Data Link Control

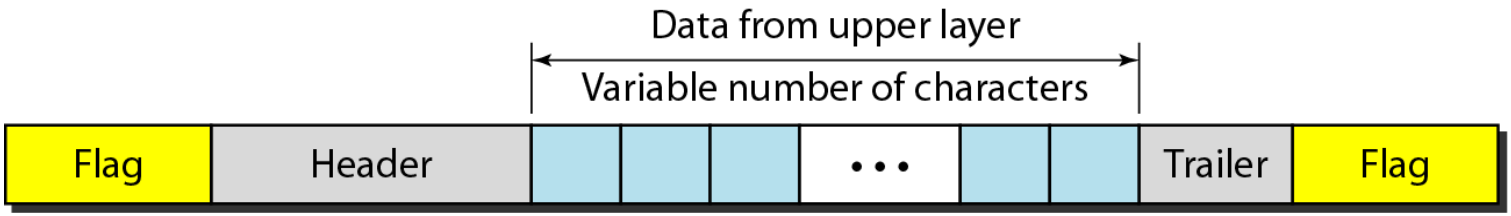
FRAMING

The data link layer needs to pack bits into frames, so that each frame is distinguishable from another. Our postal system practices a type of framing. The simple act of inserting a letter into an envelope separates one piece of information from another; the envelope serves as the delimiter.

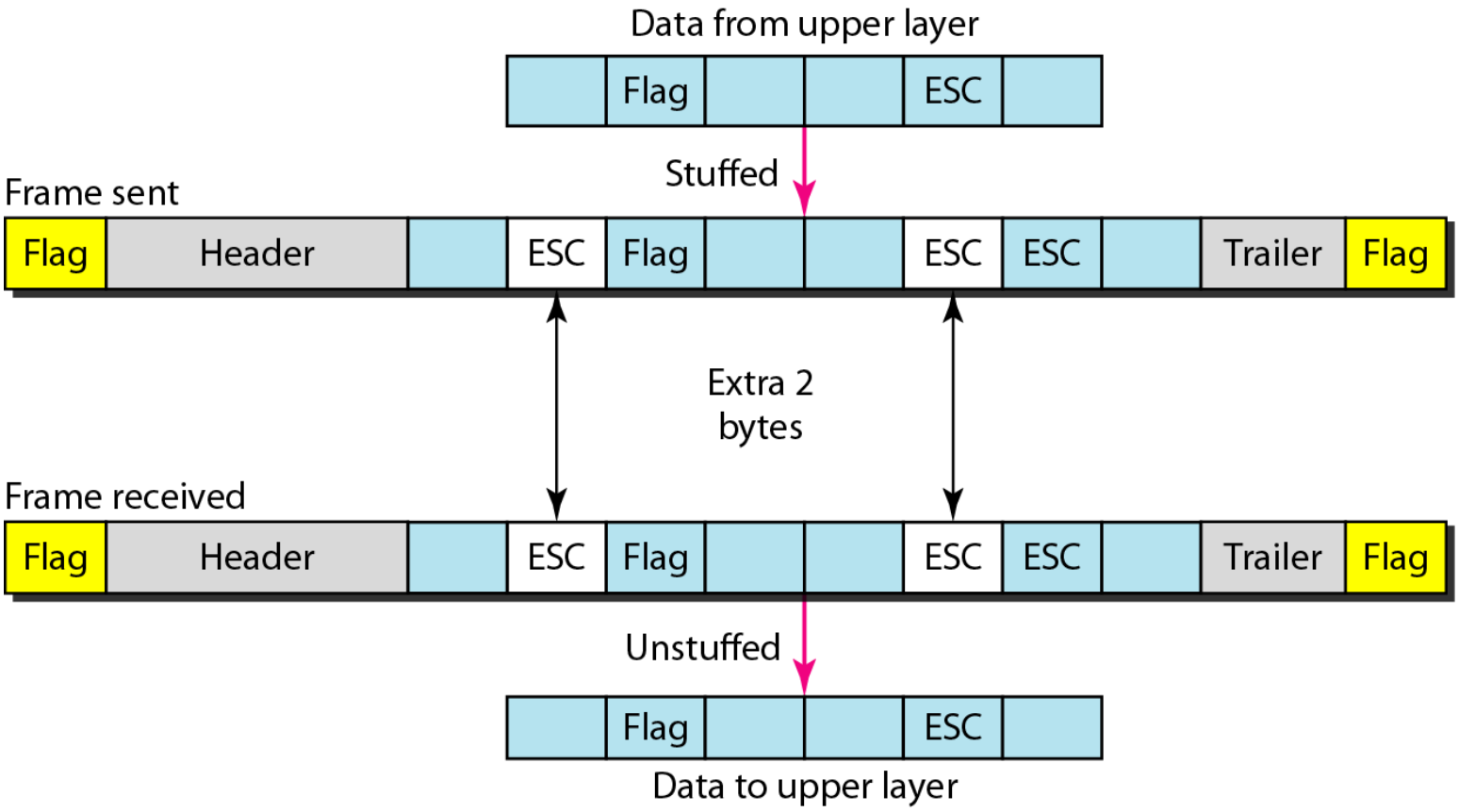
Topics discussed in this section:

Fixed-Size Framing
Variable-Size Framing

A frame in a character-oriented protocol



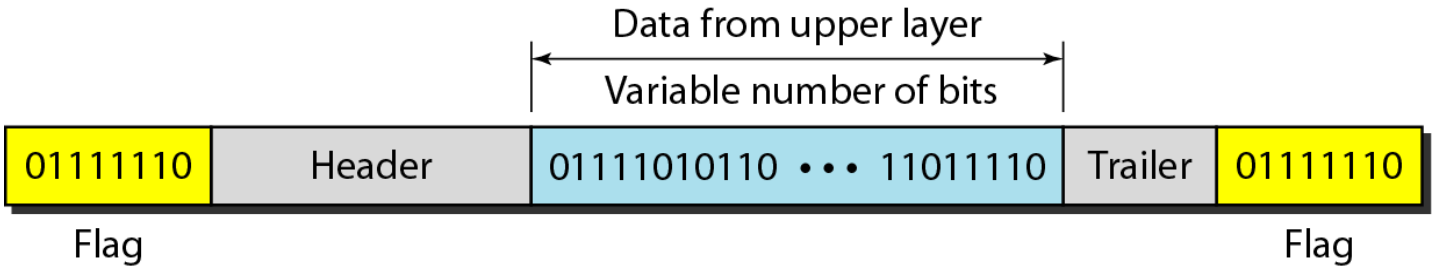
Byte stuffing and unstuffing



Note

Byte stuffing is the process of adding 1 extra byte whenever there is a flag or escape character in the text.

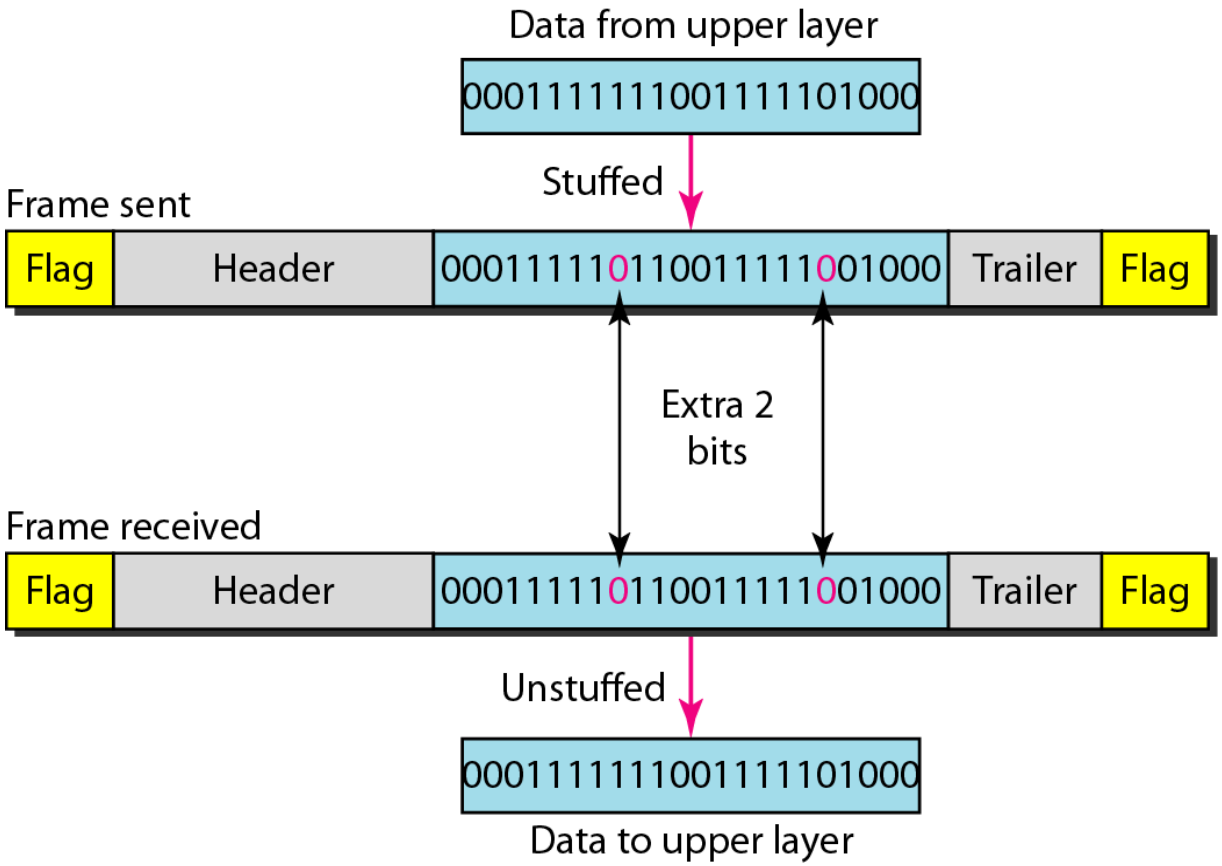
A frame in a bit-oriented protocol



Note

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.

Bit stuffing and unstuffing



FLOW AND ERROR CONTROL

The most important responsibilities of the data link layer are flow control and error control. Collectively, these functions are known as data link control.

Topics discussed in this section:

Flow Control
Error Control

Note

Flow control refers to a set of procedures used to restrict the amount of data that the sender can send before waiting for acknowledgment.

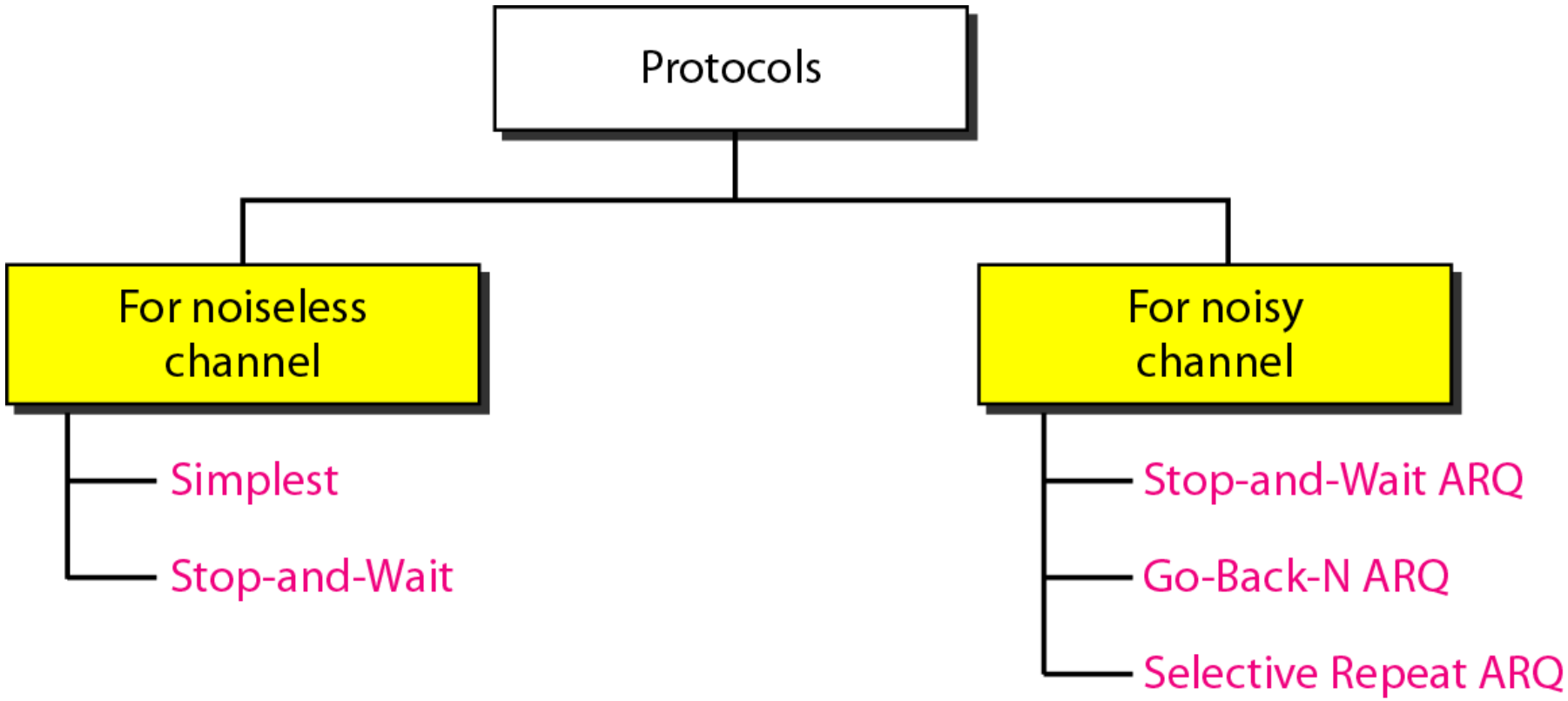
Note

Error control in the data link layer is based on automatic repeat request, which is the retransmission of data.

PROTOCOLS

Now let us see how the data link layer can combine framing, flow control, and error control to achieve the delivery of data from one node to another. The protocols are normally implemented in software by using one of the common programming languages. To make our discussions language-free, we have written in pseudocode a version of each protocol that concentrates mostly on the procedure instead of delving into the details of language rules.

Taxonomy of protocols



NOISELESS CHANNELS

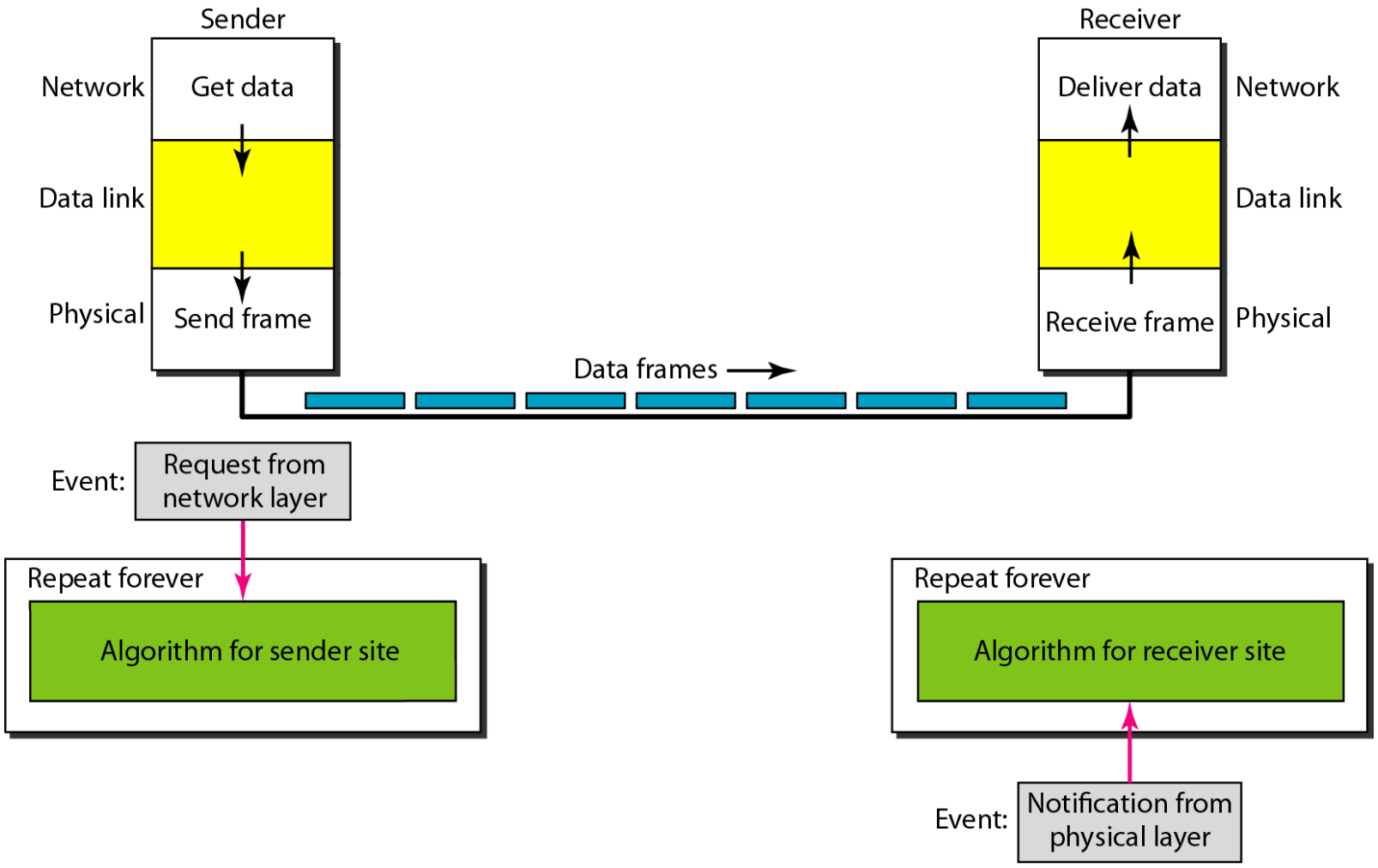
Let us first assume we have an ideal channel in which no frames are lost, duplicated, or corrupted. We introduce two protocols for this type of channel.

Topics discussed in this section:

Simplest Protocol

Stop-and-Wait Protocol

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



Sender-site algorithm for the simplest protocol

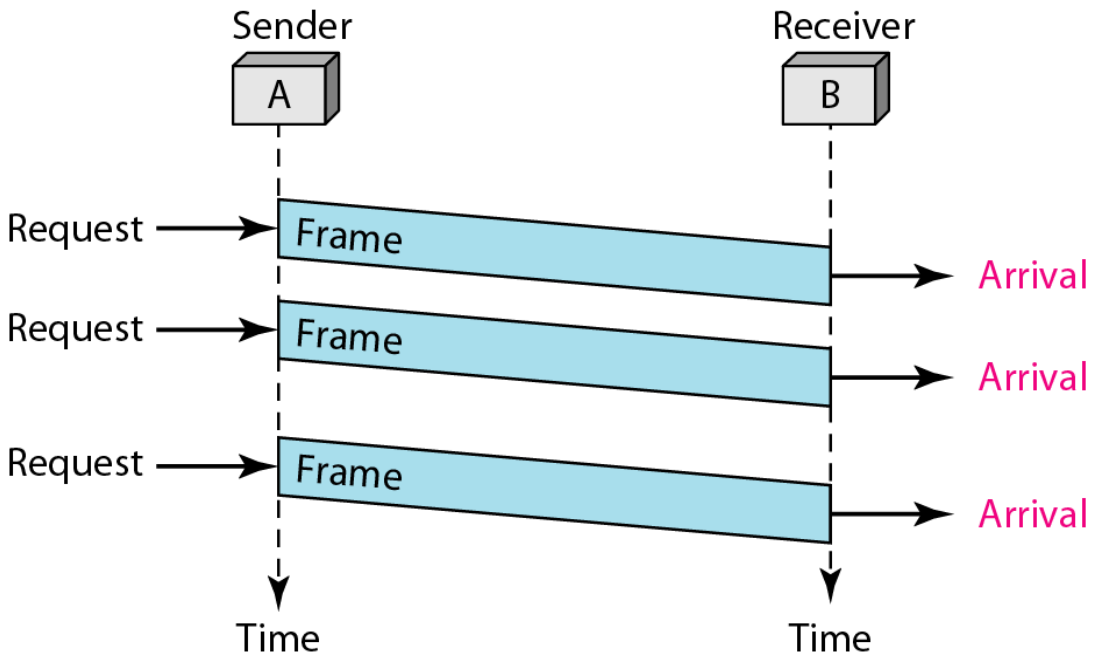
Receiver-site algorithm for the simplest protocol

```
 1  while(true)                          // Repeat forever
 2  {
 3    WaitForEvent();                    // Sleep until an event occurs
 4    if(Event(ArrivalNotification)) //Data frame arrived
 5    {
 6       ReceiveFrame();
 7       ExtractData();
 8       DeliverData();                  //Deliver data to network layer
 9    }
10  }
```

Example 1

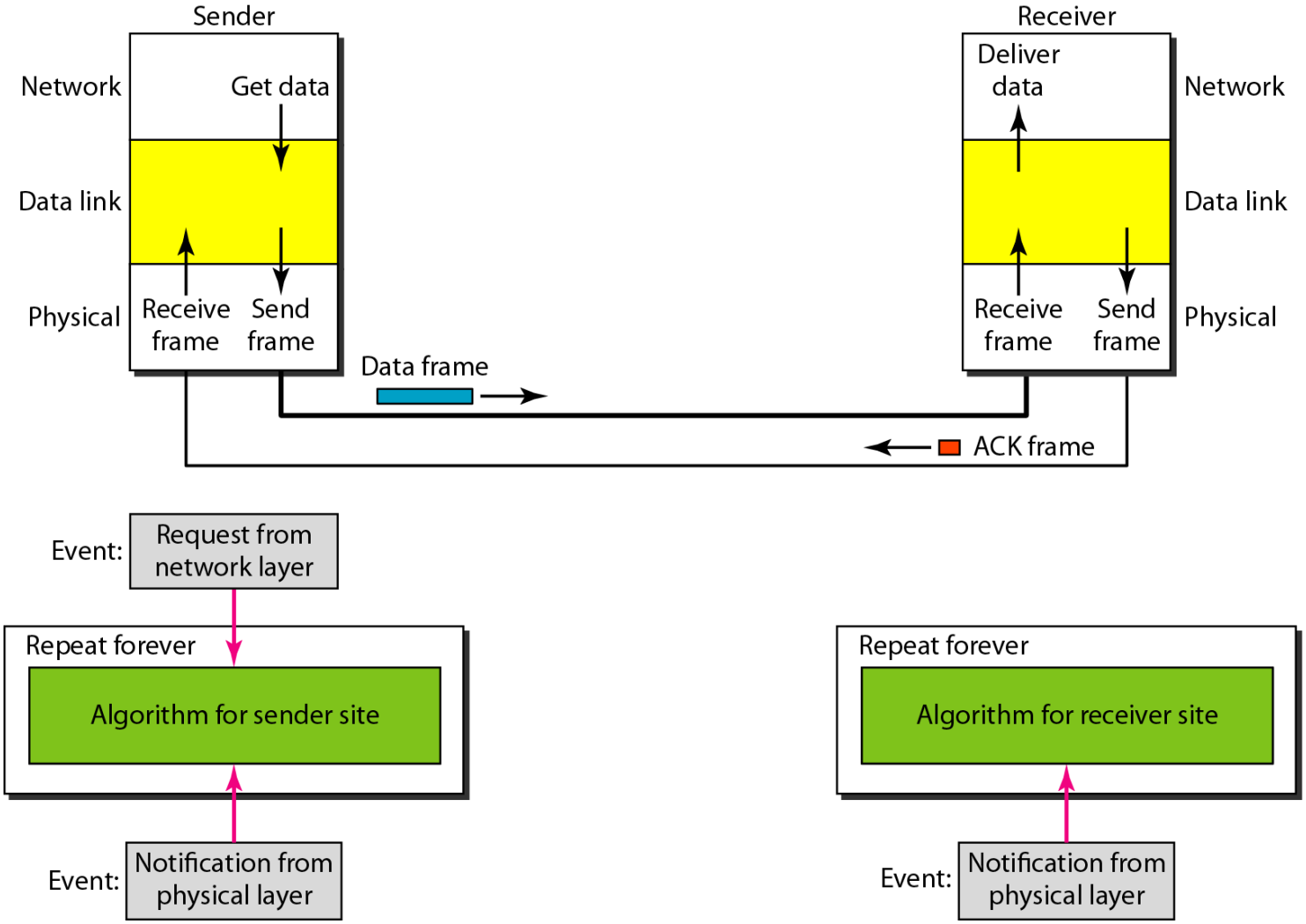
Figure shows an example of communication using this protocol. It is very simple. The sender sends a sequence of frames without even thinking about the receiver. To send three frames, three events occur at the sender site and three events at the receiver site. Note that the data frames are shown by tilted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

Flow diagram for Example 1



Stop-and-Wait Protocol

Design of Stop-and-Wait Protocol



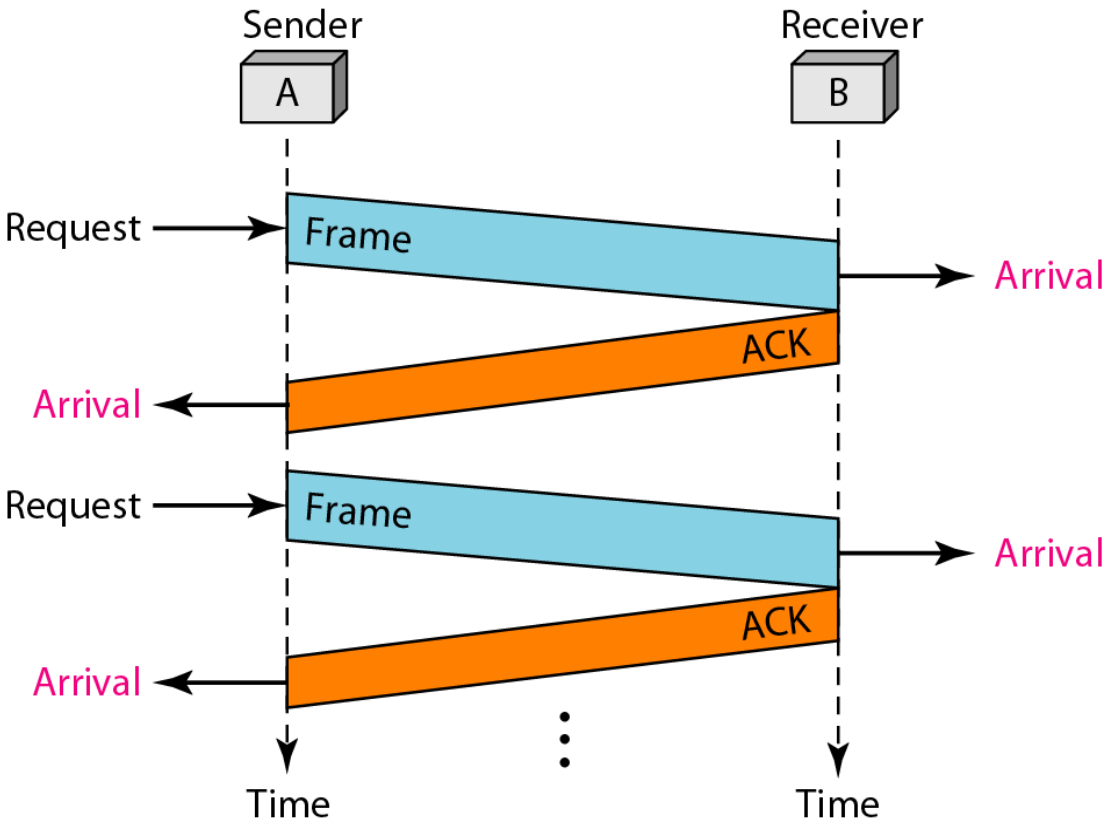
Sender-site algorithm for Stop-and-Wait Protocol

Algorithm ***Receiver-site algorithm for Stop-and-Wait Protocol***

Example 2

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

Flow diagram for Example 2



NOISY CHANNELS

Although the Stop-and-Wait Protocol gives us an idea of how to add flow control to its predecessor, noiseless channels are nonexistent. We discuss three protocols in this section that use error control.

Topics discussed in this section:

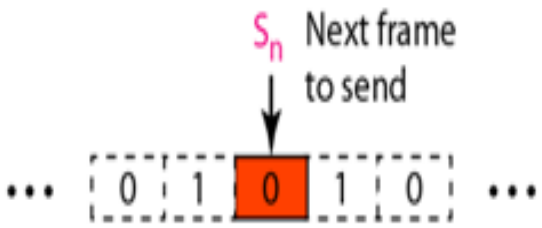
Stop-and-Wait Automatic Repeat Request
Go-Back-N Automatic Repeat Request
Selective Repeat Automatic Repeat Request

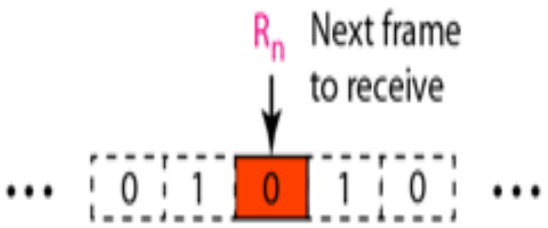
Stop and Wait ARQ

- Sequence Number
- Acknowledgement Number

Stop and Wait ARQ

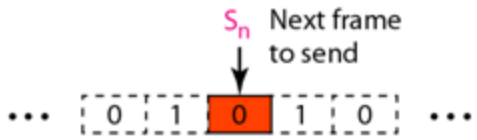
Error correction in Stop-and-Wait ARQ is done by keeping a copy of the sent frame and retransmitting of the frame when the timer expires.

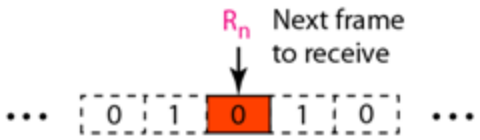




Stop and Wait ARQ

Window size of this protocol is 1.
The possible number of sequence numbers is given by
2^1
i.e only 2 SEQ_NO are possible: 0, 1

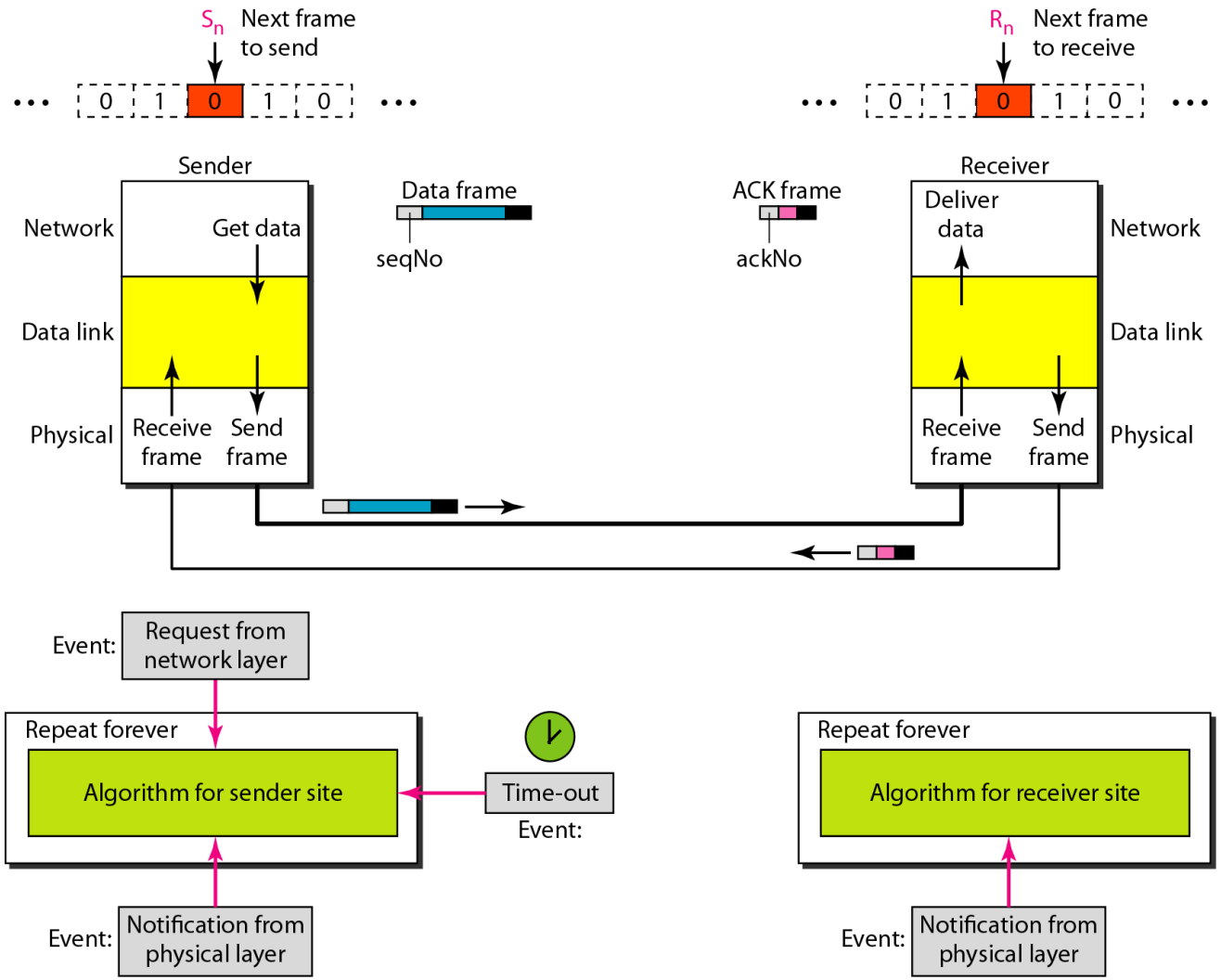




Note

In Stop-and-Wait ARQ, the acknowledgment number always announces the sequence number of the next frame expected.

Design of the Stop-and-Wait ARQ Protocol

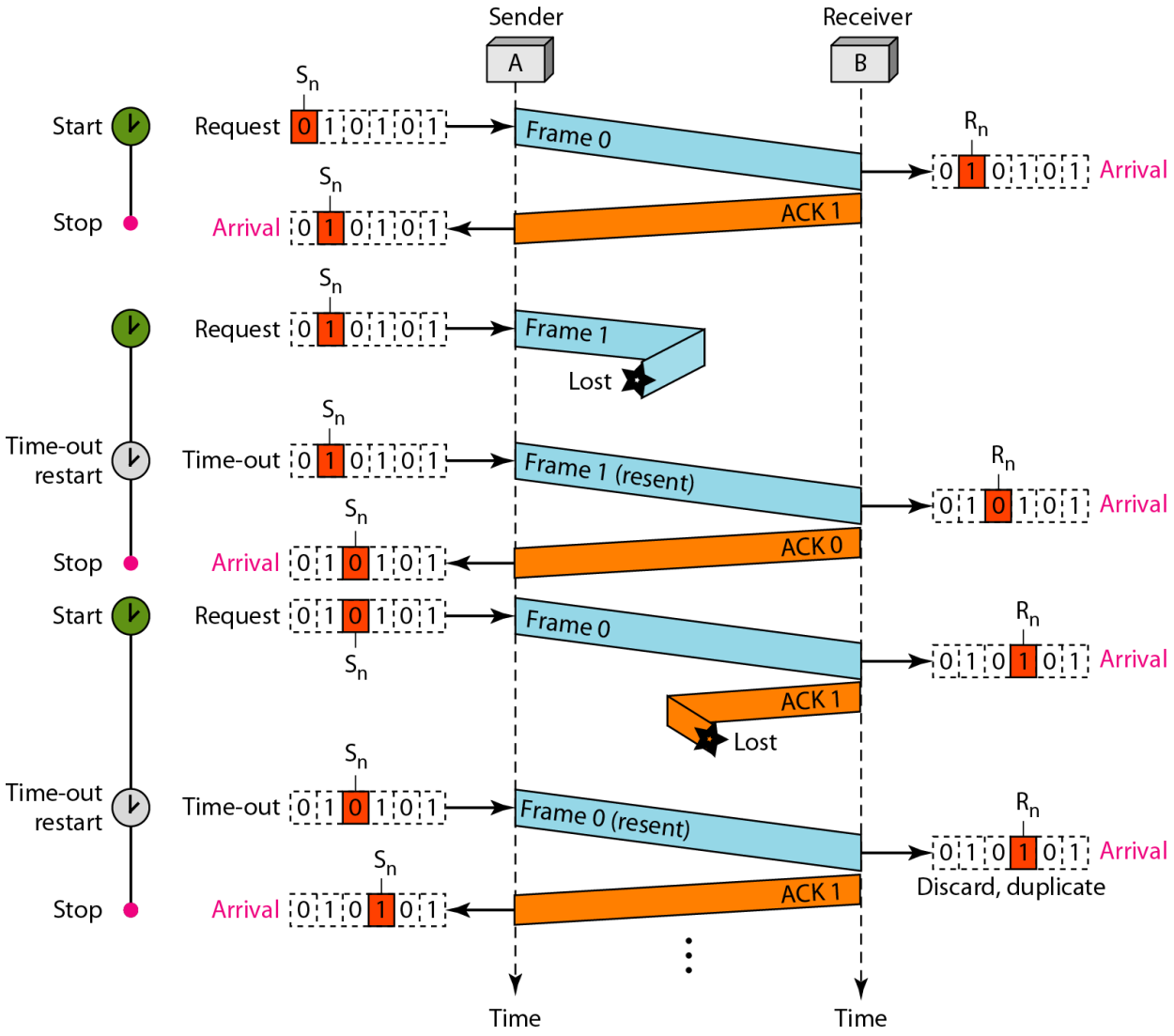


Example 3

How Stop-and-Wait ARQ reacts to

1. *Lost Frame*
2. *Damaged Frame*
3. *Lost ACK*

Flow diagram



Efficiency?..

Example

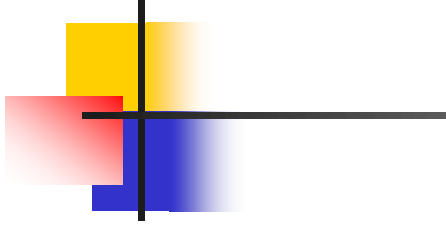
Assume that, in a Stop-and-Wait ARQ system, the bandwidth of the line is 1 Mbps, and 1 bit takes 20 ms to make a round trip. What is the bandwidth-delay product? If the system data frames are 1000 bits in length, what is the utilization percentage of the link?

Solution

The bandwidth-delay product is

The system can send 20,000 bits during the time it takes for the data to go from the sender to the receiver and then back again. However, the system sends only 1000 bits. We can say that the link utilization is only 1000/20,000, or 5 percent. For this reason, for a link with a high bandwidth or long delay, the use of Stop-and-Wait ARQ wastes the capacity of the link.

Solution ?...



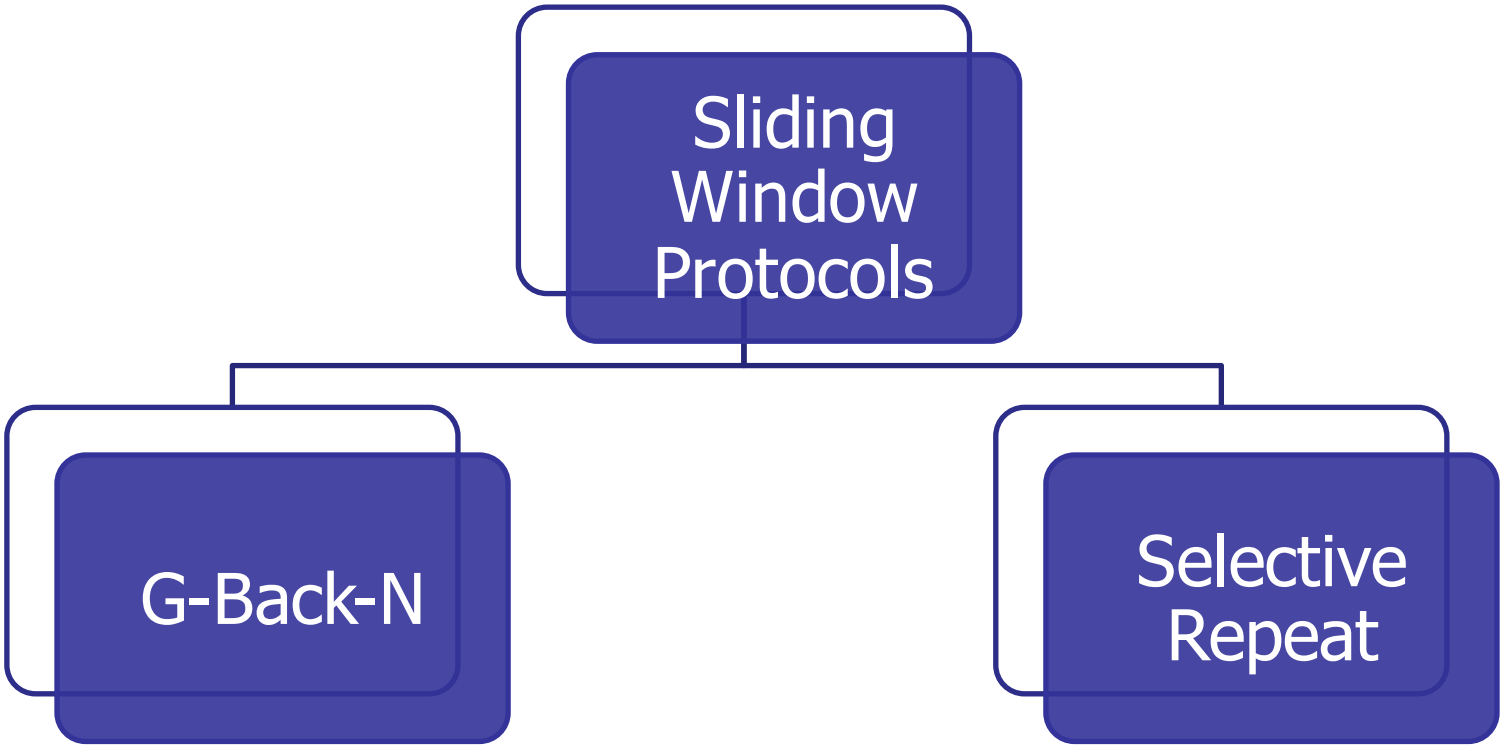
What if we can send multiple frames before waiting for an ACK?

What is the utilization percentage of the link in Example if we have a protocol that can send up to 15 frames before stopping and worrying about the acknowledgments?

Solution

The bandwidth-delay product is still 20,000 bits. The system can send up to 15 frames or 15,000 bits during a round trip. This means the utilization is 15,000/20,000, or 75 percent.

Sliding Window Protocols

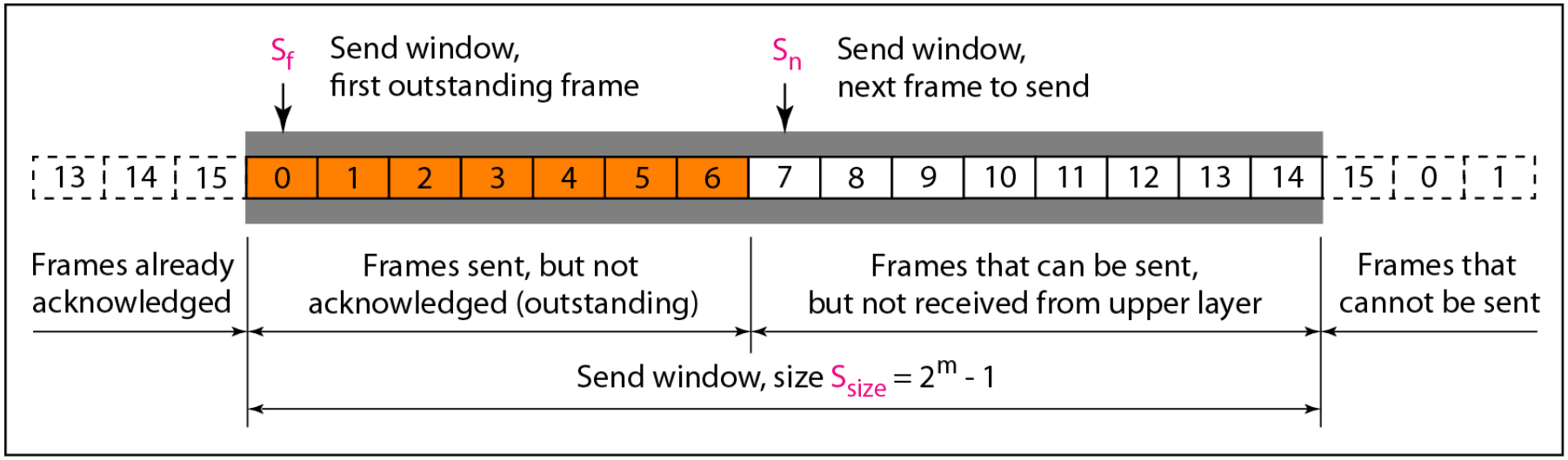


Go-Back-N

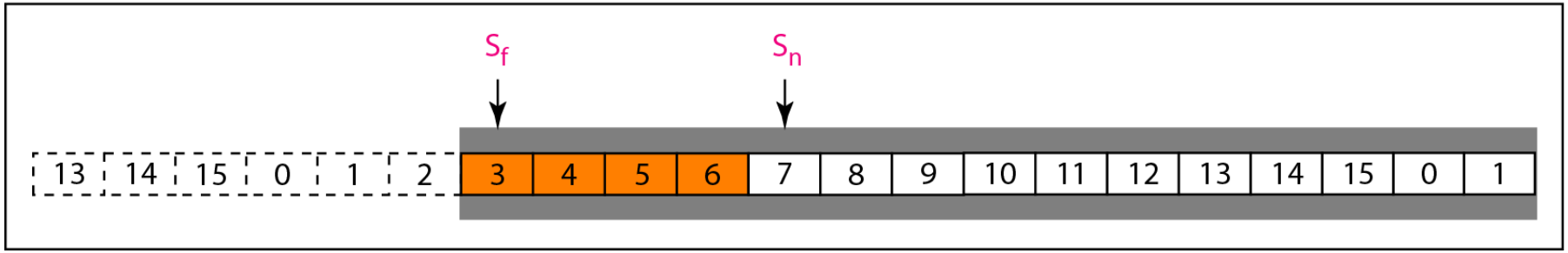
Note

In the Go-Back-N Protocol, the sequence numbers are modulo 2^m, where m is the size of the sequence number field in bits.

Send window for Go-Back-N ARQ



a. Send window before sliding



b. Send window after sliding

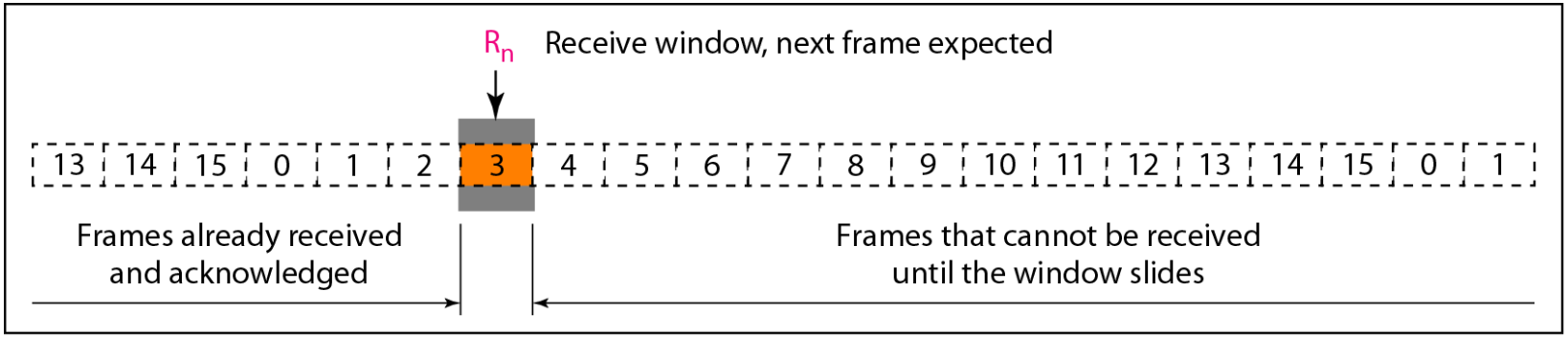
Note

The send window is an abstract concept defining an imaginary box of size $2^m - 1$ with three variables: S_f, S_n, and S_{size}.

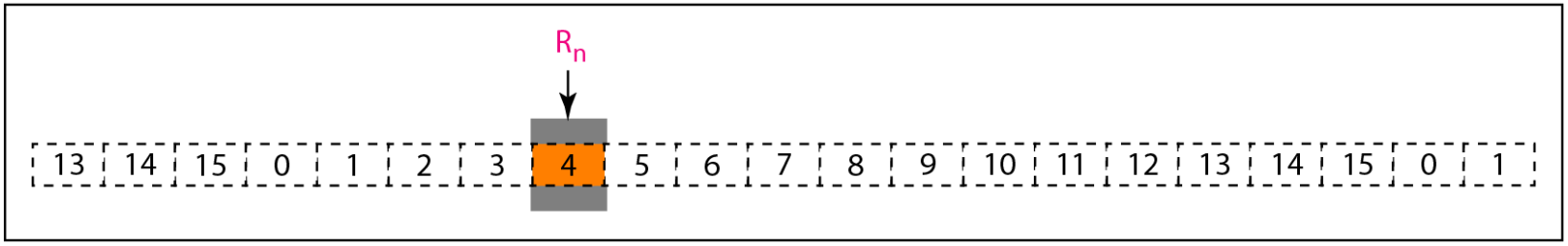
Note

The send window can slide one or more slots when a valid acknowledgment arrives.

Receive window for Go-Back-N ARQ



a. Receive window

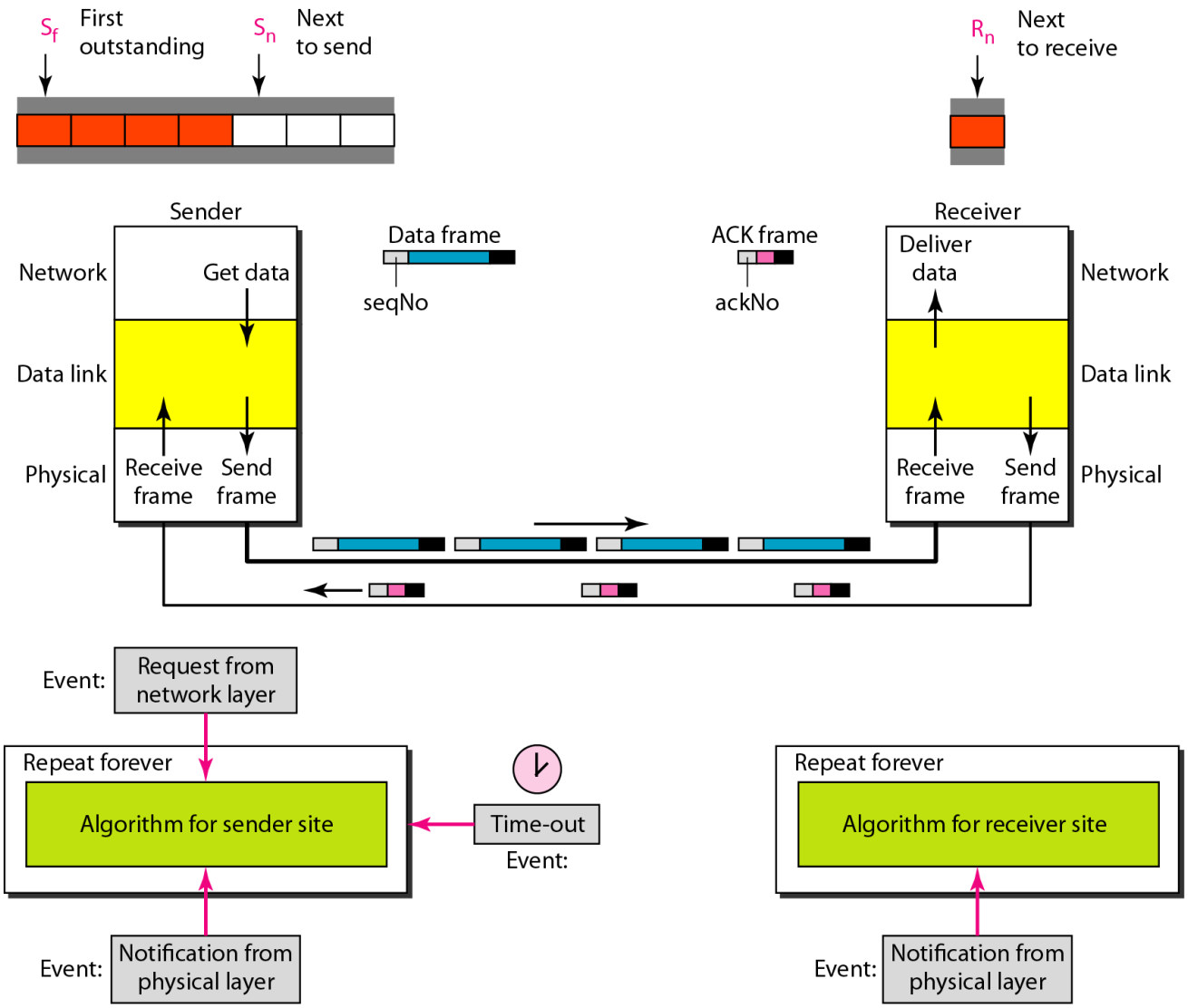


b. Window after sliding

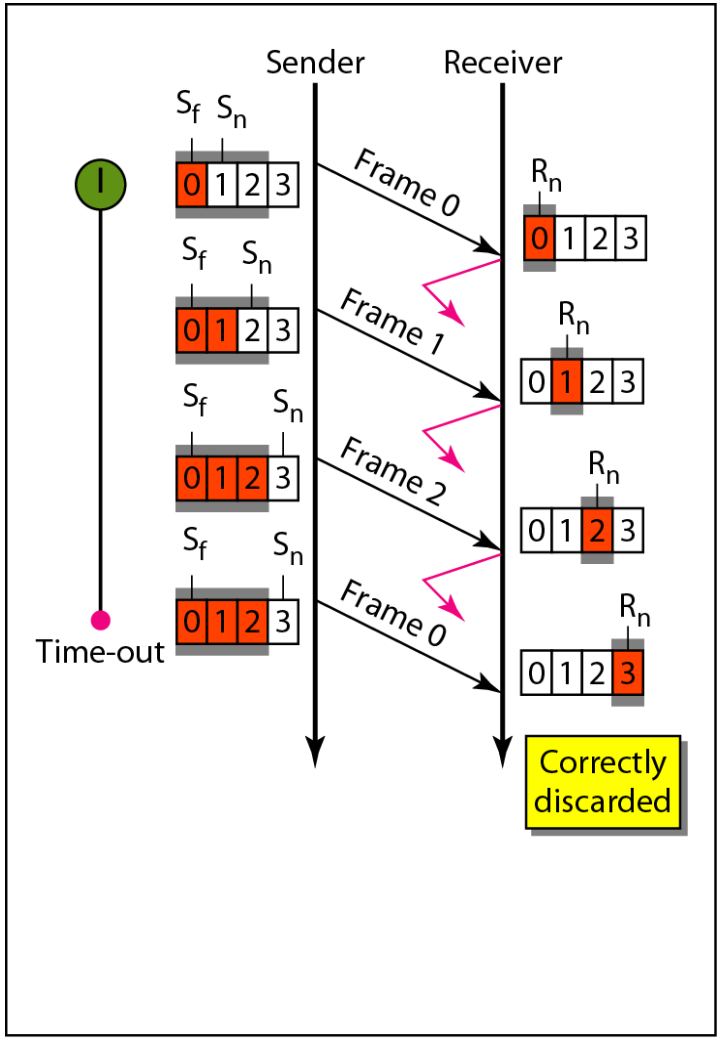
Note

The receive window is an abstract concept defining an imaginary box of size 1 with one single variable R_n. The window slides when a correct frame has arrived; sliding occurs one slot at a time.

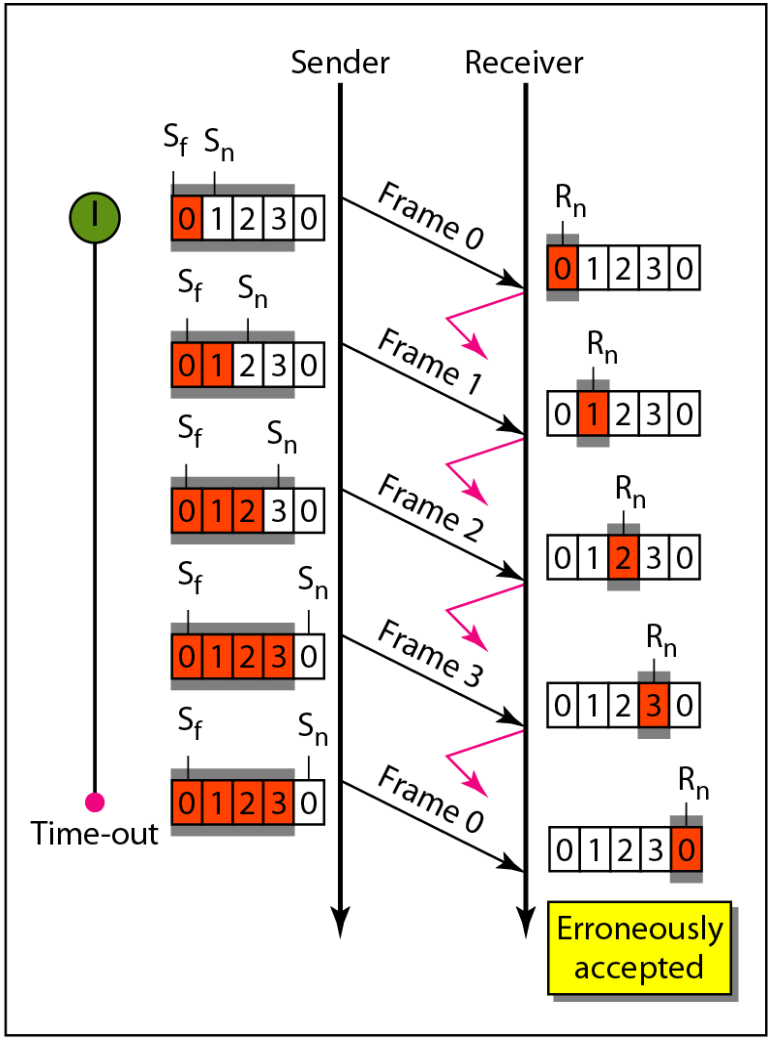
Design of Go-Back-N ARQ



Window size for Go-Back-N ARQ



a. Window size < 2^m

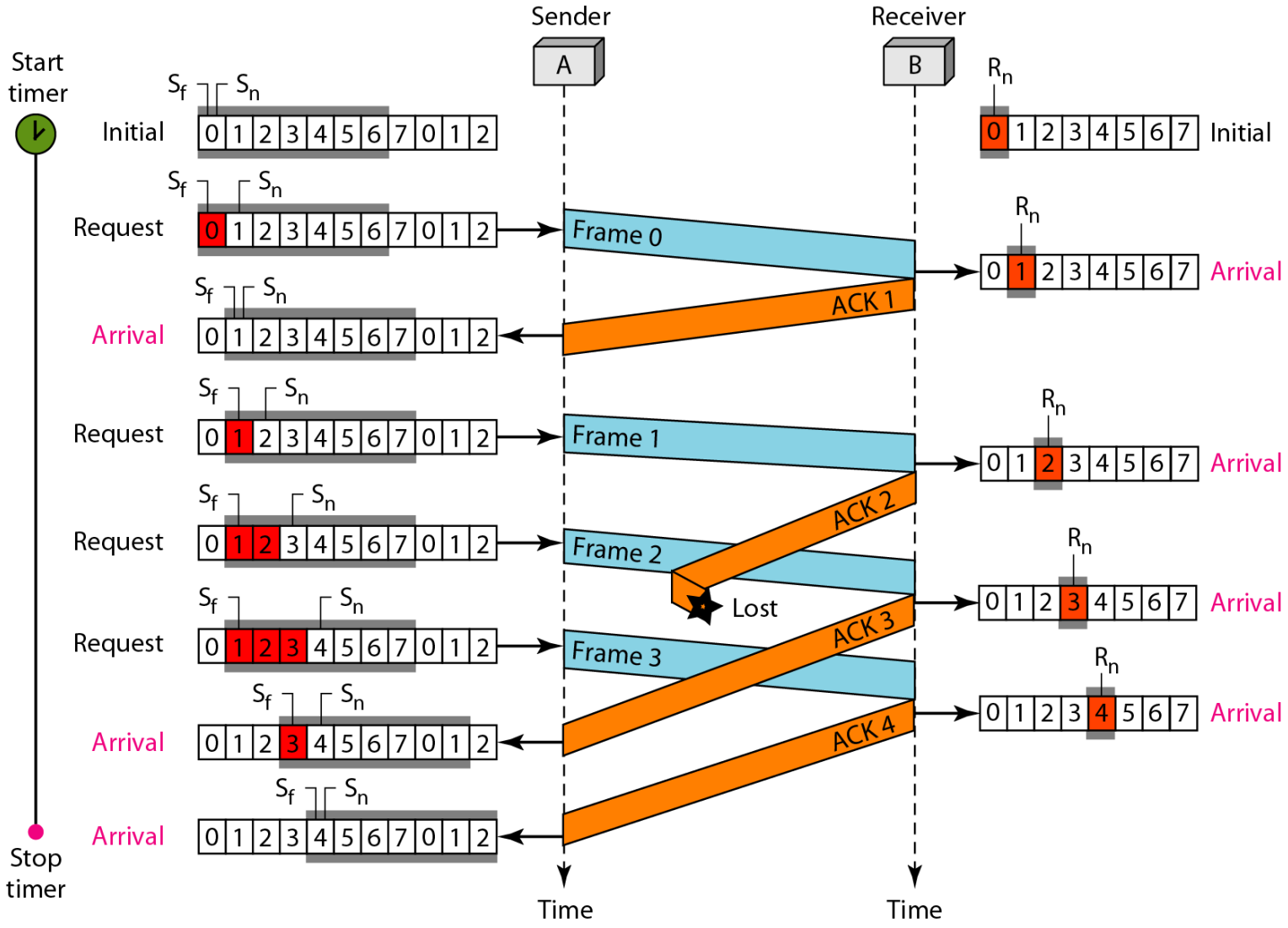


b. Window size = 2^m

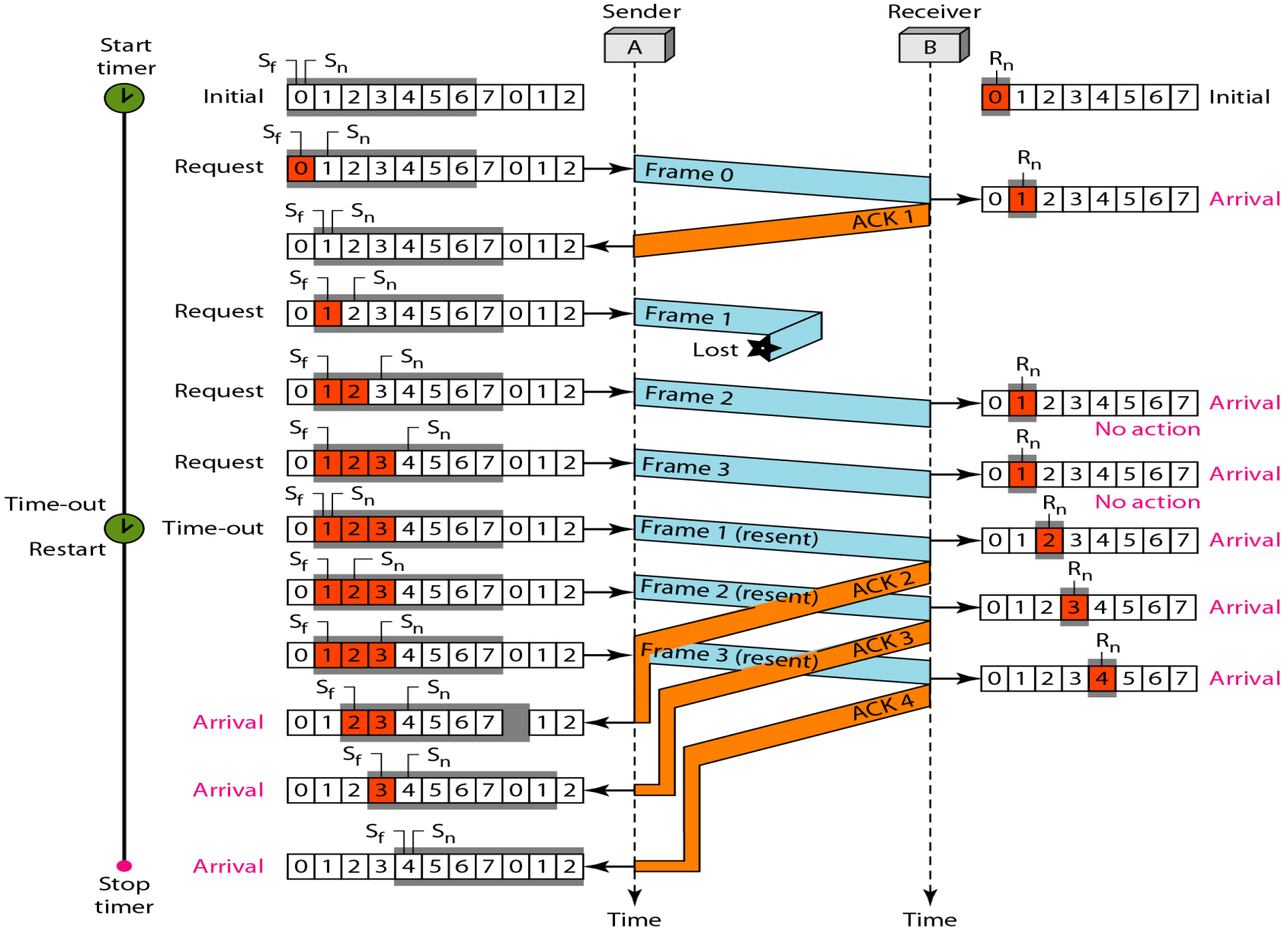
Note

In Go-Back-N ARQ, the size of the send window must be less than 2^m; the size of the receiver window is always 1.

Flow diagram Example of Go-Back-N (Delayed and Lost ACKs)



Lost Frame


Sender
Receiver
A
B
Start timer
Initial
Request
Frame 0
ACK 1
Request
Frame 1
Lost
Request
Frame 2
No action
Request
Frame 3
No action
Time-out
Restart
Time-out
Frame 1 (resent)
Frame 2 (resent)
ACK 2
Frame 3 (resent)
ACK 3
ACK 4
Arrival
Arrival
Arrival
Stop timer
Time
Time

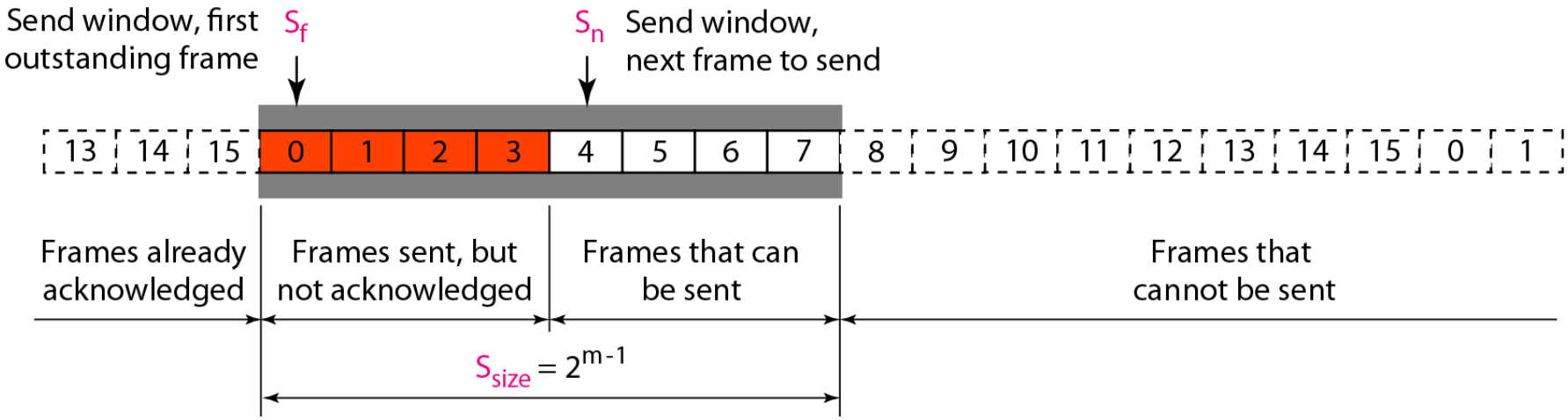
Note

Stop-and-Wait ARQ is a special case of Go-Back-N ARQ in which the size of the send window is 1.

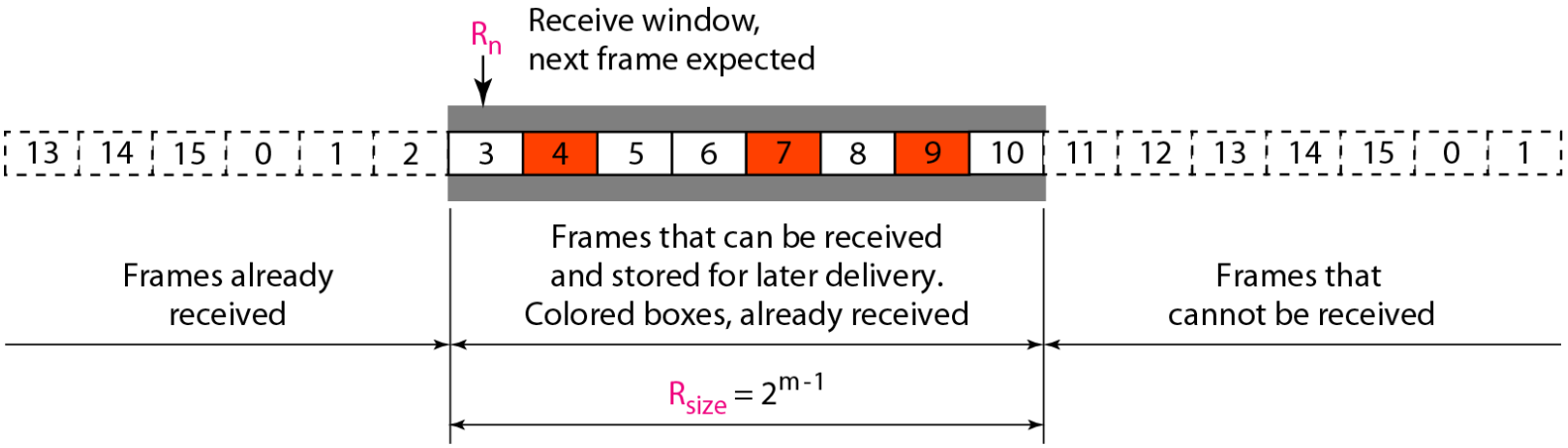
Selective Repeat ARQ

- In this protocol, only damaged/lost frame is retransmitted.
- Out-of-order frames are stored in the buffer, until it gets the ordered frame.
- Sender and receiver window size is same, coz of the need for storing the out-of-order frames.
- It is suited for noisy channel, where in more number of retransmissions are expensive.
- But the complexity is
  - Sender should have searching algorithm
  - Receiver should have sorting algorithms.

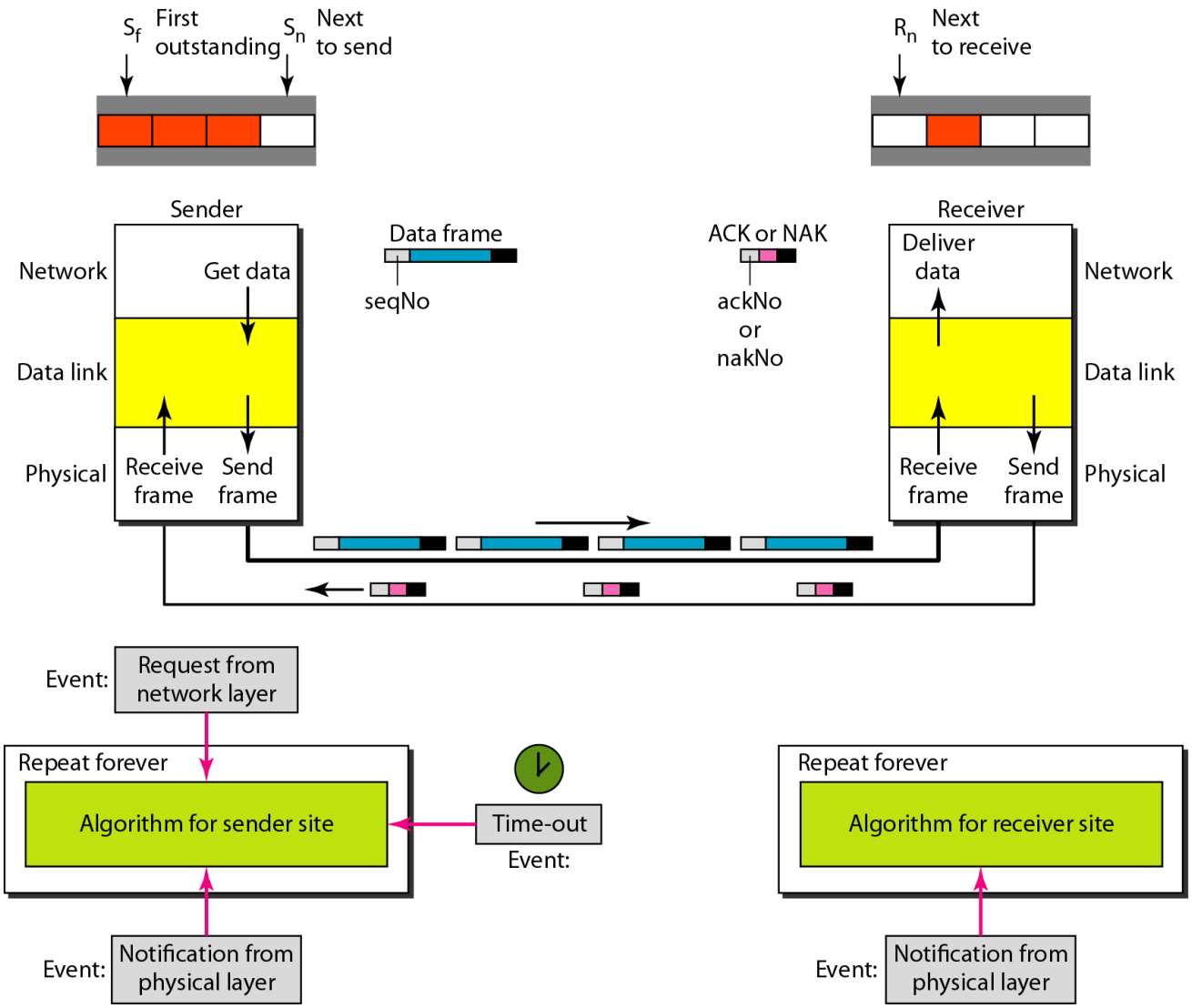
Send window for Selective Repeat ARQ



Receive window for Selective Repeat ARQ

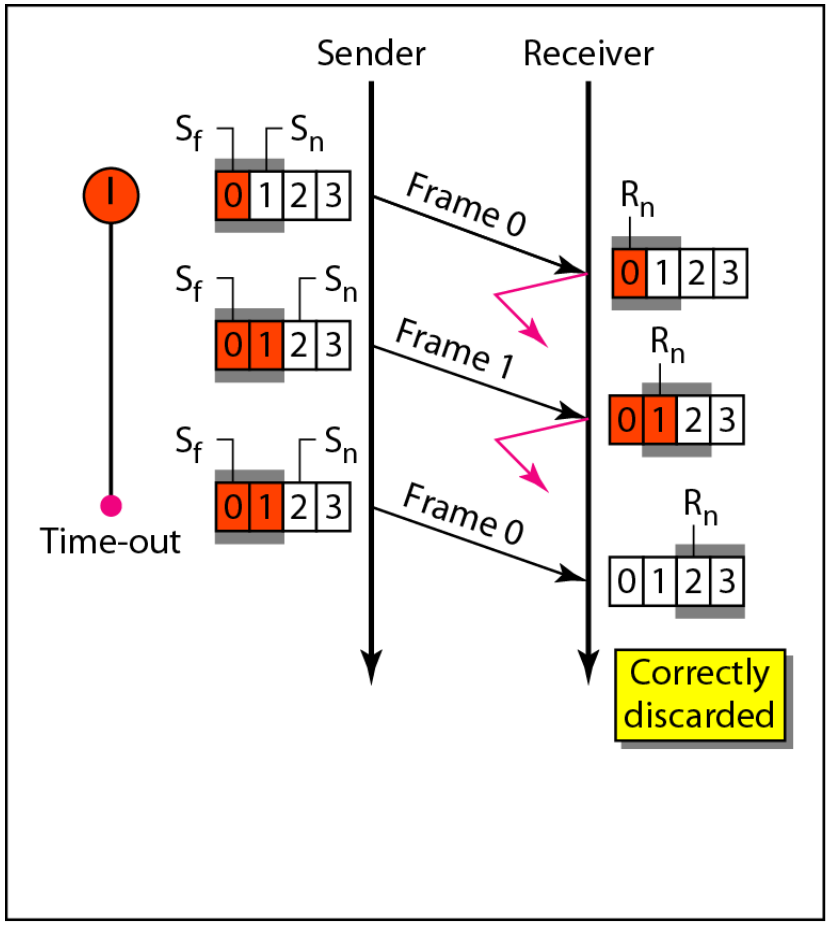


Design of Selective Repeat ARQ

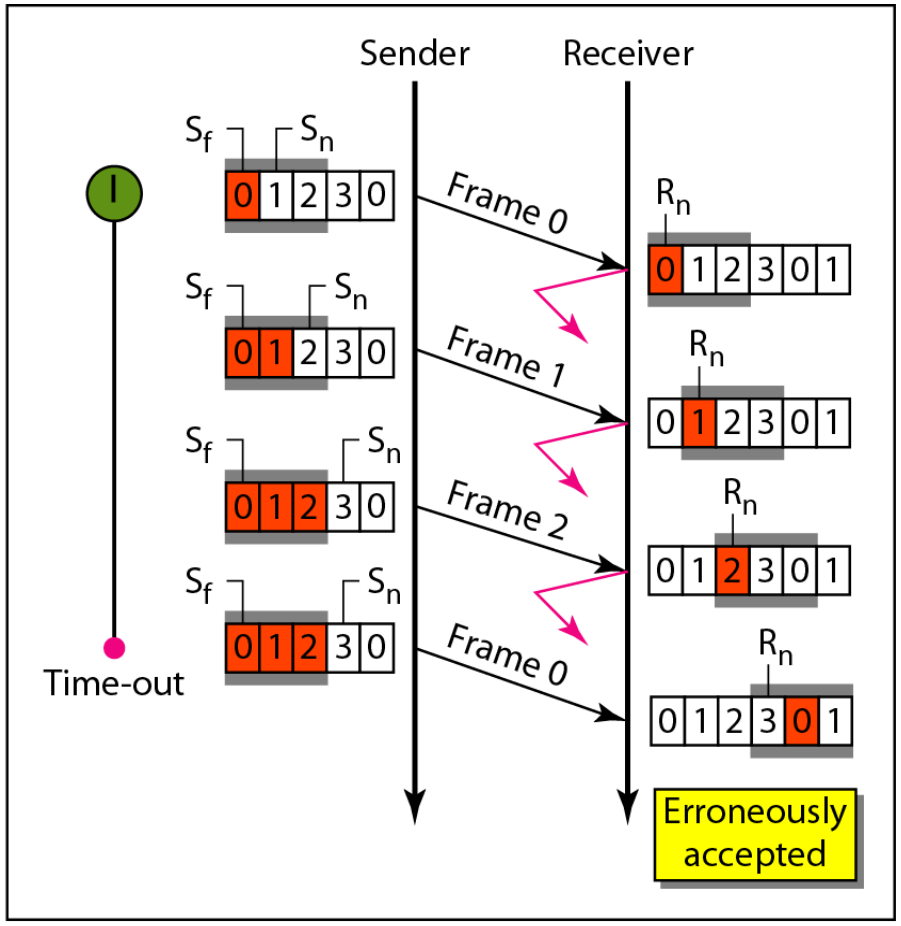


Window size=2^{m-1}

m=2



a. Window size = 2^{m-1}

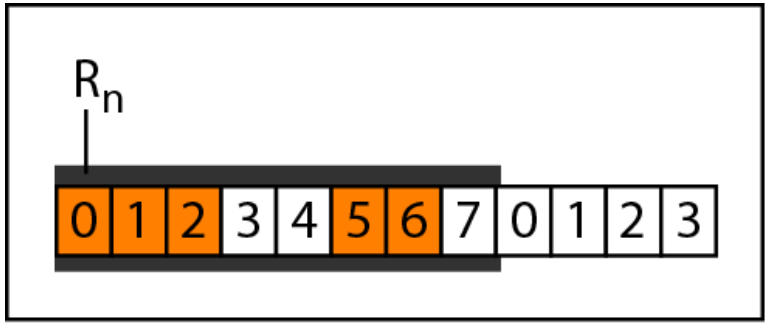


b. Window size > 2^{m-1}

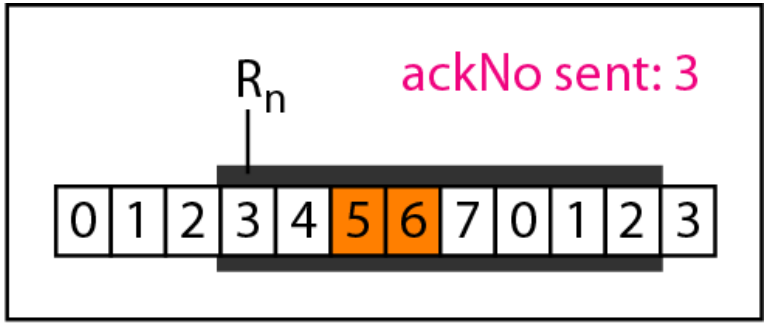
Note

In Selective Repeat ARQ, the size of the sender and receiver window must be at most one-half of 2^m.

Delivery of data in Selective Repeat ARQ

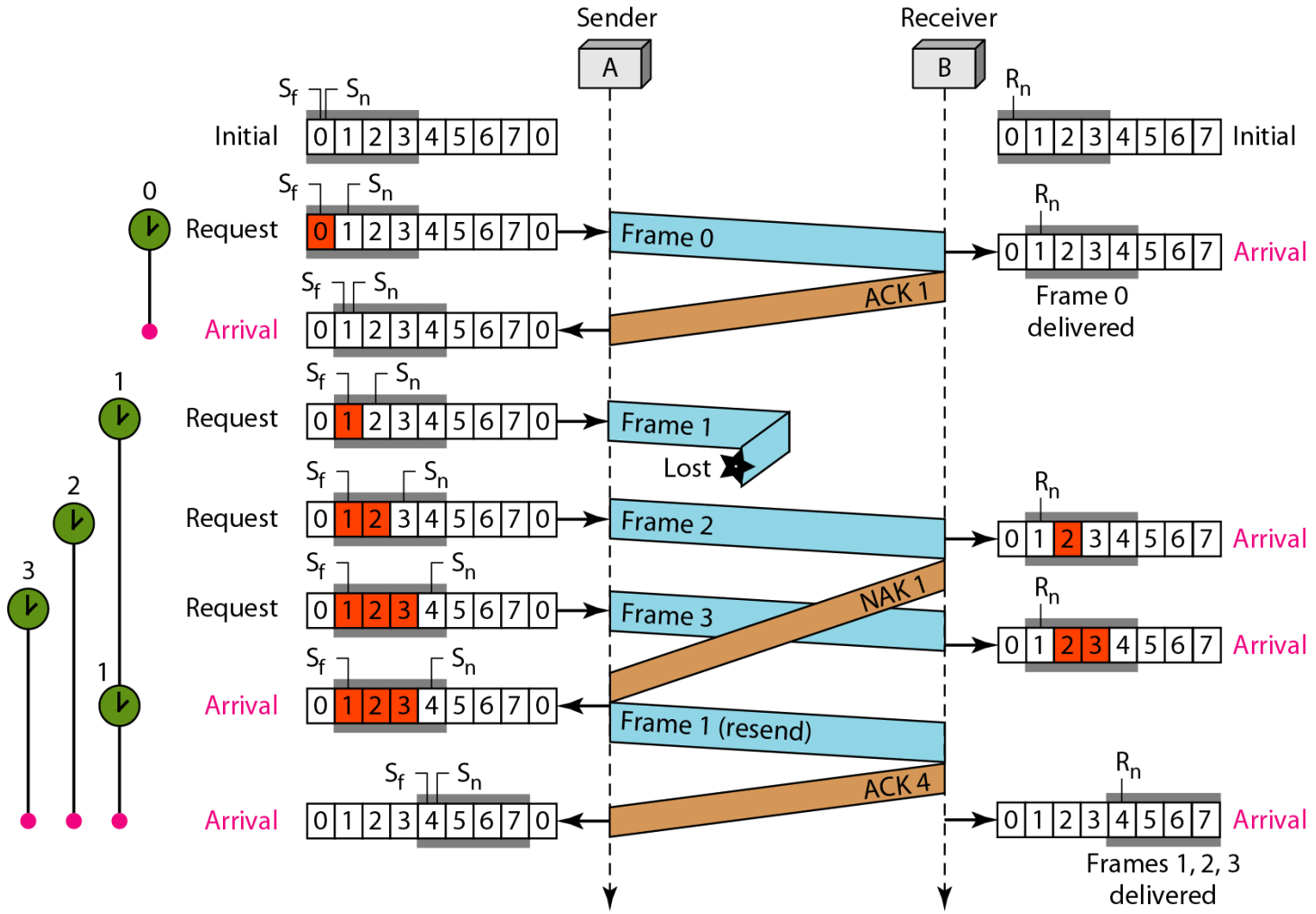


a. Before delivery



b. After delivery

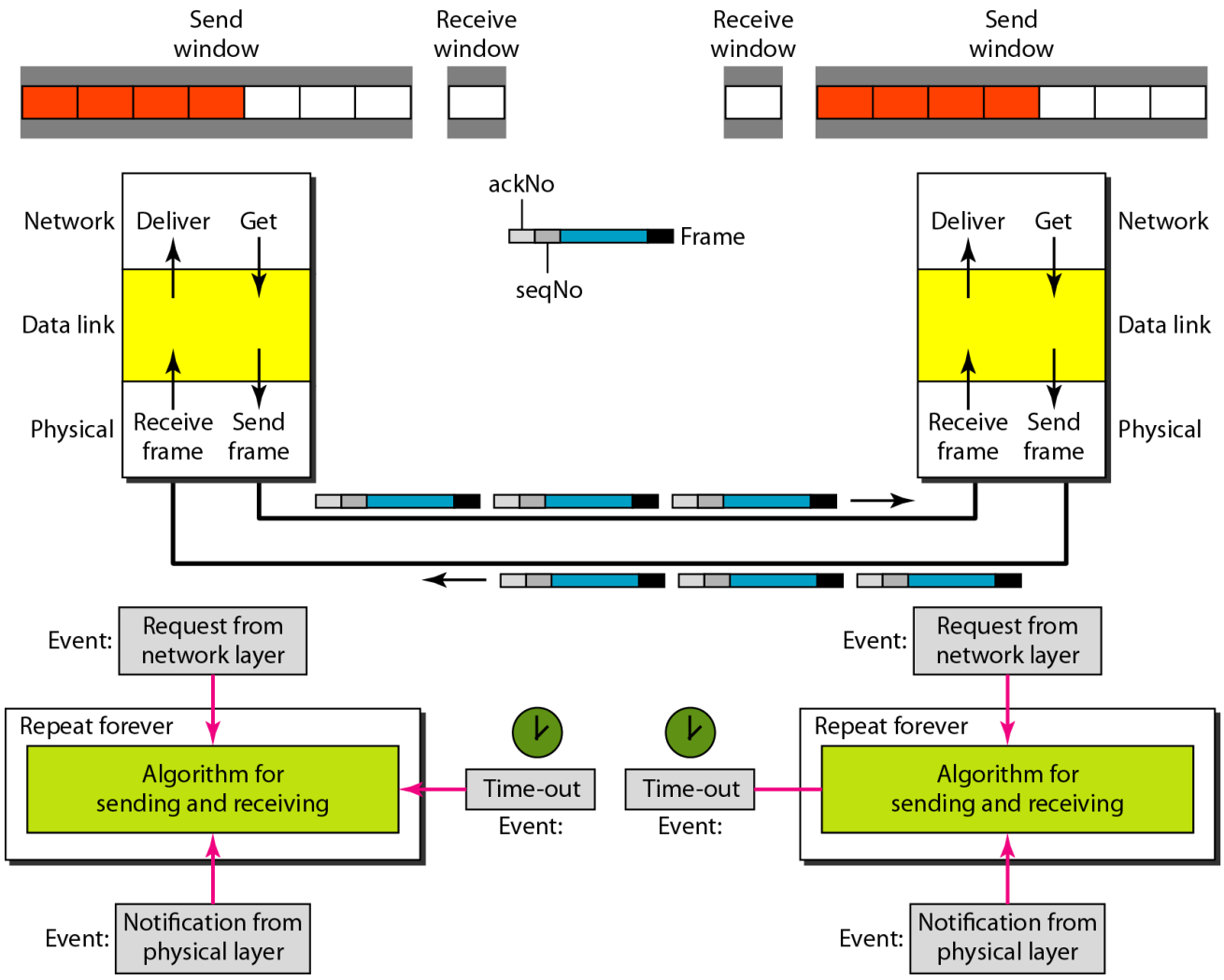
Flow diagram-Lost Frame



Piggybacking

- Normally, data flows in both directions. i.e from node A to node B and node B to node A.
- Piggybacking is used to improve the efficiency of bi-directional protocol by carrying control information (ACK) as well as data .

Design of piggybacking in Go-Back-N ARQ



HDLC

High-level Data Link Control (HDLC) ***is a*** ***bit-oriented*** ***protocol for communication over point-to-point and multipoint links. It implements the ARQ mechanisms.***

Topics discussed in this section:

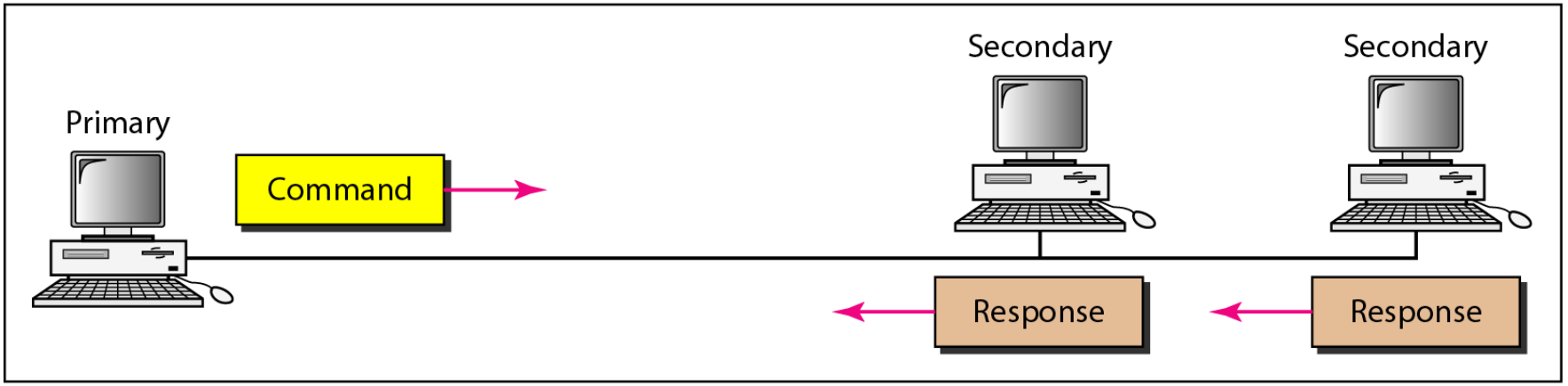
Configurations and Transfer Modes
Frames
Control Field

Configurations and Transfer Modes

- Normal Response Mode (NRM)
  - Primary sends-Commands
  - Secondary sends-Response
  - Can be point-to-point or point-to-multipoint
- Asynchronous Balanced Mode (ABM)
  - Each station can function as Primary/Secondary
  - Is always point-to-point

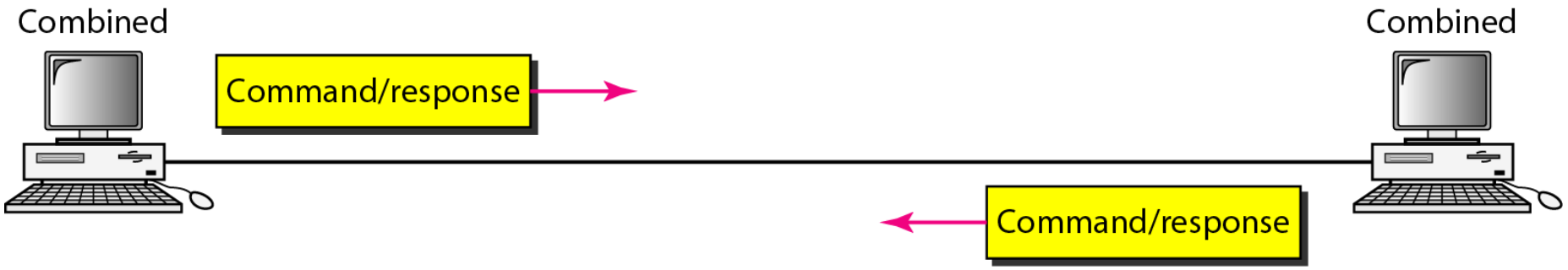
Normal response mode

a. Point-to-point



b. Multipoint

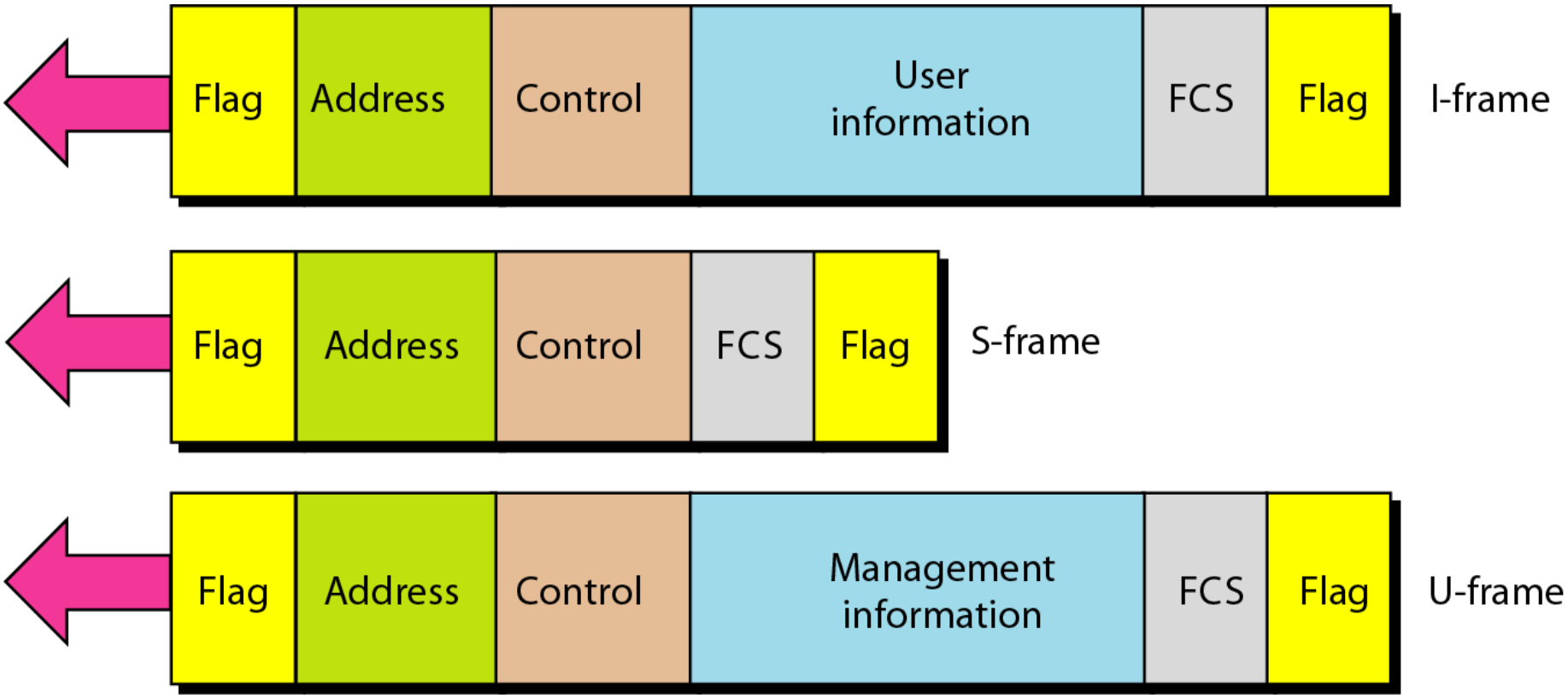
Asynchronous balanced mode



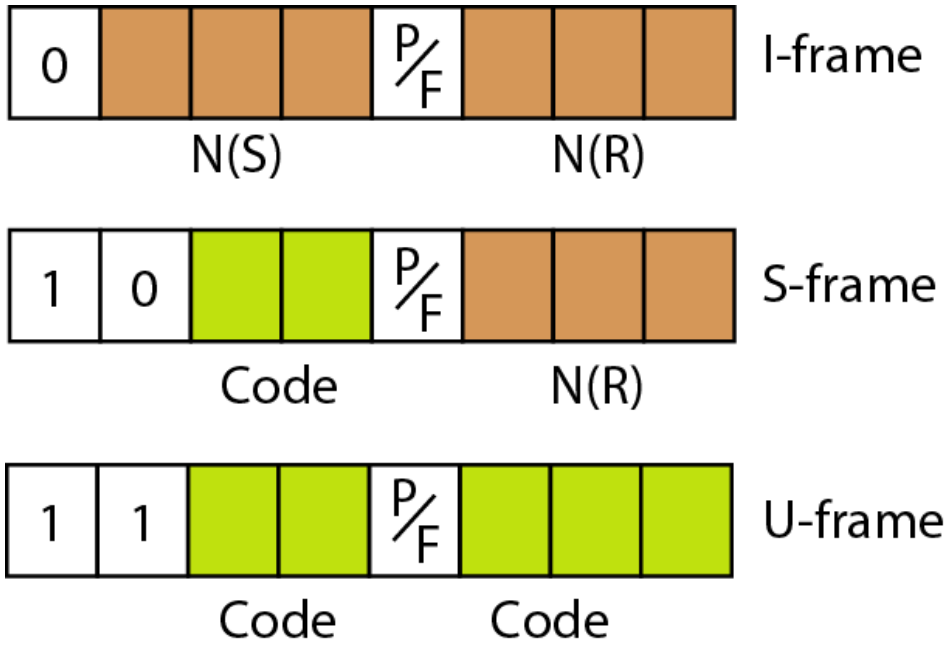
HDLC Frames

- Information Frame (I-Frame)
  - Carry user data
- Supervisory frame (S-Frame)
  - Carry control information
- Unnumbered frame (U-Frame)
  - Carry system management information.

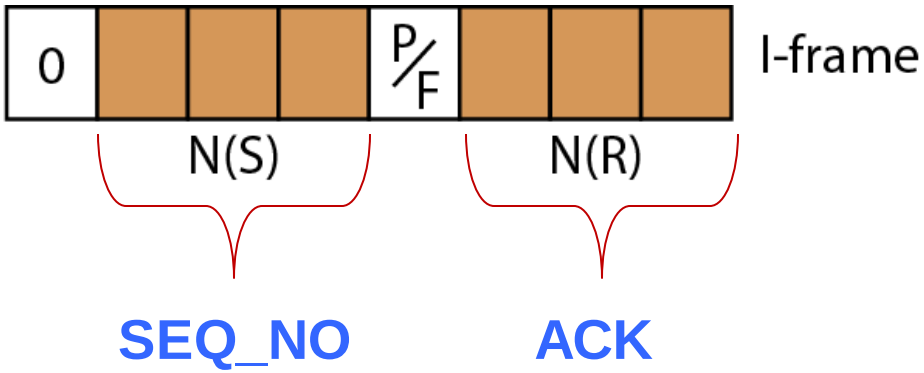
HDLC frames


Flag
Address
Control
User information
FCS
Flag
I-frame
Flag
Address
Control
FCS
Flag
S-frame
Flag
Address
Control
Management information
FCS
Flag
U-frame

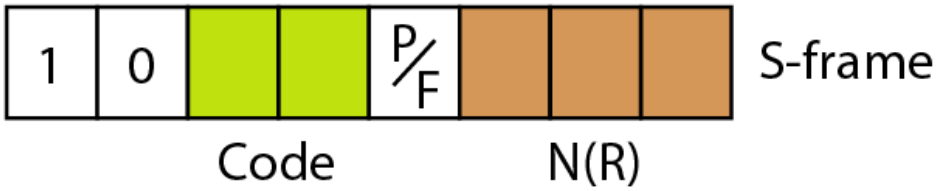
Control field format for the different frame types



I-Frame

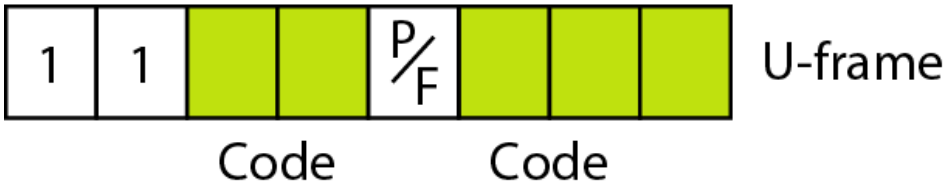


S - Frame



Code	Meaning	Purpose
00	Receive Ready (RR)	ACK frame
10	Receive Not Ready (RNR)	ACK + Congestion control
01	Reject (REJ)	NAK Frame(GBN)
11	Selective Reject (SREJ)	NAK Frame(SR)

U - Frame

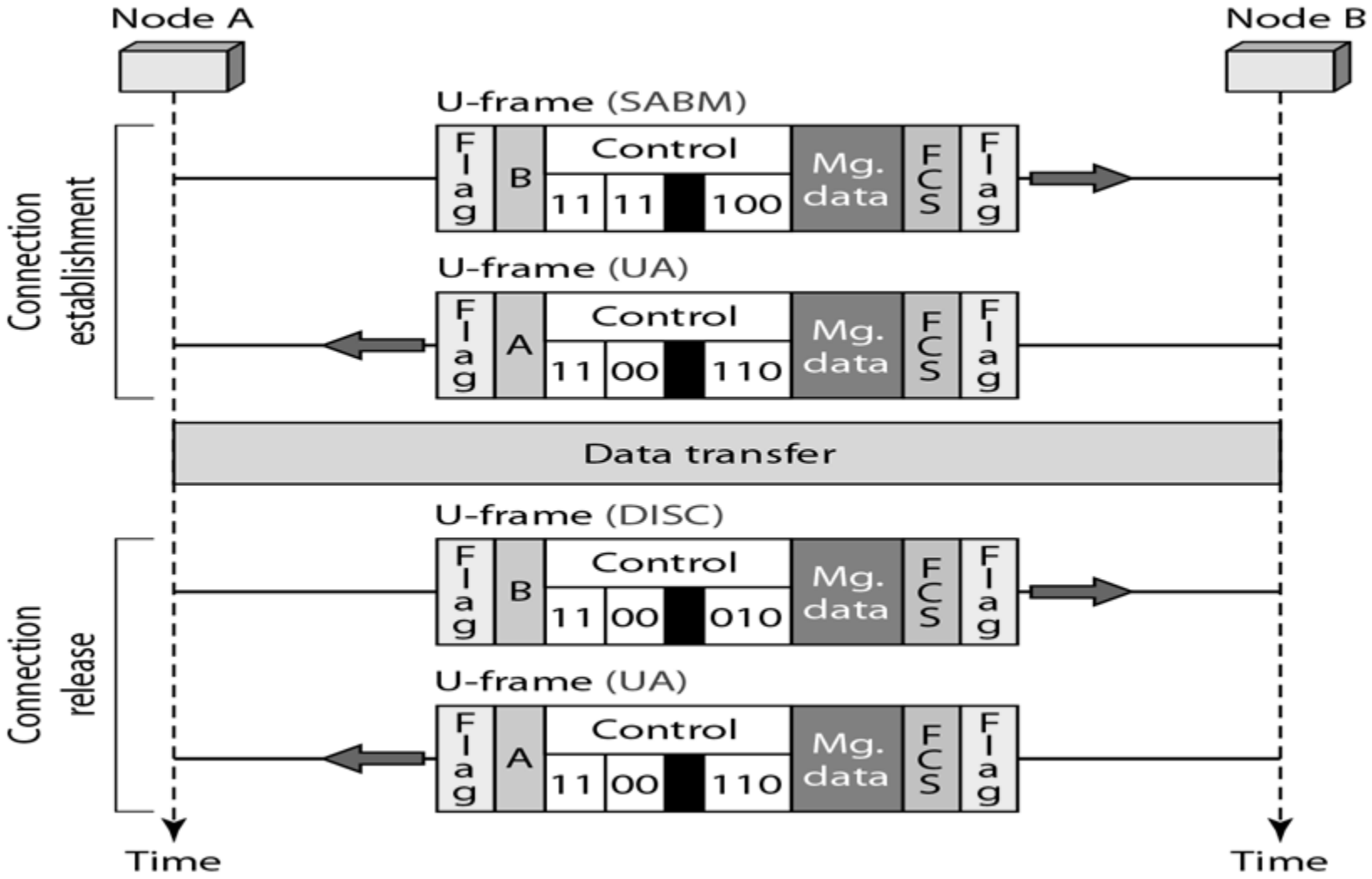


5-bit codes can be used to create 32 different types of U-frames

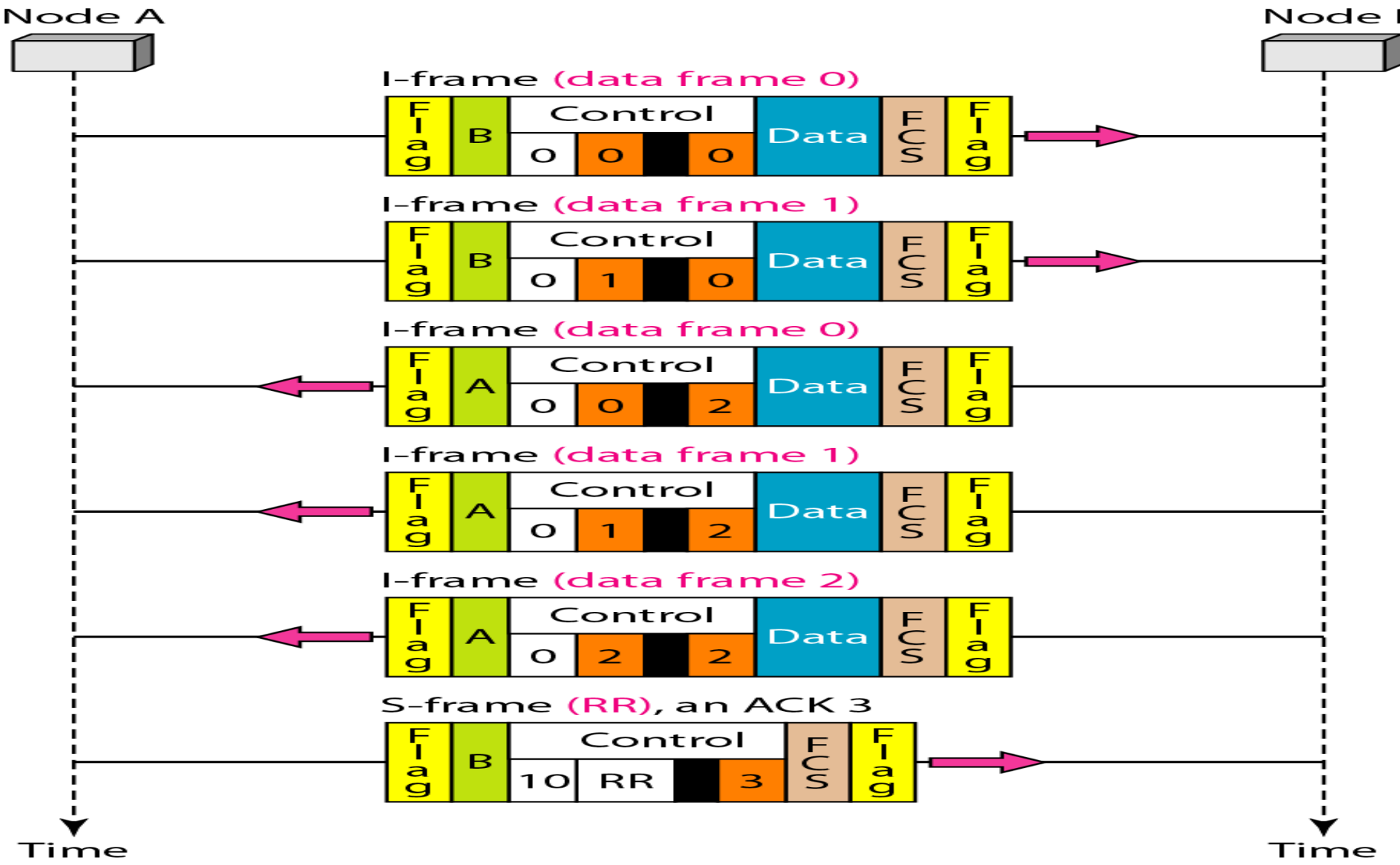
U-frame control command and response

Code	*Command*	*Response*	*Meaning*
00 001	SNRM		Set normal response mode
11 011	SNRME		Set normal response mode, extended
11 100	SABM	**DM**	Set asynchronous balanced mode or **disconnect mode**
11 110	SABME		Set asynchronous balanced mode, extended
00 000	UI	**UI**	Unnumbered information
00 110		**UA**	**Unnumbered acknowledgment**
00 010	DISC	**RD**	Disconnect or **request disconnect**
10 000	SIM	**RIM**	Set initialization mode or **request information mode**
00 100	UP		Unnumbered poll
11 001	RSET		Reset
11 101	XID	**XID**	Exchange ID
10 001	FRMR	**FRMR**	Frame reject

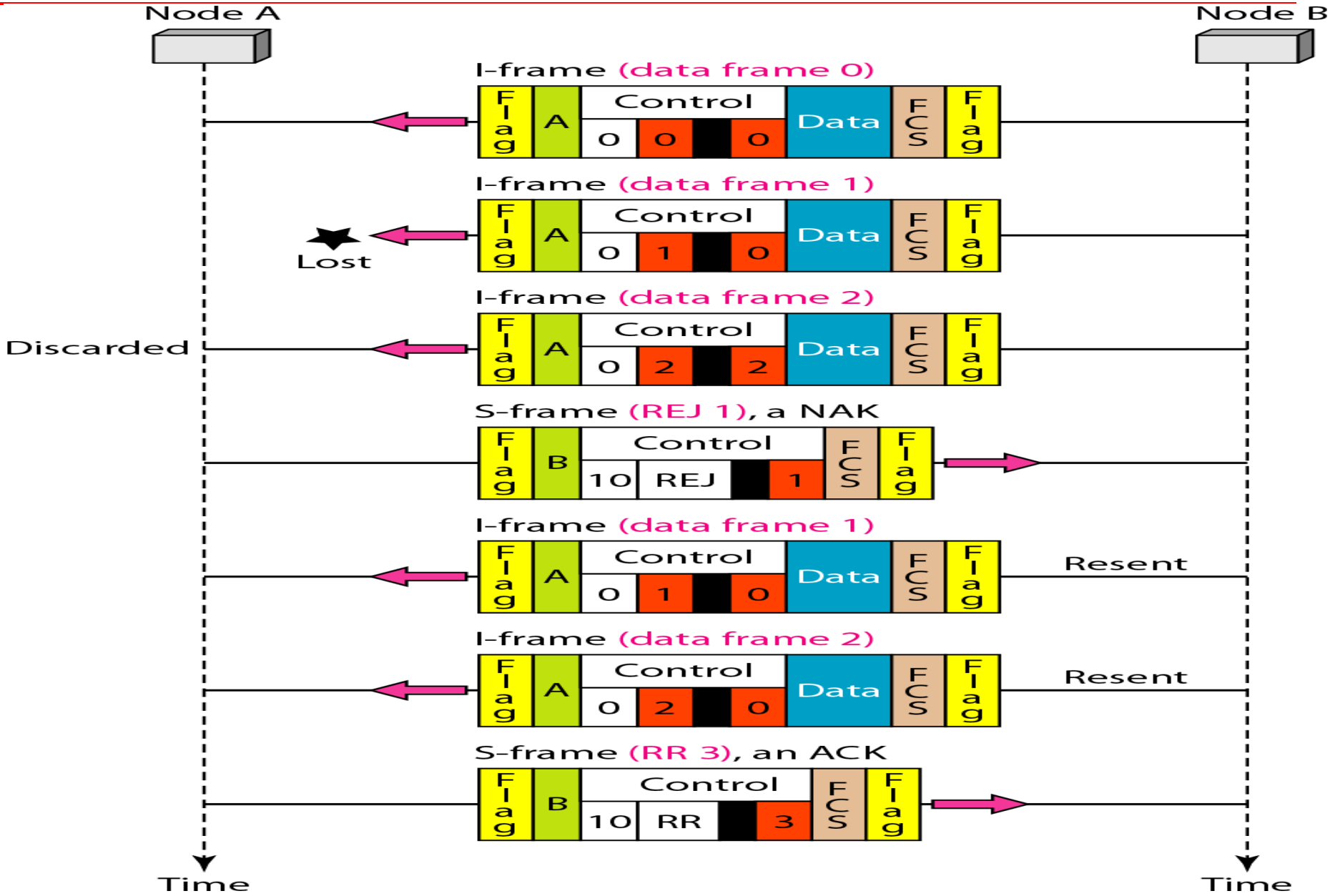
***U-frames* can be used for connection establishment and connection release**



Exchange using piggybacking (without error)



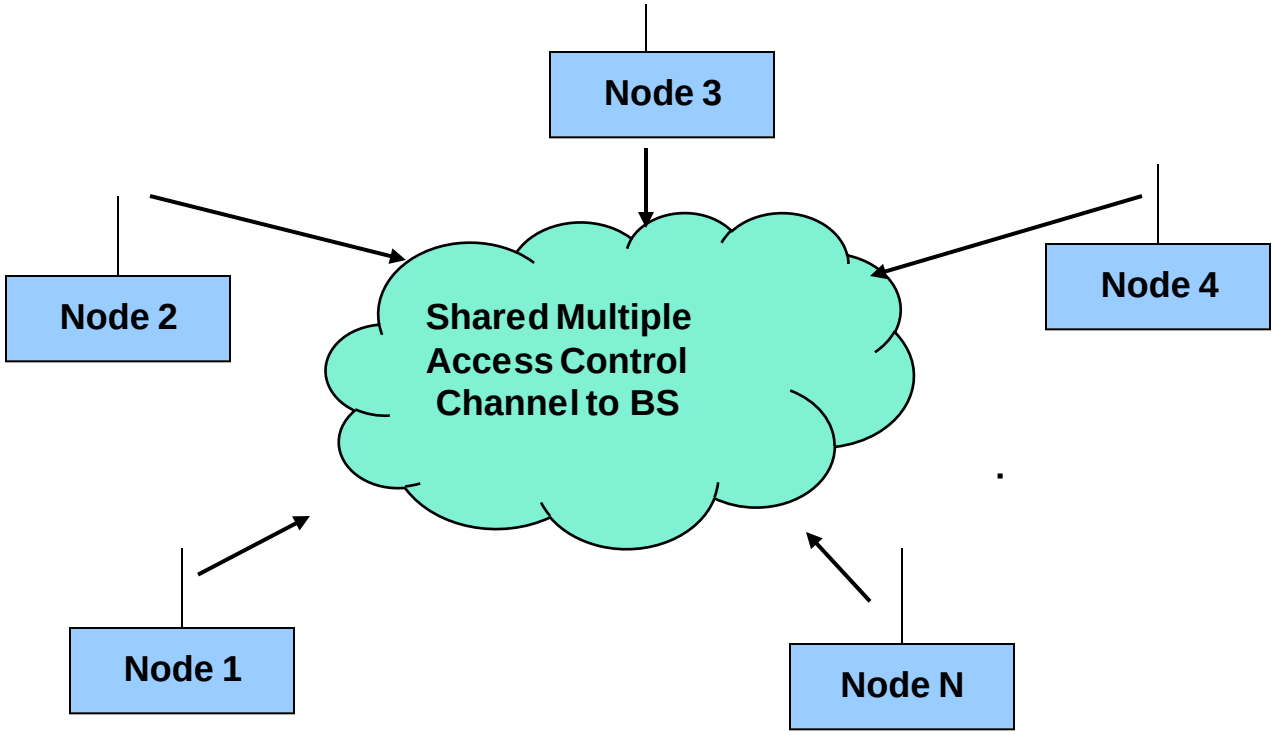
Example of piggybacking with error



Multiple Access

Introduction

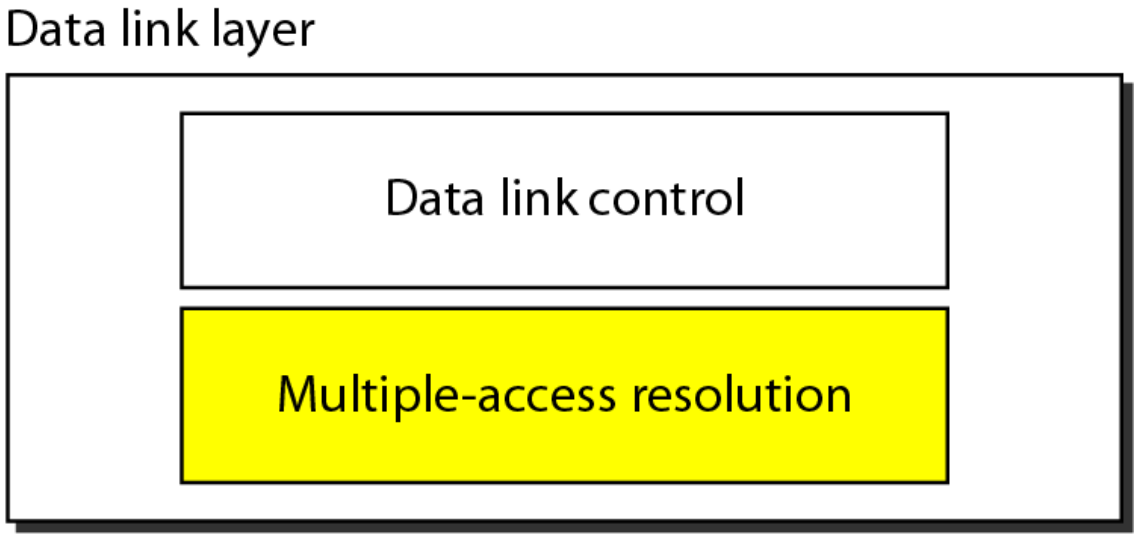
- Multiple access control channels
  - Each node is attached to a transmitter/receiver which communicates via a channel shared by other nodes
  - Transmission from any node is received by other nodes



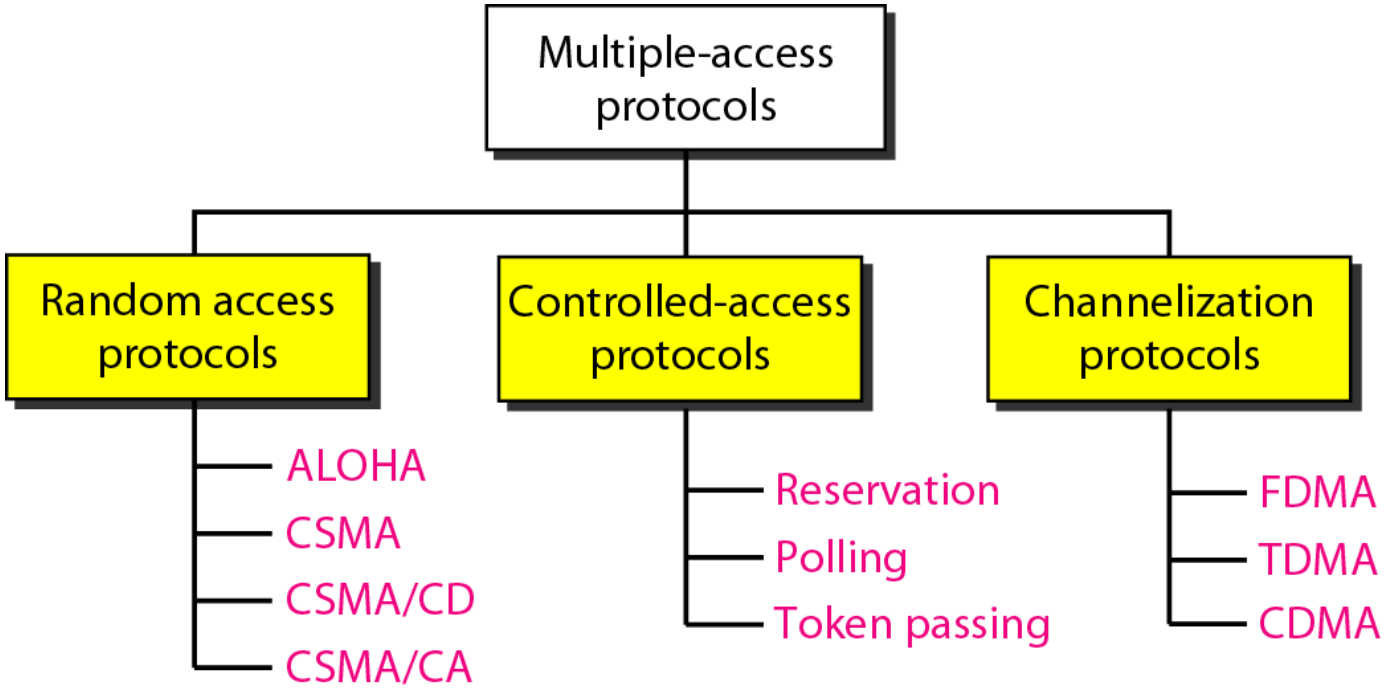
Introduction (Cont'd)

- Multiple access issues
  - If more than one node transmit at a time on the control channel to BS, a collision occurs
  - How to determine which node can transmit to BS?
- Multiple access protocols
  - Solving multiple access issues
  - Different types:
    - Contention protocols resolve a collision after it occurs. These protocols execute a collision resolution protocol after each collision
    - Collision-free protocols ensure that a collision can never occur.

Data link layer divided into two functionality-oriented sublayers



Taxonomy of multiple-access protocols



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.

ALOHA
Carrier Sense Multiple Access
Carrier Sense Multiple Access with Collision Detection
Carrier Sense Multiple Access with Collision Avoidance

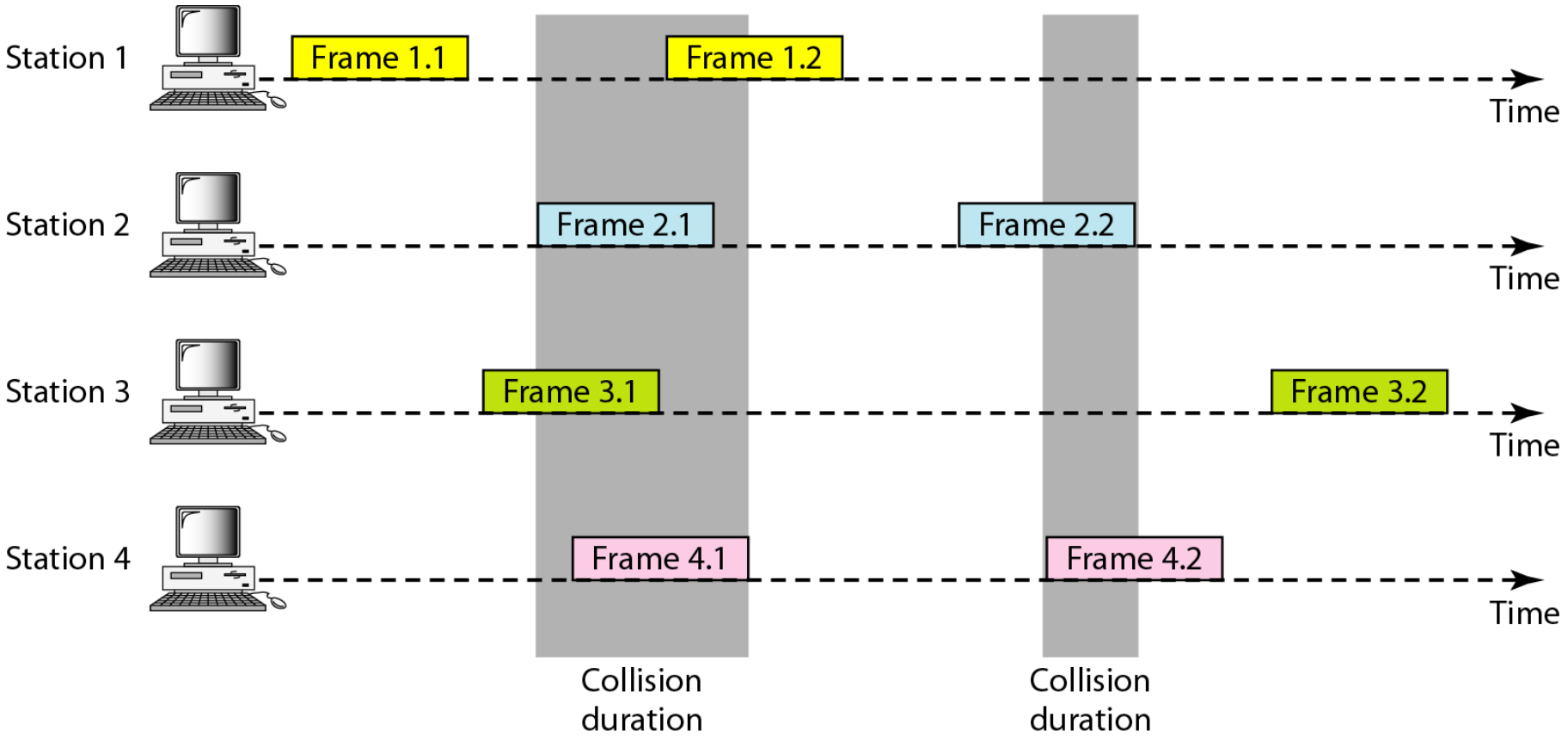
Contention Protocols

- **ALOHA**
  - Developed in the 1970s for a packet radio network by Hawaii University.
  - Whenever a station has a data, it transmits. Sender finds out whether transmission was successful or experienced a collision by listening to the broadcast from the destination station. Sender retransmits after some random time if there is a collision.
- **Slotted ALOHA**
  - Improvement: Time is slotted and a packet can only be transmitted at the beginning of one slot. Thus, it can reduce the collision duration.

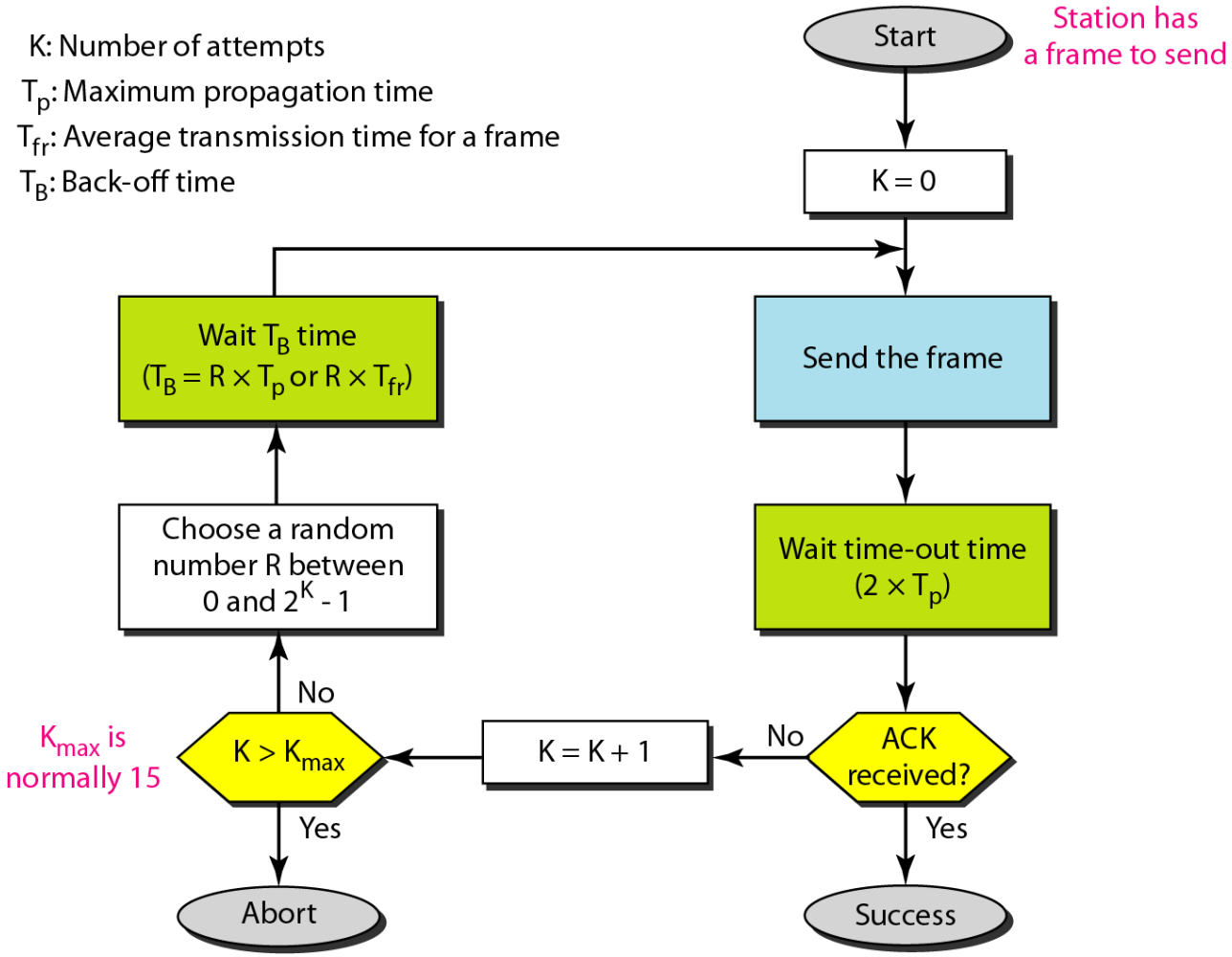
Contention Protocols (Cont'd)

- **CSMA** (Carrier Sense Multiple Access)
  - Improvement: Start transmission only if no transmission is ongoing
- **CSMA/CD** (CSMA with Collision Detection)
  - Improvement: Stop ongoing transmission if a collision is detected
- **CSMA/CA** (CSMA with Collision Avoidance)
  - Improvement: Wait a random time and try again when carrier is quiet. If still quiet, then transmit
- **CSMA/CA with ACK**
- **CSMA/CA with RTS/CTS**

Frames in a pure ALOHA network



Procedure for pure ALOHA protocol



Example 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, we find

$$T_p = (600 \times 10^5) / (3 \times 10^8) = 2 \text{ ms}.$$

Now we can find the value of T_B for different values of K .

a. For K = 1, the range is {0, 1}. The station needs to generate a random number with a value of 0 or 1. This means that T_B is either 0 ms (0 × 2) or 2 ms (1 × 2), based on the outcome of the random variable.

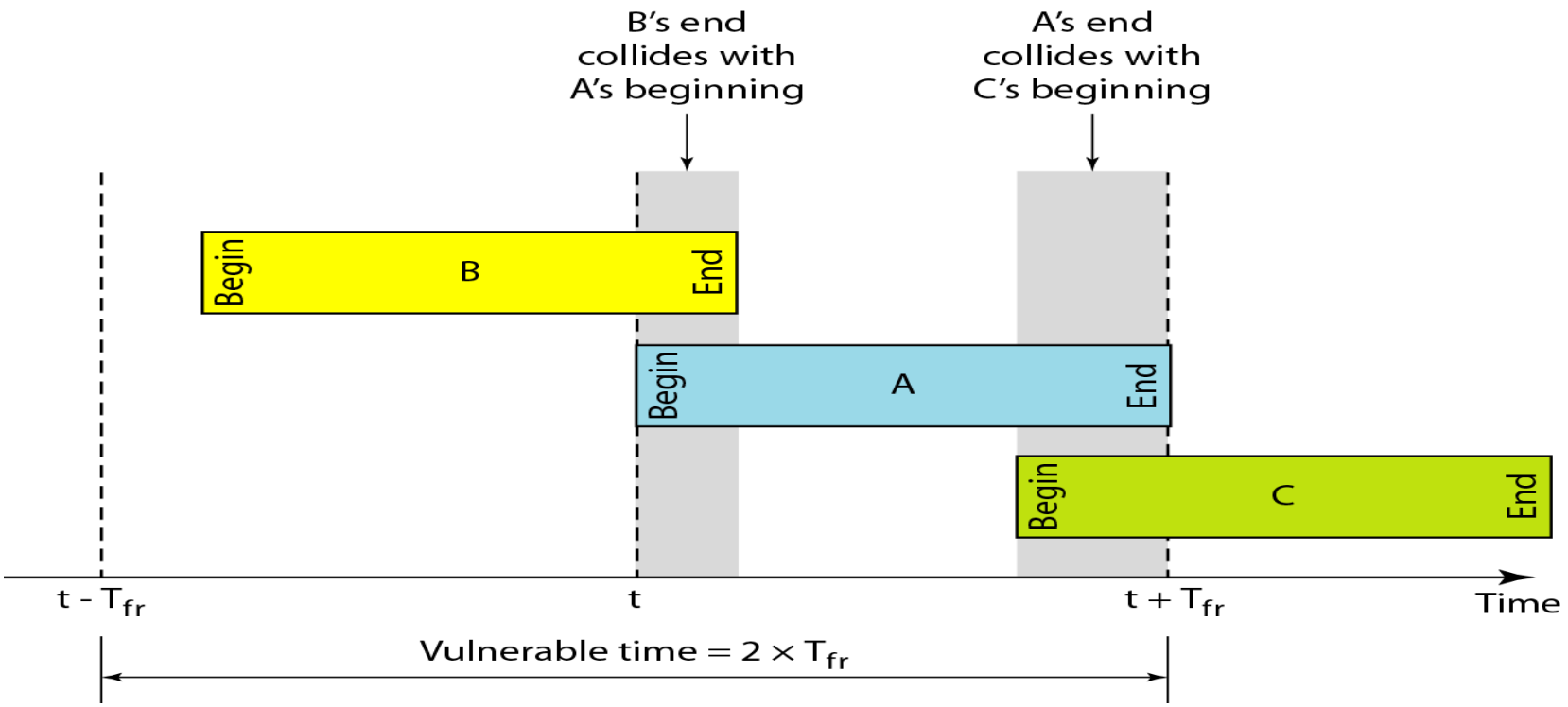
Example 1 (continued)

b. For K = 2, the range is {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.

c. For K = 3, the range is {0, 1, 2, 3, 4, 5, 6, 7}. This means that T_B can be 0, 2, 4, . . . , 14 ms, based on the outcome of the random variable.

d. We need to mention that if K > 10, it is normally set to 10.

Vulnerable time for pure ALOHA protocol



Vulnerable time –Signifies the Probability of frame collision

Example 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 2×1 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 one 1-ms period that this station is sending.

Throughput of ALOHA

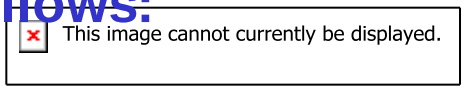
- The probability that n packets arrive in two packets time is given by

$$P(n) = \frac{(2G)^n e^{-2G}}{n!}$$

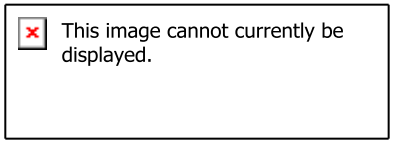
where G is traffic load.

- The probability P(0) that a packet is successfully received without collision is calculated by letting n=0 in the above equation. We get

- We can calculate throughput S with a traffic load G as follows:



- The Maximum throughput of ALOHA is



Note

The throughput for pure ALOHA is

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

The maximum throughput

$$S_{max} = 0.184$$ **when G= (1/2).**

Example 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

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 $S = G \times e^{-2G}$ or $S = 0.135$ (13.5 percent). This means that the throughput is $1000 \times 0.135 = 135$ frames. Only 135 frames out of 1000 will probably survive.

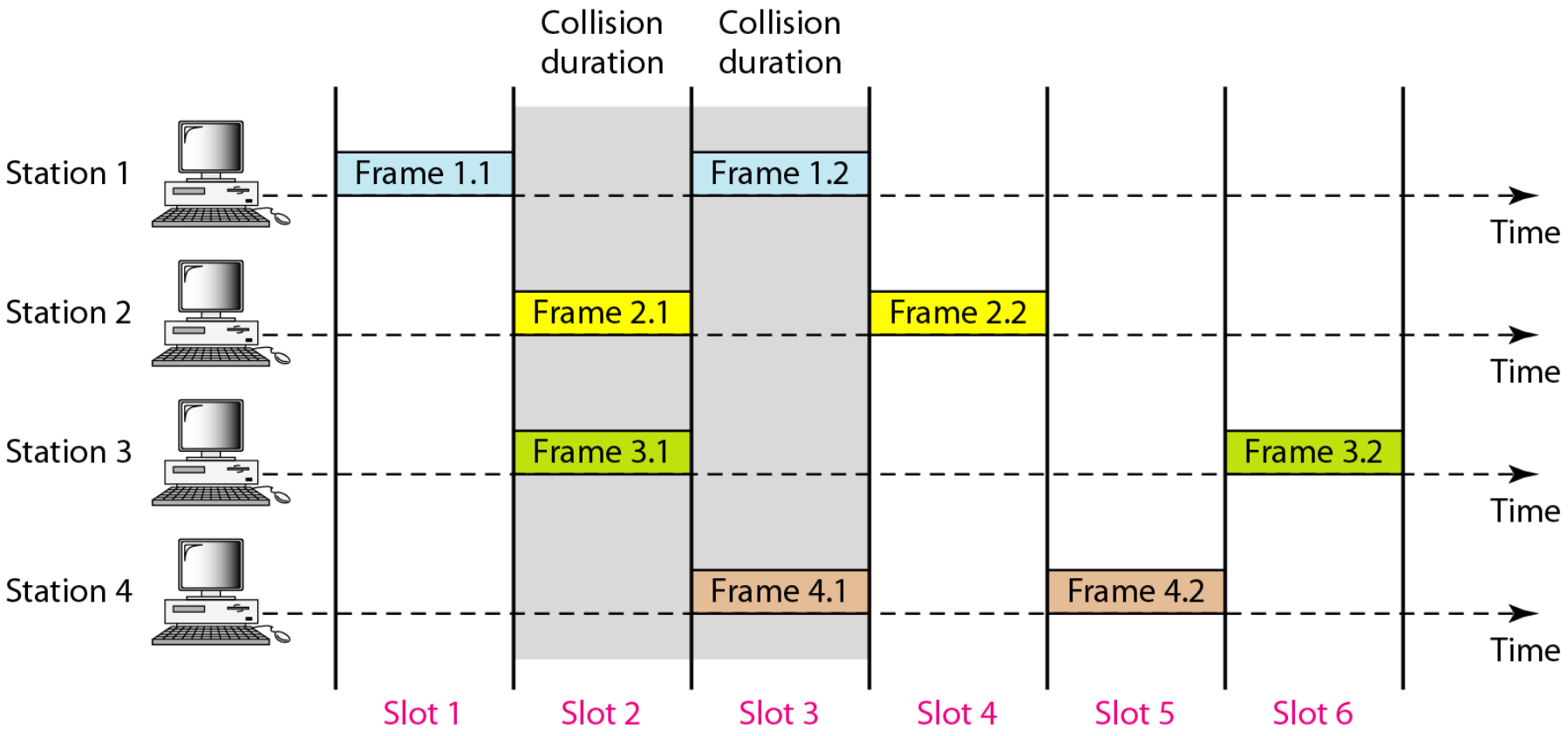
Example 3 (continued)

b. If the system creates 500 frames per second, this is (1/2) frame per millisecond. The load is (1/2). In this case $S = G \times e^{-2G}$ or $S = 0.184$ (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, 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 $S = G \times e^{-2G}$ or $S = 0.152$ (15.2 percent). This means that the throughput is $250 \times 0.152 = 38$. Only 38 frames out of 250 will probably survive.

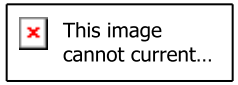
Slotted ALOHA

Frames in a slotted ALOHA network

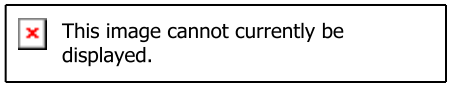


Throughput of Slotted ALOHA

- **The probability of no collision is given by**



- **The throughput S is**



- **The Maximum throughput of slotted ALOHA is**

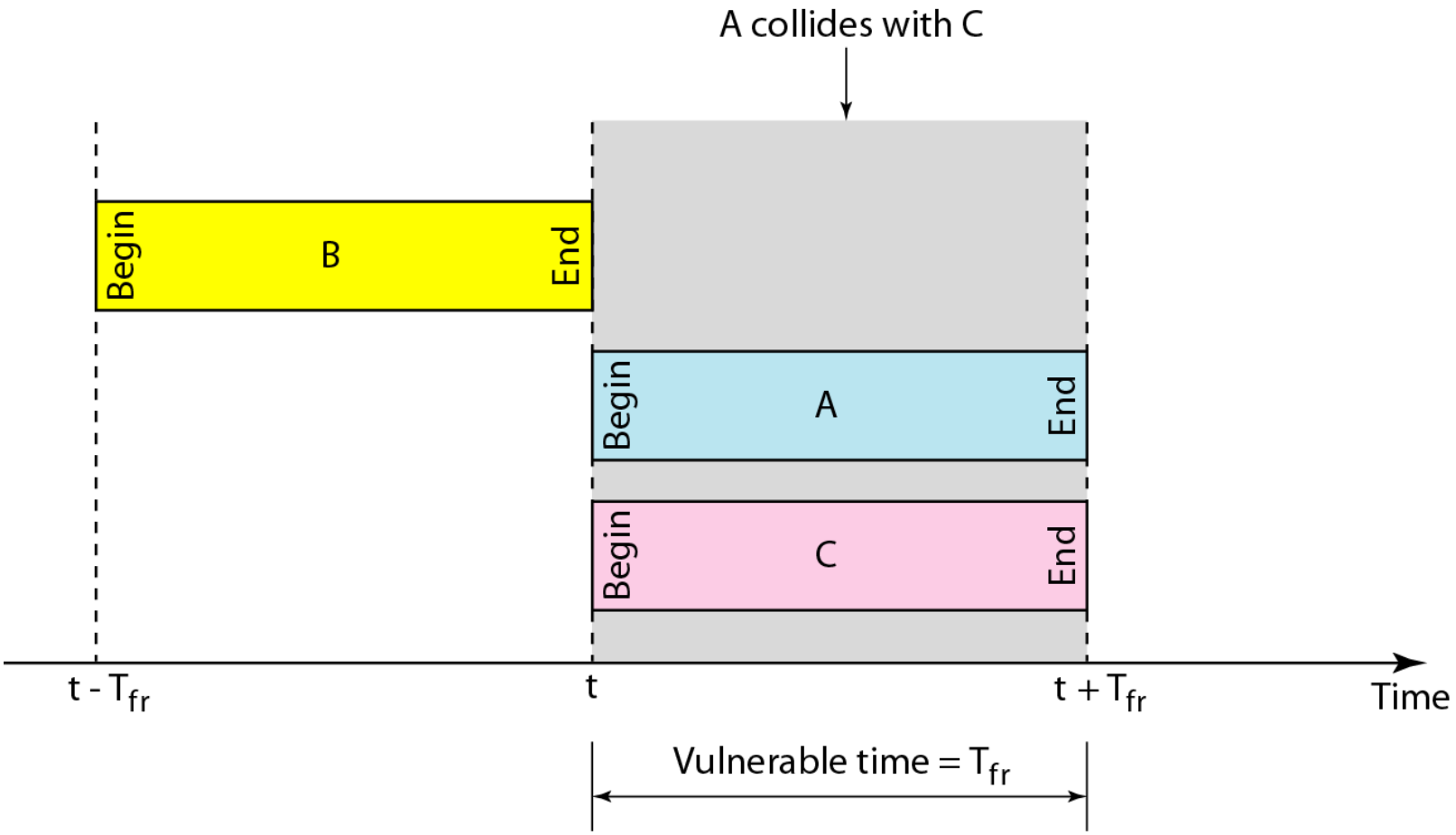
$$S_{\max} = \frac{1}{e} \approx 0.368$$

Note

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

The maximum throughput $S_{max} = 0.368$ when G = 1.

Vulnerable time for slotted ALOHA protocol



Example 4

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

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.

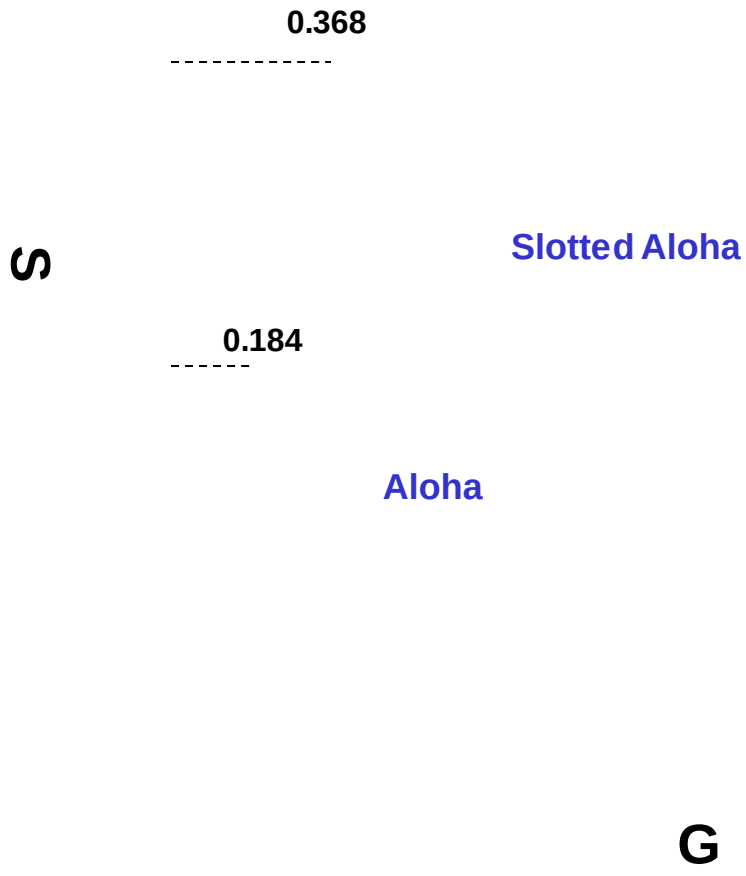
a. If the system creates 1000 frames per second, this is 1 frame per millisecond. The load is 1. In this case $S = G \times e^{-G}$ or $S = 0.368$ (36.8 percent). This means that the throughput is $1000 \times 0.0368 = 368$ frames. Only 386 frames out of 1000 will probably survive.

Example 4 (continued)

b. If the system creates 500 frames per second, this is (1/2) frame per millisecond. The load is (1/2). In this case $S = G \times e^{-G}$ or $S = 0.303$ (30.3 percent). This means that the throughput is $500 \times 0.0303 = 151$. Only 151 frames out of 500 will probably survive.

c. If the system creates 250 frames per second, this is (1/4) frame per millisecond. The load is (1/4). In this case $S = G \times e^{-G}$ or $S = 0.195$ (19.5 percent). This means that the throughput is $250 \times 0.195 = 49$. Only 49 frames out of 250 will probably survive.

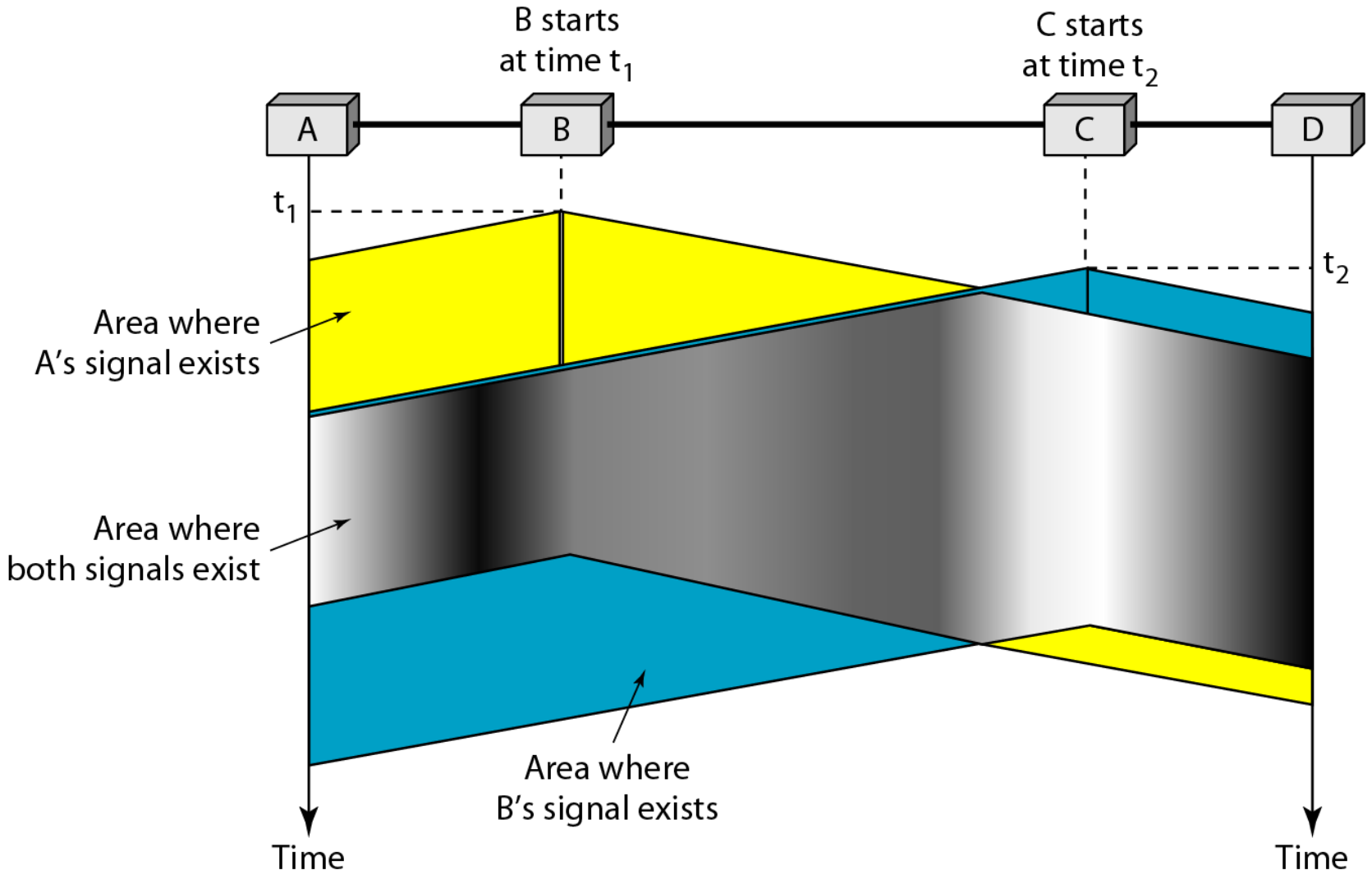
Throughput



Carrier Sense Multiple Access CSMA

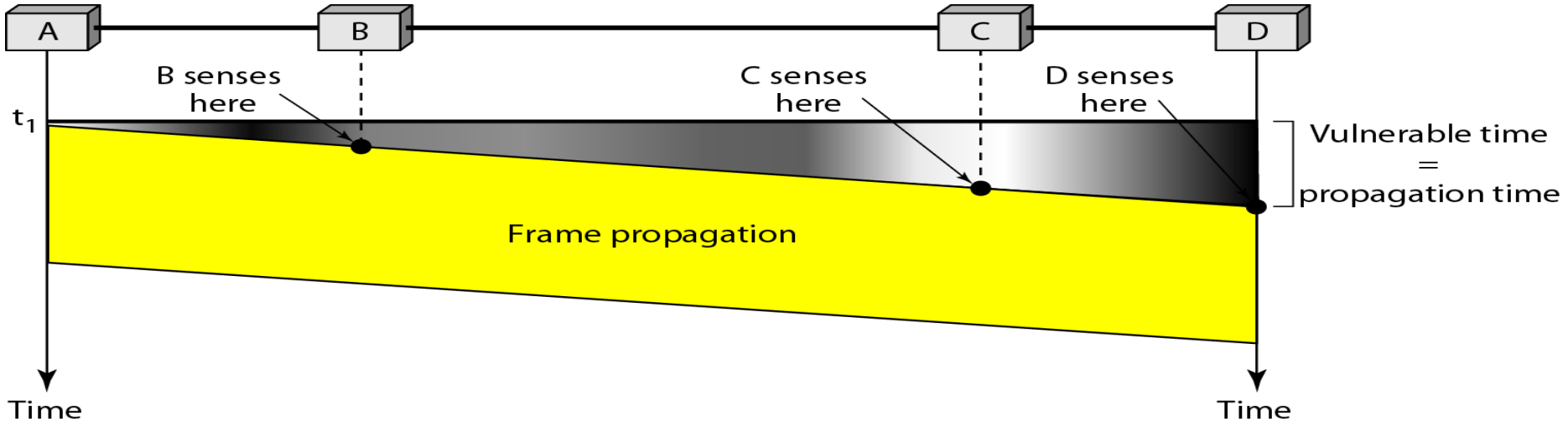
- Senses the medium before trying to use it.
- Sense-check the state of the medium-busy/free
- It reduces the possibility of the collision. But it can’t eliminate the collisions.

Space/time model of the collision in CSMA



Vulnerable time in CSMA

Vulnerable time in CSMA is propagation tim



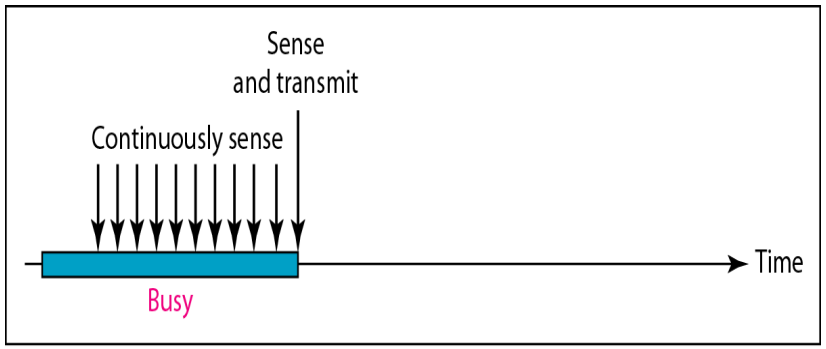
What should a station do if channel is Busy?

What should a station do if channel is idle?

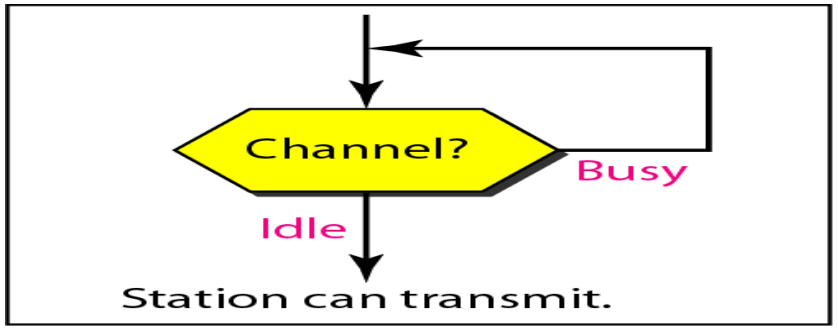
CSMA- persistent methods

- 1- persistent
- Non-persistent
- p-persistent

Behavior of 1-persistence method



a. 1-persistent



a. 1-persistent

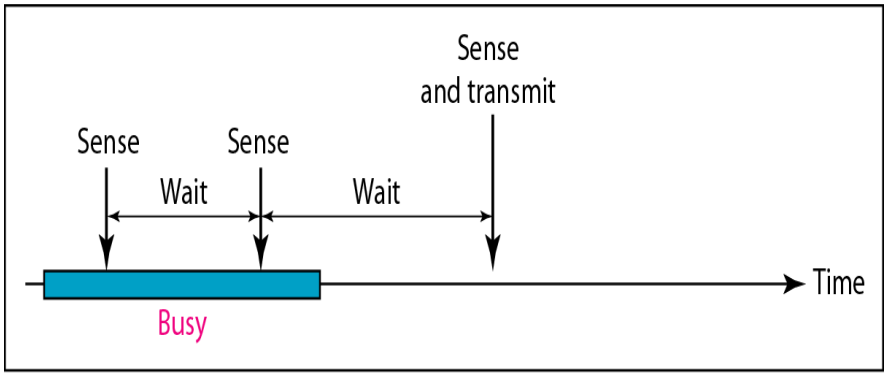
1-persistent CSMA Protocol:

Step 1: If the medium is idle, transmit immediately

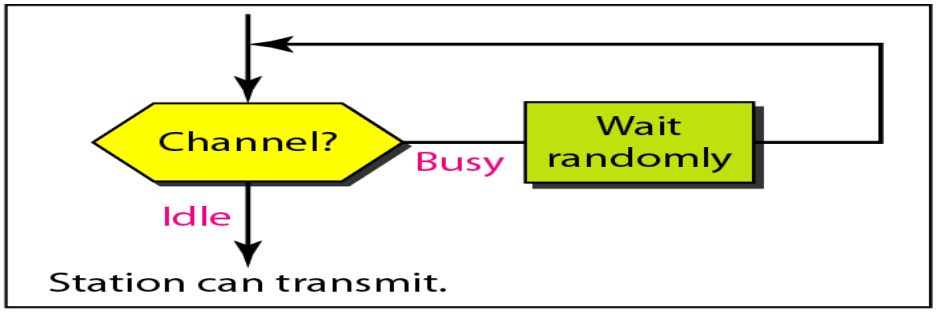
Step 2: If the medium is busy, continue to listen until medium becomes idle, and then transmit immediately.

- There will always be a collision if two nodes want to retransmit

Behavior of non-persistence method



b. Nonpersistent



b. Nonpersistent

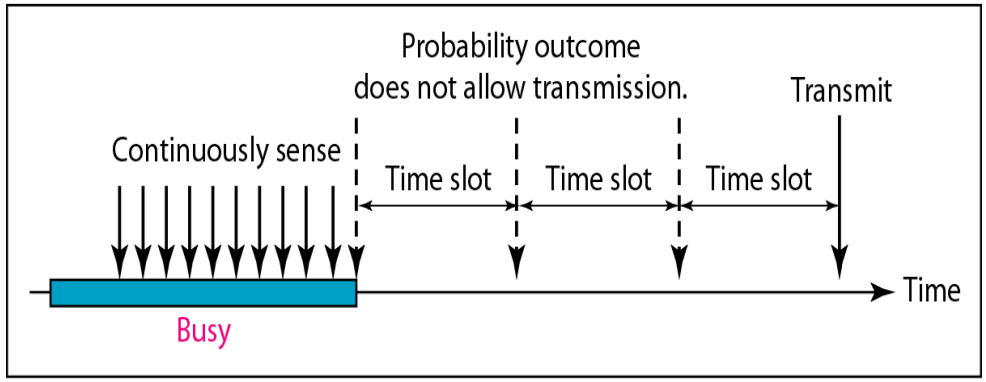
Nonpersistent CSMA Protocol:

Step 1: If the medium is idle, transmit immediately.

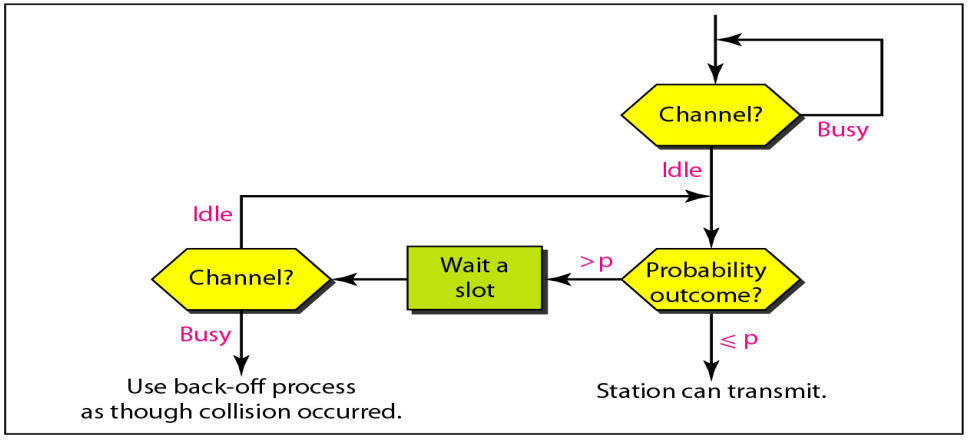
Step 2: If the medium is busy, wait a random amount of time and repeat **Step 1**

- Random backoff reduces probability of collisions
- Waste idle time if the backoff time is too long

Behavior of p-persistence method



c. p-persistent



c. p-persistent

p-persistent CSMA Protocol:

Step 1: If the medium is idle, transmit with probability p, and delay for worst case propagation delay for one packet with probability (1-p).

Step 2: If the medium is busy, continue to listen until medium becomes idle, then go to **Step 1**

Step 3: If transmission is delayed by one time slot, continue with **Step 1**

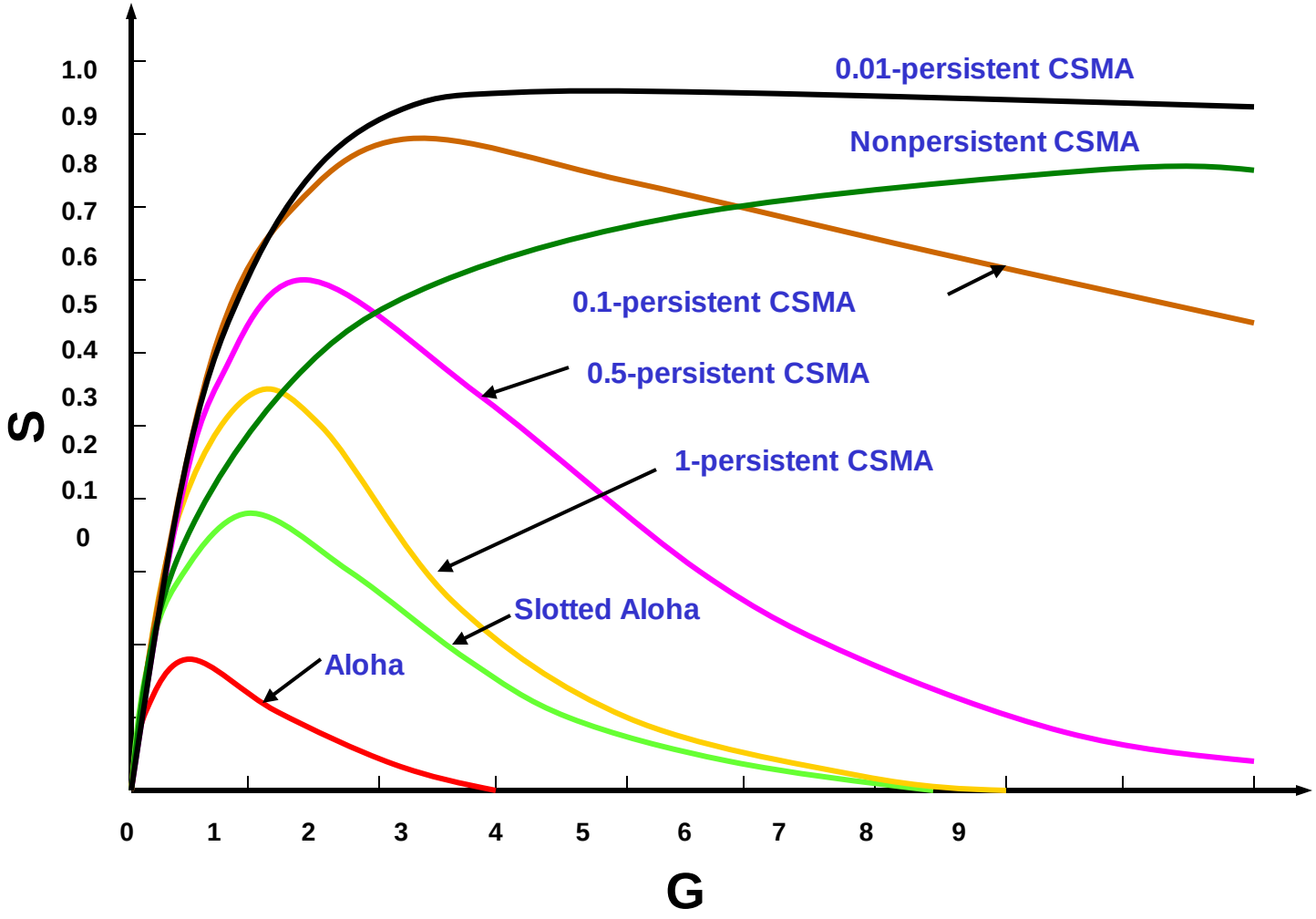
- A good tradeoff between nonpersistent and 1-persistent CSMA

How to Select Probability p ?

- Assume that N nodes have a packet to send and the medium is busy.
- Then, Np is the expected number of nodes that will attempt to transmit once the medium becomes idle.
- If Np > 1, then a collision is expected to occur

Therefore, network must make sure that Np < 1 to avoid collision, where N is the maximum number of nodes that can be active at a time

Throughput


0.01-persistent CSMA
Nonpersistent CSMA
0.1-persistent CSMA
0.5-persistent CSMA
1-persistent CSMA
Slotted Aloha
Aloha
S
G
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
1
2
3
4
5
6
7
8
9

Carrier Sense Multiple Access Collision Detection CSMA/CD

- In CSMA, if 2 terminals begin sending packet at the same time, each will transmit its complete packet (although collision is taking place).
- Wasting medium for an entire packet time.

Monitors the channel after it sends a frame to see if the transmission was successful.

- CSMA/CD

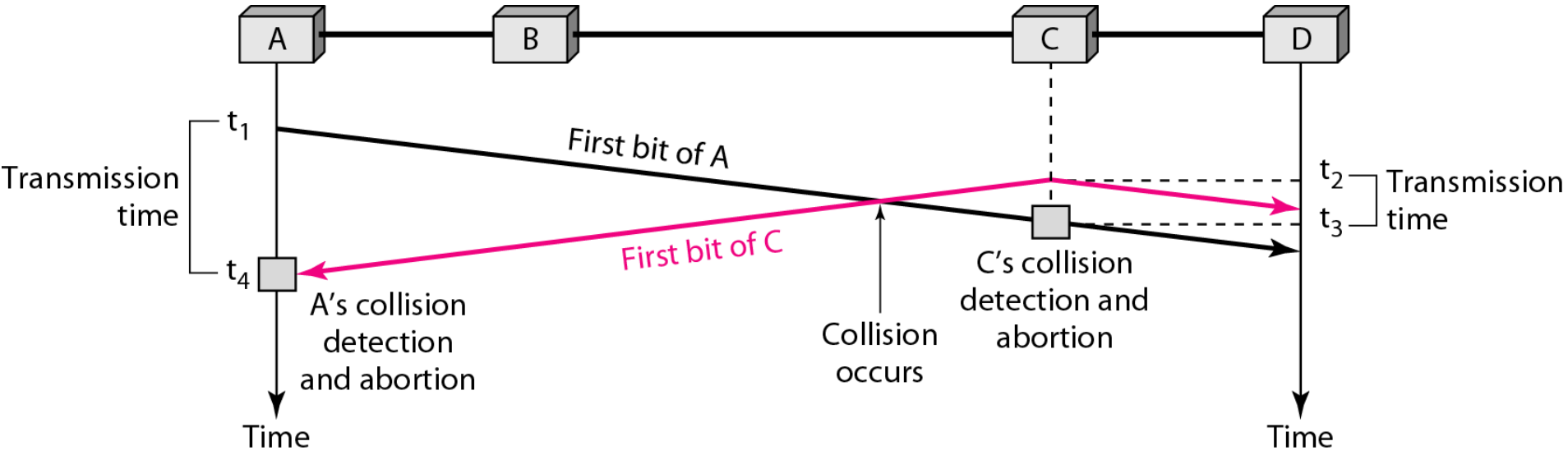
  Step 1: If the medium is idle, transmit.

  Step 2: If the medium is busy, continue to listen until the channel is idle then transmit.

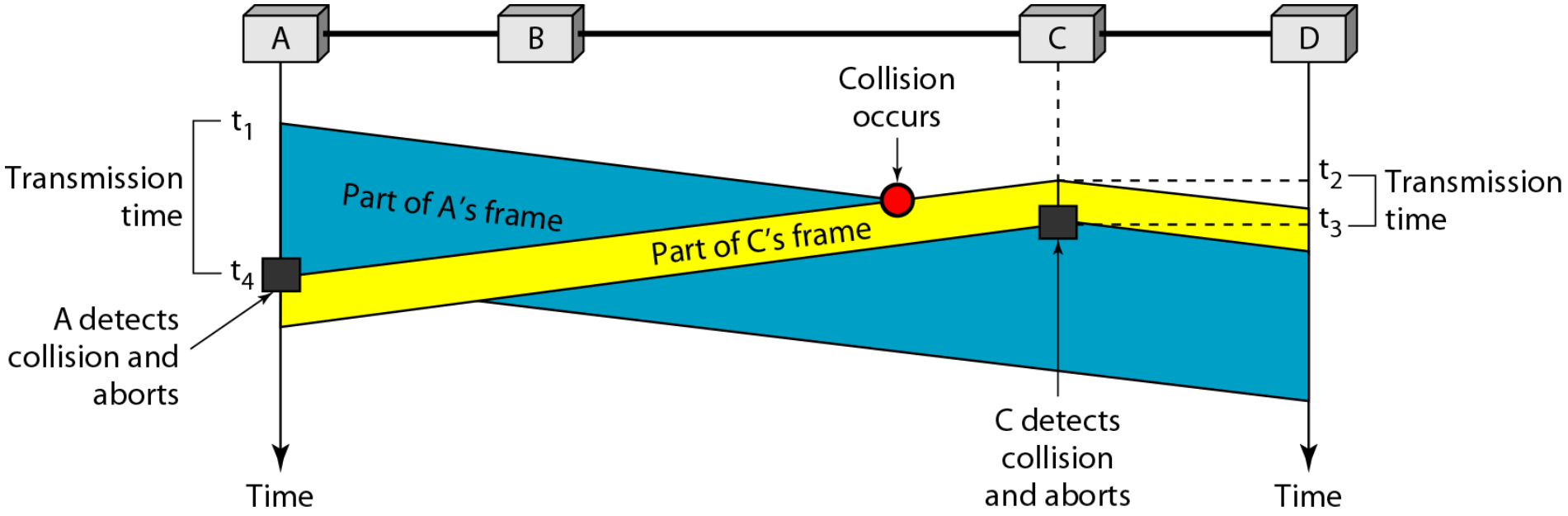
  Step 3: If a collision is detected during transmission, cease transmitting.

  Step 4: Wait a random amount of time and repeats the same algorithm.

Collision of the first bit in CSMA/CD



Collision and abortion in CSMA/CD



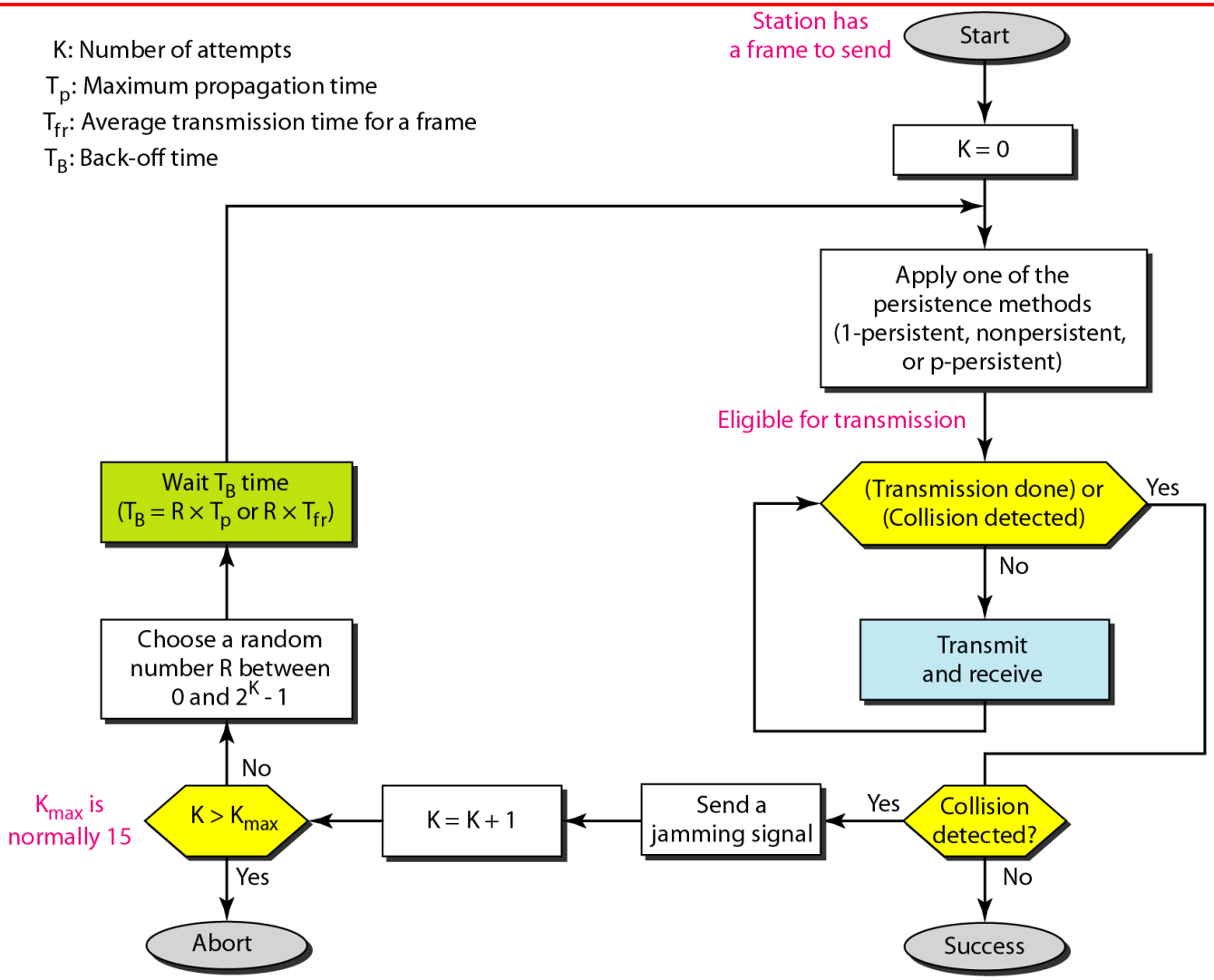
Example 5

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 25.6 μs, what is the minimum size of the frame?

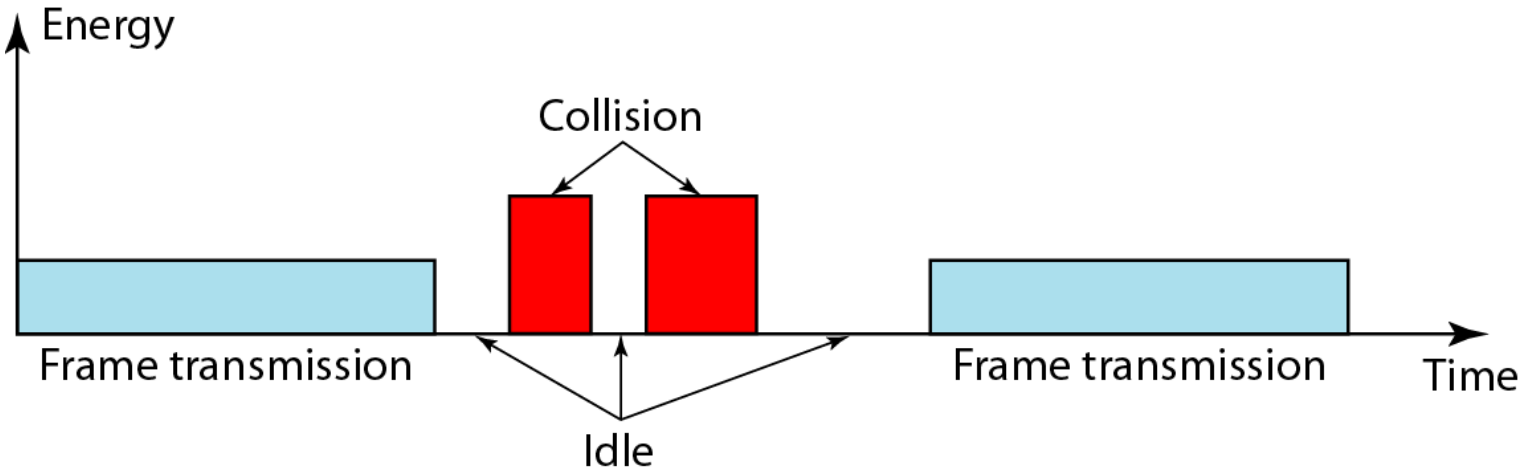
Solution

The frame transmission time is $T_{fr} = 2 \times T_p = 51.2$ 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 × 51.2 μs = 512 bits or 64 bytes. This is actually the minimum size of the frame for Standard Ethernet.

Flow diagram for the CSMA/CD



Energy level during transmission, idleness, or collision

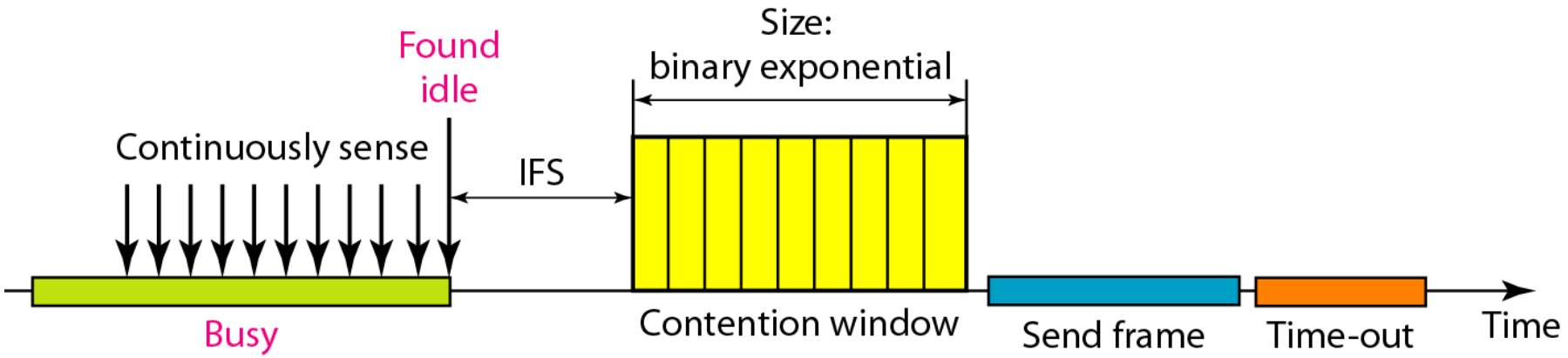


Carrier Sense Multiple Access Collision Avoidance CSMA/CA

CSMA/CA

- Avoid the collision.
- Suitable for wireless network, where collision detection is not possible because most of the energy is lost in transmission.
- Collision is avoided using
  - Interframe space
  - Contention window
  - Acknowledgements

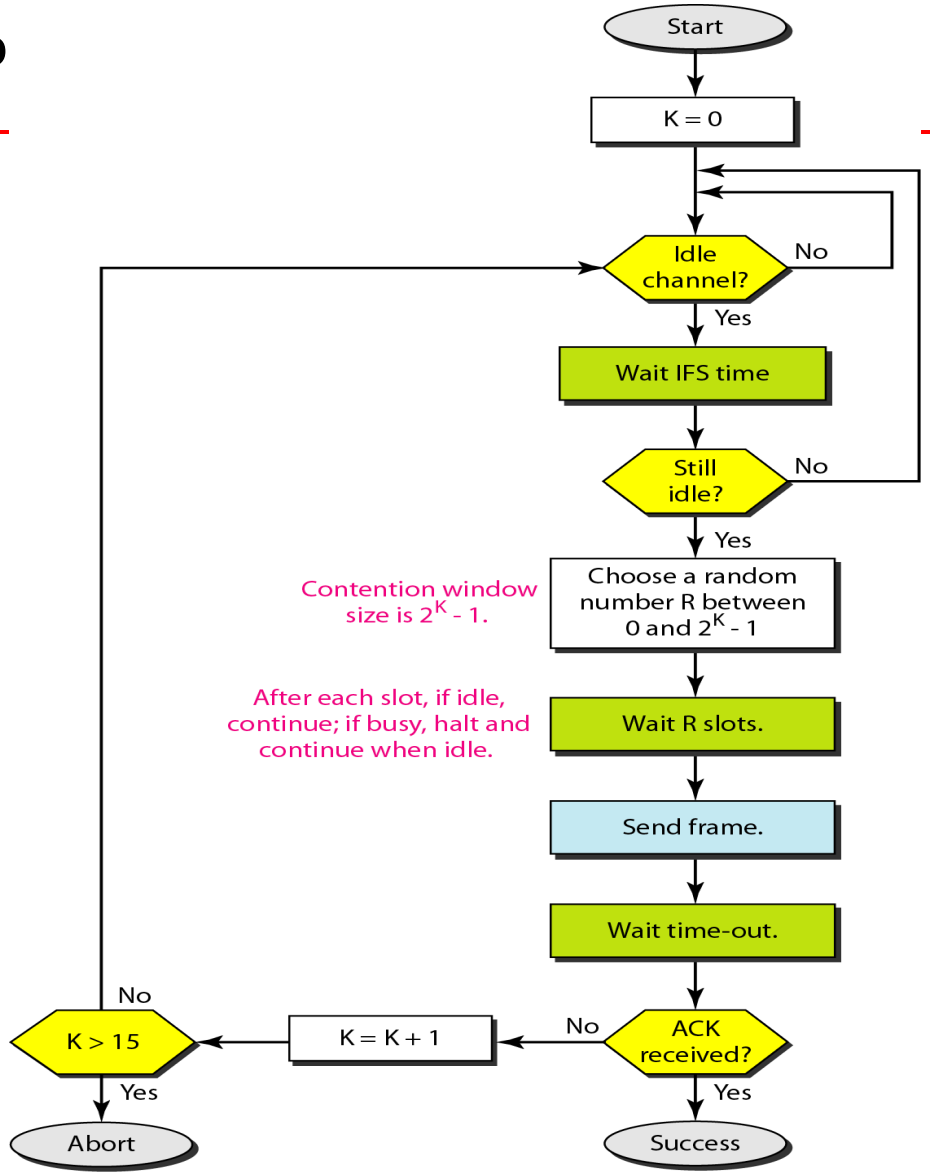
Timing in CSMA/CA



Note

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

Flow diagram fo



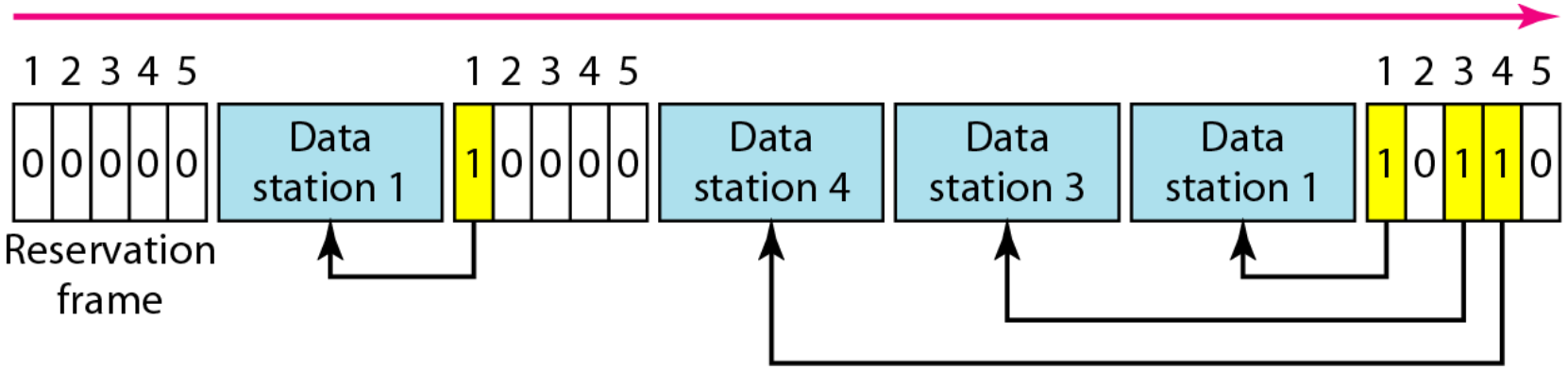
CONTROLLED ACCESS

In controlled access, the 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. We discuss three popular controlled-access methods.

- **Reservation**
- **Polling**
- **Token Passing**

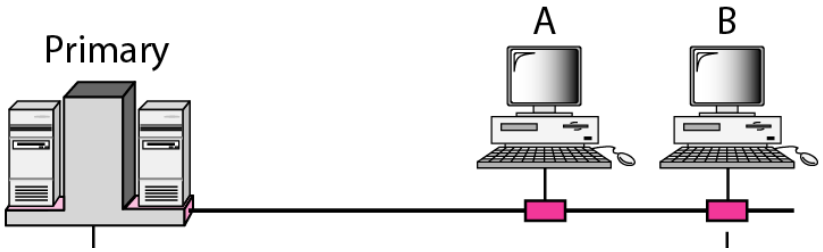
Reservation

Reservation access method



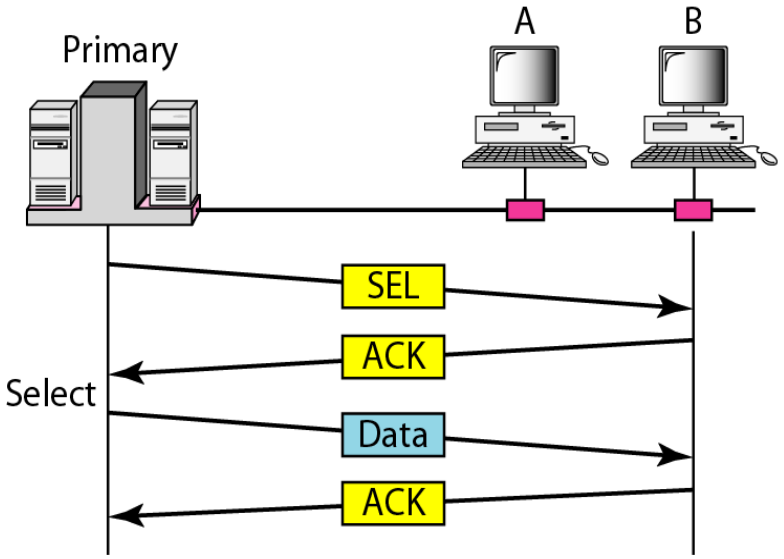
Polling

Polling



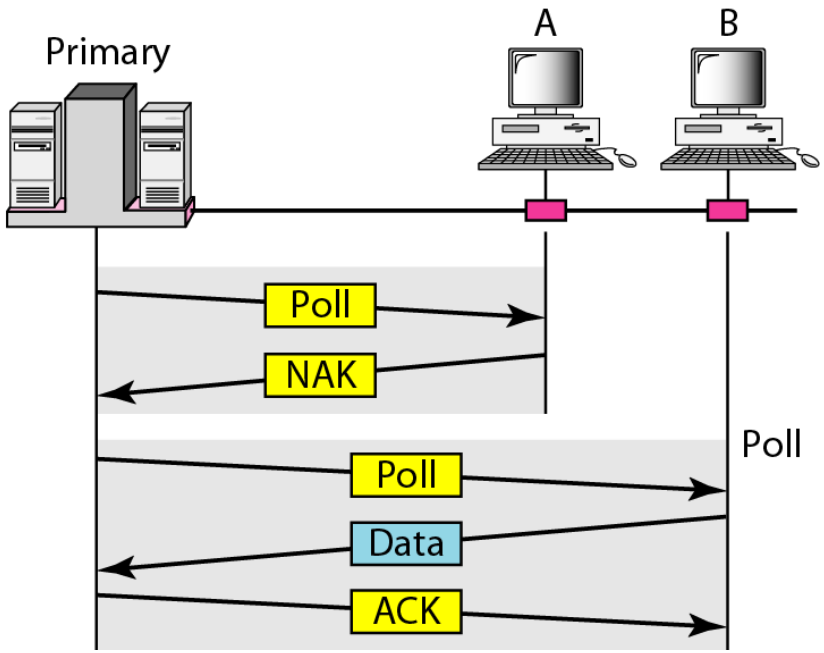
- Data exchange is made through primary device.
- Primary device controls the link; secondary devices follow its instructions.
- Primary device is always the initiator.

Select Function -Primary sending data



- Select function is used whenever primary wants to send data.
- Sends SEL frame which contains address of the destination

Poll functions: Primary receives data



- Ask each device in turn whether they have data to send.
- NAK-if nothing to send.
- After getting NAK primary polls the next device.

Token Passing

Stations are organized in logical ring.

For each station there is

predecessor(logical previous)

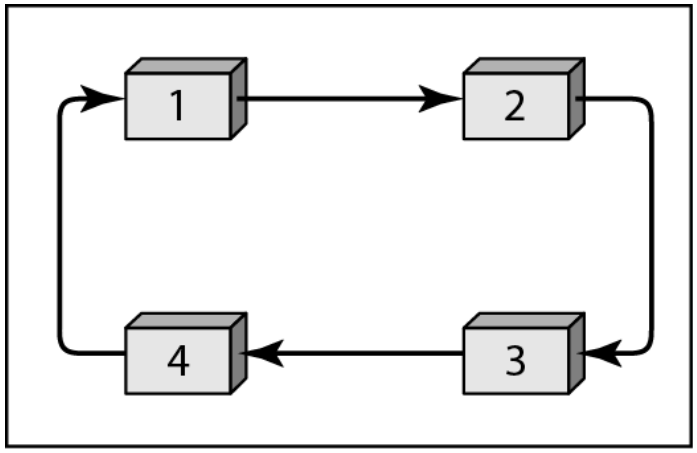
successor (logical next)

Right to access the channel given by predecessor and passed on to successor.

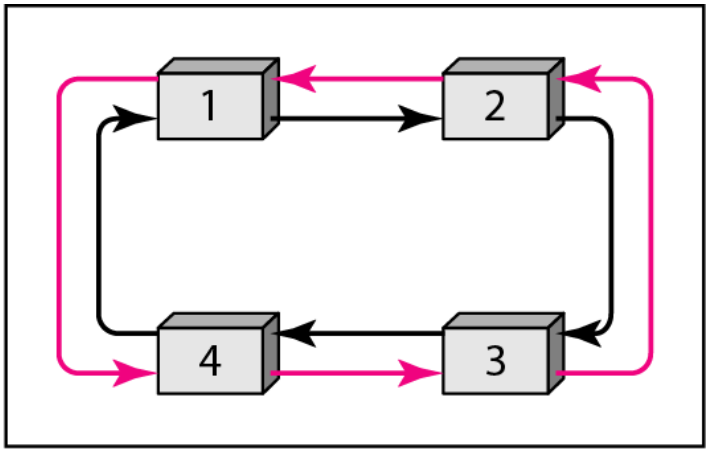
Special packet called Token is passed in the network. It circulates through the network.

When station has some data to send, should wait for the Token.

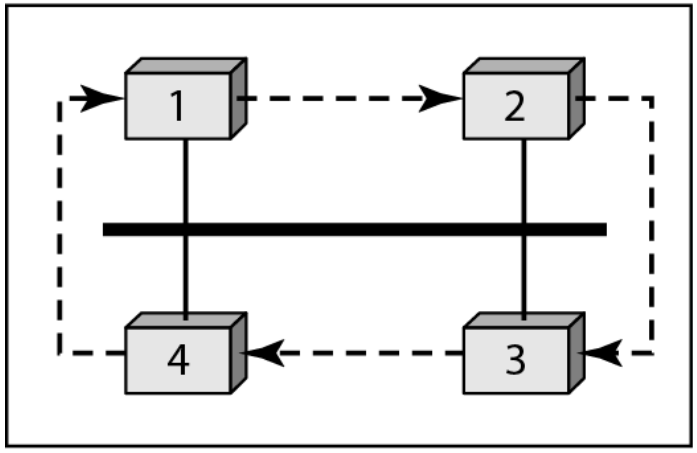
Figure 12.20 Logical ring and physical topology in token-passing acces



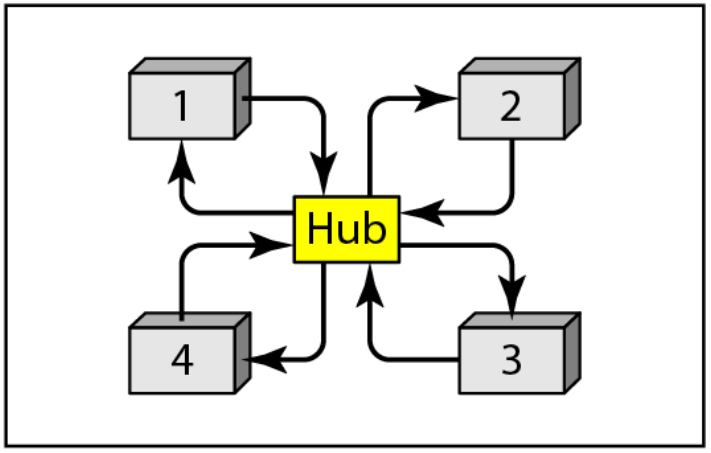
a. Physical ring



b. Dual ring



c. Bus ring



d. Star ring

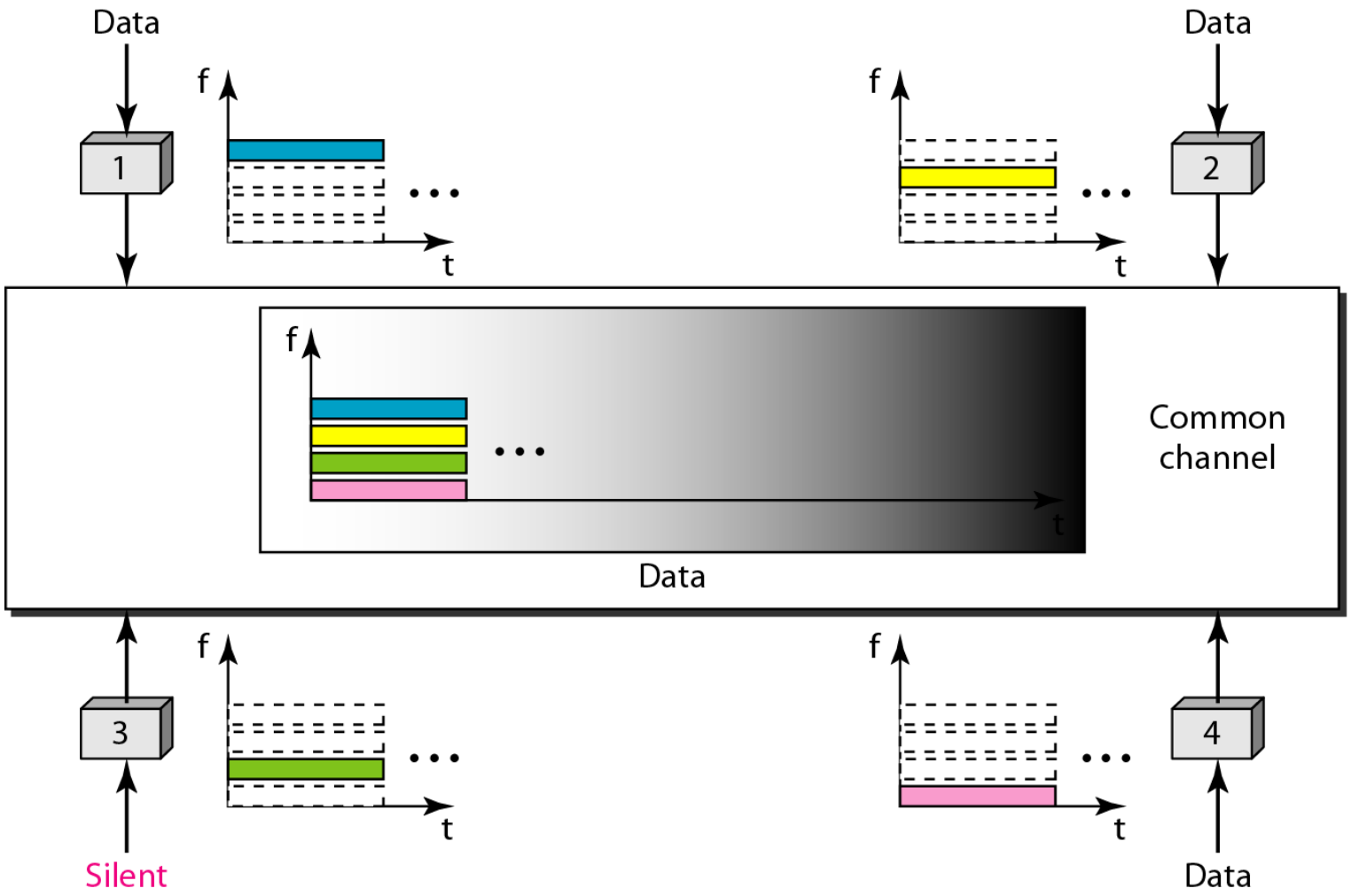
CHANNELIZATION

Channelization is a multiple-access method in which the available bandwidth of a link is shared in time, frequency, or through code, between different stations. In this section, we discuss three channelization protocols.

- **Frequency-Division Multiple Access (FDMA)**
- **Time-Division Multiple Access (TDMA)**
- **Code-Division Multiple Access (CDMA)**

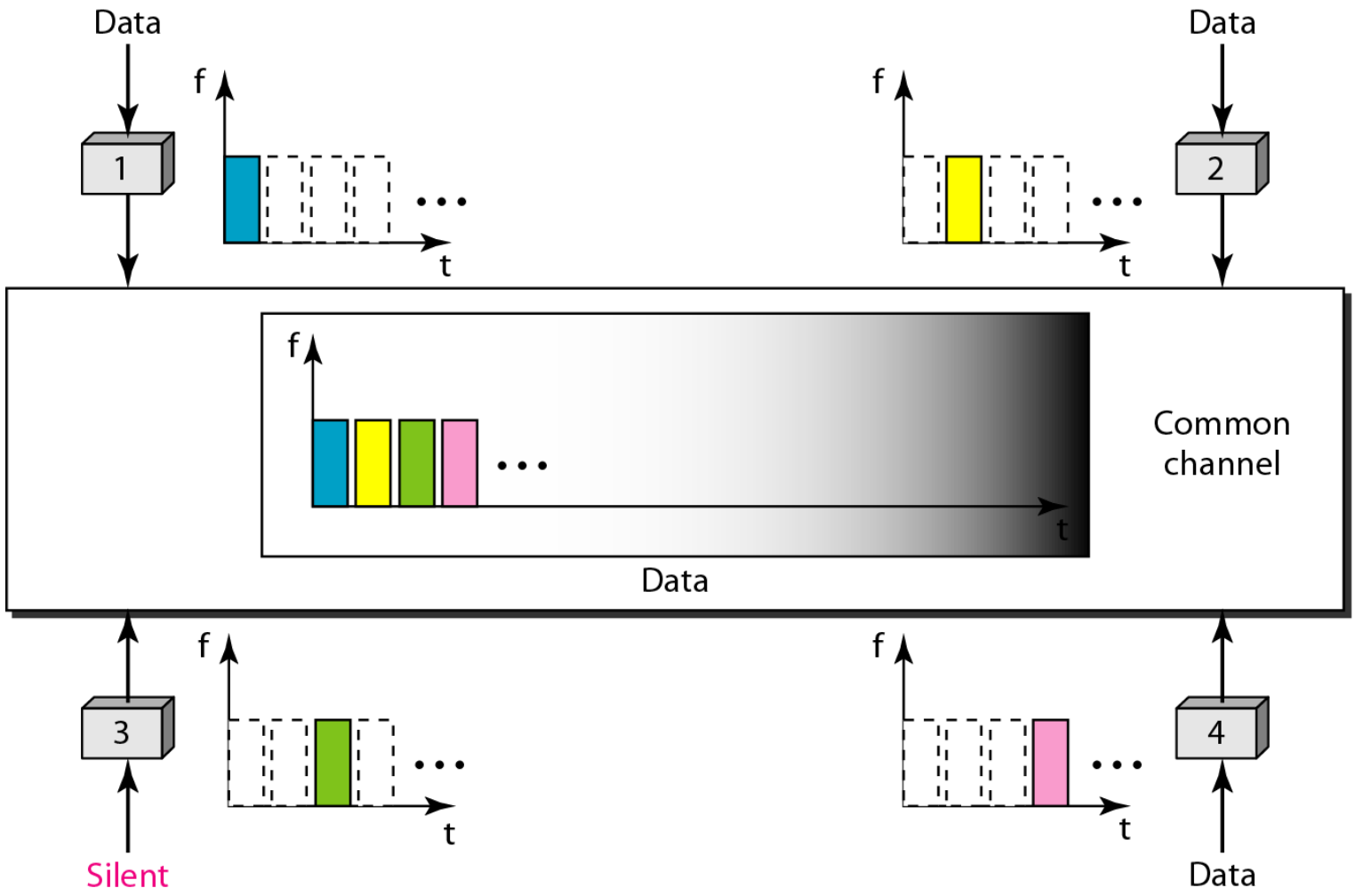
Frequency Division Multiple Access

Frequency-division multiple access (FDMA)



Time Division Multiple Access

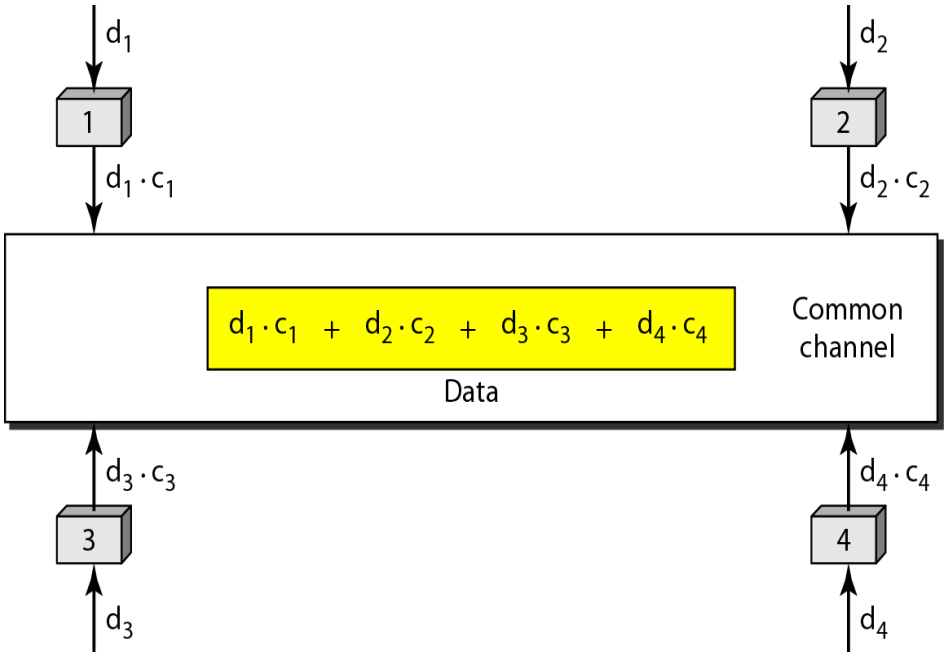
Time-division multiple access (TDMA)



Code Division Multiple Access

- One channel carries all the transmissions simultaneously.
- Differs from FDMA in the sense that, all transmissions are carried on a single frequency.
- Differs from TDMA in the sense that, all transmissions are simultaneous.
- Each station is assigned a code, which is sequence of numbers called ***chips***

Simple idea of communication with code

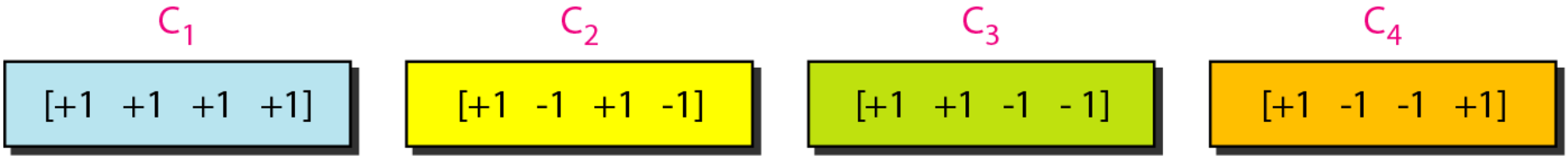


Code Properties

- Multiply different codes get 0
- Multiply same codes we get N
  - N-no of stations in the network

Data on the channel is the sum of all the codes

Chip sequences



Properties

- **If we multiply sequence by any number, each element is multiplied by that number.**

  2. [+1 +1 -1 -1]= [+2 +2 -2 -2]

- **If we multiply two equal sequences element by element, we get number of Stations N**

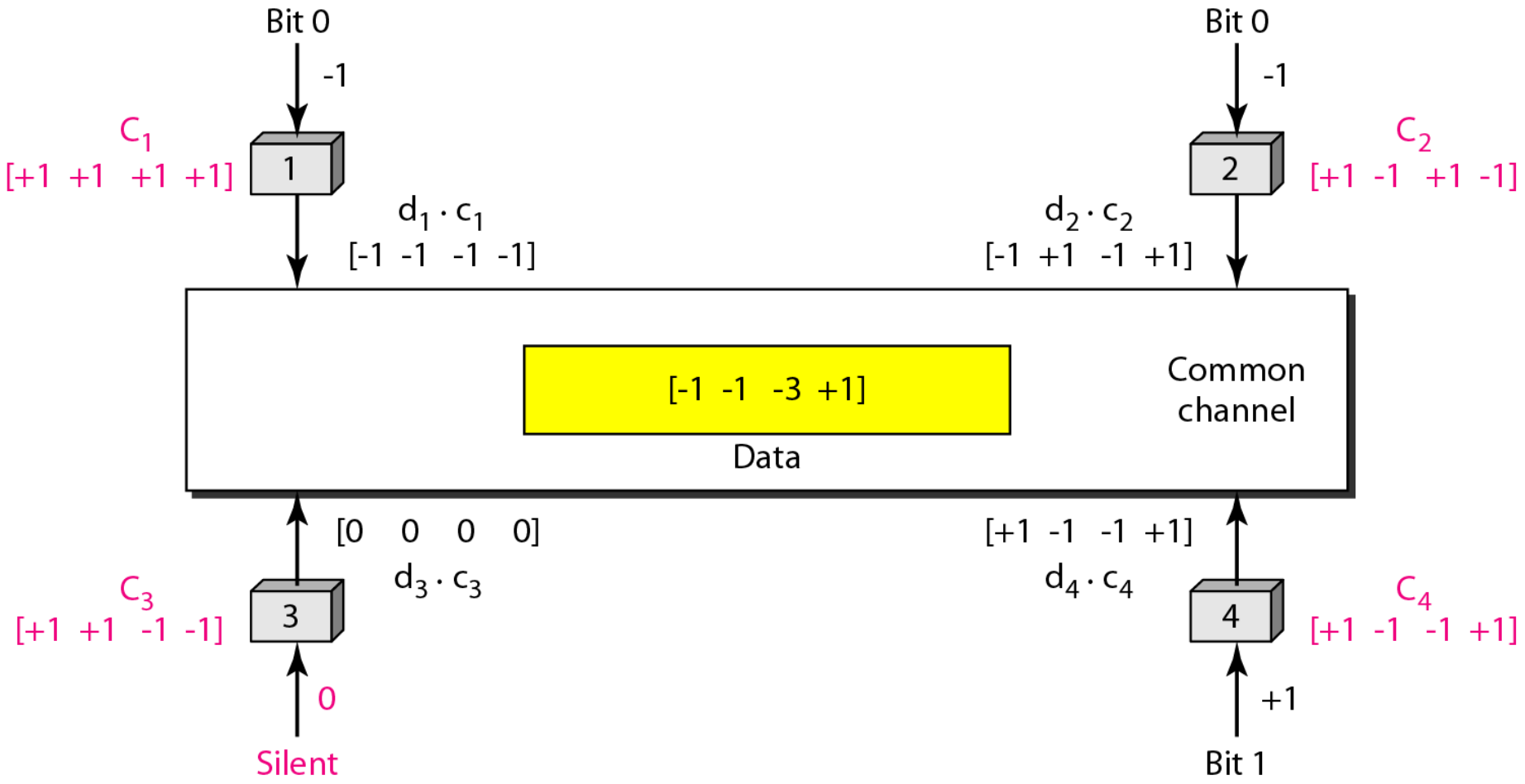
  [+1 +1 -1 -1] . [+1 +1 -1 -1] =1+1+1+1=4

- **If we multiply two different sequences element by element, we get 0**

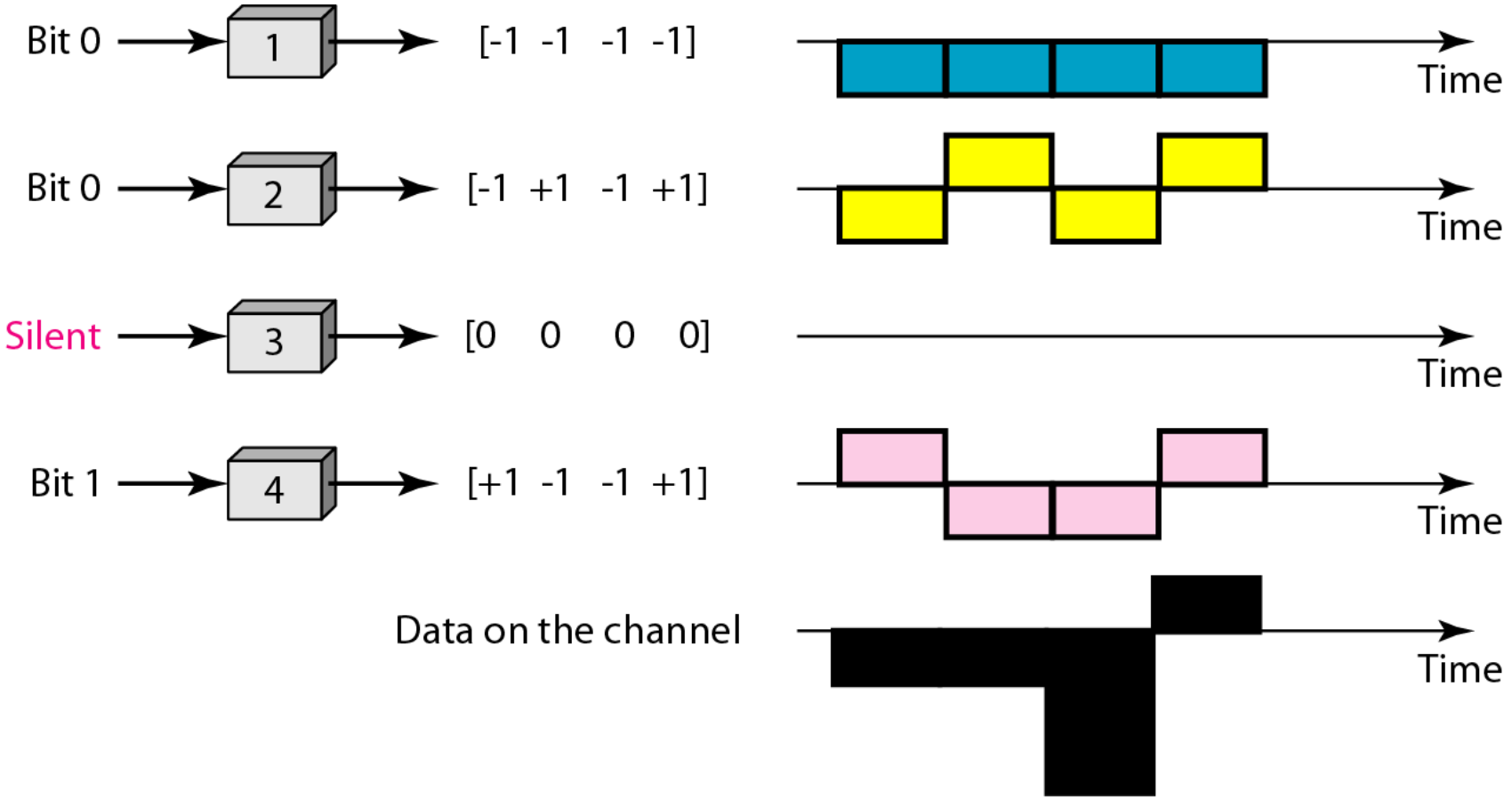
  [+1 +1 -1 -1] . [+1 +1 +1 +1]=+1+1-1-1=0

Data representation in CDMA

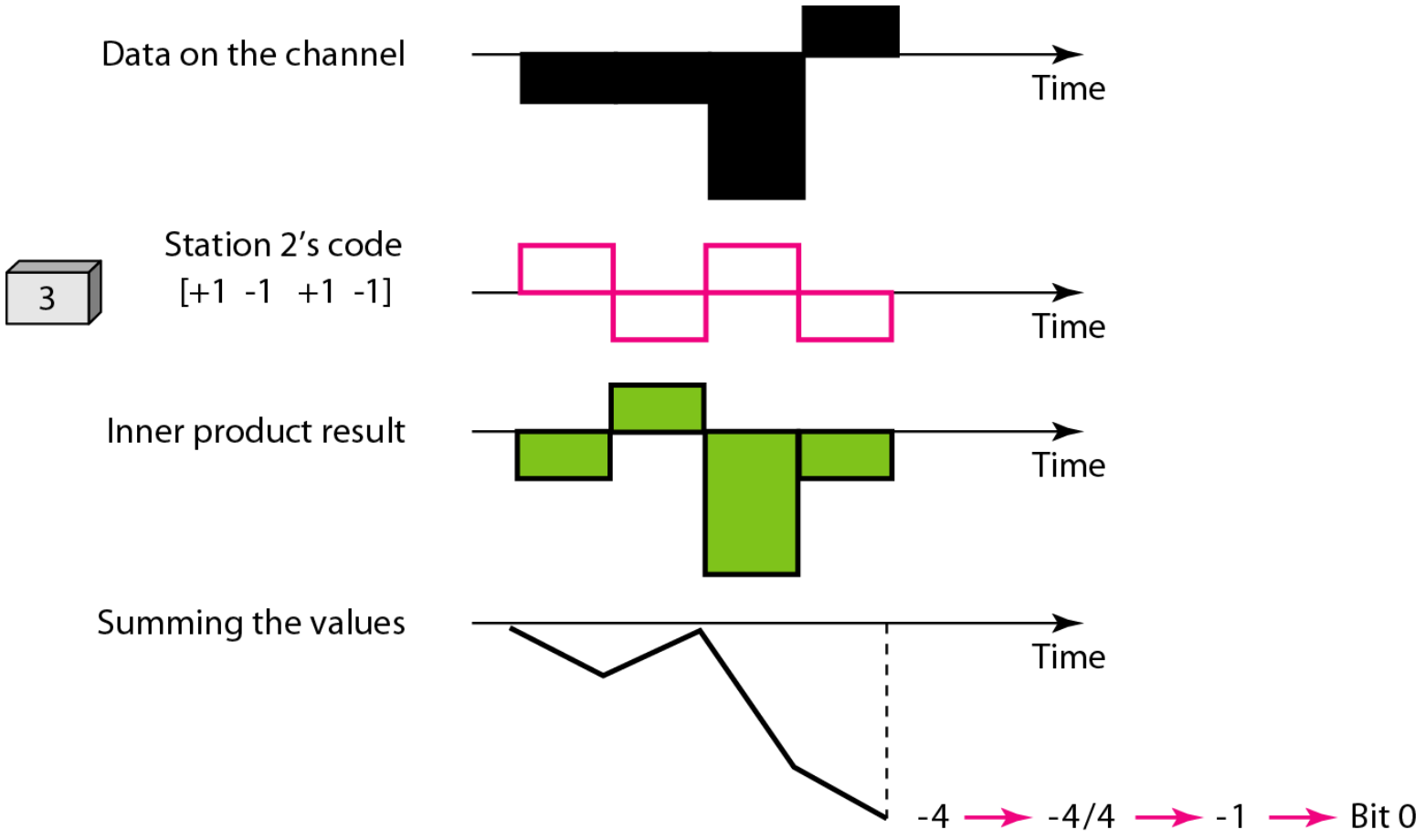
Sharing channel in CDMA



Digital signal created by four stations in CDMA



Decoding of the composite signal for one in CDMA



Sequence Generation
Walsh Table(nxn)

General rule and examples of creating Walsh tables

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

a. Two basic rules

$$W_1 = [+1]$$

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

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

b. Generation of W_1, W_2, and W_4

Note

The number of sequences in a Walsh table needs to be $N = 2^m$.

Find the chips for a network with
a. Two stations b. Four stations

Solution

We can use the rows of W_2 and W_4

a. For a two-station network, we have

[+1 +1] and [+1 −1].

b. For a four-station network we have

[+1 +1 +1 +1], [+1 −1 +1 −1], [+1 +1 −1 −1], and [+1 −1 −1 +1].

$$W_1 = \begin{bmatrix} +1 \end{bmatrix}$$

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

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

Example 1

What is the number of sequences if we have 90 stations in our network?

Solution

The number of sequences needs to be 2^m. We need to choose m = 7 and N = 2^7 or 128. We can then use 90 of the sequences as the chips.

Example 2

Prove that a receiving station can get the data sent by a specific sender if it multiplies the entire data on the channel by the sender's chip code and then divides it by the number of stations.

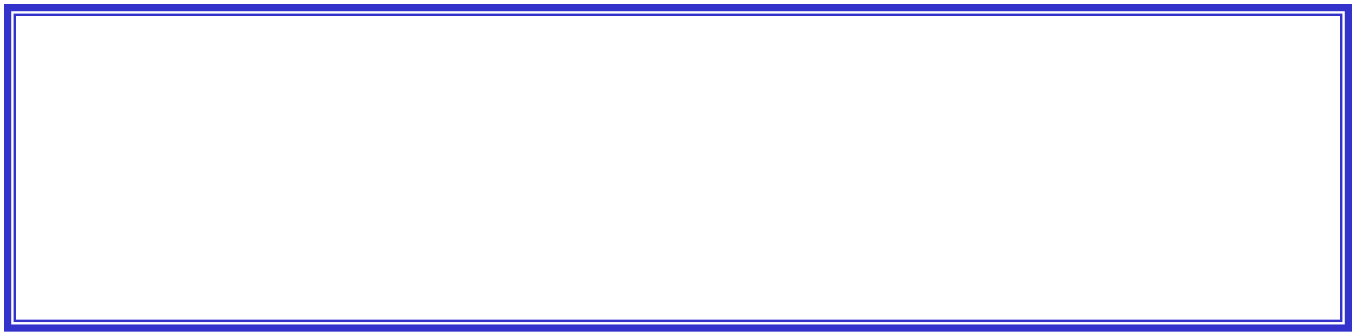
Solution

Let us prove this for the first station, using our previous four-station example. We can say that the data on the channel

$$D = (d_1 \cdot c_1 + d_2 \cdot c_2 + d_3 \cdot c_3 + d_4 \cdot c_4).$$

The receiver which wants to get the data sent by station 1 multiplies these data by c_1.

Example 2 (continued)



When we divide the result by N, we get d_1 .

ADDRESS MAPPING

The delivery of a packet to a host or a router requires two levels of addressing: logical and physical. We need to be able to map a logical address to its corresponding physical address and vice versa.

Topics discussed in this section:

Mapping Logical to Physical Address
Mapping Physical to Logical Address

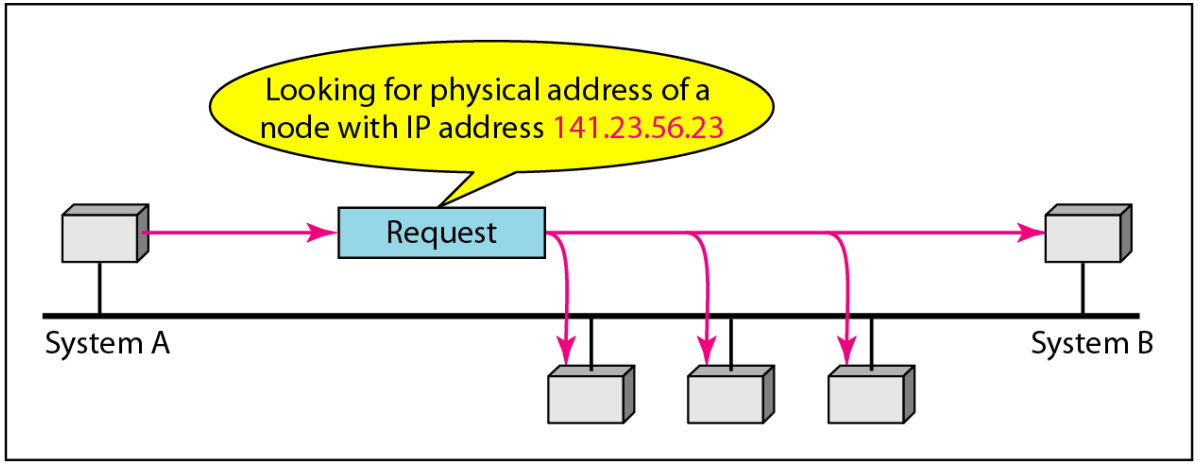
Address Resolution Protocol

- The purpose of Address Resolution Protocol (ARP) is to find out the MAC address of a device in Local Area Network (LAN), for the corresponding IP address, which network application is trying to communicate.

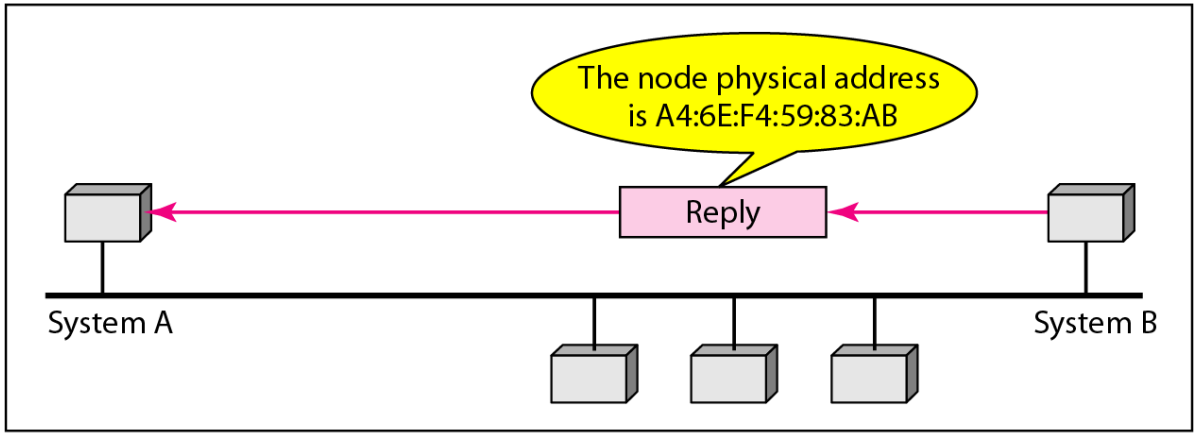
- Address Resolution Protocol (ARP) is one of the major protocol in the TCP/IP suit and the purpose of Address Resolution Protocol (ARP) is to resolve an **IP address (32 bit Logical Address)** to the **physical address (48 bit MAC Address)**.

- Network Applications at the Application Layer use IP address to communicate with another device. But at the Datalink layer, the addressing is **MAC address (48 bit Physical Address)**, and this address is burned into the network card permanently.

Figure :1 ***ARP operation***

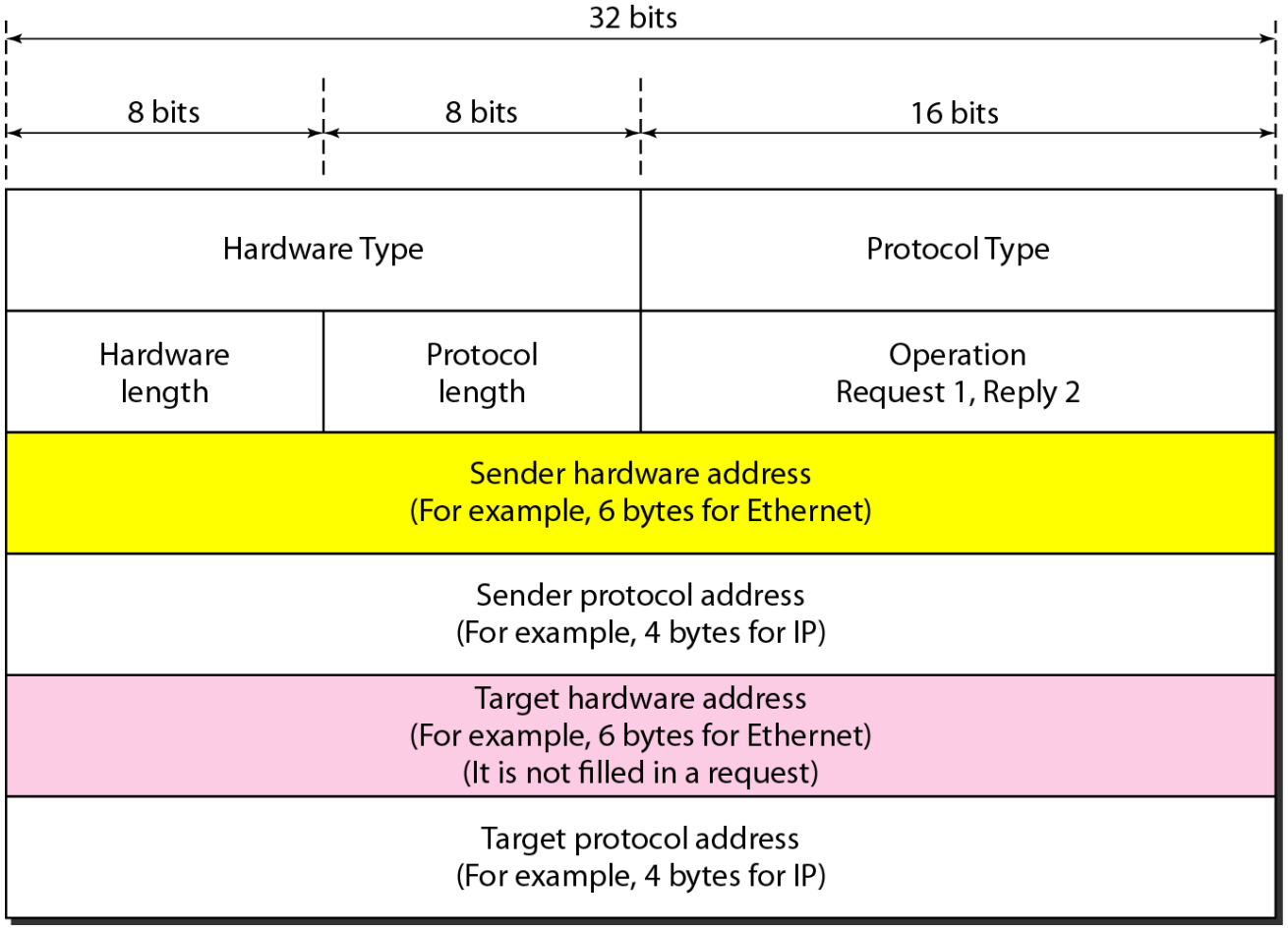


a. ARP request is broadcast



b. ARP reply is unicast

ARP packet


32 bits
8 bits
8 bits
16 bits
Hardware Type
Protocol Type
Hardware length
Protocol length
Operation Request 1, Reply 2
Sender hardware address (For example, 6 bytes for Ethernet)
Sender protocol address (For example, 4 bytes for IP)
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)
Target protocol address (For example, 4 bytes for IP)

Working of Address Resolution Protocol (ARP)

- Step 1: When a source device want to communicate with another device, source device checks its Address Resolution Protocol (ARP) cache to find it already has a resolved MAC address of the destination device. If it is there, it will use that address for communication.
- To view your Local Address Resolution Protocol (ARP) cache, Open Command Prompt and type command "arp -a" (Without double quotes using Windows Operating Systems).

- Step 2: If ARP resolution is not there in local cache, the source machine will generate an Address Resolution Protocol (ARP) request message, it puts its own data link layer address as the Sender Hardware Address and its own IP address as the Sender Protocol Address. It fills the destination IP address as the Target Protocol Address. The Target Hardware Address will be left blank, since the machine is trying to find that.

- Step 3: The source broadcast the Address Resolution Protocol (ARP) request message to the local network.

- Step 4: The message is received by each device on the LAN since it is a broadcast. Each device compare the Target Protocol Address (IP Address of the machine to which the source is trying to communicate) with its own Protocol Address (IP Address). Those who do not match will drop the packet without any action.

- Step 5: When the targeted device checks the Target Protocol Address, it will find a match and will generate an Address Resolution Protocol (ARP) reply message. It takes the Sender **Hardware Address**and the Sender **Protocol Address** fields from the Address Resolution Protocol (ARP) request message and uses these values for the Targeted Hardware Address and Targeted Protocol Address of the reply message.

- Step 6: The destination device will update its Address Resolution Protocol (ARP) cache, since it need to contact the sender machine soon.

- Step 7: Destination device send the Address Resolution Protocol (ARP) reply message and it will not be a broadcast, but a unicast.

- Step 8: The source machine will process the Address Resolution Protocol (ARP) reply from destination, it store the Sender Hardware Address as the layer 2 address of the destination.

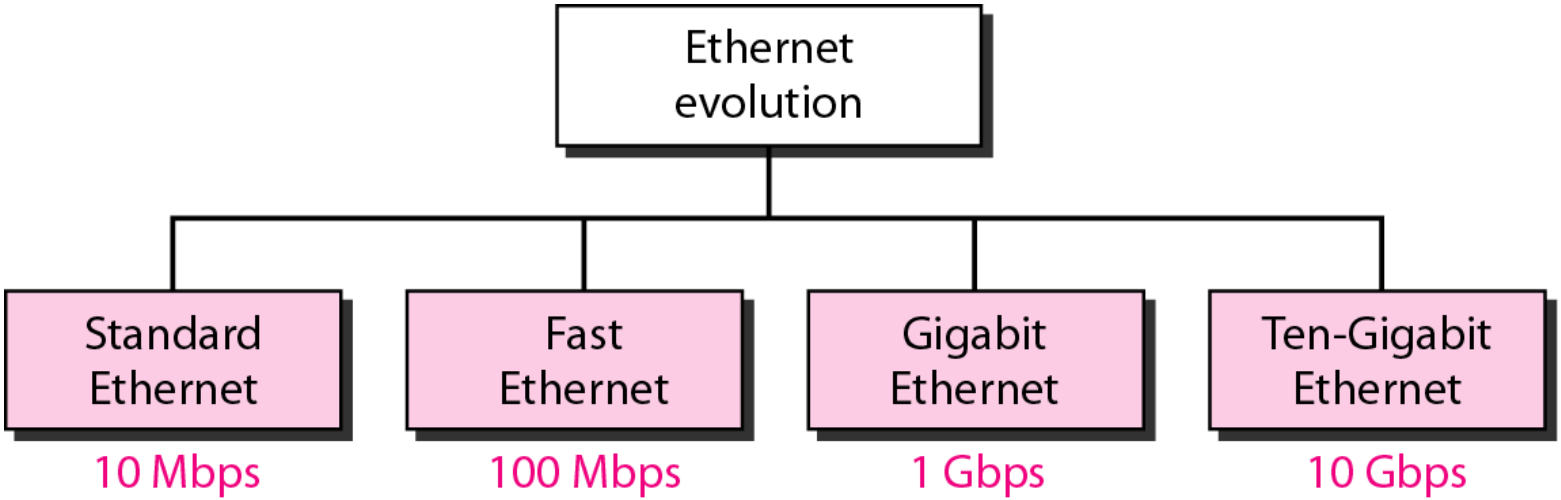
- Step 9: The source machine will update its Address Resolution Protocol (ARP) cache with the Sender Hardware Address and Sender Protocol Address it received from the Address Resolution Protocol (ARP) reply message.

STANDARD ETHERNET IEEE 802.3

The original Ethernet was created in 1976 at Xerox's Palo Alto Research Center (PARC). Since then, it has gone through four generations. We briefly discuss the Standard (or traditional) Ethernet in this section.

MAC Sublayer
Physical Layer

Ethernet evolution through four generations



STANDARD ETHERNET- *802.3*

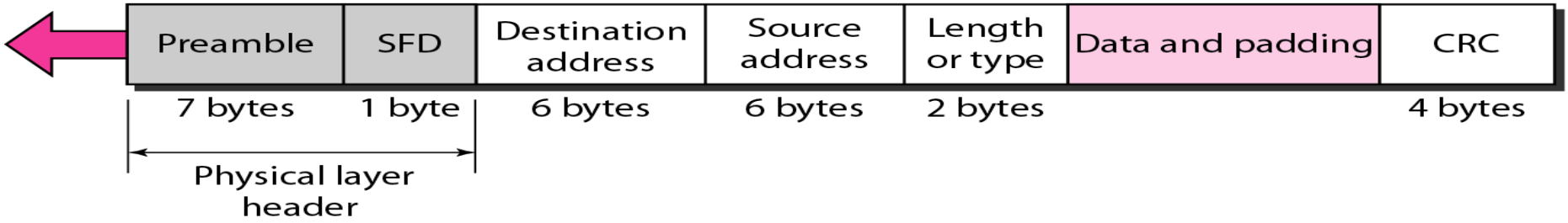
MAC Sub layer-governs the operation of access method.

Physical Layer implementations

802.3 MAC frame Format

Preamble: 56 bits of alternating 1s and 0s.

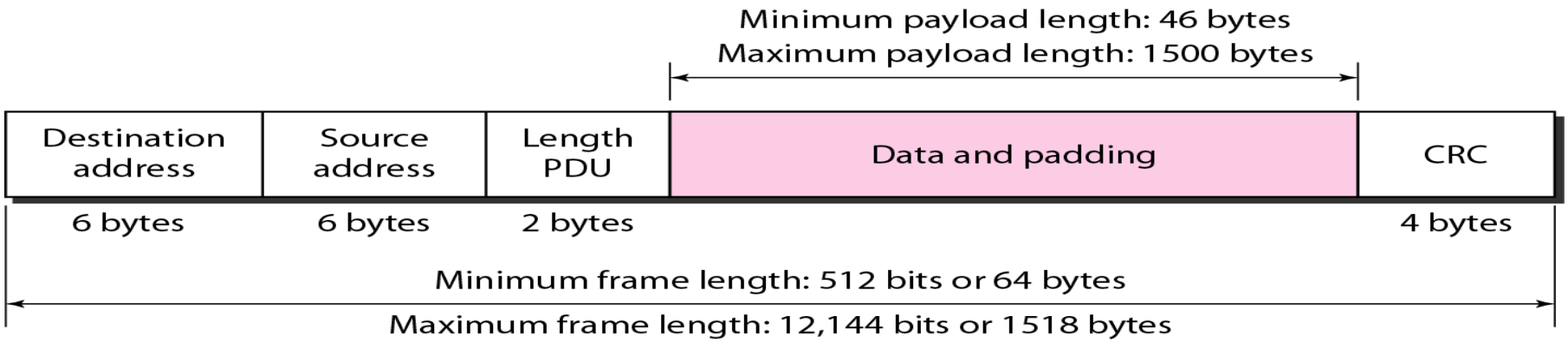
SFD: Start frame delimiter, flag (10101011)



Preamble: 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 (l 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.

Minimum and maximum lengths



The minimum length restriction is required for the correct operation of *CSMA/CD*

The maximum length restriction has two reasons.

1. **Memory was very expensive when Ethernet was designed: helped to reduce the size of the buffer.**
2. **Prevents one station from monopolizing the shared medium, blocking other stations that have data to send.**

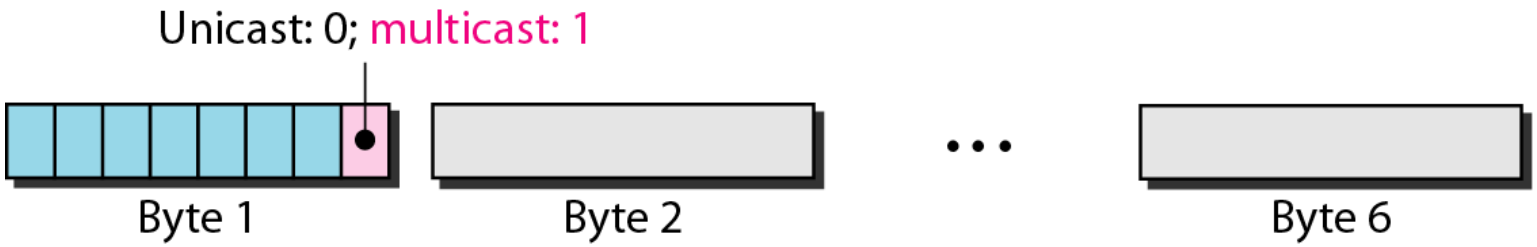
Note

Frame length:

Minimum: 64 bytes (512 bits)

Maximum: 1518 bytes (12,144 bits)

Unicast and multicast addresses



Note

The least significant bit of the first byte defines the type of address. If the bit is 0, the address is unicast; otherwise, it is multicast.

Note

The broadcast destination address is a special case of the multicast address in which all bits are 1s.

Example 1

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 F's, the address is broadcast. Therefore, we have the following:

a. This is a unicast address because A in binary is 1010.

b. This is a multicast address because 7 in binary is 0111.

c. This is a broadcast address because all digits are F's.

Example 2

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

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:

Connecting LANs, Backbone Networks, and Virtual LANs

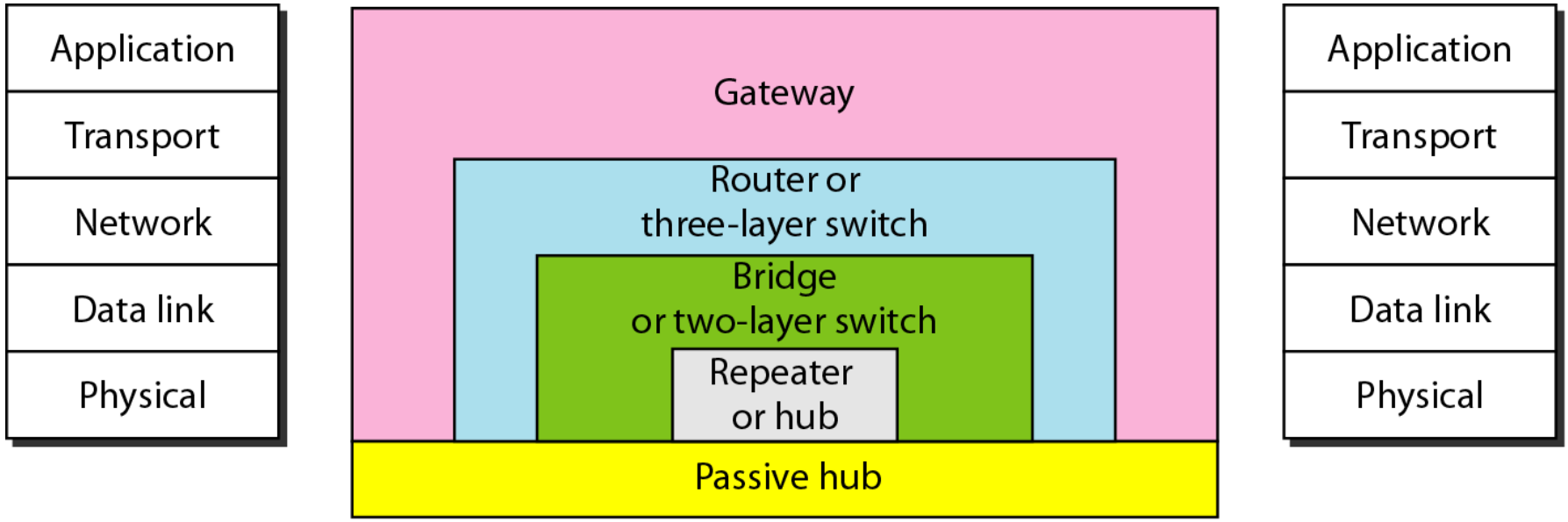
CONNECTING DEVICES

In this section, we divide connecting devices into five different categories based on the layer in which they operate in a network.

Topics discussed in this section:

Passive Hubs
Active hubs
Two-Layer Switches
Routers
Three-Layer Switches
Gateways

Five categories of connecting devices



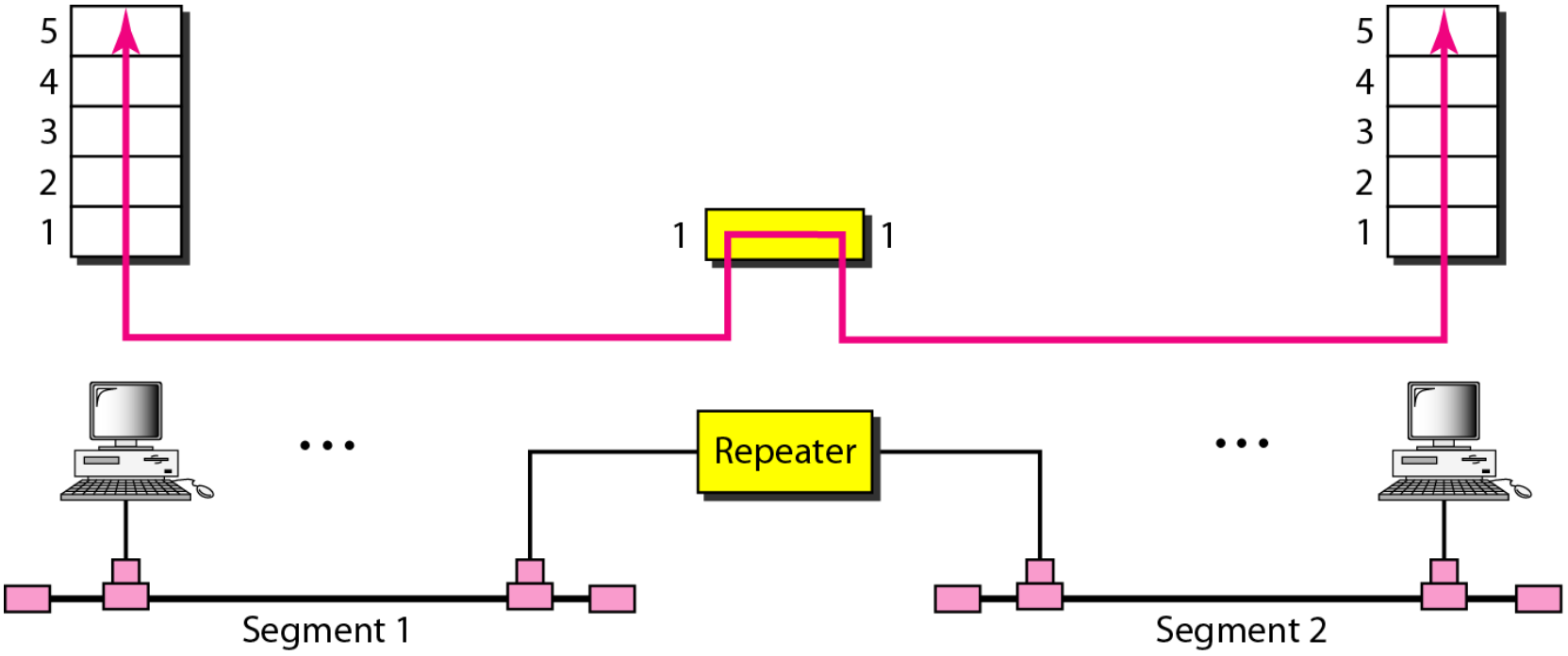
Passive Hub

- It is just a connector.
- It connects wires coming from different branches.
- Its location in Internet model is below physical layer.

Repeater

- Operates only in the physical layer.
- It regenerates the original signal.(which is attenuated)
- Sends the refreshed signal.
- Can extend the physical length of the LAN
- **It does not connect two LANs, it connects two segments of the same LAN.**

A repeater connecting two segments of a LAN



Note

A repeater connects segments of a LAN.

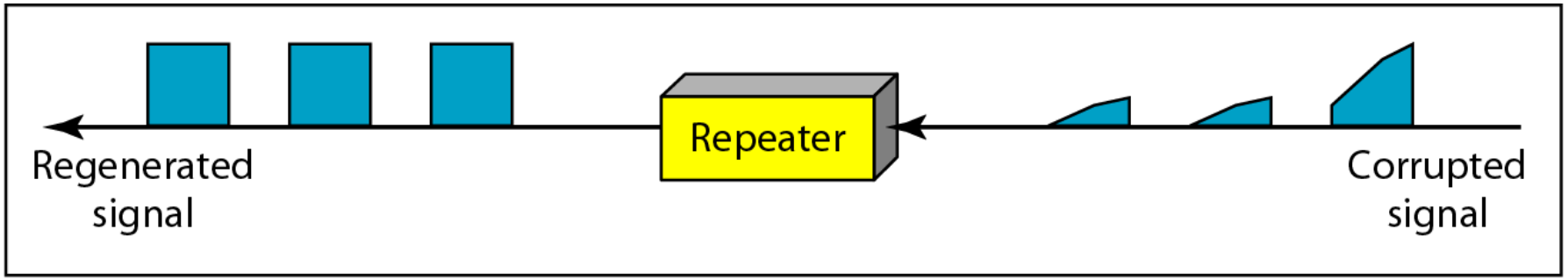
Note

A repeater forwards every frame;
it has no filtering capability.

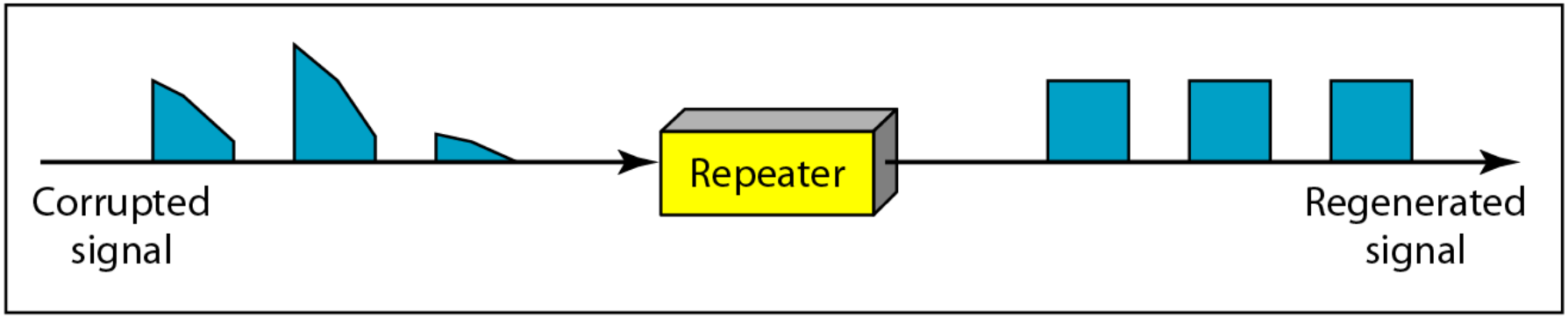
Note

A repeater is a regenerator, not an amplifier.

Function of a repeater



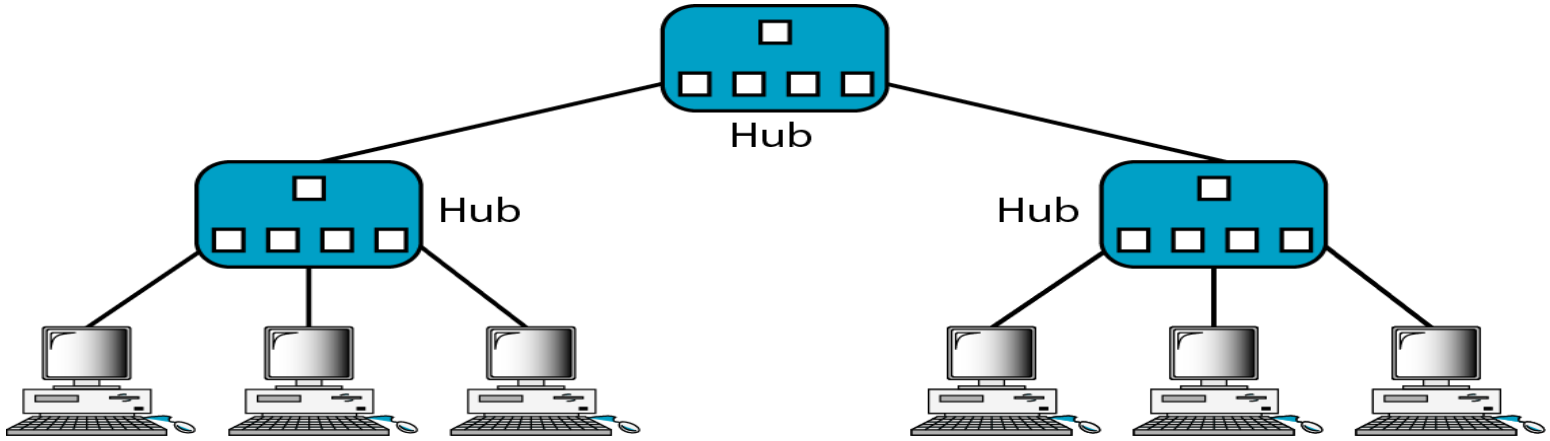
a. Right-to-left transmission.



b. Left-to-right transmission.

Active Hubs

- It is multiport repeater.
- Used to create star topology.
- Can also be used to create multilevel hierarchy.



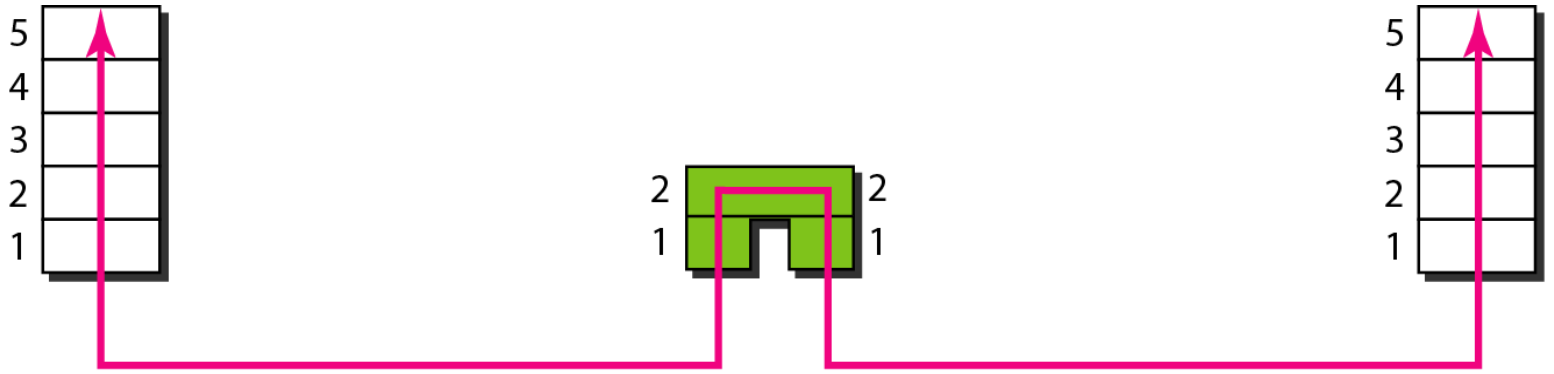
Bridge

- Operates in both physical and data link layer.
- As a part of physical layer, it regenerates the signal and as a part of data link layer it checks the MAC address.(filtering)
- **Difference between repeater and bridge?**

Note

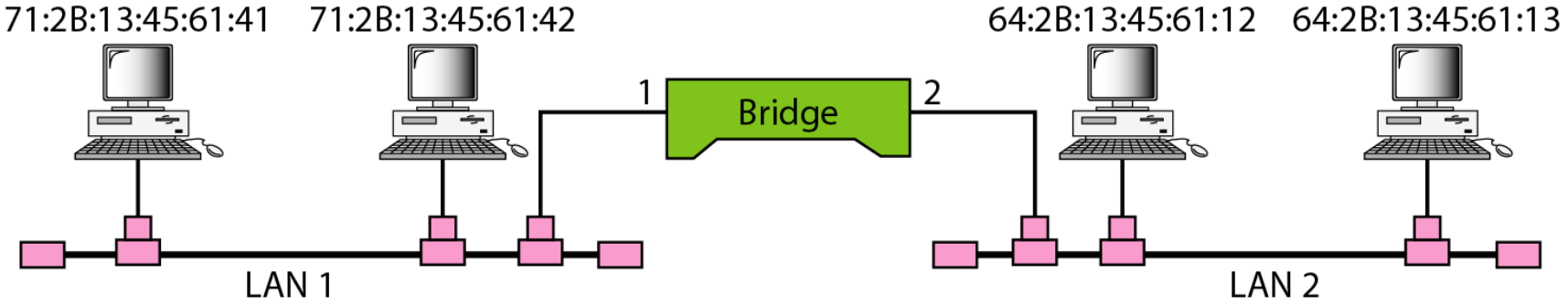
A bridge has a table used in filtering decisions.

A bridge connecting two LANs



Address	Port
71:2B:13:45:61:41	1
71:2B:13:45:61:42	1
64:2B:13:45:61:12	2
64:2B:13:45:61:13	2

Bridge Table



Note

A bridge does not change the physical (MAC) addresses in a frame.

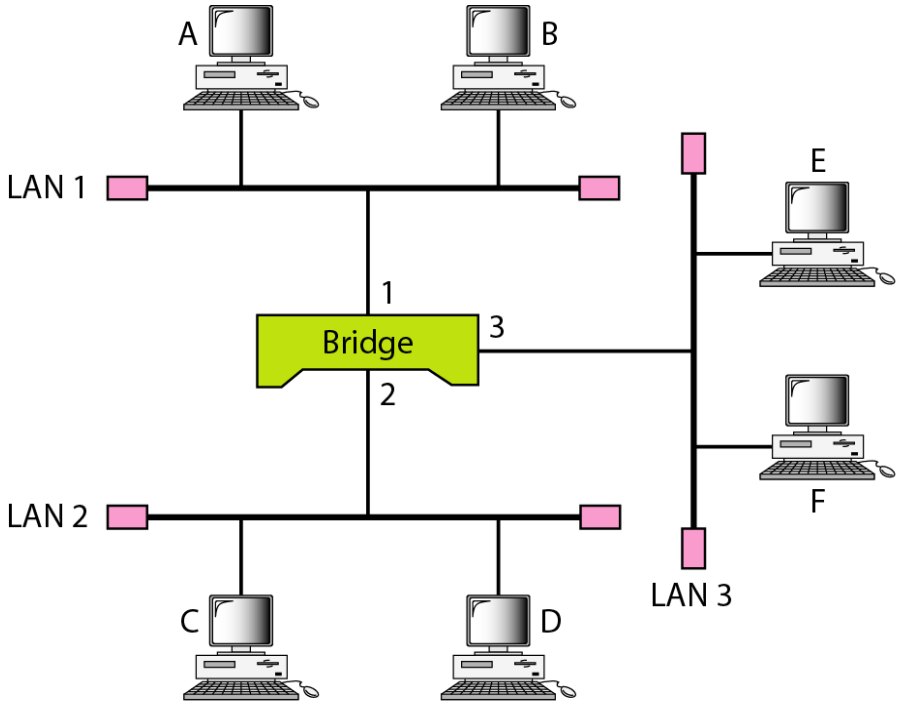
Transparent bridge

- Is a bridge in which stations are completely unaware of existence of the bridge.
- Reconfiguration of the system is not necessary (if bridge is added or deleted).
- Must meet the three criteria,
  1. Frame forwarding
  2. Building forwarding table automatically by learning process
  3. Create loopless topology.

Learning

- Earlier tables were static, managed by system administrator.

A learning bridge and the process of learning



Address	Port

a. Original

Address	Port
A	1

b. After A sends a frame to D

Address	Port
A	1
E	3

c. After E sends a frame to A

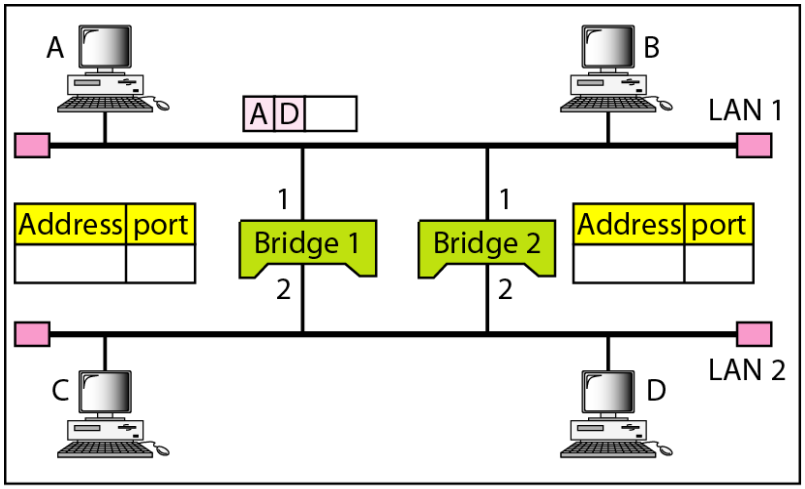
Address	Port
A	1
E	3
B	1

d. After B sends a frame to C

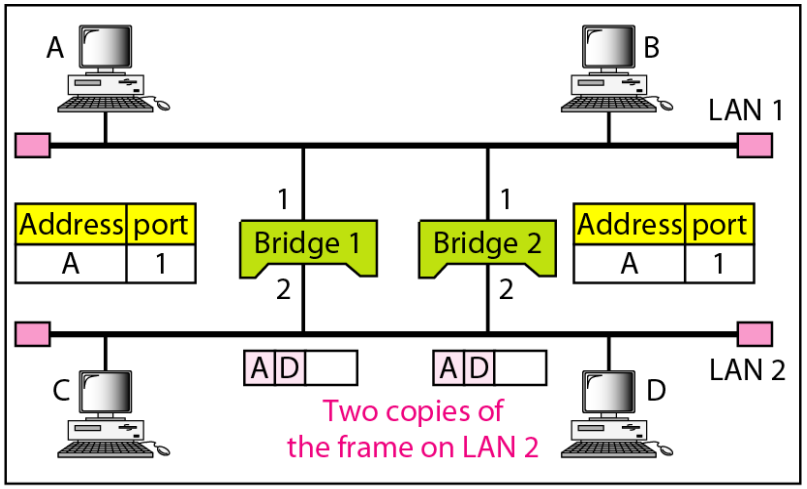
LOOP PROBLEM

(in learning process)

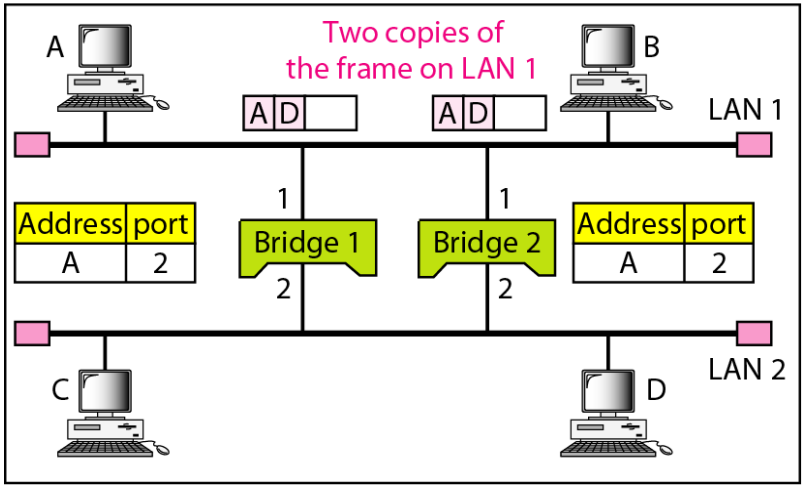
Loop problem in a learning bridge



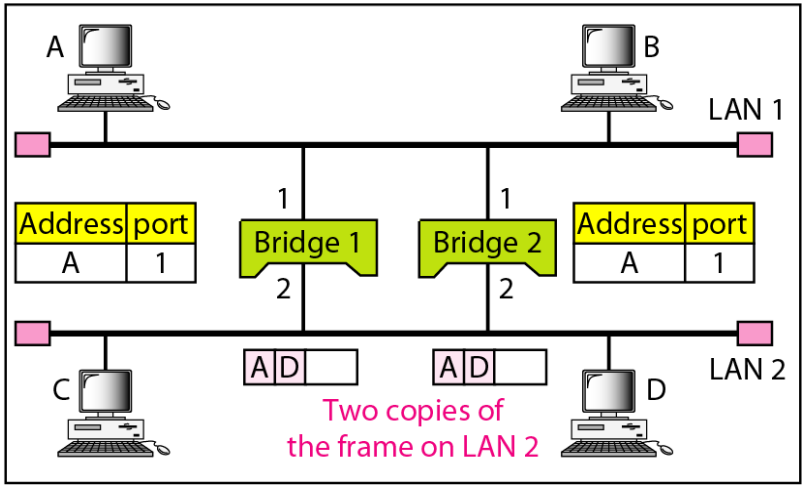
a. Station A sends a frame to station D



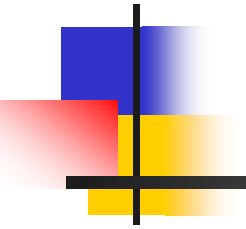
b. Both bridges forward the frame



c. Both bridges forward the frame



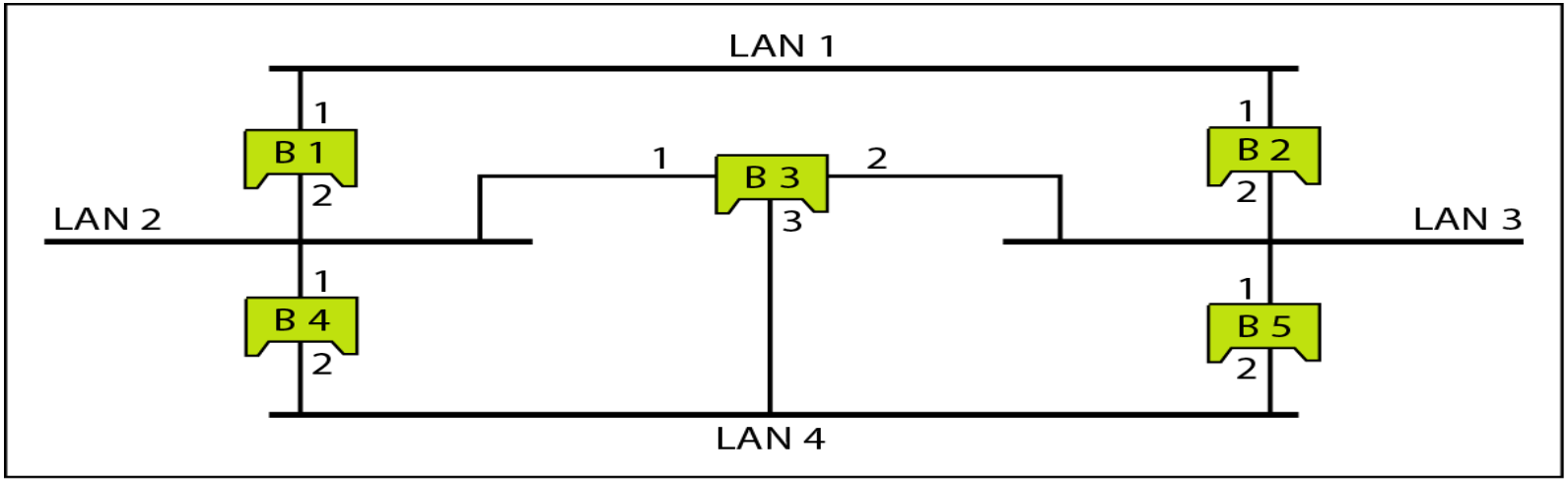
d. Both bridges forward the frame



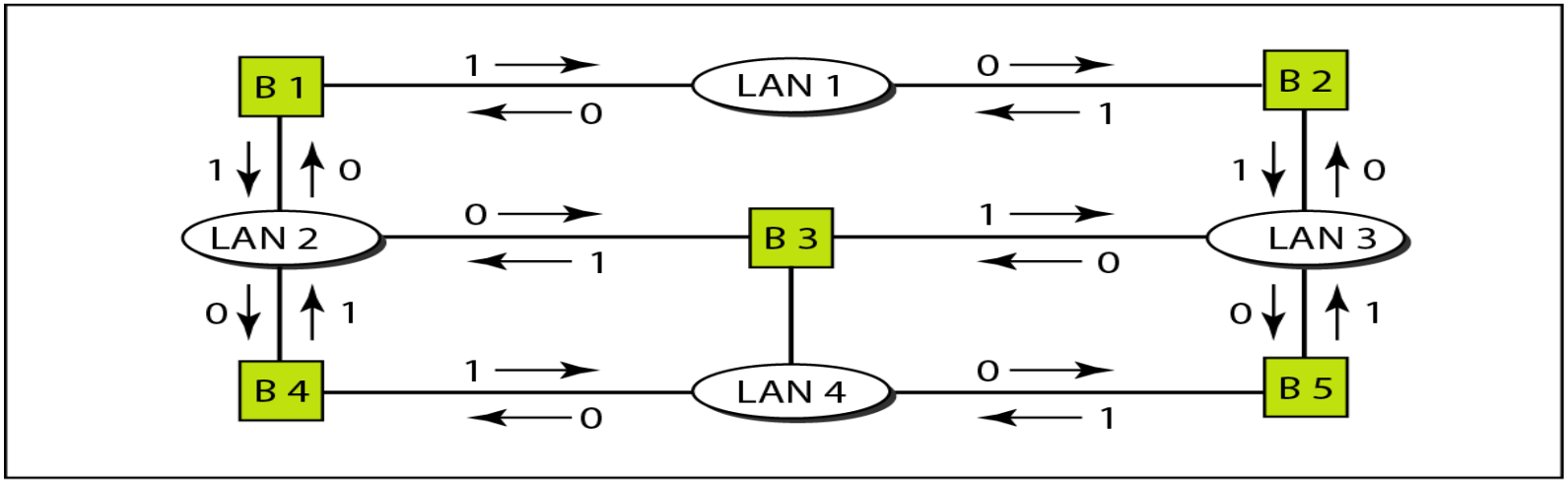
Solution?

Spanning Tree

A system of connected LANs and its graph representation

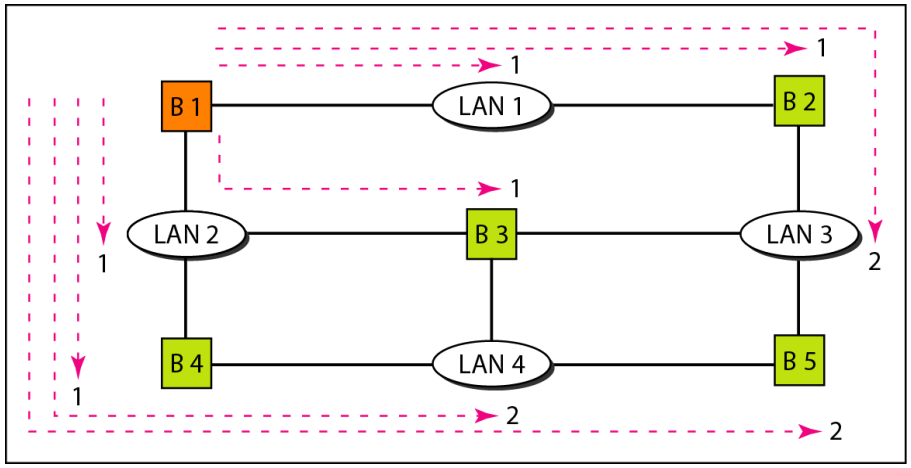


a. Actual system

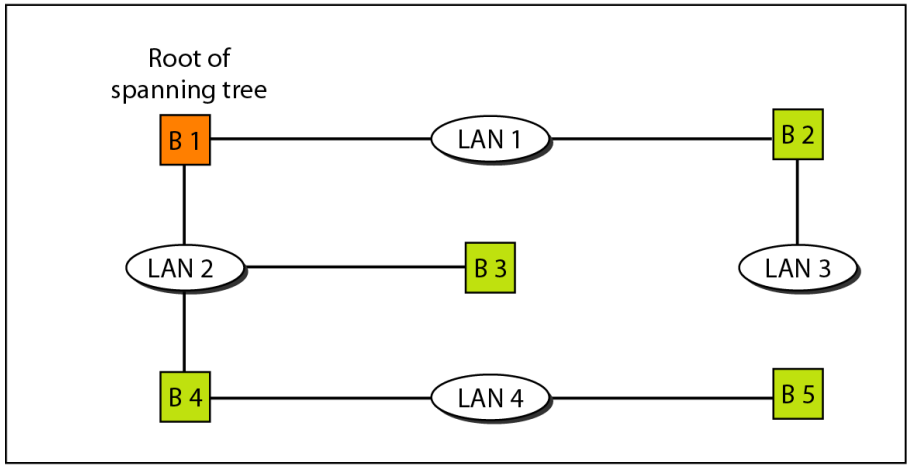


b. Graph representation with cost assigned to each arc

Finding the shortest paths and the spanning tree in a system of bridges



a. Shortest paths



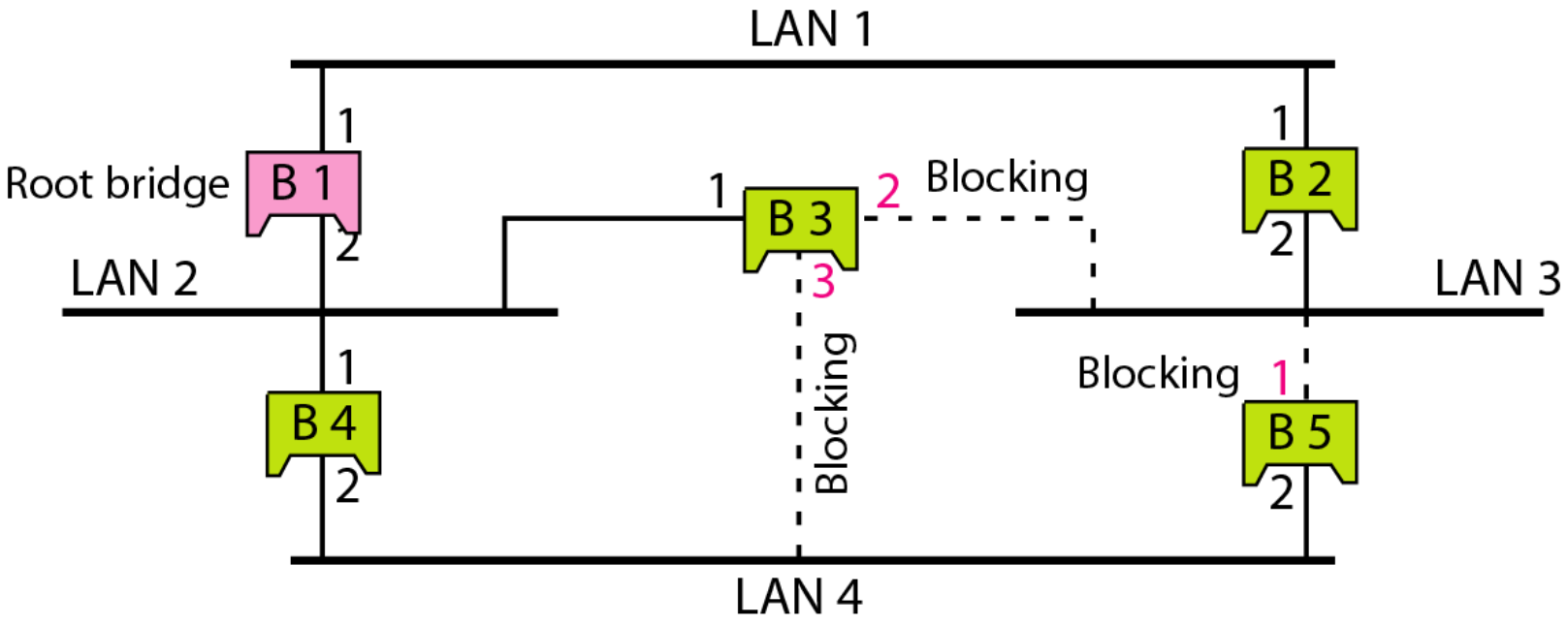
b. Spanning tree

Every bridge has a built-in ID (normally the serial number, which is unique).

Each bridge broadcasts this ID so that all bridges know which one has the smallest ID.

The bridge with the smallest ID is selected as the *root bridge (root of the tree).*

Forwarding and blocking ports after using spanning tree algorithm



Ports 2 and 3 of bridge B3 are blocking ports (no frame is sent out of these ports).
Port 1 of bridge B5 is also a blocking port (no frame is sent out of this port).

Source Routing Bridges

Another way to prevent loops in a system with redundant bridges

In source routing, a sending station defines the bridges that the frame must visit.

The frame contains not only the source and destination addresses, but also the addresses of all bridges to be visited.

The source gets these bridge addresses through the exchange of special frames with the destination prior to sending the data frame.

Source routing bridges were designed by IEEE to be used with Token Ring LANs.

Bridges connecting different LANs

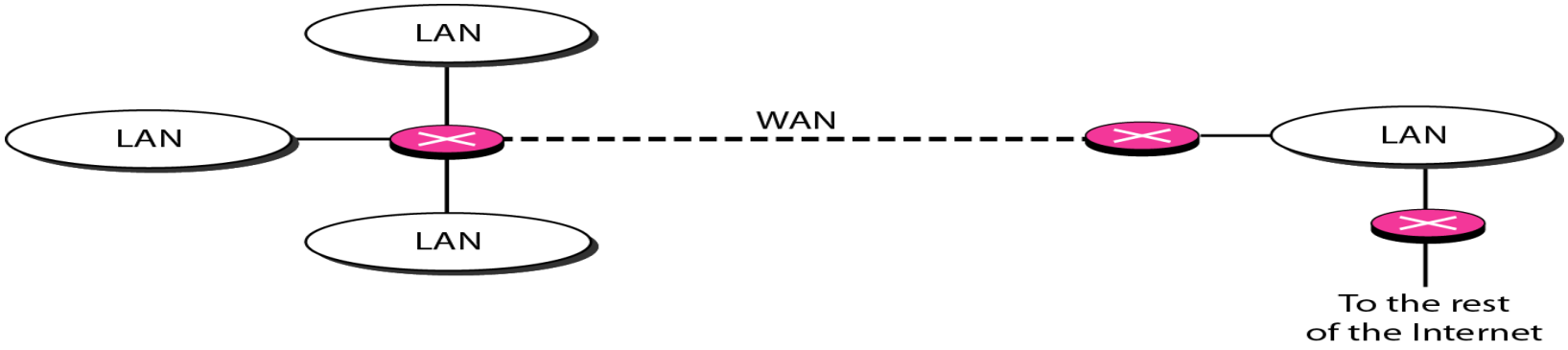
- Issues to be considered,
  - Frame format
  - Maximum data size
  - Data rate
  - Bit order
  - Security
  - Multimedia support

Two-Layer Switches

- A two-layer switch is a bridge, a bridge with many ports and a design that allows better (faster) performance.
- A bridge with many ports may be able to allocate a unique port to each station, with each station on its own independent entity.

Three layer switch-**router**

- Routes the packets based on the logical address.
- Connects two different LANs
  - LAN, WAN



Gateways

- Operates in five or seven layers of the system.
- Basically connects different models
  - OSI and Internet(TCP/IP) models

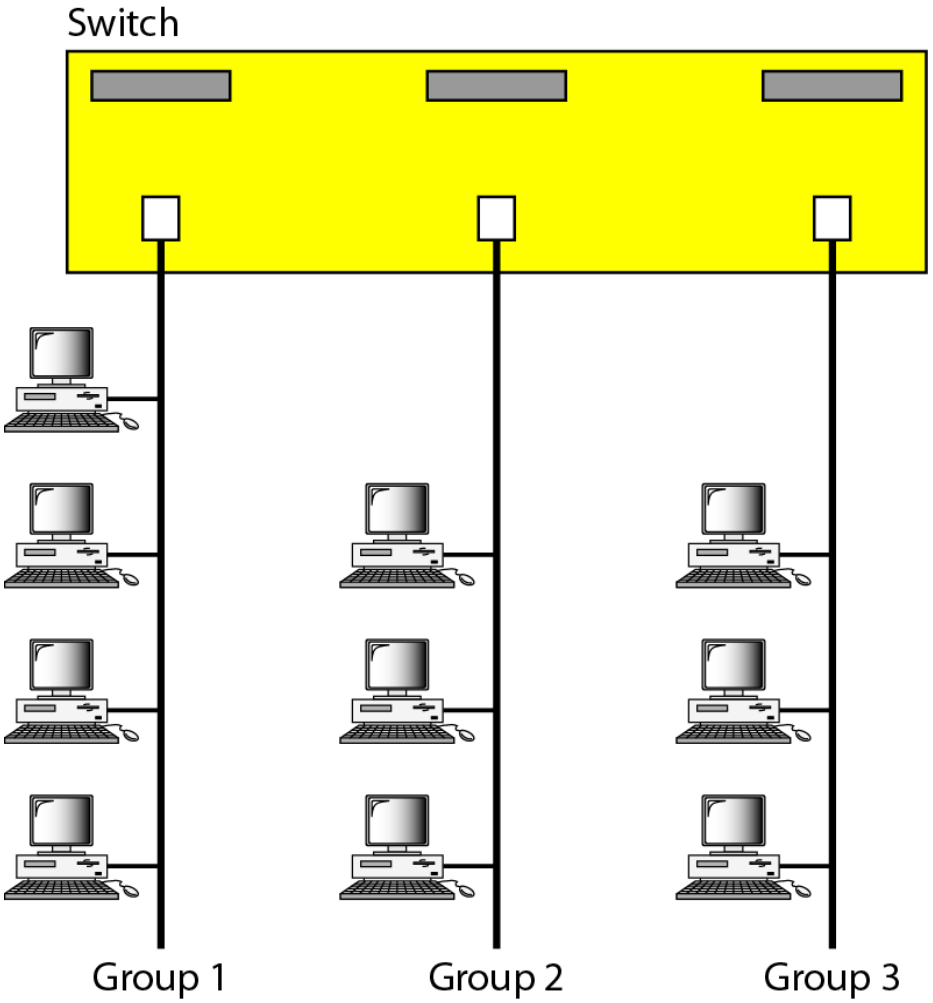
VIRTUAL LANs

We can roughly define a virtual local area network (VLAN) as a local area network configured by software, not by physical wiring.

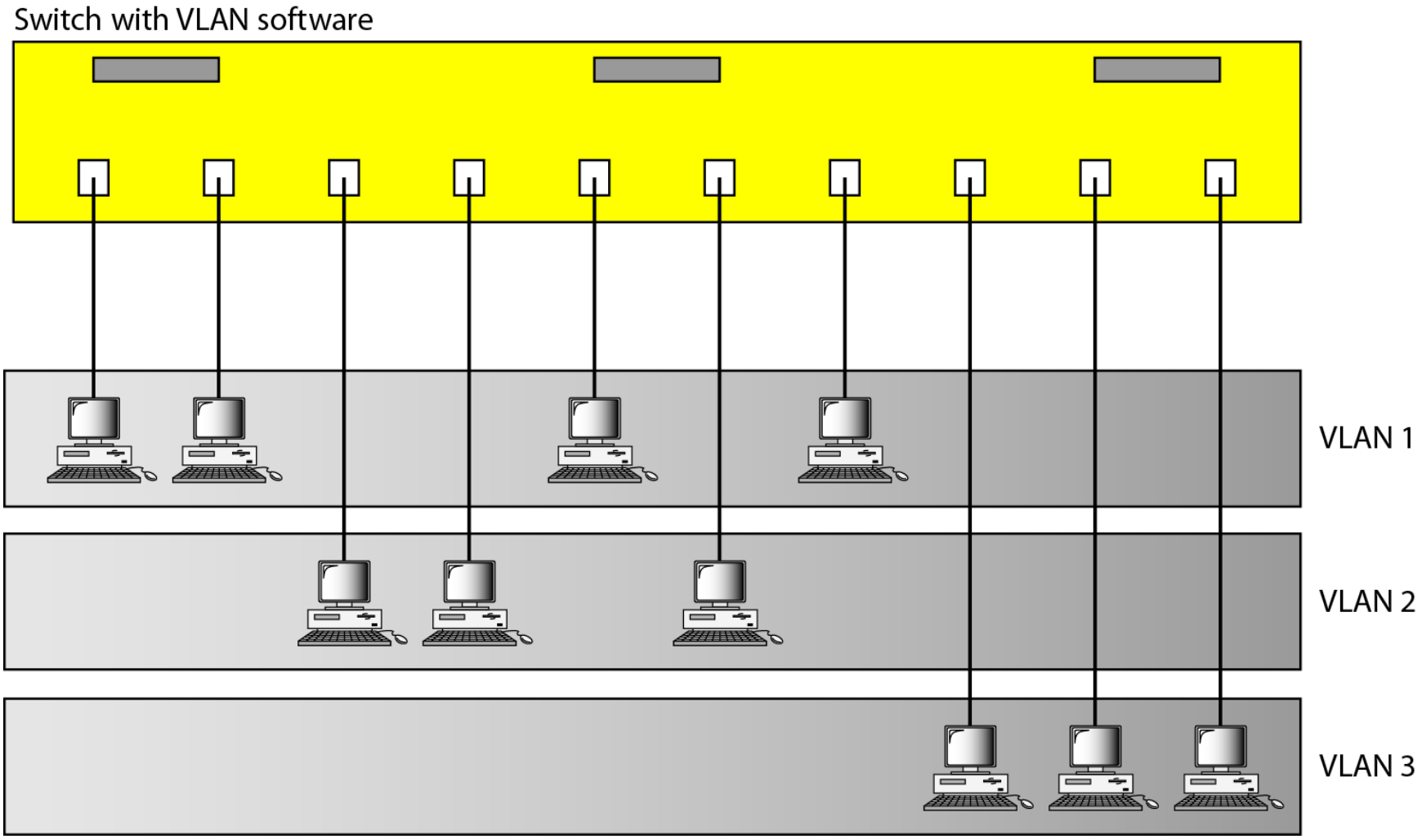
Topics discussed in this section:

Membership
Configuration
Communication between Switches
IEEE Standard
Advantages

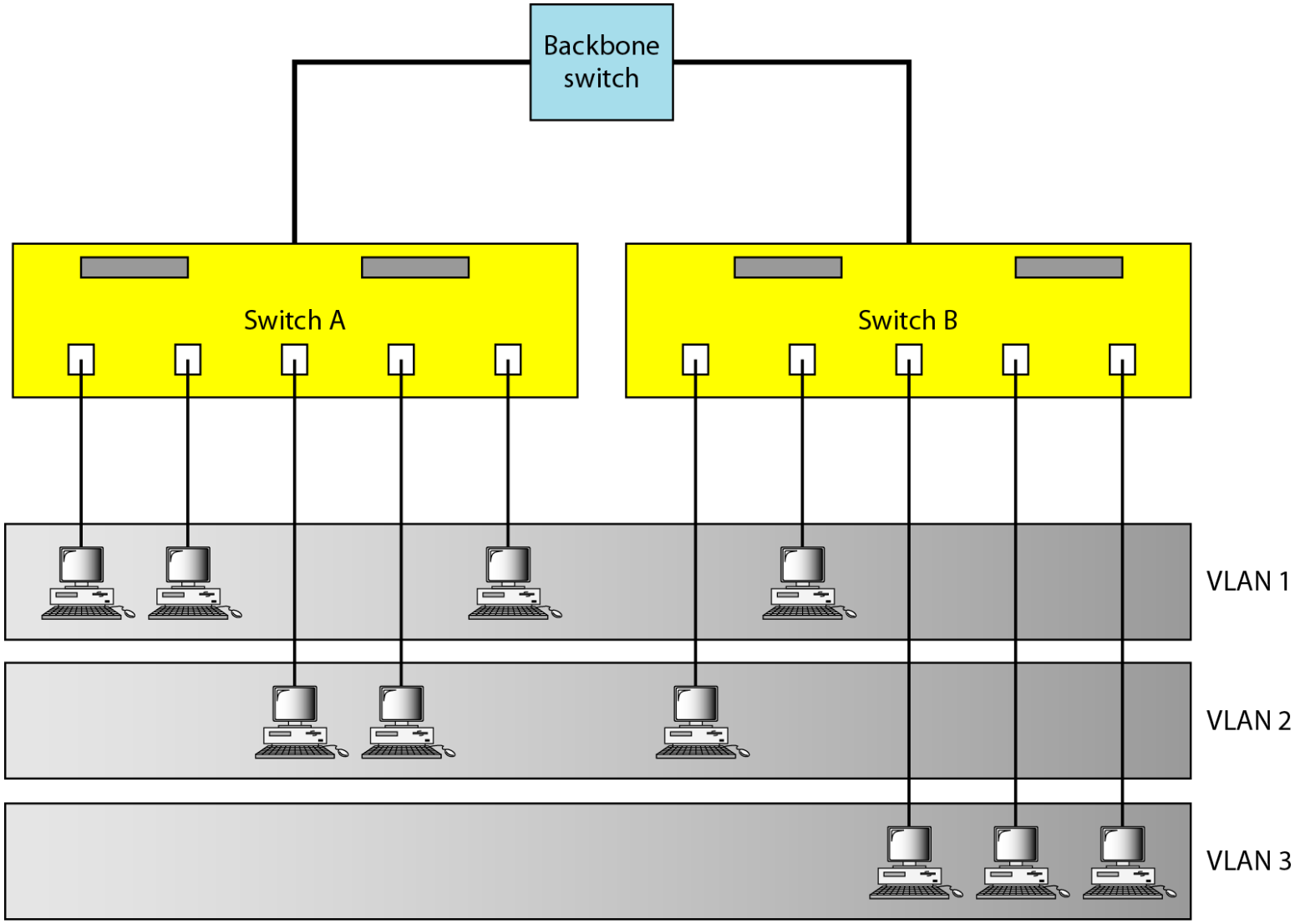
A switch connecting three LANs



A switch using VLAN software



Two switches in a backbone using VLAN software



Note

VLANs create broadcast domains.

Membership

- **Port number**
- **MAC address**
- **IP address**
- **Multicast IP address**
- **Combination**

Configuration

- **Manual configuration**
- **Automatic configuration**
- **Semiautomatic configuration**

Advantages

- **Cost and time reduction**
- **Creating virtual work group**
- **Security**

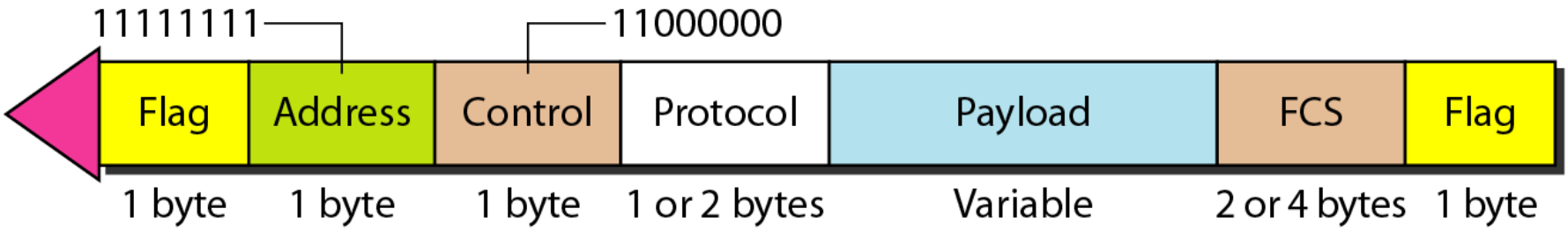
POINT-TO-POINT PROTOCOL

Although HDLC is a general protocol that can be used for both point-to-point and multipoint configurations, one of the most common protocols for point-to-point access is the Point-to-Point Protocol (PPP). PPP is a byte-oriented protocol.

Topics discussed in this section:

Framing
Transition Phases
Multiplexing
Multilink PPP

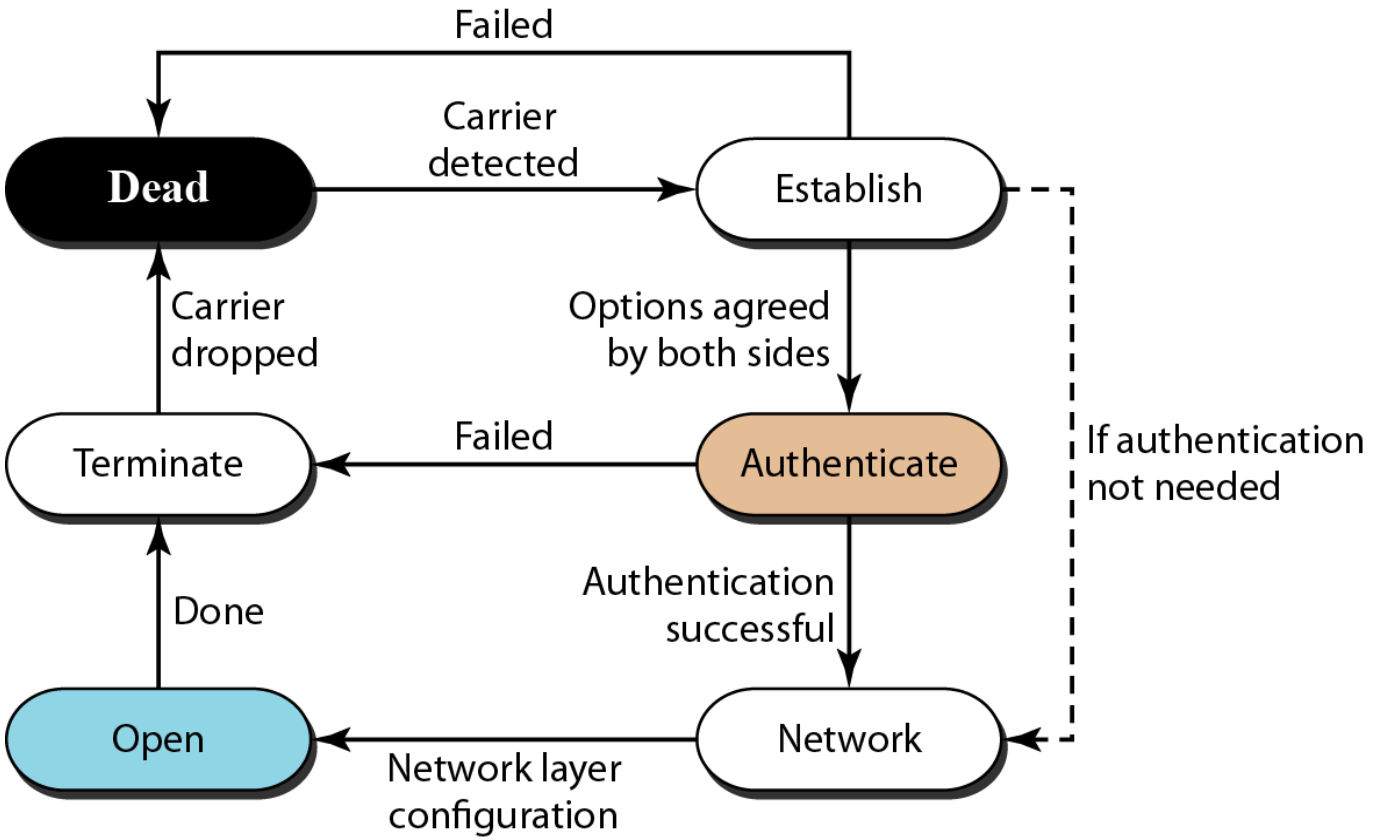
PPP frame format



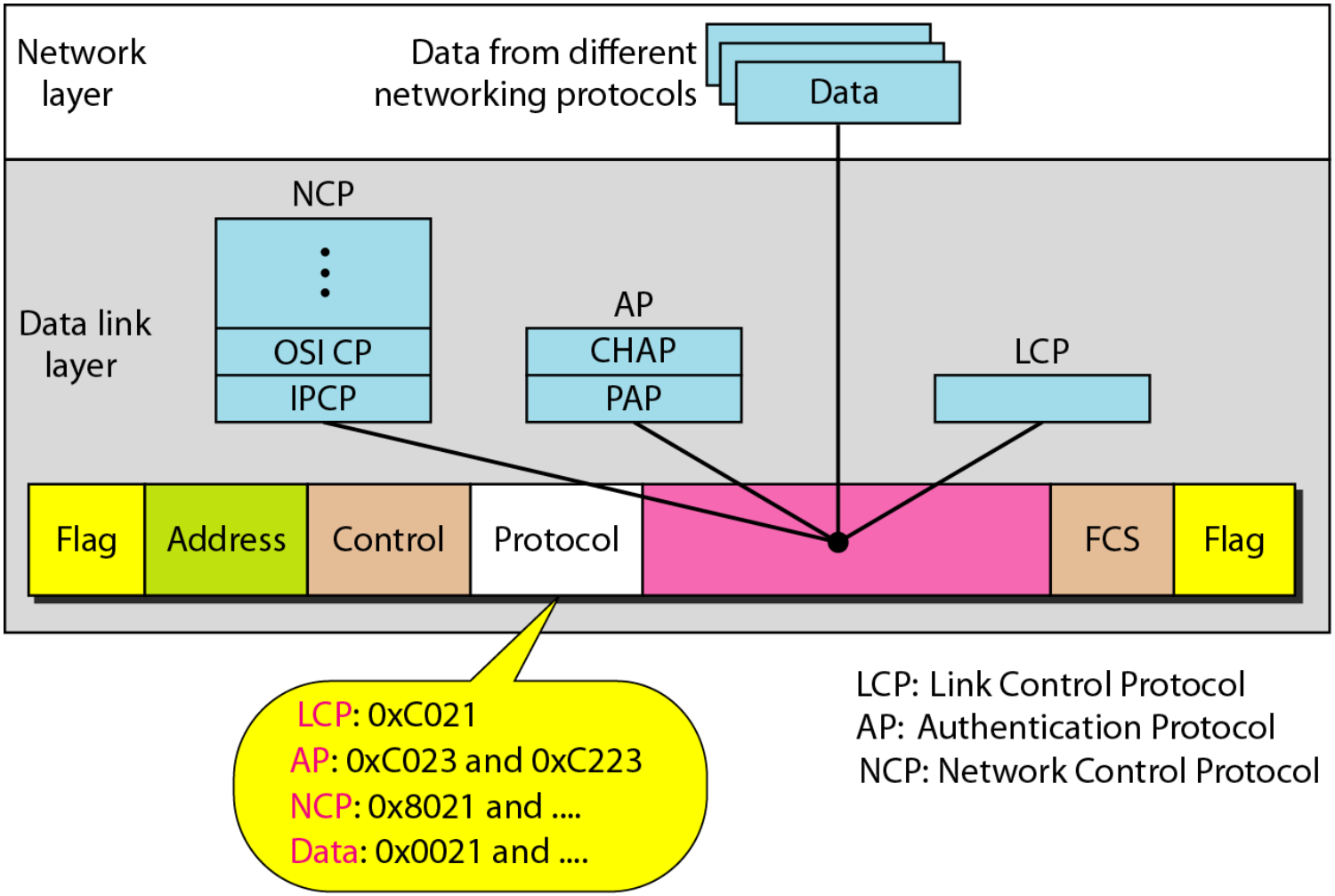
Note

PPP is a byte-oriented protocol using byte stuffing with the escape byte 01111101.

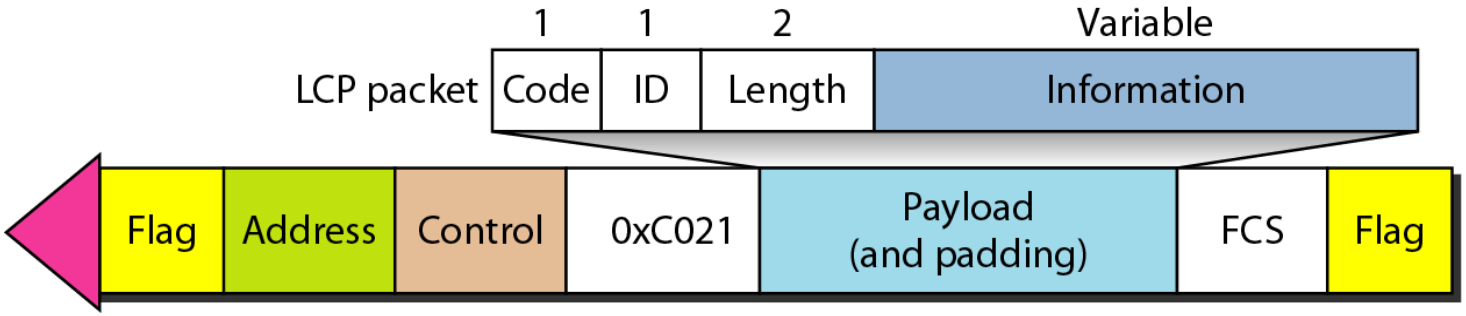
Transition phases



Multiplexing in PPP



LCP packet encapsulated in a frame

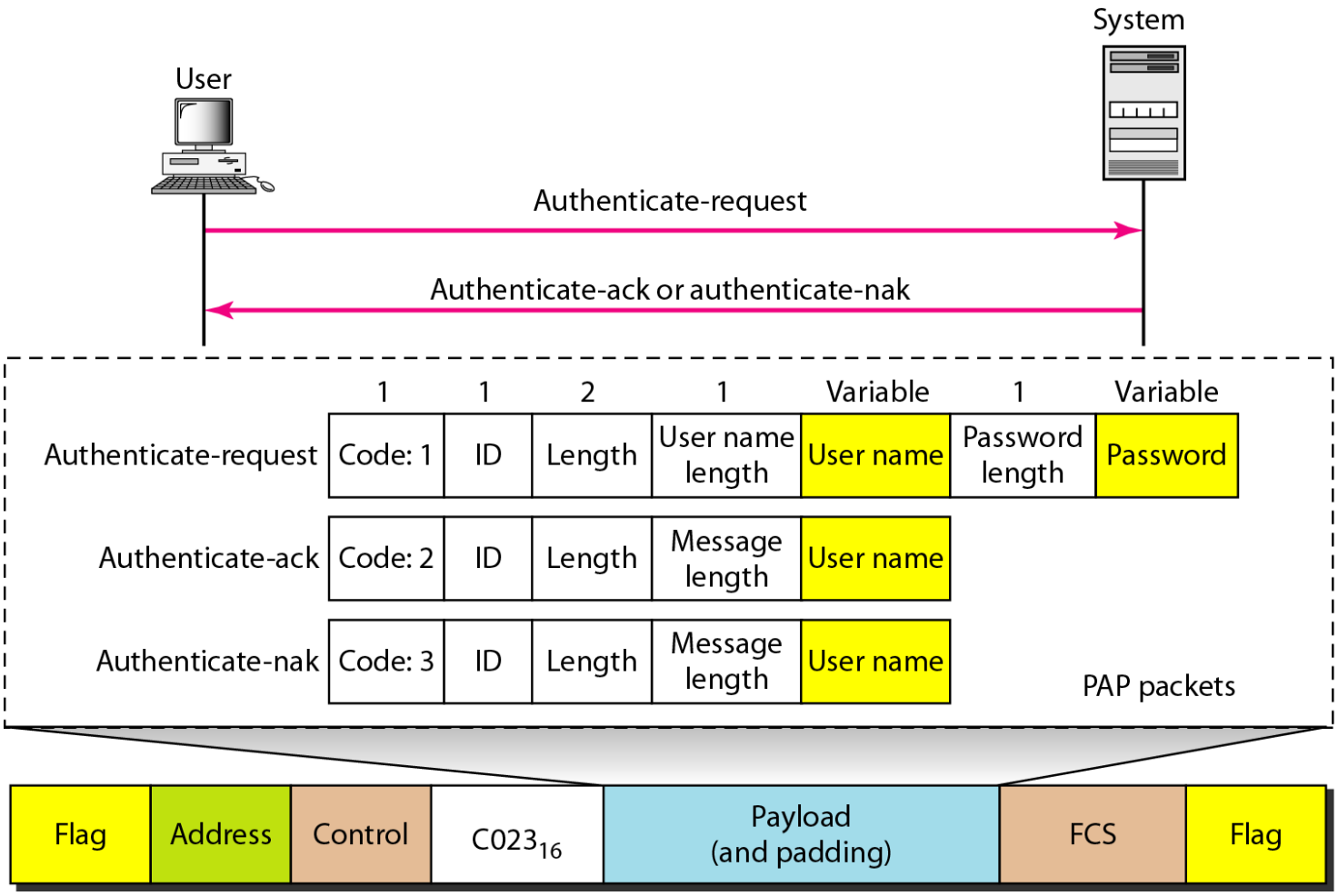


LCP packets

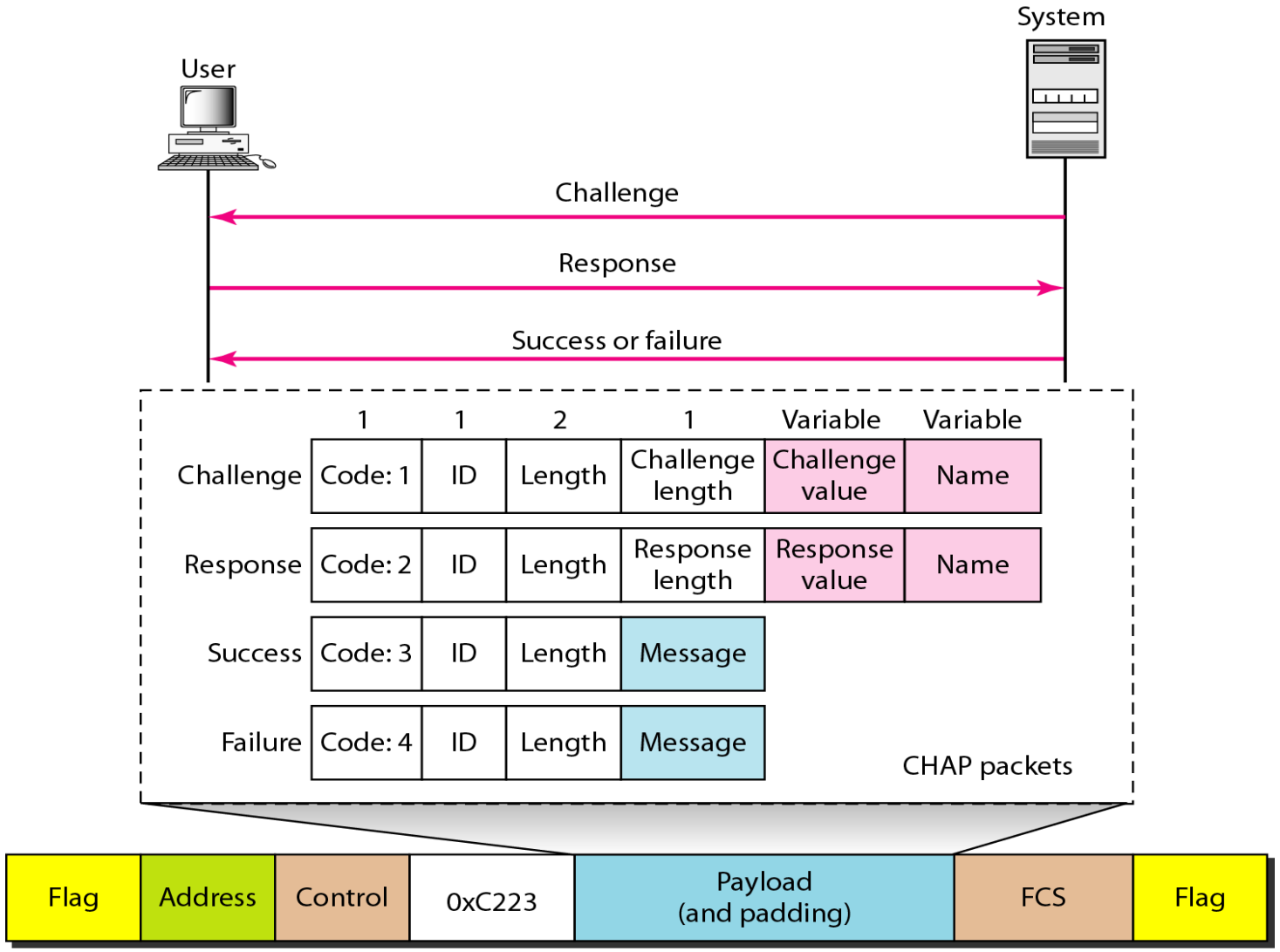
Code	*Packet Type*	*Description*
0x01	Configure-request	Contains the list of proposed options and their values
0x02	Configure-ack	Accepts all options proposed
0x03	Configure-nak	Announces that some options are not acceptable
0x04	Configure-reject	Announces that some options are not recognized
0x05	Terminate-request	Request to shut down the line
0x06	Terminate-ack	Accept the shutdown request
0x07	Code-reject	Announces an unknown code
0x08	Protocol-reject	Announces an unknown protocol
0x09	Echo-request	A type of hello message to check if the other end is alive
0x0A	Echo-reply	The response to the echo-request message
0x0B	Discard-request	A request to discard the packet

Table 1 ***Common options***

Figure *PAP packets encapsulated in a PPP frame*



CHAP packets encapsulated in a PPP frame



IPCP packet encapsulated in PPP frame

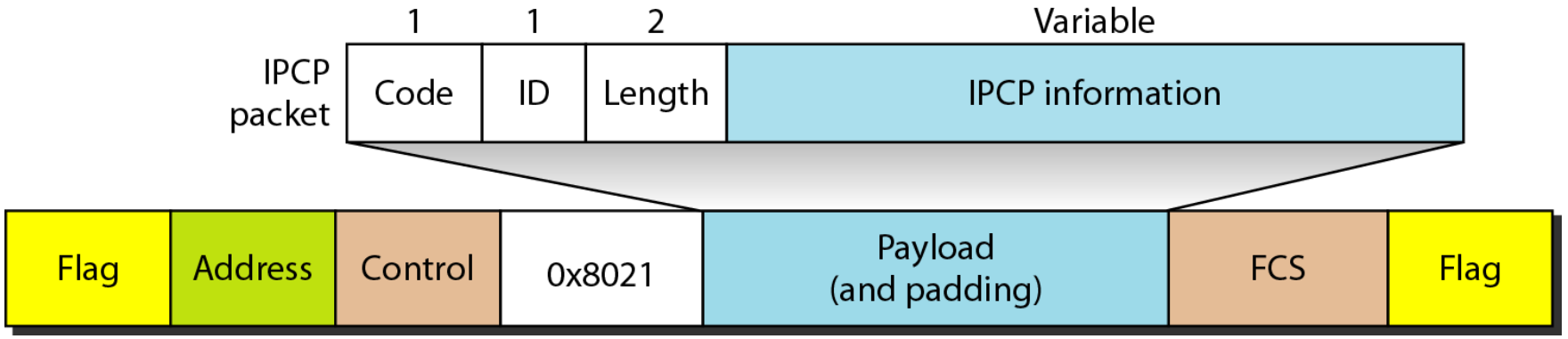
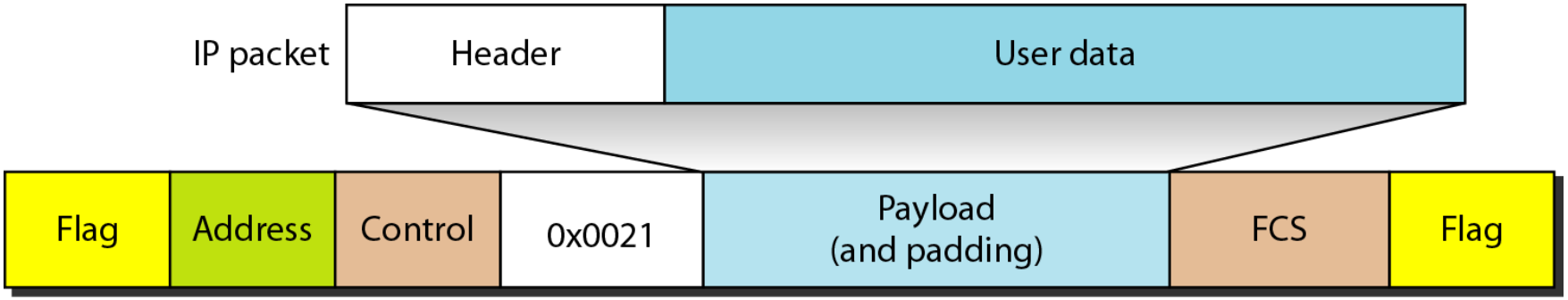
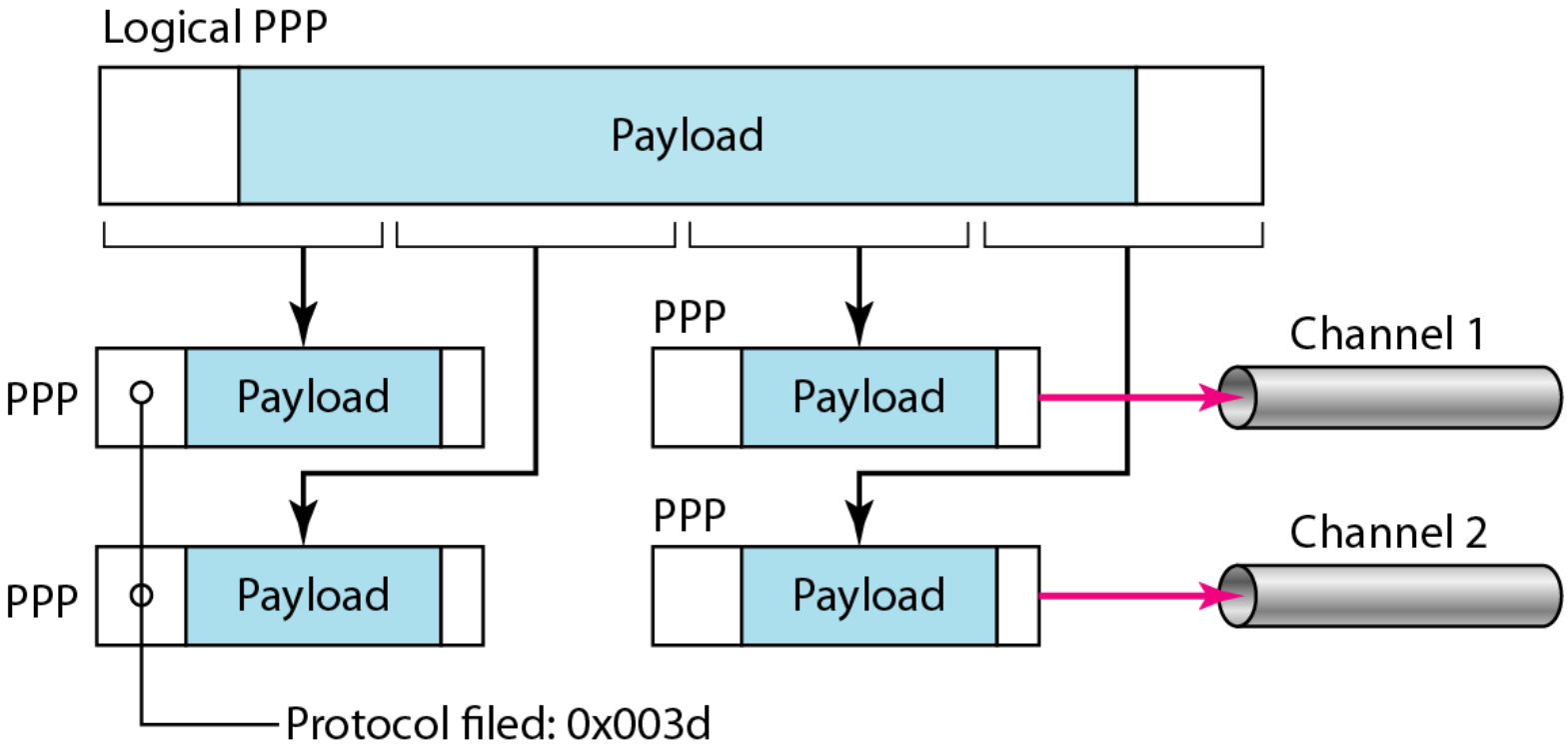


Table 2 ***Code value for IPCP packets***

IP datagram encapsulated in a PPP frame



Multilink PPP



Example 1

Let us go through the phases followed by a network layer packet as it is transmitted through a PPP connection. Figure 11.41 shows the steps. For simplicity, we assume unidirectional movement of data from the user site to the system site (such as sending an e-mail through an ISP).

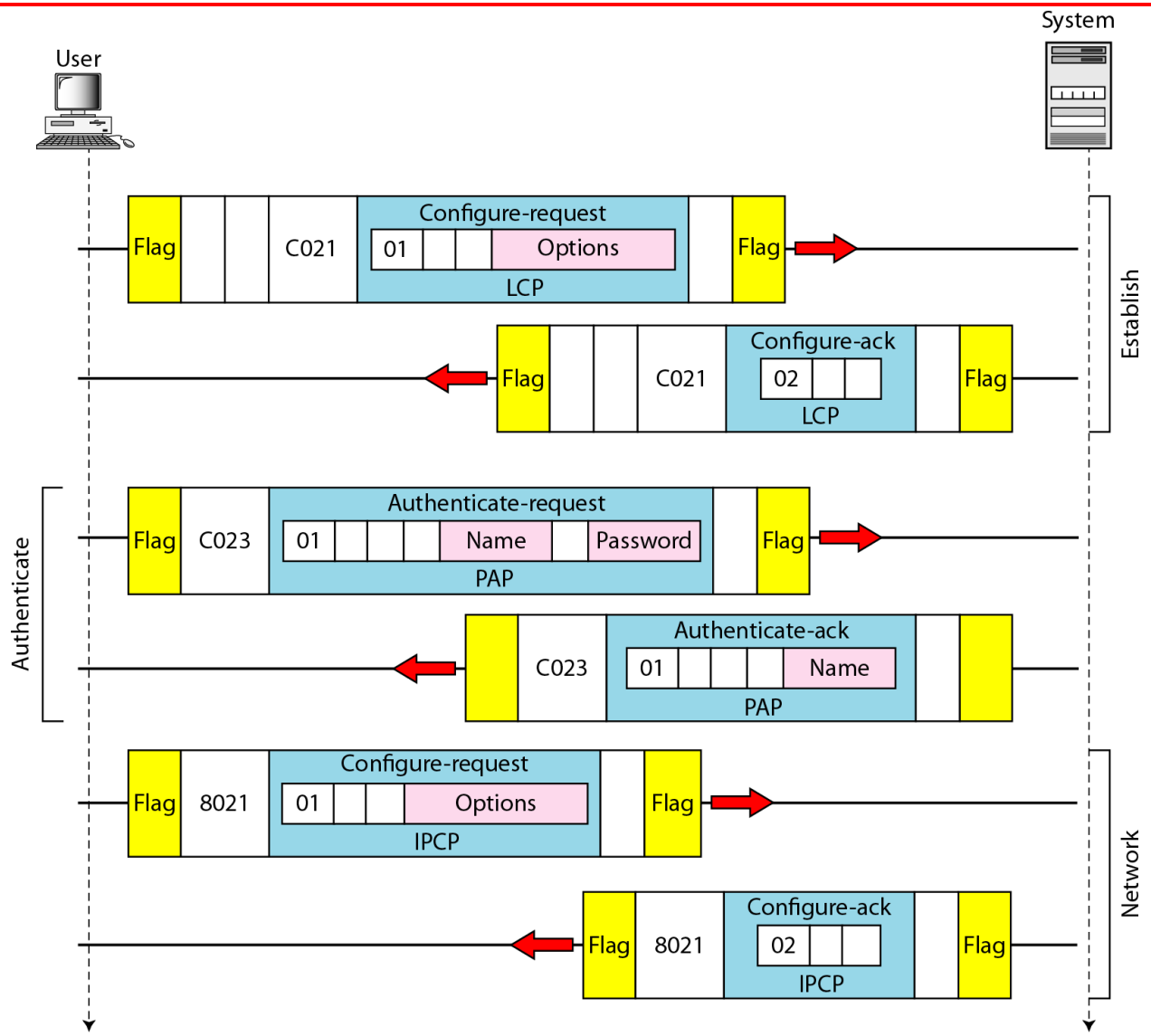
The first two frames show link establishment. We have chosen two options (not shown in the figure): using PAP for authentication and suppressing the address control fields. Frames 3 and 4 are for authentication. Frames 5 and 6 establish the network layer connection using IPCP.

Example 2(continued)

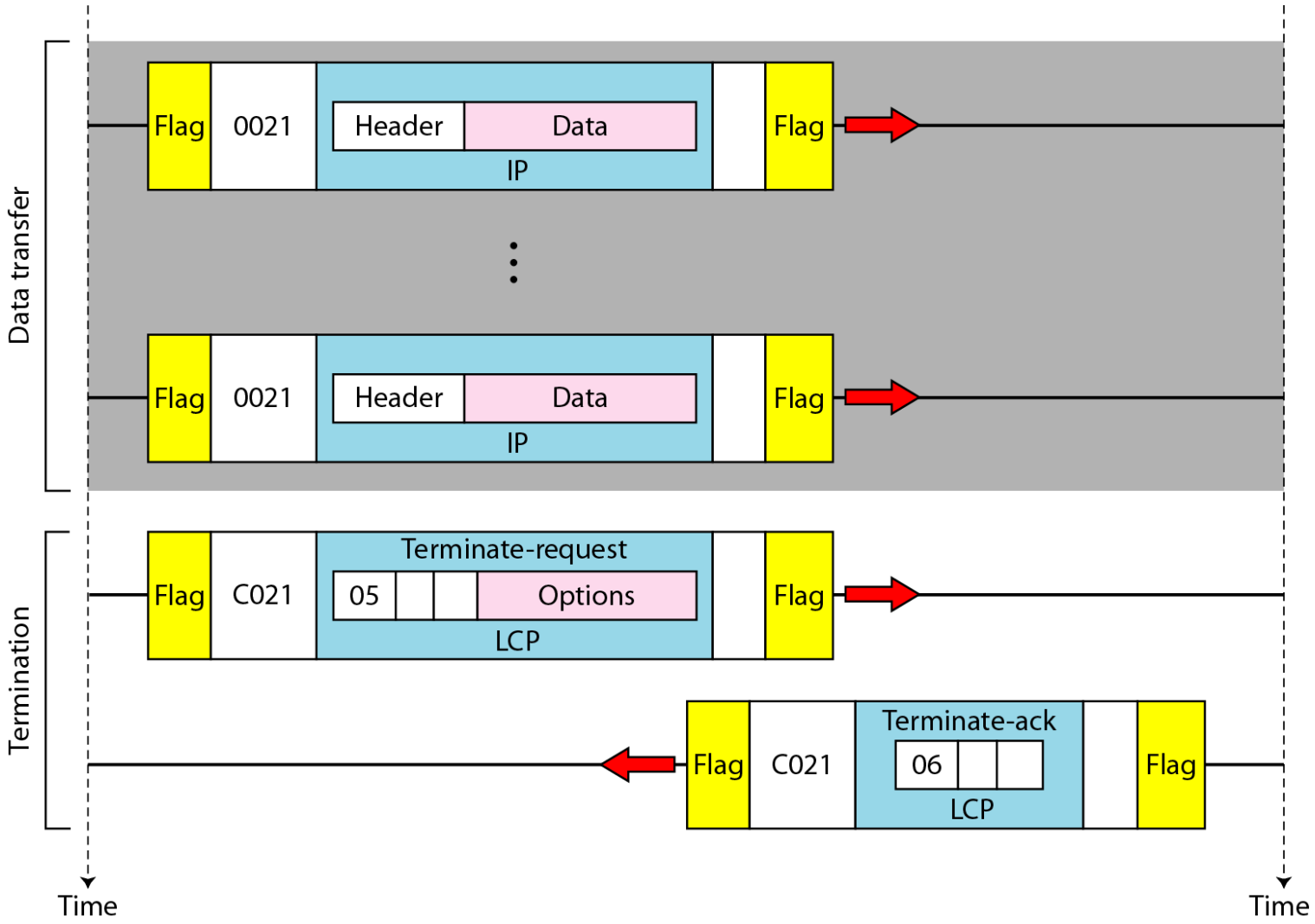
The next several frames show that some IP packets are encapsulated in the PPP frame. The system (receiver) may have been running several network layer protocols, but it knows that the incoming data must be delivered to the IP protocol because the NCP protocol used before the data transfer was IPCP.

After data transfer, the user then terminates the data link connection, which is acknowledged by the system. Of course the user or the system could have chosen to terminate the network layer IPCP and keep the data link layer running if it wanted to run another NCP protocol.

An example


User
System
Flag
C021
Configure-request
01
Options
LCP
Flag
Flag
C021
Configure-ack
02
LCP
Flag
Establish
Authenticate
Flag
C023
Authenticate-request
01
Name
Password
PAP
Flag
C023
Authenticate-ack
01
Name
PAP
Flag
8021
Configure-request
01
Options
IPCP
Flag
Flag
8021
Configure-ack
02
IPCP
Flag
Network

An example (continued)



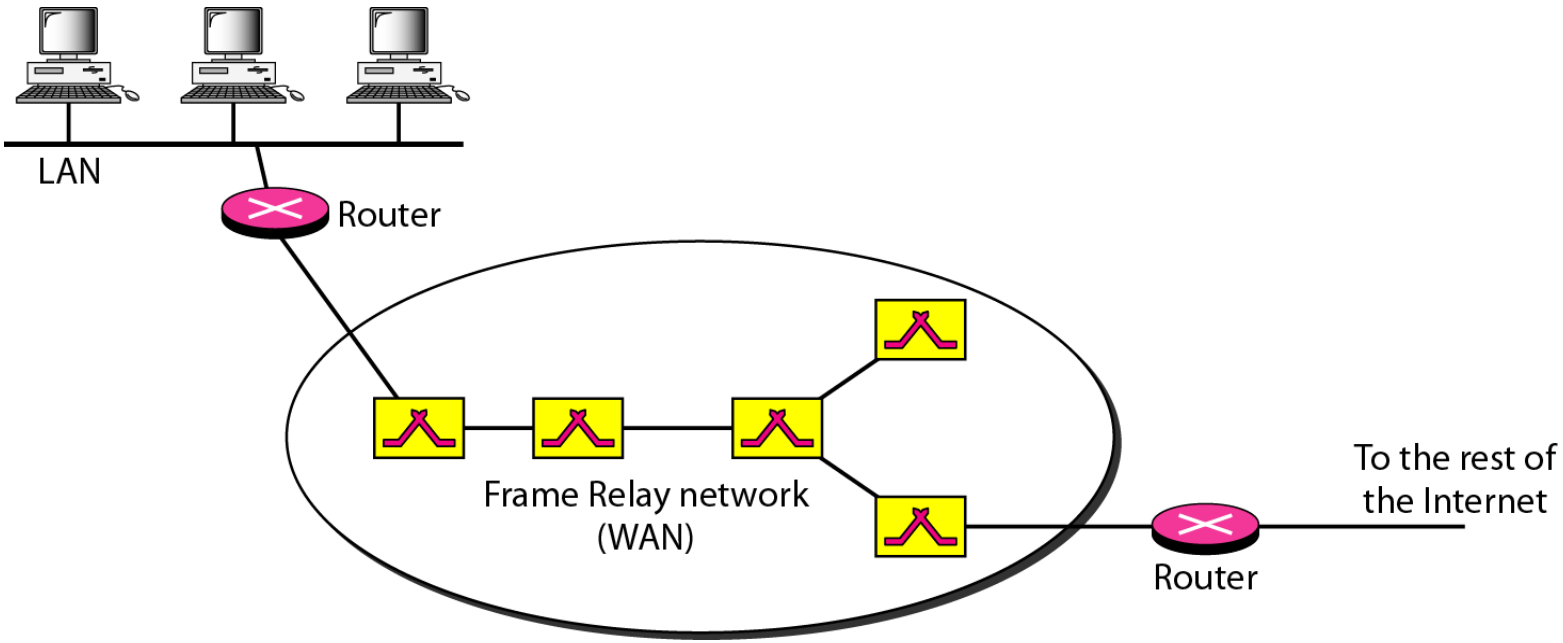
FRAME RELAY

Frame Relay is a virtual-circuit wide-area network that was designed in response to demands for a new type of WAN in the late 1980s and early 1990s.

Topics discussed in this section:

Architecture
Frame Relay Layers
Extended Address
FRADs
VOFR
LMI

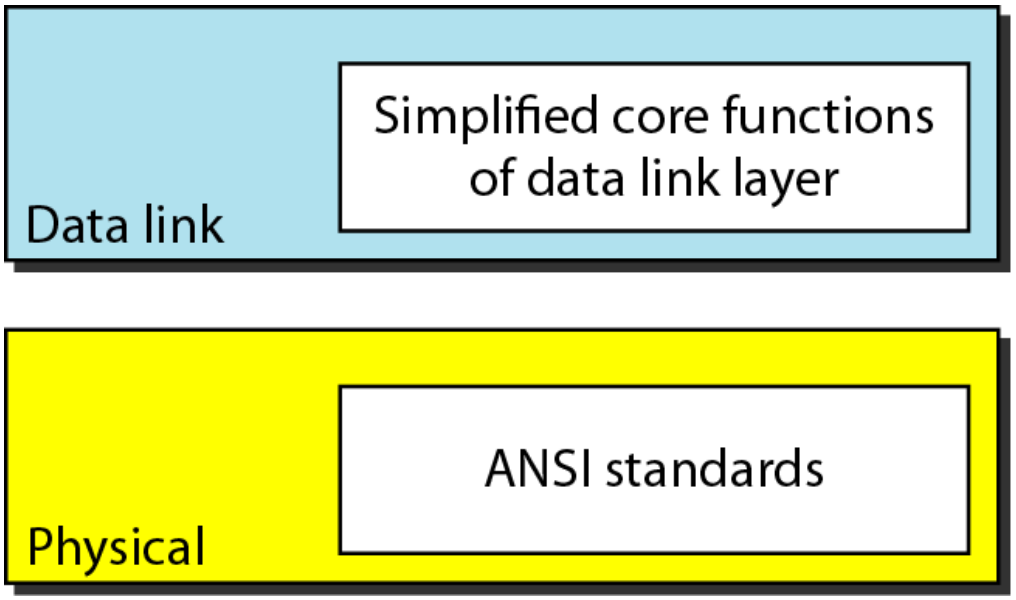
Frame Relay network



Note

VCIs in Frame Relay are called DLCIs.

Frame Relay layers



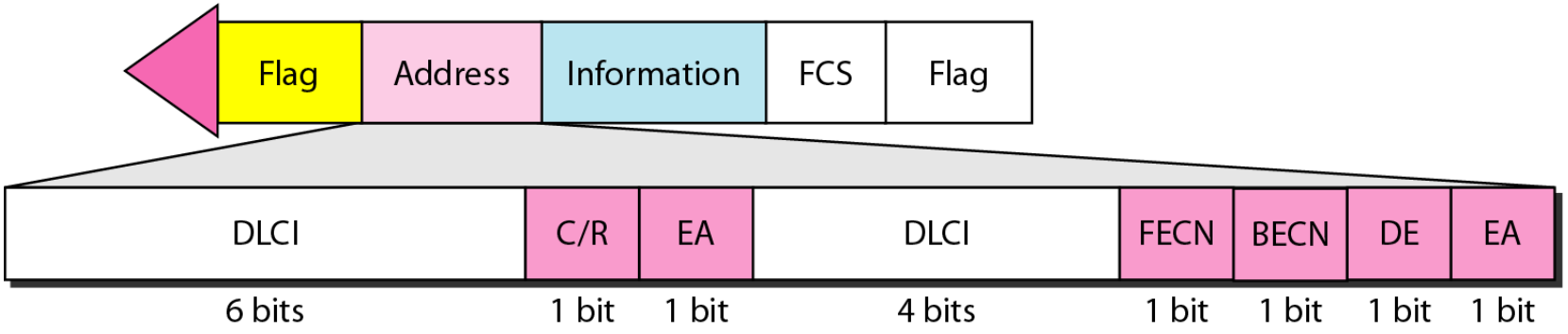
Note

Frame Relay operates only at the physical and data link layers.

Frame Relay frame

C/R: Command/response
EA: Extended address
FECN: Forward explicit congestion notification

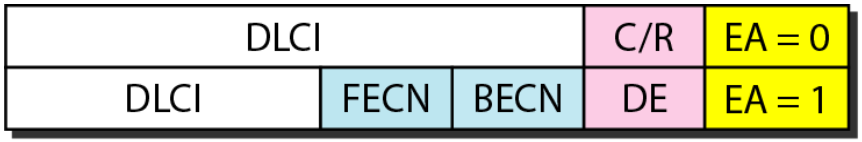
BECN: Backward explicit congestion notification
DE: Discard eligibility
DLCI: Data link connection identifier



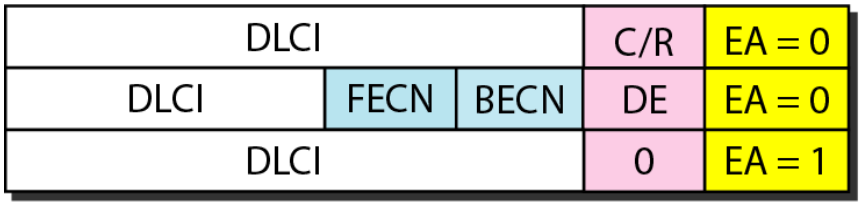
Note

Frame Relay does not provide flow or error control; they must be provided by the upper-layer protocols.

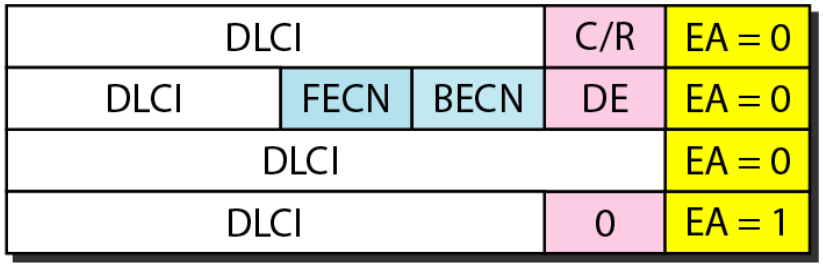
Three address formats



a. Two-byte address (10-bit DLCI)

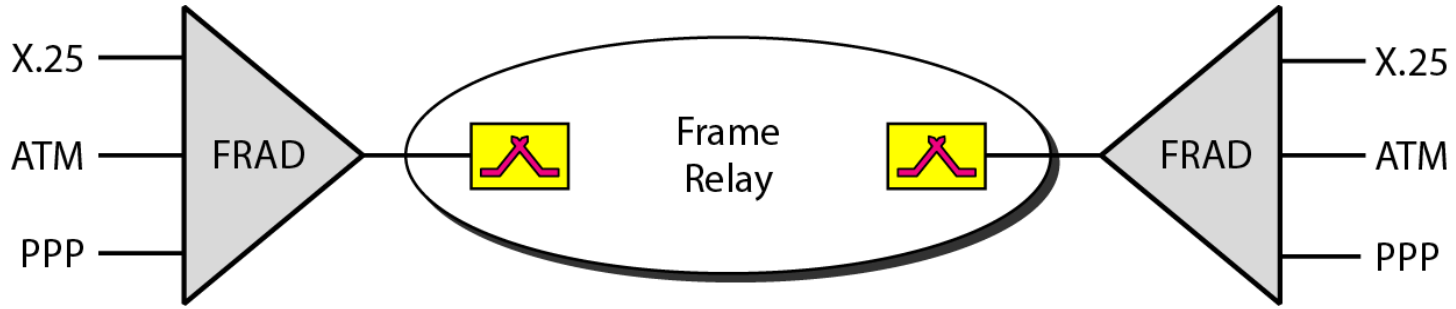


b. Three-byte address (16-bit DLCI)



c. Four-byte address (23-bit DLCI)

FRAD



IEEE 802.11

IEEE has defined the specifications for a wireless LAN, called IEEE 802.11, which covers the physical and data link layers.

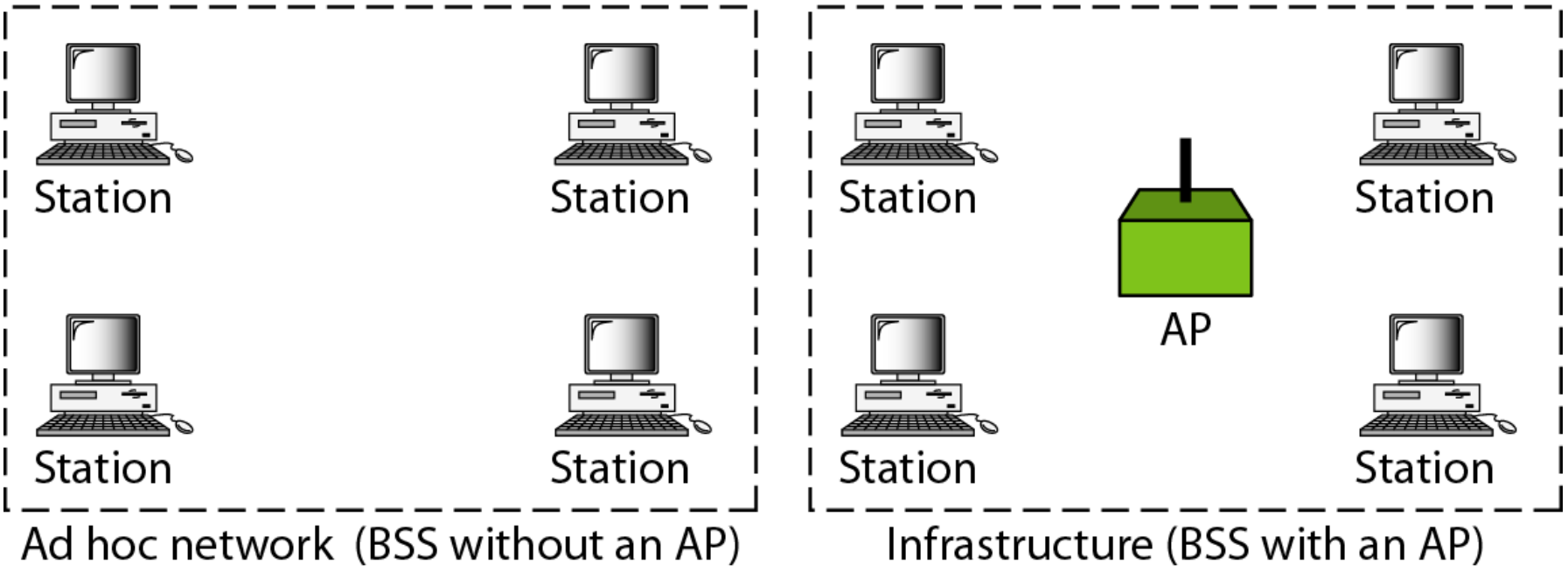
Architecture
MAC Sublayer
Physical Layer

Note

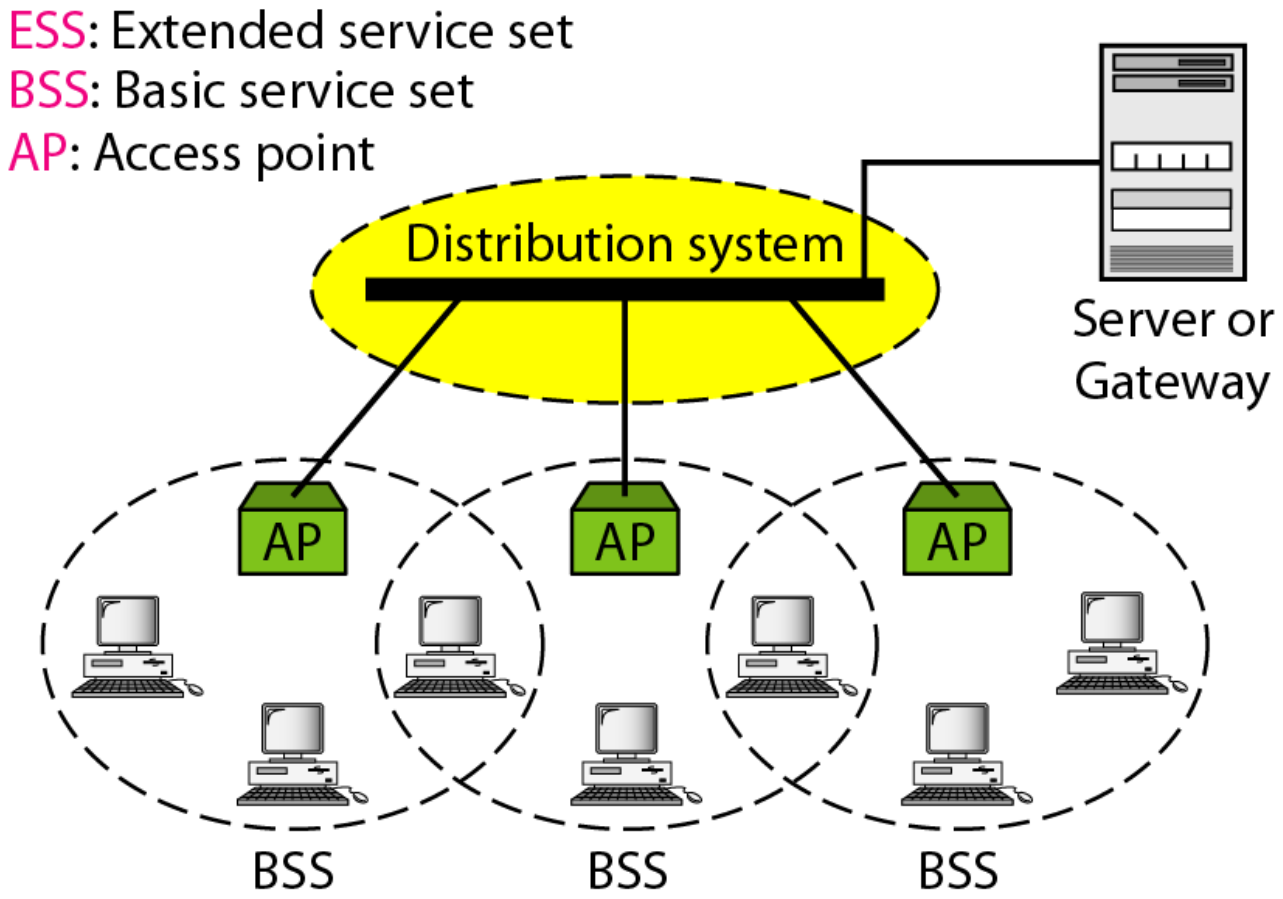
A BSS without an AP is called an ad hoc network; a BSS with an AP is called an infrastructure network.

Basic service sets (BSSs)

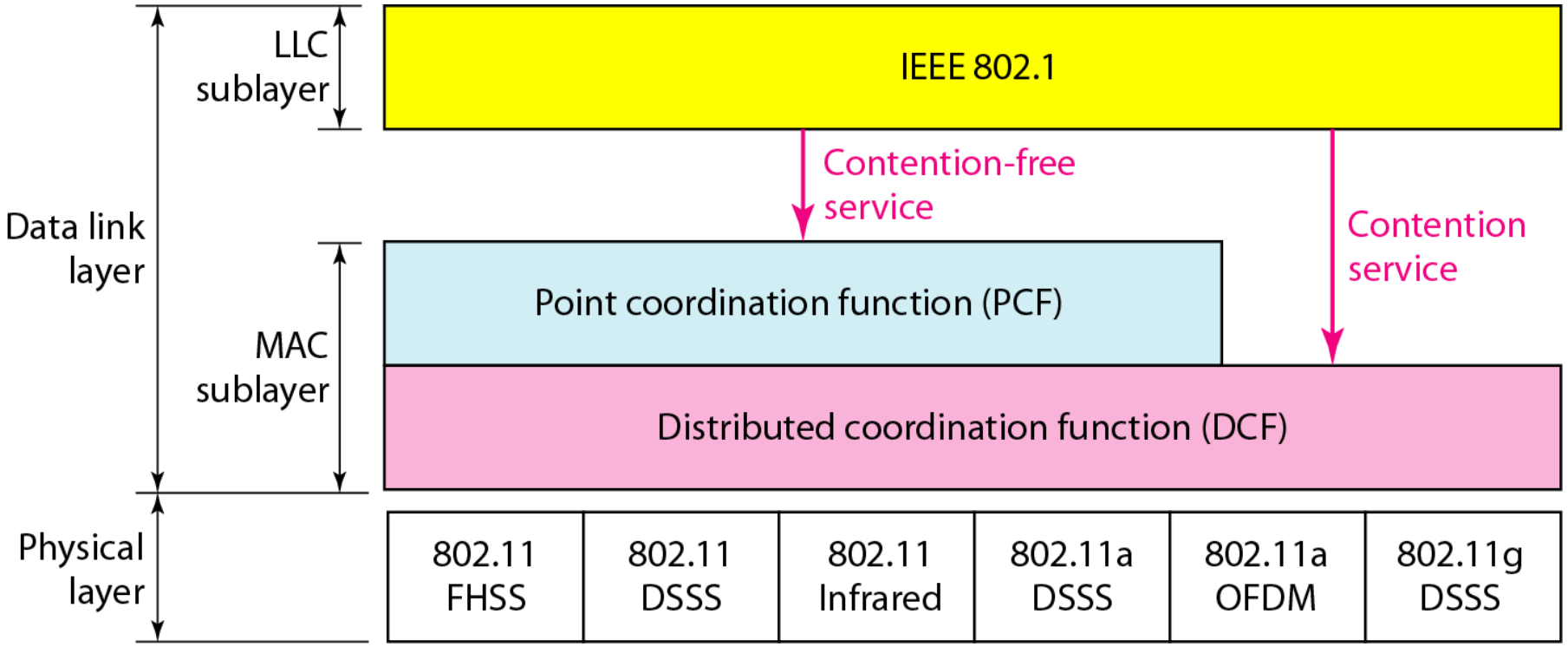
BSS: Basic service set
AP: Access point



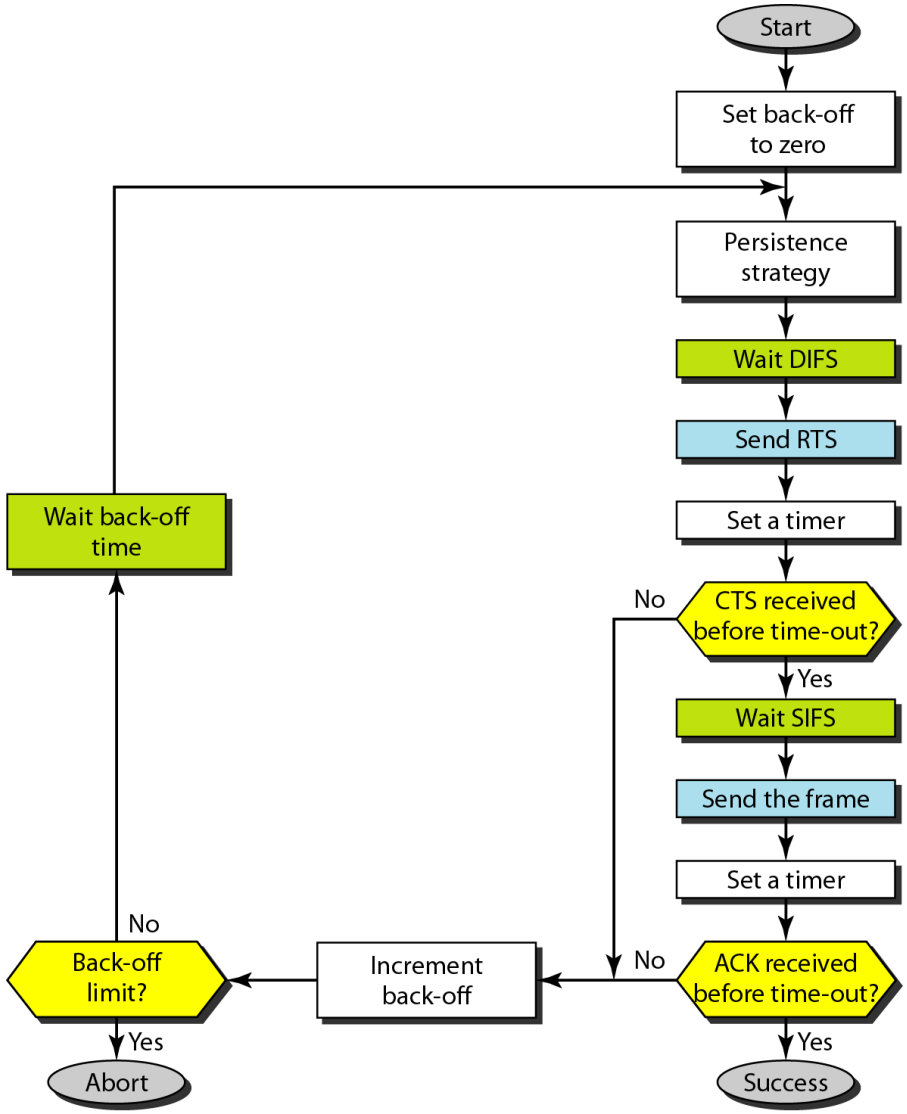
Extended service sets (ESSs)



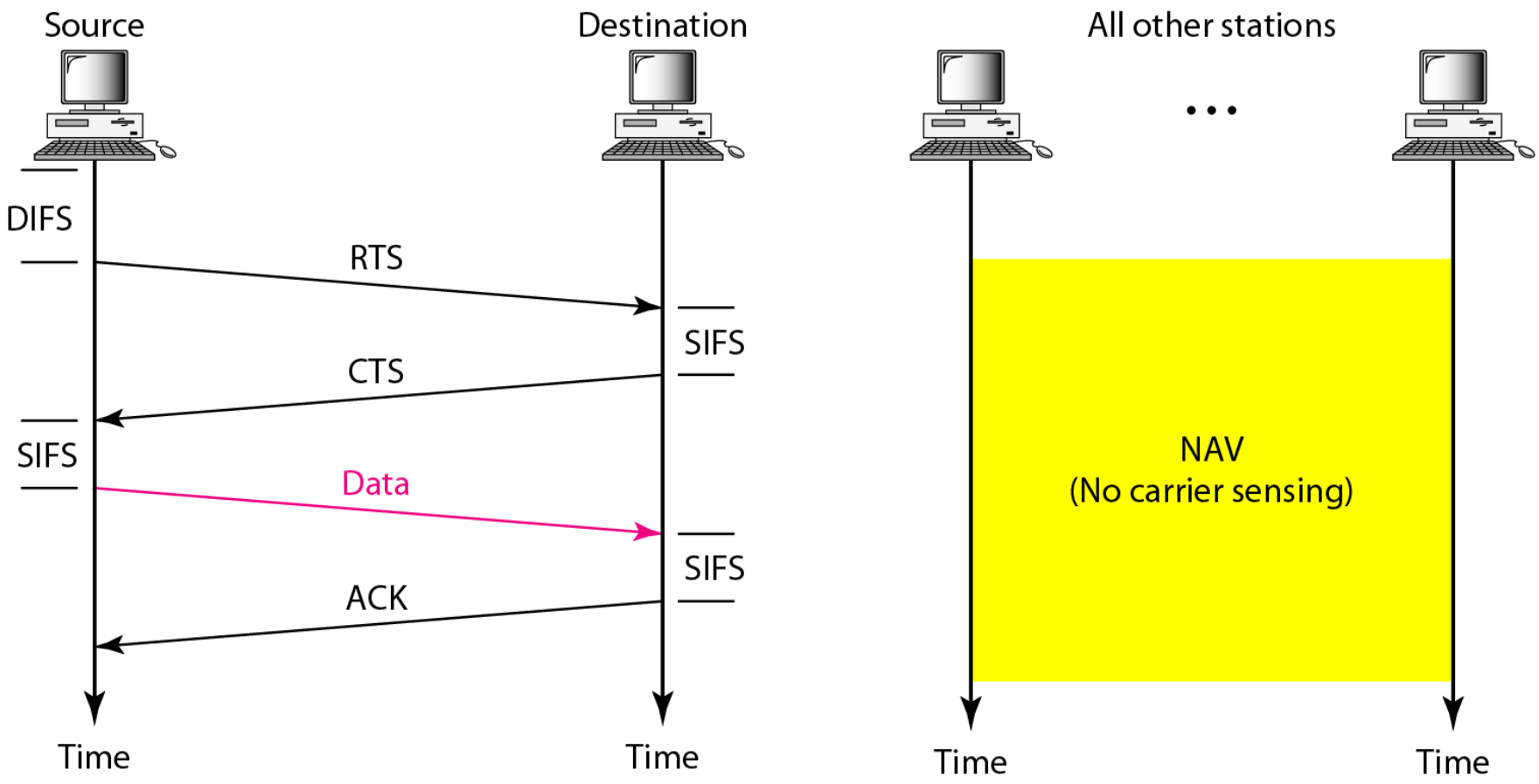
MAC layers in IEEE 802.11 standard



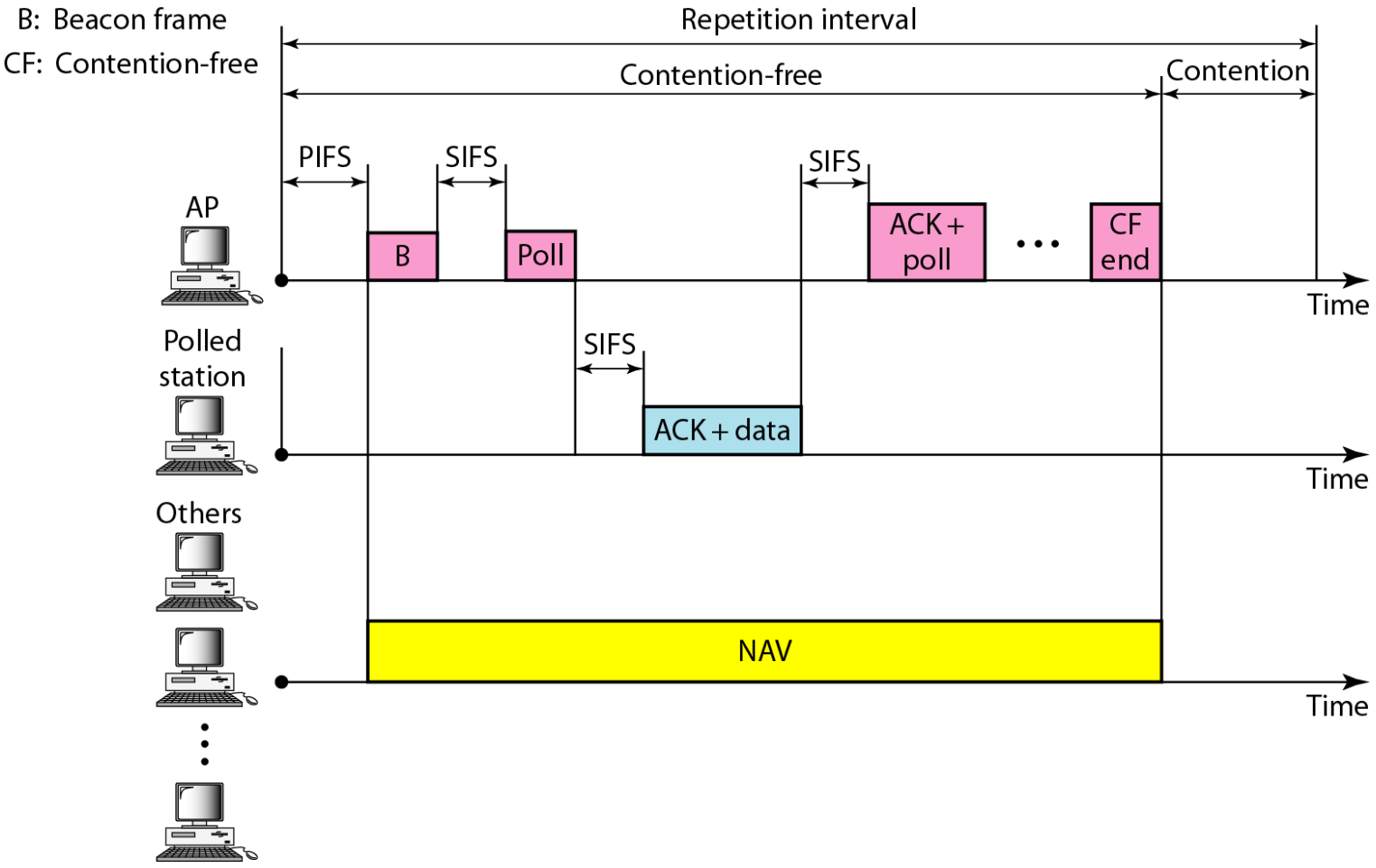
CSMA/CA flowchart



CSMA/CA and NAV



Example of repetition interval



Frame format

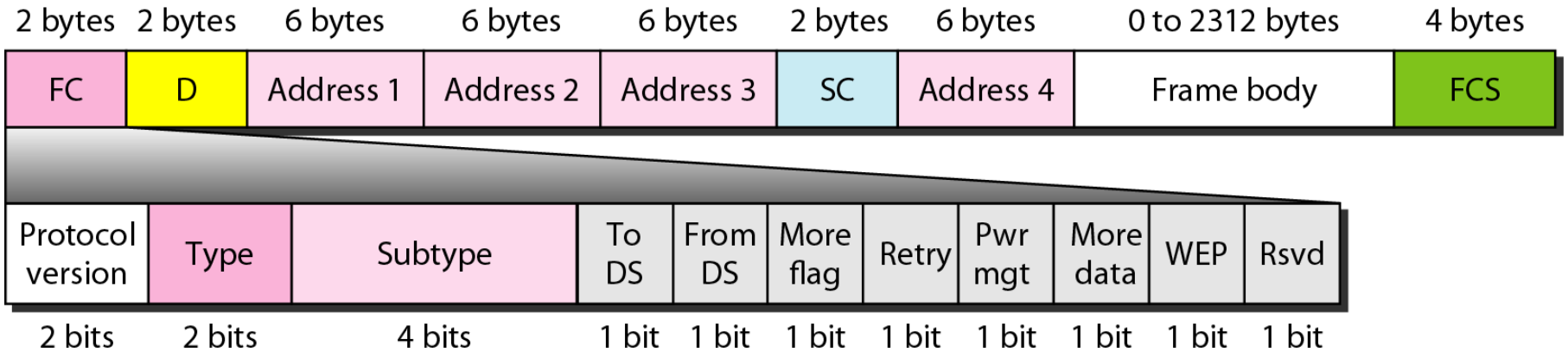


Table 1 ***Subfields in FC field***

Field	*Explanation*
Version	Current version is 0
Type	Type of information: management (00), control (01), or data (10)
Subtype	Subtype of each type (see Table 14.2)
To DS	Defined later
From DS	Defined later
More flag	When set to 1, means more fragments
Retry	When set to 1, means retransmitted frame
Pwr mgt	When set to 1, means station is in power management mode
More data	When set to 1, means station has more data to send
WEP	Wired equivalent privacy (encryption implemented)
Rsvd	Reserved

Control frames

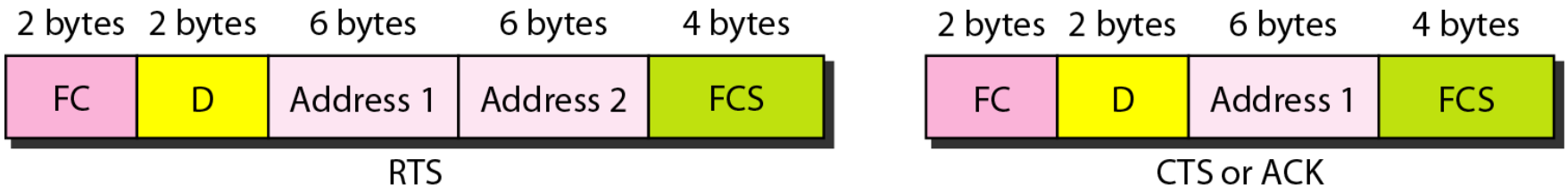
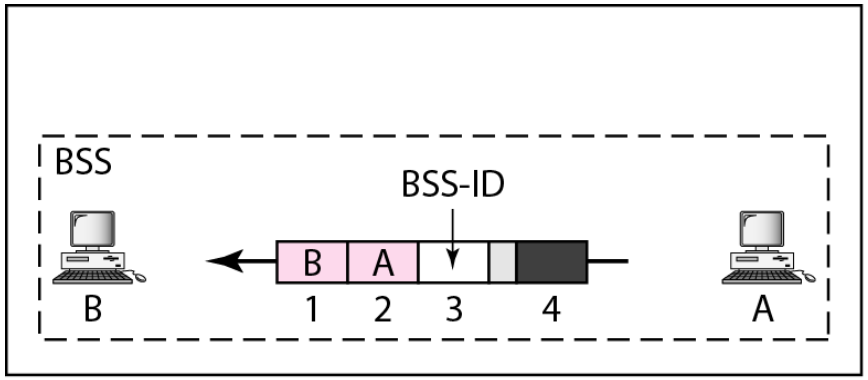


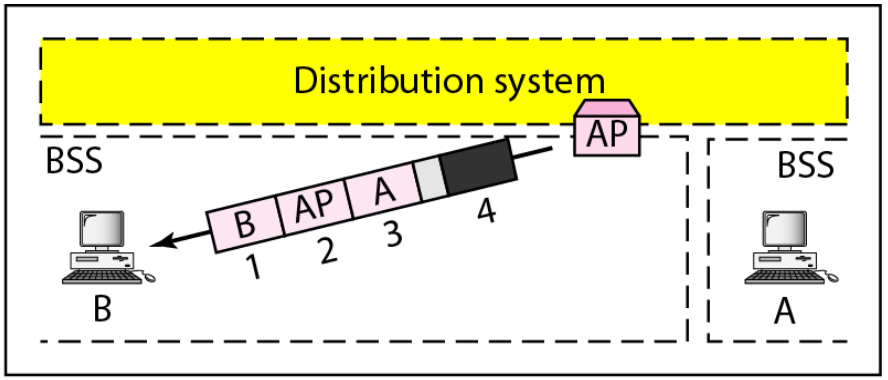
Table 2 *Values of subfields in control frames*

Table 3 ***Addresses***

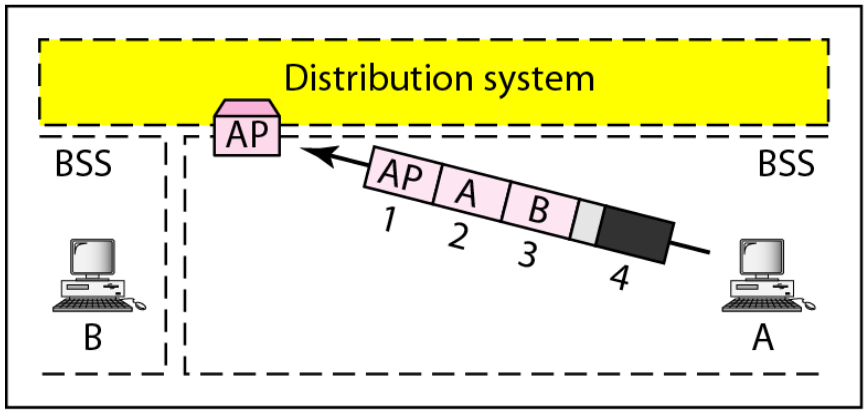
Addressing mechanisms



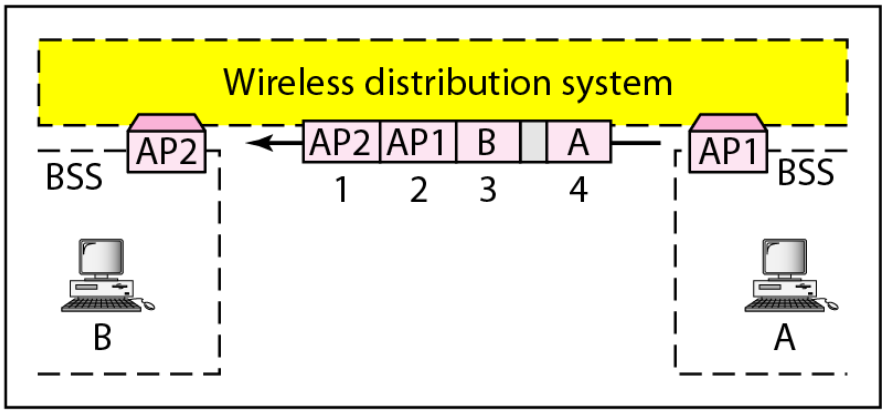
a. Case 1



b. Case 2

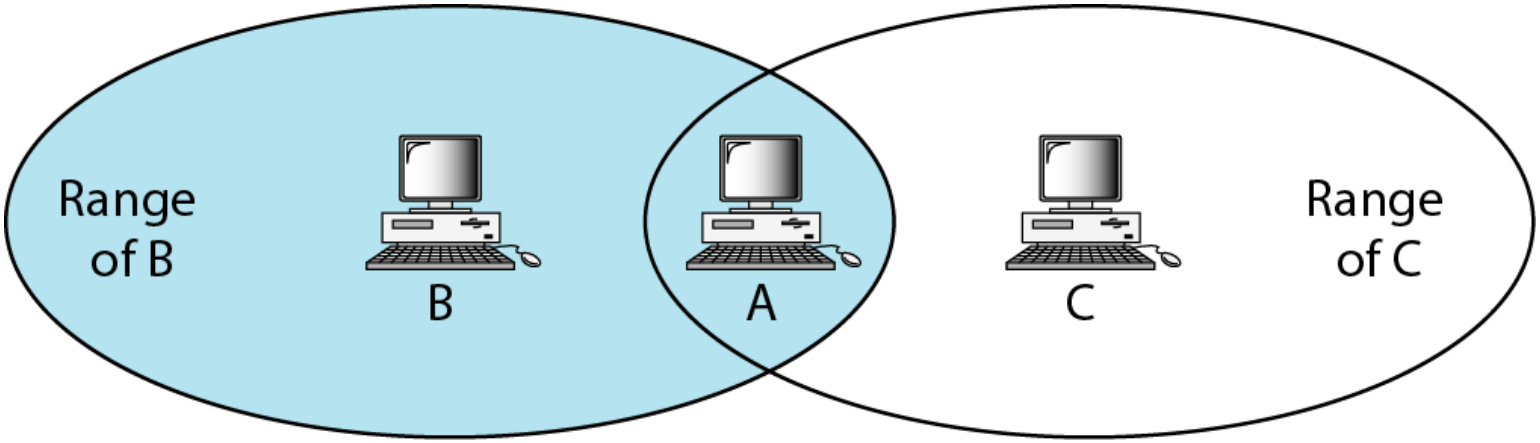


c. Case 3



d. Case 4

Hidden station problem

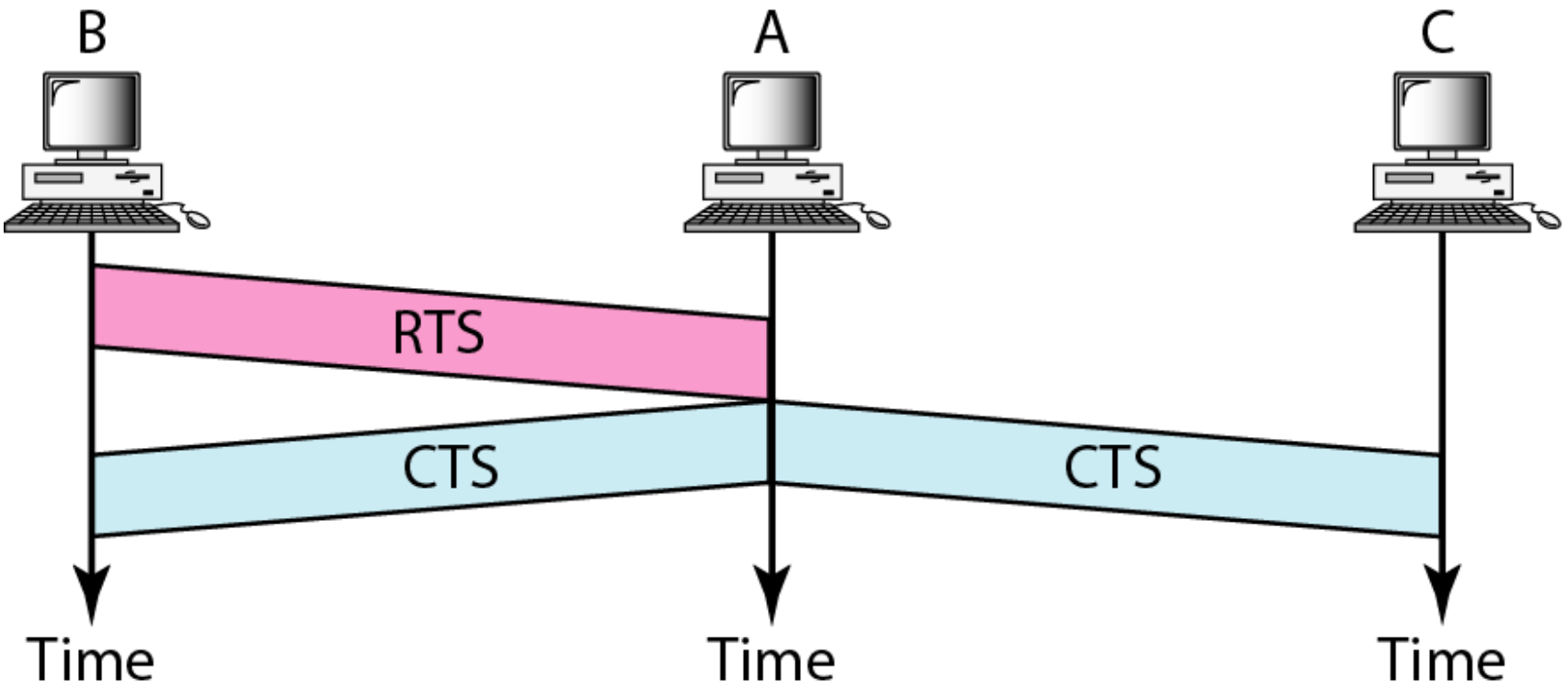


B and C are hidden from each other with respect to A.

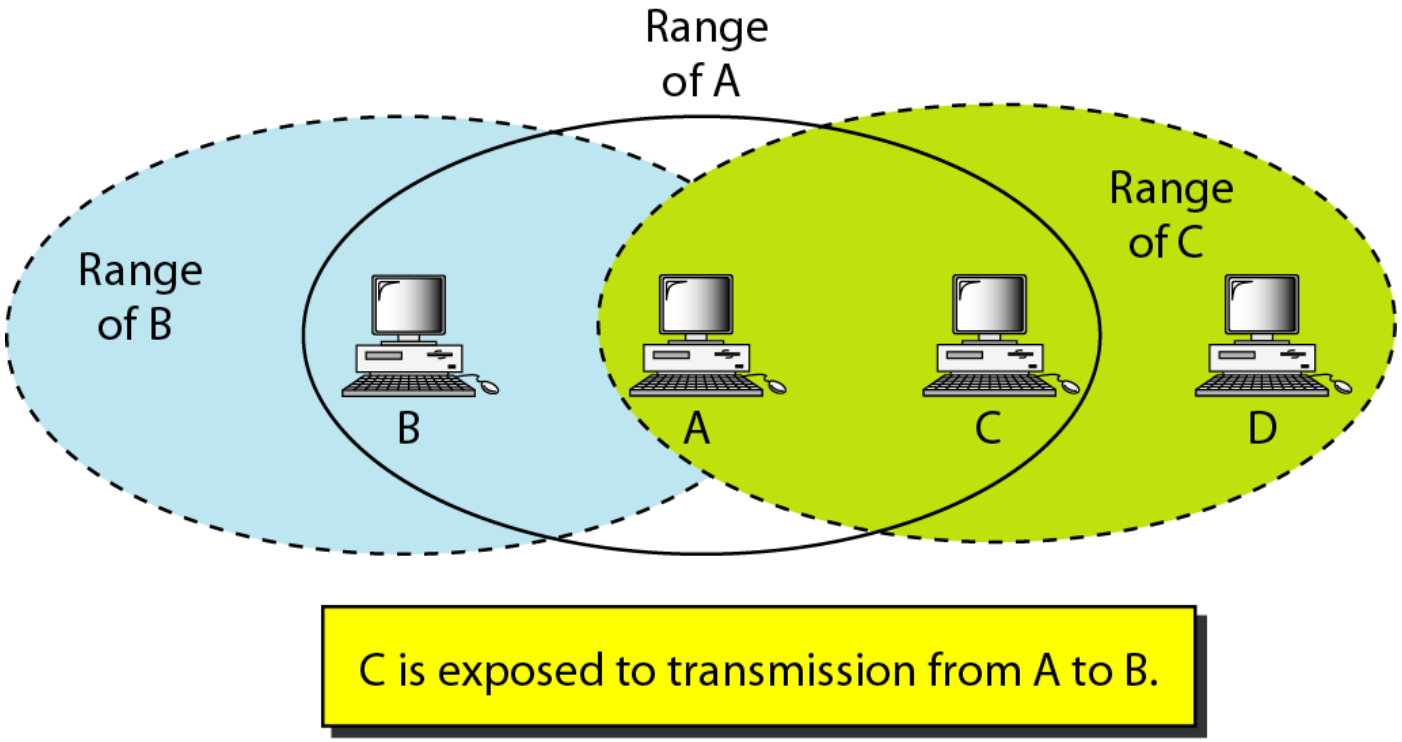
Note

The CTS frame in CSMA/CA handshake can prevent collision from a hidden station.

Use of handshaking to prevent hidden station problem



Exposed station problem



Use of handshaking in exposed station problem

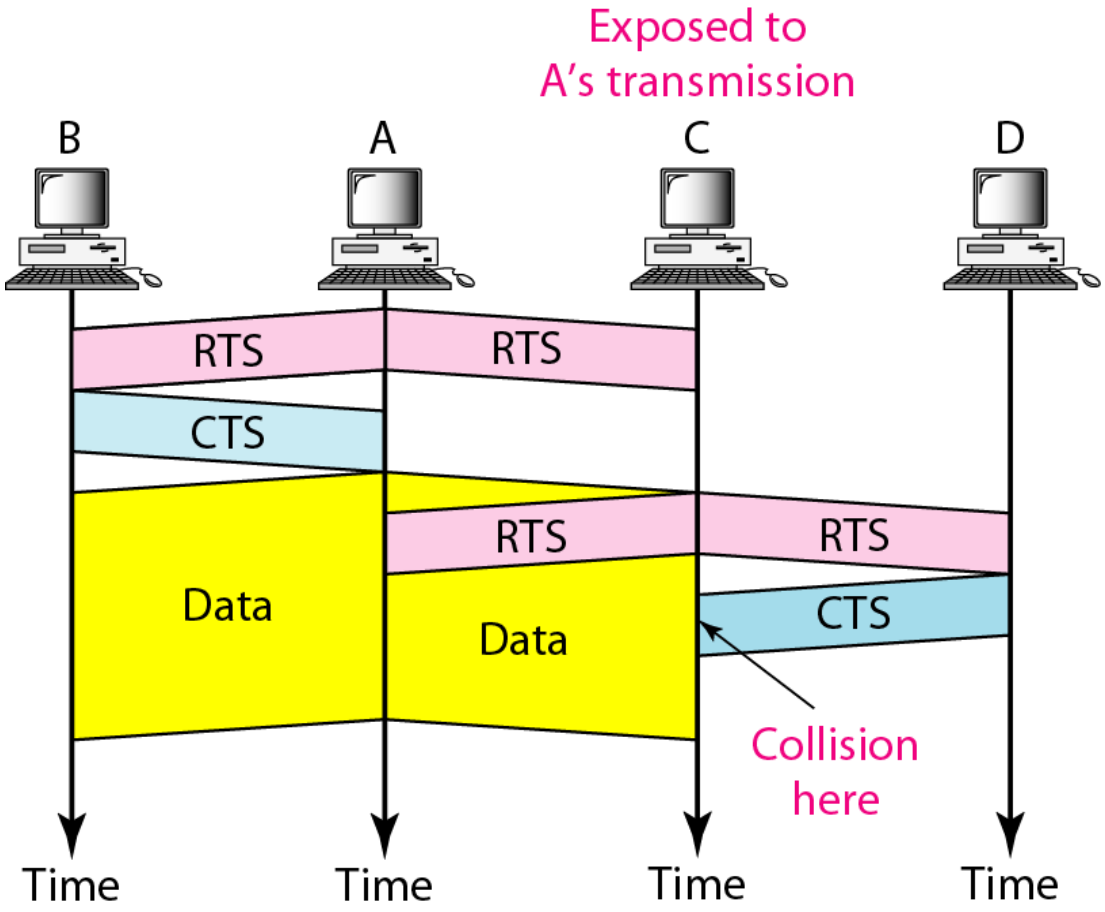
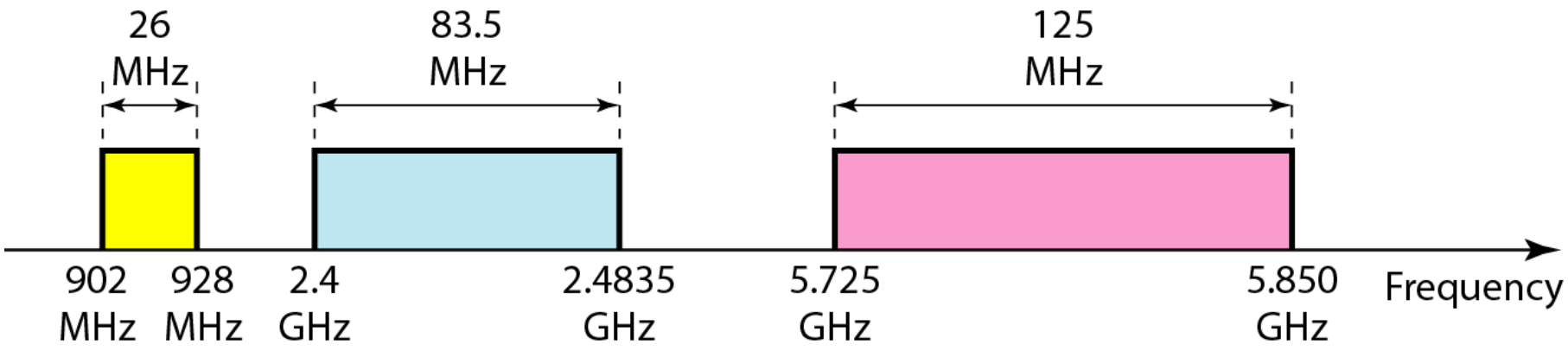


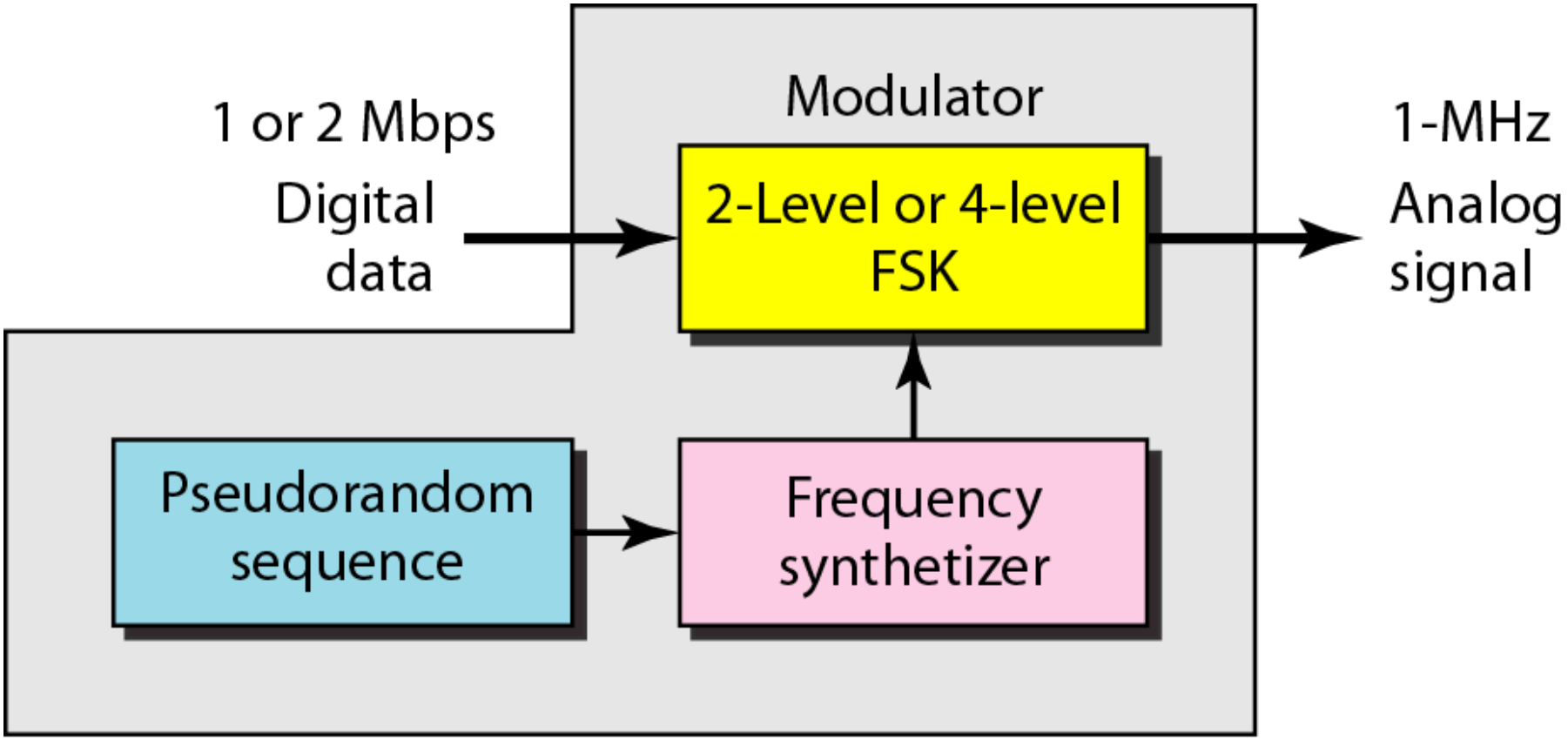
Table 4 ***Physical layers***

IEEE	*Technique*	*Band*	*Modulation*	*Rate (Mbps)*
802.11	FHSS	2.4 GHz	FSK	1 and 2
	DSSS	2.4 GHz	PSK	1 and 2
		Infrared	PPM	1 and 2
802.11a	OFDM	5.725 GHz	PSK or QAM	6 to 54
802.11b	DSSS	2.4 GHz	PSK	5.5 and 11
802.11g	OFDM	2.4 GHz	Different	22 and 54

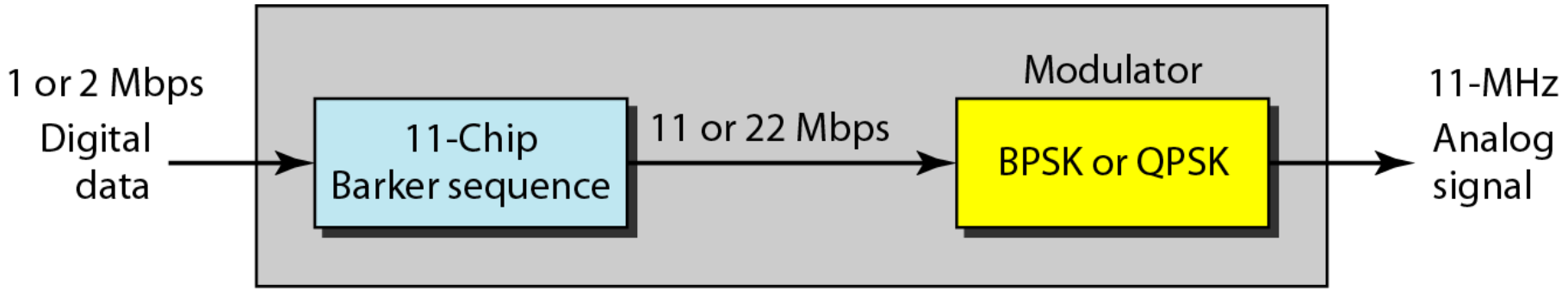
Industrial, scientific, and medical (ISM) band



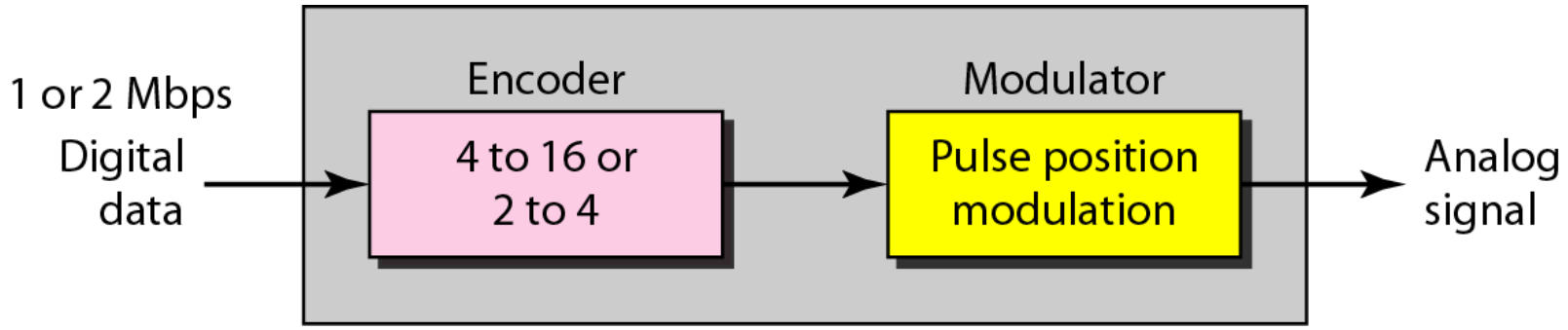
Physical layer of IEEE 802.11 FHSS



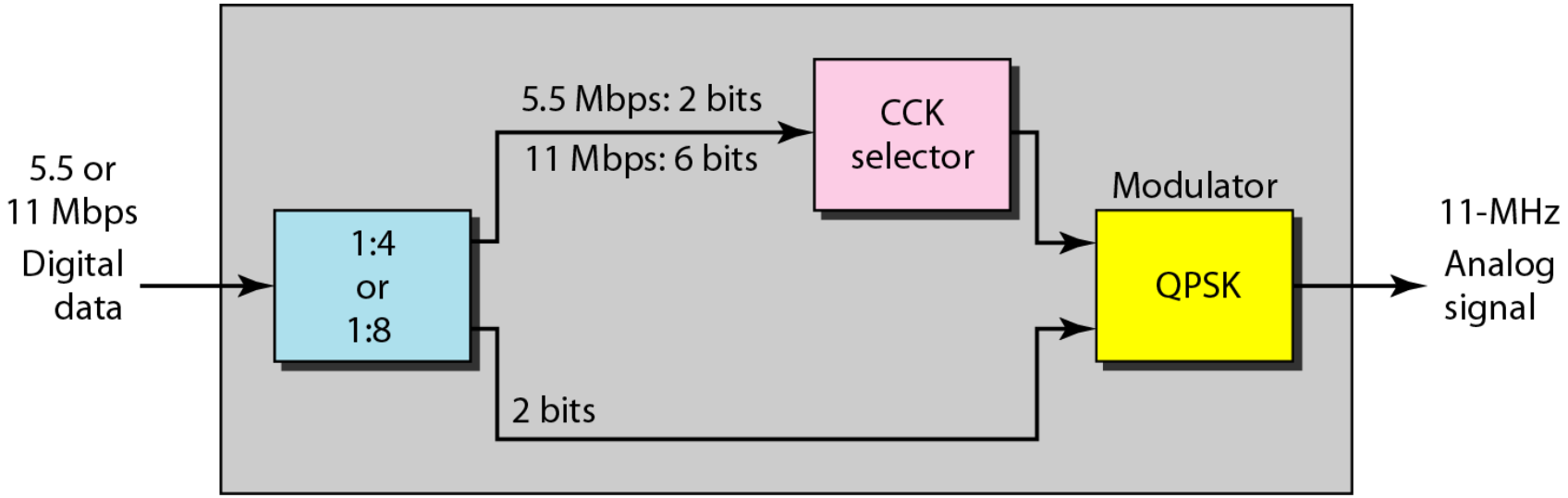
Physical layer of IEEE 802.11 DSSS



Physical layer of IEEE 802.11 infrared



Physical layer of IEEE 802.11b



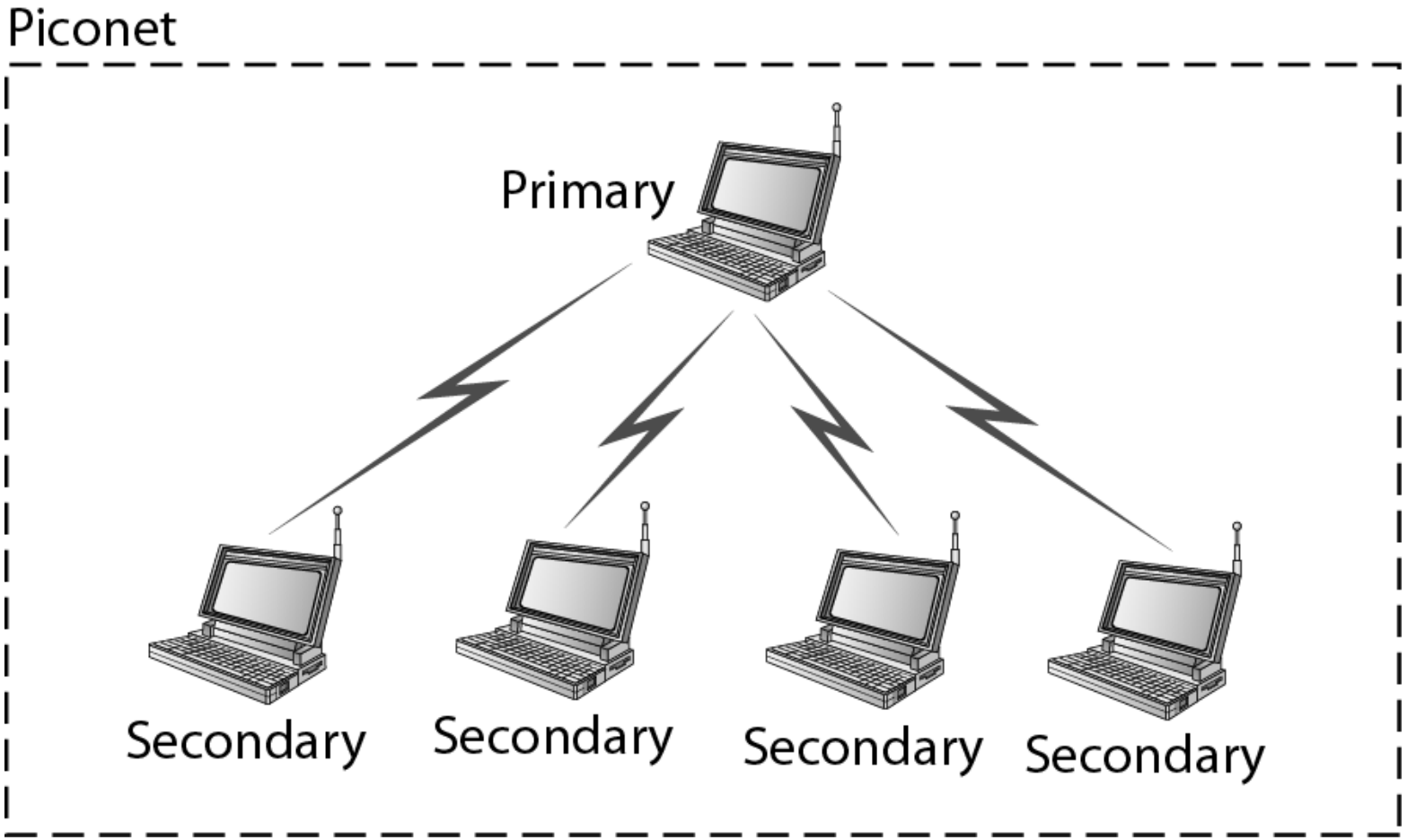
BLUETOOTH

***Bluetooth** is a wireless LAN technology designed to connect devices of different functions such as telephones, notebooks, computers, cameras, printers, coffee makers, and so on. A Bluetooth LAN is an ad hoc network, which means that the network is formed spontaneously.*

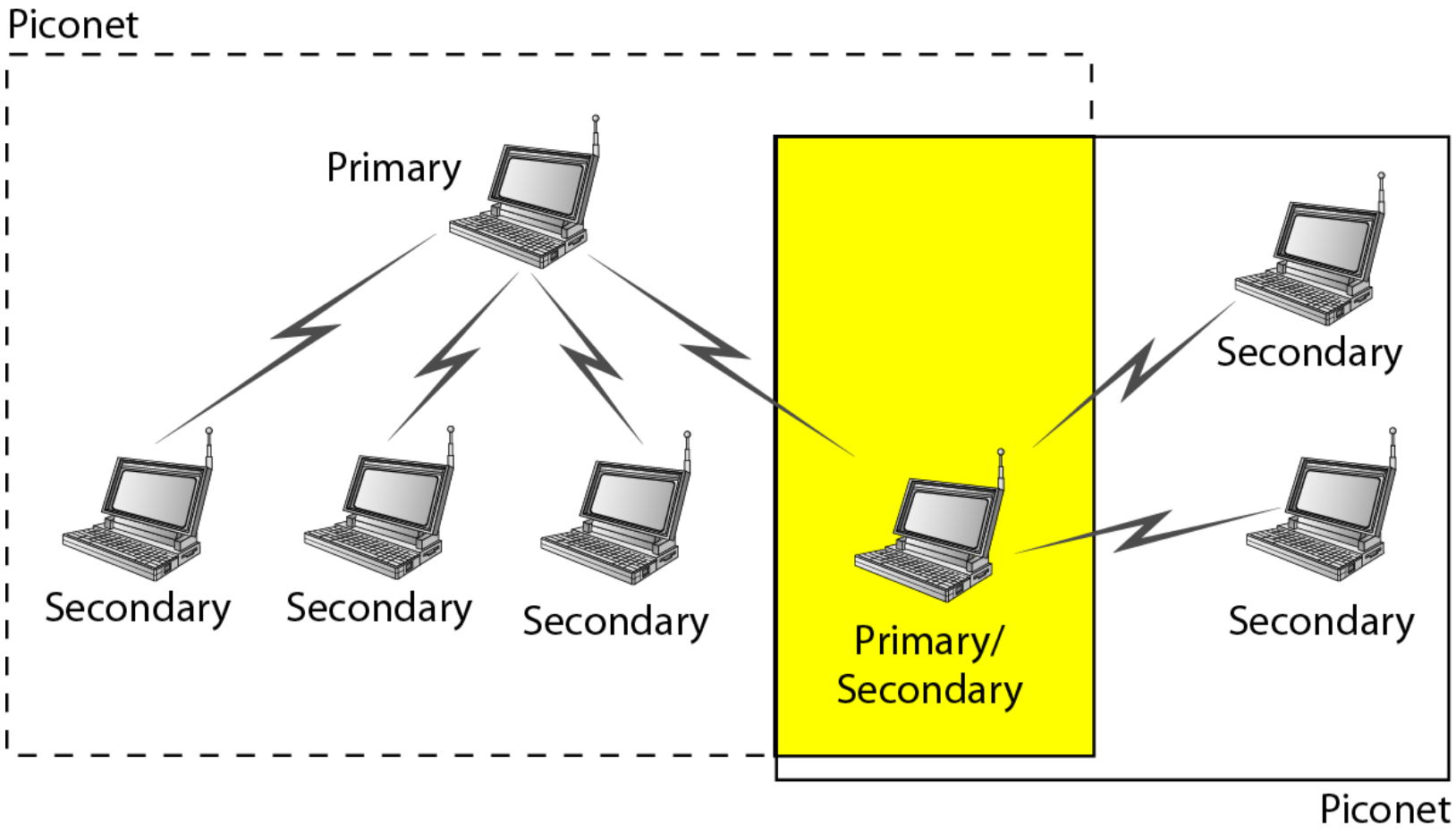
Topics discussed in this section:

Architecture
Bluetooth Layers
Baseband Layer
L2CAP

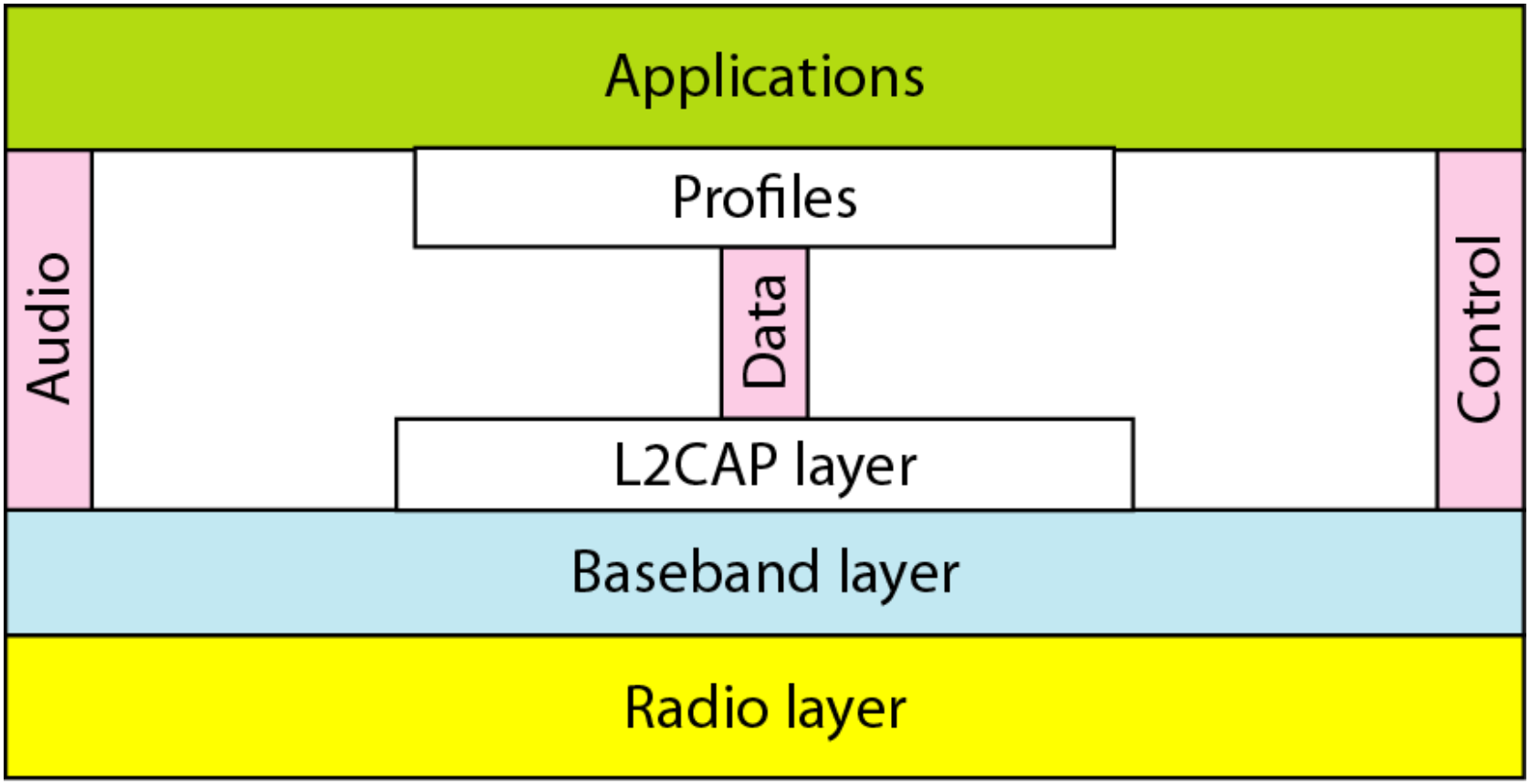
Piconet


Piconet
Primary
Secondary
Secondary
Secondary
Secondary

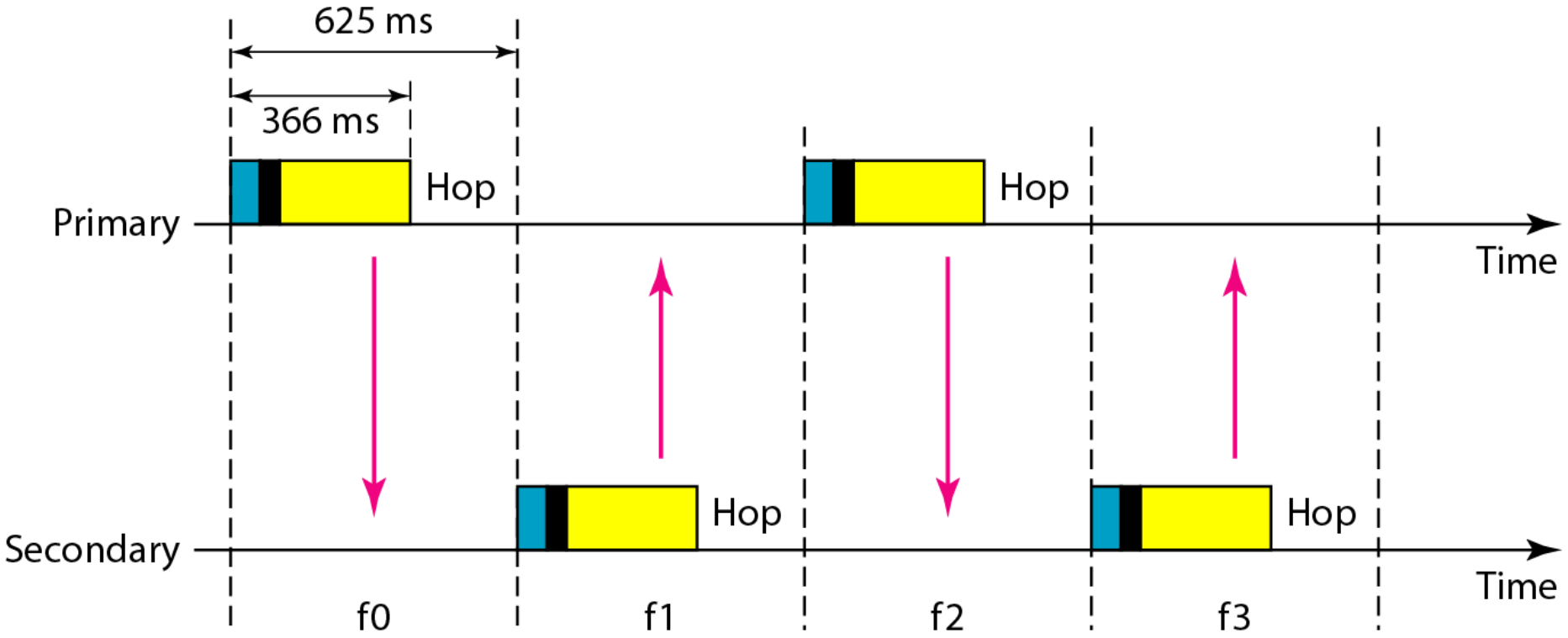
Scatternet


Piconet
Primary
Secondary
Secondary
Secondary
Primary/
Secondary
Secondary
Secondary
Piconet

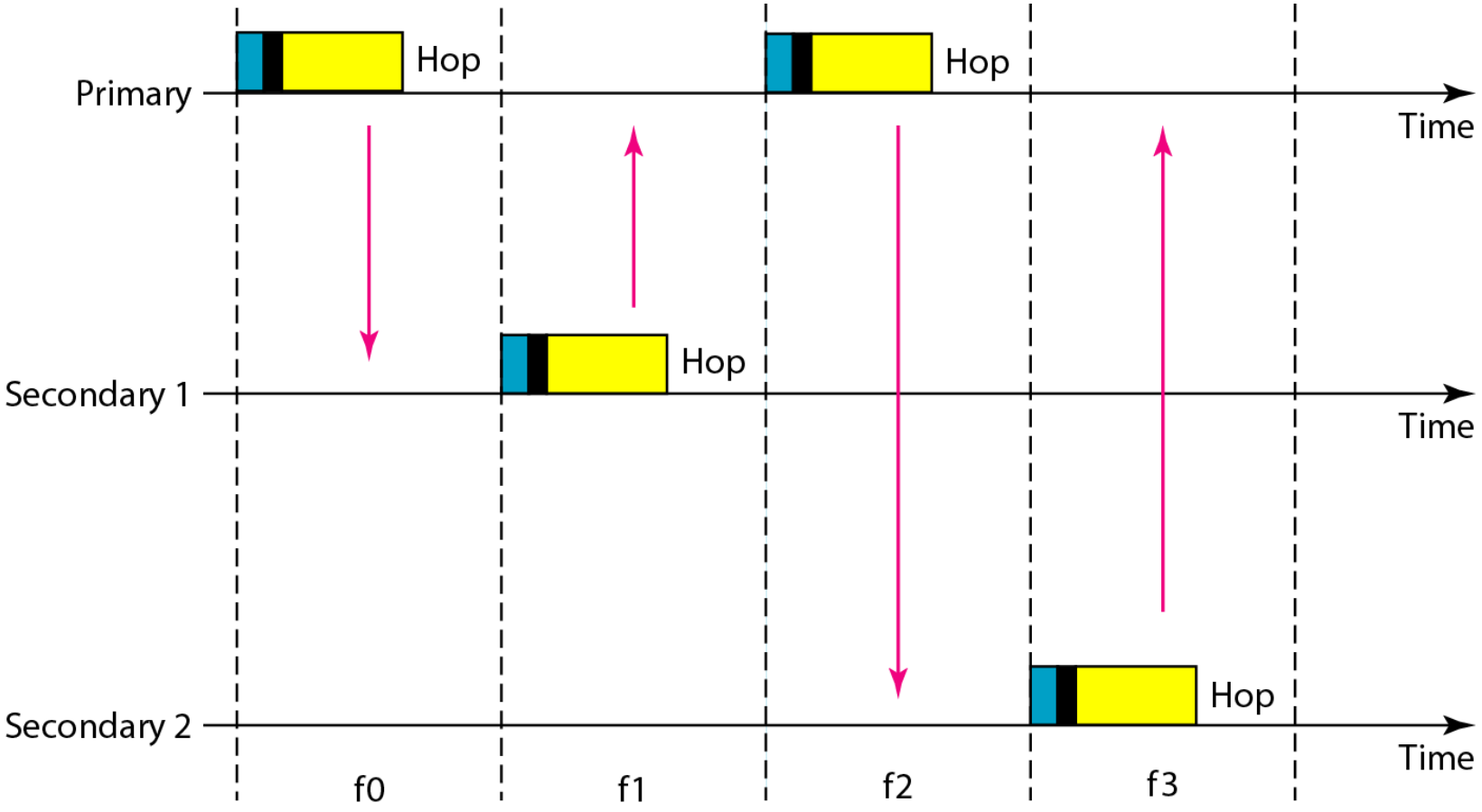
Bluetooth layers



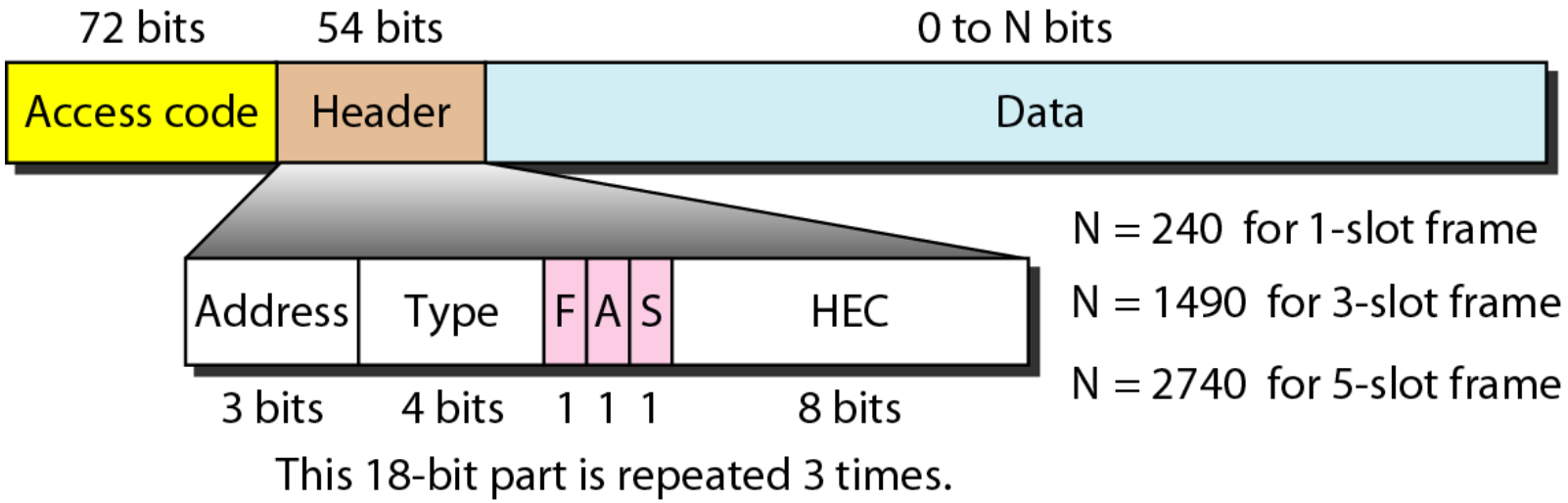
Single-secondary communication



Multiple-secondary communication



Frame format types



L2CAP data packet format