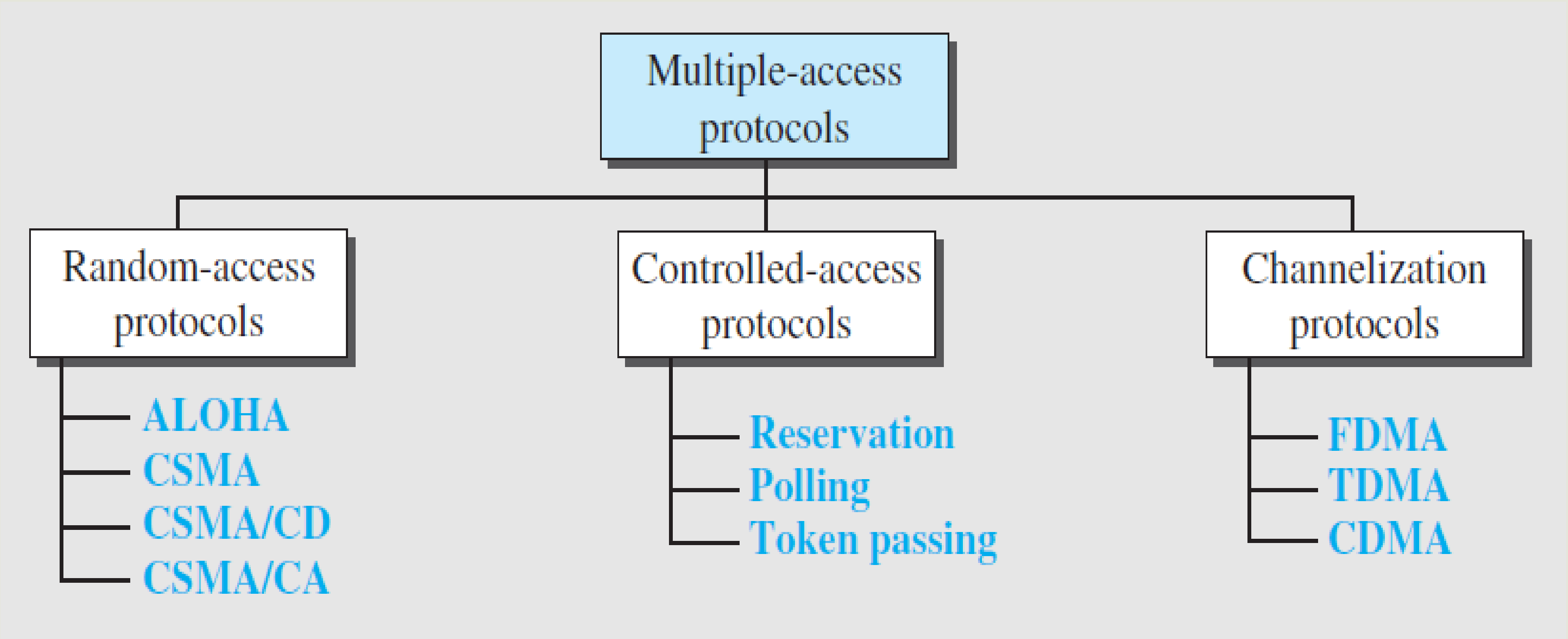
Media Access Control

Unit- 2

Media Access Control

- 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.
- The problem of controlling the access to the medium such as two people do not speak at the same time, do not interrupt each other, do not monopolize and so on.
- Many protocols have been devised to handle access to a shared link.
- All of these protocols belong to a sublayer in the data-link layer called media access control (MAC).

Taxonomy of multiple-access protocols



RANDOM ACCESS / 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).
- There is **no scheduled time** for a station to transmit.
- Transmission is **random** among the stations. No rules specify which station should send next.
- **Stations compete with one another** to access the medium.

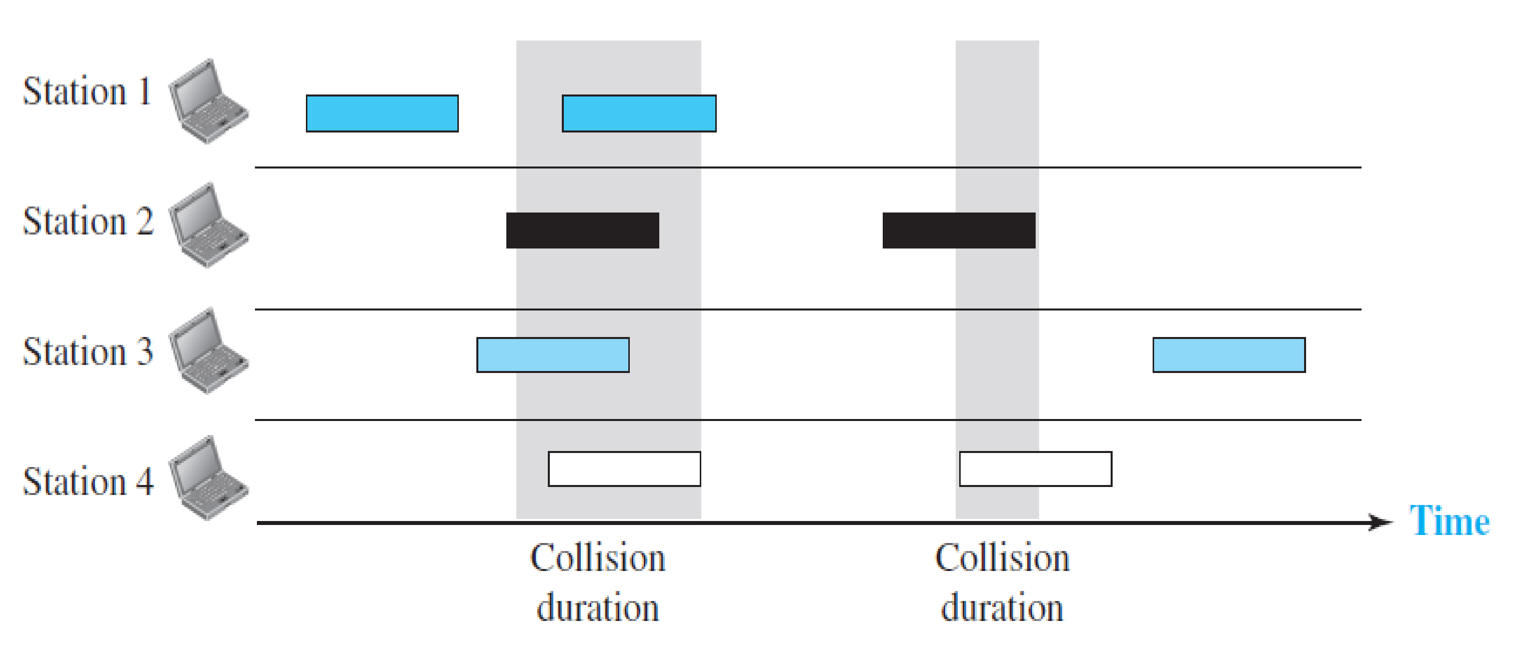
RANDOM ACCESS / CONTENTION METHODS

- 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, 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?

ALOHA

Pure ALOHA

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



Pure ALOHA

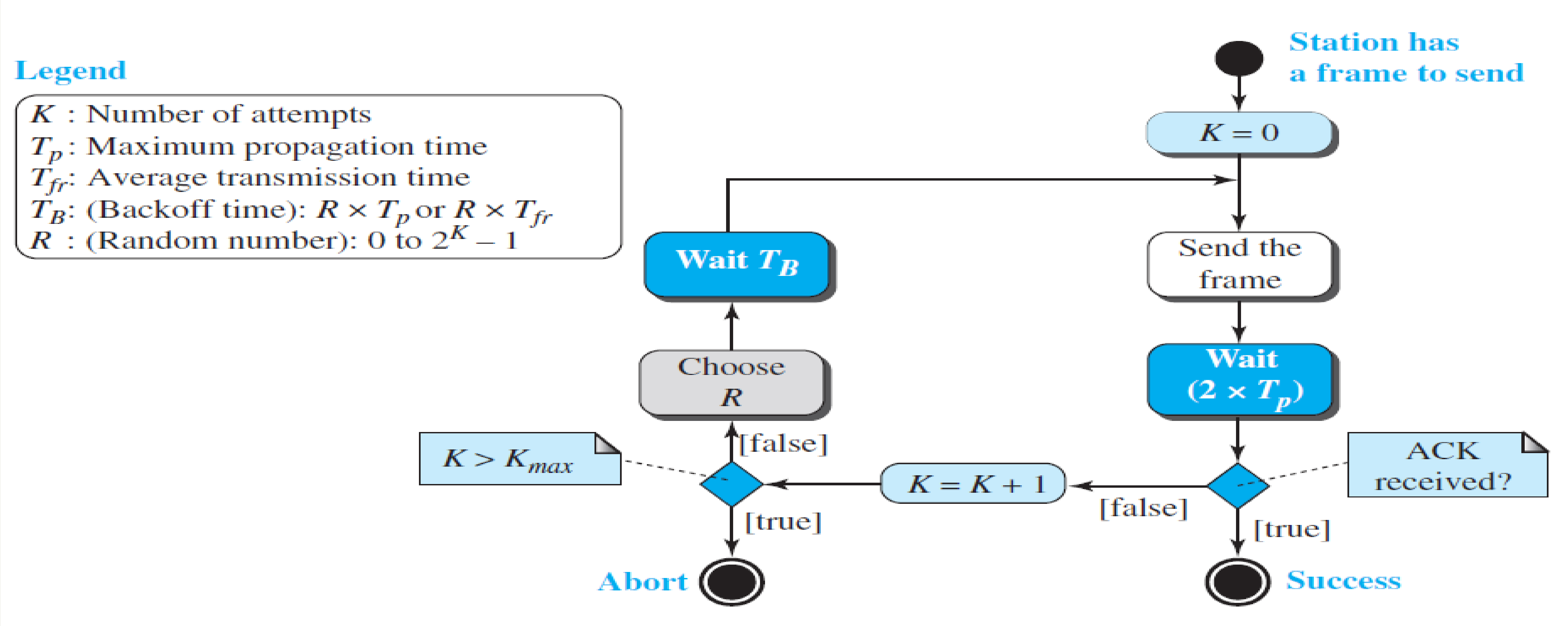
- The pure ALOHA protocol **relies on acknowledgments** from the receiver.
- If the acknowledgment does not arrive after a time-out period, the station assumes that the frame (or the acknowledgment) has been destroyed and resends the frame.
- A collision involves two or more stations.
- If all these 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 ***backoff time*** T_B.

Pure ALOHA

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.
- The **time-out period** is equal to the max. possible round-trip propagation delay, that is **twice the amount of time required to send a frame between the 2 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).
- One common formula for T_B is the binary exponential backoff.
- For each retransmission, a multiplier **$R = 0$ to $2^K - 1$** 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.
- Range of the random numbers increases after each collision. The value of $K_{max} = 15$.

Procedure for pure ALOHA protocol



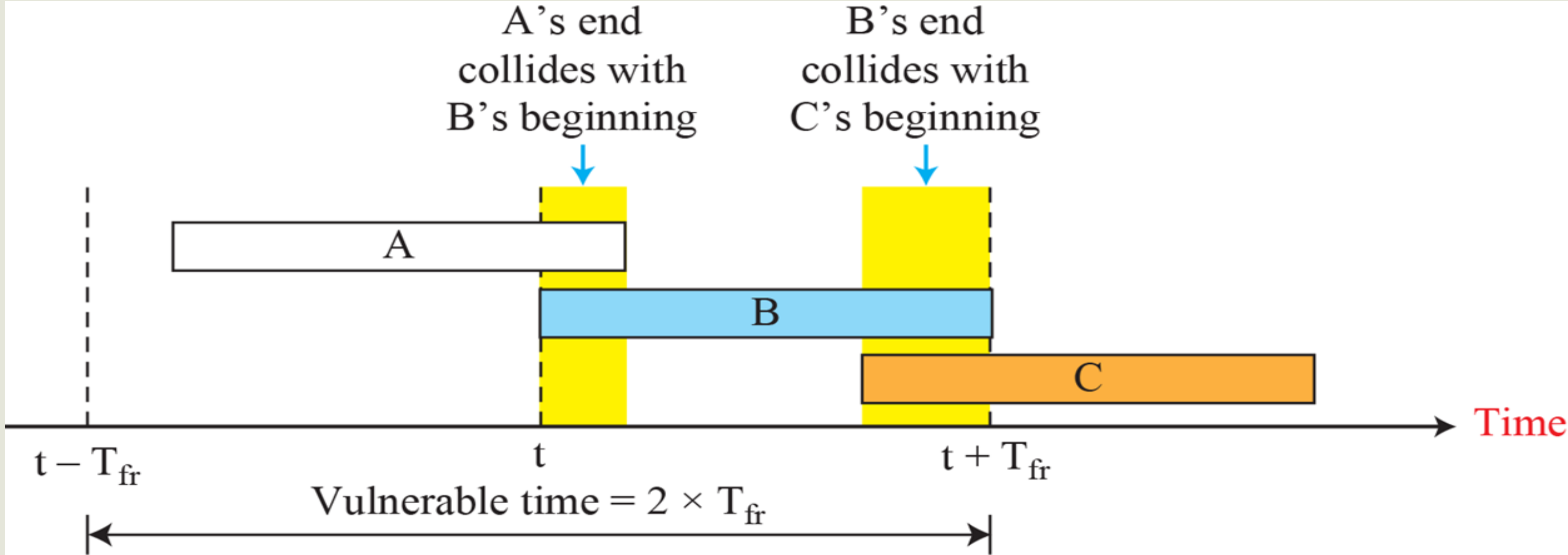
Example

- The stations on a wireless ALOHA network are a maximum of 600 km apart. If we assume that signals propagate at 3 × 108 m/s, we find Tp = (600 × 103) / (3 × 108) = 2 ms. For K = 2, the range of R is {0, 1, 2, 3}. This means that TB can be 0, 2, 4, or 6 ms, based on the outcome of the random variable R.

Vulnerable time for pure ALOHA protocol

- We assume that the stations send fixed-length frames with each frame taking T_{fr} seconds to send.
- Station B starts to send a frame at time t. Now 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.
- suppose that station C starts to send a frame before time t + Tfr, → collision between B and C.
- 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



Example

- 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 2×1 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 period (1 ms) that this station is sending.

Throughput

- Let G the average number of frames generated by the system during 1 frame transmission time.
- The average number of successfully transmitted frames for pure aloha is $s = g \times e^{-2g}$.
- The maximum throughput smax = 0.184, for G = 1/2. (We can find it by setting the derivative of S with respect to G to 0;)
- In other words, 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.
- We expect g = 1/2 to produce the maximum throughput because the vulnerable time is 2 times the frame transmission time.
- 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, or 1 frame per millisecond, then G = 12. In this case $S = G \times e^{-2G} =$ 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.

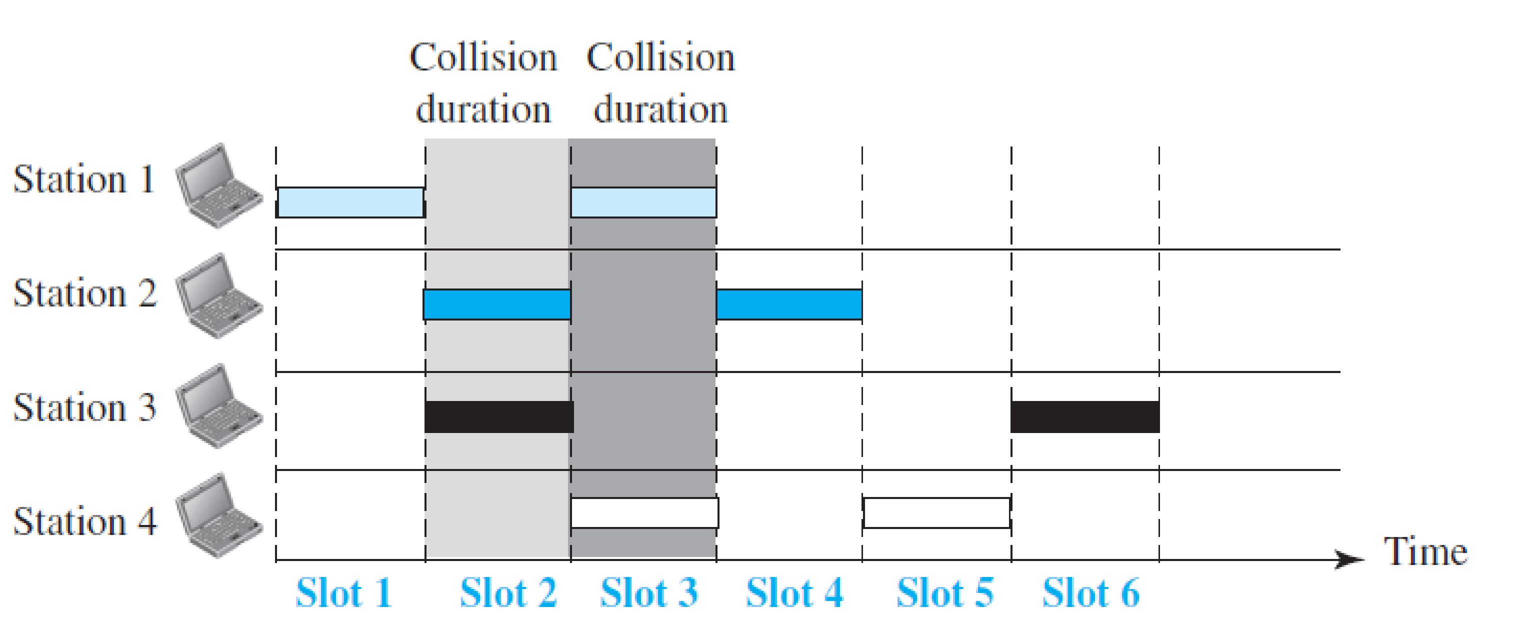
Problem

- b. If the system creates 500 frames per second, or 1/2 frames per millisecond, then $G = 1/2$. In this case $S = G \times e^{-2G} = 0.184$ (18.4 percent). This means that the throughput is $500 \times 0.184 = 92$ and that only 92 frames out of 500 will probably survive. Note that this is the maximum throughput case, percentage-wise.

- c. If the system creates 250 frames per second, or 1/4 frames per millisecond, then $G = 1/4$. In this case $S = G \times e^{-2G} = 0.152$ (15.2 percent). This means that the throughput is $250 \times 0.152 = 38$. Only 38 frames out of 250 will probably survive

Slotted ALOHA

- Pure ALOHA has a **vulnerable time of 2 × T_{fr}**. → 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 where we **divide the time into slots of T_{fr} seconds** and force the station to send **only at the beginning** of the time slot.

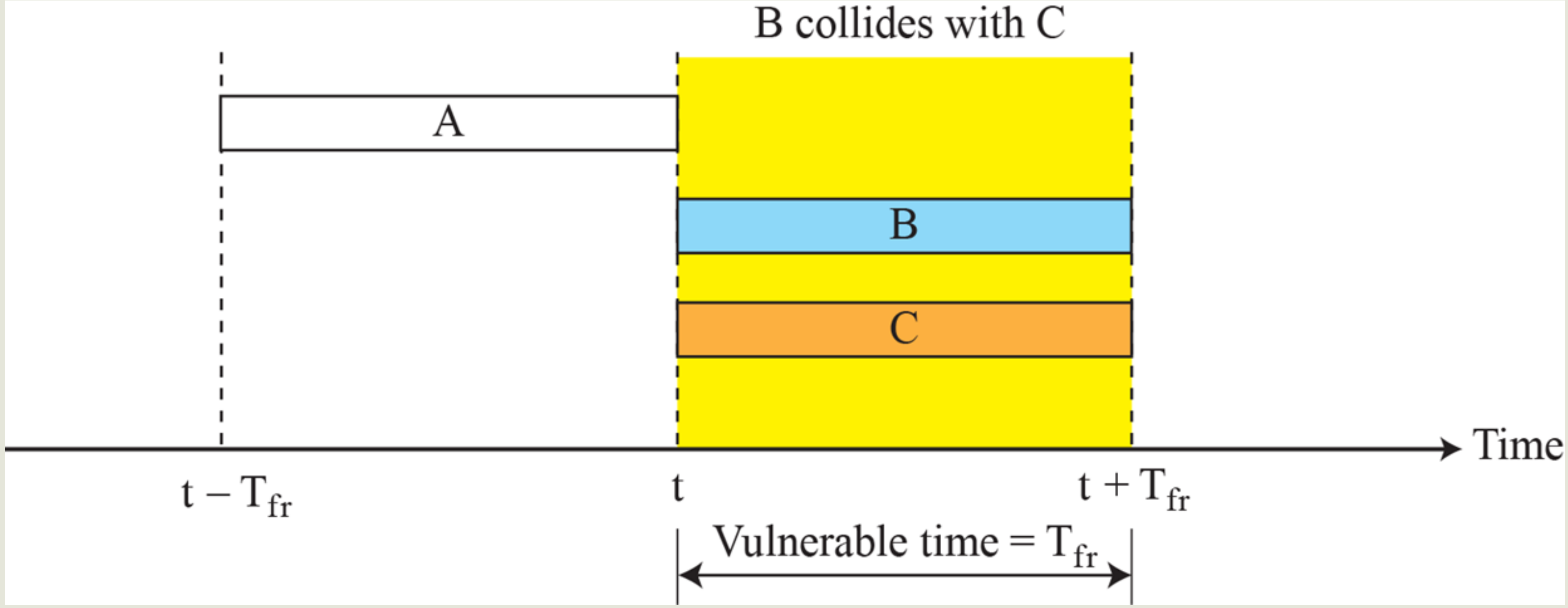


- Because a station is allowed to send only at the beginning of the synchronized time slot, if a station misses this moment, it **must wait until the beginning of the next time slot**.
- →the station which started at the beginning of this slot has already finished sending its frame.

Vulnerable time for slotted ALOHA protocol

- The vulnerable time is now reduced to one-half, equal to T_{fr}.

Slotted ALOHA 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.

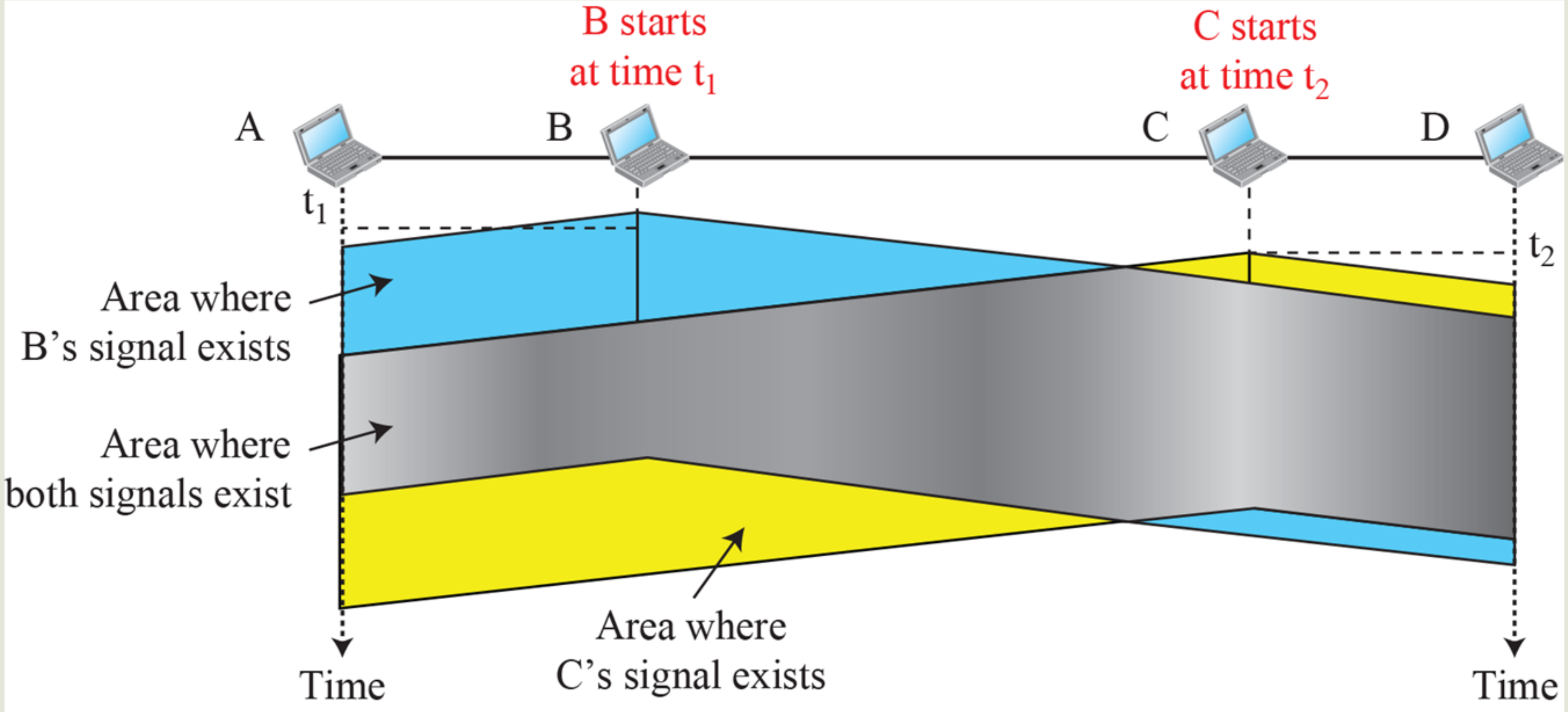
Problem

- In this case G is 12. So $S = G \times e^{-G} = 0.368$ (36.8 percent). This means that the throughput is $1000 \times 0.0368 = 368$ frames. Only 368 out of 1000 frames will probably survive. Note that this is the maximum throughput case, percentage-wise.
- Here G is 1/2. In this case $S = G \times e^{-G} = 0.303$ (30.3 percent). This means that the throughput is $500 \times 0.0303 = 1512$. Only 151 frames out of 500 will probably survive.
- Now G is 1/4. In this case $S = G \times e^{-G} = 0.195$ (19.5 percent). This means that the throughput is $250 \times 0.195 = 49$. 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 can reduce the possibility of collision because of propagation delay, but it cannot eliminate it.
- Stations are connected to a shared channel (usually a dedicated medium).
- 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.

Space/time model of a collision in CSMA



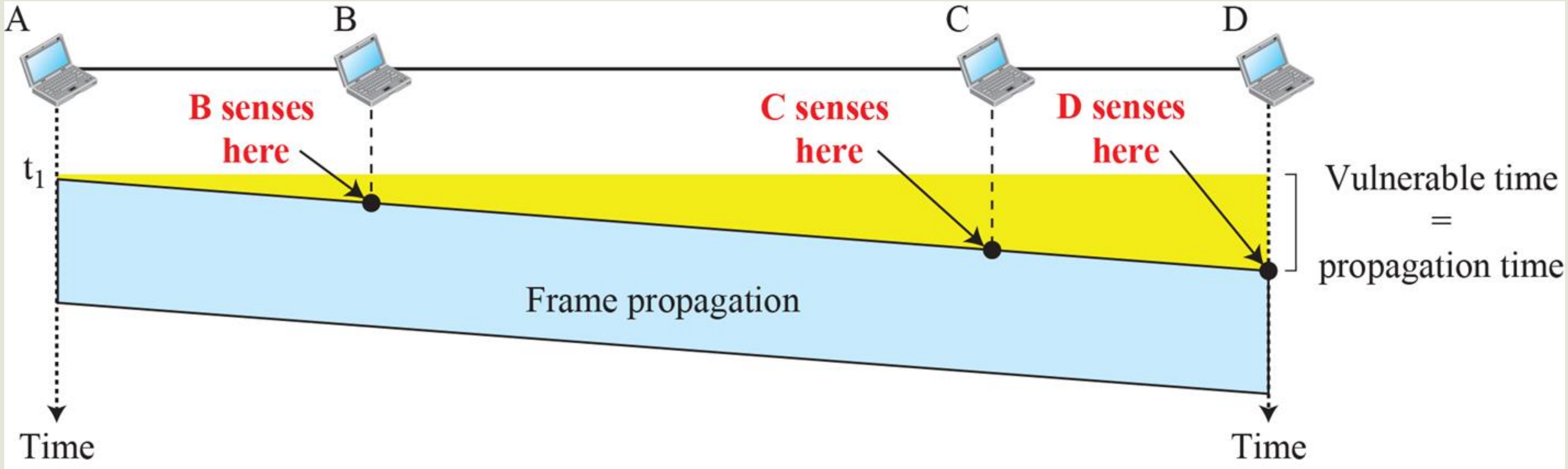
CSMA

- 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. →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.
- the worst case. The leftmost station, A, sends a frame at time t1, which reaches the rightmost station, D, at time t1 + T_p.
- The gray area shows the vulnerable area in time and space.

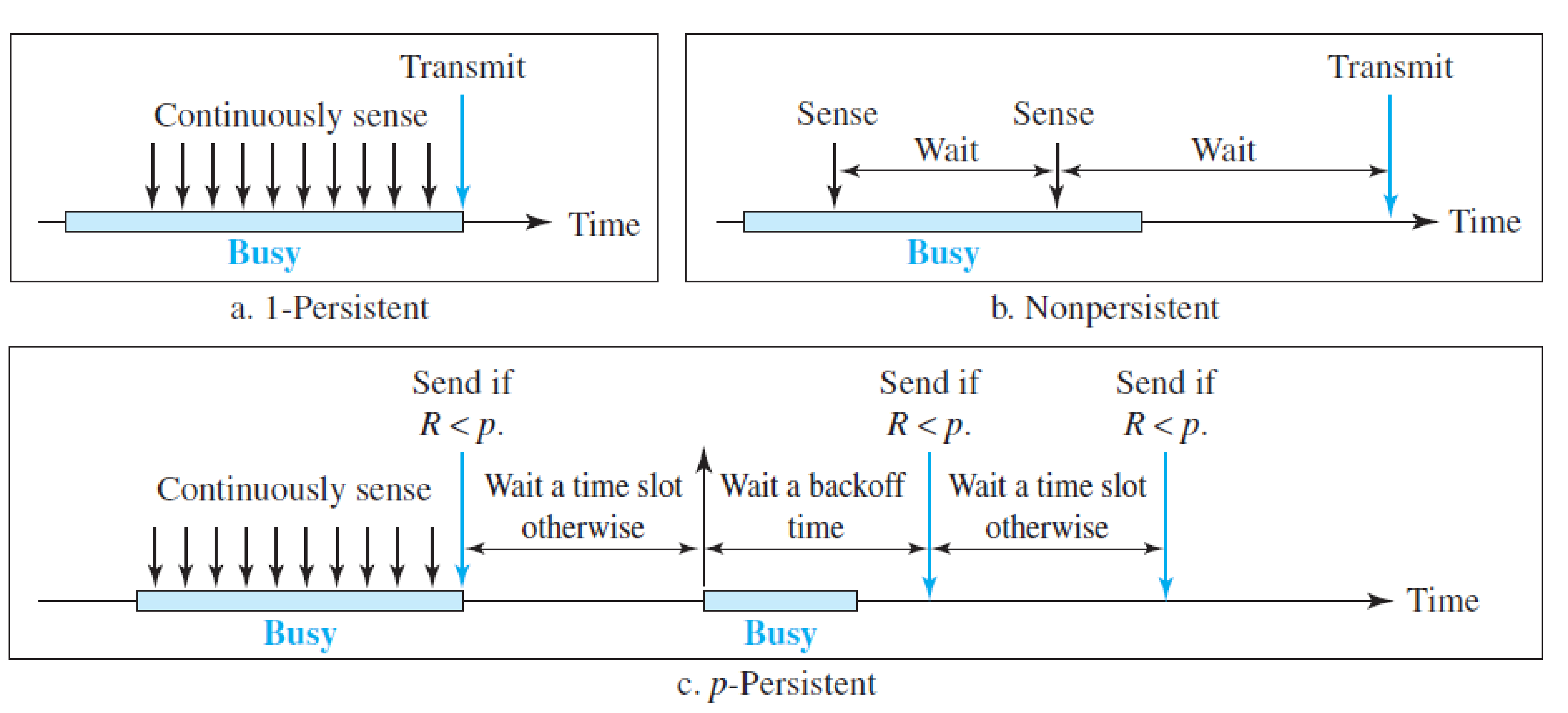
Vulnerable time in CSMA



Persistence Methods

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

Behavior of three persistence methods



1-Persistent, Nonpersistent, p-Persistent

1-Persistent

- simple and straightforward, after the station finds **the line idle**, **sends its frame immediately** - **highest chance of collision**.

Nonpersistent

- a station that has a frame to send senses the line. idle → sends immediately. **not idle** → **waits a random** amount of time and then senses again - **reduces the chance of collision** →reduces the efficiency of the network because the **medium remains idle** when there may be stations with frames to send.

p-Persistent

- Used if the channel has time slots with a slot duration equal to or greater than the maximum T_p ; combines the advantages of the other two strategies. It **reduces the chance of collision and improves efficiency.**

p-Persistent

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

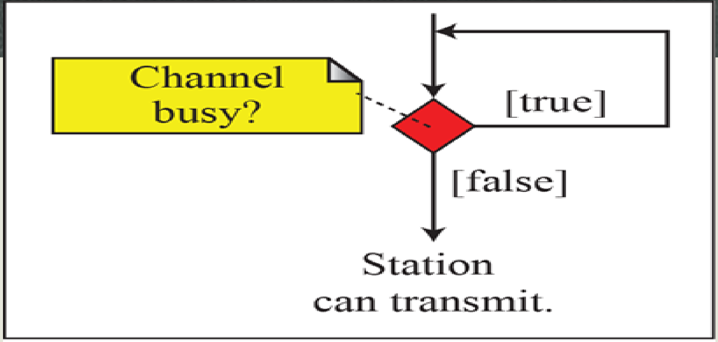
1. With probability p, the station sends its frame.

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

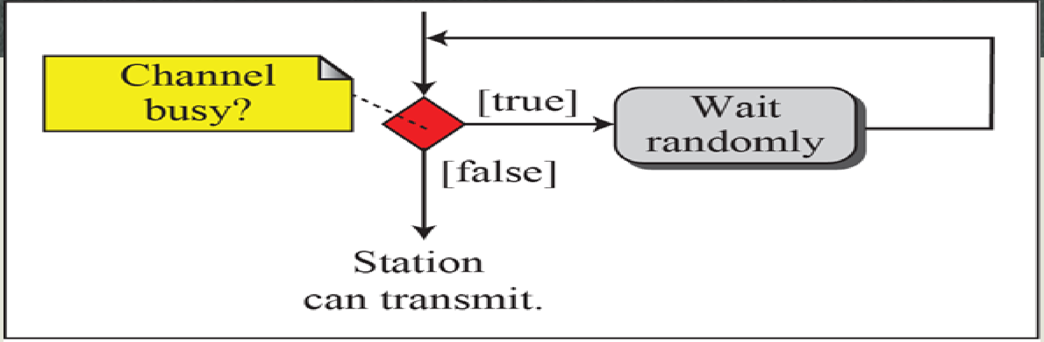
a. If the line is idle, it goes to step 1.

b. If the line is busy, it acts as though a collision has occurred and uses the backoff procedure.

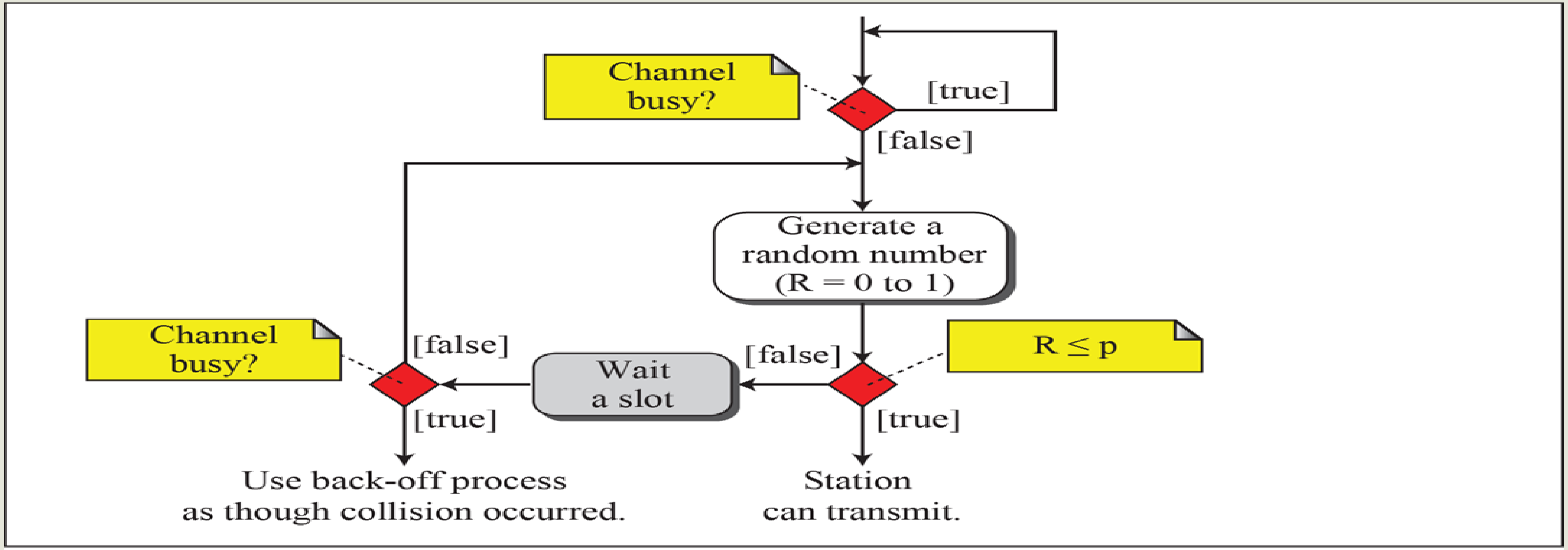
Flow diagram for three persistence methods



a. 1-persistent



b. Nonpersistent

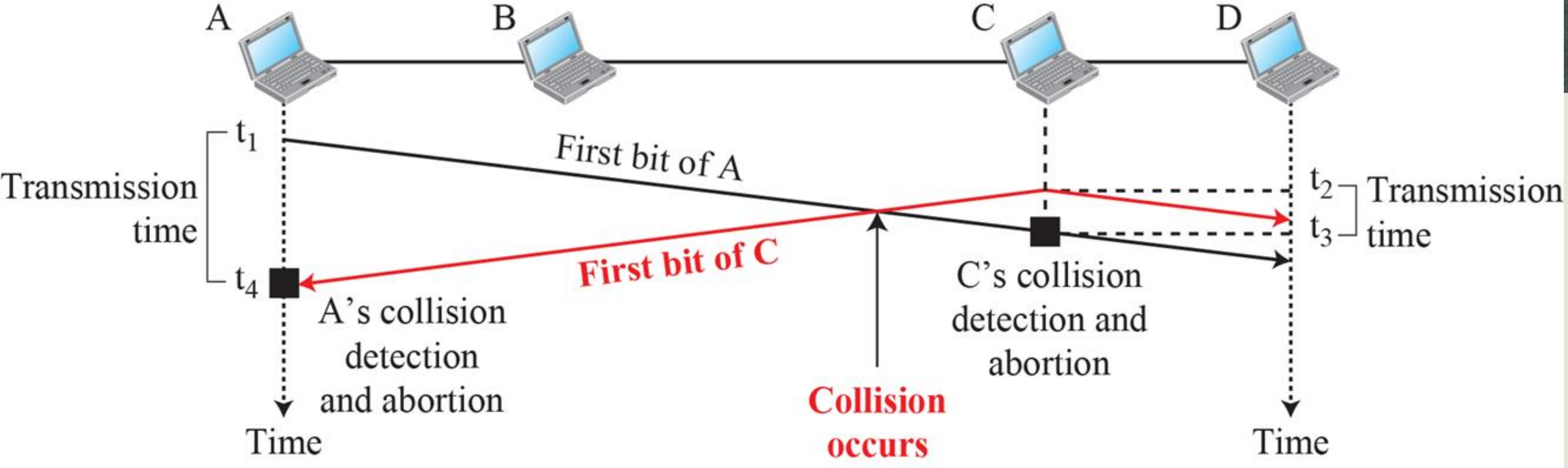


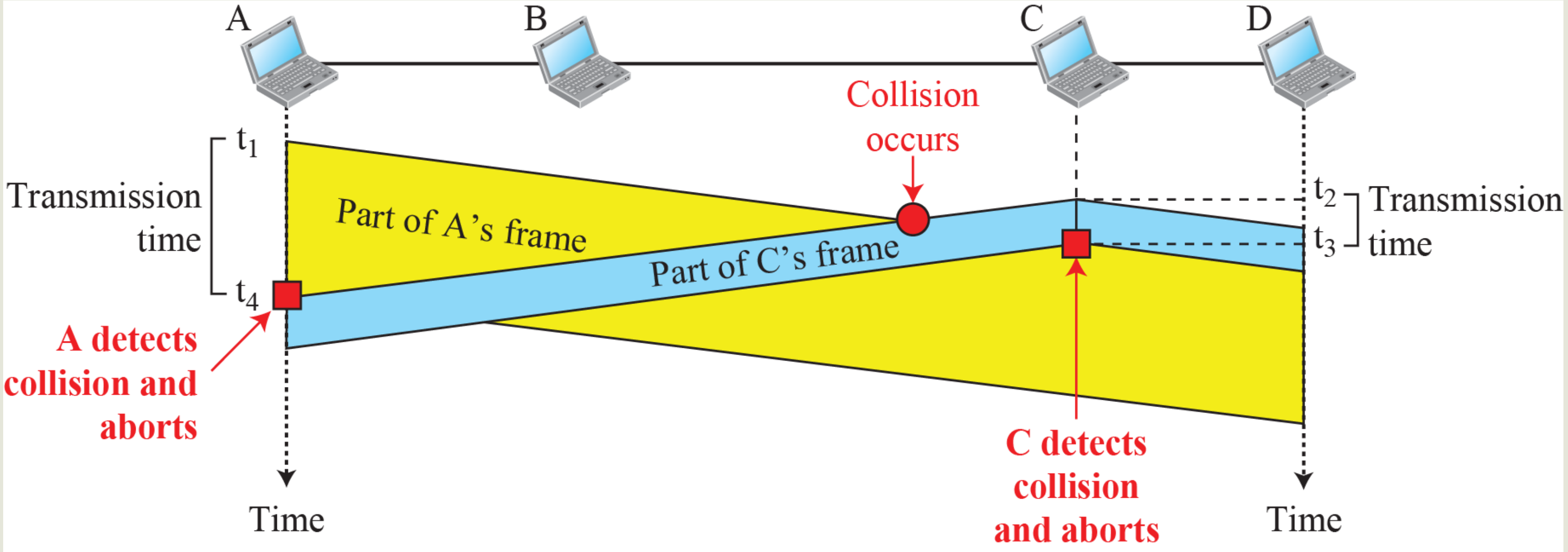
c. *p*-persistent

CSMA/CD

- A station monitors the medium after a frame to see it was successful. If so, the station is finished. Else will resent
- Each station continues to send bits in the frame until it detects the collision.
- At time $t1$, station A has executed its persistence procedure and starts sending the bits of its frame. At time $t2$, station C has not yet sensed the first bit sent by A.
- Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and to the right. The collision occurs sometime after time $t2$.
- Station C detects a collision at time $t3$ when it receives the first bit of A's frame. Station C immediately (or after a short time, but we assume immediately) aborts transmission.
- Station A detects collision at time $t4$ when it receives the first bit of C's frame; it also immediately aborts transmission.
- A transmits for the duration $t4 - t1$; C transmits for the duration $t3 - t2$.

Collision of the first bits 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.
- 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 2 T_p.

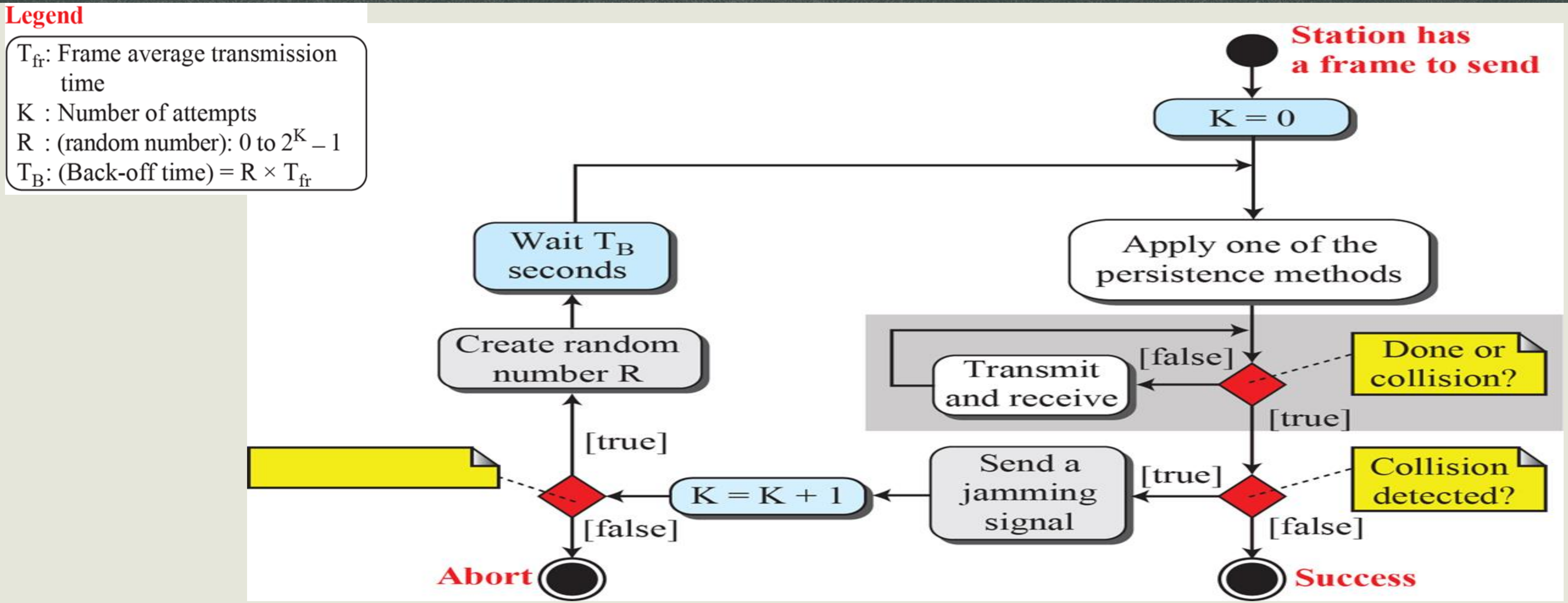
Problem

- A network using CSMA/CD has a bandwidth of 10 Mbps. If the maximum propagation time is 25.6 μs, what is the minimum size of the frame?

Solution

- The minimum frame transmission time is Tfr = 2 × Tp = 512.2 μs. This means, in the worst case, a station needs to transmit for a period of 512.2 μs to detect the collision.
- The minimum size of the frame is 10 Mbps × 512.2 μs = 512 bits or 64 bytes. This is actually the minimum size of the frame for Standard Ethernet

Flow diagram for the CSMA/CD



ALOHA Vs CSMA/CD

1. Need to sense the channel before start sending the frame by using one of the persistence processes (nonpersistent, 1-persistent, or p-persistent).

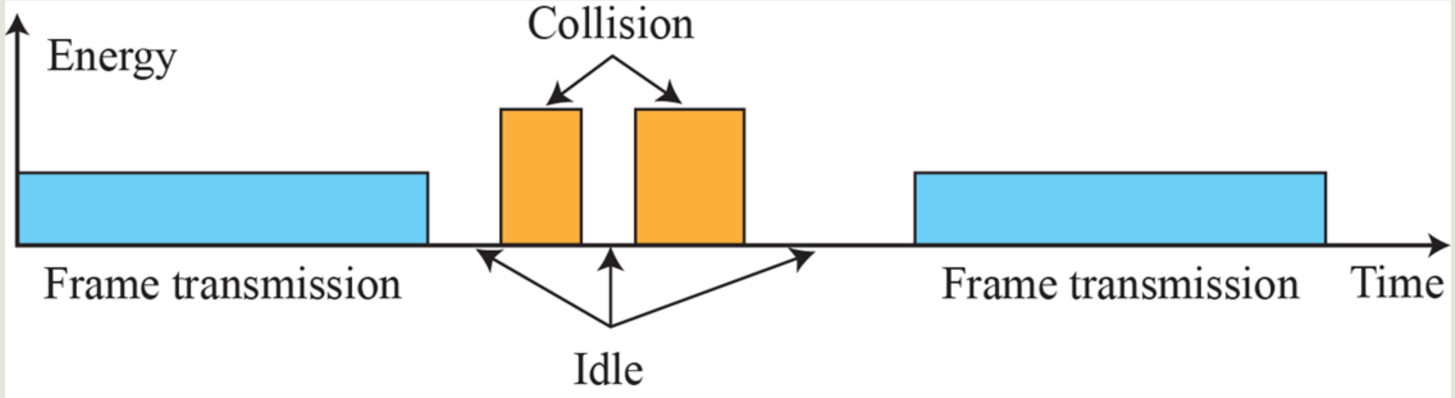
2. The frame transmission:

- In ALOHA, first transmit the entire frame and then wait for an acknowledgment.
- In CSMA/CD, transmission and collision detection are continuous processes. We do not send the entire frame and then look for a collision.
- The station transmits and receives continuously and simultaneously (using two different ports or a bidirectional port).
- We constantly monitor in order to detect one of two conditions: either transmission is finished or a collision is detected.
- Either event stops transmission. When we come 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 sending of a short jamming signal to make sure that all other stations become aware of the collision.

Energy Level

- Level of energy in a channel can have three values: zero, normal, and abnormal.
- At the zero level, the channel is idle.
- At the normal level, a station has successfully captured the channel and is sending its frame.
- At the abnormal level, there is a collision and the level of the energy is twice the normal level.
- 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.

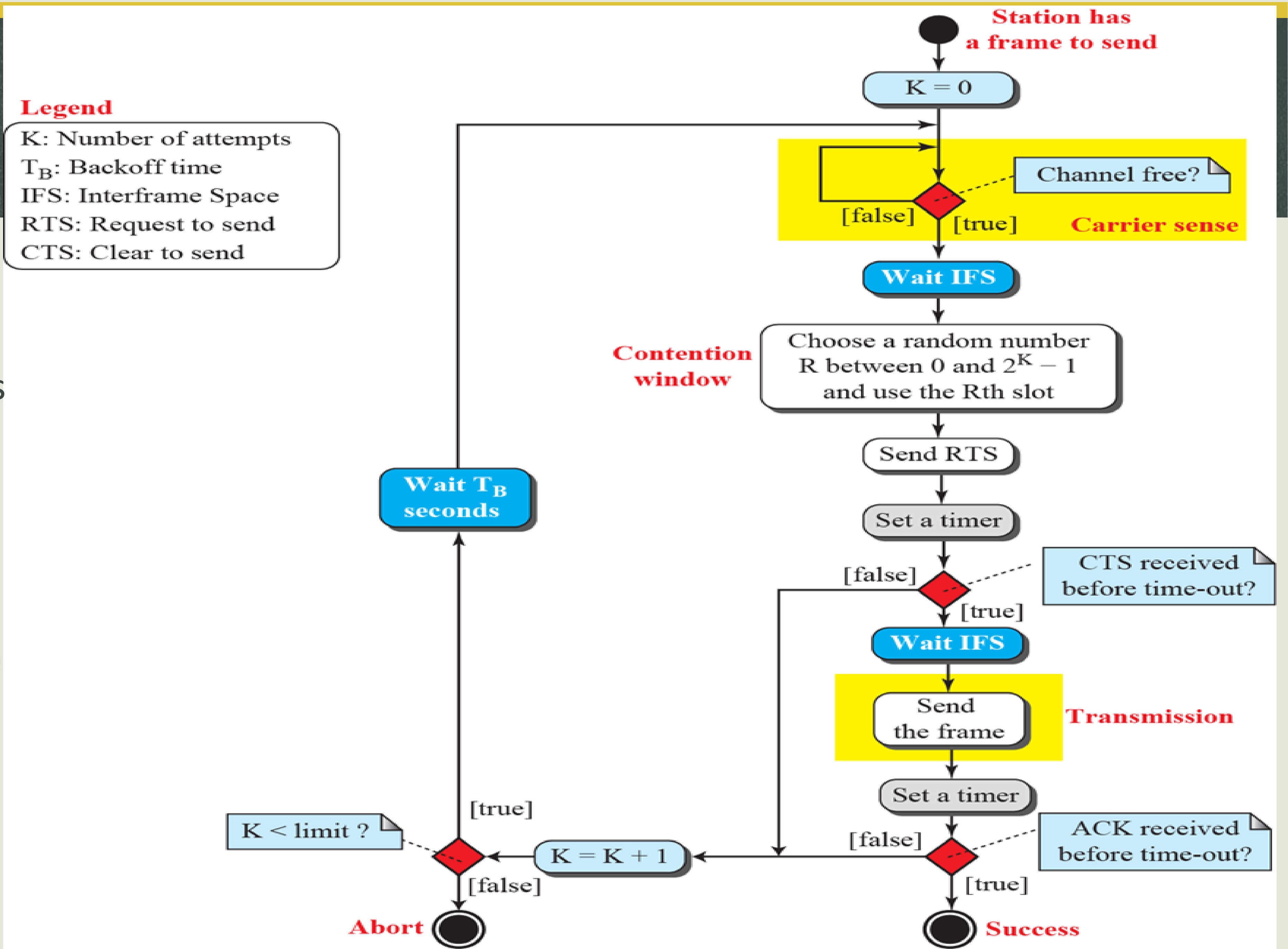


Throughput

- 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 nonpersistent 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 three strategies: **the interframe space, the contention window, and acknowledgments**



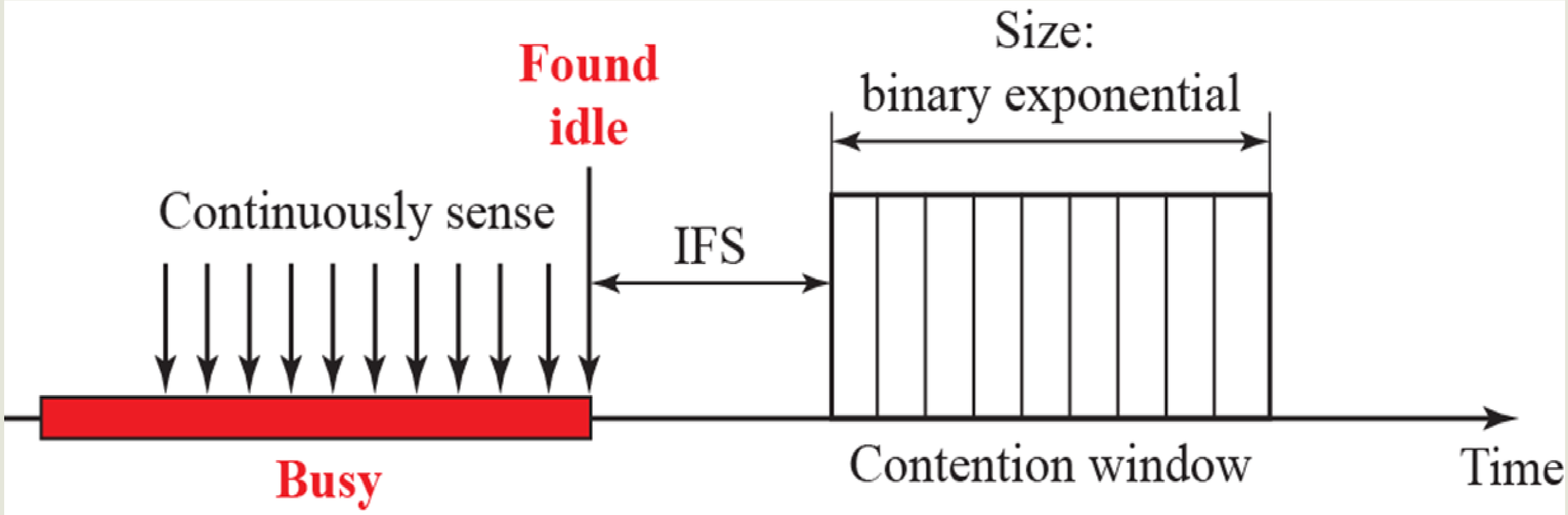
Interframe Space (IFS)

- 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. 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 backoff strategy → 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.
- Station needs to sense the channel after each time slot. 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**.

Contention window



Acknowledgment

- With all these precautions, there still may be a collision resulting in destroyed data. 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

- 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 backoff 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 interframe 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 interframe space (SIFS), the destination station sends a control frame, called the clear to send (CTS), to the source station.

Frame Exchange Time Line

- 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.
- Lack of collision in CSMA/CD is a kind of indication to the source that data have arrived.

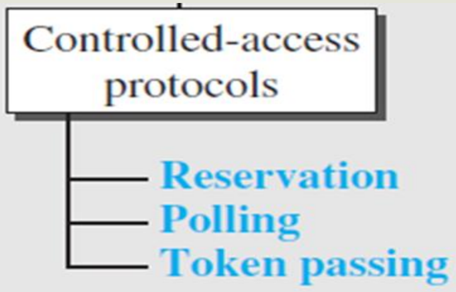
Network Allocation Vector

- 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.
- Each station, before sensing the physical medium to see if it is idle, first checks its NAV to see if it has expired.

Collision During Handshaking

- What happens 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→ may collide.
- Sender assumes there has been a collision if it has not received a CTS frame from the receiver.
- The backoff strategy is employed, and the sender tries again.
- CSMA/CA was mostly intended for use in **wireless networks**

CONTROLLED ACCESS: Reservation

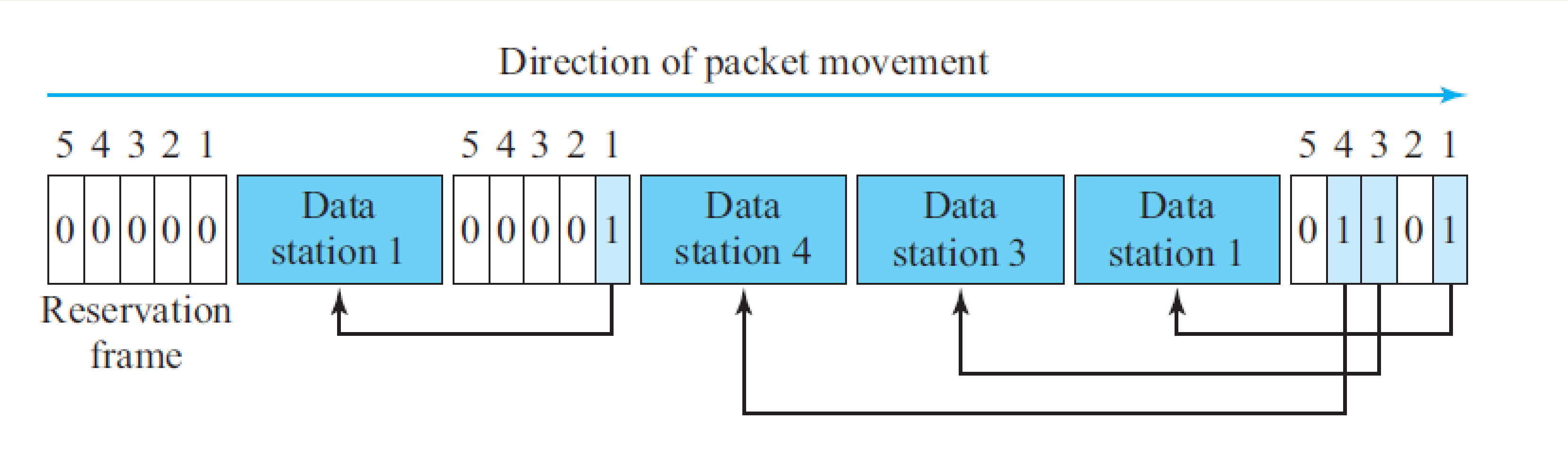


- Stations consult one another to find **which station has the right to send**. A station cannot send unless it has been authorized by other stations.

Reservation

- A station needs to make a reservation before sending data. Time is divided into intervals. In each interval, a reservation frame precedes the data frames sent in that interval.
- If there are N stations in the system, there are exactly N reservation minislots in the reservation frame. Each minislot belongs to a station.
- When a station needs to send a data frame, it makes a reservation in its own minislot. The stations that have made reservations can send their data frames after the reservation frame.
- In the first interval, only stations 1, 3, and 4 have made reservations. In the second interval, only station 1 has made a reservation.

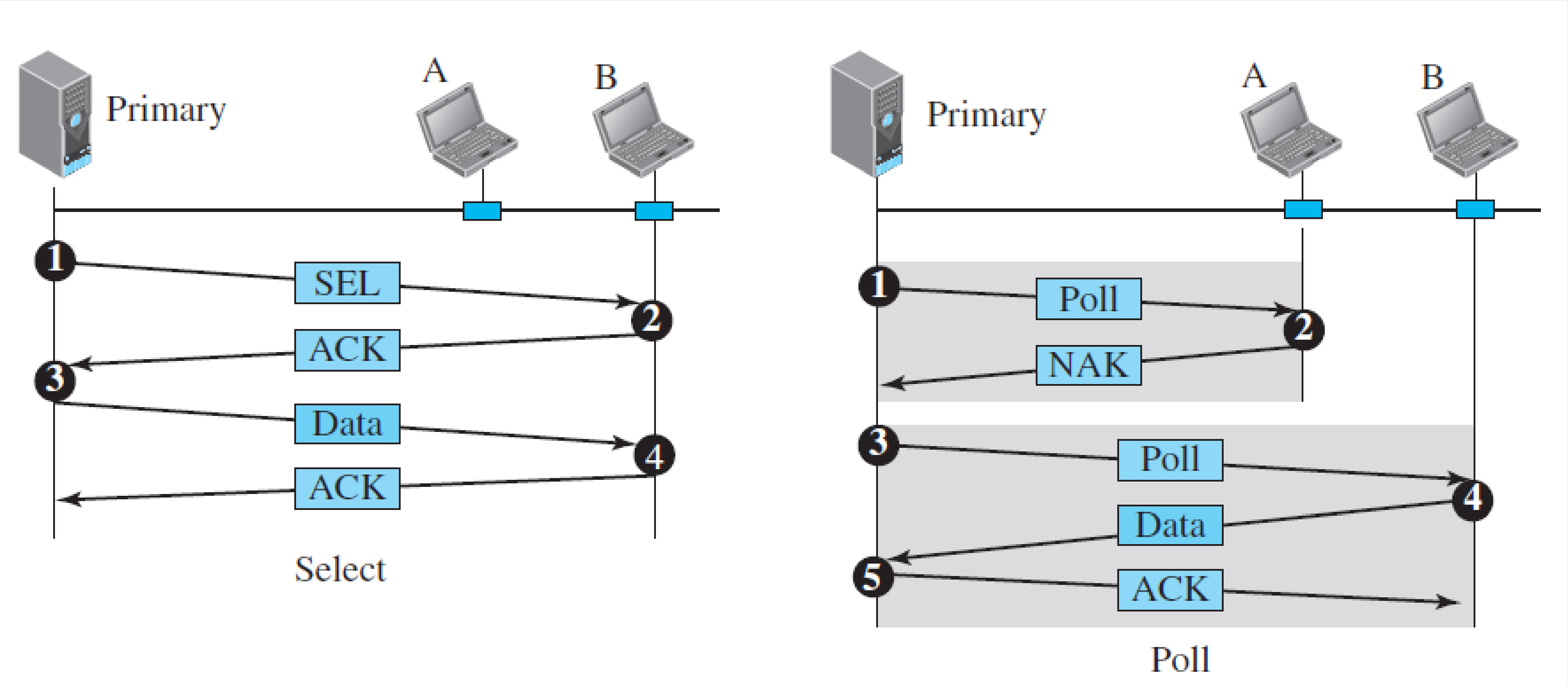
Reservation access method



Polling

- Polling works with topologies :**a primary station and the others are secondary stations**.
- All data exchanges must be made through the primary device even when the ultimate destination is a secondary device.
- The primary device controls the link; the secondary devices follow its instructions. It is up to the **primary device to determine which device is allowed to use the channel** at a given time.
- The primary device, therefore, is always the **initiator** of a session.
- This method uses poll and select functions to prevent collisions.
- Drawback : if the **primary station fails, the system goes down**.

Poll And Select



Select

- The select function is used whenever the primary device has something to send.
- link is available If the primary is neither sending nor receiving data
- the primary must alert the secondary to the upcoming transmission and wait for an acknowledgment of the secondary's ready status.
- Before sending data, the primary creates and transmits a select (SEL) frame, one field of which includes the address of the intended secondary.

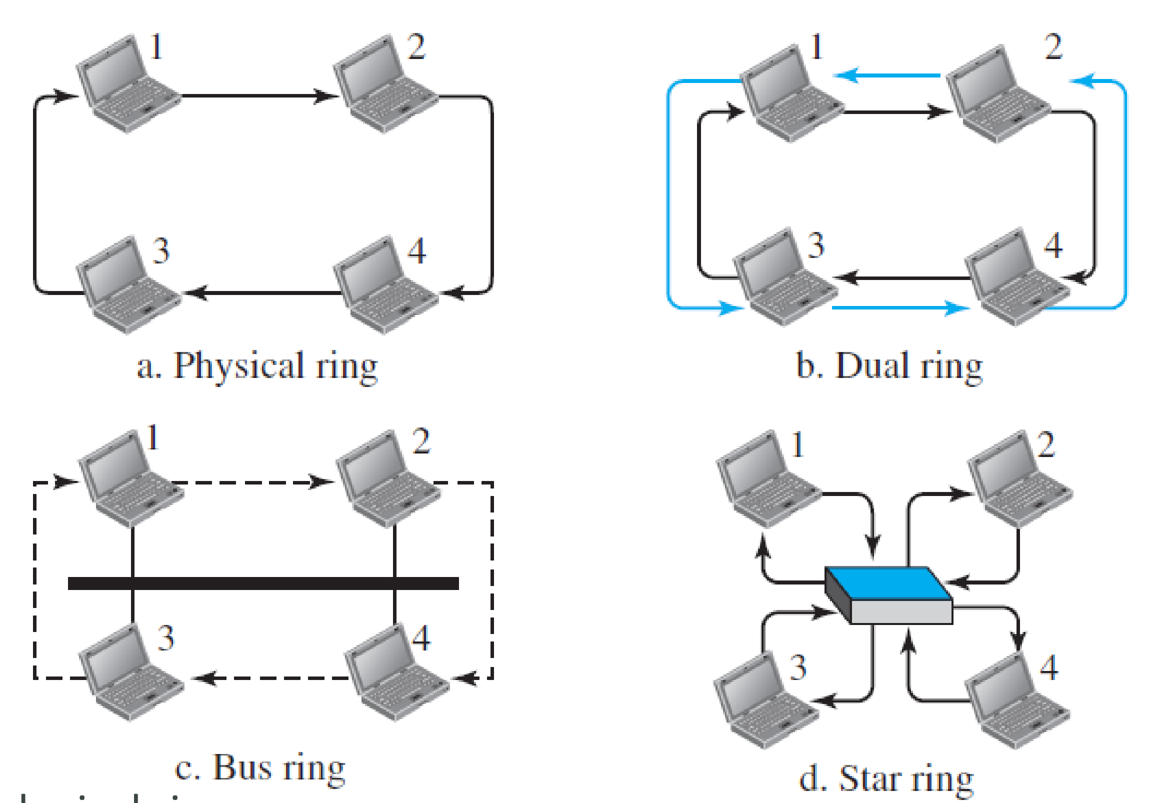
Poll

- used by the primary device. When the primary is ready to receive data, it must ask (poll) each device.
- When the first secondary is approached, it responds either with a NAK frame or positive AK
- When the response is positive (a data frame), the primary reads the frame and returns an acknowledgment (ACK frame), verifying its receipt

Token Passing

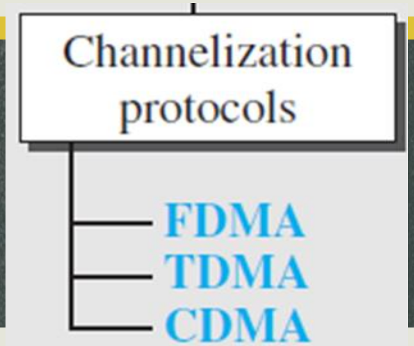
- The stations in a network are organized in a logical ring with a *predecessor* and a *successor*
- The current station is the one that is accessing the channel now.
- The right will be passed to the successor when the current station has no more data to send
- A special packet called a token circulates through the ring.
- When a station has some data to send, it waits until it receives the token from its predecessor
- When the station has no more data to send, it releases the token, passing it to the next logical station in the ring.
- When a station receives the token and has no data to send, it just passes the data to the next station.

Logical Ring



logical ring

CHANNELIZATION

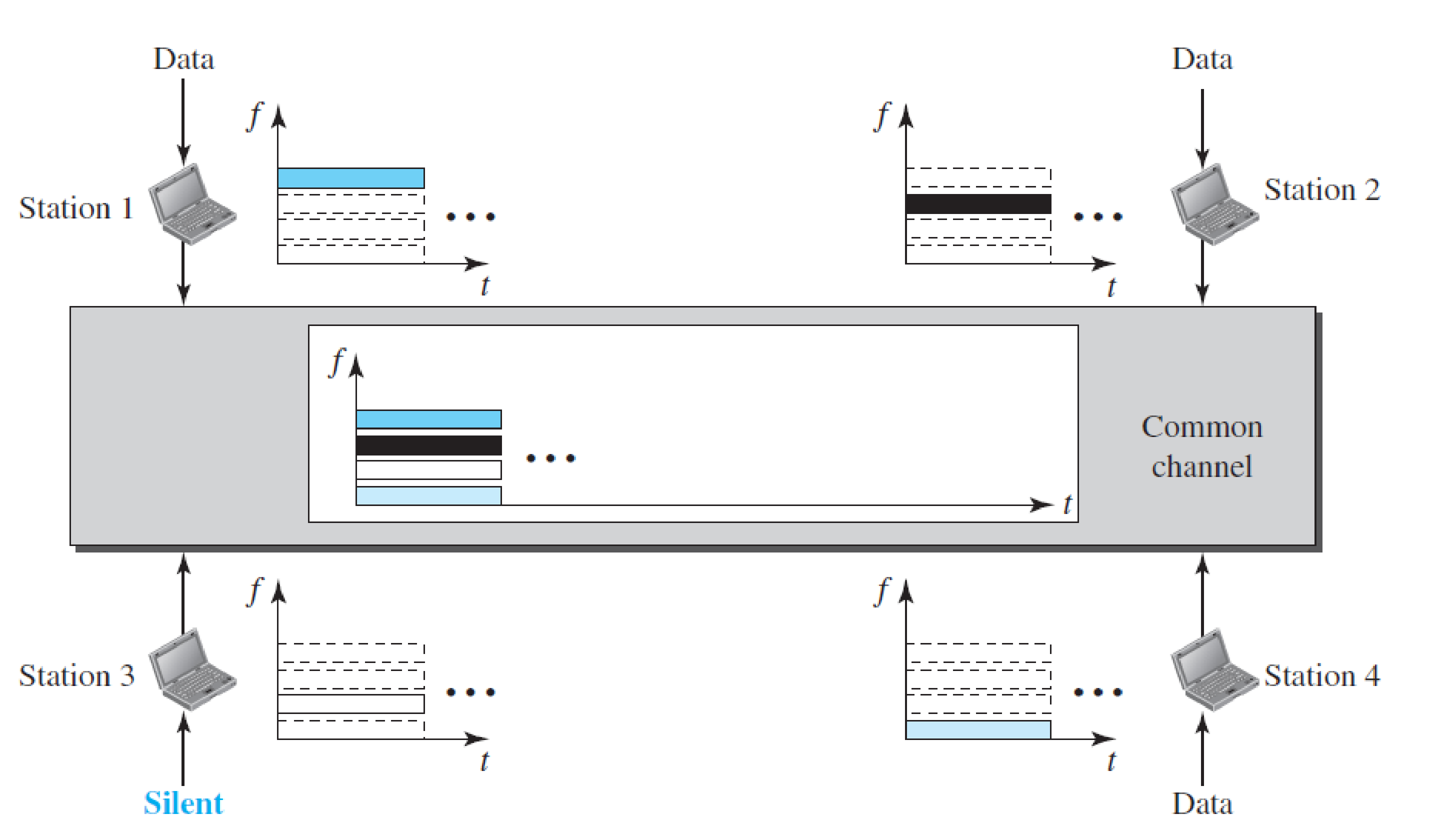


- Channelization (or channel partition, as it is sometimes called) is a multiple-access method in which the available bandwidth of a link is shared in time, frequency, or through code, among different stations

FDMA

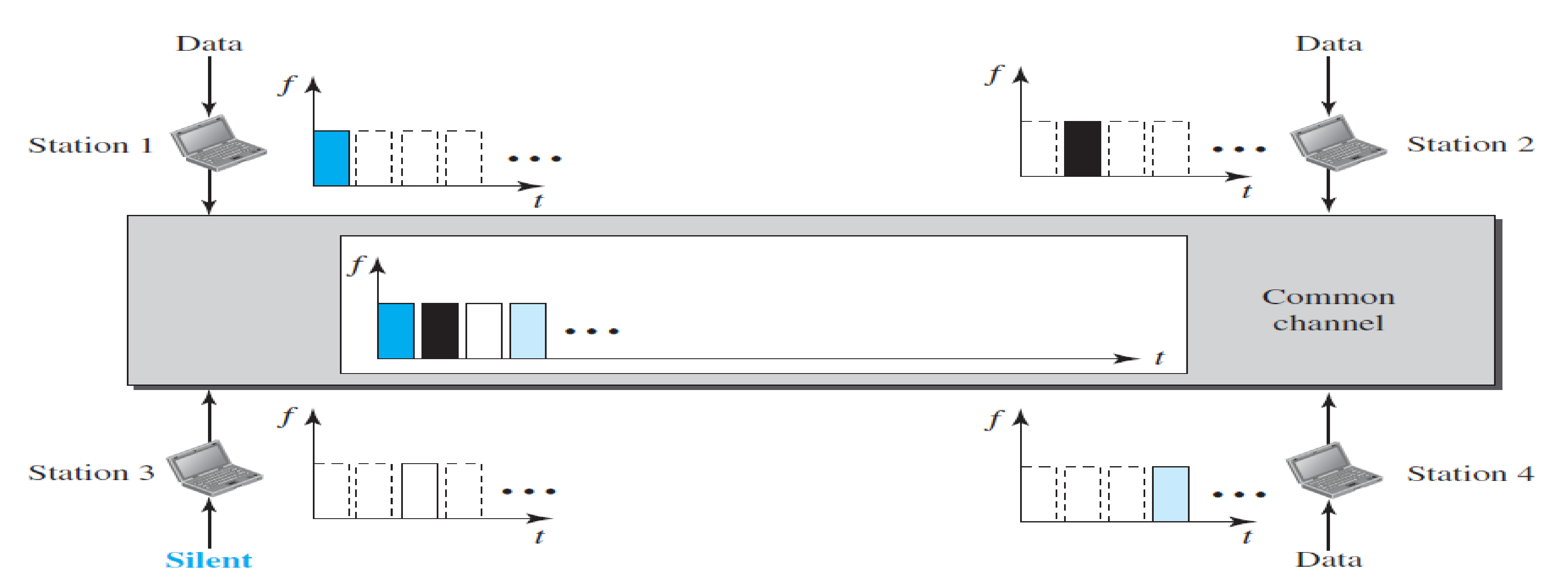
- the available bandwidth is divided into frequency bands. Each station is allocated a band to send its data.
- Each station also uses a bandpass filter to confine the transmitter frequencies; the allocated bands are separated from one another **by small guard bands**
- **stream data (a continuous flow of data that may not be packetized)**

Frequency-division multiple access (FDMA)



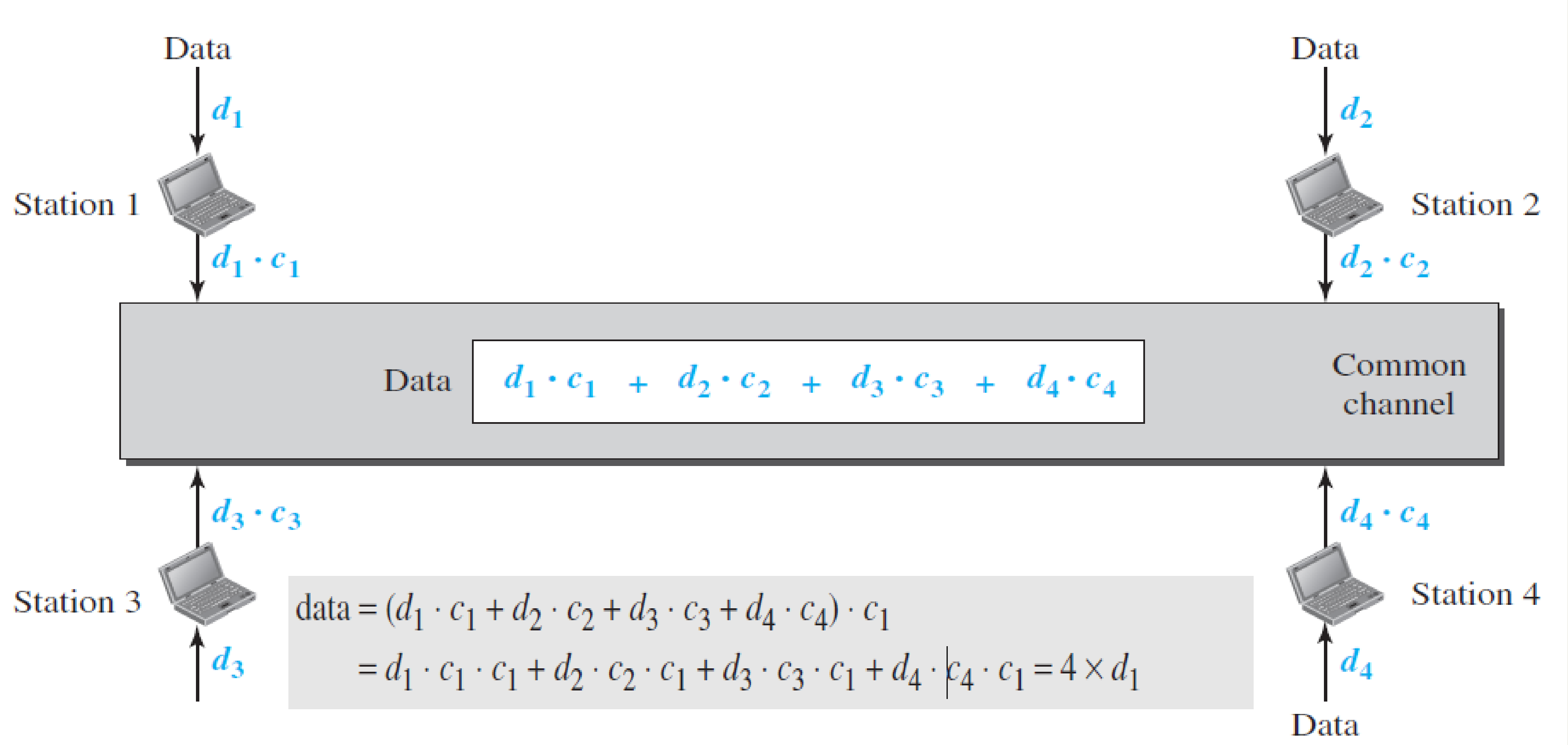
TDMA

- the stations share the bandwidth of the channel in time. Each station is allocated a time slot during which it can send data.
- Each station transmits its data in its assigned time slot. insert **guard times**.



CDMA

- CDMA differs from FDMA in that only one channel occupies the entire bandwidth of the link. It differs from TDMA in that all stations can send data simultaneously; there is no timesharing.



Chips

- CDMA is based on coding theory. Each station is assigned a code, which is a sequence of numbers called chips; orthogonal sequences

C_1	C_2	C_3	C_4
[+1 +1 +1 +1]	[+1 −1 +1 −1]	[+1 +1 −1 −1]	[+1 −1 −1 +1]

- 1. Each sequence is made of N elements, where N is the number of stations.
- 2. If we multiply a sequence by a number, every element in the sequence is multiplied by that element. This is called multiplication of a sequence by a scalar.

$$2 \bullet [+1\ +1\ -1\ -1] = [+2\ +2\ -2\ -2]$$

Chips

- 3. If we multiply two equal sequences, element by element, and add the results, we get N, where N is the number of elements in each sequence. This is called the inner product of two equal sequences.

$$[+1\ +1\ -1\ -1] \bullet [+1\ +1\ -1\ -1] = 1 + 1 + 1 + 1 = 4$$

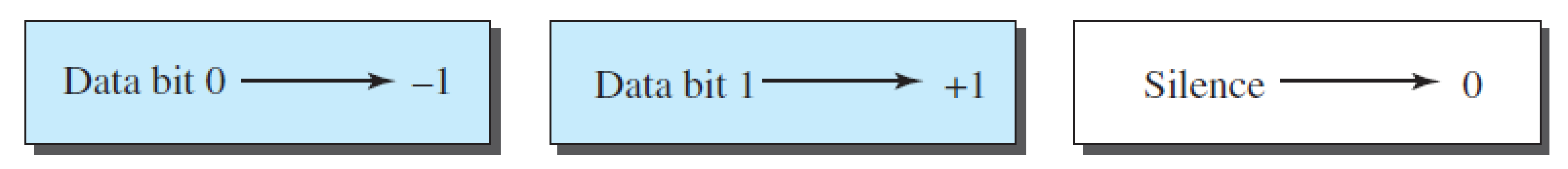
- 4. If we multiply two different sequences, element by element, and add the results, we get 0. This is called the inner product of two different sequences.

$$[+1\ +1\ -1\ -1] \bullet [+1\ +1\ +1\ +1] = 1 + 1 - 1 - 1 = 0$$

- **5.** Adding two sequences means adding the corresponding elements. The result is another sequence.

$$[+1\ +1\ -1\ -1] + [+1\ +1\ +1\ +1] = [+2\ +2\ \ 0\ \ 0]$$

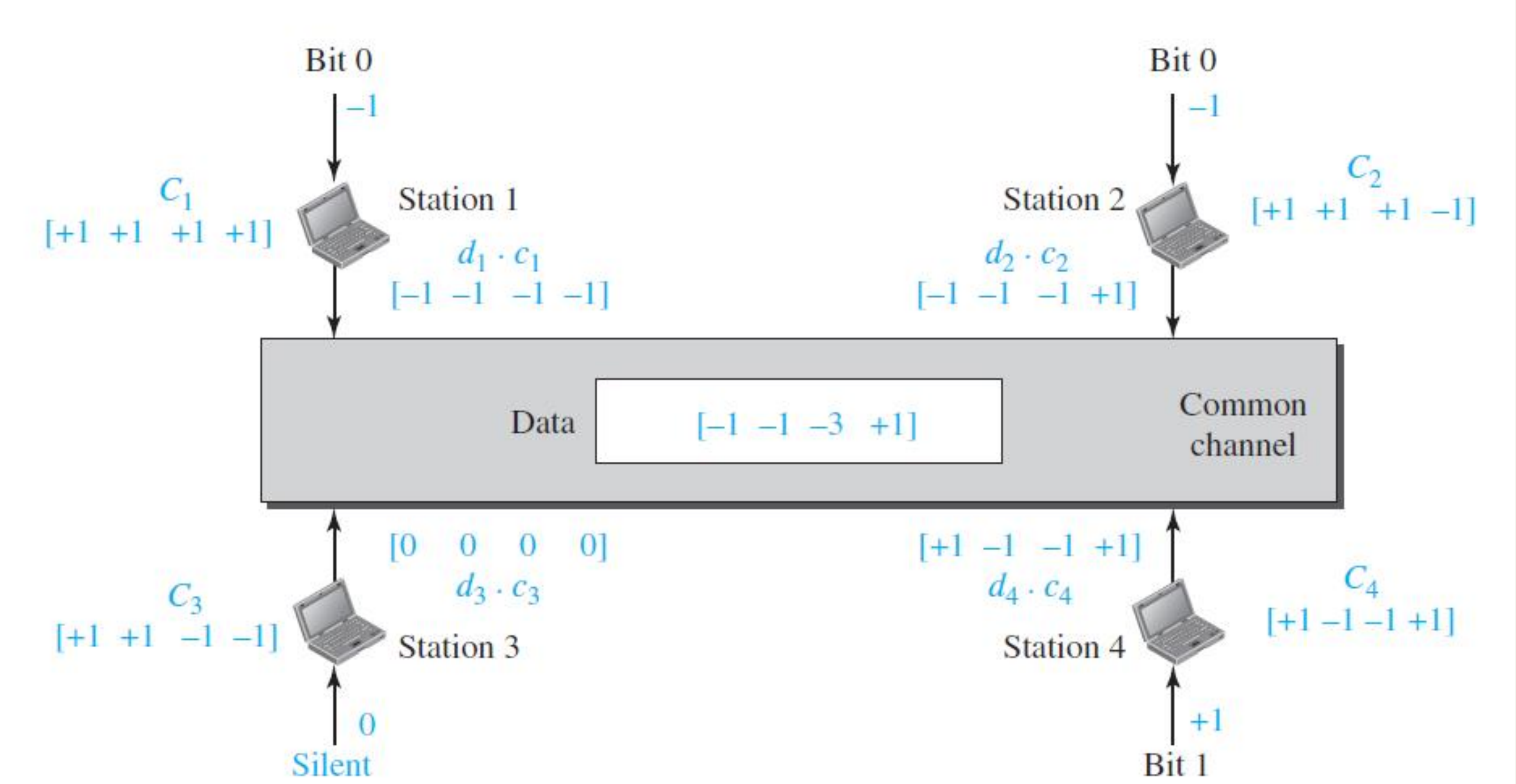
Data Representation

- 

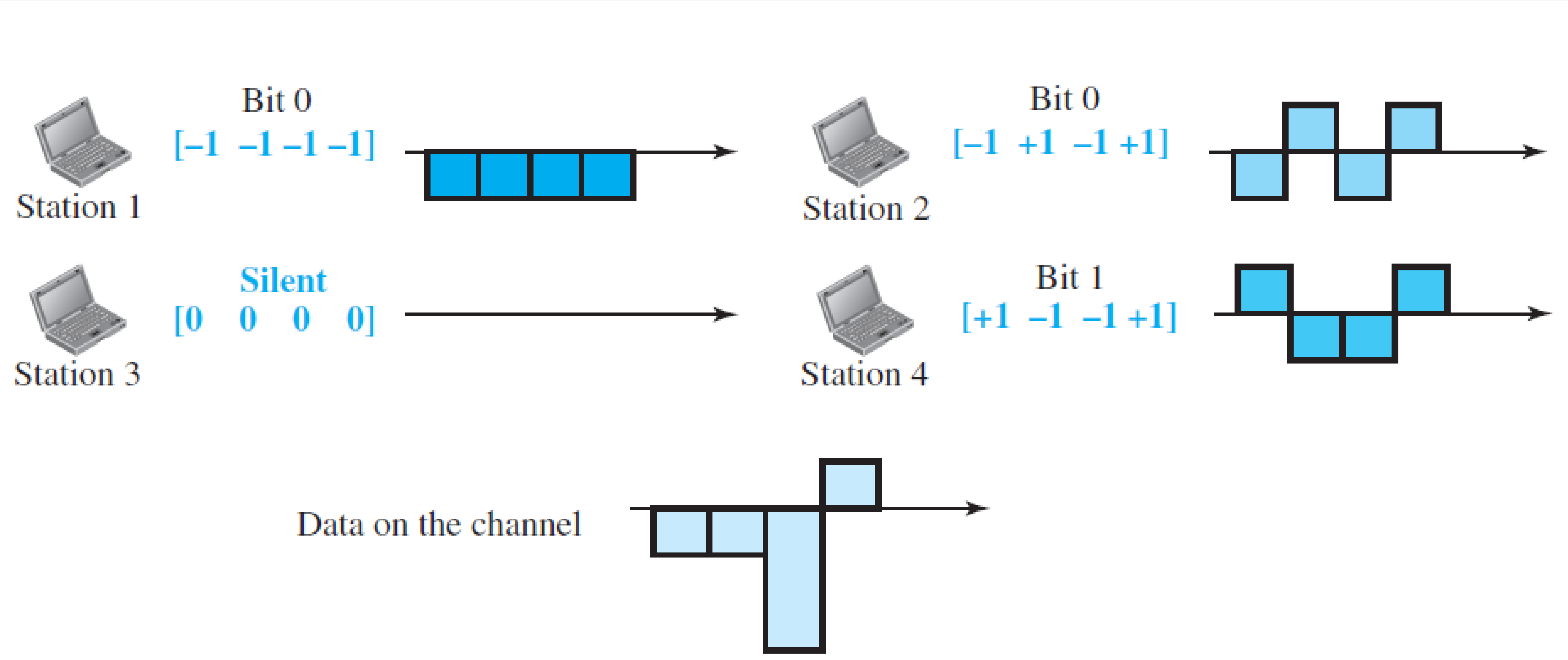
Encoding and Decoding

- assume that stations 1 and 2 are sending a 0 bit and channel 4 is sending a 1 bit. Station 3 is silent. The data at the sender site are translated to −1, −1, 0, and +1.
- Each station multiplies the corresponding number by its chip (its orthogonal sequence), which is unique for each station.

Sharing channel in CDMA



Digital signal created by four stations in CDMA



Sequence Generation

- To generate chip sequences, use a Walsh table, which is a two-dimensional table with an equal number of rows and columns
- In the Walsh table, each row is a sequence of chips. W1 for a one-chip sequence has one row and one column. We can choose −1 or +1 for the chip for this trivial table (we chose +1).
- According to Walsh, if we know the table for N sequences W_N, we can create the table for 2N sequences W_{2N},
- The W_N with the overbar W_N stands for the complement of W_N, where each +1 is changed to −1 and vice versa.
- After W_2 is generated, W4 can be made of four W2s, with the last one the complement of W_2.
- Of course, W_8 is composed of four W_{4s}, and so on.
- Note that after W_N is made, each station is assigned a chip corresponding to a row.

General rule and examples of creating Walsh tables

$$W_1 = \begin{bmatrix} +1 \end{bmatrix} \quad W_{2N} = \begin{bmatrix} W_N & W_N \\ W_N & \overline{W_N} \end{bmatrix}$$

a. Two basic rules

$$W_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix} \quad W_4 = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & -1 & +1 \end{bmatrix}$$

b. Generation of W_2 and W_4

- The number of sequences, N, needs to be a power of 2.
- In other words, we need to have $N = 2^m$.

Summary

- random access protocols, controlled access protocols, and channelization protocols
- Random access :ALOHA , CSMA , CSMA/CD , CSMA/CA
- Controlled access: reservation, Polling, Token passing
- Channelization : FDMA, TDMA, and CDMA

Test your Understanding

- Which of the following is a random-access protocol?

  a. CSMA/CD **b.** Polling **c.** TDMA

- Which of the following is a controlled-access protocol?

  a. Token-passing **b.** Polling **c.** FDMA

- Which of the following is a channelization protocol?

  a. ALOHA b. Token-passing c. CDMA

- What is the purpose of NAV in CSMA/CA?