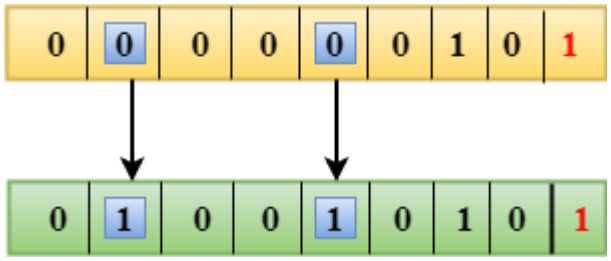
- Single Parity checking is the simple mechanism and inexpensive to detect the errors.
- In this technique, a redundant bit is also known as a parity bit which is appended at the end of the data unit so that the number of 1s becomes even. Therefore, the total number of transmitted bits would be 9 bits.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

- This technique generates the total number of 1s even, so it is known as even-parity checking.



Drawbacks Of Single Parity Checking

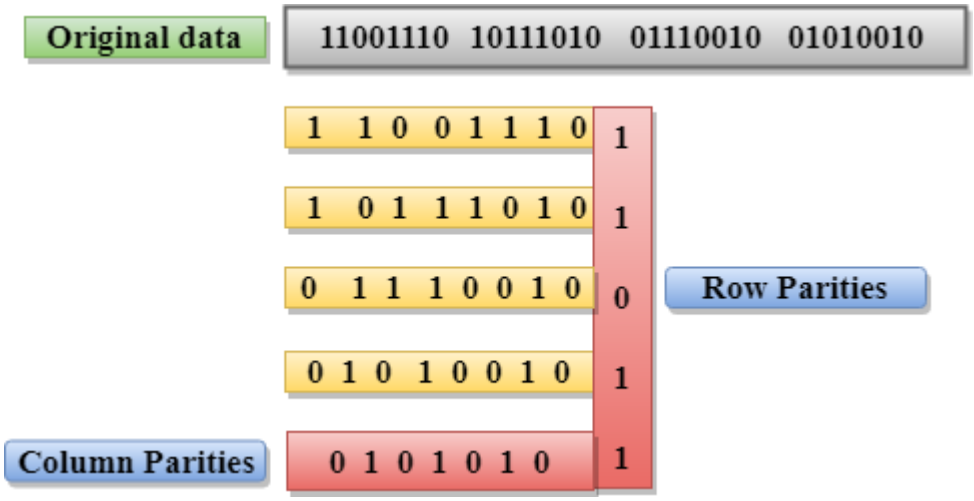
- It can only detect single-bit errors which are very rare.
- If two bits are interchanged, then it cannot detect the errors.



Two-Dimensional Parity Check

- Performance can be improved by using **Two-Dimensional Parity Check** which organizes the data in the form of a table.
- Parity check bits are computed for each row, which is equivalent to the single-parity check.
- In Two-Dimensional Parity check, a block of bits is divided into rows, and the redundant row of bits is added to the whole block.

- At the receiving end, the parity bits are compared with the parity bits computed from the received data.



Drawbacks Of 2D Parity Check

- If two bits in one data unit are corrupted and two bits exactly the same position in another data unit are also corrupted, then 2D Parity checker will not be able to detect the error.
- This technique cannot be used to detect the 4-bit errors or more in some cases.

Checksum

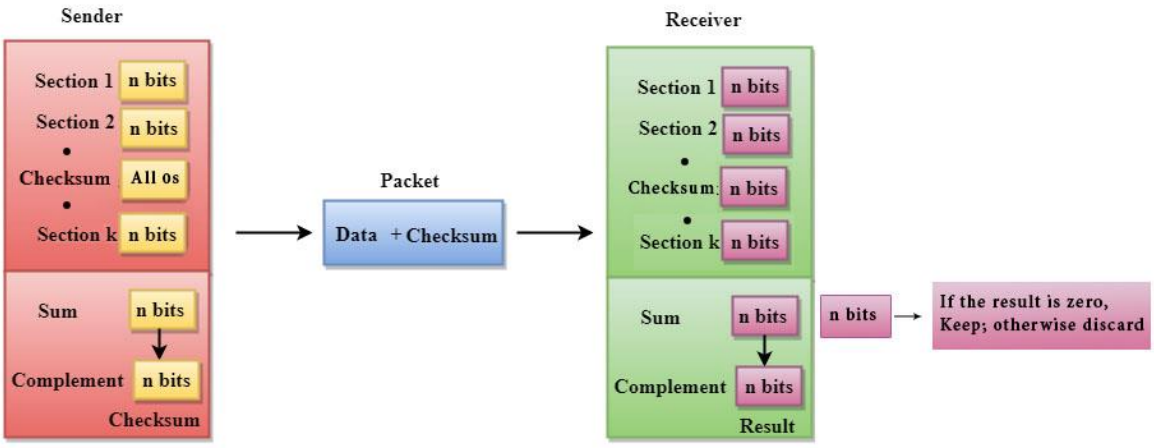
A Checksum is an error detection technique based on the concept of redundancy.

It is divided into two parts:

Checksum Generator

A Checksum is generated at the sending side. Checksum generator subdivides the data into equal segments of n bits each, and all these segments are added together by using one's complement arithmetic. The sum is complemented and appended to the original data, known as checksum field. The extended data is transmitted across the network.

Suppose L is the total sum of the data segments, then the checksum would be ?L



1. The Sender follows the given steps:
2. The block unit is divided into k sections, and each of n bits.
3. All the k sections are added together by using one's complement to get the sum.
4. The sum is complemented and it becomes the checksum field.
5. The original data and checksum field are sent across the network.

Checksum Checker

A Checksum is verified at the receiving side. The receiver subdivides the incoming data into equal segments of n bits each, and all these segments are added together, and then this sum is complemented. If the complement of the sum is zero, then the data is accepted otherwise data is rejected.

1. The Receiver follows the given steps:
2. The block unit is divided into k sections and each of n bits.
3. All the k sections are added together by using one's complement algorithm to get the sum.
4. The sum is complemented.
5. If the result of the sum is zero, then the data is accepted otherwise the data is discarded.

Cyclic Redundancy Check (CRC)

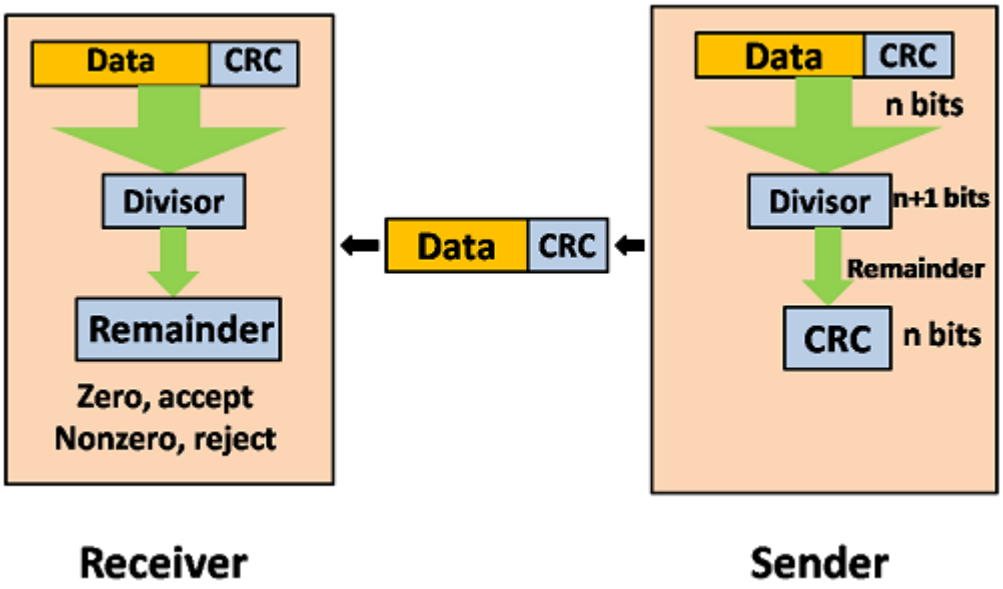
CRC is a redundancy error technique used to determine the error.

Following are the steps used in CRC for error detection:

- In CRC technique, a string of n 0s is appended to the data unit, and this n number is less than the number of bits in a predetermined number, known as division which is n+1 bits.
- Secondly, the newly extended data is divided by a divisor using a process is known as binary division. The remainder generated from this division is known as CRC remainder.
- Thirdly, the CRC remainder replaces the appended 0s at the end of the original data. This newly generated unit is sent to the receiver.
- The receiver receives the data followed by the CRC remainder. The receiver will treat this whole unit as a single unit, and it is divided by the same divisor that was used to find the CRC remainder.

If the resultant of this division is zero which means that it has no error, and the data is accepted.

If the resultant of this division is not zero which means that the data consists of an error. Therefore, the data is discarded.

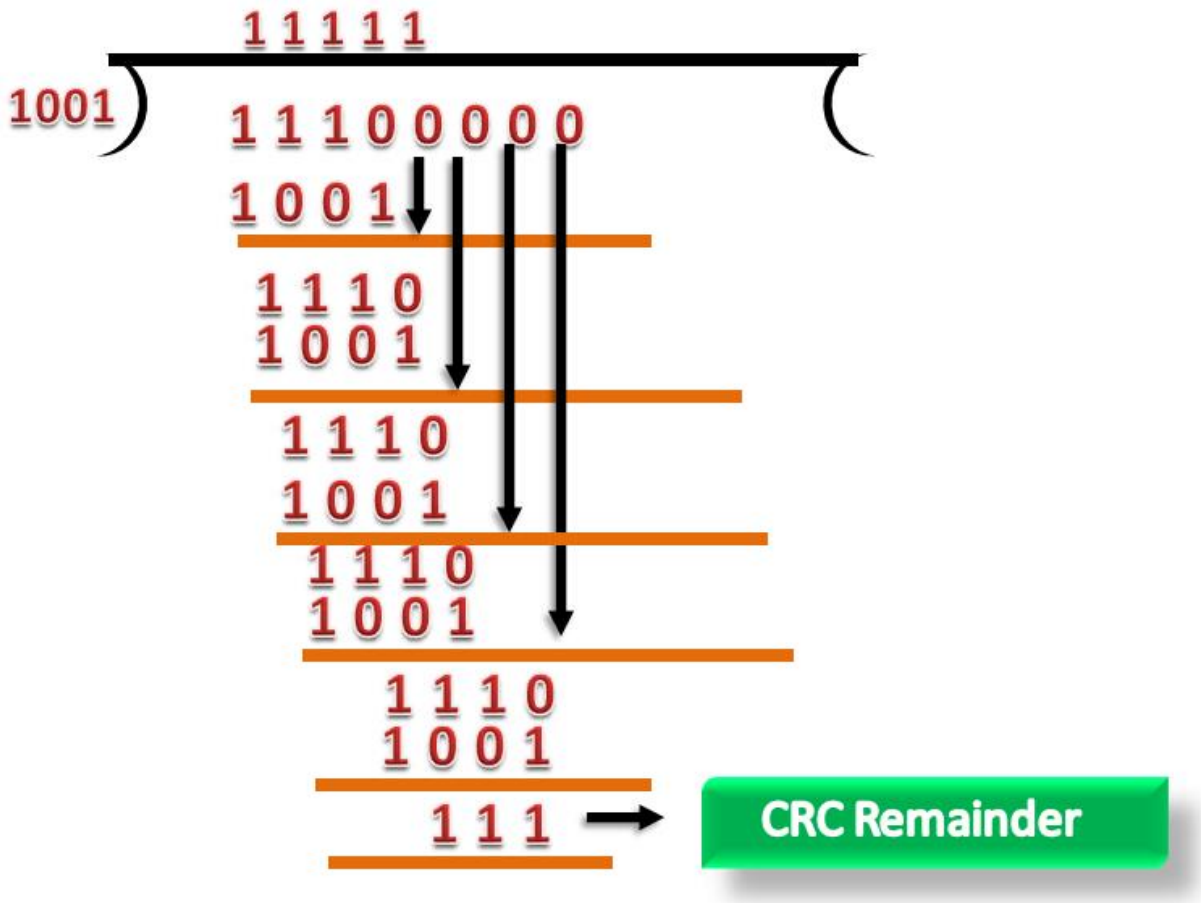


Let's understand this concept through an example:

Suppose the original data is 11100 and divisor is 1001.

CRC Generator

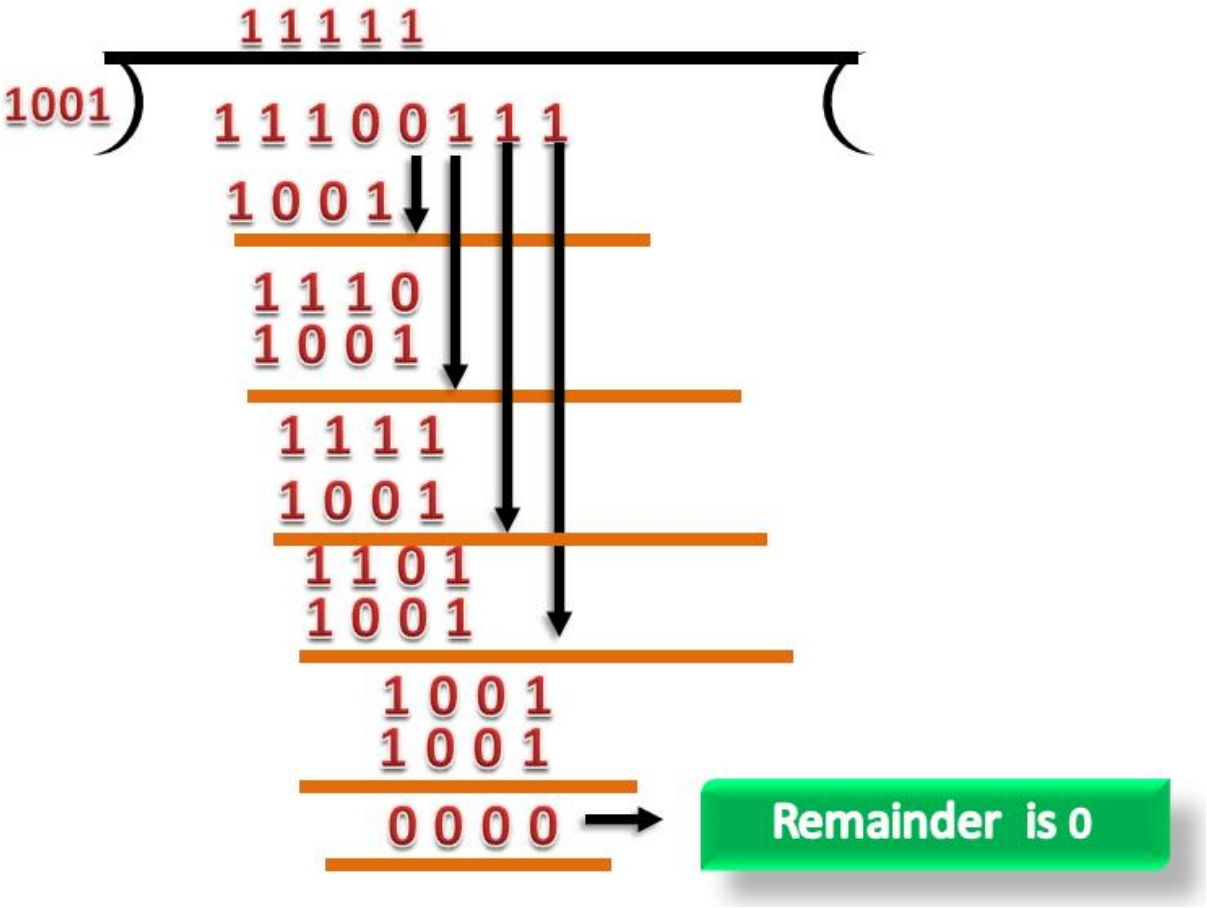
- A CRC generator uses a modulo-2 division. Firstly, three zeroes are appended at the end of the data as the length of the divisor is 4 and we know that the length of the string 0s to be appended is always one less than the length of the divisor.
- Now, the string becomes 11100000, and the resultant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



CRC Checker

- The functionality of the CRC checker is similar to the CRC generator.
- When the string 11100111 is received at the receiving end, then CRC checker performs the modulo-2 division.
- A string is divided by the same divisor, i.e., 1001.

- In this case, CRC checker generates the remainder of zero. Therefore, the data is accepted.



Error Correction

Error Correction codes are used to detect and correct the errors when data is transmitted from the sender to the receiver.

Error Correction can be handled in two ways:

- **Backward error correction:** Once the error is discovered, the receiver requests the sender to retransmit the entire data unit.
- **Forward error correction:** In this case, the receiver uses the error-correcting code which automatically corrects the errors.

A single additional bit can detect the error, but cannot correct it.

For correcting the errors, one has to know the exact position of the error. For example, If we want to calculate a single-bit error, the error correction code will determine which one of seven bits is in error. To achieve this, we have to add some additional redundant bits.

Suppose r is the number of redundant bits and d is the total number of the data bits. The number of redundant bits r can be calculated by using the formula:

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

The value of r is calculated by using the above formula. For example, if the value of d is 4, then the possible smallest value that satisfies the above relation would be 3.

To determine the position of the bit which is in error, a technique developed by R.W Hamming is Hamming code which can be applied to any length of the data unit and uses the relationship between data units and redundant units.

Hamming Code

Parity bits: The bit which is appended to the original data of binary bits so that the total number of 1s is even or odd.

Even parity: To check for even parity, if the total number of 1s is even, then the value of the parity bit is 0. If the total number of 1s occurrences is odd, then the value of the parity bit is 1.

Odd Parity: To check for odd parity, if the total number of 1s is even, then the value of parity bit is 1. If the total number of 1s is odd, then the value of parity bit is 0.

Algorithm of Hamming code:

- An information of 'd' bits are added to the redundant bits 'r' to form d+r.
- The location of each of the (d+r) digits is assigned a decimal value.
- The 'r' bits are placed in the positions 1,2,.....2^{k-1}.
- At the receiving end, the parity bits are recalculated. The decimal value of the parity bits determines the position of an error.

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

Let's understand the concept of Hamming code through an example:

Suppose the original data is 1010 which is to be sent.

Total number of data bits 'd' = 4

Number of redundant bits r : $2^r >= d+r+1$

$2^r >= 4+r+1$

Therefore, the value of r is 3 that satisfies the above relation.

Total number of bits = d+r = 4+3 = 7;

Determining the position of the redundant bits

The number of redundant bits is 3. The three bits are represented by r1, r2, r4. The position of the redundant bits is calculated with corresponds to the raised power of 2. Therefore, their corresponding positions are **1, 2^1, 2^2**.

1. The position of r1 = 1
2. The position of r2 = 2
3. The position of r4 = 4

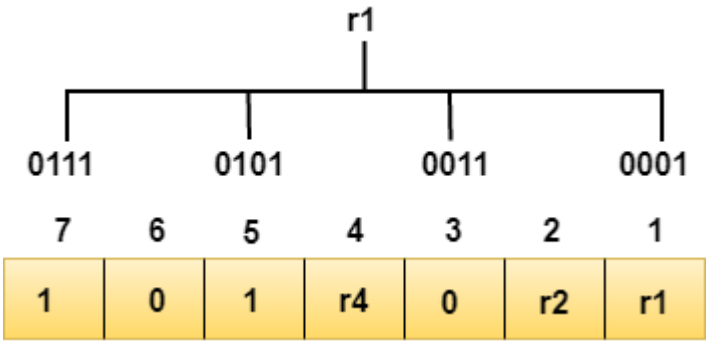
Representation of Data on the addition of parity bits:

7	6	5	4	3	2	1
1	0	1	r4	0	r2	r1

Determining the Parity bits

Determining the r1 bit

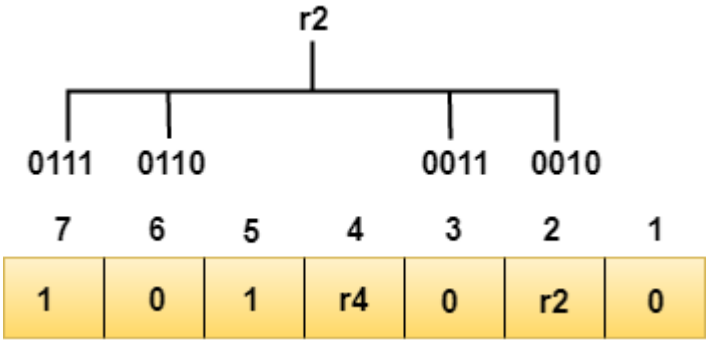
The r1 bit is calculated by performing a parity check on the bit positions whose binary representation includes 1 in the first position.



We observe from the above figure that the bit positions that includes 1 in the first position are 1, 3, 5, 7. Now, we perform the even-parity check at these bit positions. The total number of 1 at these bit positions corresponding to r1 is **even, therefore, the value of the r1 bit is 0**.

Determining r2 bit

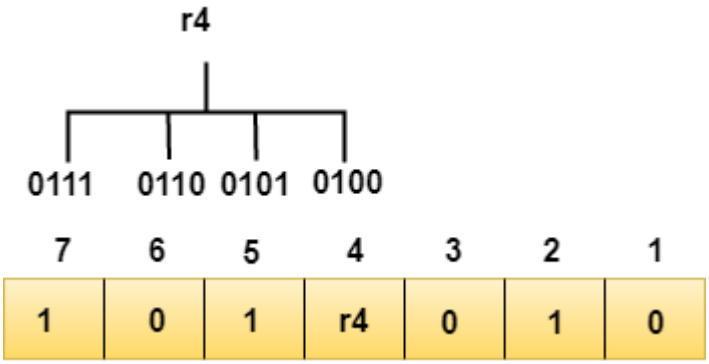
The r2 bit is calculated by performing a parity check on the bit positions whose binary representation includes 1 in the second position.



We observe from the above figure that the bit positions that includes 1 in the second position are **2, 3, 6, 7**. Now, we perform the even-parity check at these bit positions. The total number of 1 at these bit positions corresponding to r2 is **odd, therefore, the value of the r2 bit is 1**.

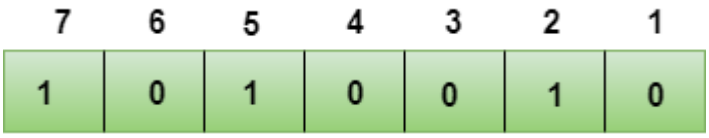
Determining r4 bit

The r4 bit is calculated by performing a parity check on the bit positions whose binary representation includes 1 in the third position.



We observe from the above figure that the bit positions that includes 1 in the third position are **4, 5, 6, 7**. Now, we perform the even-parity check at these bit positions. The total number of 1 at these bit positions corresponding to r4 is **even, therefore, the value of the r4 bit is 0**.

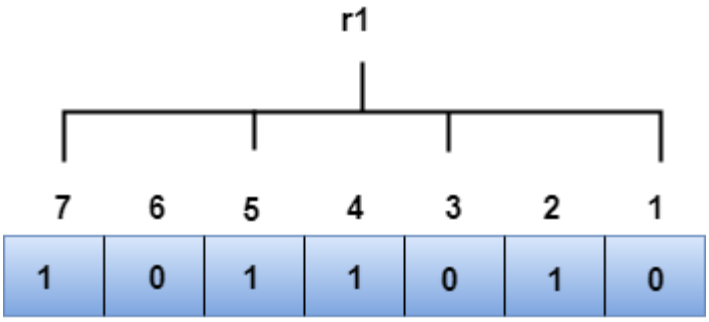
Data transferred is given below:



Suppose the 4th bit is changed from 0 to 1 at the receiving end, then parity bits are recalculated.

R1 bit

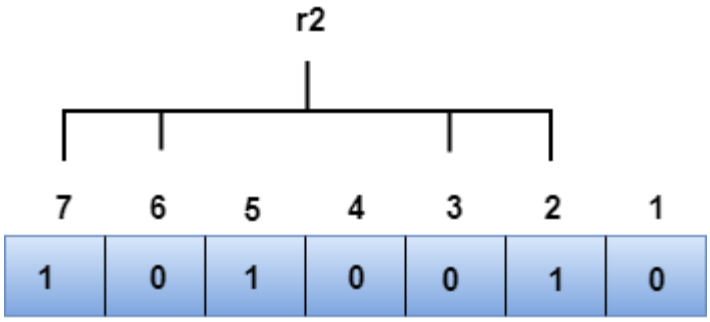
The bit positions of the r1 bit are 1,3,5,7



We observe from the above figure that the binary representation of r1 is 1100. Now, we perform the even-parity check, the total number of 1s appearing in the r1 bit is an even number. Therefore, the value of r1 is 0.

R2 bit

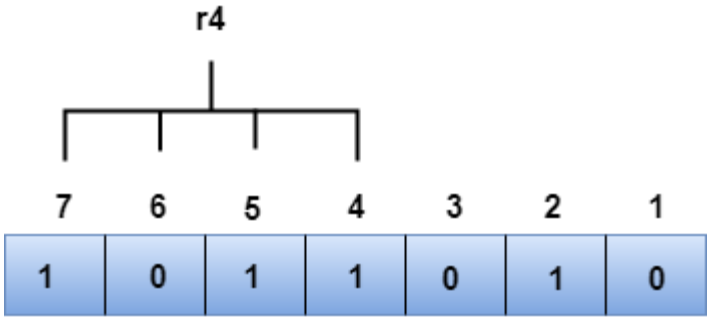
The bit positions of r2 bit are 2,3,6,7.



We observe from the above figure that the binary representation of r2 is 1001. Now, we perform the even-parity check, the total number of 1s appearing in the r2 bit is an even number. Therefore, the value of r2 is 0.

R4 bit

The bit positions of r4 bit are 4,5,6,7.



We observe from the above figure that the binary representation of r4 is 1011. Now, we perform the even-parity check, the total number of 1s appearing in the r4 bit is an odd number. Therefore, the value of r4 is 1.

- *The binary representation of redundant bits, i.e., r4r2r1 is 100, and its corresponding decimal value is 4. Therefore, the error occurs in a 4^{th} bit position. The bit value must be changed from 1 to 0 to correct the error.*

Stop and Wait ARQ

Characteristics

- Used in Connection-oriented communication.
- It offers error and flows control
- It is used in Data Link and Transport Layers
- Stop and Wait for ARQ mainly implements the Sliding Window Protocol concept with Window Size 1

Simple Stop and Wait

Sender:

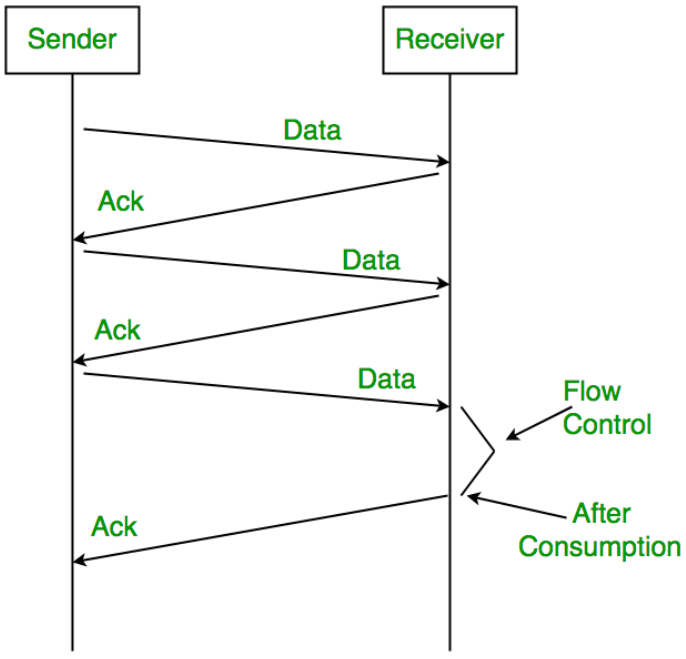
Rule 1) Send one data packet at a time.

Rule 2) Send the next packet only after receiving acknowledgement for the previous.

Receiver:

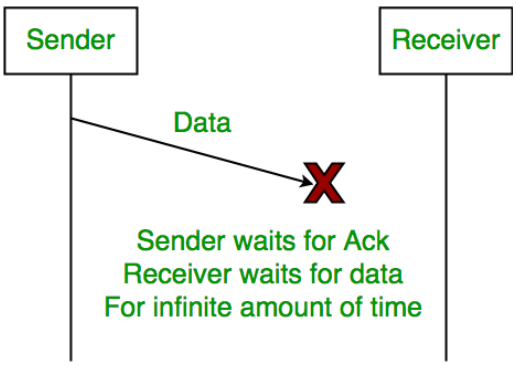
Rule 1) Send acknowledgement after receiving and consuming a data packet.

Rule 2) After consuming packet acknowledgement need to be sent (Flow Control)

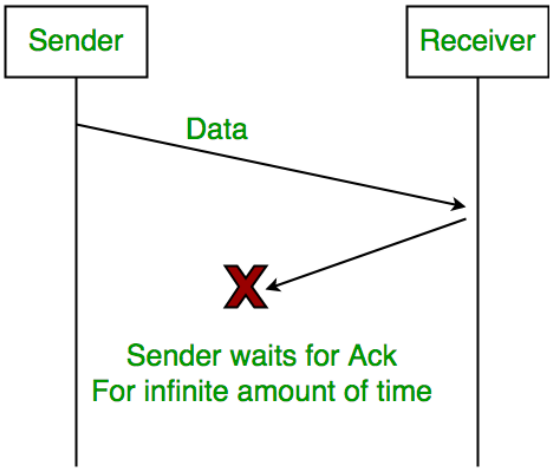


Problems :

1. Lost Data



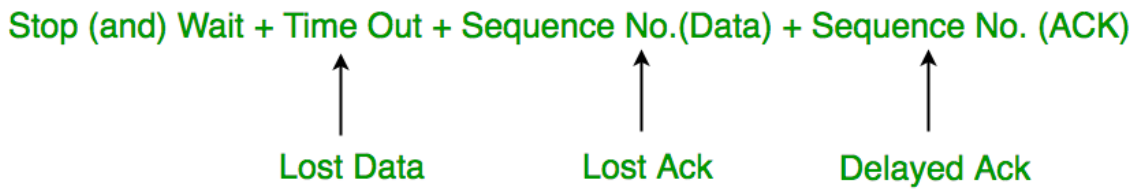
2. Lost Acknowledgement:



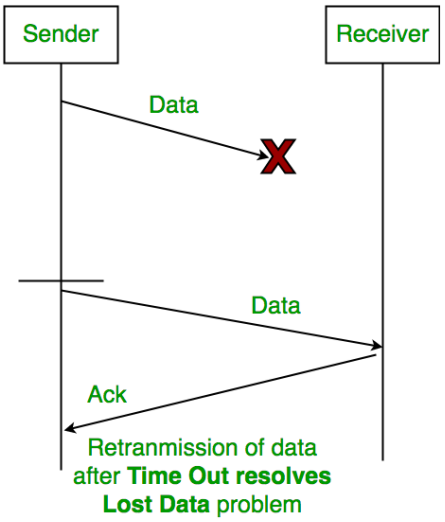
3. Delayed Acknowledgement/Data: After a timeout on the sender side, a long-delayed acknowledgement might be wrongly considered as acknowledgement of some other recent packet.

Stop and Wait for ARQ (Automatic Repeat Request)

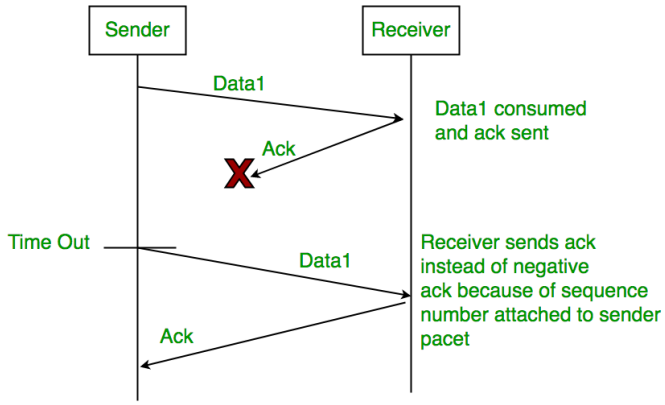
The above 3 problems are resolved by Stop and Wait for ARQ (Automatic Repeat Request) that does both error control and flow control.



1. Time Out:



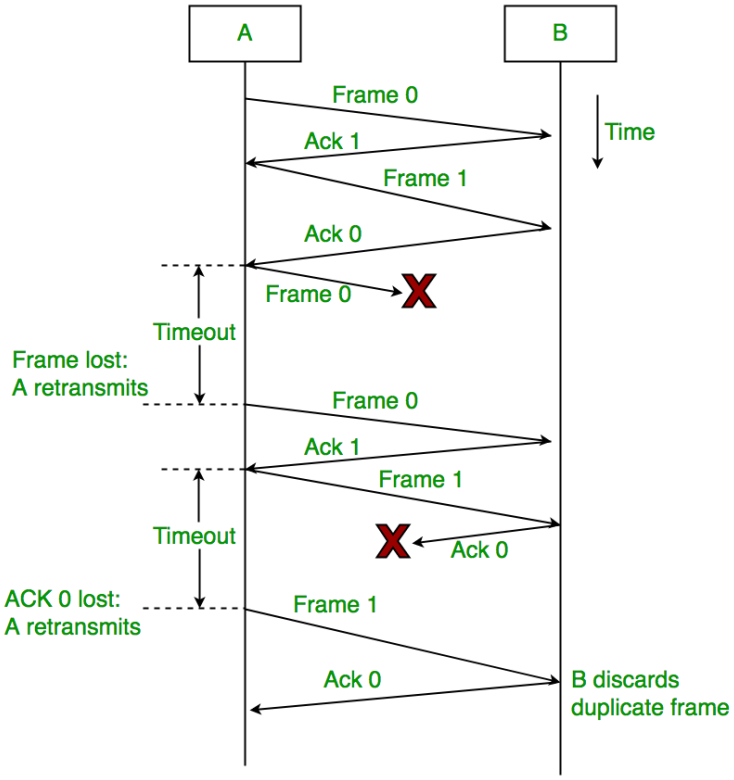
2. Sequence Number (Data)



3. Delayed Acknowledgement:
This is resolved by introducing sequence numbers for acknowledgement also.

Working of Stop and Wait for ARQ:

1) Sender A sends a data frame or packet with sequence number 0.
2) Receiver B, after receiving the data frame, sends an acknowledgement with sequence number 1 (the sequence number of the next expected data frame or packet)
There is only a one-bit sequence number that implies that both sender and receiver have a buffer for one frame or packet only.



Sliding Window Protocol

The sliding window is a technique for sending multiple frames at a time. It controls the data packets between the two devices where reliable and gradual delivery of data frames is needed. It is also used in TCP (Transmission Control Protocol).

In this technique, each frame has sent from the sequence number. The sequence numbers are used to find the missing data in the receiver end. The purpose of the sliding window technique is to avoid duplicate data, so it uses the sequence number.

Types of Sliding Window Protocol

Sliding window protocol has two types:

1. Go-Back-N ARQ
2. Selective Repeat ARQ

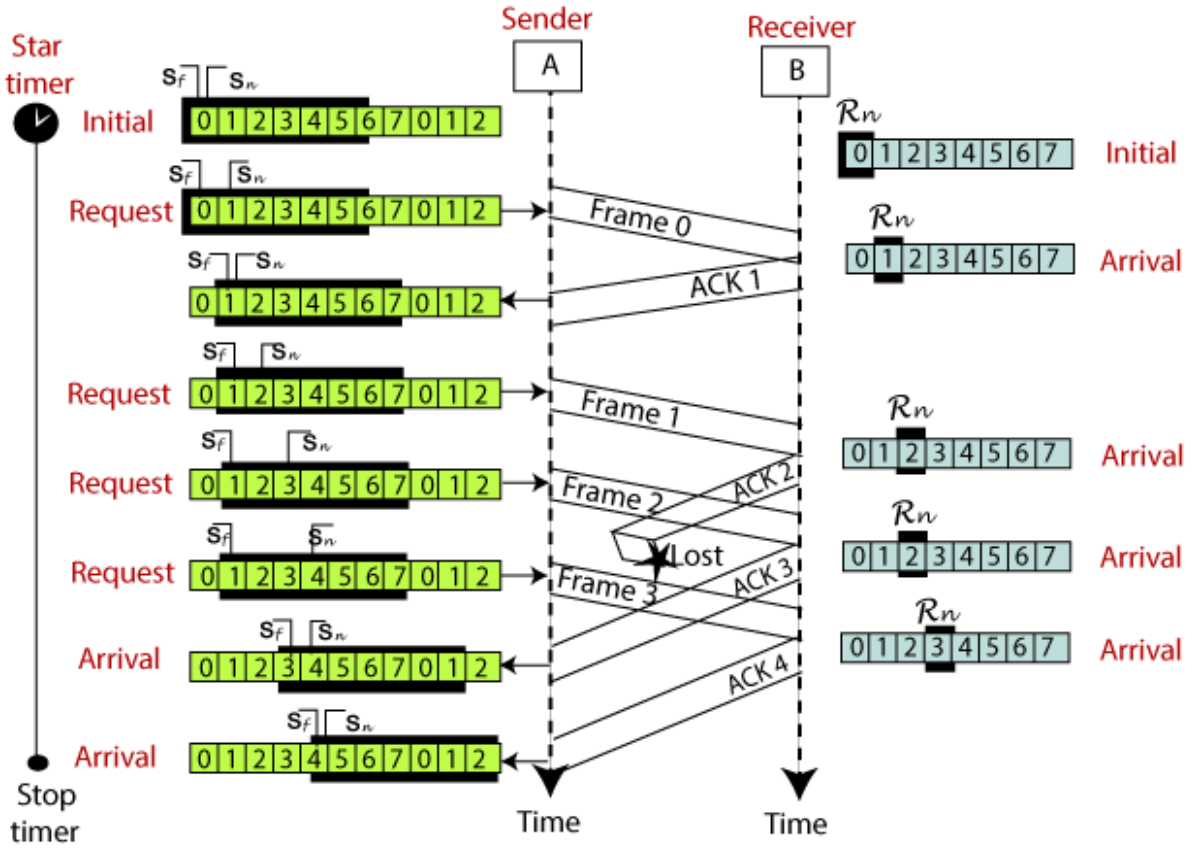
Go-Back-N ARQ

Go-Back-N ARQ protocol is also known as Go-Back-N Automatic Repeat Request. It is a data link layer protocol that uses a sliding window method. In this, if any frame is corrupted or lost, all subsequent frames have to be sent again.

The size of the sender window is N in this protocol. For example, Go-Back-8, the size of the sender window, will be 8. The receiver window size is always 1.

If the receiver receives a corrupted frame, it cancels it. The receiver does not accept a corrupted frame. When the timer expires, the sender sends the correct frame again.

The example of Go-Back-N ARQ is shown below in the figure.

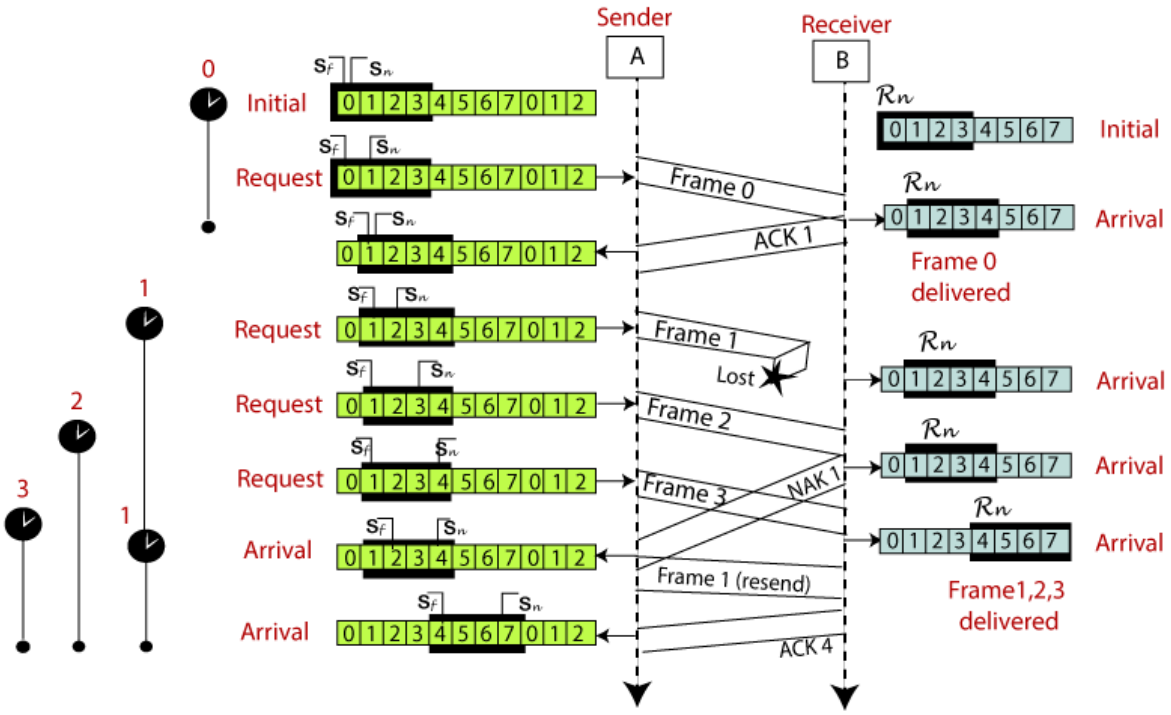


Selective Repeat ARQ

Selective Repeat ARQ is also known as the Selective Repeat Automatic Repeat Request. It is a data link layer protocol that uses a sliding window method. The Go-back-N ARQ protocol works well if it has fewer errors. But if there is a lot of error in the frame, lots of bandwidth loss in sending the frames again. So, we use the Selective Repeat ARQ protocol. In this protocol, the size of the sender window is always equal to the size of the receiver window. The size of the sliding window is always greater than 1.

If the receiver receives a corrupt frame, it does not directly discard it. It sends a negative acknowledgment to the sender. The sender sends that frame again as soon as on the receiving negative acknowledgment. There is no waiting for any time-out to send that frame.

The example of the Selective Repeat ARQ protocol is shown below in the figure.



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

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

HDLC Protocol

HDLC (High-Level Data Link Control) is a bit-oriented protocol that is used for communication over the **point-to-point and multipoint links**. This protocol implements the mechanism of ARQ(Automatic Repeat Request). With the help of the HDLC protocol,full-duplex communication is possible.

HDLC is the most widely used protocol and offers reliability, efficiency, and a high level of Flexibility.

In order to make the HDLC protocol applicable for various network configurations, there are three types of stations and these are as follows:

- **Primary Station** This station mainly looks after data like management. In the case of the communication between the primary and secondary station; it is the responsibility of the primary station to connect and disconnect the data link. The frames issued by the primary station are commonly known as commands.
- **Secondary Station** The secondary station operates under the control of the primary station. The Frames issued by the secondary stations are commonly known as responses.
- **Combined Station** The combined station acts as both Primary stations as well as Secondary stations. The combined station issues both commands as well as responses.

Transfer Modes in HDLC

- Normal Response Mode(NRM)
- Asynchronous Balance Mode(ABM)

1. Normal Response Mode(NRM)

In this mode, the configuration of the station is unbalanced. There are one primary station and multiple secondary stations. Where the primary station can send the commands and the secondary station can only respond.

This mode is used for both **point-to-point** as well as **multiple-point links.**

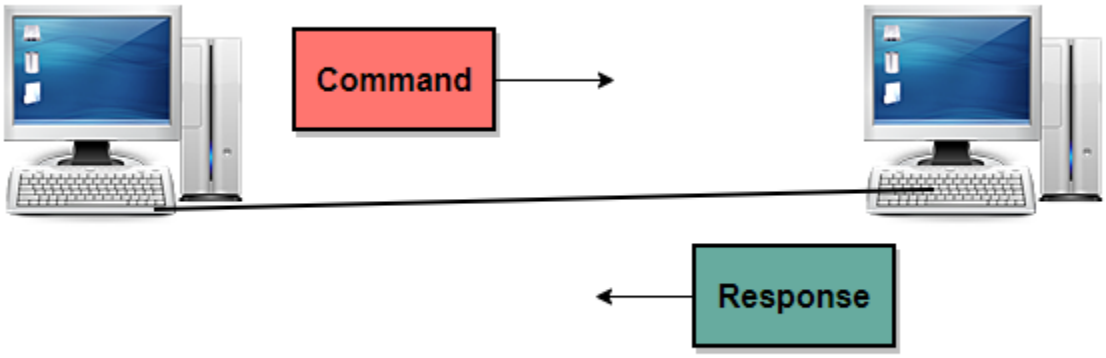


Figure: Point-to-Point

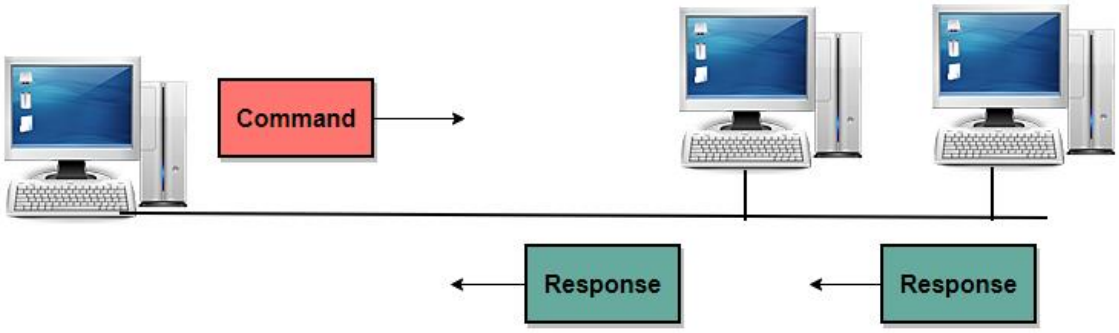


Figure: Multipoint

2. Asynchronous Balance Mode(ABM)

In this mode, the configuration of the station is balanced. In this mode, the link is point-to-point, and each station can function as a primary and as secondary.

Asynchronous Balance mode(ABM) is a commonly used mode today.

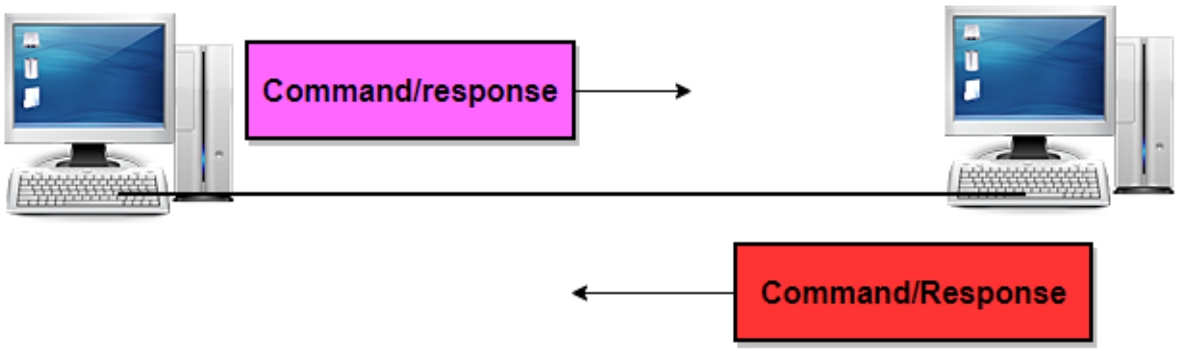
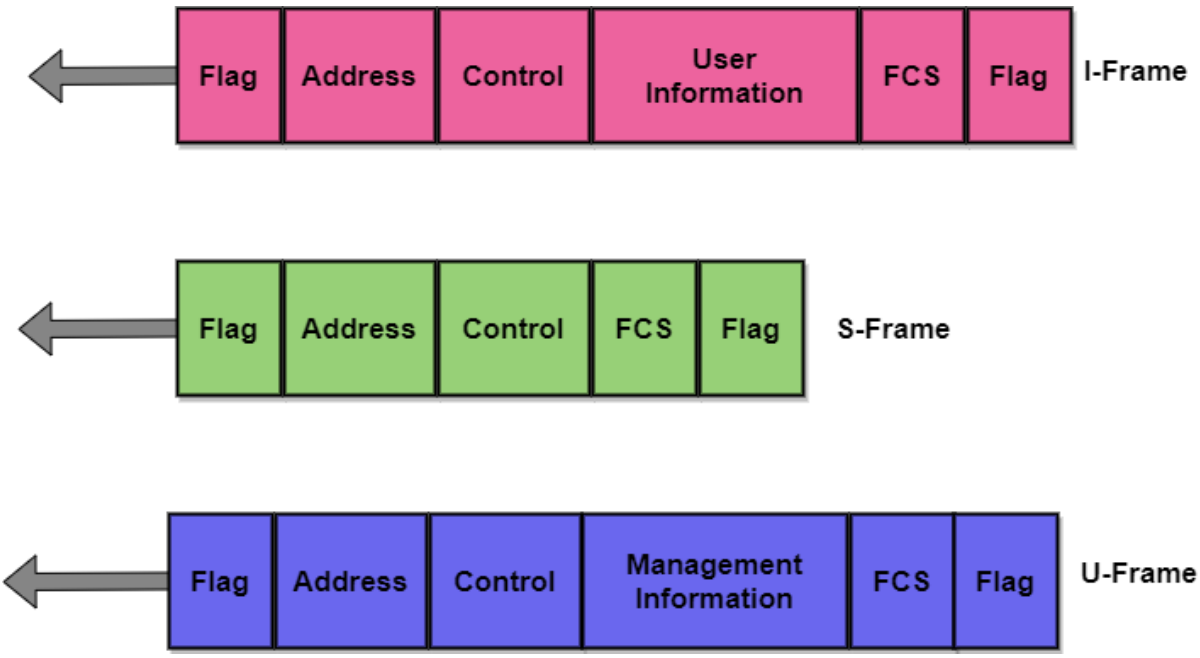


Figure: Asynchronous Balance Mode

HDLC Frames

- **Information Frames(I-frames)** These frames are used to transport the user data and the control information that is related to the user data. If the first bit of the control field is 0 then it is identified as I-frame.
- **Supervisory Frames(S-frames)** These frames are only used to transport the control information. If the first two bits of the control field are 1 and 0 then the frame is identified as S-frame
- **Unnumbered Frames(U-Frames)** These frames are mainly reserved for system management. These frames are used for exchanging control information between the communicating devices.

Frame Format



Features of HDLC Protocol

1.This protocol uses bits to stuff flags occurring in the data.

2.This protocol is used for point-to-point as well as multipoint link access.

3.HDLC is one of the most common protocols of the data link layer.

4.HDLC is a bit-oriented protocol.

5.This protocol implements error control as well as flow control.

Multiple Access Protocols in Computer Network

The Data Link Layer is responsible for transmission of data between two nodes. Its main functions are-

- Data Link Control
- Multiple Access Control

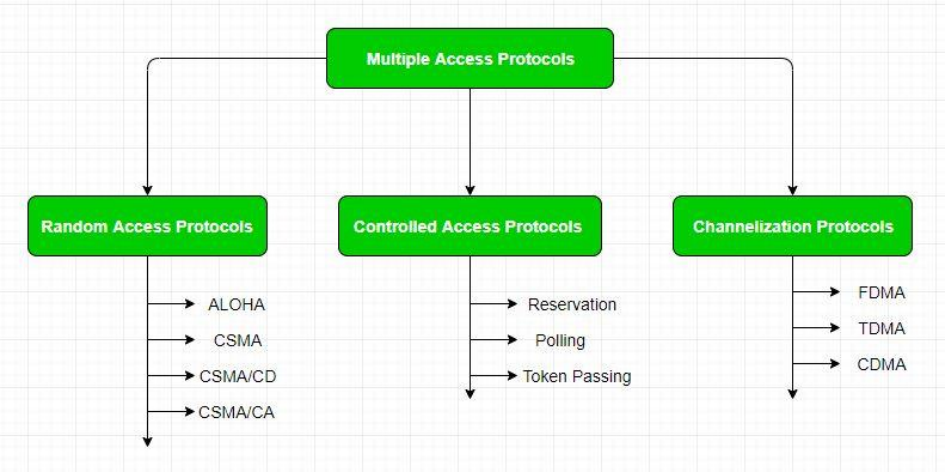
Data Link control –

The data link control is responsible for reliable transmission of message over transmission channel by using techniques like framing, error control and flow control. For Data link control refer to – Stop and Wait ARQ

Multiple Access Control –

If there is a dedicated link between the sender and the receiver then data link control layer is sufficient, however if there is no dedicated link present then multiple stations can access the channel simultaneously. Hence multiple access protocols are required to decrease collision and avoid crosstalk. For example, in a classroom full of students, when a teacher asks a question and all the students (or stations) start answering simultaneously (send data at same time) then a lot of chaos is created(data overlap or data lost) then it is the job of the teacher (multiple access protocols) to manage the students and make them answer one at a time.

Multiple access protocols can be subdivided further as –



What is the hub?

A hub is a common connection point, also known as a network hub, which is used for connection of devices in a network. It works as a central connection for all the devices that are connected through a hub. The hub has numerous ports.

Types of Hub

There are three types of the hub that are given below:

1. Passive Hub
2. Active Hub
3. Intelligent Hub

Passive Hub: The passive hubs are the connection point for wires that helps to make the physical network. It is capable of determining the bugs and faulty hardware. Simply, it accepts the packet over a port and circulates it to all ports.

Active Hub: As compared to a passive hub, it includes some additional features. It is able to monitor the data sent to the connected devices. It plays an important role between the connected devices with the help of store technology, where it checks the data to be sent and decides which packet to send first.

Intelligent Hub: It is a little smarter than passive and active hubs. These hubs have some kinds of management software that help to analyze the problem in the network and resolve them. It is beneficial to expend the business in networking; the management can assign users that help to work more quickly and share a common pool efficiently by using intelligent hubs.

Features of Hub

- It acts with shared bandwidth and broadcasting.
- It includes only one collision domain and broadcast domain.

- It works at the physical layer of the OSI model and also offers support for half-duplex transmission mode.
- It cannot create a virtual LAN and does not support spanning tree protocol.
- Furthermore, mainly packet collisions occur inside the hub.

Applications of Hub

The important applications of a hub are given below:

- Hub is used to create small home networks.
- It is used for network monitoring.
- They are also used in organizations to provide connectivity.

Bridge in Computer Network:

A bridge in a computer network is a device used to connect multiple LANs together with a larger Local Area Network (LAN). The mechanism of network aggregation is known as bridging. The bridge is a physical or hardware device but operates at the OSI model's data link layer and is also known as a layer of two switches.

The primary responsibility of a switch is to examine the incoming traffic and determine whether to filter or forward it. Basically, a bridge in computer networks is used to divide network connections into sections, now each section has separate bandwidth and a separate collision domain. Here bridge is used to improve network performance.

Types of Bridges:

There are three types of bridges in computer networks, which are as follows:

1. Transparent bridge
2. Source routing bridge
3. Translational bridge

Transparent Bridge:

Transparent bridges are invisible to other devices on the network. This bridge doesn't reconfigure the network on the addition or deletion of any station. The prime function of the transparent bridge is to block or forward the data according to the MAC address.

Source Routing Bridge:

Source routing bridges were developed and designed by IBM specifically for token ring networks. The frame's entire route is embedded with the data frames by the source station to perform the routing operation so that once the frame is forwarded it must follow a specific defined path/route.

Translational Bridge:

Translational bridges convert the received data from one networking system to another. Or it is used to communicate or transmit data between two different types of networking systems. Like if we are

sending data from a token ring to an Ethernet cable, the translational cable will be used to connect both the networking system and transmit data.

Advantages:

- Bridges can be used as a network extension like they can connect two network topologies together.
- It has a separate collision domain, which results in increased bandwidth.
- It can create a buffer when different MAC protocols are there for different segments.
- Highly reliable and maintainable. The network can be divided into multiple LAN segments.
- Simple installation, no requirement of any extra hardware or software except the bridge itself.

Disadvantages:

- Expensive as compared to hubs and repeaters.
- Slow in speed.
- Poor performance as additional processing is required to view the MAC address of the device on the network.
- As the traffic received is in bulk or is broadcasted traffic, individual filtering of data is not possible.
- During the broadcasting of data, the network has high broadcast traffic and broadcast storms can be formed.

Functions of Bridges in the Network

- The bridge is used to divide LANs into multiple segments.
- To control the traffic in the network.
- It can interconnect two LANs with a similar protocols.
- It can filter the data based on destination/MAC address.

Types of switches in Computer Network

Switches are the connectivity points of an Ethernet network. These are small devices that can receive data from multiple input ports and send it to the specific output port that takes data to its intended destination in the network. There are different types of switches in a network. These are:

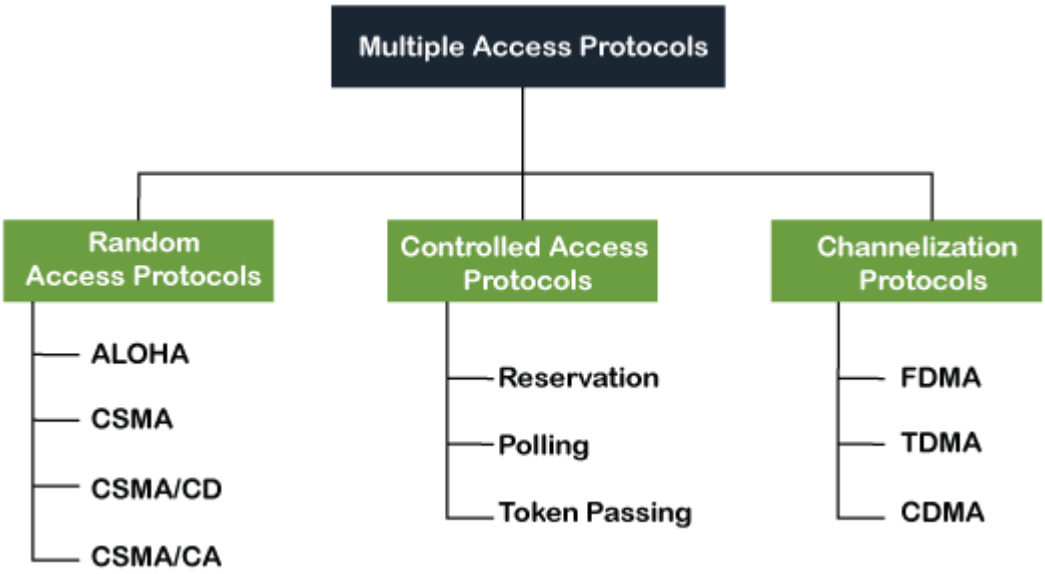
1. **Unmanaged switches –**
   These are the switches that are mostly used in home networks and small businesses as they plug in and instantly start doing their job and such switches do not need to be watched or configured. These require only small cable connections. It allows devices on a network to connect with each other such as a computer to a computer or a computer to a printer in one location. They are the least expensive switches among all categories.

2. **Managed switches –**
   These types of switches have many features like the highest levels of security, precision control, and full management of the network. These are used in organizations containing a large network and can be customized to enhance the functionality of a certain network. These

are the most costly option but their scalability makes them an ideal option for a network that is growing.

3. **LAN switches –**
   These are also known as Ethernet switches or data switches and are used to reduce network congestion or bottleneck by distributing a package of data only to its intended recipient. These are used to connect points on a LAN.

4. **PoE switches –**
   PoE switches are used in PoE technology which stands for power over Ethernet that is a technology that integrates data and power on the same cable allowing power devices to receive data in parallel to power. Thus these switches provide greater flexibility by simplifying the cabling process.



A. Random Access Protocol

In this protocol, all the station has the equal priority to send the data over a channel. In random access protocol, one or more stations cannot depend on another station nor any station control another station. Depending on the channel's state (idle or busy), each station transmits the data frame. However, if more than one station sends the data over a channel, there may be a collision or data conflict. Due to the collision, the data frame packets may be lost or changed. And hence, it does not receive by the receiver end.

Following are the different methods of random-access protocols for broadcasting frames on the channel.

- Aloha

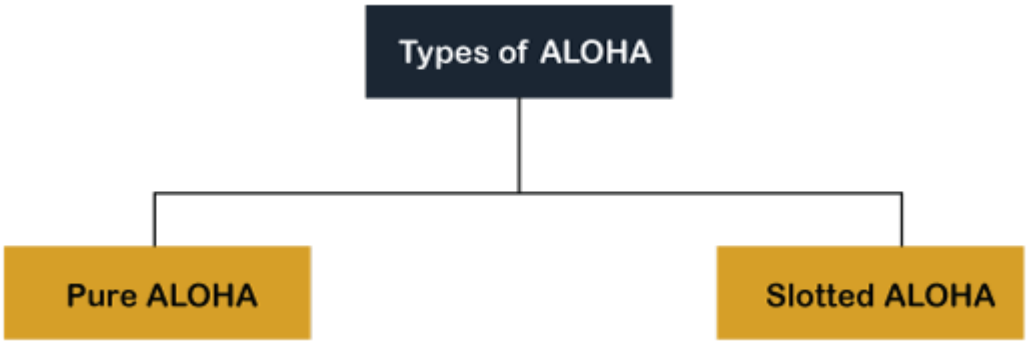
- CSMA
- CSMA/CD
- CSMA/CA

ALOHA Random Access Protocol

It is designed for wireless LAN (Local Area Network) but can also be used in a shared medium to transmit data. Using this method, any station can transmit data across a network simultaneously when a data frameset is available for transmission.

Aloha Rules

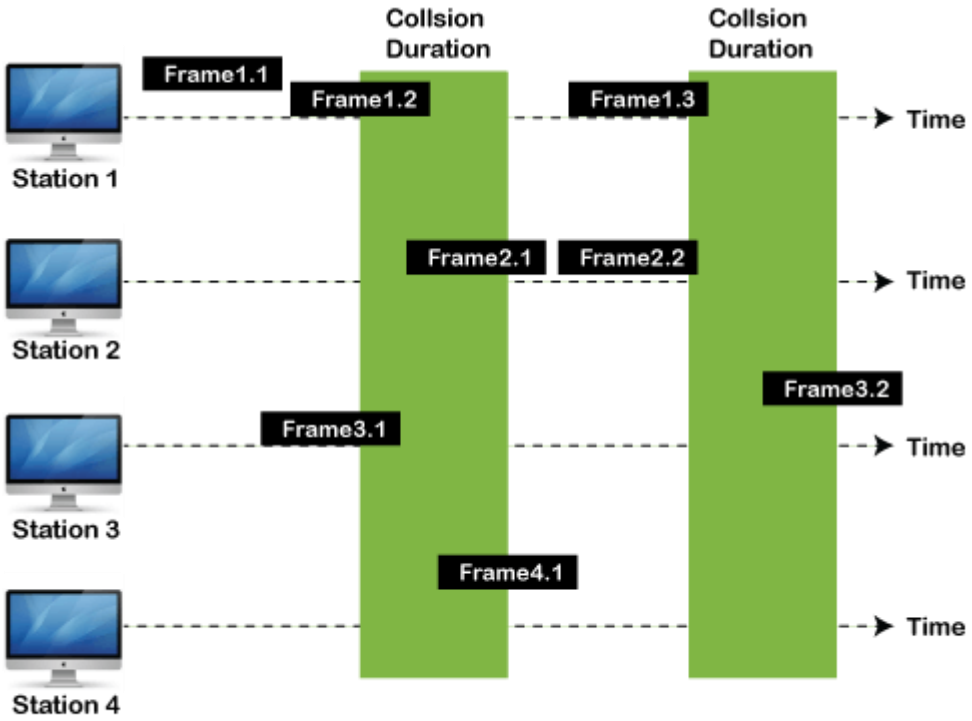
1. Any station can transmit data to a channel at any time.
2. It does not require any carrier sensing.
3. Collision and data frames may be lost during the transmission of data through multiple stations.
4. Acknowledgment of the frames exists in Aloha. Hence, there is no collision detection.
5. It requires retransmission of data after some random amount of time.



Pure Aloha

Whenever data is available for sending over a channel at stations, we use Pure Aloha. In pure Aloha, when each station transmits data to a channel without checking whether the channel is idle or not, the chances of collision may occur, and the data frame can be lost. When any station transmits the data frame to a channel, the pure Aloha waits for the receiver's acknowledgment. If it does not acknowledge the receiver end within the specified time, the station waits for a random amount of time, called the backoff time (Tb). And the station may assume the frame has been lost or destroyed. Therefore, it retransmits the frame until all the data are successfully transmitted to the receiver.

1. The total vulnerable time of pure Aloha is 2 * Tfr.
2. Maximum throughput occurs when G = 1/ 2 that is 18.4%.
3. Successful transmission of data frame is S = G * e ^ - 2 G.



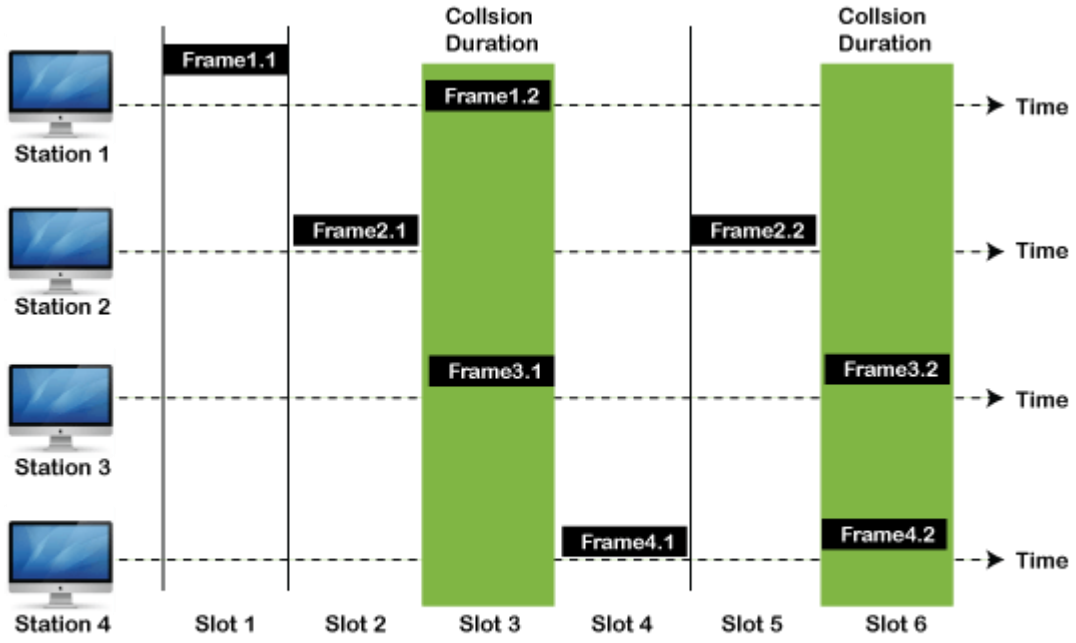
Frames in Pure ALOHA

As we can see in the figure above, there are four stations for accessing a shared channel and transmitting data frames. Some frames collide because most stations send their frames at the same time. Only two frames, frame 1.1 and frame 2.2, are successfully transmitted to the receiver end. At the same time, other frames are lost or destroyed. Whenever two frames fall on a shared channel simultaneously, collisions can occur, and both will suffer damage. If the new frame's first bit enters the channel before finishing the last bit of the second frame. Both frames are completely finished, and both stations must retransmit the data frame.

Slotted Aloha

The slotted Aloha is designed to overcome the pure Aloha's efficiency because pure Aloha has a very high possibility of frame hitting. In slotted Aloha, the shared channel is divided into a fixed time interval called **slots**. So that, if a station wants to send a frame to a shared channel, the frame can only be sent at the beginning of the slot, and only one frame is allowed to be sent to each slot. And if the stations are unable to send data to the beginning of the slot, the station will have to wait until the beginning of the slot for the next time. However, the possibility of a collision remains when trying to send a frame at the beginning of two or more station time slot.

1. Maximum throughput occurs in the slotted Aloha when G = 1 that is 37%.
2. The probability of successfully transmitting the data frame in the slotted Aloha is S = G * e ^ - 2 G.
3. The total vulnerable time required in slotted Aloha is Tfr.



Frames in Slotted ALOHA

CSMA (Carrier Sense Multiple Access)

It is a **carrier sense multiple access** based on media access protocol to sense the traffic on a channel (idle or busy) before transmitting the data. It means that if the channel is idle, the station can send data to the channel. Otherwise, it must wait until the channel becomes idle. Hence, it reduces the chances of a collision on a transmission medium.

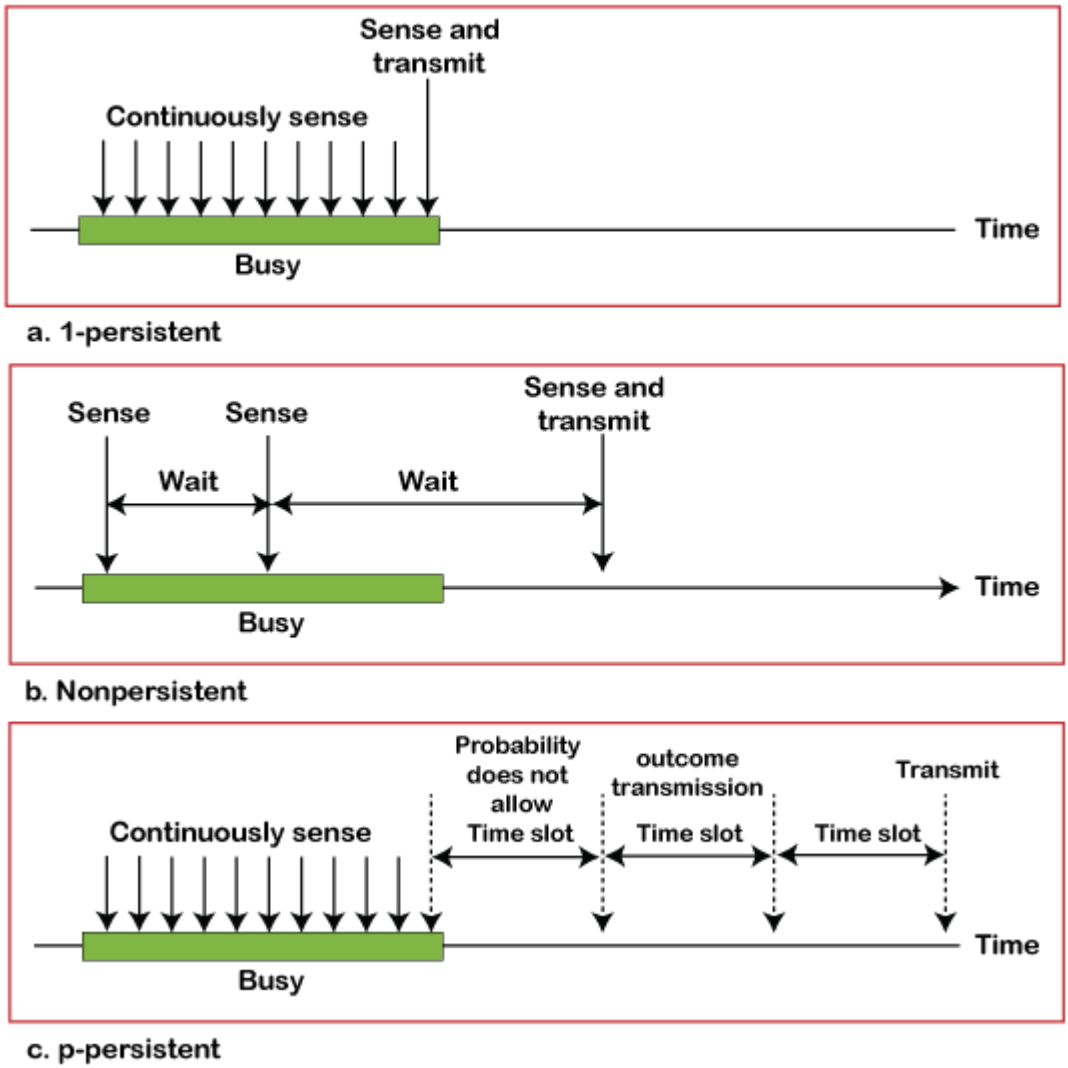
CSMA Access Modes

1-Persistent: In the 1-Persistent mode of CSMA that defines each node, first sense the shared channel and if the channel is idle, it immediately sends the data. Else it must wait and keep track of the status of the channel to be idle and broadcast the frame unconditionally as soon as the channel is idle.

Non-Persistent: It is the access mode of CSMA that defines before transmitting the data, each node must sense the channel, and if the channel is inactive, it immediately sends the data. Otherwise, the station must wait for a random time (not continuously), and when the channel is found to be idle, it transmits the frames.

P-Persistent: It is the combination of 1-Persistent and Non-persistent modes. The P-Persistent mode defines that each node senses the channel, and if the channel is inactive, it sends a frame with a **P** probability. If the data is not transmitted, it waits for a (**q = 1-p probability**) random time and resumes the frame with the next time slot.

O- Persistent: It is an O-persistent method that defines the superiority of the station before the transmission of the frame on the shared channel. If it is found that the channel is inactive, each station waits for its turn to retransmit the data.



CSMA/ CD

It is a **carrier sense multiple access/ collision detection** network protocol to transmit data frames. The CSMA/CD protocol works with a medium access control layer. Therefore, it first senses the shared channel before broadcasting the frames, and if the channel is idle, it transmits a frame to check whether the transmission was successful. If the frame is successfully received, the station sends another frame. If any collision is detected in the CSMA/CD, the station sends a jam/ stop signal to the shared channel

to terminate data transmission. After that, it waits for a random time before sending a frame to a channel.

CSMA/ CA

It is a **carrier sense multiple access/collision avoidance** network protocol for carrier transmission of data frames. It is a protocol that works with a medium access control layer. When a data frame is sent to a channel, it receives an acknowledgment to check whether the channel is clear. If the station receives only a single (own) acknowledgments, that means the data frame has been successfully transmitted to the receiver. But if it gets two signals (its own and one more in which the collision of frames), a collision of the frame occurs in the shared channel. Detects the collision of the frame when a sender receives an acknowledgment signal.

Following are the methods used in the CSMA/ CA to avoid the collision:

Interframe space: In this method, the station waits for the channel to become idle, and if it gets the channel is idle, it does not immediately send the data. Instead of this, it waits for some time, and this time period is called the **Interframe** space or IFS. However, the IFS time is often used to define the priority of the station.

Contention window: In the Contention window, the total time is divided into different slots. When the station/ sender is ready to transmit the data frame, it chooses a random slot number of slots as **wait time**. If the channel is still busy, it does not restart the entire process, except that it restarts the timer only to send data packets when the channel is inactive.

Acknowledgment: In the acknowledgment method, the sender station sends the data frame to the shared channel if the acknowledgment is not received ahead of time.

Pure Aloha	Slotted Aloha
In this Aloha, any station can transmit the data at any time.	In this, any station can transmit the data at the beginning of any time slot.
In this, The time is continuous and not globally synchronized.	In this, The time is discrete and globally synchronized.
Vulnerable time for Pure Aloha = 2 x Tt	Vulnerable time for Slotted Aloha = Tt

Pure Aloha	Slotted Aloha
In Pure Aloha, Probability of successful transmission of the data packet = G x e-2G	In Slotted Aloha, Probability of successful transmission of the data packet = G x e-G
In Pure Aloha, Maximum efficiency = 18.4%	In Slotted Aloha, Maximum efficiency = 36.8%
Pure Aloha doesn’t reduces the number of collisions to half.	Slotted Aloha reduces the number of collisions to half and doubles the efficiency of Pure Aloha.