UNIT-2

Logical Link Control : Error Detection and Error Correction

Error control in data link layer is the process of detecting and correcting data frames that have been corrupted or lost during transmission.

In case of lost or corrupted frames, the receiver does not receive the correct data-frame and sender is ignorant about the loss. Data link layer follows a technique to detect transit errors and take necessary actions, which is retransmission of frames whenever error is detected or frame is lost. The process is called Automatic Repeat Request (ARQ).

Error Detection

The error control mechanism in data link layer involves the following phases −

- **Detection of Error** − Transmission error, if any, is detected by either the sender or the receiver.
- **Acknowledgment** − acknowledgment may be positive or negative.
    - **Positive ACK** − On receiving a correct frame, the receiver sends a positive acknowledge.
    - **Negative ACK** − On receiving a damaged frame or a duplicate frame, the receiver sends a negative acknowledgment back to the sender.
- **Retransmission** − The sender maintains a clock and sets a timeout period. If an acknowledgment of a data-frame previously transmitted does not arrive before the timeout, or a negative acknowledgment is received, the sender retransmits the frame.

Error Correction

Error Control Techniques

There are three main techniques for error control −

- **Stop and Wait ARQ**

  This protocol involves the following transitions −

    - A timeout counter is maintained by the sender, which is started when a frame is sent.
    - If the sender receives acknowledgment of the sent frame within time, the sender is confirmed about successful delivery of the frame. It then transmits the next frame in queue.
    - If the sender does not receive the acknowledgment within time, the sender assumes that either the frame or its acknowledgment is lost in transit. It then retransmits the frame.
    - If the sender receives a negative acknowledgment, the sender retransmits the frame.

- **Go-Back-N ARQ**

The working principle of this protocol is −

- The sender has buffers called sending window.
- The sender sends multiple frames based upon the sending-window size, without receiving the acknowledgment of the previous ones.
- The receiver receives frames one by one. It keeps track of incoming frame's sequence number and sends the corresponding acknowledgment frames.
- After the sender has sent all the frames in window, it checks up to what sequence number it has received positive acknowledgment.
- If the sender has received positive acknowledgment for all the frames, it sends next set of frames.
- If sender receives NACK or has not receive any ACK for a particular frame, it retransmits all the frames after which it does not receive any positive ACK.

- **Selective Repeat ARQ**
  - Both the sender and the receiver have buffers called sending window and receiving window respectively.
  - The sender sends multiple frames based upon the sending-window size, without receiving the acknowledgment of the previous ones.
  - The receiver also receives multiple frames within the receiving window size.
  - The receiver keeps track of incoming frame's sequence numbers, buffers the frames in memory.
  - It sends ACK for all successfully received frames and sends NACK for only frames which are missing or damaged.
  - The sender in this case, sends only packet for which NACK is received.

Block coding

- Block coding refers to **the technique of adding extra bits to a digital word in order to improve the reliability of transmission**. The word consists of the message bits (often called information, or data) plus code bits.

Hamming Distance

- Hamming distance is a metric for comparing two binary data strings. While comparing two binary strings of equal length, Hamming distance is the number of bit positions in which the two bits are different.
- The Hamming distance between two strings, a and b is denoted as d(a,b).
- It is used for error detection or error correction when data is transmitted over computer networks. It is also using in coding theory for comparing equal length data words.
- **Calculation of Hamming Distance**
- In order to calculate the Hamming distance between two strings, and , we perform their XOR operation, (a⊕ b), and then count the total number of 1s in the resultant string.
- Example
- Suppose there are two strings 1101 1001 and 1001 1101.
- 11011001 ⊕ 10011101 = 01000100. Since, this contains two 1s, the Hamming distance, d(11011001, 10011101) = 2.

Minimum Hamming Distance

In a set of strings of equal lengths, the minimum Hamming distance is the smallest Hamming distance between all possible pairs of strings in that set.

Example

Suppose there are four strings 010, 011, 101 and 111.

10 $\oplus$ 101 = 111, d(010, 101) = 3.

010 $\oplus$ 111 = 101, d(010, 111) = 2.

011 $\oplus$ 101 = 110, d(011, 101) = 2.

011 $\oplus$ 111 = 100, d(011, 111) = 1.

101 $\oplus$ 111 = 010, d(011, 111) = 1.

Hence, the Minimum Hamming Distance, d_{min} = 1.

cyclic redundancy check (CRC)

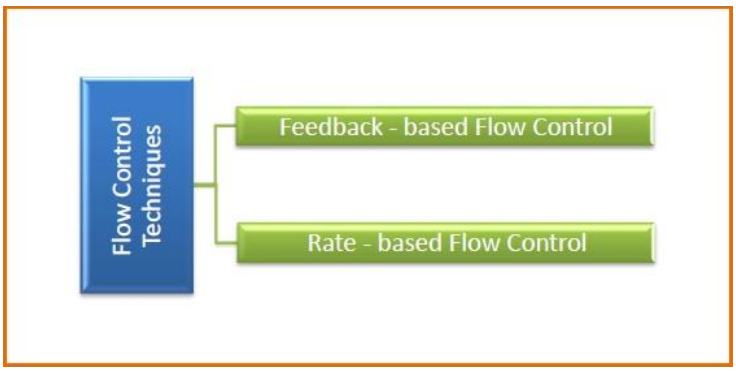
A cyclic redundancy check (CRC) is an **error-detecting code** commonly used in digital networks and storage devices to detect accidental changes to raw data. Blocks of data entering these systems get a short check value attached, based on the remainder of a polynomial division of their contents.

Flow control

Flow control is a technique that allows two stations working at different speeds to communicate with each other. It is a set of measures taken to regulate the amount of data that a sender sends so that a fast sender does not overwhelm a slow receiver. In data link layer, flow control restricts the number of frames the sender can send before it waits for an acknowledgment from the receiver.

Approaches of Flow Control

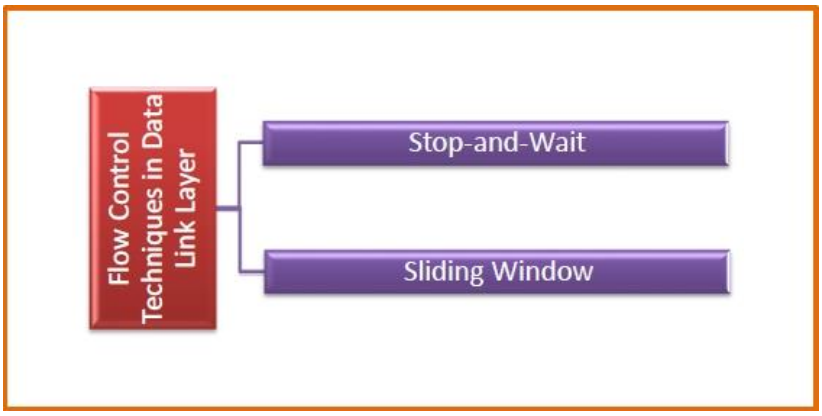
Flow control can be broadly classified into two categories −



- **Feedback based Flow Control** In these protocols, the sender sends frames after it has received acknowledgments from the user. This is used in the data link layer.
- **Rate based Flow Control** These protocols have built in mechanisms to restrict the rate of transmission of data without requiring acknowledgment from the receiver. This is used in the network layer and the transport layer.

Flow Control Techniques in Data Link Layer

Data link layer uses feedback based flow control mechanisms. There are two main techniques −



Stop and Wait

This protocol involves the following transitions −

- The sender sends a frame and waits for acknowledgment.
- Once the receiver receives the frame, it sends an acknowledgment frame back to the sender.
- On receiving the acknowledgment frame, the sender understands that the receiver is ready to accept the next frame. So it sender the next frame in queue.

Sliding Window

This protocol improves the efficiency of stop and wait protocol by allowing multiple frames to be transmitted before receiving an acknowledgment.

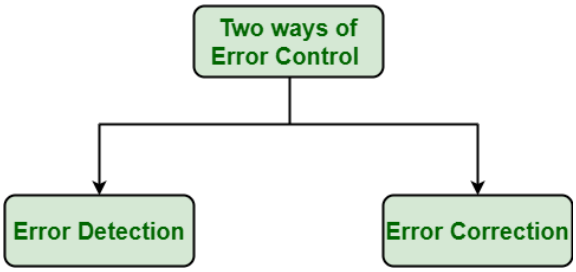
The working principle of this protocol can be described as follows −

- Both the sender and the receiver has finite sized buffers called windows. The sender and the receiver agrees upon the number of frames to be sent based upon the buffer size.
- The sender sends multiple frames in a sequence, without waiting for acknowledgment. When its sending window is filled, it waits for acknowledgment. On receiving acknowledgment, it advances the window and transmits the next frames, according to the number of acknowledgments received.

Error Control

Data-link layer uses the techniques of error control simply to ensure and confirm that all the data frames or packets, i.e. bit streams of data, are transmitted or transferred from sender to receiver with certain accuracy. Using or providing error control at this data link layer is an optimization, it was never requirement. Error control is basically process in data link layer of detecting or identifying and re-transmitting data frames that might be lost or corrupted during transmission.
In both of these cases, receiver or destination does not receive correct data-frame and sender or source does not even know anything about any such loss regarding data frames. Therefore, in such type of cases, both sender and receiver are provided with some essential protocols that are required to detect or identify such type of errors like loss of data frames.

The Data-link layer follows technique known as re-transmission of frames to detect or identify transit errors and also to take necessary actions that are required to reduce or remove such errors. Each and every time an effort is detected during transmission, particular data frames retransmitted and this process is known as**:**
There are basically two ways of doing Error control as given below :

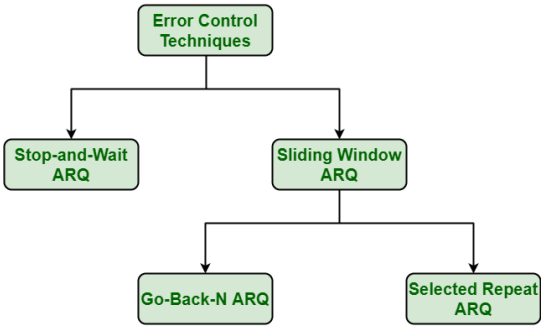


Ways of Error Control

1. **Error Detection :**
   Error detection, as name suggests, simply means detection or identification of errors. These errors may cause due to noise or any other impairments during transmission from transmitter to the receiver, in communication system. It is class of technique for detecting garbled i.e. unclear and distorted data or message.

2. **Error Correction :**
   Error correction, as name suggests, simply means correction or solving or fixing of errors. It simply means reconstruction and rehabilitation of original data that is error-free. But error correction method is very costly and is very hard.

Various Techniques for Error Control :
There are various techniques of error control as given below :



1. Stop-and-Wait ARQ **:**
Stop-and-Wait ARQ is also known as alternating bit protocol. It is one of simplest flow and error control techniques or mechanisms. This mechanism is generally required in telecommunications to transmit data or information among two connected devices. Receiver simply indicates its readiness to receive data for each frame. In these, sender sends information or data packet to receiver. Sender then stops and waits for ACK (Acknowledgment) from receiver. Further, if ACK does not arrive within given time period i.e., time-out, sender then again resends frame and waits for ACK. But, if sender receives ACK, then it will transmi next data packet to receiver and then again wait for ACK fro receiver. This process to stop and wait continues until sender has no data frame or packet to send.

2. Sliding Window ARQ **:**
This technique is generally used for continuous transmission error control. It is further categorized into two categories as given below :

- Go-Back-N ARQ **:**
  Go-Back-N ARQ is form of ARQ protocol in which transmission process continues to send or transmit total number of frames that are specified by window size even without receiving an ACK (Acknowledgement) packet from the receiver. It uses sliding window flow control protocol. If no errors occur, then operation is identical to sliding window.
- Selective Repeat ARQ **:**
  Selective Repeat ARQ is also form of ARQ protocol in which only suspected or damaged or lost data frames are only retransmitted. This technique is similar to Go-Back-N ARQ though much more efficient than the Go-Back-N ARQ technique due to reason that it reduces number of retransmission. In this, the sender only retransmits frames for which NAK is received. But this technique is used less because of more complexity at sender and receiver and each frame must be needed to acknowledged individually.

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

Piggybacking

Communications are mostly full – duplex in nature, i.e. data transmission occurs in both directions. A method to achieve full – duplex communication is to consider both the communication as a pair of simplex communication. Each link comprises a forward channel for sending data and a reverse channel for sending acknowledgments.

However, in the above arrangement, traffic load doubles for each data unit that is transmitted. Half of all data transmission comprise of transmission of acknowledgments.

So, a solution that provides better utilization of bandwidth is piggybacking. Here, sending of acknowledgment is delayed until the next data frame is available for transmission. The acknowledgment is then hooked onto the outgoing data frame. The data frame consists of an *ack* field. The size of the *ack* field is only a few bits, while an acknowledgment frame comprises of several bytes. Thus, a substantial gain is obtained in reducing bandwidth requirement.

Working Principle

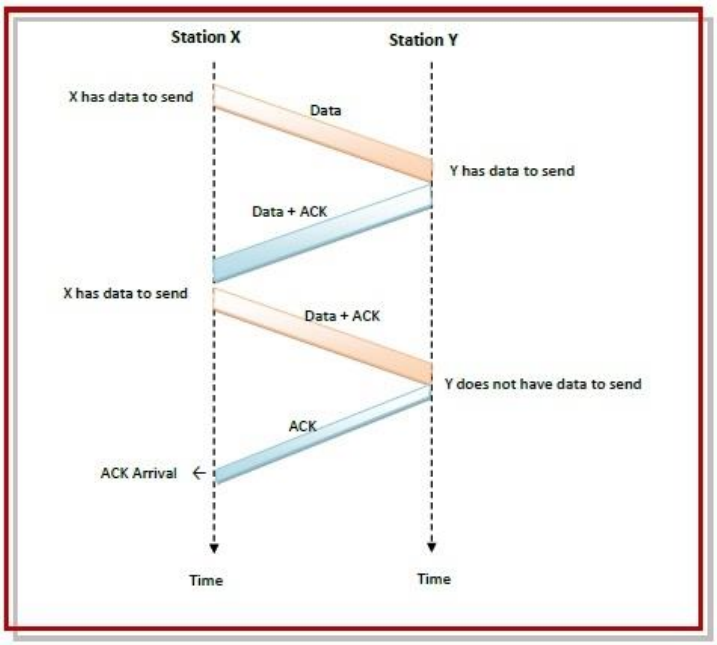
Suppose that there are two communication stations X and Y. The data frames transmitted have an acknowledgment field, *ack* field that is of a few bits length. Additionally, there are frames for sending acknowledgments, ACK frames. The purpose is to minimize the ACK frames.

The three principles governing piggybacking when the station X wants to communicate with station Y are –

- If station X has both data and acknowledgment to send, it sends a data frame with the *ack* field containing the sequence number of the frame to be acknowledged.
- If station X has only an acknowledgment to send, it waits for a finite period of time to see whether a data frame is available to be sent. If a data frame becomes available, then it piggybacks the acknowledgment with it. Otherwise, it sends an ACK frame.
- If station X has only a data frame to send, it adds the last acknowledgment with it. The station Y discards all duplicate acknowledgments. Alternatively, station X may send the data frame with the *ack* field containing a bit combination denoting no acknowledgment.

Example

The following diagram illustrates the three scenario −



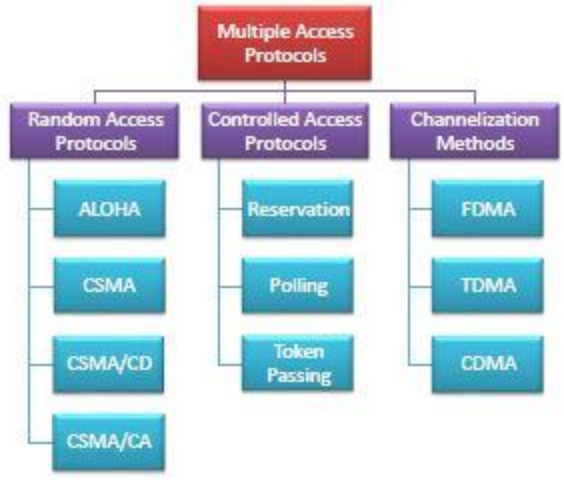
Multiple access protocols

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.

Categories of Multiple Access Protocols

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



Random Access Protocols

Random access protocols assign uniform priority to all connected nodes. Any node can send data if the transmission channel is idle. No fixed time or fixed sequence is given for data transmission.

The four random access protocols are−

- ALOHA
- Carrier sense multiple access (CMSA)
- Carrier sense multiple access with collision detection (CMSA/CD)
- Carrier sense multiple access with collision avoidance (CMSA/CA)

Controlled Access Protocols

Controlled access protocols allow only one node to send data at a given time.Before initiating transmission, a node seeks information from other nodes to determine which station has the right to send. This avoids collision of messages on the shared channel.

The station can be assigned the right to send by the following three methods−

- Reservation
- Polling
- Token Passing

Channelization

Channelization are a set of methods by which the available bandwidth is divided among the different nodes for simultaneous data transfer.

The three channelization methods are−

- Frequency division multiple access (FDMA)
- Time division multiple access (TDMA)
- Code division multiple access (CDMA)

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

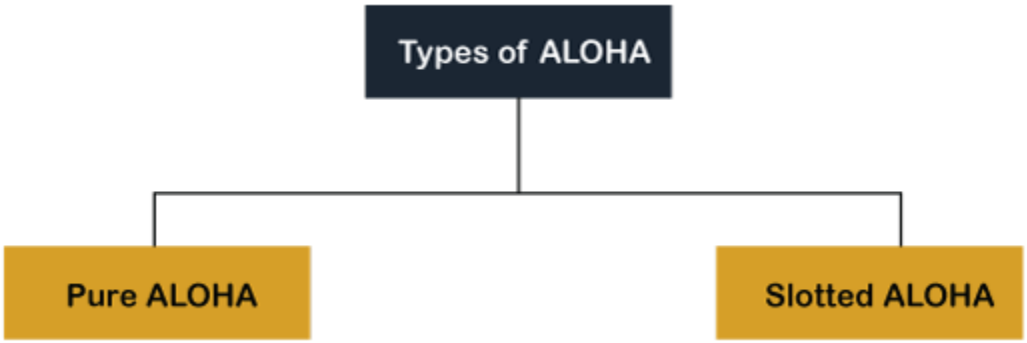
ALOHA Random Access Protocol

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.

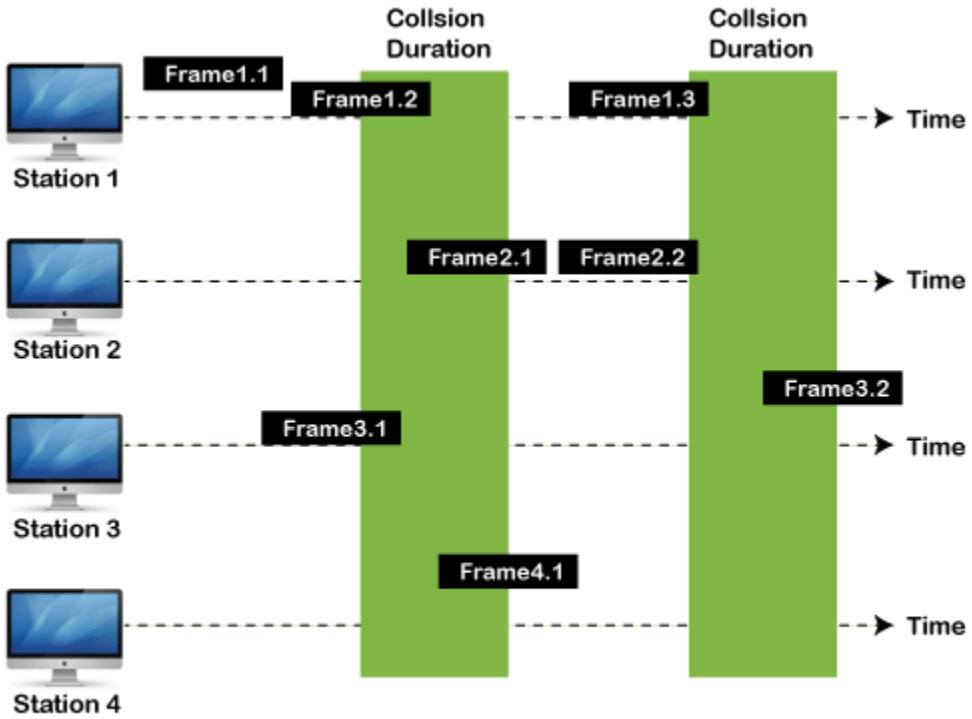
5. It requires retransmission of data after some random amount of time.



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.



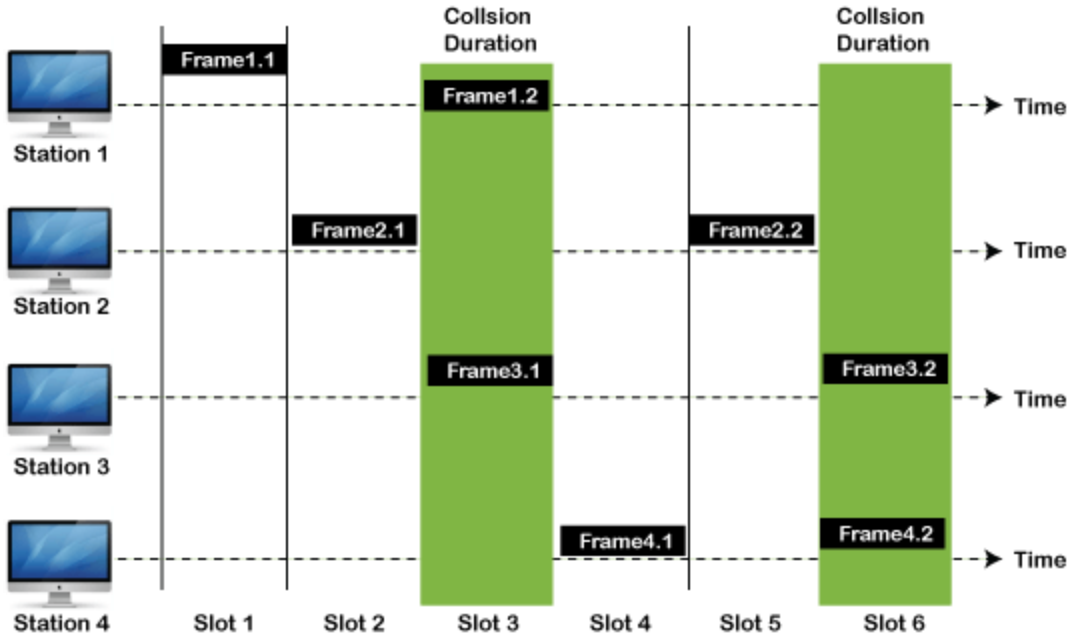
Frames in Pure ALOHA

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.

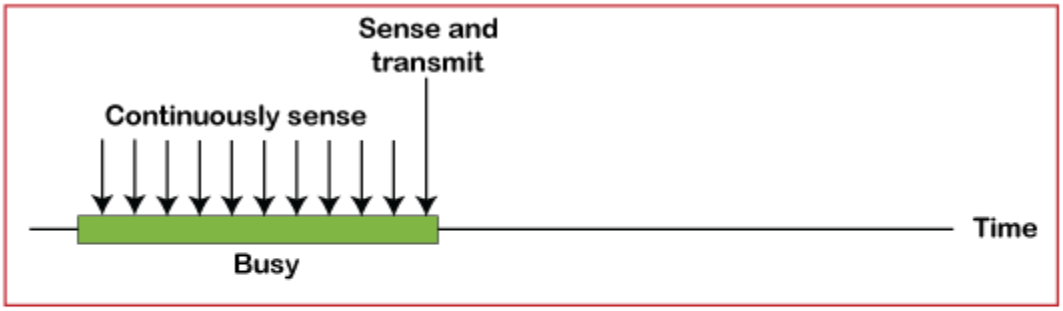
CSMA Access Modes

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.

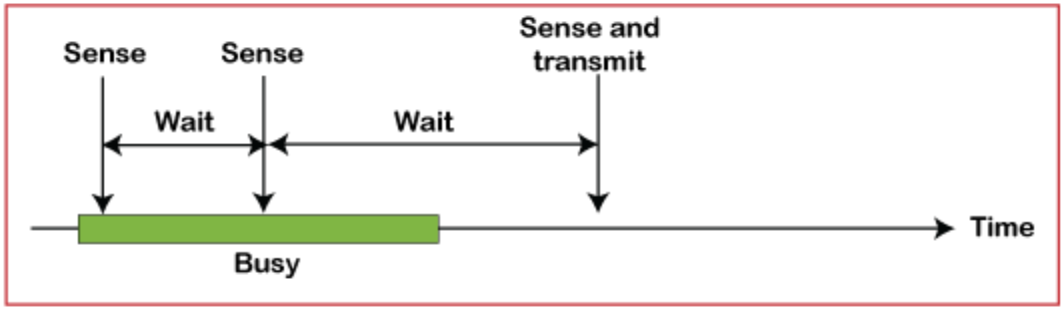
Non-Persistent: It is the access mode of CSMA that defines before transmitting the data, each node must sense the channel, and if the channel is inactive, it immediately sends the data. Otherwise, the station must wait for a random time (not continuously), and when the channel is found to be idle, it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

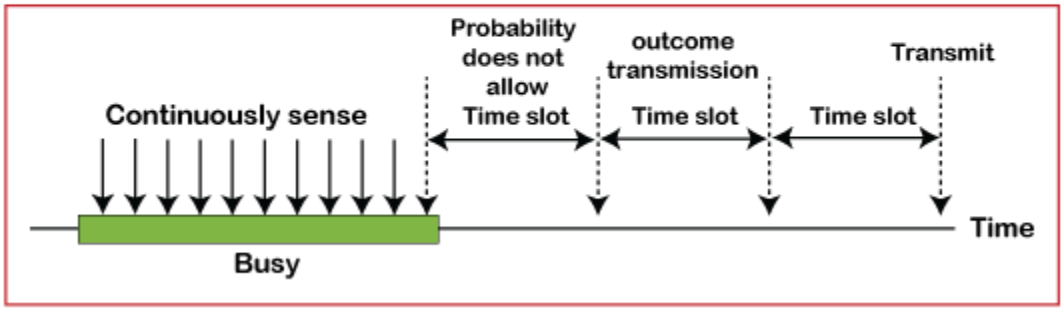
O- Persistent: It is an O-persistent method that defines the superiority of the station before the transmission of the frame on the shared channel. If it is found that the channel is inactive, each station waits for its turn to retransmit the data.



a. 1-persistent



b. Nonpersistent



c. p-persistent

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.

B. Controlled Access Protocol

It is a method of reducing data frame collision on a shared channel. In the controlled access method, each station interacts and decides to send a data frame by a particular station approved by all other stations. It means that a single station cannot send the data frames unless all other stations are not approved. It has three types of controlled access: **Reservation, Polling**, and **Token Passing**.

C. Channelization Protocols

It is a channelization protocol that allows the total usable bandwidth in a shared channel to be shared across multiple stations based on their time, distance and codes. It can access all the stations at the same time to send the data frames to the channel.

Following are the various methods to access the channel based on their time, distance and codes:

1. FDMA (Frequency Division Multiple Access)
2. TDMA (Time Division Multiple Access)

3. CDMA (Code Division Multiple Access)

FDMA

It is a frequency division multiple access (**FDMA**) method used to divide the available bandwidth into equal bands so that multiple users can send data through a different frequency to the subchannel. Each station is reserved with a particular band to prevent the crosstalk between the channels and interferences of stations.

TDMA

Time Division Multiple Access (**TDMA**) is a channel access method. It allows the same frequency bandwidth to be shared across multiple stations. And to avoid collisions in the shared channel, it divides the channel into different frequency slots that allocate stations to transmit the data frames. The same **frequency** bandwidth into the shared channel by dividing the signal into various time slots to transmit it. However, TDMA has an overhead of synchronization that specifies each station's time slot by adding synchronization bits to each slot.

CDMA

The code division multiple access (CDMA) is a channel access method. In CDMA, all stations can simultaneously send the data over the same channel. It means that it allows each station to transmit the data frames with full frequency on the shared channel at all times. It does not require the division of bandwidth on a shared channel based on time slots. If multiple stations send data to a channel simultaneously, their data frames are separated by a unique code sequence. Each station has a different unique code for transmitting the data over a shared channel. For example, there are multiple users in a room that are continuously speaking. Data is received by the users if only two-person interact with each other using the same language. Similarly, in the network, if different stations communicate with each other simultaneously with different code language.