Unit-3

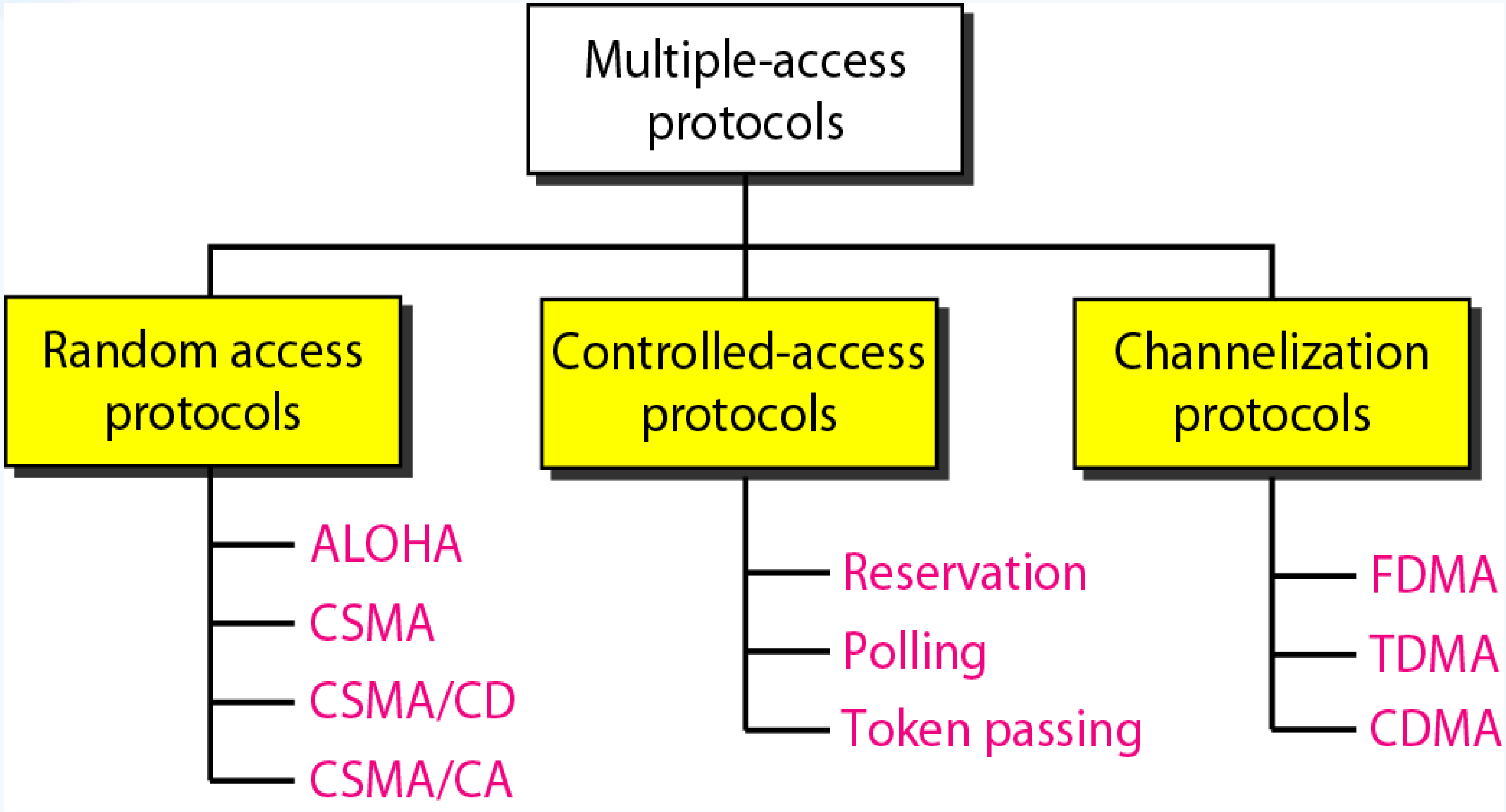
Medium Access Sub Layer

Dr. Yogesh Thakare

CSE(AIML)

RCOEM, Nagpur

Multiple Access



RANDOM ACCESS

- In random access or contention methods, no station is superior to another station and none is assigned the control over another.
- No station permits, or does not permit, another station to send.
- 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.

- 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.
- To avoid access conflict or to resolve it when it happens, each station follows a procedure that answers the following questions:

- When can the station access the medium?
- What can the station do if the medium is busy?
- How can the station determine the success or failure of the transmission?
- What can the station do if there is an access conflict?

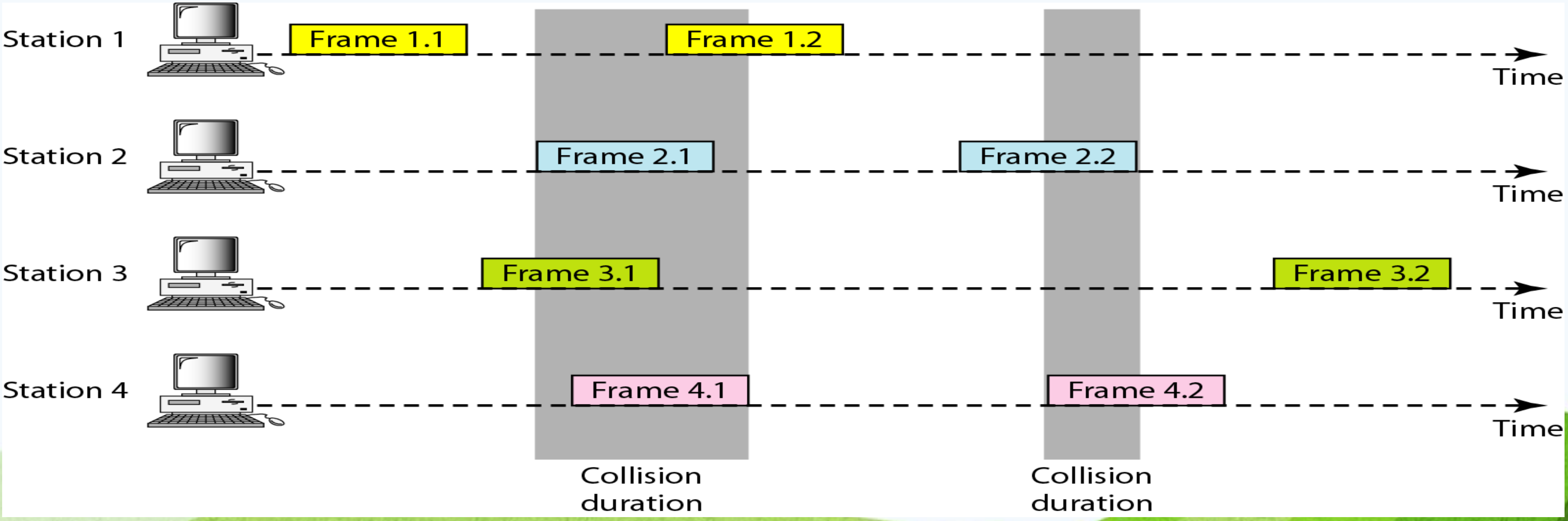
- The random access methods we study in this chapter have evolved from a very interesting protocol known as ALOHA, which used a very simple procedure called multiple access (MA).
- The method was improved with the addition of a procedure that forces the station to sense the medium before transmitting. This was called carrier sense multiple access. This method later evolved into two parallel methods: carrier sense multiple access with collision detection (CSMA/CD) and carrier sense multiple access with collision avoidance *(CSMA/CA)*.
- *CSMA/CD* tells the station what to do when a collision is detected. *CSMA/CA* 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.
- It is obvious that there are potential collisions in this arrangement. 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.
- However, since there is only one channel to share, there is the possibility of collision between frames from different stations.



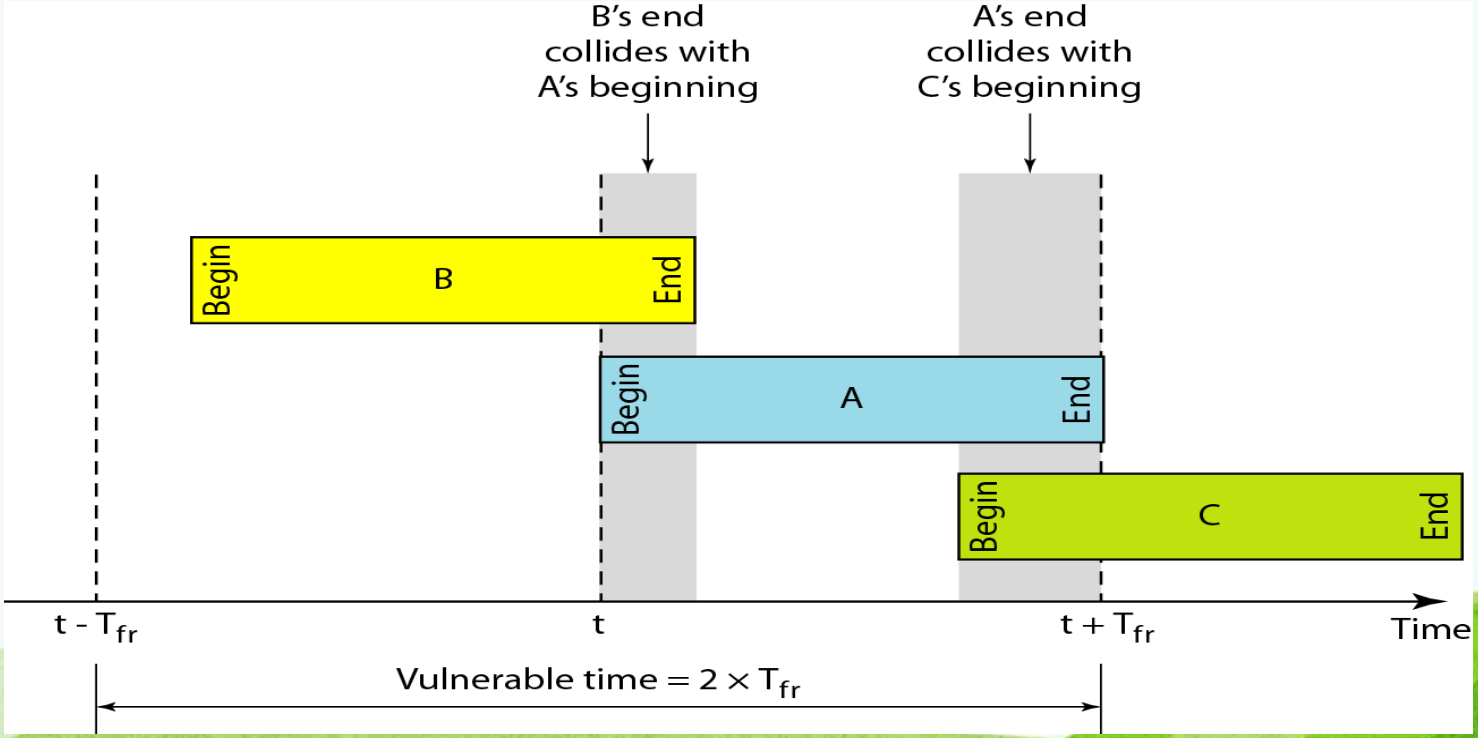
- There are four stations (unrealistic assumption) that contend with one another for access to the shared channel.
- The figure shows that each station sends two frames; there are a total of eight frames on the shared medium. Some of these frames collide because multiple frames are in contention for the shared channel.

- Only two frames survive: frame 1.1 from station 1 and frame 3.2 from station 3.
- We need to mention that even 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 we need to resend the frames 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. We call this time the back-off time *TB*.

Vulnerable time

- Let us find the length of time, the **vulnerable time,** in which there is a possibility of collision. We assume that the stations send fixed-length frames with each frame taking *Tfr* S to send.
- Figure 12.5 shows the vulnerable time for station A.



- Station A sends a frame at time *t*. Now imagine station B has already sent a frame between *t* - *T*fr and *t*. This leads to a collision between the frames from station A and station B. The end of B's frame collides with the beginning of A's frame. On the other hand, suppose that station C sends a frame between *t* and *t* + *Tfr* .

- Here, there is a collision between frames from station A and station C. The beginning of C's frame collides with the end of A's frame.
- Looking at Figure 12.5, we see that the vulnerable time, during which a collision may occur in pure ALOHA, is 2 times the frame transmission time.
    - Pure ALOHA vulnerable time = 2 x Tfr

Problem

- 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 *Tfr* is 200 bits/200 kbps or 1 ms. The vulnerable time is 2 x1 ms =2 ms.
- This means no station should send later than 1 ms before this station starts transmission and no station should start sending during the one 1-ms period that this station is sending.

Throughput

- Let us call G the average number of frames generated by the system during one frame transmission time.
- Then it can be proved that the average number of successful transmissions for pure ALOHA is

$$S = G \times e^{-2G}$$

- The maximum throughput Smax is 0.184, for G = 0.5.

- In other words, if one-half a frame is generated during one frame transmission time (in other words, one frame during two frame transmission times), then 18.4 percent of these frames reach their destination successfully.
- This is an expected result 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.

Problem

- 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

a. 1000 frames per second

b. 500 frames per second

c. 250 frames per second

Solution

- The frame transmission time is *200/200* kbps or 1 ms.
- a. If the system creates 1000 frames per second, this is 1 frame per millisecond.
- The load is
- 1. In this case S =G x e^{-2G} or S =0.135 (13.5 percent).
- This means that the throughput is 1000 X 0.135 =135 frames.
- Only 135 frames out of 1000 will probably survive.

- b. If the system creates 500 frames per second, this is (1/2) frame per millisecond.
- The load is *(0.5).*
- In this case S = G x *e-2G* or S = 0.184 (18.4 percent).
- This means that the throughput is 500 x 0.184 =92 and that only 92 frames out of 500 will probably survive.
- Note that this is the maximum throughput case, percentagewise.

- c. If the system creates 250 frames per second, this is (1/4) frame per millisecond.
- The load is (1/4).
- In this case S = G x *e-2G* or S =0.152 (15.2 percent).
- This means that the throughput is 250 x 0.152 = 38.
- Only 38 frames out of 250 will probably survive.

Problem

- A group of N stations share a 56-kbps pure ALOHA channels. Each station outputs a 1000-bit frame on average once every 100 sec, even if the previous one has not yet been sent (e. g., the stations can buffer outgoing frames). What is the maximum value of N?

- Answer

With pure ALOHA,

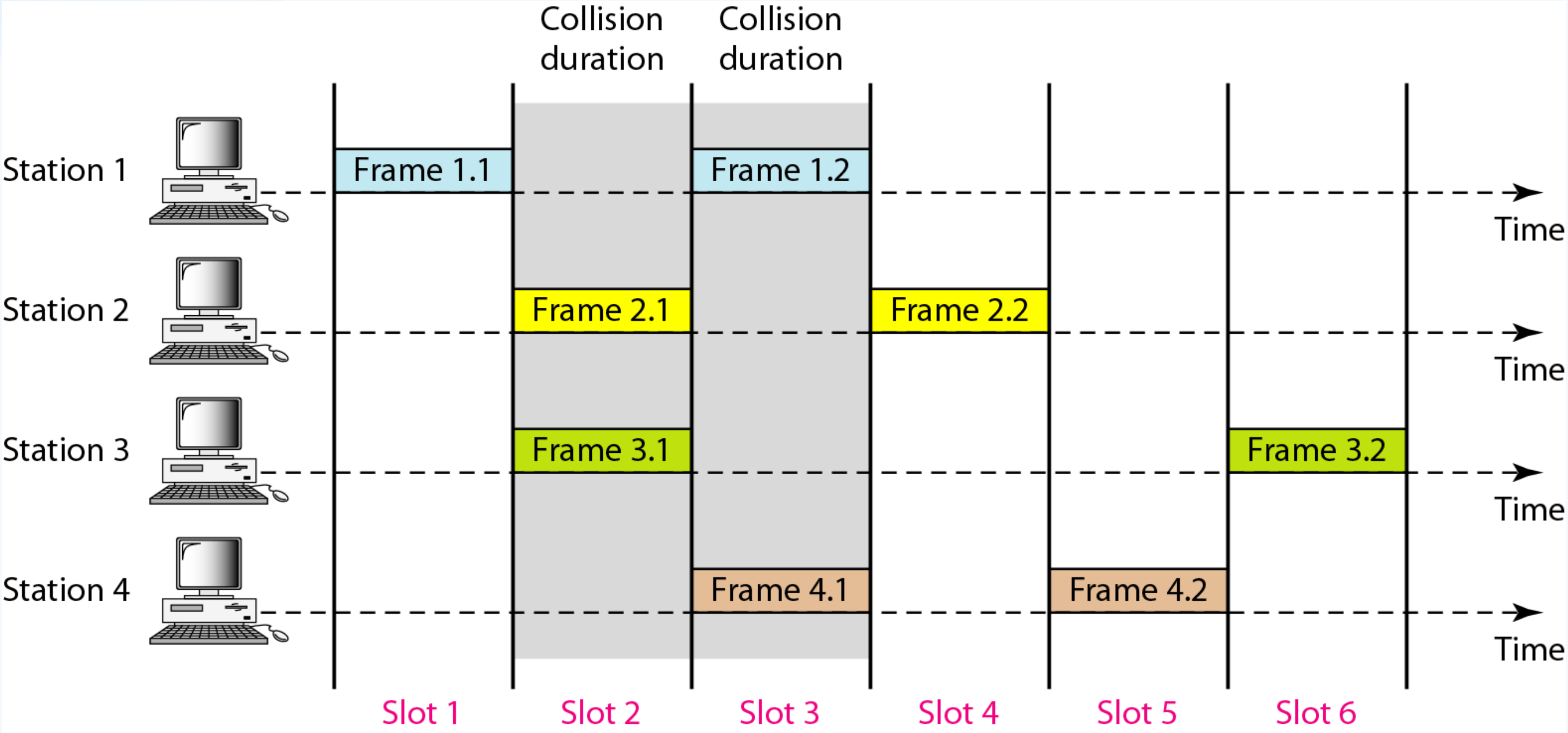
the usable bandwidth is 0.184×56 kbps = 10.3 kbps.

Each station requires 10 bps, so $N = 10300/10 = 1030$ stations

Slotted ALOHA

- Pure ALOHA has a vulnerable time of 2 x *Tfr .* 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 soon 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 Tfr s and force the station to send only at the beginning of the time slot.

Slotted ALOHA



- Because 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.
- Of course, 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 Tfr.

Vulnerable Time

- The vulnerable time for slotted ALOHA is one-half that of pure ALOHA.

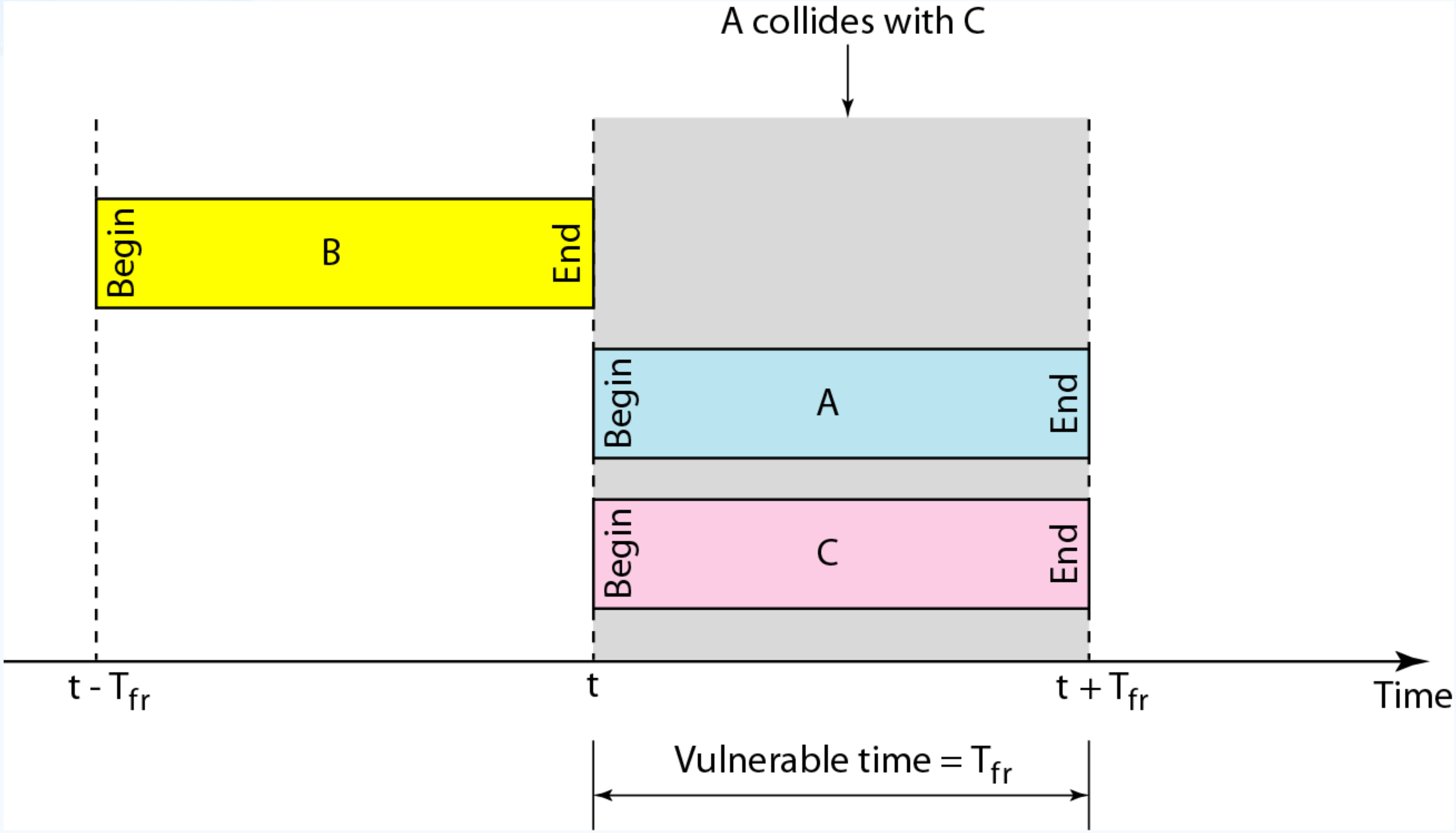
Slotted ALOHA vulnerable time = Tfr

- Throughput

- It can be proved that the average number of successful transmissions for slotted ALOHA is

$$S = G \times e^{-G}.$$

- The maximum throughput S_{max} is 0.368, when G = 1. In other words, if a frame is generated during one frame transmission time, then 36.8 percent of these frames reach their destination successfully. This result can be expected 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 station generates a frame during this time), the frame will reach its destination successfully.



A collides with C
Begin
B
End
Begin
A
End
Begin
C
End
t - T_fr
t
t + T_fr
Time
Vulnerable time = T_fr

Problem

- A slotted ALOHA network transmits 200-bit frames using a shared channel with a 200-kbps bandwidth. Find the throughput if the system (all stations together) produces
- a. 1000 frames per second
- b. 500 frames per second
- c. 250 frames per second

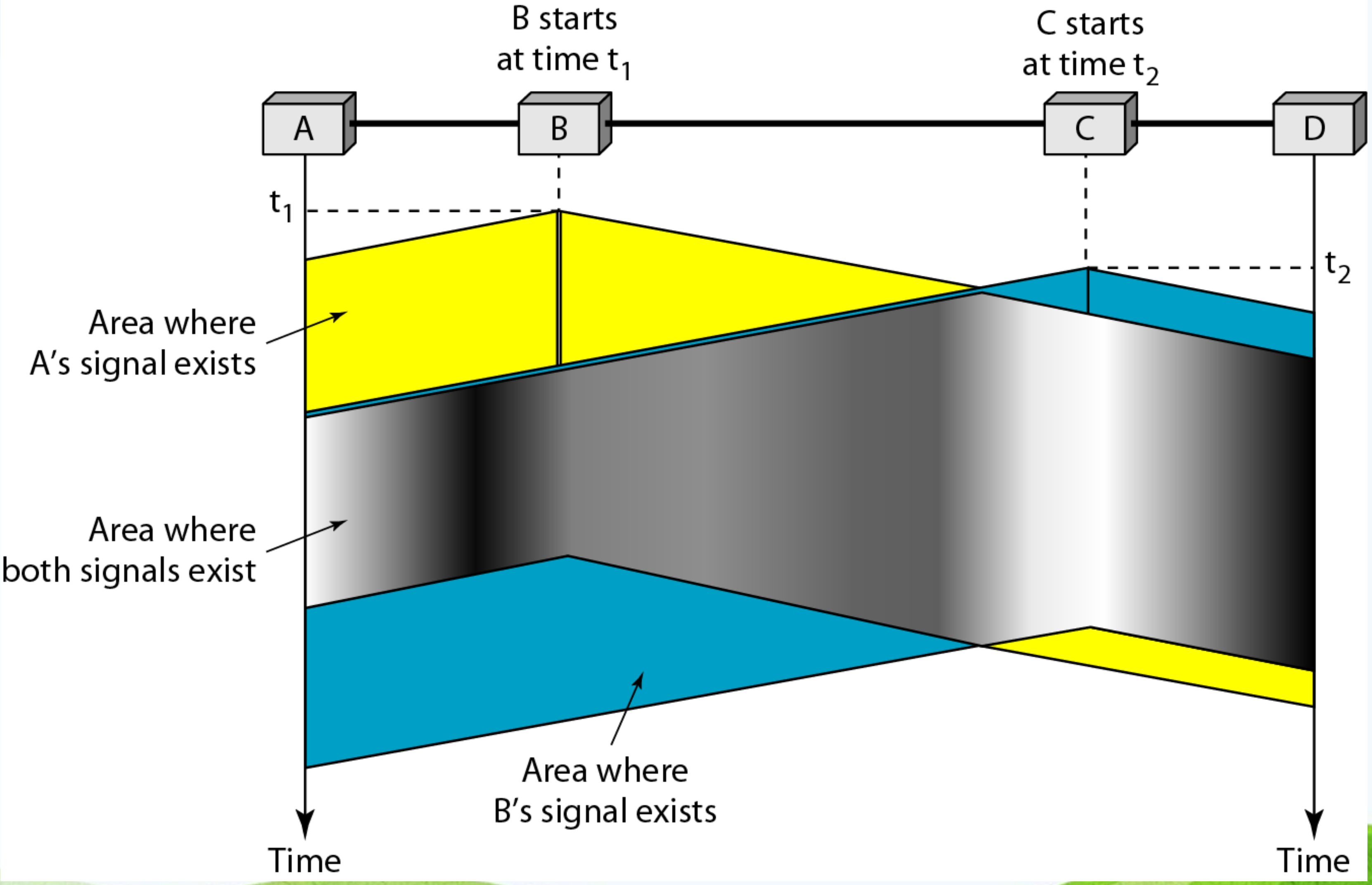
Solution

- This situation is similar to the previous exercise except that the network is using slotted ALOHA instead of pure ALOHA. The frame transmission time is *200/200* kbps or 1 ms.
- a. In this case G is 1. So S =G x *e-G* or *S* =0.368 (36.8 percent). This means that the throughput is 1000 x 0.0368 =368 frames. Only 368 out of 1000 frames will probably survive. Note that this is the maximum throughput case, percentagewise.
- b. Here G is 1/2. In this case S =G x *e-G* or S =0.303 (30.3 percent). This means that the throughput is 500 x 0.0303 =151. Only 151 frames out of 500 will probably survive.
- c. Now Gis 1/4. In this case S =G x *e-G* or S =0.195 (19.5 percent). This means that the throughput is 250 x 0.195 = 49. Only 49 frames out of 250 will probably survive.

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 Figure

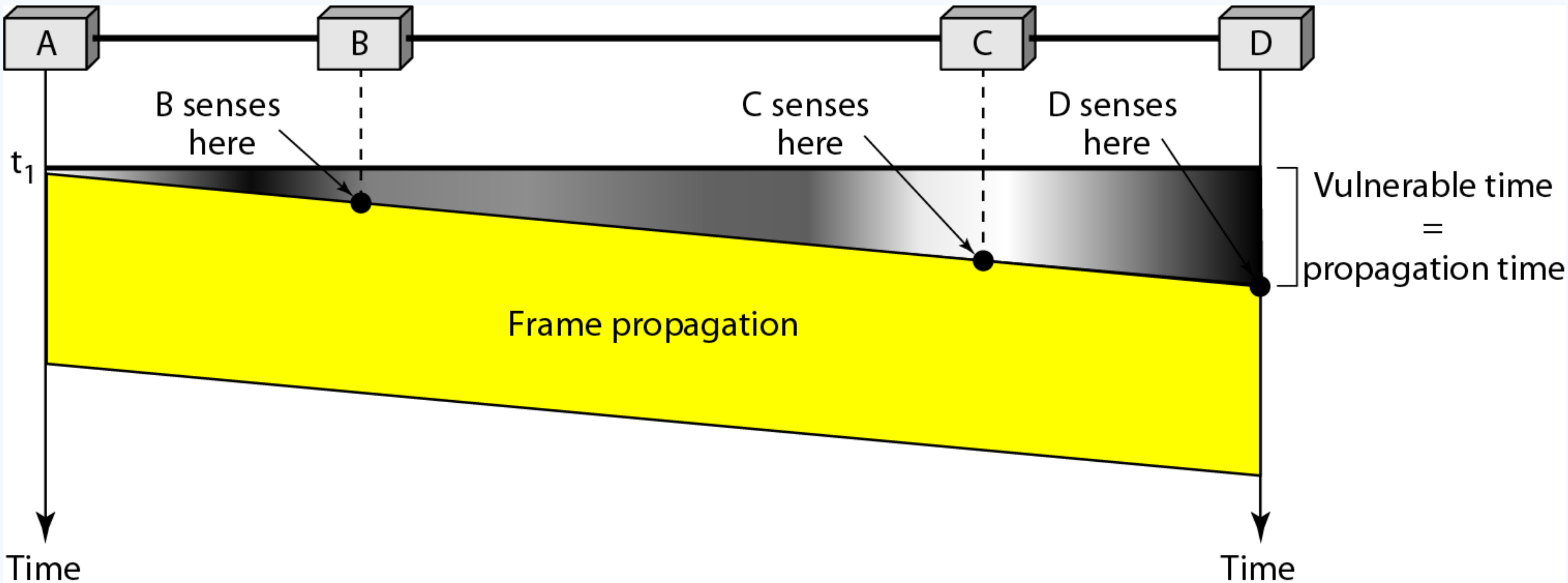


- Stations are connected to a shared channel (usually a dedicated medium).
- The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time (although very short) for the first bit to reach every station and for every station to sense it.
- In other words, a 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 *t1'* station B senses the medium and finds it idle, so it sends a frame.
- At time *t2 (t2> t1)'* 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

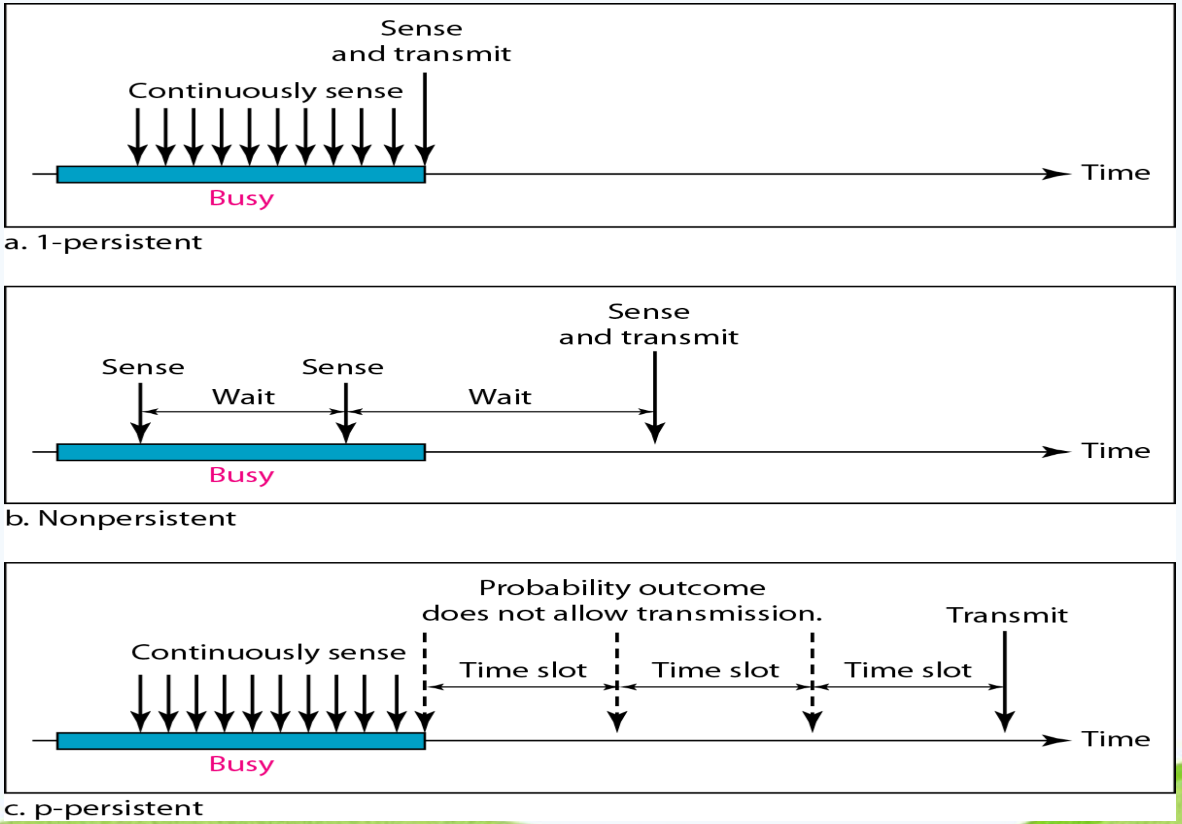
- The vulnerable time for CSMA is the propagation time T_p . This is the time needed for a signal to propagate from one end of the medium to the other.
- When a station sends a frame, and any other station tries to send a frame during this time, a collision will result. But if the first bit of the frame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



A
B
C
D
B senses here
C senses here
D senses here
t_1
Vulnerable time = propagation time
Frame propagation
Time
Time

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 I-persistent method, the nonpersistent method, and the p-persistent method.



I-Persistent

- The **I-persistent method** is simple and straightforward.
- **In** this method, after the station finds the line idle, it sends its frame immediately (with probability I).
- This method has the highest chance of collision because two or more stations may find the line idle and send their frames immediately.

Nonpersistent

- **In** the **nonpersistent 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 nonpersistent 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.

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.

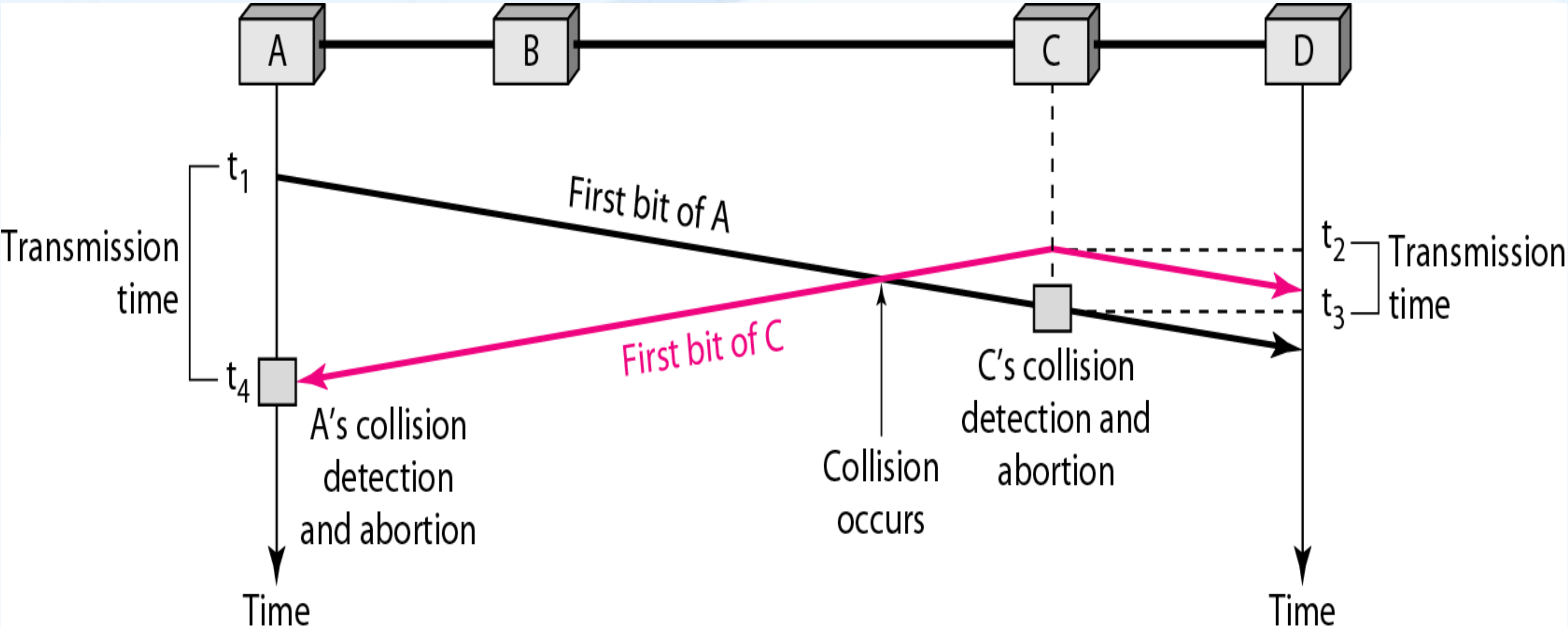
2. With probability *q* = 1 - *p,* the station waits for the beginning of the next time slot and checks the line again.

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.

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.


A
B
C
D
t_1
Transmission time
First bit of A
t_2
t_3
Transmission time
First bit of C
t_4
A's collision detection and abortion
Collision occurs
C's collision detection and abortion
Time
Time

- 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 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 - tl;* C transmits for the duration *t3 - t2'.*
- At time t4, the transmission of A:s frame, though incomplete, is aborted; at time t3, the transmission of B's frame, though incomplete, is aborted.
- Minimum Frame Size
- Therefore, the frame transmission time *T*fr must be at least two times the maximum propagation time *Tp.*

$$T_{fr} = 2 \times T_p$$

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, as we see later) is 25.611S, what is the minimum size of the frame?

- Answer:-

- The frame transmission time is T_{fr} = 2 x T_p =51.2 μs. This means, in the worst case, a station needs to transmit for a period of 51.2 μs to detect the collision. The minimum size of the frame is 10 Mbps x 51.2 μs =512 bits or 64 bytes.

Problem

- Consider a CSMA/CD network running at 1 Gbps over a 1 km long cable with no repeaters. The signal speed in the cable is 200,000 km/sec. What is the minimum frame size?

- Answer:-

- We must keep in mind that in CSMA/CD , for a station to get some surety of successful transmission the contention interval (time during which the station is transmitting) should have at least 2T slot width where T is time for signal to propagate between two farthest stations i.e. there must be enough time for the front of the frame to reach the end of the cable and then for an error message to be sent back to the start before the entire frame is transmitted.

- As a result for a 1 km cable the one way propagation time = 1/200000 = 5 x 10-6 = 5 µsec s
- o for both ways it would be = 2 x 5 µsec = 10 µsec
- To make CSMA/CD work, it must be impossible to transmit an entire frame in this interval.
- At 1 Gbps, all frames shorter than 10,000 bits can be completely transmitted in under 10 µsec, so the minimum frame is 10,000 bits or 1250 bytes. i.e.
- 10^9 bps x 10 x 10^-6 sec = 10^4 bits
- 104 bits / 8 = 1250 bytes

Problem

- What is the length of a contention slot in CSMA/CD for (a) a 2-km twin-lead cable (signal propagation speed is 82% of the signal propagation speed in vacuum)?, and (b) a 40-km multimode fiber optic cable (signal propagation speed is 65% of the signal propagation speed in vacuum)?
- Answer:-

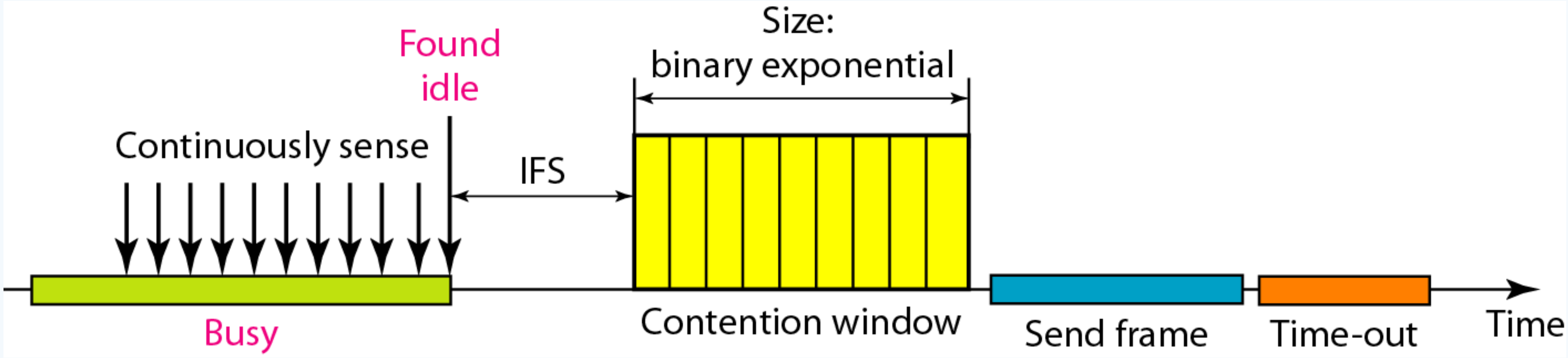
(a) Signal propagation speed in twin lead is 2.46×10^8 m/sec. Signal propagation time for 2 km is 8.13 μsec. So, the length of contention slot is 16.26 μsec. (b) Signal propagation speed in multimode fiber is 1.95×10^8 m/s. Signal propagation time for 40 km is 205.13 μsec. So, the length of contention slot is 410.26 μsec.

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

- The basic idea behind *CSMA/CD* is that a station needs to be able to receive while transmitting to detect a collision. When there is no collision, the station receives one signal: its own signal. When there is a collision, the station receives two signals: its own signal and the signal transmitted by a second station. To distinguish between these two cases, the received signals in these two cases must be significantly different. In other words, the signal from the second station needs to add a significant amount of energy to the one created by the first station.

- We need to avoid collisions on wireless networks because they cannot be detected. Carrier sense multiple access with collision avoidance *(CSMAICA)* was invented for this network.

- Collisions are avoided through the use of CSMA/CA's three strategies: the interframe space, the contention window, and acknowledgments,



Interframe 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 interframe 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.

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.

- *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 that the receiver has received the frame.

IEEE 802 standard Protocol

- The data-link layer and the physical layer are the territory of the local and wide area networks. This means that when we discuss these two layers, we are talking about networks that are using them.

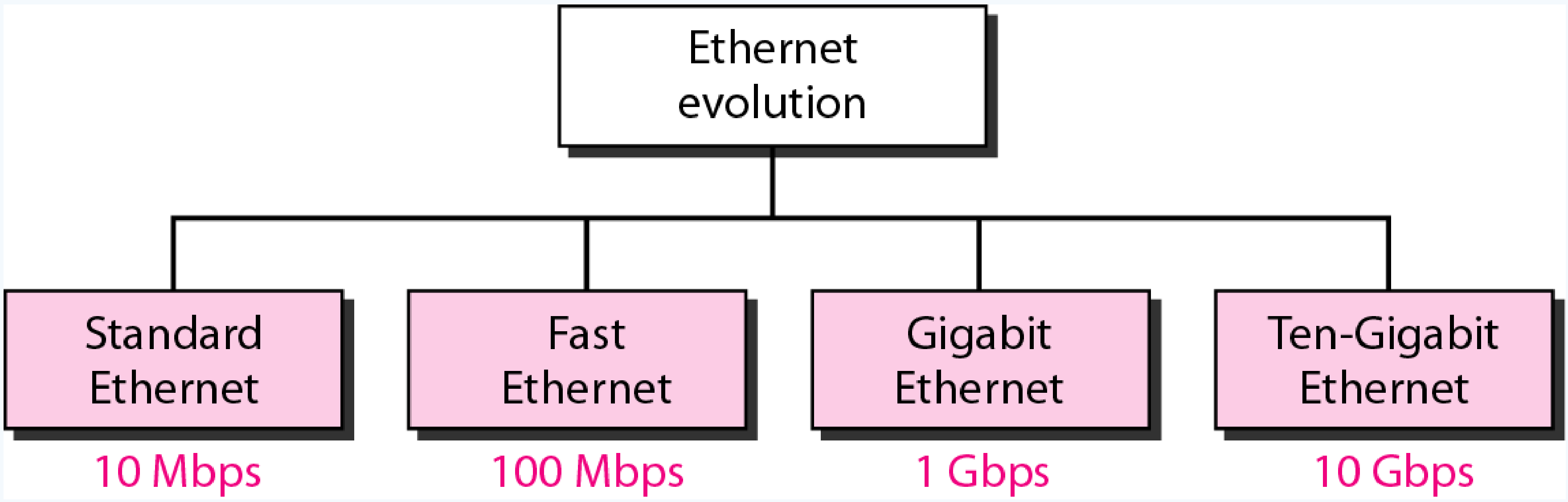
IEEE Project 802

- 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.

IEEE divided the Data link layer into two sublayer:

- upper layer : logical link control (LLC); flow and error control.
- Lower sublayer : Multiple access (MAC); media access control.
    - Multiple access (MAC) :for resolving access to the shared media.
    - If channel is dedicated (point to point) we do not need the (MAC); sublayer.

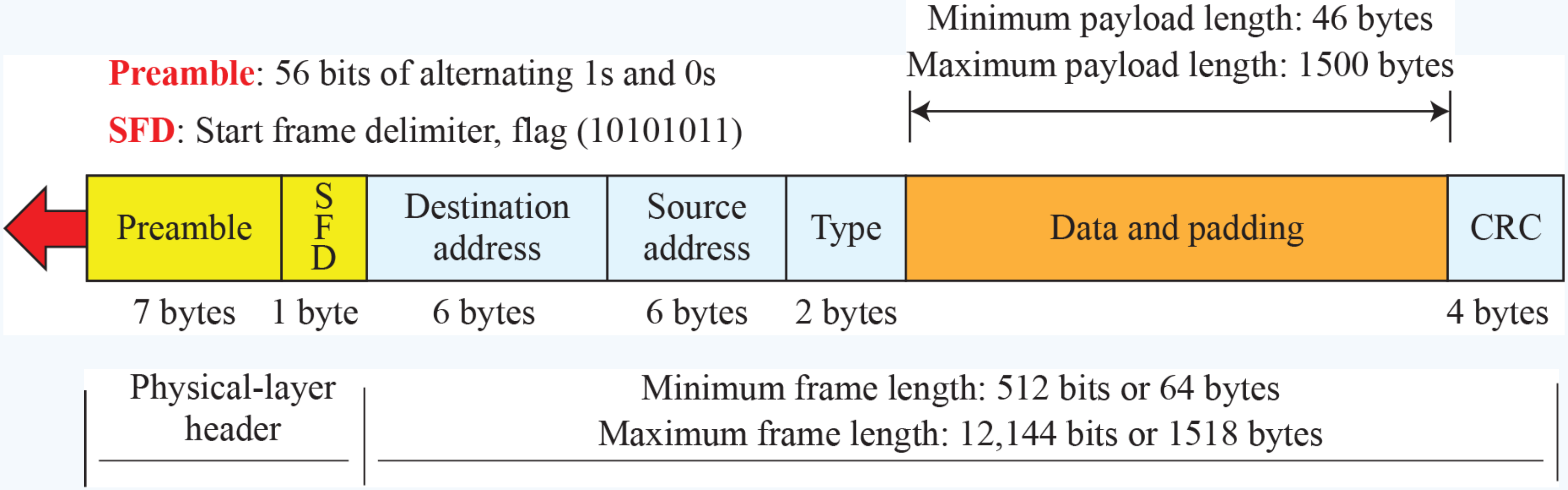
ETHERNET Evolution



IEEE frame Format

- The Ethernet frame contains seven fields: preamble, SFD, DA, SA, length or type of protocol data unit (PDU), upper-layer data, and the CRC. Ethernet does not provide any mechanism for acknowledging received frames, making it what is known as an unreliable medium. Acknowledgments must be implemented at the higher layers.

Figure 13.3: Ethernet frame



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

Field in Frame format

- Preamble:-
- The first field of the 802.3 frame contains 7 bytes (56 bits) of alternating 0's and 1's that alerts the receiving system to the coming frame and enables it to synchronize its input timing. 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 (formally) part of the frame.

Start frame delimiter (SFD)

- The second 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 is 11 and alerts the receiver that the next field is the destination address.

Destination address (DA)

- The DA field is 6 bytes and contains the physical address of the destination station or stations to receive the packet.

- Source address (SA)
- The SA field is also 6 bytes and contains the physical address of the sender of the packet. We will discuss addressing shortly.

- Length or type

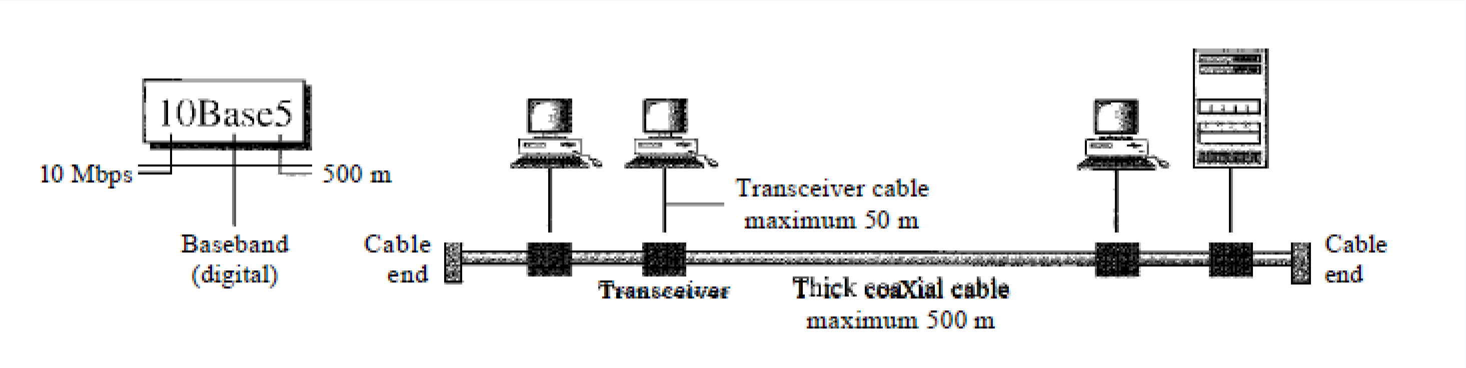
- This field is defined as a type field or length field. The original Ethernet used this field as the type field to define the upper-layer protocol using the MAC frame. The IEEE standard used it as the length field to define the number of bytes in the data field. Both uses are common today.

- Data:-
- This field carries data encapsulated from the upper-layer protocols. It is a minimum of 46 and a maximum of 1500 bytes, as we will see later.
- CRC:-
- The last field contains error detection information, in this case a CRC-32

1OBase5: Thick Ethernet

- The first implementation is called **10Base5, thick Ethernet, or Thicknet.** 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 13.10 shows a schematic diagram of a 10Base5 implementation.

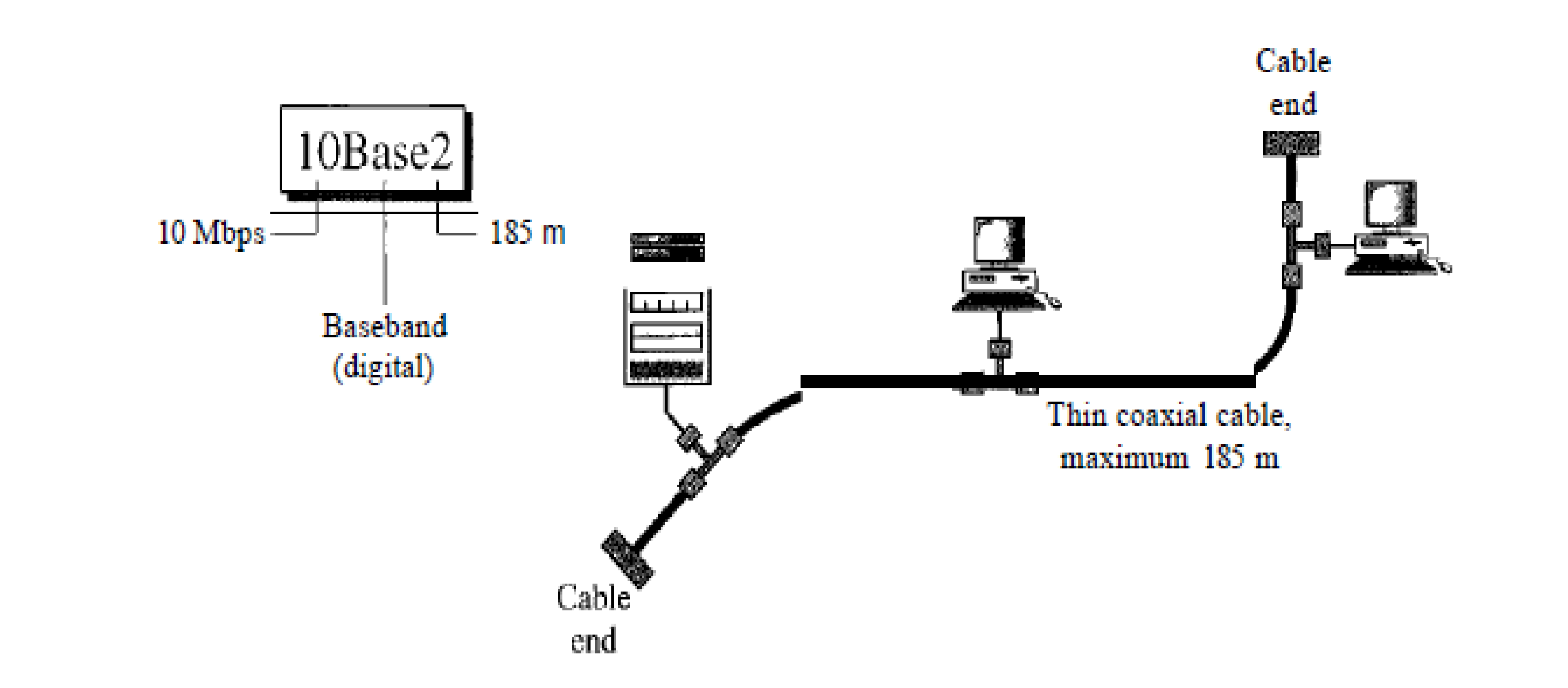
10Base5



10Base5
10 Mbps
500 m
Baseband
(digital)
Transceiver cable
maximum 50 m
Cable
end
Transceiver
Thick coaxial cable
maximum 500 m
Cable
end

10Base2: Thin Ethernet

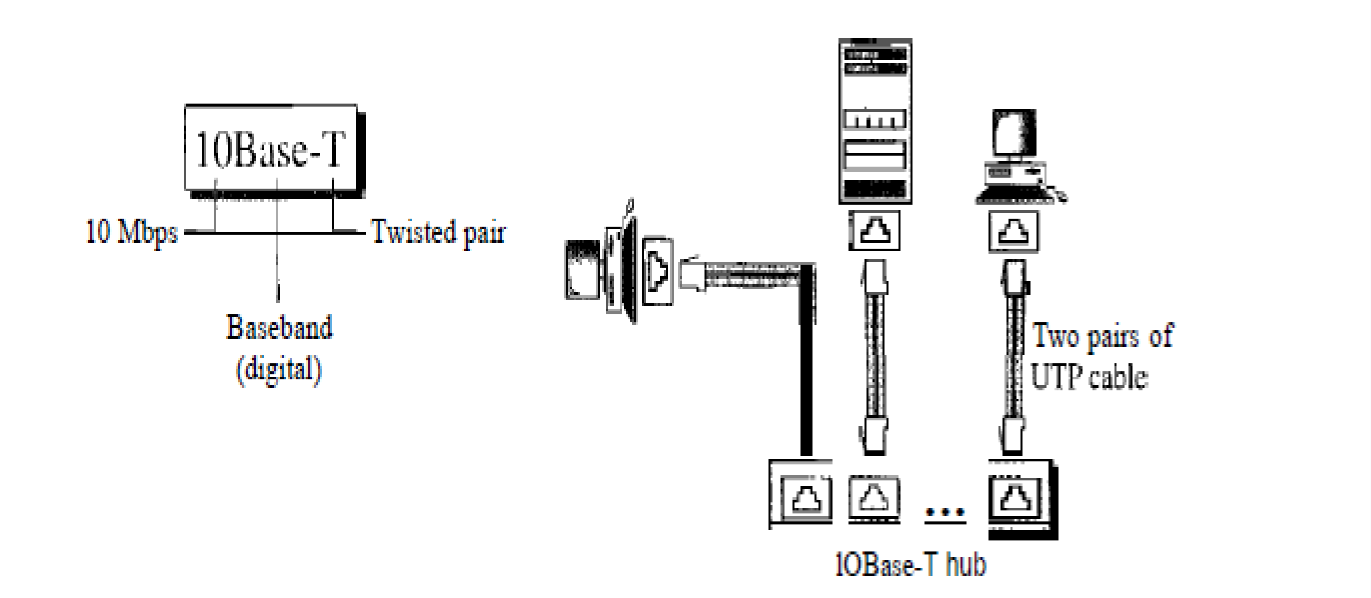
- The second implementation is called lOBase2, **thin** Ethernet, or Cheapernet. IOBase2 also uses a bus topology, but the cable is much thinner and more flexible. The cable can be bent to pass very close to the stations. In this case, the transceiver is normally part of the network interface card (NIC), which is installed inside the station. Figure 13.11 shows the schematic diagram of a IOBase2 implementation.


10Base2
10 Mbps
185 m
Baseband
(digital)
Cable
end
Thin coaxial cable,
maximum 185 m
Cable
end

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 13.12.
- Note that 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 13.13.