

MODULE 2

DATA LINK CONTROL

DLC SERVICES

The data link control (DLC) deals with procedures for communication between two adjacent nodes—node-to-node communication—no matter whether the link is dedicated or broadcast. Data link control functions include framing and flow and error control.

Framing Data

Transmission in the physical layer means moving bits in the form of a signal from the source to the destination. The physical layer provides bit synchronization to ensure that the sender and receiver use the same bit durations and timing.

The data-link layer, on the other hand, needs to pack bits into frames, so that each frame is distinguishable from another.

Framing in the data-link layer separates a message from one source to a destination by adding a sender address and a destination address. The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt.

Although the whole message could be packed in one frame, that is not normally done. One reason is that a frame can be very large, making flow and error control very inefficient. When a message is carried in one very large frame, even a single-bit error would require the retransmission of the whole frame. When a message is divided into smaller frames, a single-bit error affects only that small frame.

Frame Size

Frames can be of fixed or variable size.

Fixed-size framing: there is no need for defining the boundaries of the frames; the size itself can be used as a delimiter. An example of this type of framing is the (Asynchronous transfer mode)ATM WAN, which uses frames of fixed size called cells.

Variable-size framing: prevalent in local-area networks. In variable-size framing, we need a way to define the end of one frame and the beginning of the next.

Two approaches were used for this purpose: a character-oriented approach and a bit-oriented approach.

Character-Oriented Framing

In character-oriented (or byte-oriented) framing, data to be carried are 8-bit characters from a coding system such as ASCII. The header, which normally carries the source and destination addresses and other control information, and the trailer, which carries error detection redundant bits, are also multiples of 8 bits.

To separate one frame from the next, an 8-bit (1-byte) flag is added at the beginning and the end of a frame. The flag, composed of **protocol-dependent special characters**, signals the start or end of a frame. Figure shows the format of a frame in a character-oriented protocol.

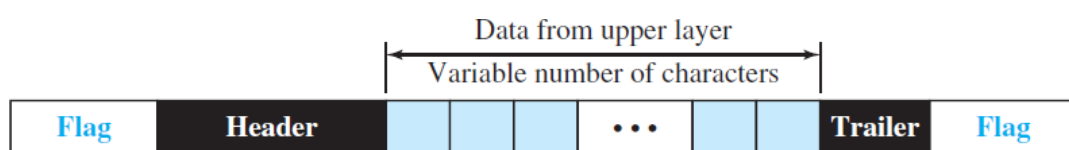


fig: A frame in a character-oriented protocol

Character-oriented framing was popular when only **text was exchanged** by the data-link layers. The flag could be selected to be any character not used for text communication. Now, however,

we send other types of information such as graphs, audio, and video;

Any character used for the flag could also be part of the information. If this happens, the receiver, when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame.

To fix this problem, a **byte-stuffing** strategy was added to character-oriented framing. In byte stuffing (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag. The data section is stuffed with an extra byte. This byte is usually called the escape character (ESC) and has a predefined bit pattern. Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not as a delimiting flag. Figure shows the situation.

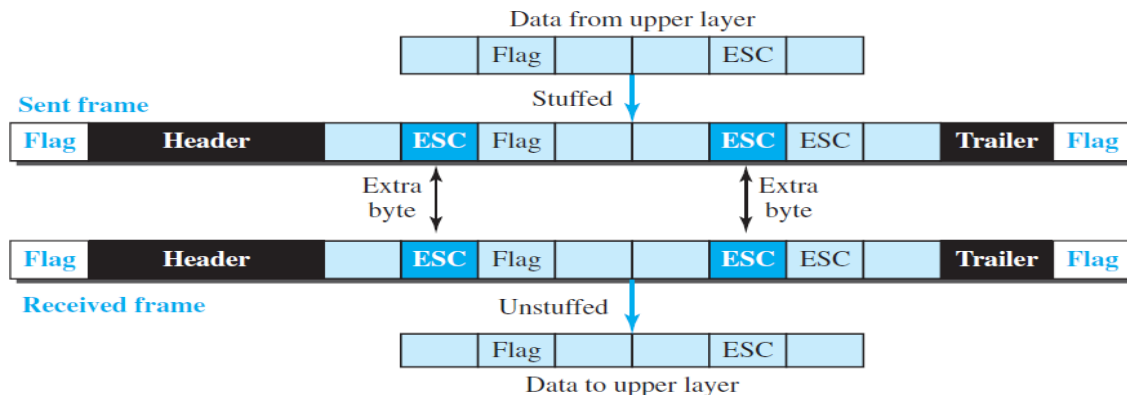


fig: Byte stuffing and unstuffing

Byte stuffing by the escape character allows the presence of the flag in the data section of the frame, but it creates another problem. What happens if the text contains one or more escape characters followed by a byte with the same pattern as the flag?

The receiver removes the escape character, but keeps the next byte, which is incorrectly interpreted as the end of the frame. To solve this problem, the escape characters that are part of the text must also be marked by another escape character. In other words, if the escape character is part of the text, an extra one is added to show that the second one is part of the text.

Character-oriented protocols present another problem in data communications. The universal coding systems in use today, such as Unicode, have 16-bit and 32-bit characters that conflict with 8-bit characters. We can say that, in general, the tendency is moving toward the bit-oriented protocols that we discuss next.

Bit-Oriented Framing

In bit-oriented framing, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on. However, in addition to headers (and possible trailers), we still need a delimiter to separate one frame from the other. Most protocols use a special 8-bit pattern flag, 01111110, as the delimiter to define the beginning and the end of the frame, as shown in **fig**

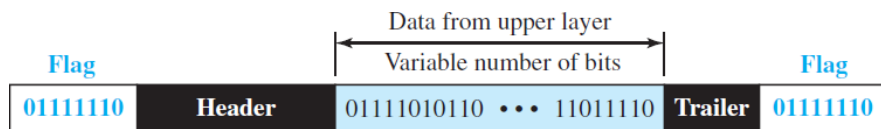


fig: A frame in bit-oriented protocol

If the flag pattern appears in the data, need to somehow inform the receiver that this is not the end of the frame. We do this by stuffing 1 single bit (instead of 1 byte) to prevent the pattern from looking like a flag. The strategy is called bit stuffing. In bit stuffing, if a 0 and five consecutive 1 bits are encountered, an extra 0 is added. This extra stuffed bit is eventually

removed from the data by the receiver. Note that the extra bit is added after one 0 followed by five 1s regardless of the value of the next bit. This guarantees that the flag field sequence does not inadvertently appear in the frame.

Bit stuffing is the process of adding one extra 0 whenever five consecutive 1s follow a 0 in the data, so that the receiver does not mistake the pattern 0111110 for a flag.

Figure shows bit stuffing at the sender and bit removal at the receiver. Note that even if we have a 0 after five 1s, we still stuff a 0. The 0 will be removed by the receiver. This means that if the flag like pattern 01111110 appears in the data, it will change to 011111010 (stuffed) and is not mistaken for a flag by the receiver. The real flag 01111110 is not stuffed by the sender and is recognized by the receiver.

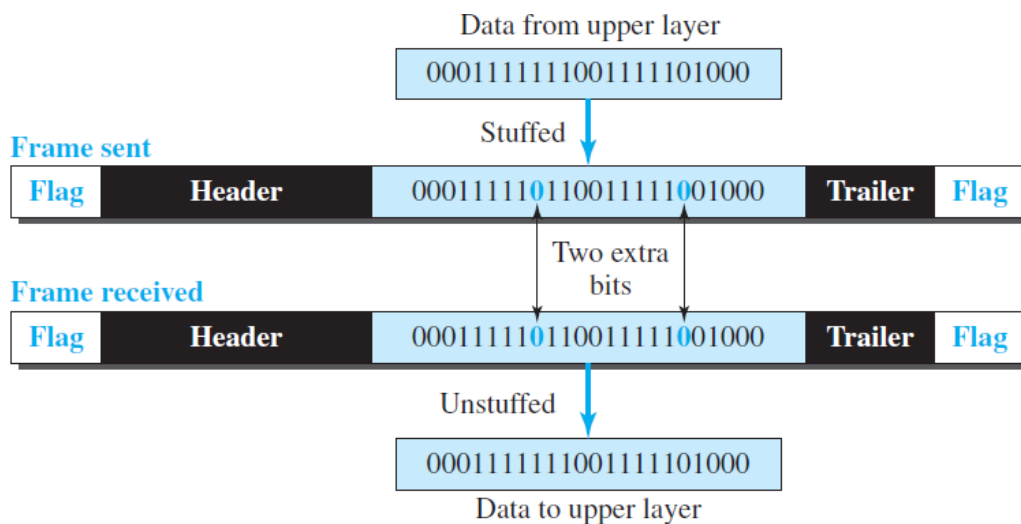


fig: Bit stuffing and unstuffing

Flow and Error Control

One of the responsibilities of the data-link control sublayer is flow and error control at the data-link layer.

Flow Control Whenever an entity produces items and another entity consumes them, there should be a balance between production and consumption rates. If the items are produced faster than they can be consumed, the consumer can be overwhelmed and may need to discard some items. If the items are produced more slowly than they can be consumed, the consumer must wait, and the system becomes less efficient. Flow control is related to the first issue. We need to prevent losing the data items at the consumer site.

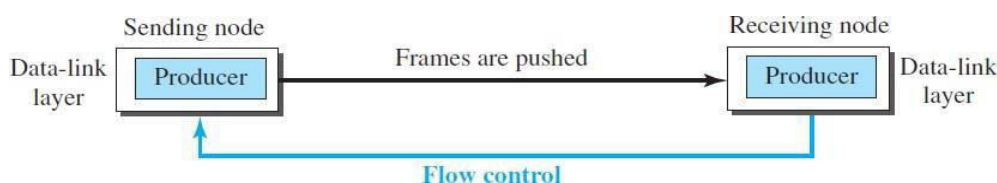


fig: Flow control at the data link layer

The figure shows that the data-link layer at the sending node tries to push frames toward the data-link layer at the receiving node. If the receiving node cannot process and deliver the packet to its network at the same rate that the frames arrive, it becomes overwhelmed with frames. Flow control in this case can be feedback from the receiving node to the sending node to stop or slow down pushing frames.

Buffers

Although flow control can be implemented in several ways, one of the solutions is normally to use two buffers; one at the sending data-link layer and the other at the receiving data-link layer. A buffer is a set of memory locations that can hold packets at the sender and receiver. The flow control communication can occur by sending signals from the consumer to the producer. When the buffer of the receiving data-link layer is full, it informs the sending data-link layer to stop pushing frames.

Error Control

Since the underlying technology at the physical layer is not fully reliable, we need to implement error control at the data-link layer to prevent the receiving node from delivering corrupted packets to its network layer. Error control at the data-link layer is normally very simple and implemented using one of the following two methods. In both methods, a CRC is added to the frame header by the sender and checked by the receiver.

❑ In the first method, if the frame is corrupted, it is silently discarded; if it is not corrupted, the packet is delivered to the network layer. This method is used mostly in wired LANs such as Ethernet.

❑ In the second method, if the frame is corrupted, it is silently discarded; if it is not corrupted, an acknowledgment is sent (for the purpose of both flow and error control) to the sender.

Combination of Flow and Error Control

Flow and error control can be combined. In a simple situation, the acknowledgment that is sent for flow control can also be used for error control to tell the sender the packet has arrived uncorrupted. The lack of acknowledgment means that there is a problem in the sent frame. We show this situation when we discuss some simple protocols in the next section. A frame that carries an acknowledgment is normally called an ACK to distinguish it from the data frame.

Connectionless and Connection-Oriented

A DLC protocol can be either connectionless or connection-oriented..

Connectionless Protocol

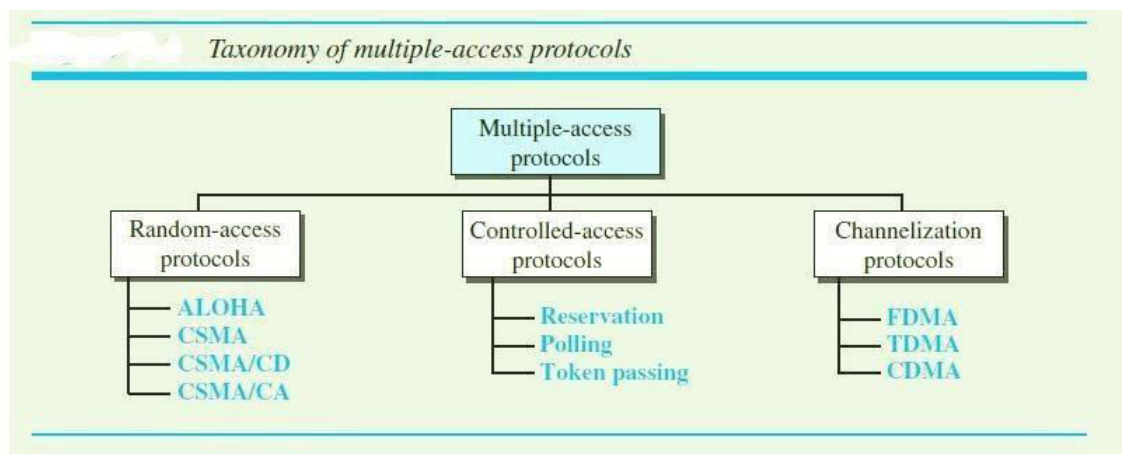
In a connectionless protocol, frames are sent from one node to the next without any relationship between the frames; each frame is independent. Note that the term connectionless here does not mean that there is no physical connection (transmission medium) between the nodes; it means that there is no connection between frames. The frames are not numbered and there is no sense of ordering. Most of the data-link protocols for LANs are connectionless protocols.

Connection-Oriented Protocol

In a connection-oriented protocol, a logical connection should first be established between the two nodes (setup phase). After all frames that are somehow related to each other are transmitted (transfer phase), the logical connection is terminated (teardown phase). In this type of communication, the frames are numbered and sent in order. If they are not received in order, the receiver needs to wait until all frames belonging to the same set are received and then deliver them in order to the network layer. Connection oriented protocols are rare in wired LANs, but we can see them in some point-to-point protocols, some wireless LANs, and some WANs.

Media Access Control (MAC)

- When nodes or stations are connected and use a common link, called a multipoint or broadcast link, we need a multiple-access protocol to coordinate access to the link.
- Many protocols have been devised to handle access to a shared link. All of these protocols belong to a sub layer in the data-link layer called media access control (MAC).



RANDOM ACCESS

- In **random-access** or **contention** methods, no station is superior to another station and none is assigned control over another.
- At each instance, a station that has data to send uses a procedure defined by the protocol to make a decision on whether or not to send.
- This decision depends on the state of the medium (idle or busy). In other words, each station can transmit when it desires on the condition that it follows the predefined procedure, including testing the state of the medium.
- Two features give this method its name. First, there is no scheduled time for a station to transmit. Transmission is random among the stations. That is why these methods are called **random access**. Second, no rules specify which station should send next. Stations compete with one another to access the medium. That is why these methods are also called **contention methods**.
- In a random-access method, each station has the right to the medium without being controlled by any other station. However, if more than one station tries to send, there is an access conflict “**collision**” and the frames will be either destroyed or modified.
- The random-access methods have evolved from a very interesting protocol known as ALOHA, which used a very simple procedure called **multiple access (MA)**.
- The method was improved with the addition of a procedure that forces the station to sense the medium before transmitting. This was called **carrier sense multiple access (CSMA)**.
- CSMA method later evolved into two parallel methods: **carrier sense multiple access with collision detection (CSMA/CD)**, which tells the station what to do when a collision is detected, and **carrier sense multiple access with collision avoidance (CSMA/CA)**, which tries to avoid the collision.

ALOHA

- ALOHA, the earliest random access method, was developed at the University of Hawaii in early 1970. It was designed for a radio (wireless) LAN, but it can be used on any shared medium.
- The medium is shared between the stations. When a station sends data, another station may attempt to do so at the same time. The data from the two stations collide and become garbled.

Pure ALOHA

- The original ALOHA protocol is called **pure ALOHA**. This is a simple but elegant protocol.
- The idea is that each station sends a frame whenever it has a frame to send (multiple access) there is only one channel to share, there is the possibility of collision between frames from different stations.

Figure 1 below shows an example of frame collisions in pure ALOHA

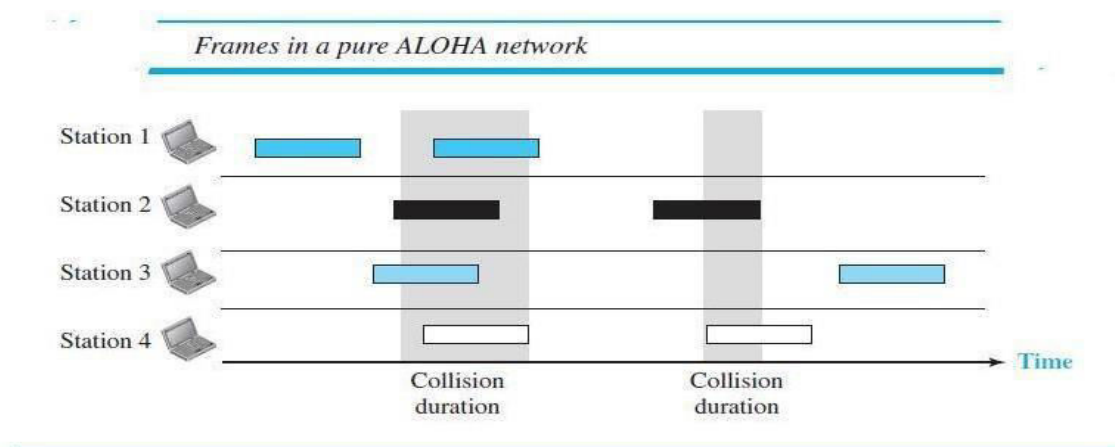


Figure 1: Frames in pure ALOHA network

- There are four stations (unrealistic assumption) that contend with one another for access to the shared channel. The above figure shows that each station sends two frames, there are a total of eight frames on the shared medium.
- Some of these frames collide because multiple frames are in contention for the shared channel. Figure 1 shows that only two frames survive: one frame from station 1 and one frame from station 3.
- If one bit of a frame coexists on the channel with one bit from another frame, there is a collision and both will be destroyed. It is obvious that the frames have to be resend that have been destroyed during transmission.
- The pure ALOHA protocol relies on acknowledgments from the receiver. When a station sends a frame, it expects the receiver to send an acknowledgment. If the acknowledgment does not arrive after a time-out period, the station assumes that the frame (or the acknowledgment) has been destroyed and resends the frame.
- A collision involves two or more stations. If all these stations try to resend their frames after the time-out, the frames will collide again.
- Pure ALOHA dictates that when the time-out period passes, each station waits a random amount of time before resending its frame. The randomness will help avoid more collisions. This time is called as the back off time T_B .

- Pure ALOHA has a second method to prevent congesting the channel with retransmitted frames. After a maximum number of retransmission attempts K_{max} , a station must give up and try later. Figure 1 shows the procedure for pure ALOHA based on the above strategy.
- The time-out period is equal to the maximum possible round-trip propagation delay, which is twice the amount of time required to send a frame between the two most widely separated stations ($2 \times T_p$).
- The backoff time T_B is a random value that normally depends on K (the number of attempted unsuccessful transmissions).
- In this method, for each retransmission, a multiplier $R = 0$ to 2^K is randomly chosen and multiplied by T_p (maximum propagation time) or T_{fr} (the average time required to send out a frame) to find T_B .

Note: The range of the random numbers increases after each collision. The value of K_{max} is usually chosen as 15.

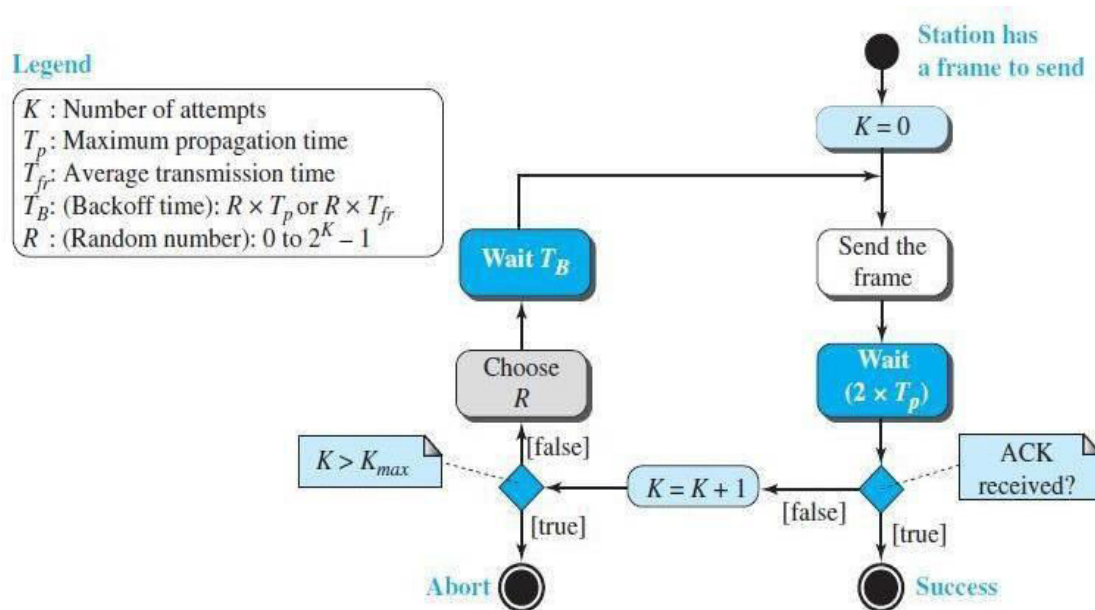


Figure 3: Procedure for pure ALOHA protocol

PROBLEM 1

The stations on a wireless ALOHA network are a maximum of 600 km apart. If we assume that signals propagate at 3×10^8 m/s, find T_p . Assume $K = 2$, Find the range of R .

Solution: $T_p = (600 \times 10^3) / (3 \times 10^8) = 2$ ms.

The range of R is, $R = (0 \text{ to } 2^k) = \{0, 1, 2, 3\}$.

This means that T_B can be 0, 2, 4, or 6 ms, based on the outcome of the random variable R .

Vulnerable time

The length of time in which there is a possibility of collision. The stations send fixed-length frames with each frame taking T_{fr} seconds to send. Figure 4 shows the vulnerable time for station B.

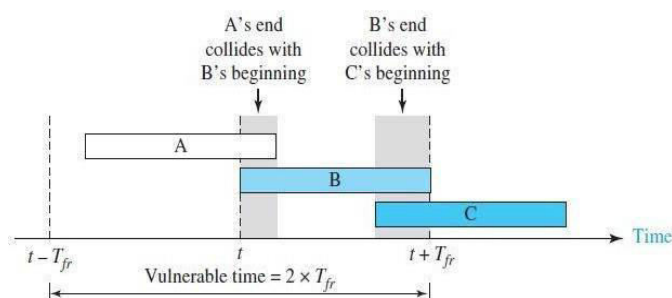


Figure 4: Vulnerable time for pure ALOHA protocol

Station B starts to send a frame at time t . Imagine station A has started to send its frame after $t - T_{fr}$. This leads to a collision between the frames from station B and station A. On the other hand, suppose that station C starts to send a frame before time $t + T_{fr}$. There is also a collision between frames from station B and station C.

From the Figure 4, it can be seen that the vulnerable time during which a collision may occur in pure ALOHA is 2 times the frame transmission time.

$$\text{Pure ALOHA vulnerable time} = 2 \times T_{fr}$$

PROBLEM 2

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the requirement to make this frame collision-free?

Solution: Average frame transmission time T_{fr} is 200 bits/200 kbps or 1 ms.

The vulnerable time is $2 \times 1 \text{ ms} = 2 \text{ ms}$.

This means no station should send later than 1 ms before this station starts transmission and no station should start sending during the period (1 ms) that this station is sending.

Throughput

G = the average number of frames generated by the system during one frame transmission time (T_{fr})

S = the average number of successfully transmitted frames for pure ALOHA. And is given by,

$$S = G \times e^{-2G} \text{. -----(1)}$$

Differentiate equation (1) with respect to G and equate it to 0, we get $G = 1/2$. Substitute $G=1/2$ in equation (1) to get S_{max} .

The maximum throughput $S_{max} = 0.184$.

- If one-half a frame is generated during one frame transmission time (one frame during two frame transmission times), then 18.4 percent of these frames reach their destination successfully.
- G is set to $G = 1/2$ to produce the maximum throughput because the vulnerable time is 2 times the frame transmission time. Therefore, if a station generates only one frame in this vulnerable time (and no other stations generate a frame during this time), the frame will reach its destination successfully.

NOTE: The throughput for pure ALOHA is $S = G \times e^{-2G}$.

The maximum throughput $S_{max} = 1/(2e) = 0.184$ when $G = (1/2)$.

PROBLEM 3

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces

- 1000 frames per second?
- 500 frames per second?
- 250 frames per second?

Solution: The frame transmission time T_{fr} is 200/200 kbps or 1 ms.

(a) If the system creates 1000 frames per second, or 1 frame per millisecond ($1 \text{ s} = 1000 \text{ ms}$) then $G = 1$ (because G = number of frames generated for one T_{fr}).

$S = G \times e^{-2G} = 0.135$ (13.5 percent). This means that the throughput is $1000 \times 0.135 = 135$ frames.

Only 135 frames out of 1000 will probably survive.

(b) If the system creates 500 frames per second, or 1/2 frame per millisecond (1s = 1000ms) then $G = 1/2$ (because $G =$ number of frames generated for one T_{fr}).

$S = G \times e^{-2G} = 0.184$ (18.4 percent). This means that the throughput is $500 \times 0.184 = 92$ frames. Only 92 frames out of 500 will probably survive.

(c) If the system creates 250 frames per second, or 1/4 frame per millisecond (1s = 1000ms) then $G = 1/4$ (because $G =$ number of frames generated for one T_{fr}).

$S = G \times e^{-2G} = 0.152$ (15.2 percent). This means that the throughput is $250 \times 0.152 = 38$ frames. Only 38 frames out of 250 will probably survive.

Slotted ALOHA

- Pure ALOHA has a vulnerable time of $2 \times T_{fr}$. This is so because there is no rule that defines when the station can send.
- A station may send soon after another station has started or just before another station has finished. Slotted ALOHA was invented to improve the efficiency of pure ALOHA.
- In **slotted ALOHA** we divide the time into slots of T_{fr} seconds and force the station to send only at the beginning of the time slot. Figure 5 shows an example of frame collisions in slotted ALOHA.

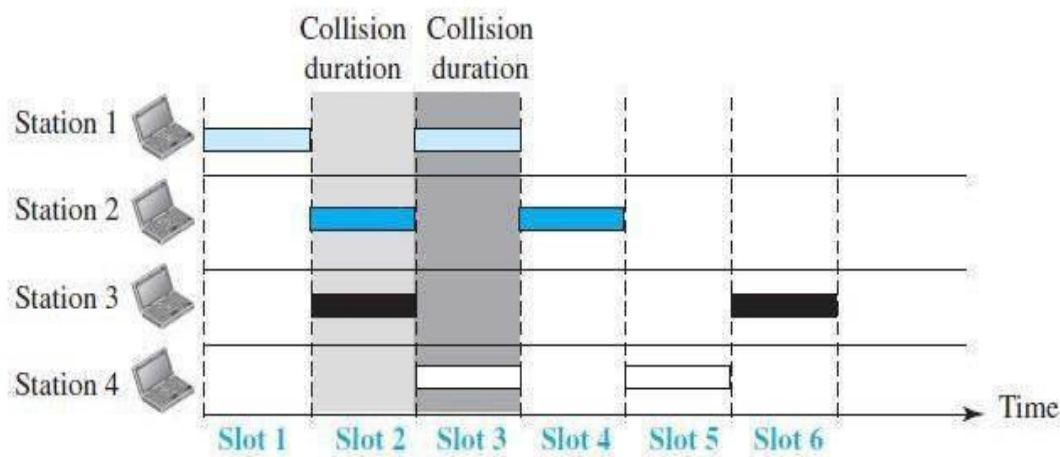


Figure 5: Frames in Slotted ALOHA network

- A station is allowed to send only at the beginning of the synchronized time slot, if a station misses this moment, it must wait until the beginning of the next time slot.
- This means that the station which started at the beginning of this slot has already finished sending its frame.
- There is still the possibility of collision if two stations try to send at the beginning of the same time slot. However, the vulnerable time is now reduced to one-half, equal to T_{fr} . Figure 6 shows the situation.

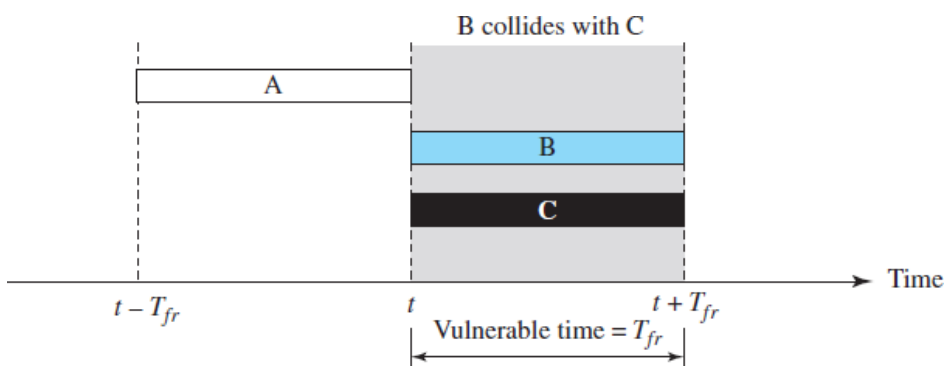


Figure 6: Vulnerable time for slotted ALOHA protocol

Slotted ALOHA vulnerable time = T_{fr}

Throughput

G = the average number of frames generated by the system during one frame transmission time (T_{fr})

S = the average number of successfully transmitted frames for Slotted ALOHA. And is given by,

$$S = G \times e^{-G}. \text{-----(1)}$$

Differentiate equation (1) with respect to G and equate it to 0, we get $G = 1$. Substitute $G=1$ in equation (1) to get S_{max} .

The maximum throughput $S_{max} = 0.368$.

- If one frame is generated during one frame transmission time then 36.8 percent of these frames reach their destination successfully.
- G is set to $G = 1$ to produce the maximum throughput because the vulnerable time is equal to the frame transmission time. Therefore, if a station generates only one frame in this vulnerable time (and no other stations generate a frame during this time), the frame will reach its destination successfully.

NOTE: The throughput for Slotted ALOHA is $S = G \times e^{-G}$.

The maximum throughput $S_{max} = 1/(e) = 0.368$ when $G = 1$.

PROBLEM 3

A Slotted ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces

- 1000 frames per second?
- 500 frames per second?
- 250 frames per second?

Solution: The frame transmission time T_{fr} is 200/200 kbps or 1 ms.

- If the system creates 1000 frames per second, or 1 frame per millisecond ($1s = 1000ms$) then $G = 1$ (because G = number of frames generated for one T_{fr}).
 $S = G \times e^{-G} = 0.368$ (36.8 percent). This means that the throughput is
 $1000 \times 0.368 = 368$ frames. Only 368 frames out of 1000 will probably survive.
- If the system creates 500 frames per second, or 1/2 frame per millisecond ($1s = 1000ms$) then $G = 1/2$ (because G = number of frames generated for one T_{fr}).
 $S = G \times e^{-G} = 0.303$ (30.3 percent). This means that the throughput is
 $500 \times 0.303 = 151$ frames. Only 151 frames out of 500 will probably survive.
- If the system creates 250 frames per second, or 1/4 frame per millisecond ($1s = 1000ms$) then $G = 1/4$ (because G = number of frames generated for one T_{fr}).
 $S = G \times e^{-G} = 0.195$ (19.5 percent). This means that the throughput is
 $250 \times 0.195 = 49$ frames. Only 49 frames out of 250 will probably survive.

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.

- 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 Figure 7, a space and time model of a CSMA network. Stations are connected to a shared channel.
- The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time (although very short) for the first bit to reach every station and for every station to sense it.
- A station may sense the medium and find it idle, only because the first bit sent by another station has not yet been received.

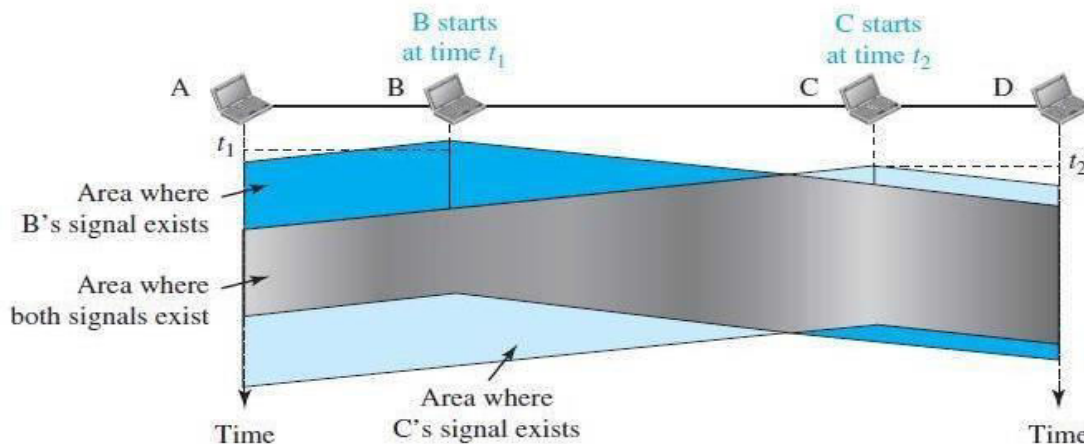


Figure 7: Space/time model of a collision in CSMA

- At time t_1 , station B senses the medium and finds it idle, so it sends a frame. At time t_2 ($t_2 > t_1$), 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.

Vulnerable Time

- The vulnerable time for CSMA is the **propagation time** T_p . 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.
- Figure 8 below shows the worst case. The leftmost station, A, sends a frame at time t_1 , which reaches the rightmost station, D, at time $t_1 + T_p$. The gray area shows the vulnerable area in time and space.

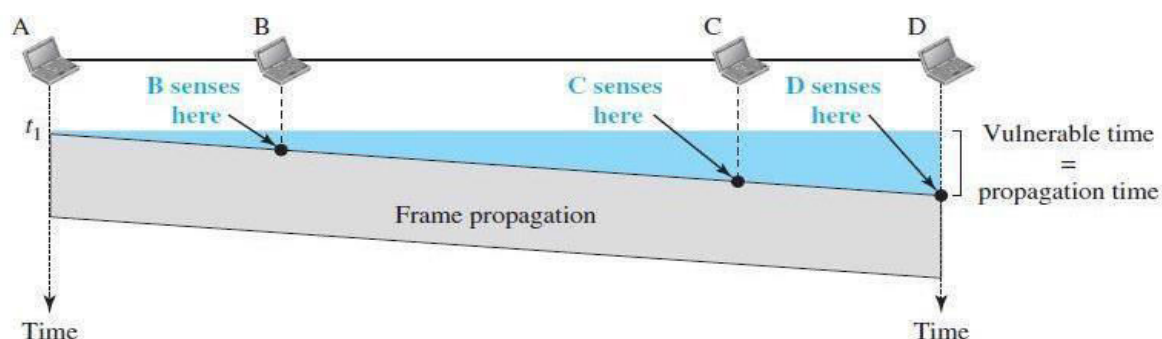


Figure 8: Vulnerable time in CSMA

Persistence Methods

Persistence method is developed to determine what the station has to do whenever it encounters the channel is idle or busy. There are 3 persistent methods

1. 1-persistent method
2. Non persistent method, and
3. p-Persistent method.

Figure 9 shows the behaviour of three persistence methods when a station finds a channel busy.

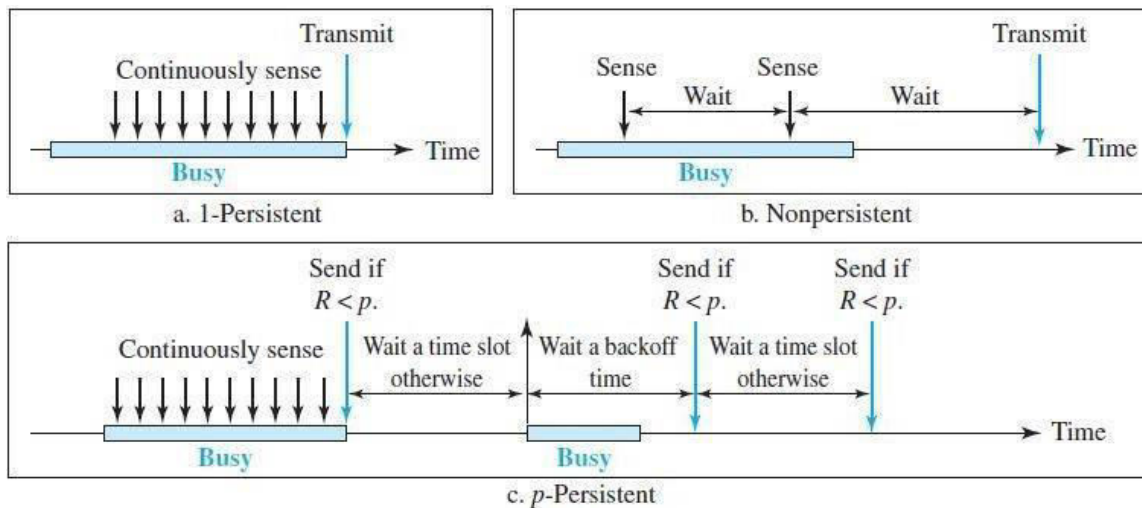


Figure 9: Behaviour of three persistence methods

1-Persistent

- The 1-persistent method is simple and straightforward.
- 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

- In the non persistent method, 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.
- The non persistent 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.
- This method reduces the efficiency of the network because the medium remains idle when there may be stations with frames to send.

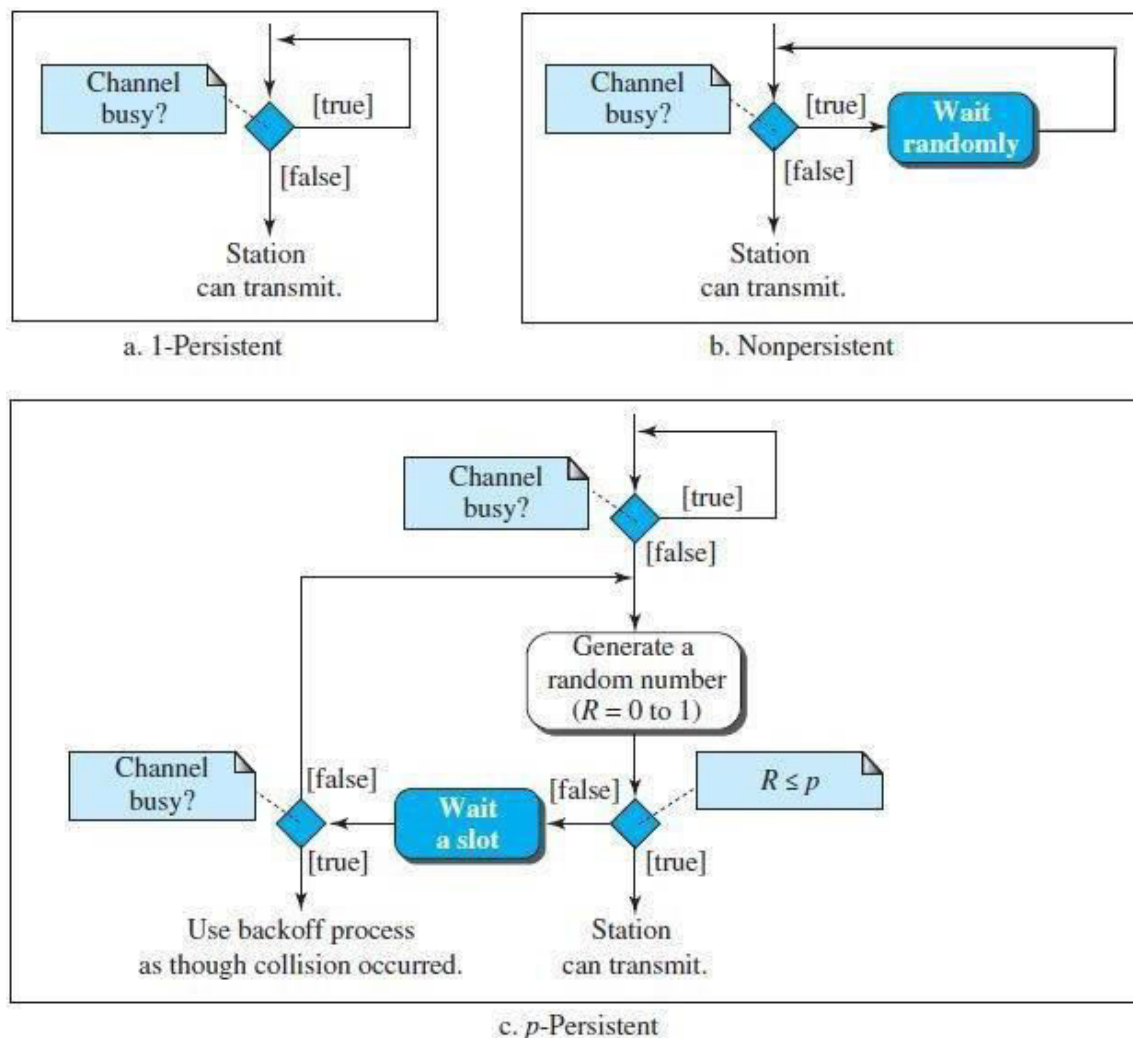


Figure 10: Flow diagram for three persistence methods

p-persistent

- The p-persistent method 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.

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.
- Station monitors the medium after it sends a frame to see if the transmission was successful.
- 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

In Figure 11, stations A and C are involved in the collision.

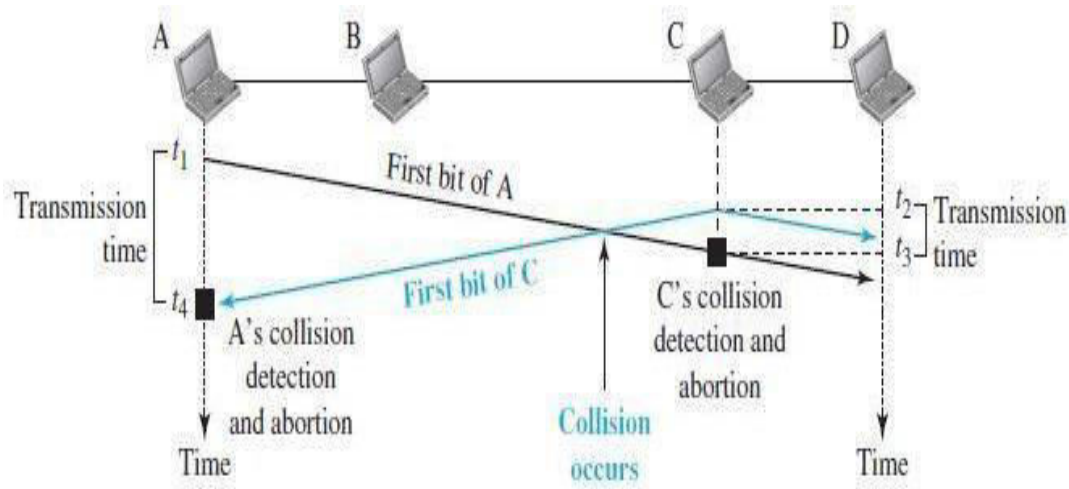


Figure 11: Collision of the first bits in CSMA/CD

- At time t_1 , station A has executed its persistence procedure and starts sending the bits of its frame. At time t_2 , 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 t_2 . Station C detects a collision at time t_3 when it receives the first bit of A's frame. Station C immediately aborts transmission.
- Station A detects collision at time t_4 when it receives the first bit of C's frame, it also immediately aborts transmission.

From the Figure 11, A transmits for the duration $t_4 - t_1$, C transmits for the duration $t_3 - t_2$.

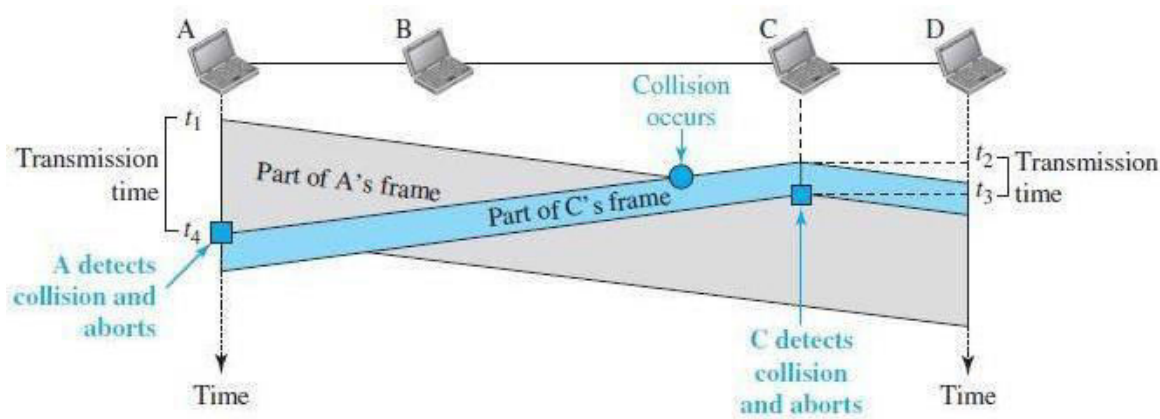


Figure 12: Collision and abortion in CSMA/CD

Minimum Frame Size

- Before sending the last bit of the frame, the sending station must detect a collision, if any, and abort the transmission.
- Once the entire frame is sent, station 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 T_p .
- If the two stations involved in a collision are the maximum distance apart, the signal from the first takes time T_p to reach the second, and the effect of the collision takes another time T_p to reach

the first. So the requirement is that the first station must still be transmitting after $2T_p$.

PROBLEM:

A network using CSMA/CD has a bandwidth of 10 Mbps. If the maximum propagation time (including the delays in the devices and ignoring the time needed to send a jamming signal) is $25.6 \mu s$, what is the minimum size of the frame?

Solution:

The minimum frame transmission time is $T_{fr} = 2 \times T_p = 51.2 \mu s$. This means, in the worst case, a station needs to transmit for a period of $51.2 \mu s$ to detect the collision.

The minimum size of the frame is, Band width $\times T_{fr} = 10 \text{ Mbps} \times 51.2 \mu s = 512 \text{ bits}$ or 64 bytes. This is actually the minimum size of the frame for Standard Ethernet.

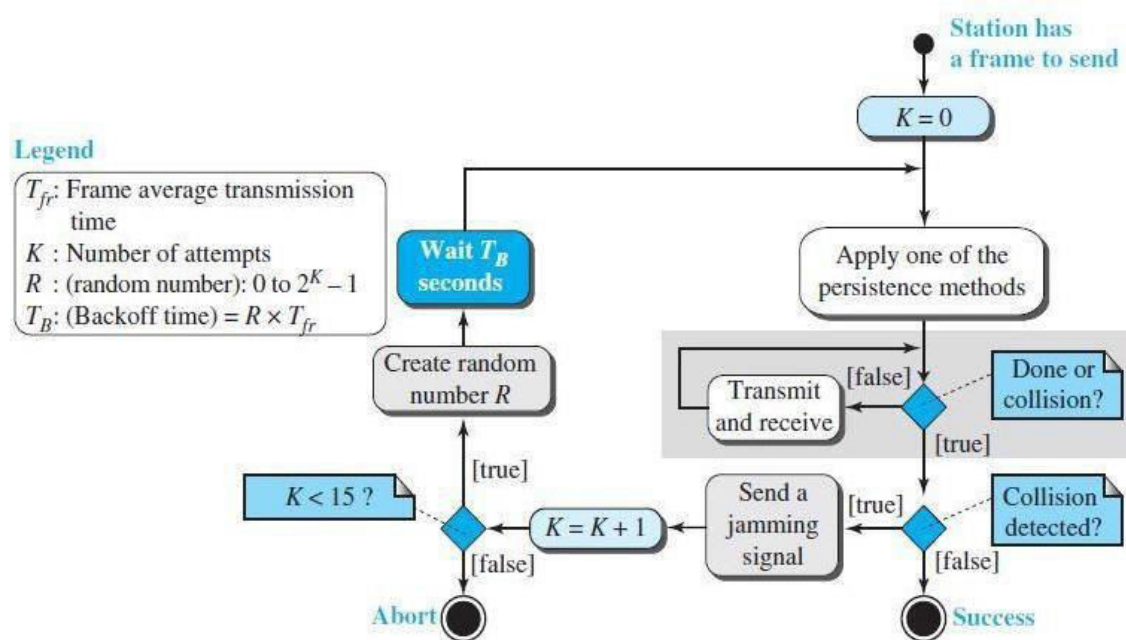


Figure 13: Flow diagram for the CSMA/CD

The flow diagram for CSMA/CD is as shown in Figure 13. It is similar to the one for the ALOHA protocol, but there are differences.

1. The first difference is the addition of the persistence process. It is required to sense the channel before sending the frame by using one of the persistence processes (non persistent, 1 persistent, or p-persistent).
2. The second difference is the frame transmission. In ALOHA, there is transmission of the entire frame and then wait for an acknowledgment. In CSMA/CD, transmission and collision detection are continuous processes.
 - It is not like the entire frame is sent and then look for a collision. The station transmits and receives continuously and simultaneously (using two different ports or a bidirectional port).
 - Loop is used to show that transmission is a continuous process. It is constantly monitored in order to detect one of two conditions: either transmission is finished or a collision is detected.
 - Either event stops transmission. When it comes out of the loop, if a collision has not been detected, it means that transmission is complete; the entire frame is transmitted. Otherwise, a collision has occurred.
3. The third difference is the sending of a short **jamming signal** to make sure that all other stations become aware of the collision.

Energy Level

The level of energy in a channel can have three values:

- 1) Zero level : The channel is idle
- 2) Normal level: A station has successfully captured the channel and is sending its frame.
- 3) Abnormal level: There is a collision and the level of the energy is twice the normal level.

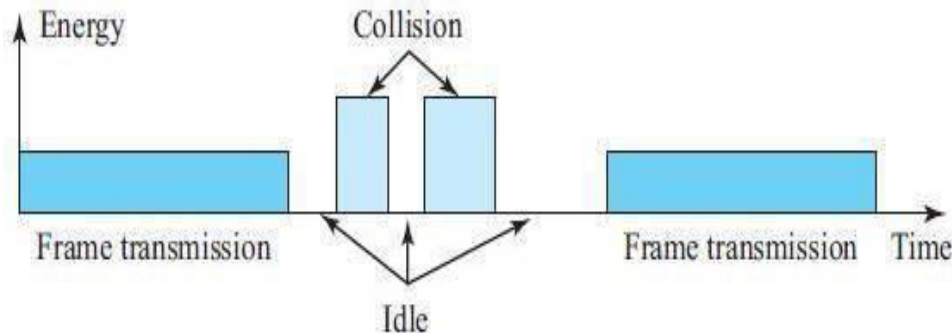


Figure 14: Energy level during transmission, idleness, or collision

NOTE: A station that has a frame to send or is sending a frame needs to monitor the energy level to determine if the channel is idle, busy, or in collision mode.

Throughput

- The throughput of CSMA/CD is greater than that of pure or slotted ALOHA.
- The maximum throughput occurs at a different value of G and is based on the persistence method and the value of p in the p -persistent approach.
- For the 1-persistent method, the maximum throughput is around 50 percent when $G = 1$. For the non persistent method, the maximum throughput can go up to 90 percent when G is between 3 and 8.

CSMA/CA

- **Carrier sense multiple access with collision avoidance (CSMA/CA)** was invented for wireless networks.
- Collisions are avoided through the use of CSMA/CA's three strategies: the inter frame space, the contention window, and acknowledgments.

Inter frame Space (IFS):

- 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.
- After waiting an IFS time, if the channel is still idle, the station can send, but it still needs to wait a time equal to the contention window.
- The IFS variable can also be used to prioritize stations or frame types. For example, a station

that is assigned a shorter IFS has a higher priority.

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.

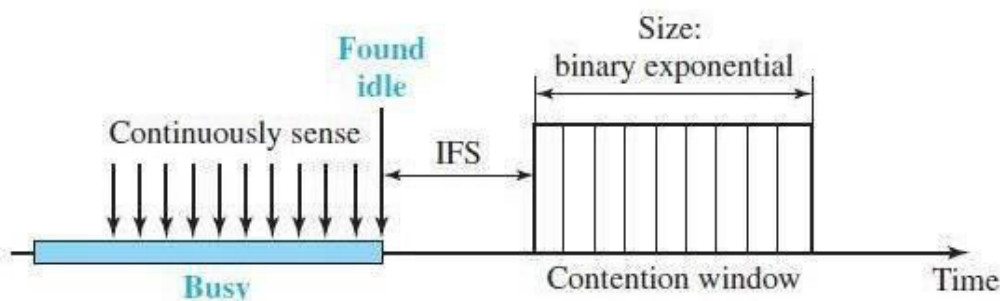


Figure 16: Contention window

Acknowledgement

Even with all the precautions considered, 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 that the receiver has received the frame.

Frame Exchange Time Line

Figure 17 shows the exchange of data and control frames in time.

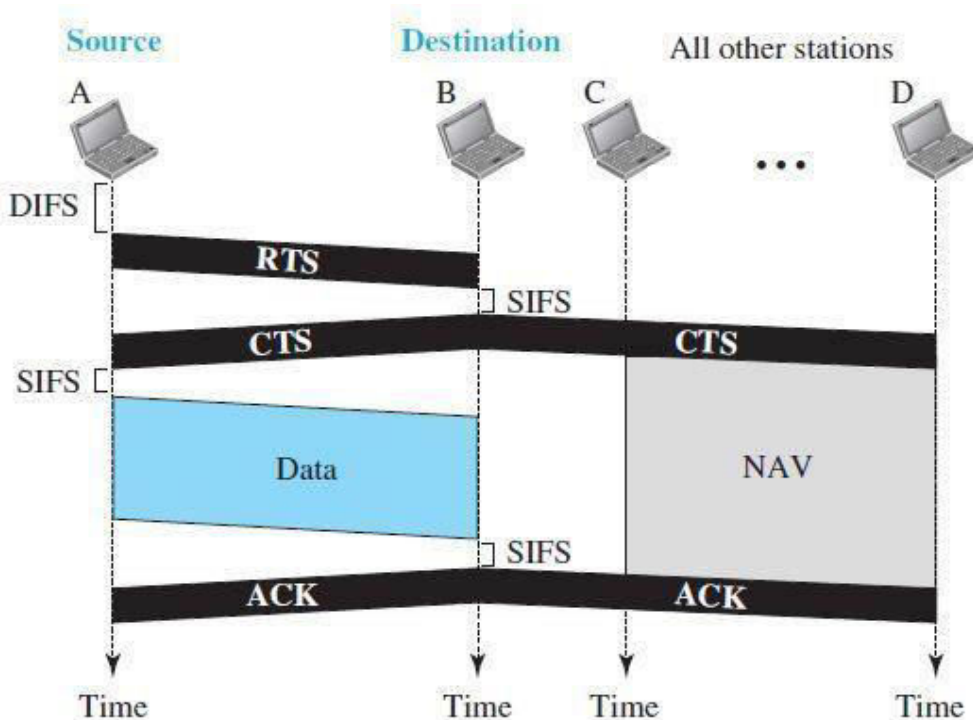


Figure 17: CSMA/CA and NAV

1. Before sending a frame, the source station senses the medium by checking the energy level at the carrier frequency.
 - a. The channel uses a persistence strategy with back off until the channel is idle.
 - b. After the station is found to be idle, the station waits for a period of time called the DCF inter frame space (DIFS), then the station sends a control frame called the request to send (RTS).
2. After receiving the RTS and waiting a period of time called the short inter frame space (SIFS), the destination station sends a control frame, called the clear to send (CTS), to the source station. This control frame indicates that the destination station is ready to receive data.
3. The source station sends data after waiting an amount of time equal to SIFS.
4. The destination station, after waiting an amount of time equal to SIFS, sends an acknowledgment to show that the frame has been received. Acknowledgment is needed in this protocol because the station does not have any means to check for the successful arrival of its data at the destination. On the other hand, the lack of collision in CSMA/CD is a kind of indication to the source that data have arrived.

NOTE: DIFS=DCF Inter frame space or Distributed Coordination Function Inter frame Space time.

Network Allocation Vector

- When a station sends an RTS frame, it includes the duration of time that it needs to occupy the channel.
- The stations that are affected by this transmission create a timer called a **Network Allocation Vector (NAV)** that shows how much time must pass before these stations are allowed to check the channel for idleness.
- Each time a station accesses the system and sends an RTS frame, other stations start their NAV. In other words, each station, before sensing the physical medium to see if it is idle, first checks its NAV to see if it has expired. Figure 17 shows the idea of NAV.

Collision during Handshaking

- If there is a collision during the time when RTS or CTS control frames are in transition, often called the handshaking period.
- Two or more stations may try to send RTS frames at the same time. These control frames may collide. However, because there is no mechanism for collision detection, the sender assumes there has been a collision if it has not received a CTS frame from the receiver. The back off strategy is employed, and the sender tries again.

Hidden-Station Problem

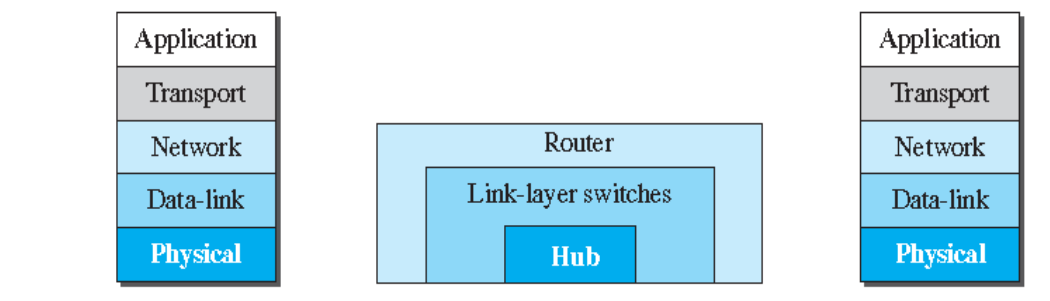
- The solution to the hidden station problem is the use of the handshake frames (RTS and CTS). Figure 17 shows that the RTS message from A reaches B, but not C.
- Both A and C are within the range of B, the CTS message, which contains the duration of data transmission from B to A, reaches C.
- Station C knows that some hidden station is using the channel and refrains from transmitting until that duration is over.

CONNECTING DEVICES

We use **connecting devices** to connect hosts together to make a network or to connect networks together to make an internet. Connecting devices can operate in different layers of the Internet model.

There are 3 kinds of *connecting devices*: 1. *Hubs*, 2. *Link-layer switches*, and 3. *Routers*.

Figure 17.1 Three categories of connecting devices



Hubs

A hub is a device that operates only in the physical layer. Signals that carry information within a network can travel a fixed distance before attenuation. A **repeater receives a signal and, before it becomes too weak or corrupted, regenerates and retimes the original bit pattern and then sends the refreshed signal.**

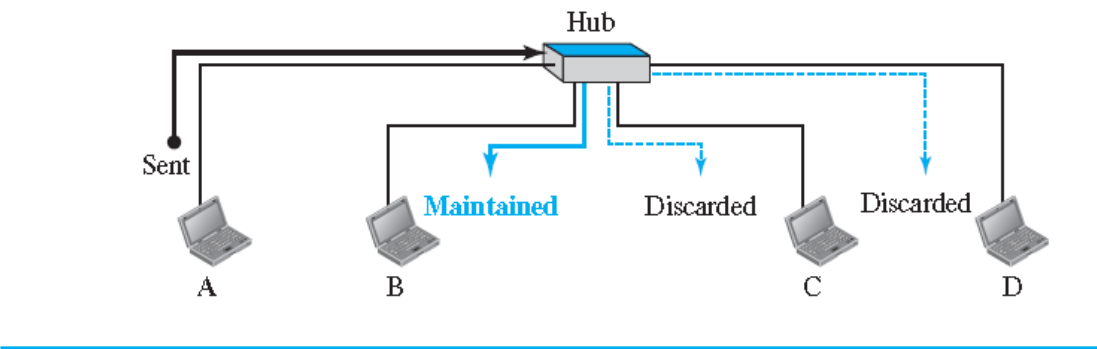
In the past, Ethernet LANs were using bus topology, a repeater was used to connect two segments of a LAN to overcome the length restriction of the coaxial cable. Today, Ethernet LANs use star topology. In a star topology, a repeater is a multiport device, often called a *hub*.

Hub can be used to serve as the connecting point and at the same time function as a repeater. When a packet from station A to station B arrives at the hub, the hub forwards the packet to all outgoing ports except the one from which the signal was received.

The figure definitely shows that a hub does not have a filtering capability;

It does not have the intelligence to find from which port the frame should be sent out. A hub or a repeater is a physical-layer device. They do not have a link-layer address and they do not check the link-layer address of the received frame. They just regenerate the corrupted bits and send them out from every port.

Figure 17.2 A hub



Link-Layer Switches

A link-layer switch (or *switch*) *operates in both the physical and the data-link layers.* As a physical-layer device, it regenerates the signal it receives. As a link-layer device, the link- layer

switch can check the MAC addresses (source and destination) contained in the frame.

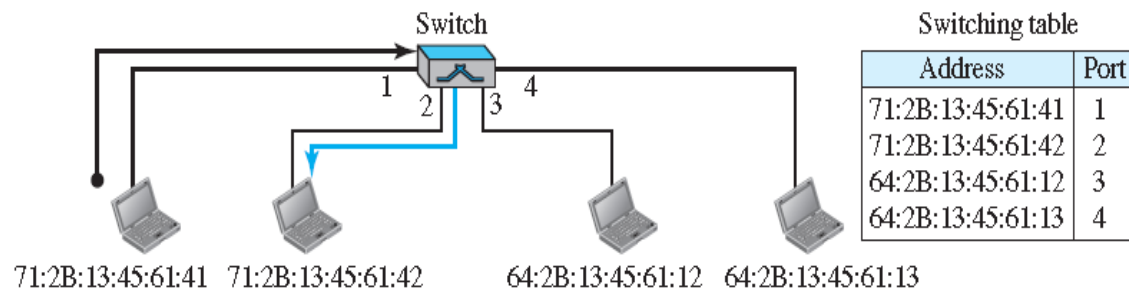
What is the difference in functionality is between a link-layer switch and a hub.?

Answer: A link-layer switch has **filtering capability**.

A Link layer switch can check the destination address of a frame and can decide from which outgoing port the frame should be sent.

Filtering

Figure 17.3 Link-layer switch



- If a frame destined for station 71:2B:13:45:61:42 arrives at port 1,
- The link-layer switch consults its table to find the departing port.
- According to its table, frames for 71:2B:13:45:61:42 should be sent out only through port 2;
- Therefore, there is no need for forwarding the frame through other **ports**.

Transparent Switches

A transparent switch is a switch in which the stations are completely unaware of the switch's existence. If a switch is added or deleted from the system, reconfiguration of the stations is unnecessary.

According to the IEEE 802.1d specification, a system equipped with transparent switches must meet three criteria:

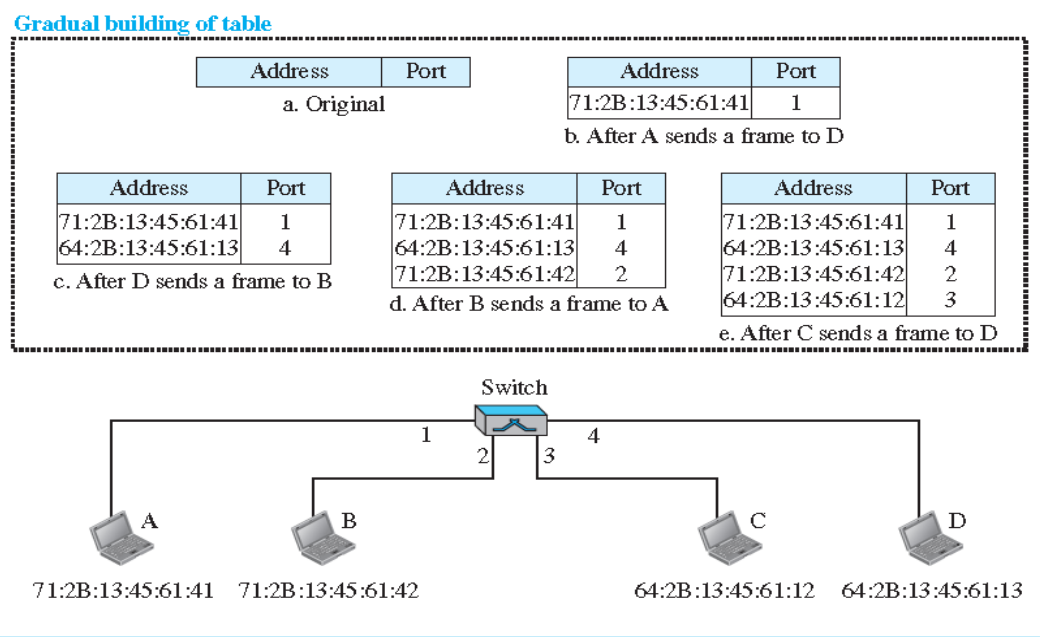
- ☐ Frames must be forwarded from one station to another.
- ☐ The forwarding table is automatically made by learning frame movements in the network.
- ☐ Loops in the system must be prevented.

Learning

The earliest switches had switching tables that were static. The system administrator would manually enter each table entry during switch setup. Although the process was simple, it was not practical. If a station was added or deleted, the table had to be modified manually. The same was true if a station's MAC address changed, which is not a rare event. For example, putting in a new network card means a new MAC address.

A better solution to the static table is a dynamic table that maps addresses to ports (interfaces) automatically. To make a table dynamic, we need a switch that gradually learns from the frames movements. To do this, the switch inspects both the destination and the source addresses in each frame that passes through the switch. The destination address is used for the forwarding decision (table lookup); the source address is used for adding entries to the table and for updating purposes.

Figure 17.4 Learning switch

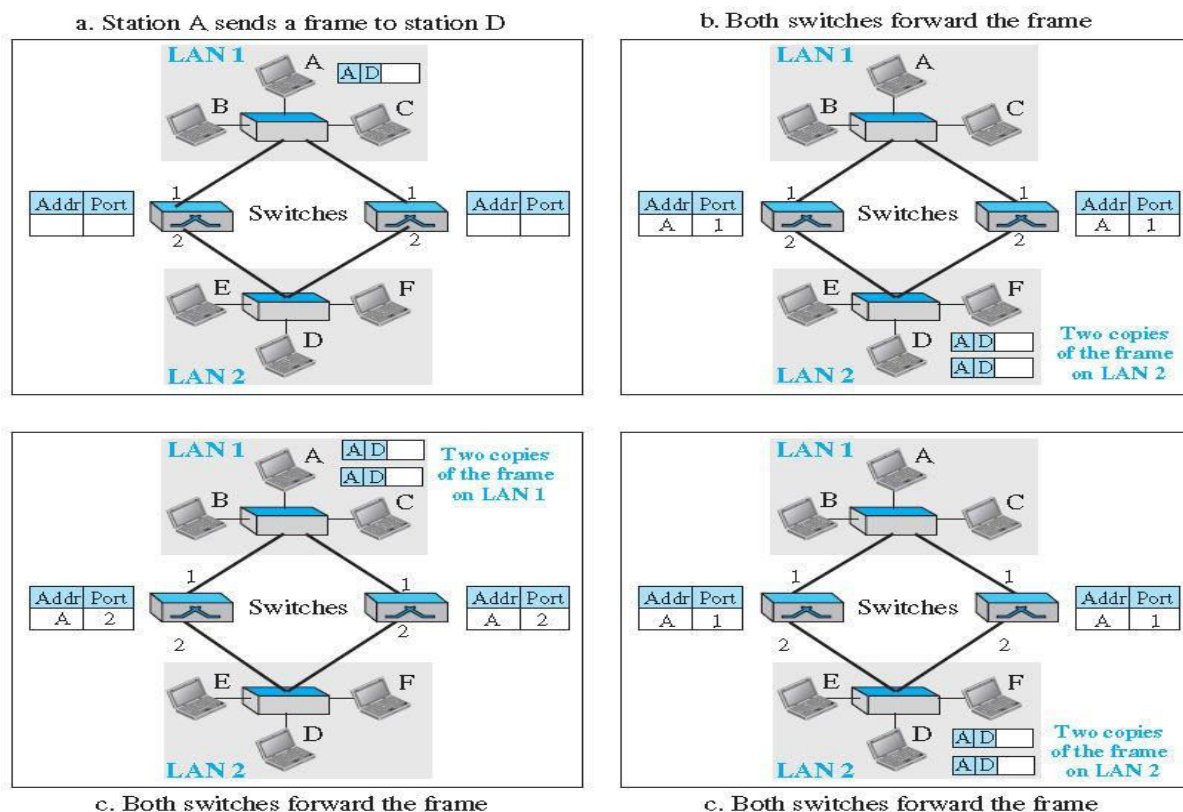


- When station A sends a frame to station D, the frame goes out from all three ports; the frame floods the network.
- However, by looking at the source address, the switch learns that station A must be connected to port 1.
- This means that frames destined for A, in the future, must be sent out through port 1.
- The switch adds this entry to its table.
- When station D sends a frame to station B, the switch has no entry for B, it adds one more entry to the table related to station D.

The learning process continues until the table has information about every port. However, the learning process may take a long time. For example, if a station does not send out a frame (a rare situation), the station will never have an entry in the table

Loop Problem

Transparent switches work fine as long as there are no redundant switches in the system. Systems administrators, however, like to have redundant switches (more than one switch between a pair of LANs) to make the system more reliable. Redundancy can create loops in the system, which is very undesirable. Loops can be created only when two or more broadcasting LANs (those using hubs, for example) are connected by more than one switch.

Figure 17.5 Loop problem in a learning switch

1. Station A sends a frame to station D. The tables of both switches are empty. Both forward the frame and update their tables based on the source address A.

2. Now there are two copies of the frame on LAN 2. The copy sent out by the left switch is received by the right switch, which does not have any information about the destination address D; it forwards the frame.

The copy sent out by the right switch is received by the left switch and is sent out for lack of information about D.

Note:- Each frame is handled separately because switches, as two nodes on a broadcast network sharing the medium, use an access method such as CSMA/CD.

3. Now there are two copies of the frame on LAN 1. Step 2 is repeated, and both copies are sent to LAN2.

4. The process continues on and on. This is called Loop problem

Spanning Tree Algorithm

To solve the looping problem, the IEEE specification requires that switches use the spanning tree algorithm to create a loopless topology. In graph theory, a **spanning tree is a graph in which there is no loop.**

In a switched LAN, this means creating a topology in which each LAN can be reached from any other LAN through one path only (no loop).

To find the spanning tree, we need to assign a cost (metric) to each arc. The interpretation of the cost is left up to the systems administrator. We have chosen the minimum hops. However, the hop count is normally 1 from a switch to the LAN and 0 in the reverse direction.

Steps to find Spanning tree

1. Every switch has a built-in ID (normally the serial number, which is unique). Each switch broadcasts this ID so that all switches know which one has the smallest ID. The switch with the

smallest ID is selected as the root switch (root of the tree). We assume that switch S1 has the smallest ID. It is, therefore, selected as the root switch.

2. The algorithm tries to find the shortest path (a path with the shortest cost) from the root switch to every other switch or LAN. The shortest path can be found by examining the total cost from the root switch to the destination.
3. The combination of the shortest paths creates the shortest tree, which is also shown in Figure 17.7.
4. Based on the spanning tree, we mark the ports that are part of it, the **forwarding ports**, which forward a frame that the switch receives. We also mark those ports that are not part of the spanning tree, the **blocking ports**, which block the frames received by the switch

Figure 17.6 A system of connected LANs and its graph representation

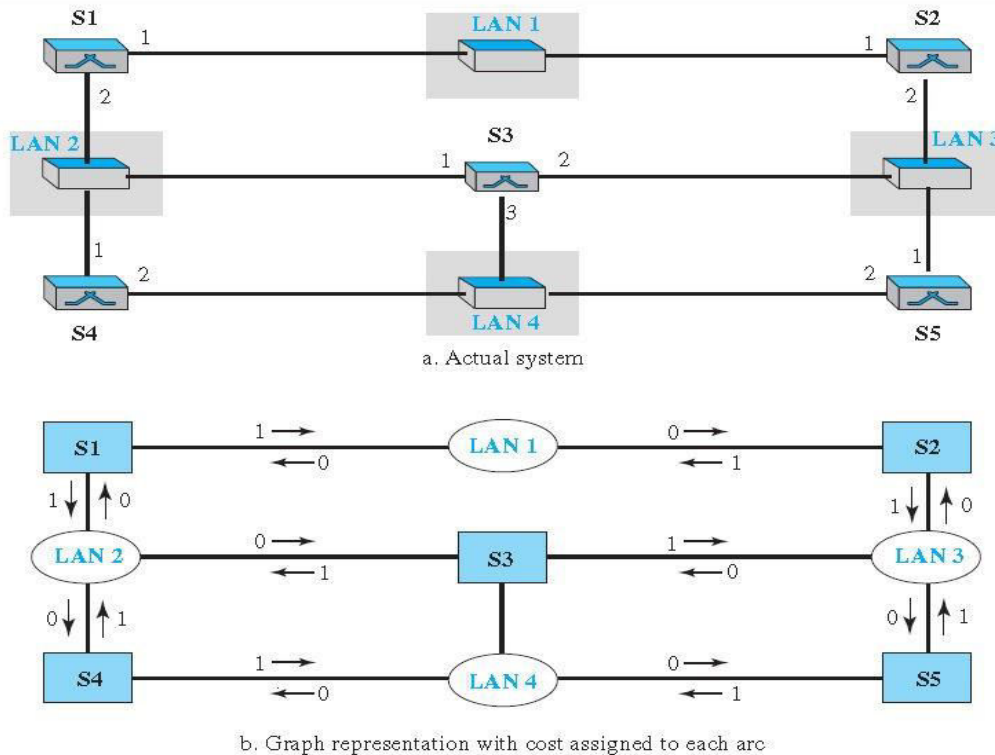


Figure 17.7 Finding the shortest paths and the spanning tree in a system of switches

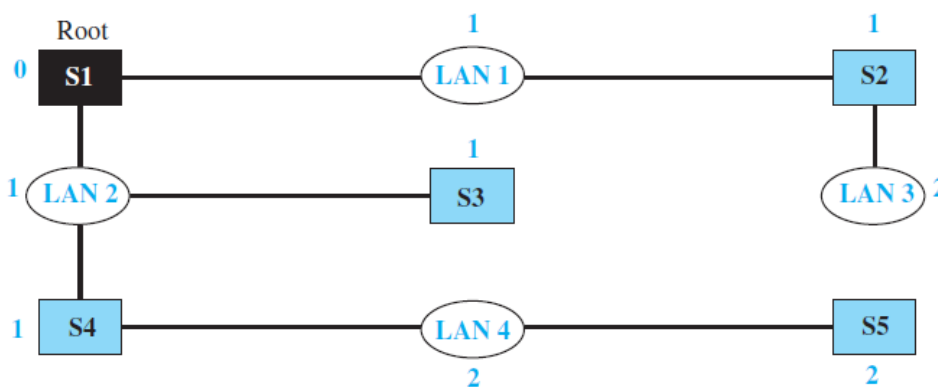
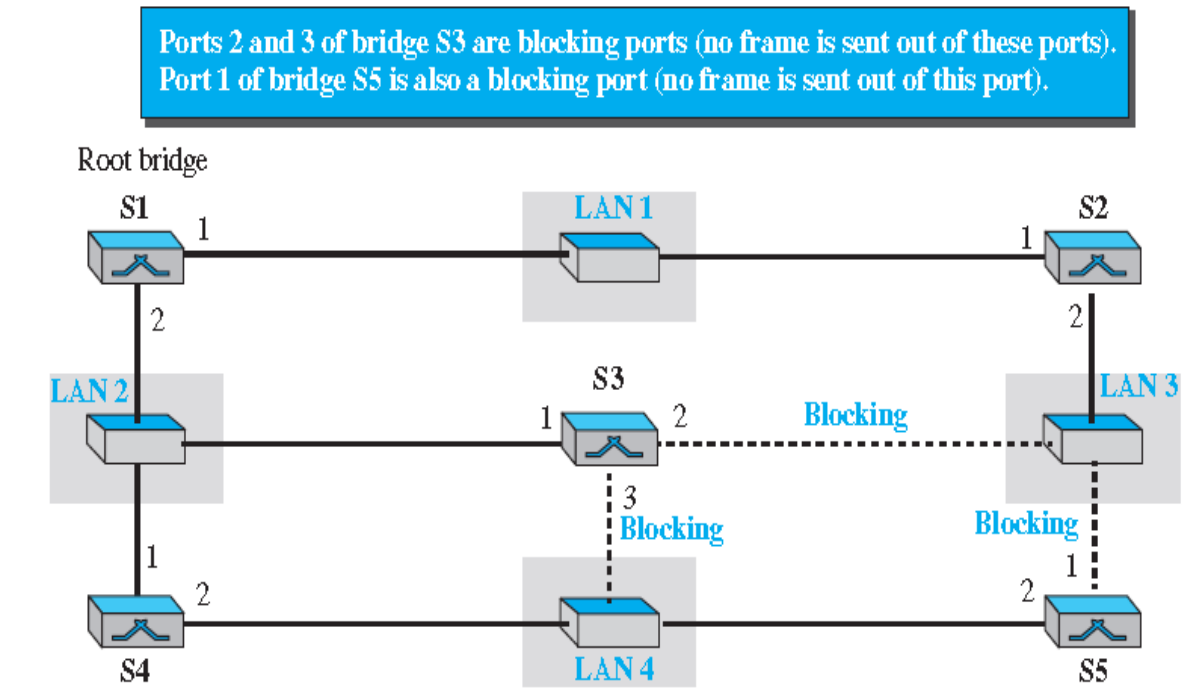


Figure 17.8 Forwarding and blocking ports after using spanning tree algorithm

Note that there is only one path from any LAN to any other LAN in the spanning tree system. This means there is only one path from one LAN to any other LAN. No loops are created. We have described the spanning tree algorithm as though it required manual entries. This is not true. Each switch is equipped with a software package that carries out this process dynamically.

Advantages of Switches

1. Collision Elimination

A link-layer switch eliminates the collision. This means increasing the average bandwidth available to a host in the network. In a switched LAN, there is no need for carrier sensing and collision detection; each host can transmit at any time.

2. Connecting Heterogeneous Devices

A link-layer switch can connect devices that use different protocols at the physical layer (data rates) and different transmission media.

As long as the format of the frame at the data-link layer does not change, a switch can receive a frame from a device that uses twisted-pair cable and sends data at 10 Mbps and deliver the frame to another device that uses fiber-optic cable and can receive data at 100 Mbps.

Routers

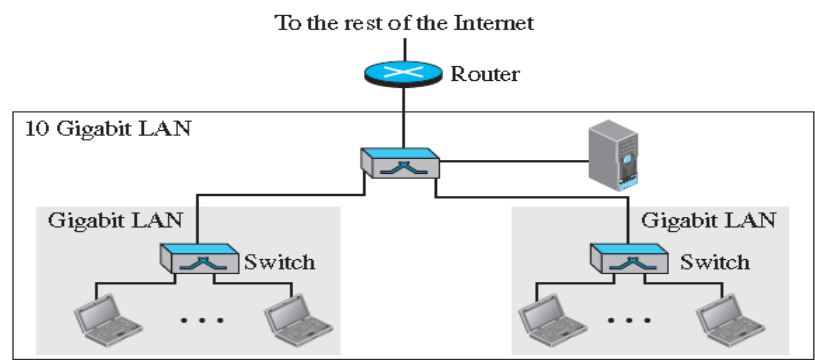
A router is a three-layer device; it operates in the physical, data-link, and network layers. As a physical-layer device, it regenerates the signal it receives. As a link-layer device, the router checks the physical addresses (source and destination) contained in the packet. As a network-layer device, a router checks the network-layer addresses. A router can connect networks. In other words, a router is an internetworking device; It connects independent networks to form an internetwork.

Differences between a router and a switch

1. A router has a physical and logical (IP) address for each of its interfaces.
2. A router acts only on those packets in which the link-layer destination address matches the address of the interface at which the packet arrives.

3. A router changes the link-layer address of the packet (both source and destination) when it forwards the packet.

Figure 17.9 Routing example



A router, will change the MAC addresses it receives because the MAC addresses have only local jurisdictions.

VIRTUAL LANs

A station is considered part of a LAN if it physically belongs to that LAN. The criterion of membership is geographic.

What happens if we need a virtual connection between two stations belonging to two different physical LANs? We can roughly define a virtual local area network (VLAN) as a local area network configured by software, not by physical wiring.

Figure 17.10 A switch connecting three LANs

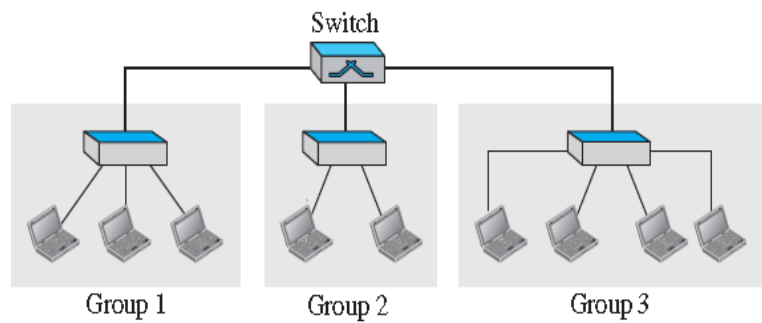
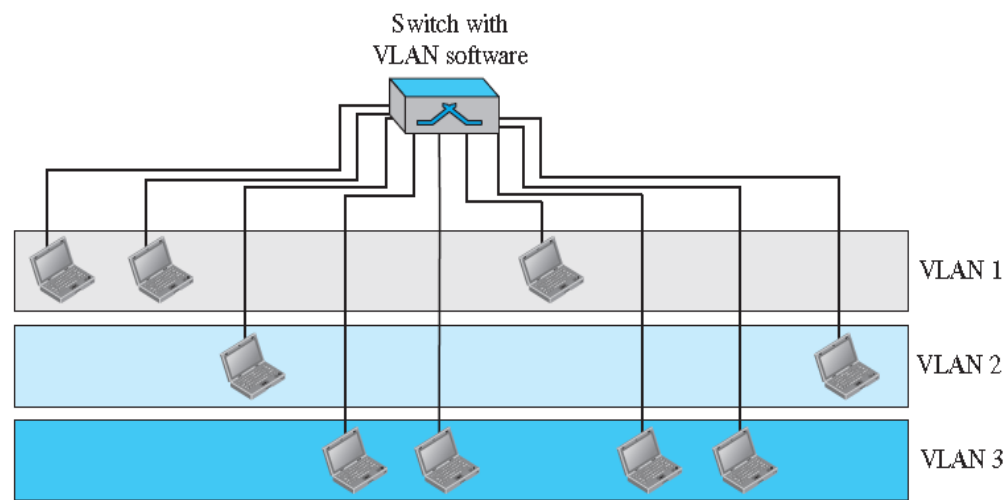


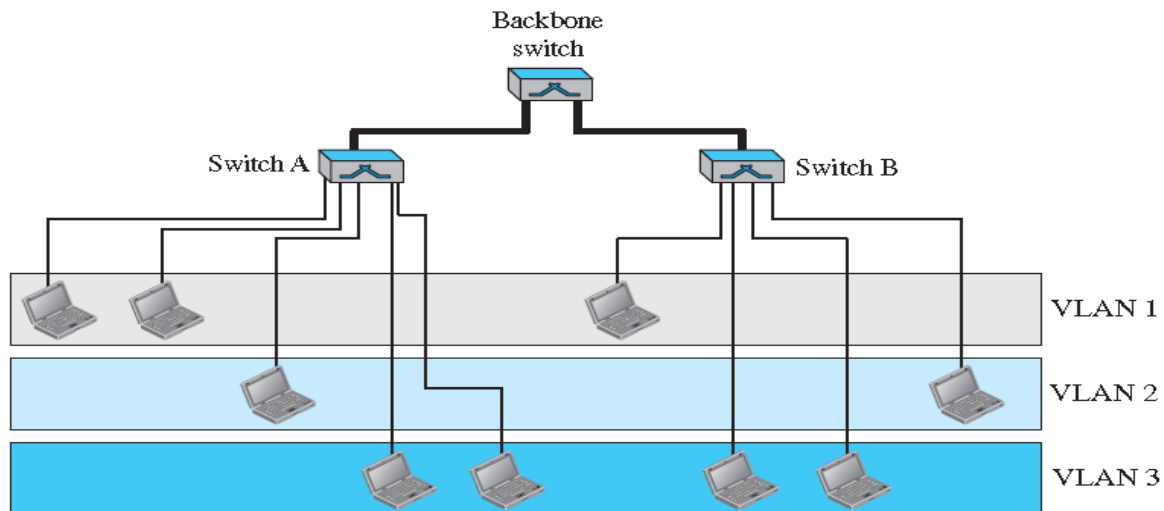
Figure 17.11 A switch using VLAN software



The whole idea of VLAN technology is to divide a LAN into logical, instead of physical, segments. A LAN can be divided into several logical LANs, called *VLANs*. Each *VLAN* is a work group in the organization. If a person moves from one group to another, there is no need to change the physical configuration. Any station can be logically moved to another VLAN. All members belonging to a VLAN can receive broadcast messages sent to that particular VLAN.

This means that if a station moves from VLAN 1 to VLAN 2, it receives broadcast messages sent to VLAN 2, but no longer receives broadcast messages sent to VLAN 1.

Figure 17.12 Two switches in a backbone using VLAN software



VLAN technology even allows the grouping of stations connected to different switches in a VLAN. Stations from switches A and B belong to each VLAN. This is a good configuration for a company with two separate buildings. Each building can have its own switched LAN connected by a backbone. People in the first building and people in the second building can be in the same work group even though they are connected to different physical LANs.

Membership

What characteristic can be used to group stations in a VLAN?

Vendors use different characteristics such as interface numbers, port numbers, MAC addresses, IP addresses, IP multicast addresses, or a combination of two or more of these.

Interface Numbers

Some VLAN vendors use switch interface numbers as a membership characteristic. For example, the administrator can define that stations connecting to ports 1, 2, 3, and 7 belong to VLAN 1, stations connecting to ports 4, 10, and 12 belong to VLAN 2, and so on.

MAC Addresses

Some VLAN vendors use the 48-bit MAC address as a membership characteristic. For example, the administrator can stipulate that stations having MAC addresses E2:13:42:A1:23:34 and F2:A1:23:BC:D3:41 belong to VLAN 1.

IP Addresses

Some VLAN vendors use the 32-bit IP address as a membership characteristic. For example, the administrator can stipulate that stations having IP addresses 181.34.23.67, 181.34.23.72, 181.34.23.98, and 181.34.23.112 belong to VLAN 1.

Multicast IP Addresses

Some VLAN vendors use the multicast IP address as a membership characteristic.

Combination

Recently, the software available from some vendors allows all these characteristics to be combined. The administrator can choose one or more characteristics when installing the software.

Configuration

How are the stations grouped into different VLANs? Stations are configured in one of three ways:

1. **Manual Configuration:** In a manual configuration, the network administrator uses the VLAN software to manually assign the stations into different VLANs at setup. Later migration from one VLAN to another is also done manually. Note that this is not a physical configuration; it is a logical configuration. The term *manually here means that the administrator types* the port numbers, the IP addresses, or other characteristics, using the VLAN software.
2. **Automatic Configuration:** In an automatic configuration, the stations are automatically connected or disconnected from a VLAN using criteria defined by the administrator.
For example, the administrator can define the project number as the criterion for being a member of a group.
When a user changes projects, he or she automatically migrates to a new VLAN.
3. **Semiautomatic Configuration:** *A semiautomatic configuration is somewhere between a manual configuration and an automatic configuration. Usually, the initializing is done manually, with migrations done automatically.*

Communication between Switches

In a multi-switched backbone, each switch must know not only which station belongs to which VLAN, but also the membership status of stations connected to other switches. For example, in Figure 17.12, switch A must know the membership status of stations connected to switch B, and switch B must know the same about switch A.

Three methods have been devised for this purpose: they are

1. Table Maintenance

In this method, when a station sends a broadcast frame to its group members, the switch creates an a table and records station membership. The switches send their tables to one another periodically for updating.

2. Frame Tagging

In this method, when a frame is traveling between switches, an extra header is added to the MAC frame to define the destination VLAN. The frame tag is used by the receiving switches to determine the VLANs to be receiving the broadcast message.

IEEE Standard

In 1996, the IEEE 802.1 subcommittee passed a standard called 802.1Q that defines the format for frame tagging. The standard also defines the format to be used in multiswitched backbones and enables the use of multivendor equipment in VLANs

3. Time-Division Multiplexing (TDM)

In this method, the connection (trunk) between switches is divided into time-shared channels. For example, if the total number of VLANs in a backbone is five, each trunk is divided into five channels. The traffic destined for VLAN 1 travels in channel 1, the traffic destined for VLAN 2 travels in channel 2, and so on. The receiving switch determines the destination VLAN by checking the channel from which the frame arrived.

Advantages

1. **Cost and Time Reduction**

VLANs can reduce the migration cost of stations going from one group to another. Physical reconfiguration takes time and is costly. Instead of physically moving one station to another segment or even to another switch, it is much easier and quicker to move it by using software.

2. **Creating Virtual Work Groups**

VLANs can be used to create virtual work groups. For example, in a campus environment, professors working on the same project can send broadcast messages to one another without the necessity of belonging to the same department. This can reduce traffic if the multicasting capability of IP was previously used.

3. **Security**

VLANs provide an extra measure of security. People belonging to the same group can send broadcast messages with the guaranteed assurance that users in other groups will not receive these messages.

Wired LANs: Ethernet PROTOCOL

A local area network (LAN) is a computer network that is designed for a limited geographic area such as a building or a campus. Although a LAN can be used as an isolated network to connect computers in an organization for the sole purpose of sharing resources, most LANs today are also linked to a wide area network (WAN) or the Internet. Almost every LAN except Ethernet has disappeared from the marketplace because Ethernet was able to update itself to meet the needs of the time

IEEE Project 802

- In 1985, the Computer Society of the IEEE started a project, called **Project 802**, to set standards to enable intercommunication among equipment from a variety of manufacturers.
- Project 802 does not seek to replace any part of the OSI model or TCP/IP protocol suite. Instead, it is a way of specifying functions of the physical layer and the data-link layer of major LAN protocols.
- The relationship of the 802 Standard to the TCP/IP protocol suite is shown in Figure 13.1. The IEEE has subdivided the data-link layer into two sub layers:
 - Logical link control (LLC)
 - Media access control (MAC)

IEEE has also created several physical-layer standards for different LAN protocols.

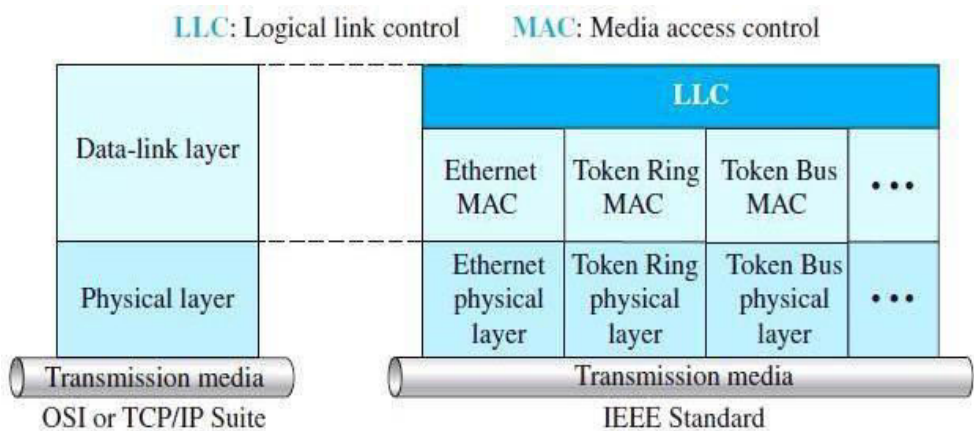


Figure 1: IEEE standard for LANs

Logical Link Control (LLC)

- In IEEE Project 802, flow control, error control, and part of the framing duties are collected into one sub layer called the logical link control (LLC). Framing is handled in both the LLC sublayer and the MAC sublayer.
- The LLC provides a single link-layer control protocol for all IEEE LANs. This means LLC protocol can provide interconnectivity between different LANs because it makes the MAC sub layer transparent.

Media Access Control (MAC)

- IEEE Project 802 has created a sublayer called media access control that defines the specific access method for each LAN. For example, it defines CSMA/CD as the media access method for Ethernet LANs and defines the token-passing method for Token Ring and Token Bus LANs.
- Part of the framing function is also handled by the MAC layer.

Ethernet Evolution

The Ethernet LAN was developed in the 1970s by Robert Metcalfe and David Boggs. The four generations of Ethernet are :

1. Standard Ethernet (10 Mbps)
2. Fast Ethernet (100 Mbps)
3. Gigabit Ethernet (1 Gbps) and
4. 10 Gigabit Ethernet (10 Gbps)

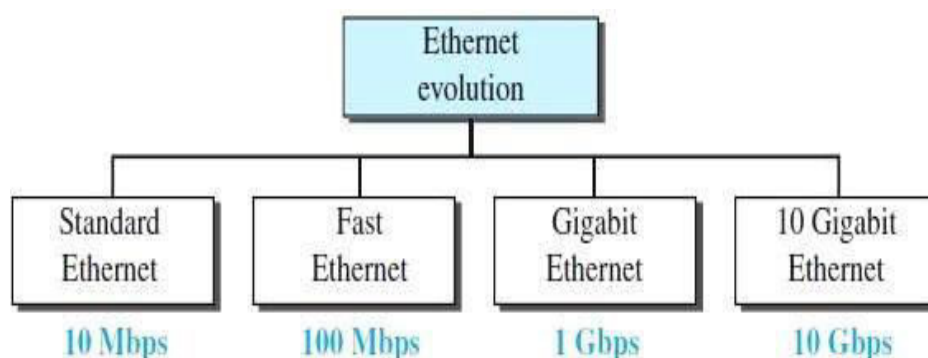


Figure 2: Ethernet evolution through four generations

STANDARD ETHERNET

Characteristics

1. Connectionless and Unreliable Service

- Ethernet provides a connectionless service, which means each frame sent is independent of the previous or next frame. Ethernet has no connection establishment or connection termination phases.
- The sender sends a frame whenever it has, the receiver may or may not be ready for it. The sender may overwhelm the receiver with frames, which may result in dropping frames. If a frame drops, the sender will not know about it. Since IP, which is using the service of Ethernet, is also connectionless, it will not know about it either. If the transport layer is also a connectionless protocol, such as UDP, the frame is lost and salvation may only come from the application layer. However, if the transport layer is TCP, the sender TCP does not receive acknowledgment for its segment and sends it again.

- Ethernet is also unreliable like IP and UDP. If a frame is corrupted during transmission and the receiver finds out about the corruption, which has a high level of probability of happening because of the CRC-32, the receiver drops the frame silently. It is the duty of high-level protocols to find out about it.

2. Frame Format

The Ethernet frame contains seven fields, as shown in Figure 3

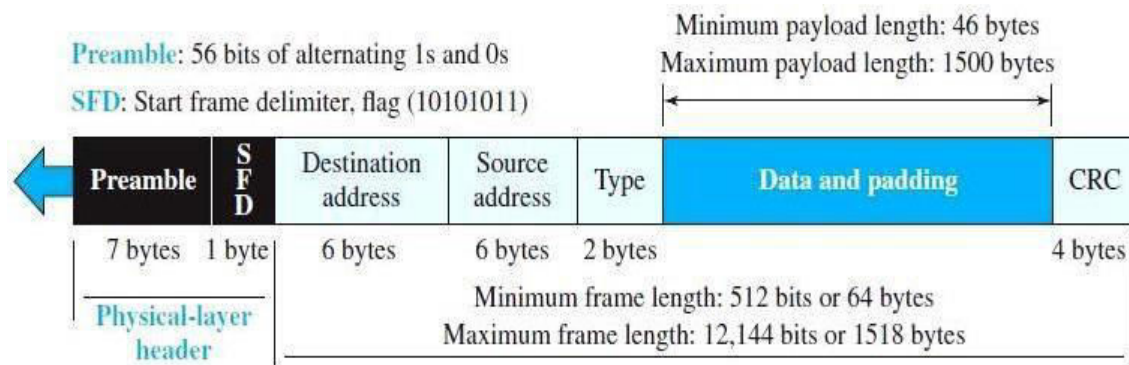


Figure 3: Ethernet frame

- **Preamble.** This field contains 7 bytes (56 bits) of alternating 0s and 1s that alert the receiving system to the coming frame and enable it to synchronize its clock if it's out of synchronization. The pattern provides only an alert and a timing pulse. The 56-bit pattern allows the stations to miss some bits at the beginning of the frame. The preamble is actually added at the physical layer and is not part of the frame.
- **Start frame delimiter (SFD).** This field (1 byte: 10101011) signals the beginning of the frame. The SFD warns the station or stations that this is the last chance for synchronization. The last 2 bits are $(11)_2$ and alert the receiver that the next field is the destination address. This field is actually a flag that defines the beginning of the frame, an Ethernet frame is a variable-length frame. It needs a flag to define the beginning of the frame. The SFD field is also added at the physical layer.
- **Destination address (DA).** This field is six bytes (48 bits) and contains the link layer address of the destination station or stations to receive the packet. When the receiver sees its own link-layer address, or a multicast address for a group that the receiver is a member of, or a broadcast address, it decapsulates the data from the frame and passes the data to the upper layer protocol defined by the value of the type field.
- **Source address (SA).** This field is also six bytes and contains the link-layer address of the sender of the packet.
- **Type.** This field defines the upper-layer protocol whose packet is encapsulated in the frame. This protocol can be IP, ARP, OSPF, and so on. In other words, it serves the same purpose as the protocol field in a datagram and the port number in a segment or user datagram. It is used for

multiplexing and demultiplexing.

- **Data.** This field carries data encapsulated from the upper-layer protocols. It is a minimum of 46 and a maximum of 1500 bytes. If the data coming from the upper layer is more than 1500 bytes, it should be fragmented and encapsulated in more than one frame. If it is less than 46 bytes, it needs to be padded with extra 0s. A padded data frame is delivered to the upper-layer protocol as it is (without removing the padding), which means that it is the responsibility of the upper layer to remove or, in the case of the sender, to add the padding. The upper-layer protocol needs to know the length of its data. For example, a datagram has a field that defines the length of the data.
- **CRC.** The last field contains error detection information, in this case a CRC-32. The CRC is calculated over the addresses, types, and data field. If the receiver calculates the CRC and finds that it is not zero (corruption in transmission), it discards the frame.

3. Frame Length

- Ethernet has imposed restrictions on both the minimum and maximum lengths of a frame. The minimum length restriction is required for the correct operation of CSMA/CD.
- An Ethernet frame needs to have a minimum length of 512 bits or 64 bytes. Part of this length is the header and the trailer. If we count 18 bytes of header and trailer (6 bytes of source address, 6 bytes of destination address, 2 bytes of length or type, and 4 bytes of CRC), then the minimum length of data from the upper layer is $64 - 18 = 46$ bytes. If the upper-layer packet is less than 46 bytes, padding is added to make up the difference.
- The standard defines the maximum length of a frame (without preamble and SFD field) as 1518 bytes. If we subtract the 18 bytes of header and trailer, the maximum length of the payload is 1500 bytes.
- The maximum length restriction has two historical reasons. First, memory was very expensive when Ethernet was designed; a maximum length restriction helped to reduce the size of the buffer. Second, the maximum length restriction prevents one station from monopolizing the shared medium, blocking other stations that have data to send.

NOTE:

Minimum frame length: 64 bytes
length: 1518 bytes

Minimum data length: 46 bytes
Maximum data length: 1500 bytes

Addressing

Each station on an Ethernet network (such as a PC, workstation, or printer) has its own network interface card (NIC). The NIC fits inside the station and provides the station with a link-layer address. The Ethernet address is 6 bytes (48 bits), normally written in hexadecimal notation, with a colon between the bytes. For example, the following shows an Ethernet MAC address:

4A:30:10:21:10:1A

Transmission of Address Bits

The way the addresses are sent out online is different from the way they are written in hexadecimal notation. The transmission is left to right, byte by byte; however, for each byte, the least significant bit is sent first and the most significant bit is sent last. This means that the bit that

defines an address as unicast or multicast arrives first at the receiver. This helps the receiver to immediately know if the packet is unicast or multicast.

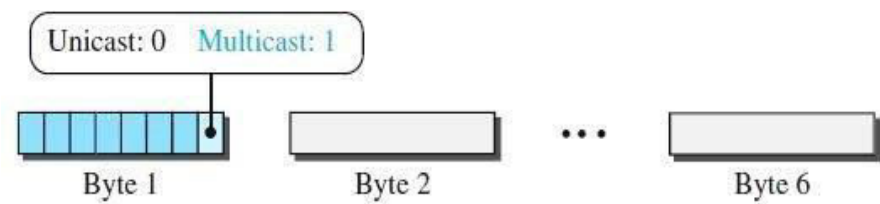
Example

Show how the address 47:20:1B:2E:08:EE is sent out online.

Solution: The address is sent left to right, byte by byte; for each byte, it is sent right to left, bit by bit, as shown below

Hexadecimal	47	20	1B	2E	08	EE
Binary	01000111	00100000	00011011	00101110	00001000	11101110
Transmitted ←	11100010	00000100	11011000	01110100	00010000	01110111

Unicast, Multicast, and Broadcast Addresses



A source address is always a unicast address, the frame comes from only one station. The destination address, however, can be unicast, multicast, or broadcast. Figure 4 shows how to distinguish a unicast address from a multicast address. If the least significant bit of the first byte in a destination address is 0, the address is unicast; otherwise, it is multicast. With the way the bits are transmitted, the unicast/multicast bit is the first bit which is transmitted or received. The broadcast address is a special case of the multicast address: the recipients are all the stations on the LAN. A broadcast destination address is forty-eight 1s.

Example

- Define the type of the following destination addresses
- a. 4A:30:10:21:10:1A
 - b. 47:20:1B:2E:08:EE
 - c. FF:FF:FF:FF:FF:FF

Solution: To find the type of the address, we need to look at the second hexadecimal digit from the left. If it is even, the address is unicast. If it is odd, the address is multicast. If all digits are Fs, the address is broadcast. Therefore, we have the following:

- a. This is a unicast address because A in binary is 1010 (even).
- b. This is a multicast address because 7 in binary is 0111 (odd).
- c. This is a broadcast address because all digits are Fs in hexadecimal.

Access Method

Since the network that uses the standard Ethernet protocol is a broadcast network, The standard Ethernet chose CSMA/CD with 1-persistent method, Let us use a scenario to see how this method works for the Ethernet protocol.

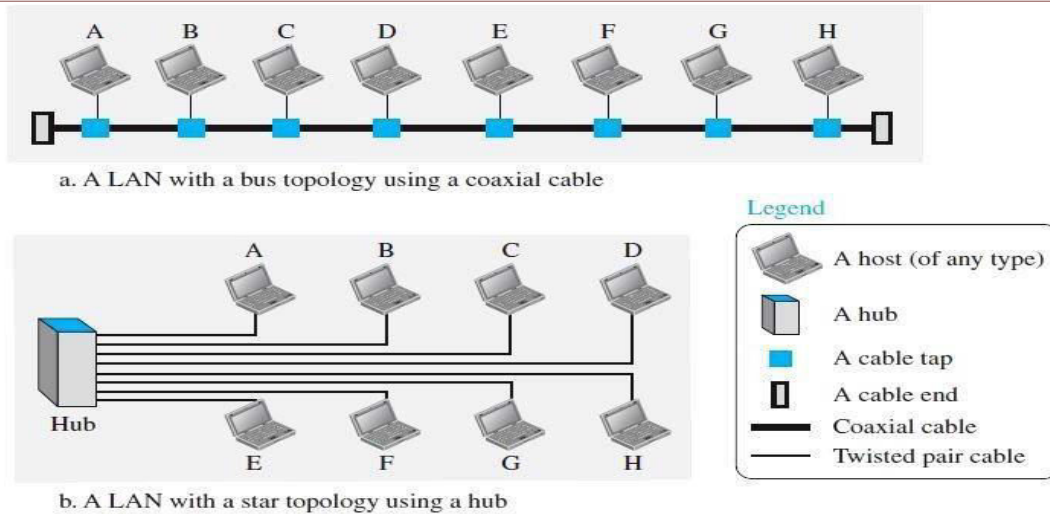


Figure 5: Implementation of standard Ethernet

- Assume station A in Figure.5 has a frame to send to station D. Station A first should check whether any other station is sending (carrier sense). Station A measures the level of energy on the medium (for a short period of time, normally less than $100\mu\text{s}$). If there is no signal energy on the medium, it means that no station is sending (or the signal has not reached station A). Station A interprets this situation as idle medium. It starts sending its frame. On the other hand, if the signal energy level is not zero, it means that the medium is being used by another station. Station A continuously monitors the medium until it becomes idle for $100\mu\text{s}$. It then starts sending the frame. However, station A needs to keep a copy of the frame in its buffer until it is sure that there is no collision.
- The medium sensing does not stop after station A has started sending the frame. Station A needs to send and receive continuously. Two cases may occur:
 - (a) Station A has sent 512 bits and no collision is sensed (the energy level did not go above the regular energy level), the station then is sure that the frame will go through and stops sensing the medium. Where does the number 512 bits come from? If we consider the transmission rate of the Ethernet as 10 Mbps, this means that it takes the station $512/(10 \text{ Mbps}) = 51.2 \mu\text{s}$ to send out 512 bits. With the speed of propagation in a cable (2×10^8 meters), the first bit could have gone 10,240 meters (one way) or only 5120 meters (round trip), have collided with a bit from the last station on the cable, and have gone back. In other words, if a collision were to occur, it should occur by the time the sender has sent out 512 bits (worst case) and the first bit has made a round trip of 5120 meters, if the collision happens in the middle of the cable, not at the end, station A hears the collision earlier and aborts the transmission. The above assumption is that the length of the cable is 5120 meters. The designer of the standard Ethernet actually put a restriction of 2500 meters because we need to consider the delays encountered throughout the journey. It means that they considered the worst case. The whole idea is that if station A does not sense the collision before sending 512 bits, there must have been no collision, because during this time, the first bit has reached the end of the line and all other stations know that a station is sending and refrain from sending. In other words, the problem occurs when another station (for example, the last station) starts sending before the first bit of station A has reached it. The other station mistakenly thinks that the line is free because the first bit has not yet reached it. The restriction of 512 bits actually helps the sending station: The sending

station is certain that no collision will occur if it is not heard during the first 512 bits, so it can discard the copy of the frame in its buffer.

- (b) Station A has sensed a collision before sending 512 bits. This means that one of the previous bits has collided with a bit sent by another station. In this case both stations should refrain from sending and keep the frame in their buffer for resending when the line becomes available. However, to inform other stations that there is a collision in the network, the station sends a 48-bit jam signal. The jam signal is to create enough signal (even if the collision happens after a few bits) to alert other stations about the collision. After sending the jam signal, the stations need to increment the value of K (number of attempts). If after increment $K = 15$, the experience has shown that the network is too busy, the station needs to abort its effort and try again. If $K < 15$, the station can wait a backoff time (T_B) and restart the process. The station creates a random number between 0 and $2^K - 1$, which means each time the collision occurs, the range of the random number increases exponentially. After the first collision ($K = 1$) the random number is in the range (0, 1). After the second collision ($K = 2$) it is in the range (0, 1, 2, 3). After the third collision ($K = 3$) it is in the range (0, 1, 2, 3, 4, 5, 6, 7). So after each collision, the probability increases that the backoff time becomes longer. This is due to the fact that if the collision happens even after the third or fourth attempt, it means that the network is really busy; a longer backoff time is needed.

Efficiency of Standard Ethernet

The efficiency of the Ethernet is defined as the ratio of the time used by a station to send data to the time the medium is occupied by this station. The practical efficiency of standard Ethernet has been measured to be, $\text{Efficiency} = 1 / (1 + 6.4 \times a)$

in which the parameter “a” is the number of frames that can fit on the medium. It can be calculated as $a = (\text{propagation delay}) / (\text{transmission delay})$ because the transmission delay is the time it takes a frame of average size to be sent out and the propagation delay is the time it takes to reach the end of the medium. Note that as the value of parameter a decreases, the efficiency increases. This means that if the length of the media is shorter or the frame size longer, the efficiency increases. In the ideal case, $a = 0$ and the efficiency is 1.

Example 13.3

In the Standard Ethernet with the transmission rate of 10 Mbps, we assume that the length of the medium is 2500 m and the size of the frame is 512 bits. The propagation speed of a signal in a cable is normally 2×10^8 m/s.

$$\begin{aligned} \text{Propagation delay} &= 2500 / (2 \times 10^8) = 12.5 \mu\text{s} & \text{Transmission delay} &= 512 / (10^7) = 51.2 \mu\text{s} \\ a &= 12.5 / 51.2 = 0.24 & \text{Efficiency} &= 39\% \end{aligned}$$

The example shows that $a = 0.24$, which means only 0.24 of a frame occupies the whole medium in this case. The efficiency is 39 percent, which is considered moderate; it means that only 61 percent of the time the medium is occupied but not used by a station.

Implementation

The Standard Ethernet defined several implementations, but only four of them became popular during the 1980s. Table below shows a summary of Standard Ethernet implementations.

Implementation	Medium	Medium Length	Encoding
10Base5	Thick coax	500 m	Manchester
10Base2	Thin coax	185 m	Manchester
10Base-T	2 UTP	100 m	Manchester
10Base-F	2 Fiber	2000 m	Manchester

In the nomenclature 10BaseX, the number defines the data rate (10 Mbps), the term *Base* means baseband (digital) signal, and X approximately defines either the maximum size of the cable in 100 meters (for example 5 for 500 or 2 for 185 meters) or the type of cable, T for unshielded twisted pair cable (UTP) and F for fiber-optic. The standard Ethernet uses a baseband signal, which means that the bits are changed to a digital signal and directly sent on the line.

Encoding and Decoding

All standard implementations use digital signaling (baseband) at 10 Mbps. At the sender, data are converted to a digital signal using the Manchester scheme; at the receiver, the received signal is interpreted as Manchester and decoded into data. Manchester encoding is self-synchronous, providing a transition at each bit interval. Figure 6 shows the encoding scheme for Standard Ethernet.

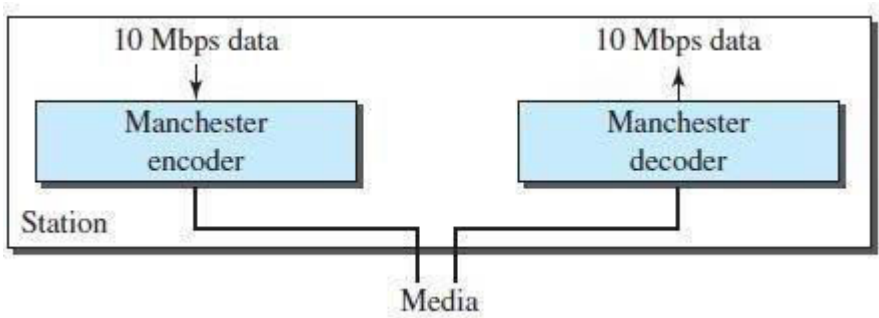


Figure 6: Encoding in a Standard Ethernet implementation

10Base5: Thick Ethernet

The first implementation is called 10Base5, thick Ethernet, or Thickenet. The nickname derives from the size of the cable, which is roughly the size of a garden hose and too stiff to bend with your hands. 10Base5 was the first Ethernet specification to use a bus topology with an external transceiver (transmitter/receiver) connected via a tap to a thick coaxial cable. Figure 7 shows a schematic diagram of a 10Base5 implementation.

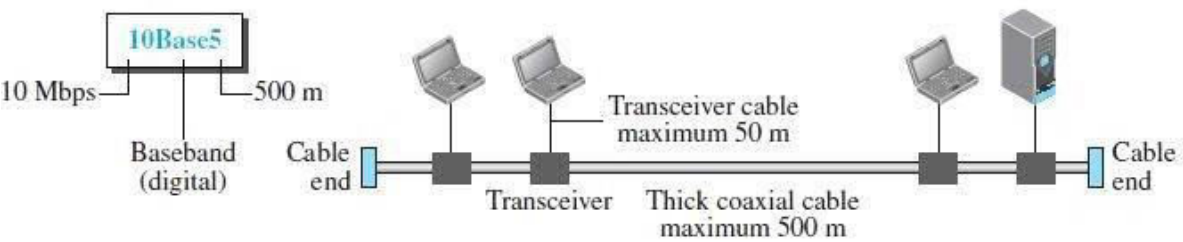


Figure 7 : 10Base5 implementation

The transceiver is responsible for transmitting, receiving, and detecting collisions. The transceiver is connected to the station via a transceiver cable that provides separate paths for sending and receiving. This means that collision can only happen in the coaxial cable.

The maximum length of the coaxial cable must not exceed 500 m, otherwise, there is excessive degradation of the signal. If a length of more than 500 m is needed, up to five segments, each a maximum of 500 meters, can be connected using repeaters.

10Base2: Thin Ethernet

The second implementation is called 10Base2, thin Ethernet, or Cheaper net. 10Base2 also uses a bus topology, but the cable is much thinner and more flexible. The cable can be bent to pass very close to the stations. In this case, the transceiver is normally part of the network interface card (NIC), which is installed inside the station. Figure 8 shows the schematic diagram of a 10Base2 implementation.

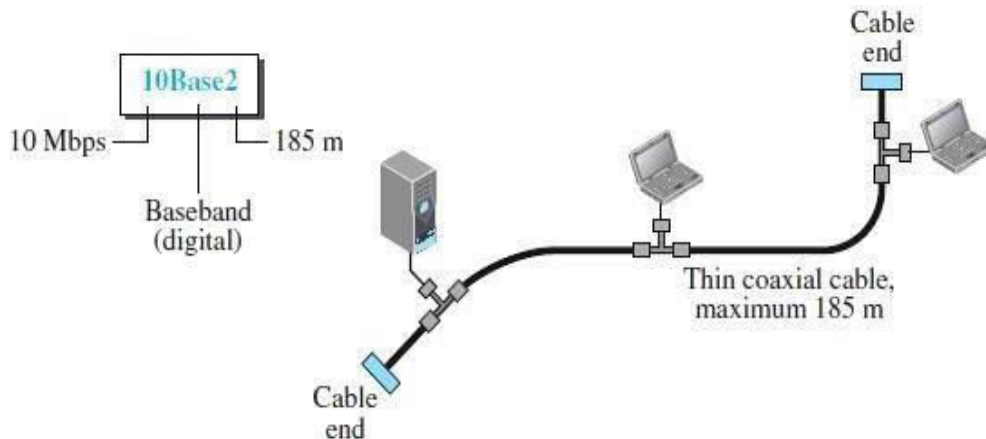


Figure 8: 10Base2 implementation

The collision here occurs in the thin coaxial cable. This implementation is more cost effective than 10Base5 because thin coaxial cable is less expensive than thick coaxial and the tee connections are much cheaper than taps. Installation is simpler because the thin coaxial cable is very flexible. However, the length of each segment cannot exceed 185 m (close to 200 m) due to the high level of attenuation in thin coaxial cable.

10Base-T: Twisted-Pair Ethernet

The third implementation is called **10Base-T** or **twisted-pair Ethernet**. 10Base-T uses a physical star topology. The stations are connected to a hub via two pairs of twisted cable, as shown in Figure 9.

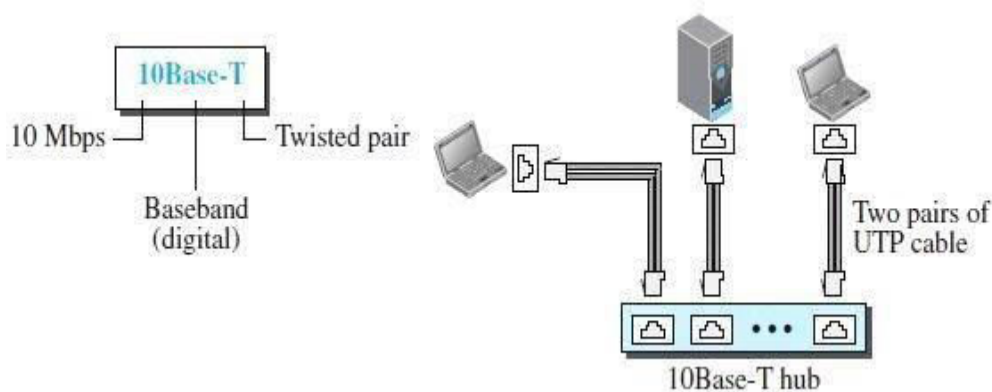


Figure 9: 10Base-T implementation

Two pairs of twisted cable create two paths (one for sending and one for receiving) between the station and the hub. Any collision here happens in the hub. Compared to 10Base5 or 10Base2, we can see that the hub actually replaces the coaxial cable as far as a collision is concerned. The maximum length of the twisted cable here is defined as 100 m, to minimize the effect of

attenuation in the twisted cable.

10Base-F: Fiber Ethernet

Although there are several types of optical fiber 10-Mbps Ethernet, the most common is called **10Base-F**. 10Base-F uses a star topology to connect stations to a hub. The stations are connected to the hub using two fiber-optic cables, as shown in Figure 10.

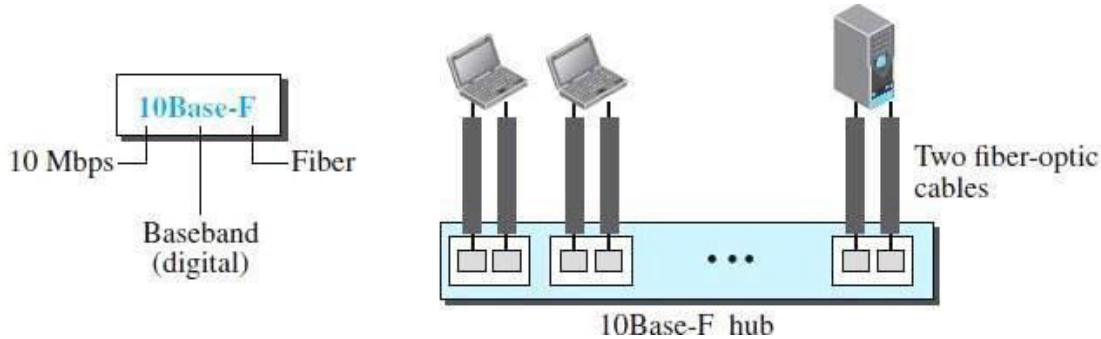


Figure 10: 10Base-F implementation

INTRODUCTION WIRELESS LANs

Wireless communication is one of the fastest-growing technologies. The demand for connecting devices without the use of cables is increasing everywhere. Wireless LANs can be found on college campuses, in office buildings, and in many public areas.,

Architectural Comparison

1. Medium

In a wireless LAN, the medium is air, the signal is generally broadcast. When hosts in a wireless LAN communicate with each other, they are sharing the same medium (multiple access). In a very rare situation, we may be able to create a point-to-point communication between two wireless hosts by using a very limited bandwidth and two-directional antennas.

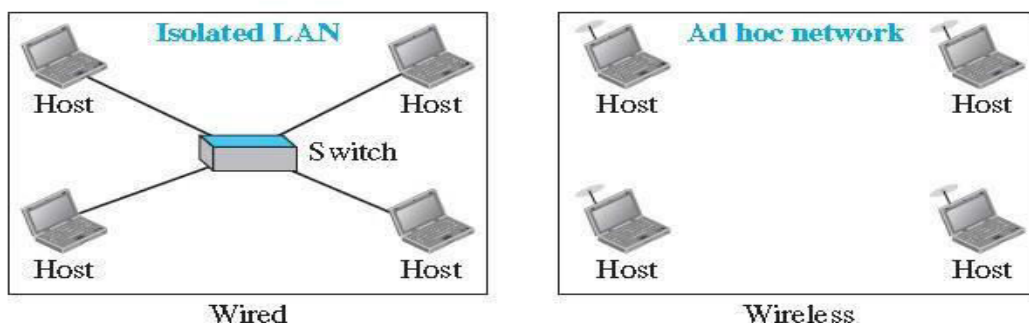
2. Hosts

In a wired LAN, a host is always connected to its network at a point with a fixed link layer address related to its network interface card (NIC). Of course, a host can move from one point in the Internet to another point. In this case, its link-layer address remains the same, but its network-layer address will change. However, before the host can use the services of the Internet, it needs to be physically connected to the Internet. In a wireless LAN, a host is not physically connected to the network; it can move freely and can use the services provided by the network

3. Isolated LANs

A wired isolated LAN is a set of hosts connected via a link-layer switch. A wireless isolated LAN, called an **ad hoc network** in wireless LAN terminology, is a set of hosts that communicate freely with each other. The concept of a link-layer switch does not exist in wireless LANs.

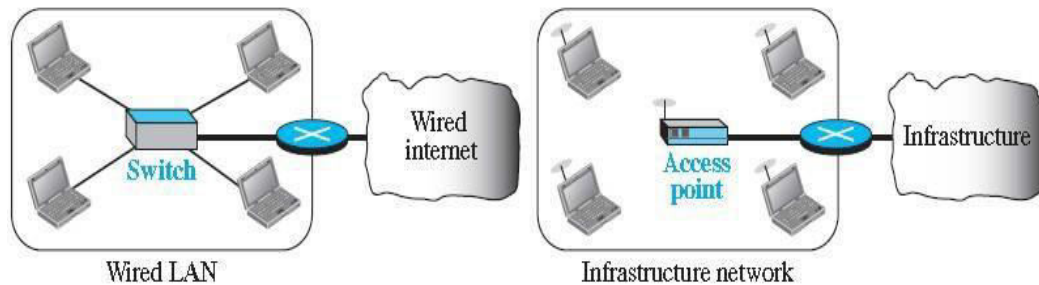
15.1 Isolated LANs: wired versus wireless



4. Connection to Other Networks

A wired LAN can be connected to another network or an internetwork such as the Internet using a router. A wireless LAN may be connected to a wired infrastructure network, to a wireless infrastructure network, or to another wireless LAN.

Figure 15.2 Connection of a wired LAN and a wireless LAN to other networks



5. Moving between Environments

In order to move from the wired environment to a wireless environment we need to change the network interface cards designed for wired environments to the ones designed for wireless environments. We replace the link-layer switch with an access point. In this change, the link-layer addresses will change but the network-layer addresses (IP addresses) will remain the same; we are moving from wired links to wireless links.

Access Control

The most important issue we need to discuss in a wireless LAN is access control. The CSMA/CD algorithm does not work in wireless LANs for three reasons:

1. To detect a collision, a host needs to send and receive at the same time (sending the frame and receiving the collision signal), which means the host needs to work in a duplex mode. Wireless hosts do not have enough power to do so (the power is supplied by batteries). They can only send or receive at one time.
2. Hidden station problem, In this a station may not be aware of another station's transmission due to some obstacles or range problems, collision may occur but not be detected. Hidden stations can reduce the capacity of the network because of the possibility of collision.
3. Since the distance between stations can be great. Signal fading could prevent a station at one end from hearing a collision at the other end.

To overcome the above three problems, Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) was invented for wireless LANs

Figure 15.3 Hidden station problem

