Noiseless Channels:

1. Simplest Protocol

It has no flow or error control. It is a unidirectional protocol in which data frames are travelling in only one direction-from the sender to receiver. The data link layer of the receiver immediately removes the header from the frame and hands the data packet to its network layer, which can also accept the packet immediately.

Design

The sender site cannot send a frame until its network layer has a data packet to send. The receiver site cannot deliver a data packet to its network layer until a frame arrives.

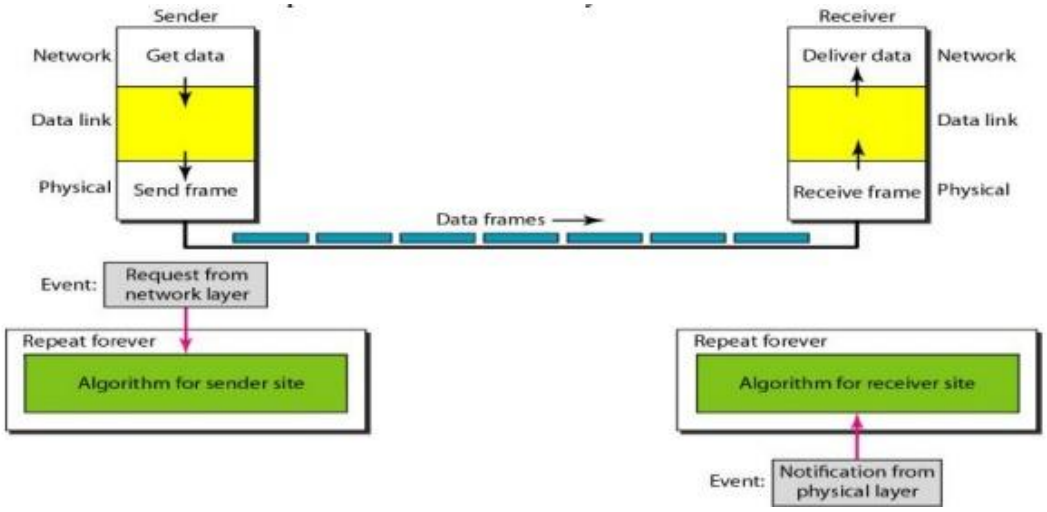


Figure 2.6 The design of the simplest protocol with no flow or error control

If the protocol is implemented as a procedure, we need to introduce the idea of events in the protocol. The procedure at the sender site is constantly running; there is no action until there is a request from the network layer. The procedure at the receiver site is also constantly running, but there is no action until notification from the physical layer arrives.

Example 2.1

It is very simple. The sender sends a sequence of frames without even thinking about the receiver. To send three frames, three events occur at the sender site and three events at the receiver site. Note that the data frames are shown by tilted boxes; the height of the box defines the transmission time difference between the first bit and the last bit in the frame.

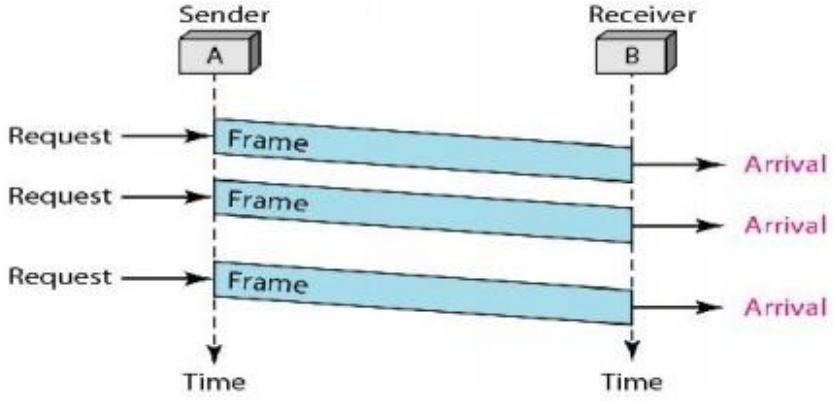


Figure 2.7 Flow diagram for Example 2.1

2. Stop-and-Wait Protocol

If data frames arrive at the receiver site faster than they can be processed, the frames must be stored until their use.

In Stop-and-Wait Protocol the sender sends one frame, stops until it receives confirmation from the receiver (okay to go ahead), and then sends the next frame.

Design

Figure 2.8 illustrates the mechanism.

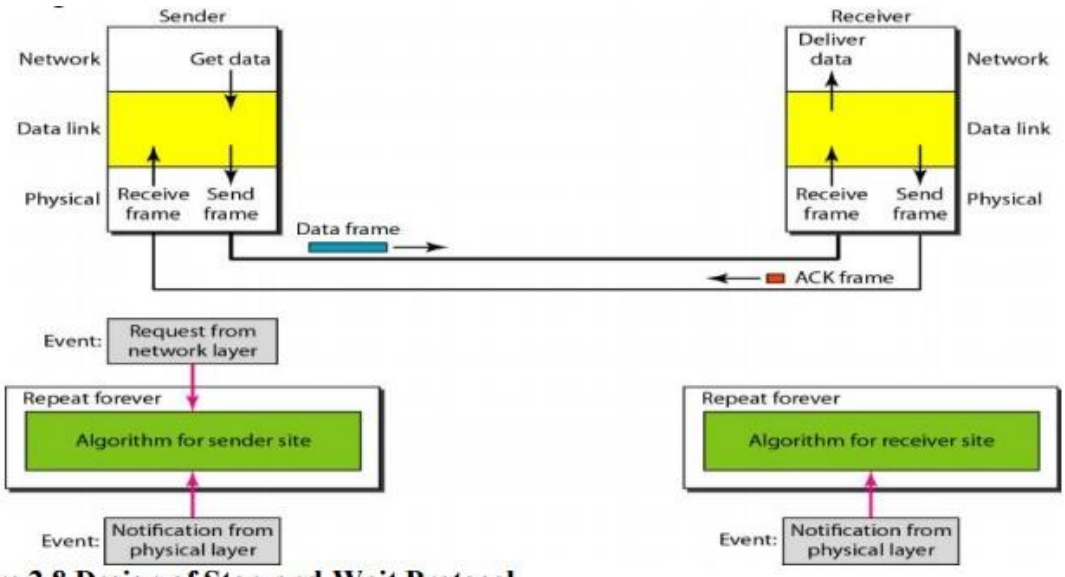


Figure 2.8 Design of Stop-and-Wait Protocol

Comparing this figure with Figure 2.6, we can see the traffic on the forward channel (from sender to receiver) and the reverse channel. At any time, there is either one data frame on the forward channel or one ACK frame on the reverse channel. We therefore need a half-duplex link.

Example 2.2

Figure 2.9 shows an example of communication using this protocol. It is still very simple. The sender sends one frame and waits for feedback from the receiver. When the ACK arrives, the sender sends the next frame. Note that sending two frames in the protocol involves the sender in four events and the receiver in two events.

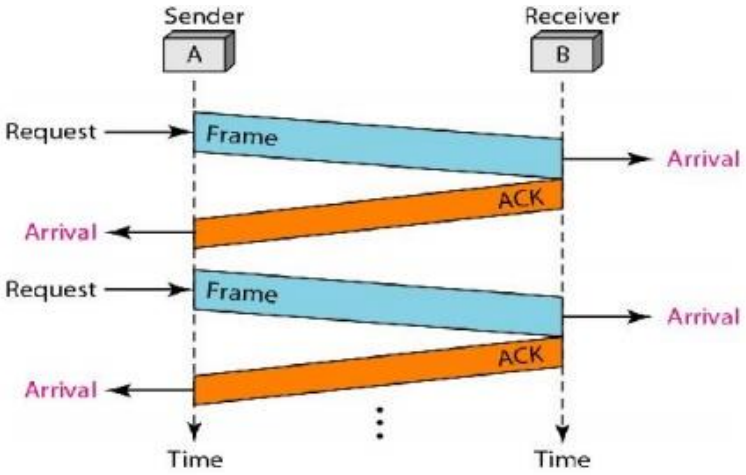


Figure 2.9 Flow diagram for Example 2.2

Noisy Channels:

1. Stop-and-Wait Automatic Repeat Request

The Stop-and-Wait Automatic Repeat Request (Stop-and-Wait ARQ), adds a simple error control mechanism to the Stop-and-Wait Protocol. To detect and correct corrupted frames, we need to add redundancy bits to our data frame. When the frame arrives at the receiver site, it is checked and if it is corrupted, it is silently discarded. The detection of errors in this protocol is manifested by the silence of the receiver.

Sequence Numbers

The protocol specifies that frames need to be numbered. This is done by using sequence numbers. A field is added to the data frame to hold the sequence number of that frame. For example, if we decide that the field is *m* bits long, the sequence numbers start from 0, go to $2m - 1$, and then are repeated.

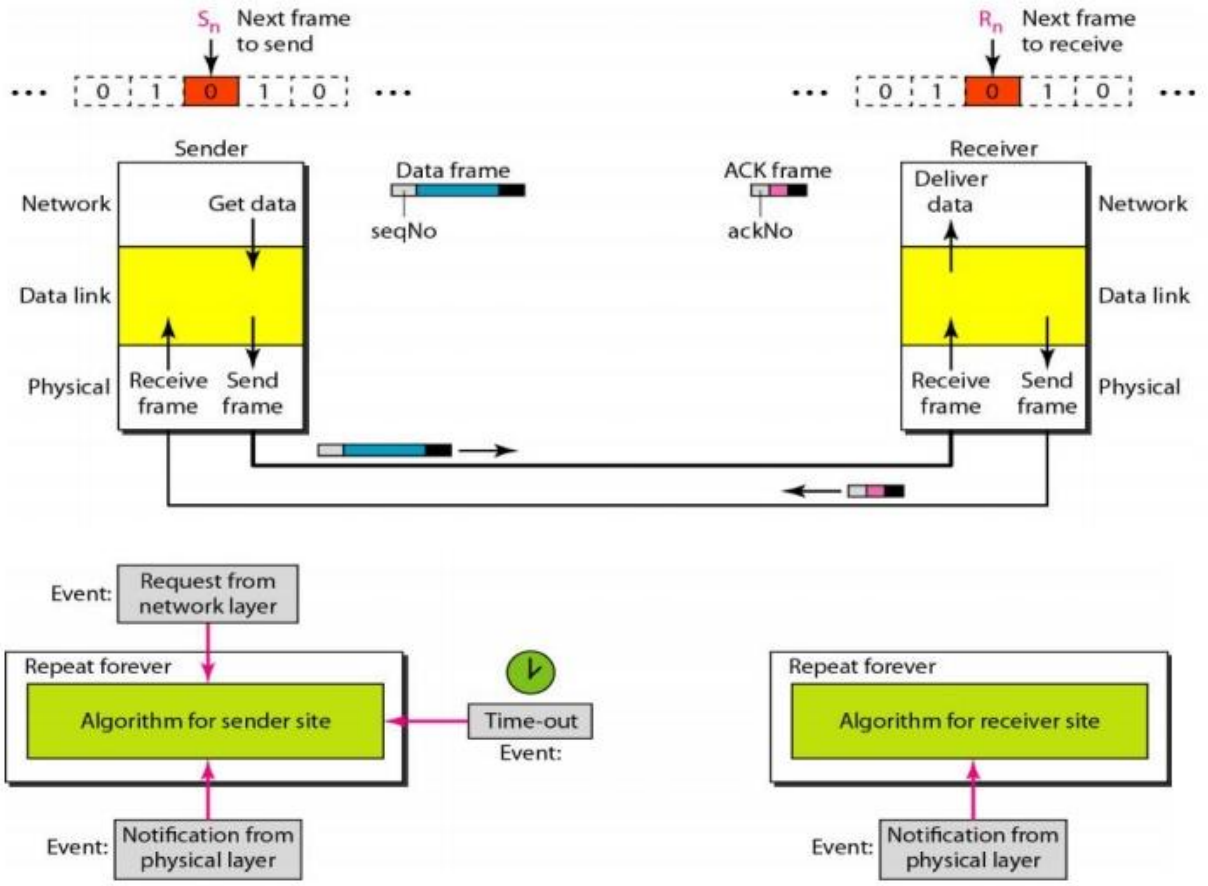


Figure 2.10 Design of the Stop-and-wait ARQ Protocol

Acknowledgment Numbers

Since the sequence numbers must be suitable for both data frames and ACK frames, we use this convention: The acknowledgment numbers always announce the sequence number of the next frame expected by the receiver. For example, if frame 0 has arrived safe and sound, the receiver sends an ACK frame with acknowledgment 1 (meaning frame 1 is expected next). If frame 1 has arrived safe and sound, the receiver sends an ACK frame with acknowledgment 0 (meaning frame 0 is expected).

Design

Figure 2.10 shows the design of the Stop-and-Wait ARQ Protocol. The sending device keeps a copy of the last frame transmitted until it receives an acknowledgment for that frame. A data frames uses a

seq No (sequence number); an ACK frame uses an ack No (acknowledgment number). The sender has a control variable, which we call *Sn* (sender, next frame to send), that holds the sequence number for the next frame to be sent (0 or 1).

The receiver has a control variable, which we call *Rn* (receiver, next frame expected), that holds the number of the next frame expected. When a frame is sent, the value of *Sn* is incremented (modulo-2), which means if it is 0, it becomes 1 and vice versa. When a frame is received, the value of *Rn* is incremented (modulo-2), which means if it is 0, it becomes 1 and vice versa. Three events can happen at the sender site; one event can happen at the receiver site. Variable *Sn* points to the slot that matches the sequence number of the frame that has been sent, but not acknowledged; *Rn* points to the slot that matches the sequence number of the expected frame.

Example 2.3

Frame 0 is sent and acknowledged. Frame 1 is lost and resent after the time-out. The resent frame 1 is acknowledged and the timer stops. Frame 0 is sent and acknowledged, but the acknowledgment is lost. The sender has no idea if the frame or the acknowledgment is lost, so after the time-out, it resends frame 0, which is acknowledged.

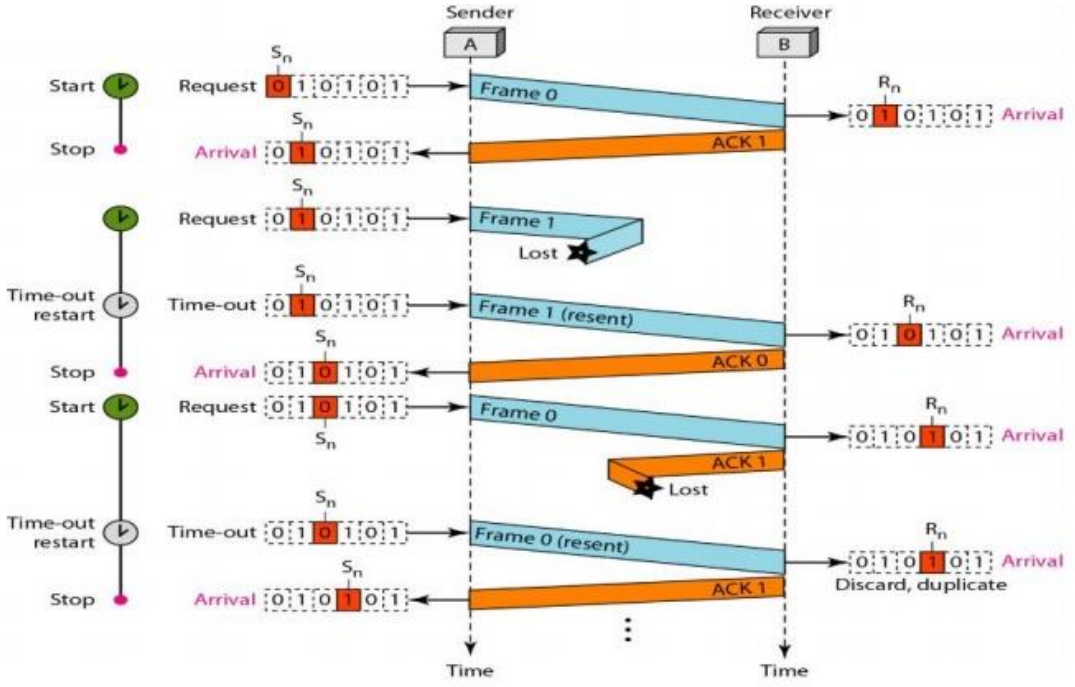


Figure 2.11 Flow diagram for Example 2.3

Example 2.4

Assume that, in a Stop-and-Wait ARQ system, the bandwidth of the line is 1 Mbps, and 1 bit takes 20 ms to make a round trip. What is the bandwidth-delay product? If the system data frames are 1000 bits in length, what is the utilization percentage of the link?

Solution

The bandwidth-delay product is (1x106)x(20x10-3) =20,000bit

Pipelining

In networking and in other areas, a task is often begun before the previous task has ended. This is known as pipelining. There is no pipelining in Stop-and-Wait ARQ because we need to wait for a frame to reach the destination and be acknowledged before the next frame can be sent. However, pipelining does apply to our next two protocols because several frames can be sent before we receive news about the previous frames. Pipelining improves the efficiency of the transmission if the number of bits in transition is large with respect to the bandwidth-delay product.

2. Go-Back-N Automatic Repeat Request

In this protocol we can send several frames before receiving acknowledgments; we keep a copy of these frames until the acknowledgments arrive.

Sequence Numbers

Frames from a sending station are numbered sequentially. However, because we need to include the sequence number of each frame in the header, we need to set a limit. If the header of the frame allows m bits for the sequence number, the sequence numbers range from 0 to 2m - 1.

Sliding Window

The sliding window is an abstract concept that defines the range of sequence numbers that is the concern of the sender and receiver. In other words, the sender and receiver need to deal with only part of the possible sequence numbers. The range which is the concern of the sender is called the send sliding window; the range that is the concern of the receiver is called the receive sliding window.

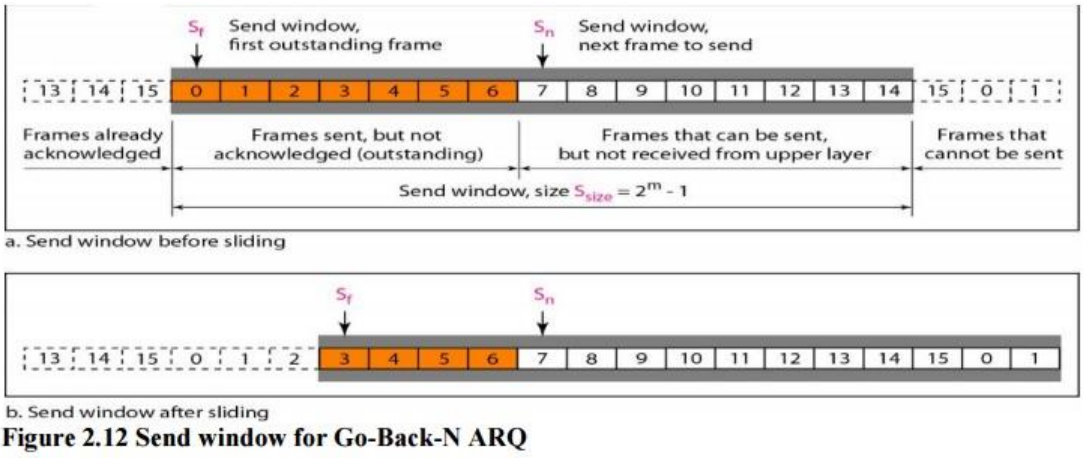


Figure 2.12 Send window for Go-Back-N ARQ

The sender does not worry about these frames and keeps no copies of them. The second region, colored in Figure 2.12 a, defines the range of sequence numbers belonging to the frames that are sent and have an unknown status.

The window itself is an abstraction; three variables define its size and location at any time. We call these variables *Sf(send* window, the first outstanding frame), *Sn* (send window, the next frame to be sent), and Ssize (send window, size). The variable *Sf* defines the sequence number of the first (oldest) outstanding frame. The variable *Sn* holds the sequence number that will be assigned to the next frame to be sent. Finally, the variable Ssize defines the size of the window, which is fixed in our protocol.

The receive window makes sure that the correct data frames are received and that the correct acknowledgments are sent. The size of the receive window is always 1.

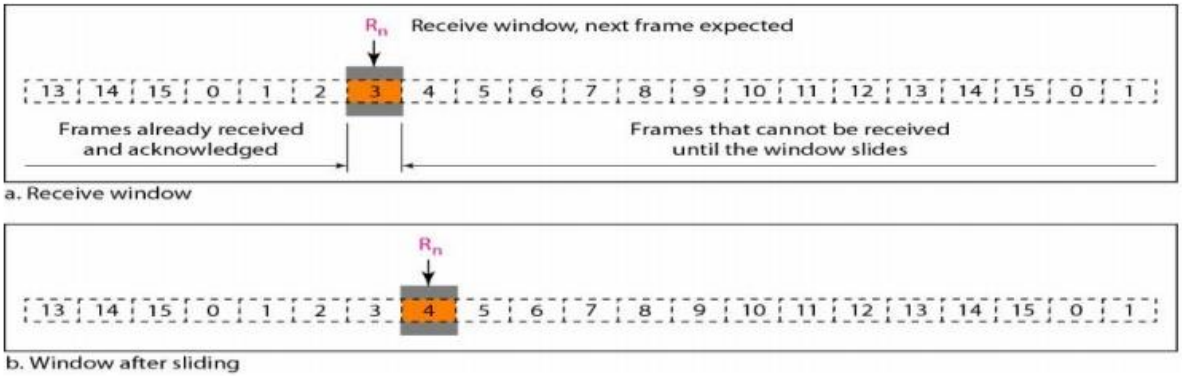


Figure 2.13 Receive window for Go-Back-N ARQ

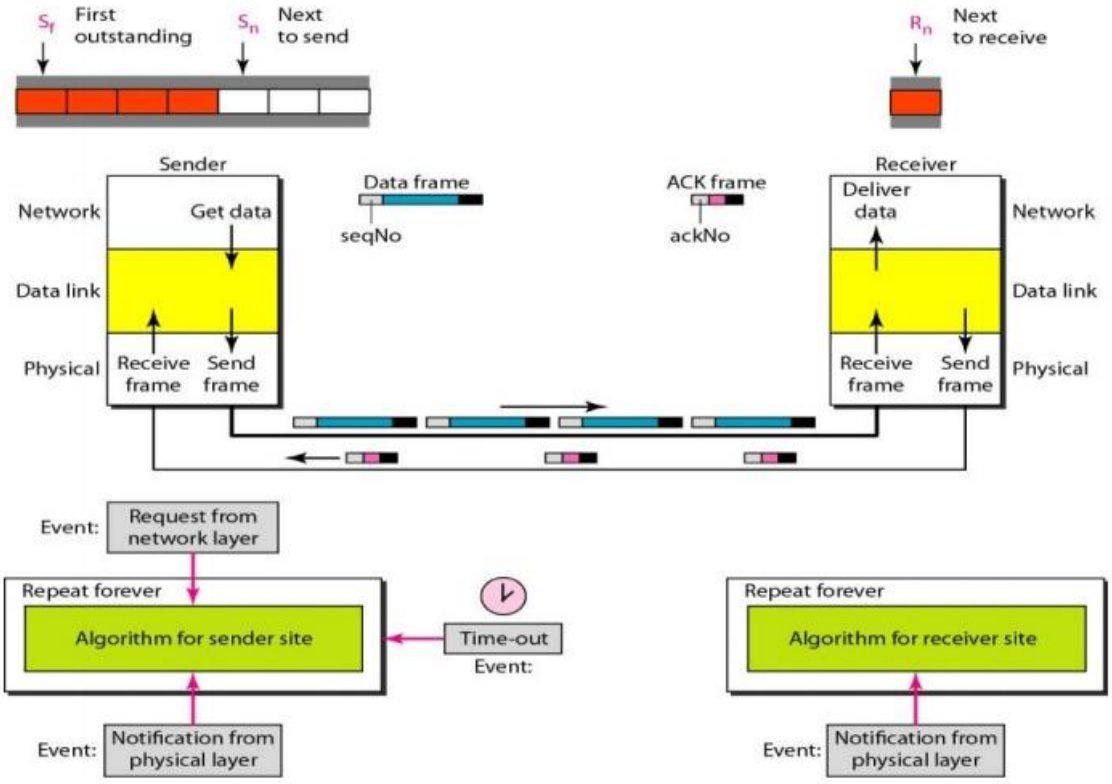


Figure 2.14 Design of Go-Back-N ARQ

Note that we need only one variable Rn (receive window, next frame expected) to define this abstraction. The sequence numbers to the left of the window belong to the frames already received and acknowledged; the sequence numbers to the right of this window define the frames that cannot be received. Any received frame with a sequence number in these two regions is discarded. Only a frame with a sequence number matching the value of Rn is accepted and acknowledged. The receive window also slides, but only one slot at a time.

Design

Figure 2.14 shows the design for this protocol. As we can see, multiple frames can be in transit in the forward direction, and multiple acknowledgments in the reverse direction. The idea is similar to Stop-and-Wait ARQ; the difference is that the send window allows us to have as many frames in transition as there are slots in the send window.

Send Window Size

We can now show why the size of the send window must be less than $2m$. As an example, we choose m =2, which means the size of the window can be $2m$- 1, or 3. Figure 2.15 compares a window size of 3 against a window size of 4. If the size of the window is 3 (less than 22) and all three acknowledgments are lost, the frame 0 timer expires and all three frames are resent. The receiver is now expecting frame 3, not frame 0, so the duplicate frame is correctly discarded.

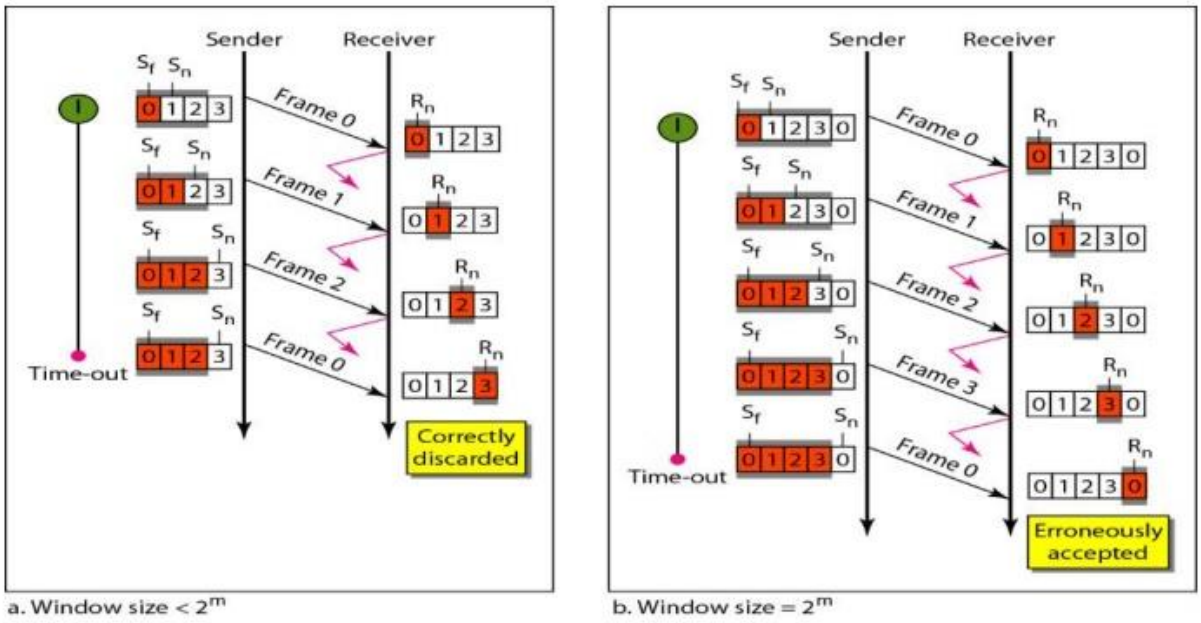


Figure 2.15 Window size for Go-Back-N ARQ

Example 2.4

Figure 2.16 shows an example of Go-Back-*N*. This is an example of a case where the forward channel is reliable, but the reverse is not. No data frames are lost, but some ACKs are delayed and one is lost. The example also shows how cumulative acknowledgments can help if acknowledgments are delayed or lost.

After initialization, there are seven sender events. Request events are triggered by data from the network layer; arrival events are triggered by acknowledgments from the physical layer. There is no time-out event here because all outstanding frames are acknowledged before the timer expires. Note that although ACK 2 is lost, ACK 3 serves as both ACK 2 and ACK3. There are four receiver events, all triggered by the arrival of frames from the physical layer.

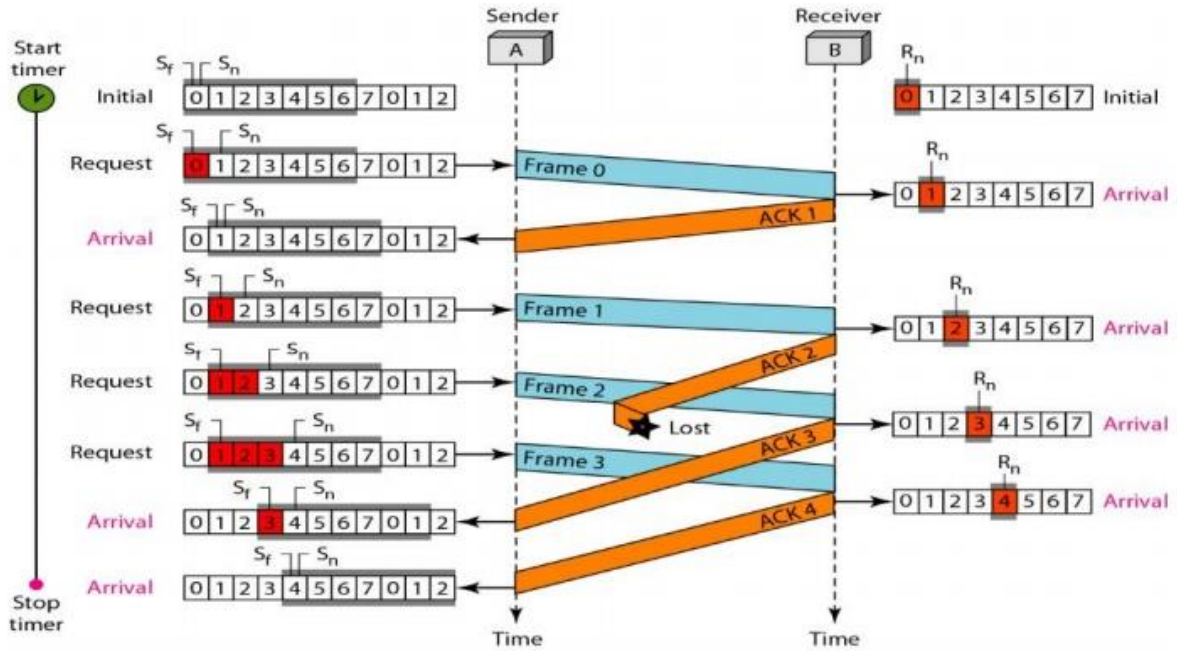


Figure 2.16 Flow diagrams for Example 2.4

3. Selective Repeat Automatic Repeat Request

Go-Back-N ARQ simplifies the process at the receiver site. The receiver keeps track of only one variable, and there is no need to buffer out-of-order frames; they are simply discarded. However, this protocol is very inefficient for a noisy link. In a noisy link a frame has a higher probability of damage, which means the resending of multiple frames. This resending uses up the bandwidth and slows down the transmission.

Windows

he Selective Repeat Protocol also uses two windows: a send window and a receive window. First, the size of the send window is much smaller; it is 2m- 1. Second, the receive window is the same size as the send window. The send window maximum size can be 2m- 1. For example, if m = 4, the sequence numbers go from 0 to 15, but the size of the window is just 8 (it is 15 in the Go-Back-N Protocol). The smaller window size means less efficiency in filling the pipe, but the fact that there are fewer duplicate frames can compensate for this.

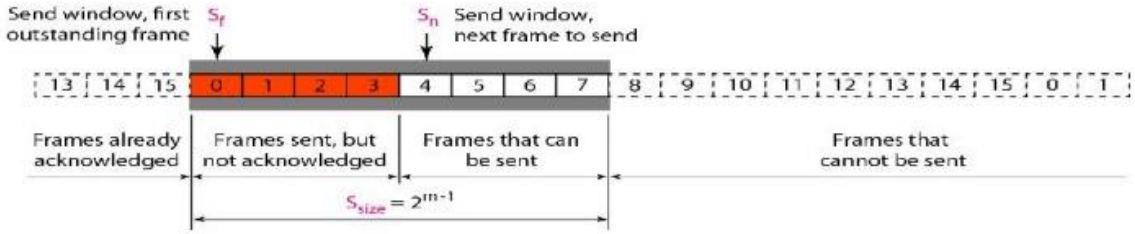


Figure 2.17 Send window for Selective Repeat ARQ

The receive window in Selective Repeat is totally different from the one in Go Back- N. First, the size of the receive window is the same as the size of the send window (2m- 1). Figure 2.18 shows the receive window in this protocol. Those slots inside the window that are colored define frames that have arrived out of order and are waiting for their neighbors to arrive before delivery to the network layer.

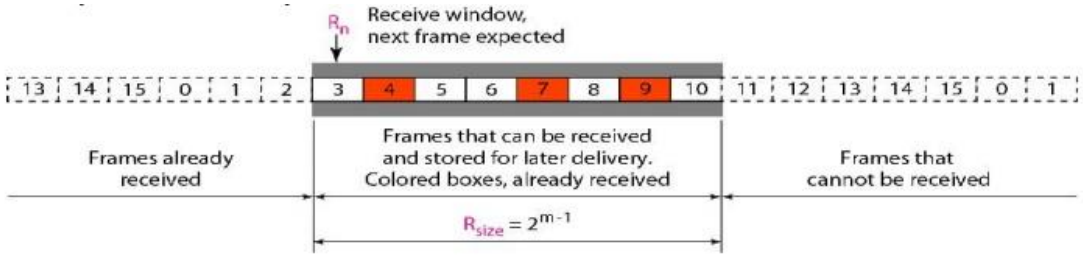


Figure 2.18 Receive window for Selective Repeat ARQ

Design

The design in this case is to some extent similar to the one we described for the 00Back-N, but more complicated, as shown in Figure 2.19.

Window Sizes

We can now show why the size of the sender and receiver windows must be at most on half of *2m*. For an example, we choose *m* = 2, which means the size of the window is *2m/2*, or 2. If thesize of the window is 2 and all acknowledgments are lost, the timer for frame 0 expires and frame 0 is resent. However, this time, the window of the receiver expects to receive frame 0 (0 is part of the window), so it accepts frame 0, not as a duplicate, but as the first frame in the next cycle. This is clearly an error. In Selective Repeat ARQ, the size of the sender and receiver window must be at most one-half of *2m*

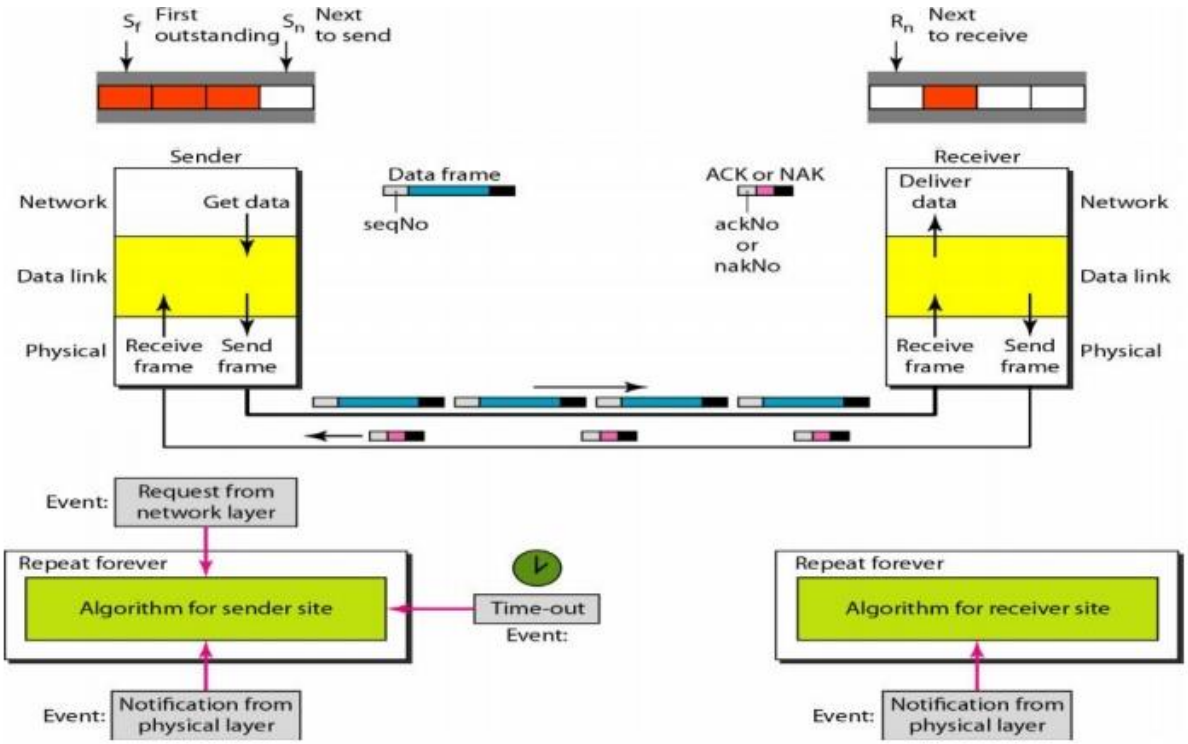


Figure 2.19 Design of Selective Repeat ARQ

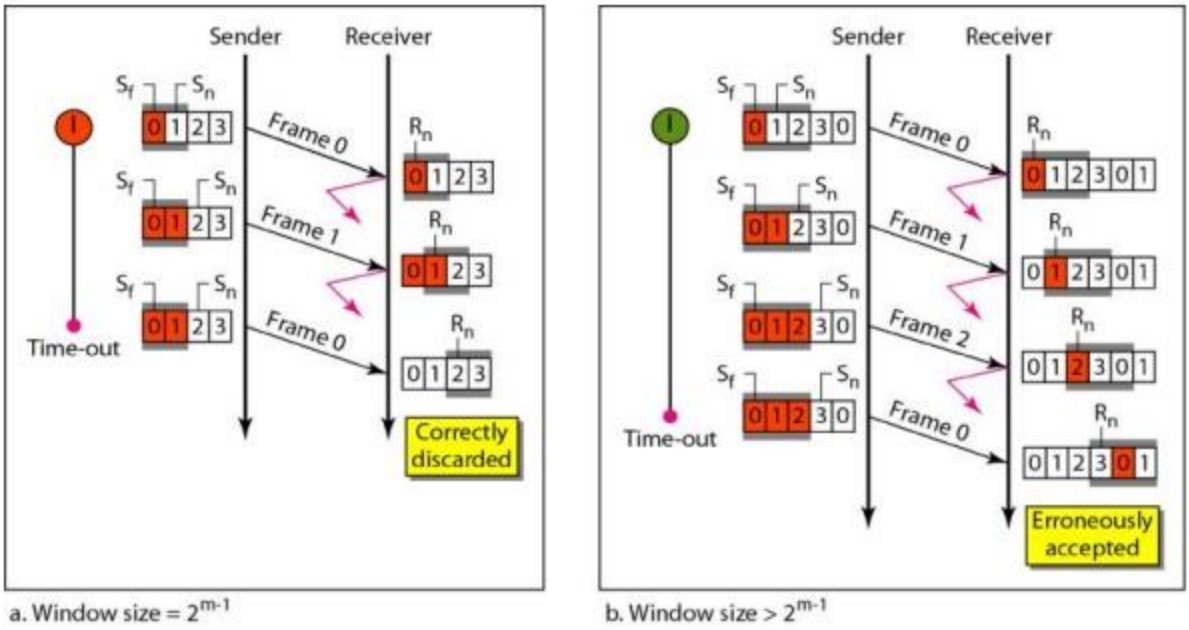


Figure 2.20 Selective Repeat ARQ, Window size

Example 2.5

Frame 1 is lost. We show how Selective Repeat behaves in this case.

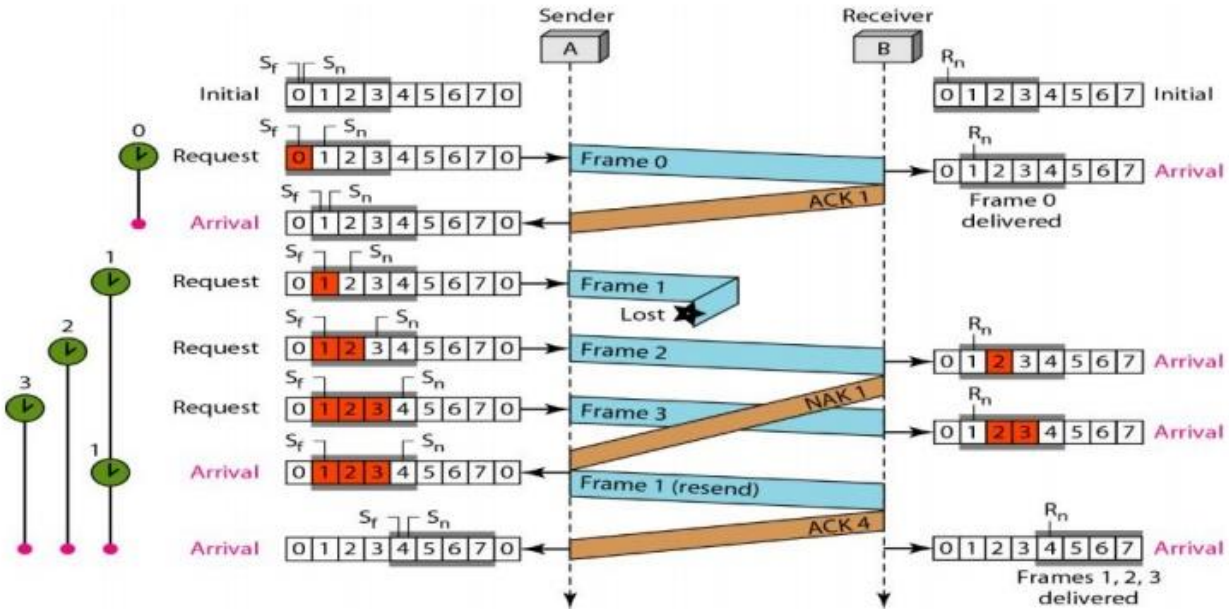


Figure 2.21 Flow diagram for Example 2.5

One main difference is the number of timers. Here, each frame sent or resent needs a timer, which means that the timers need to be numbered (0, 1, 2, and 3). The timer for frame 0 starts at the first request, but stops when the ACK for this frame arrives. The timer for frame 1 starts at the second request restarts when a NAK arrives, and finally stops when the last ACK arrives. The other two timers start when the corresponding frames are sent and stop at the last arrival event.

Piggybacking

The three protocols we discussed in this section are all unidirectional: data frames flow in only one direction although control information such as ACK and NAK frames can travel in the other direction. In real life, data frames are normally flowing in both directions: from node A to node B and from node B to node A. This means that the control information also needs to flow in both directions.

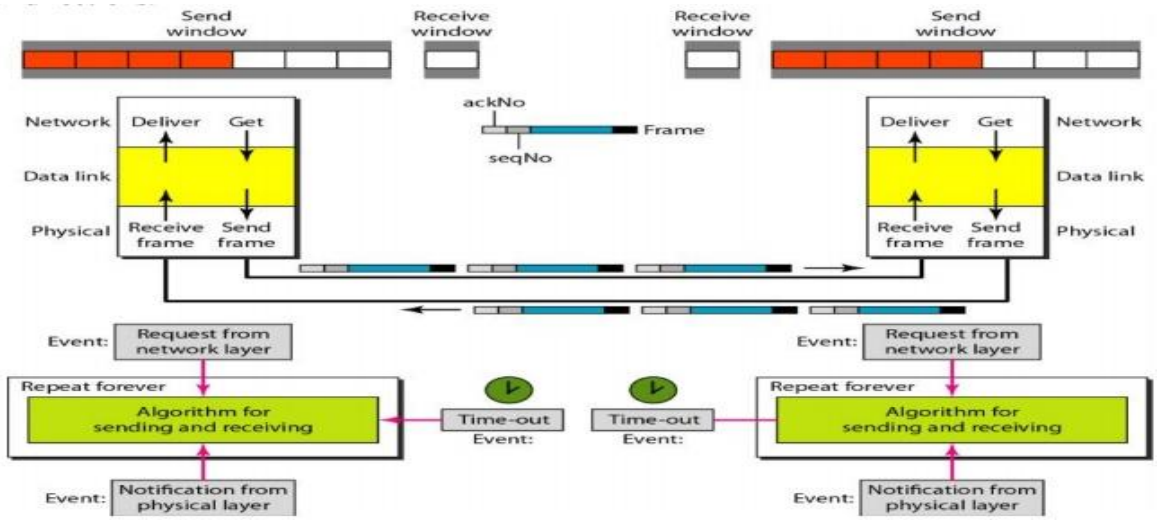


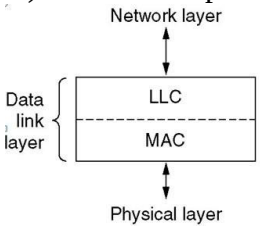
Figure 2.22 Design of Piggybacking in Go-Back-N ARQ

A technique called piggybacking is used to improve the efficiency of the bidirectional protocols. When a frame is carrying data from A to B, it can also carry control information about arrived (or lost) frames from B; when a frame is carrying data from B to A, it can also carry control information about the arrived (or lost) frames from A.

THE MEDIUM ACCESS SUB LAYER:

To coordinate the access to the channel, multiple access protocols are requiring. All these protocols belong to the MAC sub layer. Data Link layer is divided into twosub layers:

1. Logical Link Control (LLC)- is responsible for error control & flow control.
2. Medium Access Control (MAC)- MAC is responsible for multiple access resolutions



THE CHANNEL ALLOCATION PROBLEM

In broadcast networks, single channel is shared by several stations. This channel can be allocated to only one transmitting user at a time. There are two different methods of channel allocations:

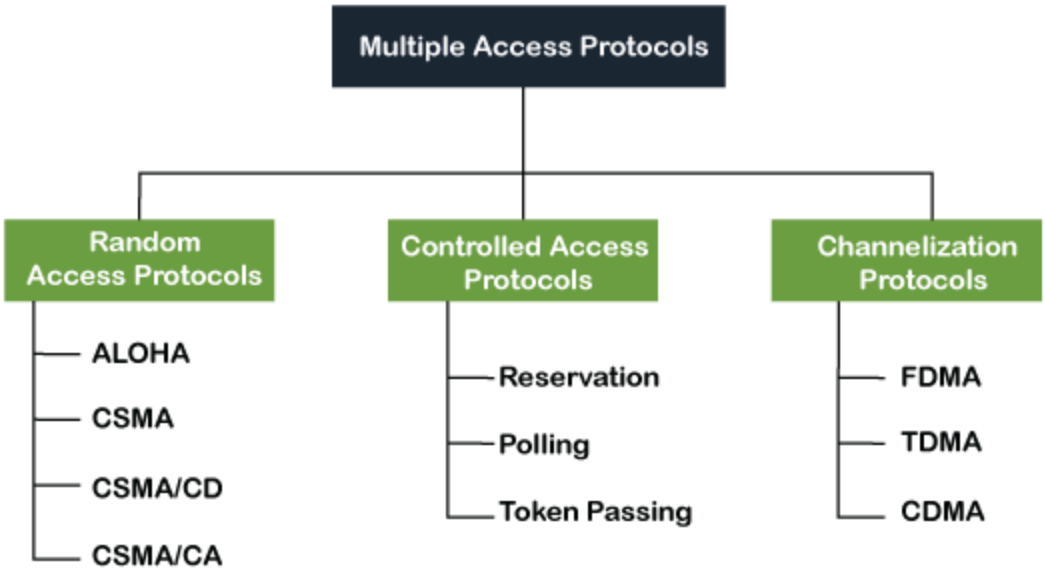
3. **Static Channel Allocation-** a single channel is divided among various users either on the basis of frequency (FDM) or on the basis of time (TDM). In FDM, fixed frequency is assigned to each user, whereas, in TDM, fixed time slot is assigned to each user.
4. **Dynamic Channel Allocation-** no user is assigned fixed frequency or fixed time slot. All users are dynamically assigned frequency or time slot, depending upon the requirements of the user

Multiple access protocols

When a sender and receiver have a dedicated link to transmit data packets, the data link control is enough to handle the channel. Suppose there is no dedicated path to communicate or transfer the data between two devices. In that case, multiple stations access the channel and simultaneously transmits the data over the channel. It may create collision and cross talk. Hence, the multiple access protocol is required to reduce the collision and avoid crosstalk between the channels.

For example, suppose that there is a classroom full of students. When a teacher asks a question, all the students (small channels) in the class start answering the question at the same time (transferring the data simultaneously). All the students respond at the same time due to which data is overlap or data lost. Therefore it is the responsibility of a teacher (multiple access protocol) to manage the students and make them one answer.

Following are the types of multiple access protocol that is subdivided into the different process as:



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.

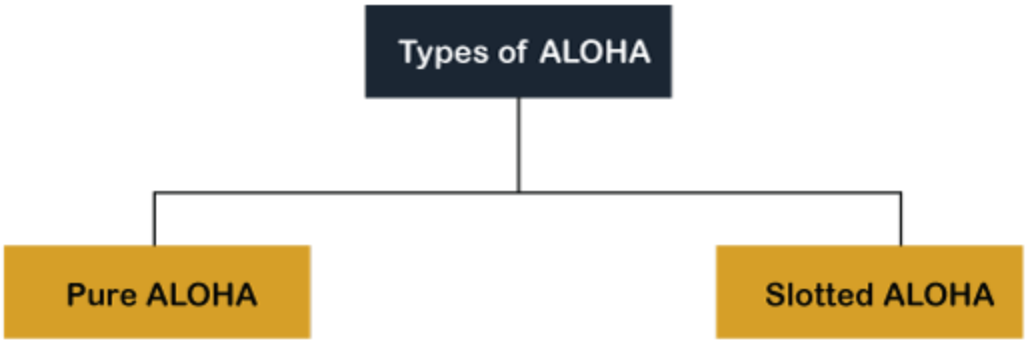
- Aloha
- CSMA
- CSMA/CD
- CSMA/CA

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.

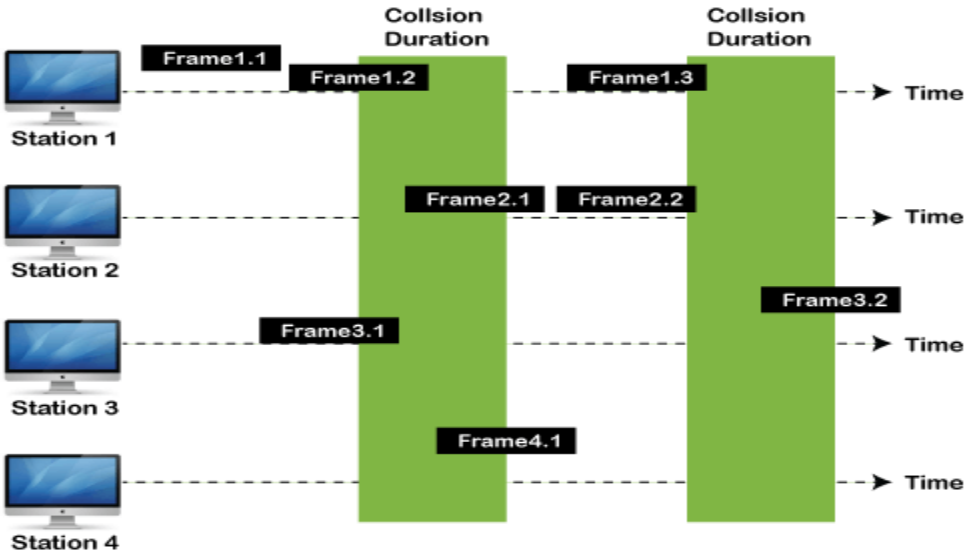


Pure Aloha

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 $S = G * e^{-2G}$.



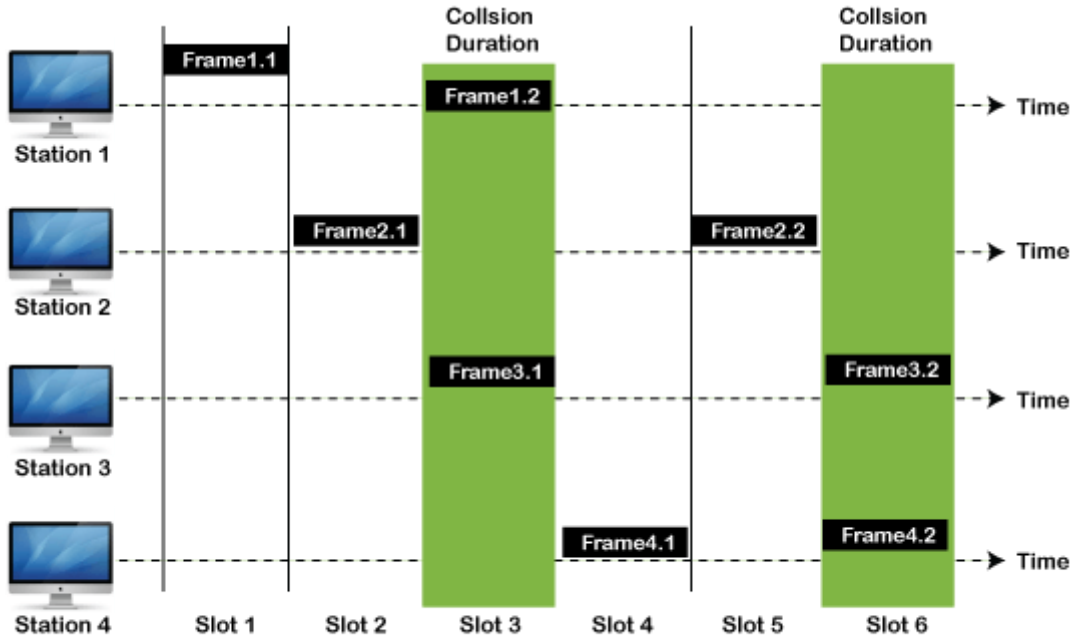
Frames in Pure ALOHA

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

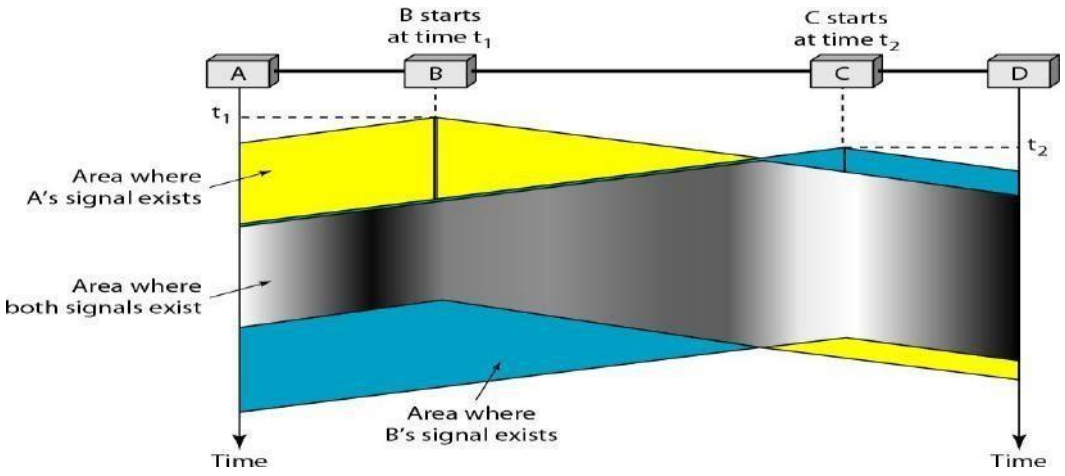
Carrier Sense Multiple Access (CSMA)

To minimize the chance of collision and, therefore, increase the performance, the CSMA method was developed. The chance of collision can be reduced if a station senses the medium before trying to use it. Carrier sense multiple access (CSMA) requires that each station first listen to the medium (or check the state of the medium) before sending. In other words, CSMA is based on the principle "sense before transmit" or "listen before talk."

CSMA can reduce the possibility of collision, but it cannot eliminate it. The reason for this is shown in below Figure. Stations are connected to a shared channel (usually a dedicated medium).

The possibility of collision still exists because of propagation delay; station may sense the medium and find it idle, only because the first bit sent by another station has not yet been received.

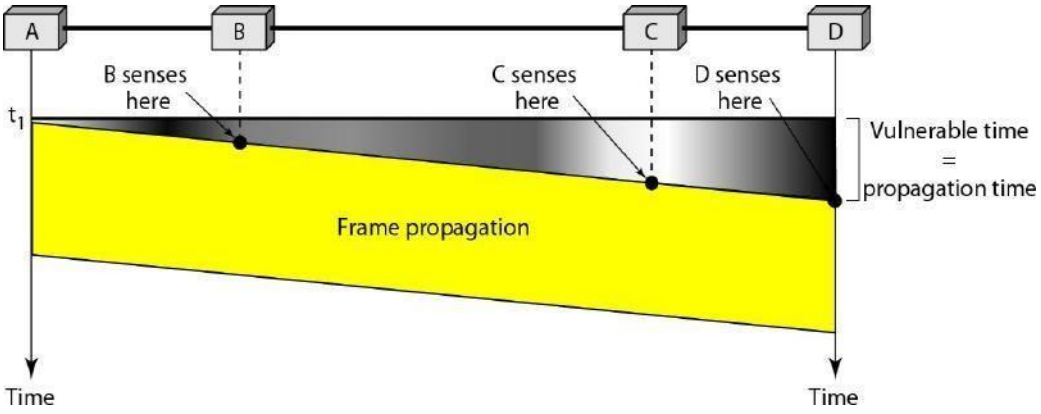
At time *tI'* station B senses the medium and finds it idle, so it sends a frame. At time *t2 (t2> tI)'* station C senses the medium and finds it idle because, at this time, the first bits from station B have not reached station C. Station C also sends a frame. The two signals collide and both frames are destroyed.



Space/time model of the collision in CSMA

Vulnerable Time

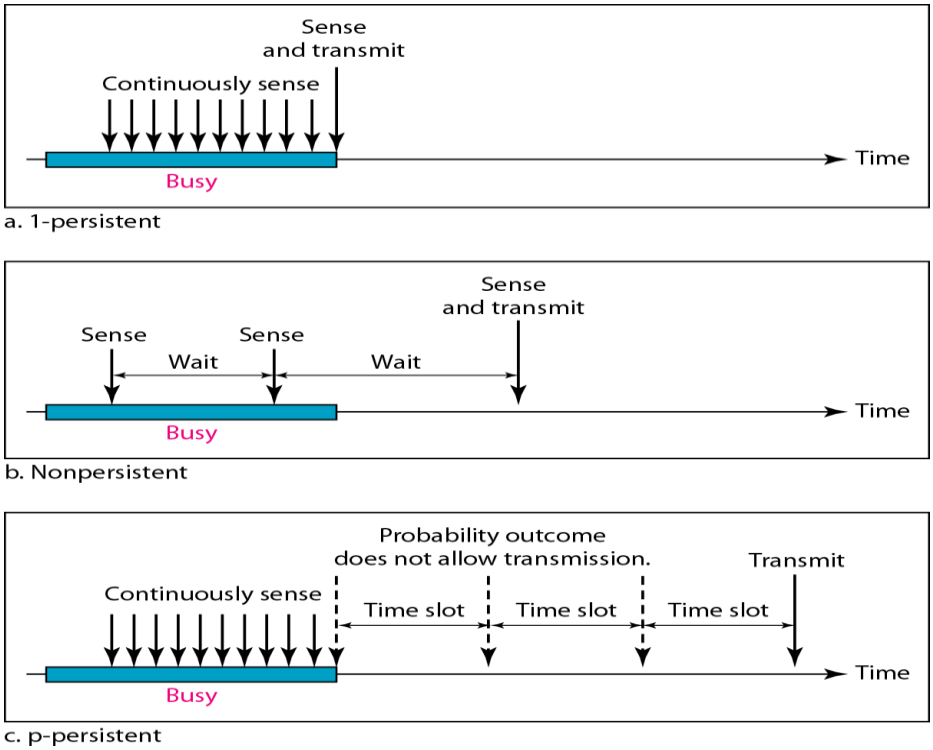
The vulnerable time for CSMA is the propagation time Tp . This is the time needed for a signal to propagate from one end of the medium to the other. When a station sends a frame, and any other station tries to send a frame during this time, a collision will result. But if the first bit of the frame reaches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



Vulnerable time in CSMA

Persistence Methods

What should a station do if the channel is busy? What should a station do if the channel is idle? Three methods have been devised to answer these questions: the 1-persistent method, the non-persistent method, and the p-persistent method



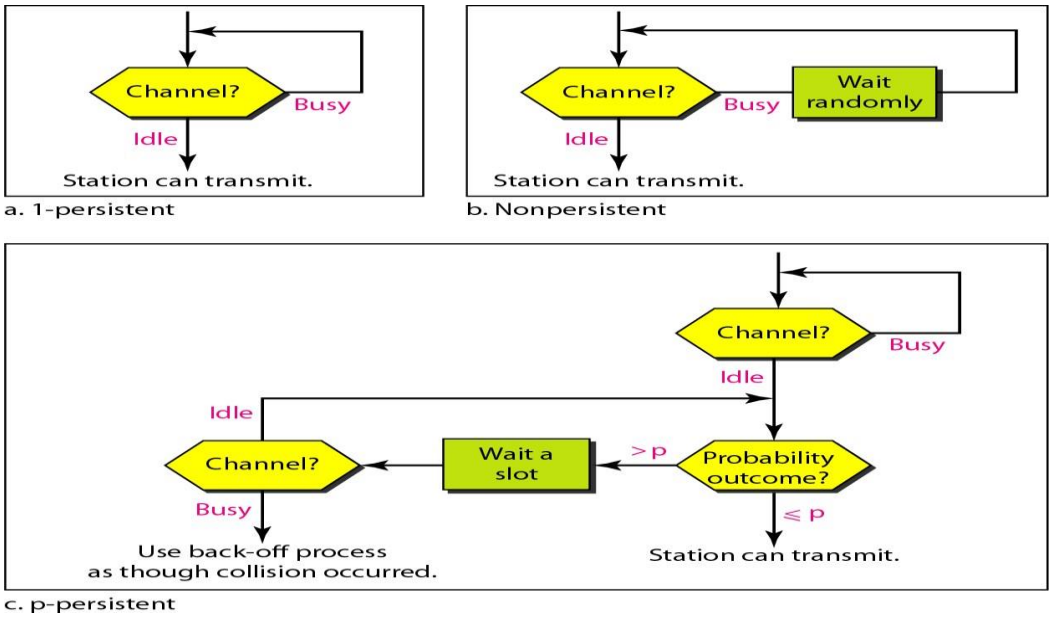
1-Persistent: In this method, after the station finds the line idle, it sends its frame immediately (with probability 1). This method has the highest chance of collision because two or more stations may find the line idle and send their frames immediately.

Non-persistent: a station that has a frame to send senses the line. If the line is idle, it sends immediately. If the line is not idle, it waits a random amount of time and then senses the line again. This approach reduces the chance of collision because it is unlikely that two or more stations will wait the same amount of time and retry to send simultaneously. However, this method reduces the efficiency of the network because the medium remains idle when there may be stations with frames to send.

p-Persistent: This is used if the channel has time slots with a slot duration equal to or greater than the maximum propagation time. The p-persistent approach combines the advantages of the other two strategies. It reduces the chance of collision and improves efficiency.

In this method, after the station finds the line idle it follows these steps:

1. With probability *p*, the station sends its frame.
2. With probability $q = 1 - p$, the station waits for the beginning of the next time slot and checks the line again.
   a. If the line is idle, it goes to step 1.
   b. If the line is busy, it acts as though a collision has occurred and uses the backoff procedure.

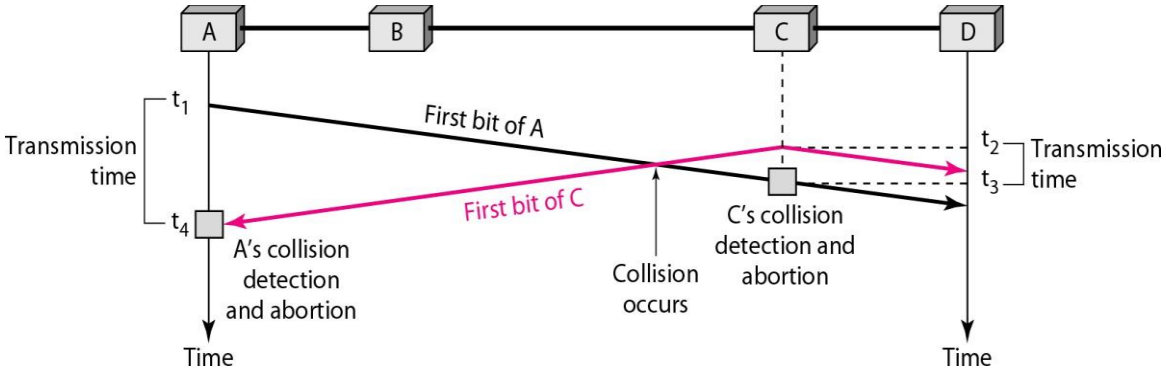


Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

The CSMA method does not specify the procedure following a collision. Carrier sense multiple access with collision detection (CSMA/CD) augments the algorithm to handle the collision.

In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If, however, there is a collision, the frame is sent again.

To better understand CSMA/CD, let us look at the first bits transmitted by the two stations involved in the collision. Although each station continues to send bits in the frame until it detects the collision, we show what happens as the first bits collide. In below Figure, stations A and C are involved in the collision.



Collision of the first bit in CSMA/CD

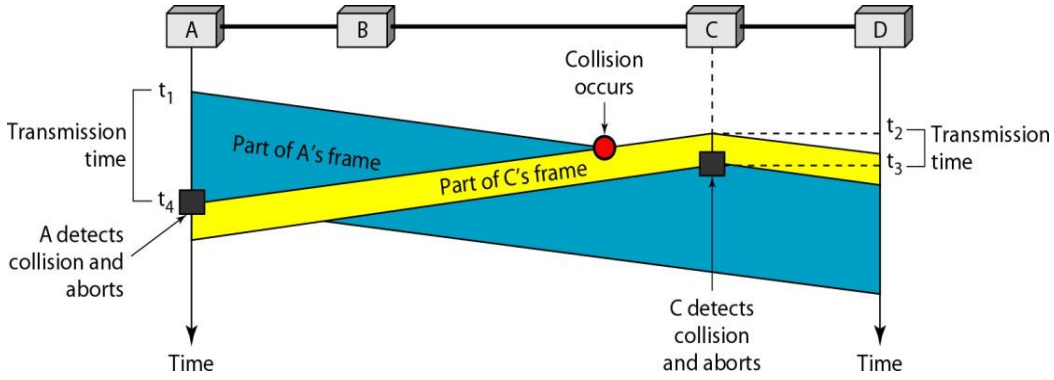
At time *t*1, station A has executed its persistence procedure and starts sending the bits of its frame. At time *t2*, station C has not yet sensed the first bit sent by A. Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and to the right. The collision occurs sometime after time *t2*. Station C detects a collision at time *t3* when it receives the first bit of A's frame. Station C immediately (or after a short time, but we

assume immediately) aborts transmission.

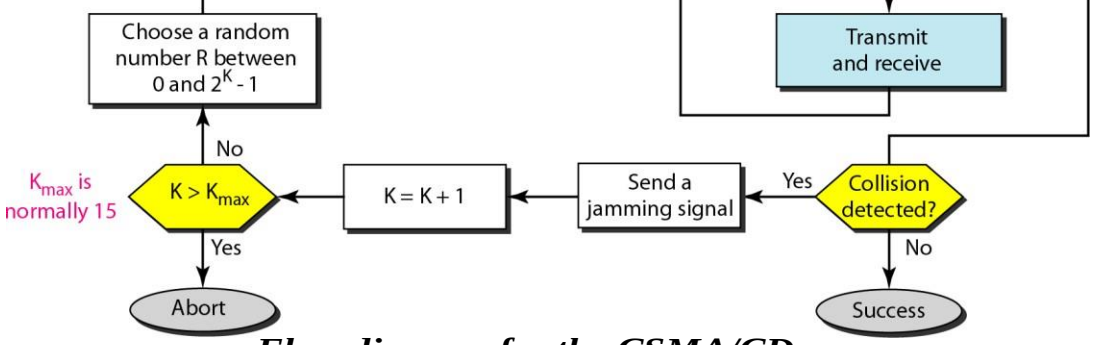
Station A detects collision at time *t4* when it receives the first bit of C's frame; it also immediately aborts transmission. Looking at the figure, we see that A transmits for the duration *t4*-t1*;* C transmits for the duration *t3* - *t2*.

Minimum Frame Size

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



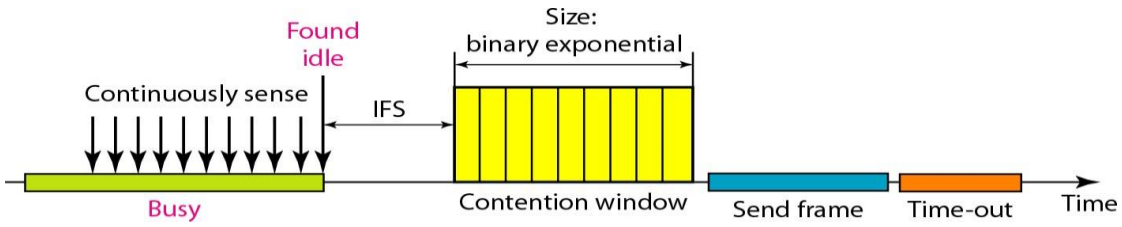
Collision and abortion in CSMA/CD



Flow diagram for the CSMA/CD

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

We need to avoid collisions on wireless networks because they cannot be detected. Carrier sense multiple access with collision avoidance *(CSM/CA)* was invented for wireless network. Collisions are avoided through the use of CSMA/CA's three strategies: the inter frame space, the contention window, and acknowledgments, as shown in Figure



Timing in CSMA/CA

Inter frame Space (IFS)

First, collisions are avoided by deferring transmission even if the channel is found idle. When an idle channel is found, the station does not send immediately. It waits for a period of time called the inter frame space or IFS.

Even though the channel may appear idle when it is sensed, a distant station may have already started transmitting. The distant station's signal has not yet reached this station. The IFS time allows the front of the transmitted signal by the distant station to reach this station. If after the IFS time the channel is still idle, the station can send, but it still needs to wait a time equal to the contention time. The IFS variable can also be used to prioritize stations or frame types. For example, a station that is assigned shorter IFS has a higher priority.In CSMA/CA, the IFS can also be used to define the priority of a station or a frame.

Contention Window

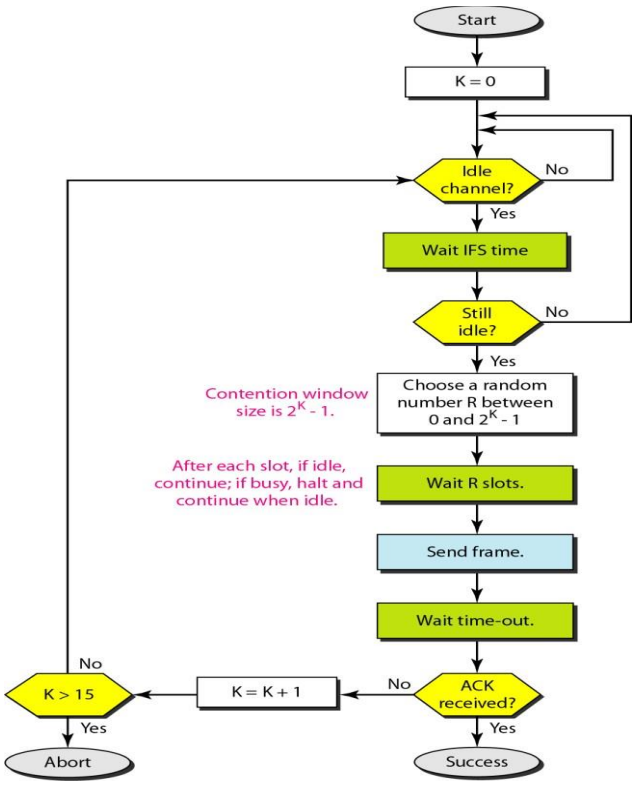
The contention window is an amount of time divided into slots. A station that is ready to send chooses a random number of slots as its wait time. The number of slots in the window changes according to the binary exponential back-off strategy. This means that it is set to one slot the first time and then doubles each time the station cannot detect an idle channel after the IFS time. This is very similar to the p-persistent method except that a random outcome defines the number of slots taken by the waiting station.

One interesting point about the contention window is that the station needs to sense the channel after each time slot. However, if the station finds the channel busy, it does not restart the process; it just stops the timer and restarts it when the channel is sensed as idle. This gives priority to the station with the longest waiting time.

In CSMA/CA, if the station finds the channel busy, it does not restart the timer of the contention window; it stops the timer and restarts it when the channel becomes idle.

Acknowledgment

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



This is the CSMA protocol with collision avoidance.

- The station ready to transmit, senses the line by using one of the persistent strategies.
- As soon as it finds the line to be idle, the station waits for an IFS (Inter frame space) amount of time.
- If then waits for some random time and sends the frame.
- After sending the frame, it sets a timer and waits for the acknowledgement from the receiver.
- If the acknowledgement is received before expiry of the timer, then the

transmission is successful.

- But if the transmitting station does not receive the expected acknowledgement before the timer expiry then it increments the back off parameter, waits for the back off time and re senses the line

Controlled Access Protocols

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

1 Reservation

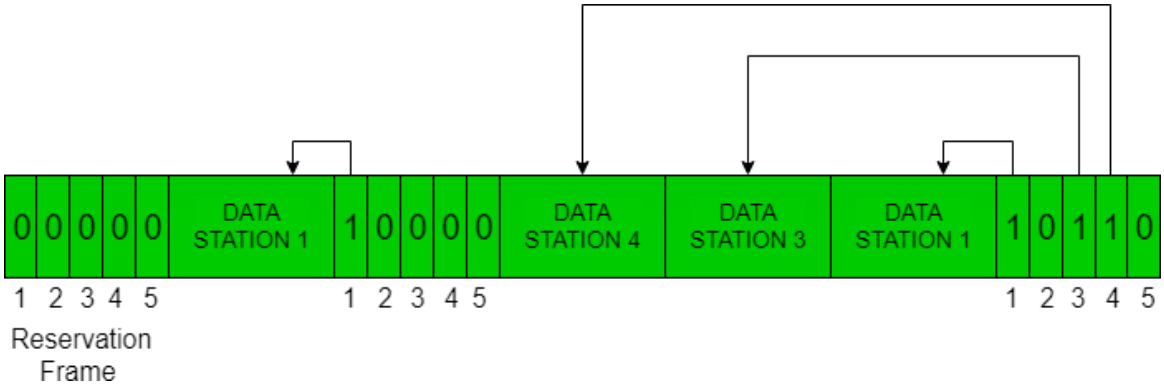
2 Polling

3 Token Passing

Reservation

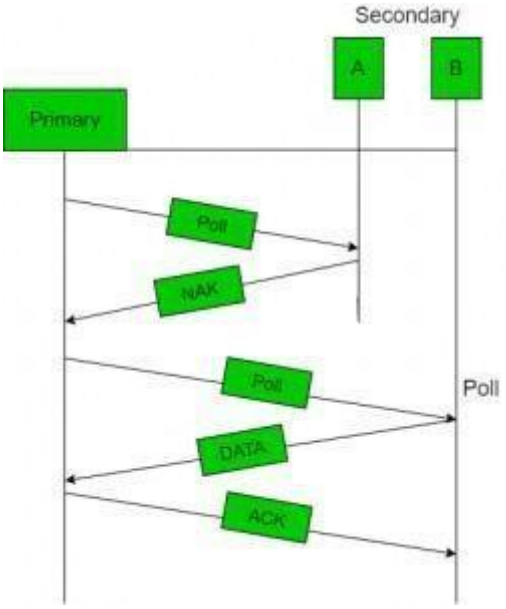
- In the reservation method, a station needs to make a reservation before sending data.
- The time line has two kinds of periods:
    1. Reservation interval of fixed time length
    2. Data transmission period of variable frames.
- If there are M stations, the reservation interval is divided into M slots, and each station has one slot.
- Suppose if station 1 has a frame to send, it transmits 1 bit during the slot1. No other station is allowed to transmit during this slot.
- In general, i th station may announce that it has a frame to send by inserting a 1 bit into i th slot. After all N slots have been checked, each station knows which stations wish to transmit.
- The stations which have reserved their slots transfer their frames in that order.
- After data transmission period, next reservation interval begins.
- Since everyone agrees on who goes next, there will never be any collisions.

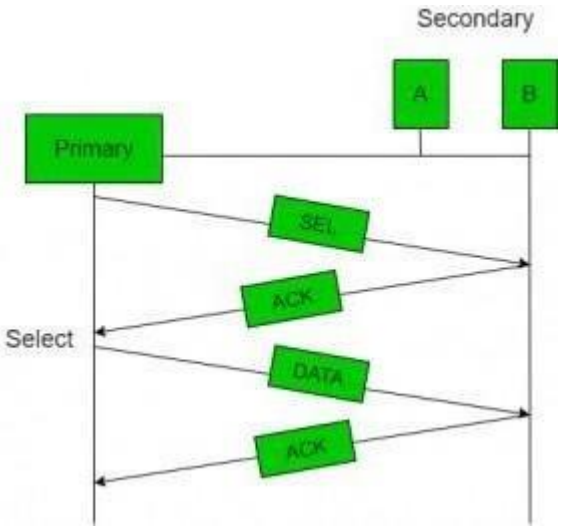
The following figure shows a situation with five stations and a five slot reservation frame. In the first interval, only stations 1, 3, and 4 ha ve made reservations. In the second interval, only station 1 has made a reservation.



Polling

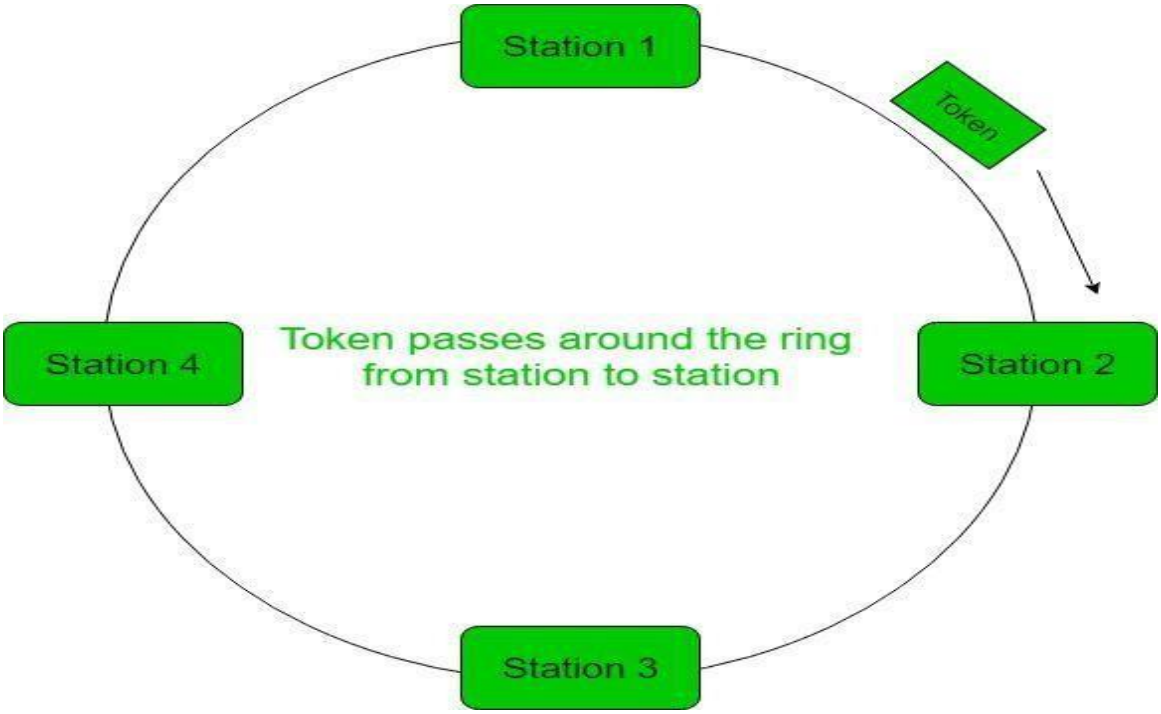
- Polling process is similar to the roll-call performed in class. Just like the teacher, a controller sends a message to each node in turn.
- In this, one acts as a primary station(controller) and the others are secondary stations. All data exchanges must be made through the controller.
- The message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





Token Passing

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



COLLISSION FREE PROTOCOLS:

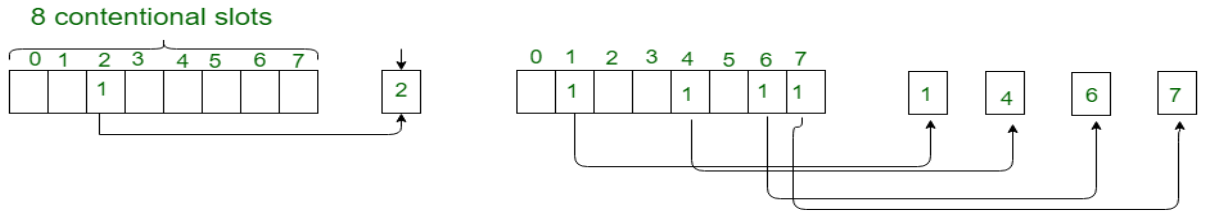
Almost collisions can be avoided in **CSMA/CD**, they can still occur during the contention period. the collision during contention period adversely affects the system performance, this happens when the cable is long and length of packet are short. This problem becomes serious as fiber optics network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

1. Bit-map Protocol:

Bit map protocol is collision free Protocol in 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 respective slot. For example if station 2 has a frame to send, it transmits a 1 bit during the second slot.

In general Station 1 Announce the fact that it has a frame questions 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*.



A Bit-map Protocol.

For analysing 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.

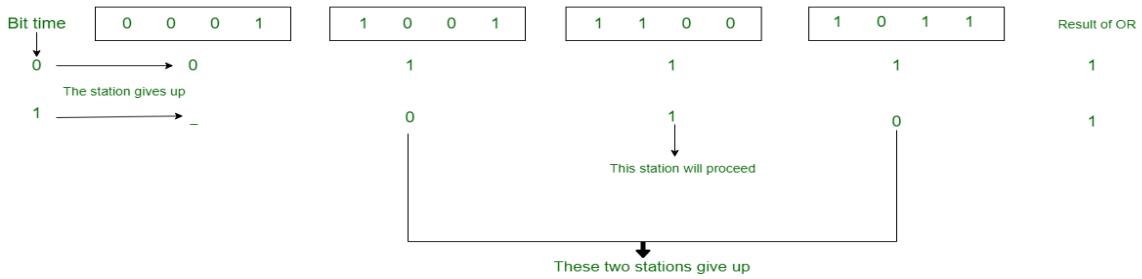
Generally, high numbered stations have to wait for half a scan before starting to transmit low numbered stations have to wait for half a scan (N/2 bit slots) before starting to transmit, low numbered stations have to wait on an average 1.5 N slots.

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 O Red 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 O Red together. Station 0001 see the 1MSB in another station addresses 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 bit is 1 at station 1100, swiss station 1011 and 1001 give up. Then station 110 starts transmitting a frame, after which another bidding cycle starts.



Binary countdown

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.

Adaptive Tree Walk Protocol:

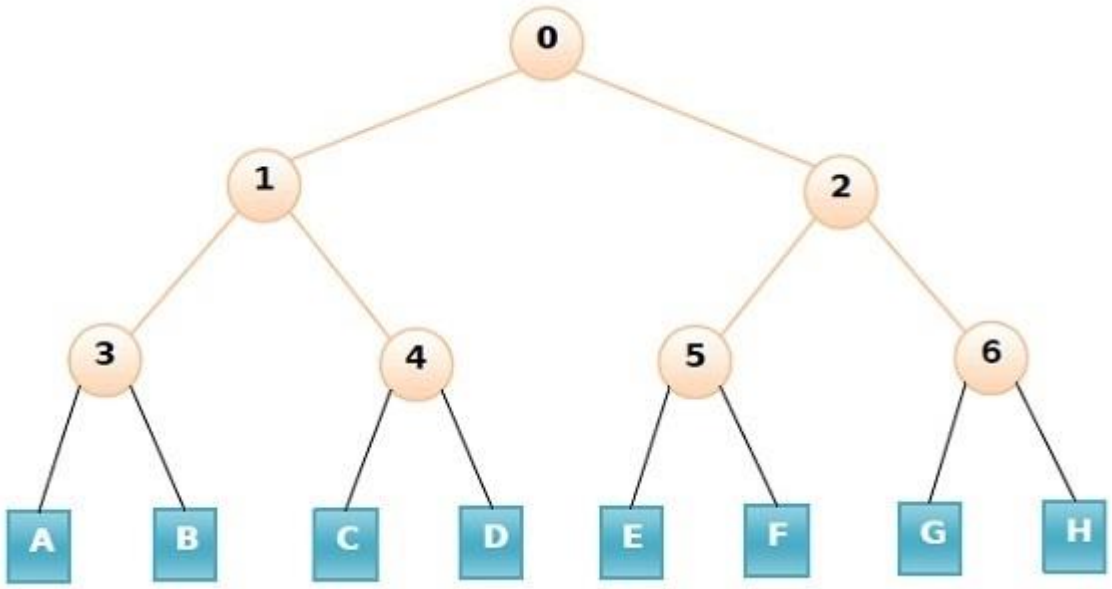
Adaptive Tree Walk Protocol is a technique for transmitting data over shared channels that combines the advantages of collision based protocols and collision free protocols.

In computer networks, when more than one station tries to transmit simultaneously via a shared channel, the transmitted data is garbled, an event called collision. In collision based protocols like ALOHA, all stations are permitted to transmit a frame without trying to detect whether the transmission channel is idle or busy. This works very good under light loads. Under heavy loads, collision free protocols are suitable, since channel access is resolved in the contention period that eliminates the possibilities of collisions.

In adaptive tree walk protocol, the stationed are partitioned into groups in a hierarchical manner. The contention period is divided into discrete time slots, and for each slot the contention rights of the stations are limited. Under light loads, all the stations can participate for contention each slot like ALOHA. However, under heavy loads, only a group can try for a given slot.

Working Principle

In adaptive tree walk protocol, the stations or nodes are arranged in the form of a binary tree as shown in the diagram. Here, the internal nodes (marked from 0 to 6) represent the groups while the leaf nodes (marked A to H) are the stations contending for network access.



Initially all nodes (A, B ……. G, H) are permitted to compete for the channel. If a node is successful in acquiring the channel, it transmits its frame. In case of collision, the nodes are divided into two groups –

- Stations under the group 1, i.e. A, B, C, D
- Stations under the group 2, i.e. E, F, G, H

Nodes belonging to only one of them is permitted for competing. Say, for slot 1, all stations under group 1 are allowed to contend. If one of the stations successfully acquires the channel, then it transmits to completion. In the next slot, i.e. slot 2, all stations under group 2 can contend.

However, if there is a collision, then the stations are further divided into groups as follows

- Stations under the group 3, i.e. A, B
- Stations under the group 4, i.e. C, D
- Stations under the group 5, i.e. E, F
- Stations under the group 6, i.e. G, H

In order to locate the contending stations, depth-first search algorithm is used. The same principle of contention is applied, only for those groups that has some contending stations. The division continues if collisions occur, until each group contains only 1 node.

Wireless Local Area Network: WIRELESS LANS

Wireless LAN stands for **Wireless Local Area Network**. It is also called LAWN (**Local Area Wireless Network**). WLAN is one in which a mobile user can connect to a Local Area Network (LAN) through a wireless connection.

The IEEE 802.11 group of standards defines the technologies for wireless LANs. For path sharing, 802.11 standard uses the Ethernet protocol and CSMA/CA (carrier sense multiple access with collision avoidance). It also uses an encryption method i.e. wired equivalent privacy algorithm.

Wireless LANs provide high speed data communication in small areas such as building or an office. WLANs allow users to move around in a confined area while they are still connected to the network.

In some instance wireless LAN technology is used to save costs and avoid laying cable, while in other cases, it is the only option for providing high-speed internet access to the public. Whatever the reason, wireless solutions are popping up everywhere.

Examples of WLANs that are available today are NCR's waveLAN and Motorola's ALTAIR.

Advantages of WLANs

- **Flexibility:** Within radio coverage, nodes can communicate without further restriction. Radio waves can penetrate walls, senders and receivers can be placed anywhere (also non-visible, e.g., within devices, in walls etc.).
- **Planning:** Only wireless ad-hoc networks allow for communication without previous planning, any wired network needs wiring plans.
- **Design:** Wireless networks allow for the design of independent, small devices which can for example be put into a pocket. Cables not only restrict users but also designers of small notepads, PDAs, etc.
- **Robustness:** Wireless networks can handle disasters, e.g., earthquakes, flood etc. whereas, networks requiring a wired infrastructure will usually break down completely in disasters.
- **Cost:** The cost of installing and maintaining a wireless LAN is on average lower than the cost of installing and maintaining a traditional wired LAN, for two reasons. First, after providing wireless access to the wireless network via an access point for the first user, adding additional users to a network will not increase the cost. And second, wireless LAN eliminates the direct costs of cabling and the labor associated with installing and repairing it.
- **Ease of Use:** Wireless LAN is easy to use and the users need very little new information to take advantage of WLANs.

Disadvantages of WLANs

- **Quality of Services:** Quality of wireless LAN is typically lower than wired networks. The main reason for this is the lower bandwidth due to limitations is radio transmission, higher

error rates due to interference and higher delay/delay variation due to extensive error correction and detection mechanisms.

- **Proprietary Solutions:** Due to slow standardization procedures, many companies have come up with proprietary solutions offering standardization functionality plus many enhanced features. Most components today adhere to the basic standards IEEE 802.11a or 802.11b.
- **Restrictions:** Several govt. and non-govt. institutions world-wide regulate the operation and restrict frequencies to minimize interference.
- **Global operation:** Wireless LAN products are sold in all countries so, national and international frequency regulations have to be considered.
- **Low Power:** Devices communicating via a wireless LAN are typically power consuming, also wireless devices running on battery power. Whereas the LAN design should take this into account and implement special power saving modes and power management functions.
- **License free operation:** LAN operators don't want to apply for a special license to be able to use the product. The equipment must operate in a license free band, such as the 2.4 GHz ISM band.
- **Robust transmission technology:** If wireless LAN uses radio transmission, many other electrical devices can interfere with them (such as vacuum cleaner, train engines, hair dryers, etc.).Wireless LAN transceivers cannot be adjusted for perfect transmission is a standard office or production environment.

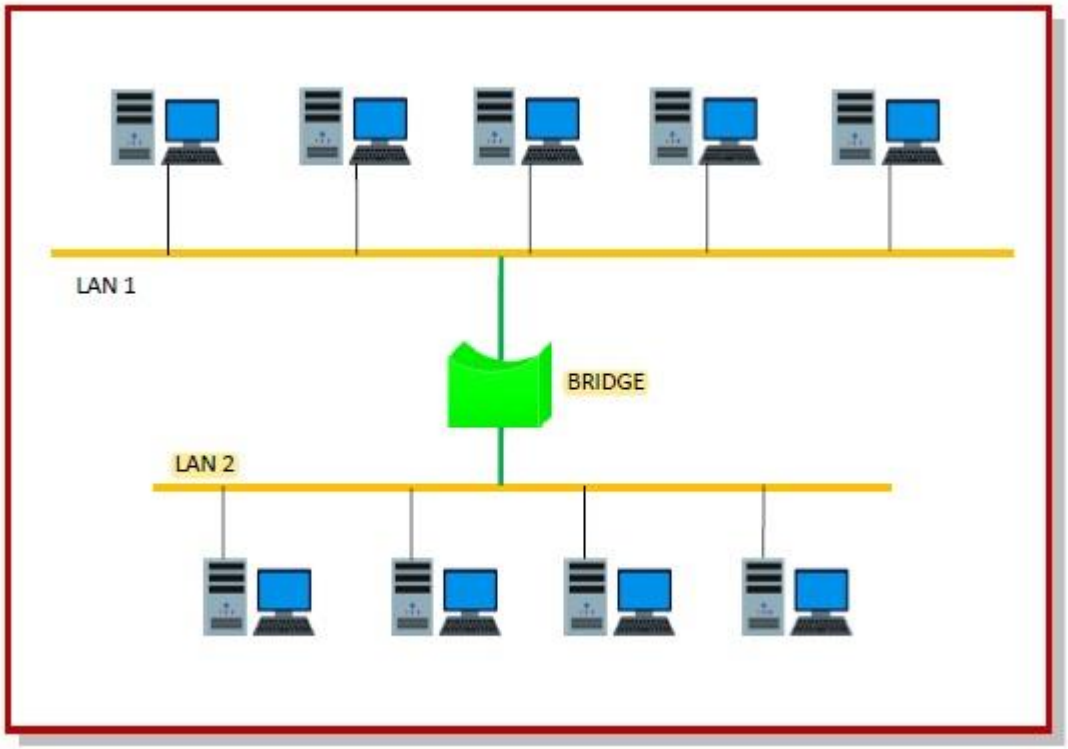
What is Data Link Layer Switching

Network switching is the process of forwarding data frames or packets from one port to another leading to data transmission from source to destination. Data link layer is the second layer of the Open System Interconnections (OSI) model whose function is to divide the stream of bits from physical layer into data frames and transmit the frames according to switching requirements. 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 −

- simple bridging
- multi-port bridging
- learning or transparent bridging