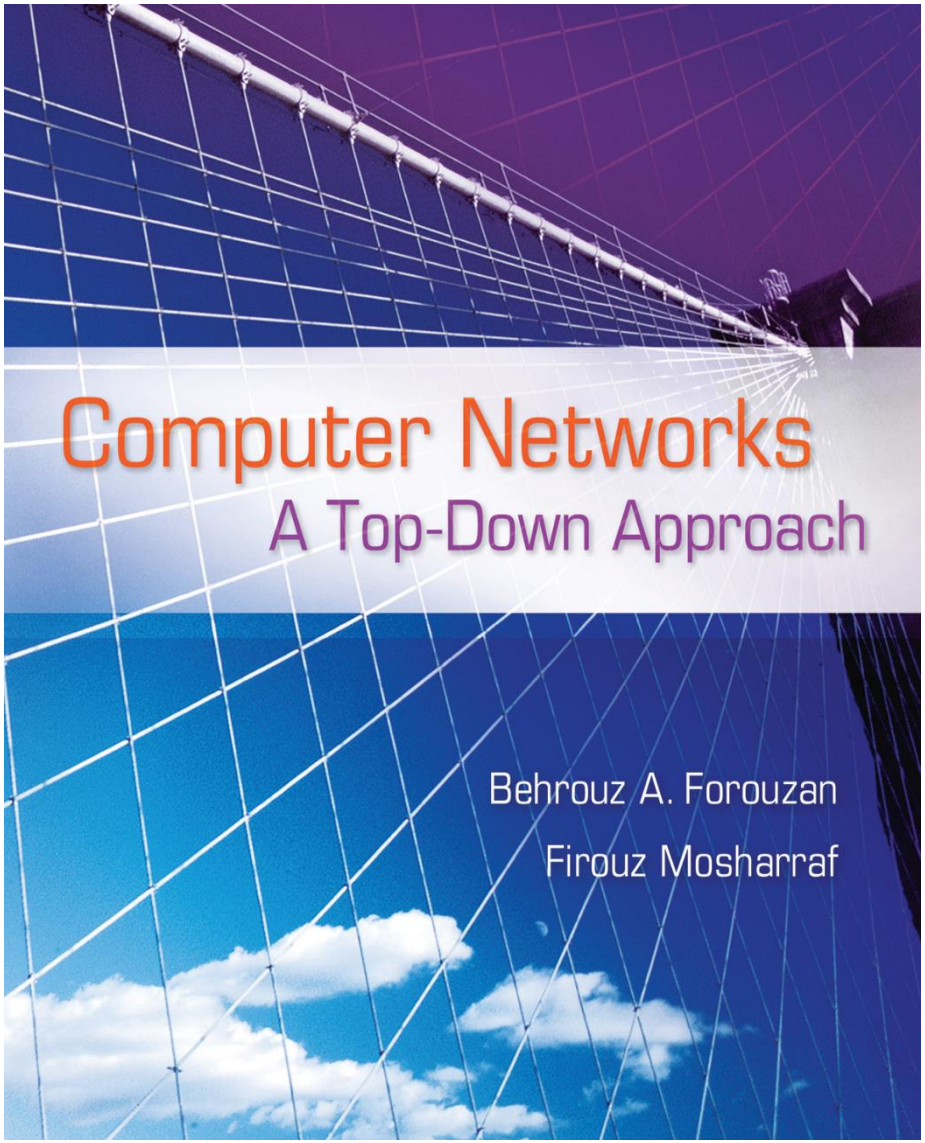
Chapter 5

Data-Link Layer



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Chapter 5: Outline

5.1 INTRODUCTION

5.2 DATA LINK CONTROL (DLC)

5.3 MULTIPLE ACCESS PROTOCOLS

5.4 LINK-LAYER ADDRESSING

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Chapter 5: Objective

- *We introduce the concept of nodes and links and the types of links, and show how the data-link layer is actually divided into two sublayers: data link control and media access control.*

- *We discuss data link control (DLC) of the data-link layer and explain services provided by this layer, such as framing, flow and error control, and error detection.*

- *We discuss the media access control (MAC) sublayer of the data-link layer. We explain different approaches such as random access, controlled access, and channelization.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5-1 INTRODUCTION

- *The Internet is a combination of networks glued together by connecting devices (routers or switches).*
- *If a datagram is to travel from a host to another host, it needs to pass through these networks.*
- *Figure 5.1 shows communication between Alice and Bob, using the same scenario we followed in the last three chapters.*

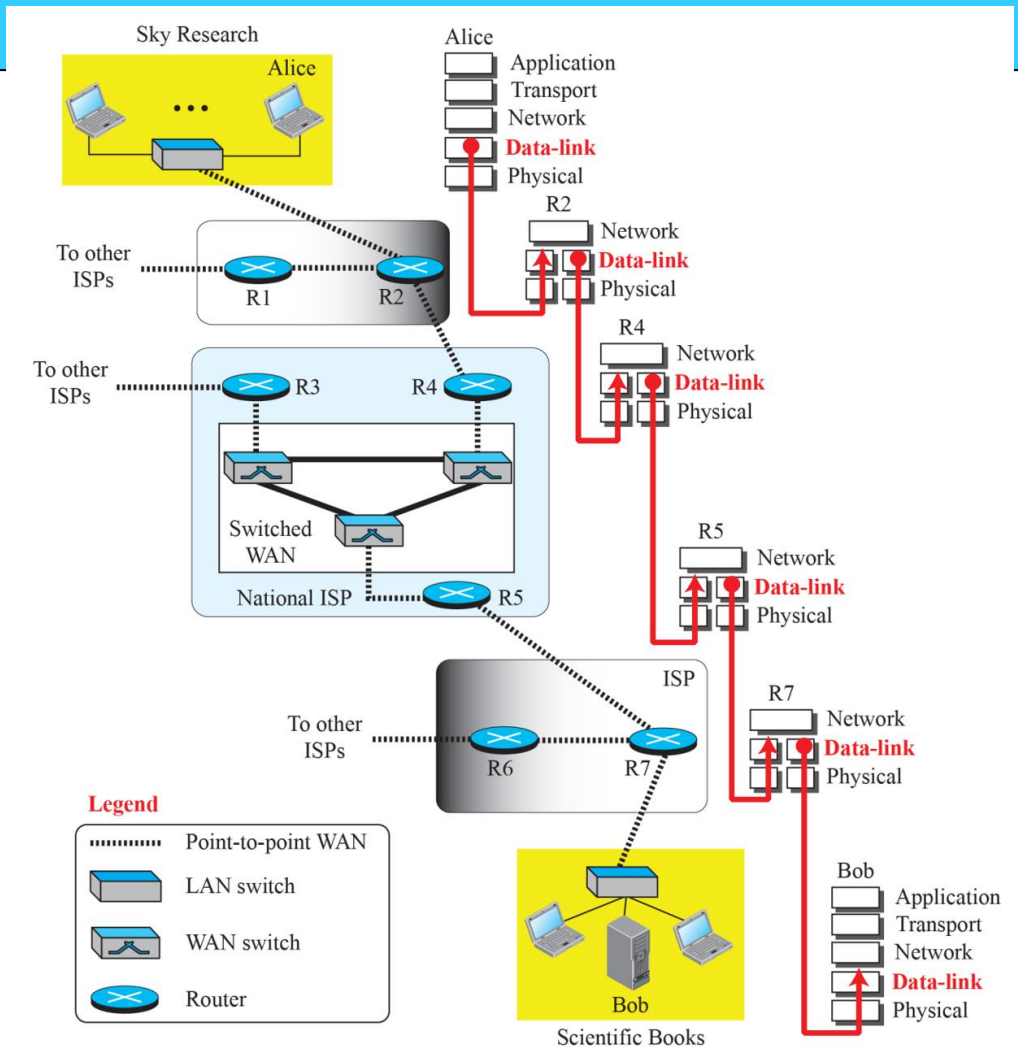


***Figure 5.1:* Communication at the data-link layer**

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5.1.1 Nodes and Links

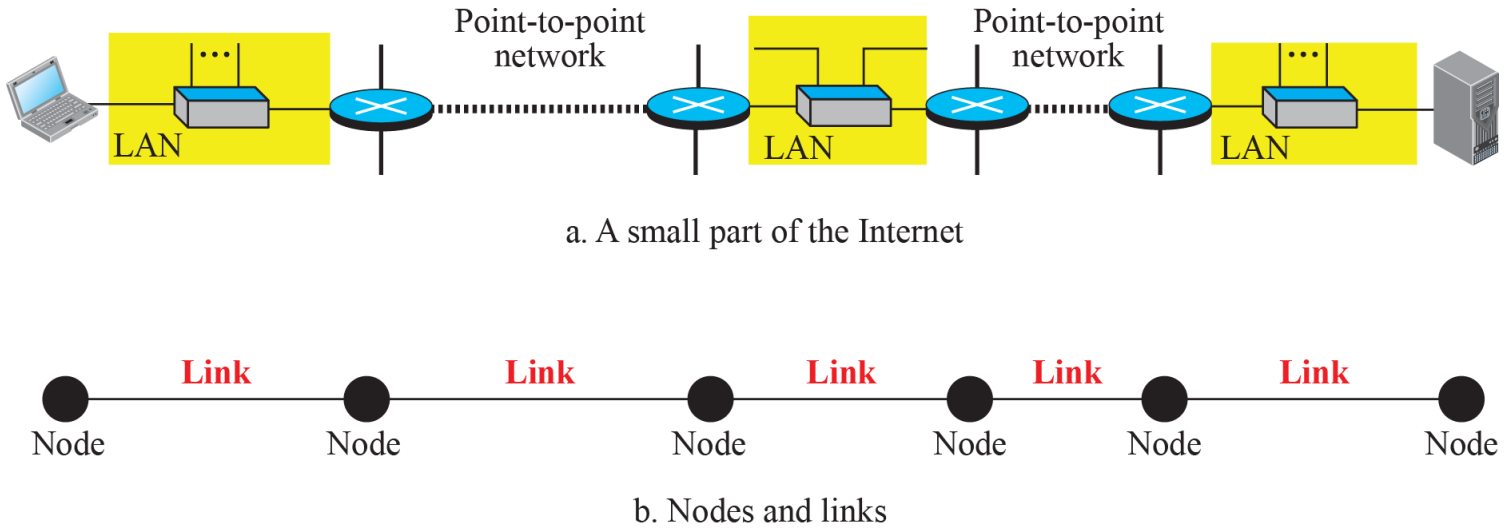


Figure 5.2: ***Nodes and Links***

- *Although communication at the application, transport, and network layers is end-to-end;*
- *Communication at the data-link layer is node-to node.*
- *As we have learned in the previous chapters, a data unit from one point in the Internet needs to pass through many networks (LANs and WANs) to reach another point.*
- *Theses LANs and WANs are connected by routers. It is customary to refer to the two end hosts and the routers as nodes and the networks in between as links.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5.1.2 Two Types of Links

- *Although two nodes are physically connected by a transmission medium such as cable or air, we need to remember that the data-link layer controls how the medium is used.*
- *We can have a data-link layer that uses the whole capacity of the medium; we can also have a data-link layer that uses only part of the capacity of the link. In other words, we can have a* ***point-to-point link*** *or* ***a broadcast link****.*

5.1.3 Two Sublayers

- *To better understand the functionality and services provided by the link layer, we can divide the data-link layer into* ***two sublayers****:*
- ***data link control (DLC)*** *and* ***media access control (MAC).***
- *The* ***data link control (DLC)*** *sublayer deals with all issues common to both point-to-point and broadcast links;*
- *the* ***media access control (MAC)*** *sublayer deals only with issues specific to broadcast links.*

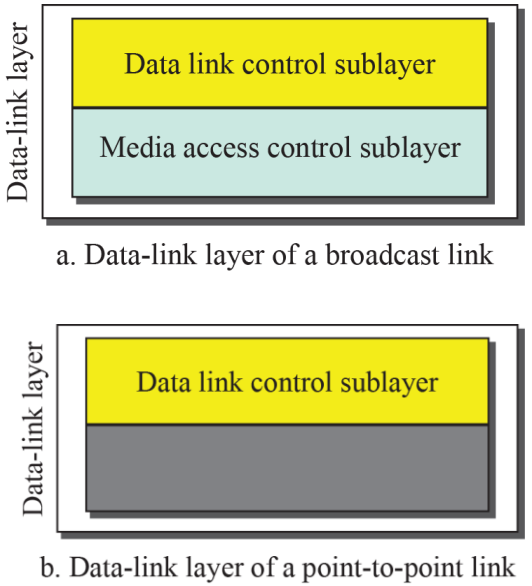


Figure 5.3: ***Data-link layer divided as two sublayers***

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5-2 DATA LINK CONTROL (DLC)

*The **data link control** deals*

- *with procedures for communication between two adjacent nodes in the network*
- ***functions** include:*
  - ***framing** - how to organize the bits that are carried by the physical layer.*
  - ***flow and error control** - Techniques for error detection are discussed at the end of this section.*
  - ***Error detection and correction***

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5.2.1 Framing

- *Data transmission in the physical layer means **moving bits in the form of a signal** from the source to the destination.*
- *The data-link layer, on the other hand, needs to pack bits into frames, so that each frame is distinguishable from another.*

❑ Frame Size

- ❖ ***Character-Oriented Framing***
- ❖ ***Bit-Oriented Framing***

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.4: A frame in a character-oriented protocol

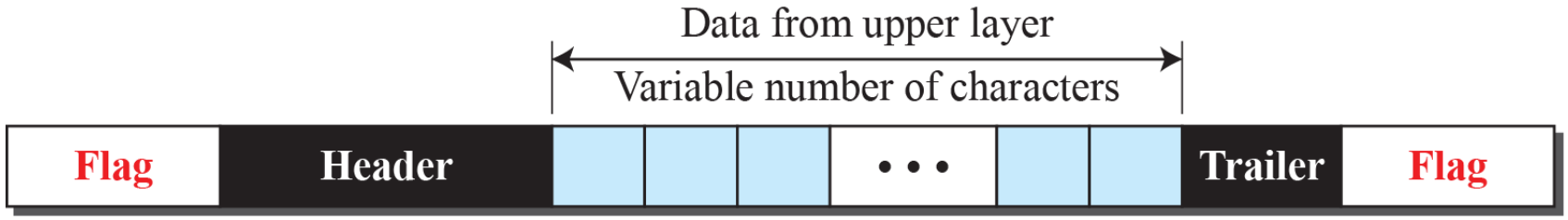
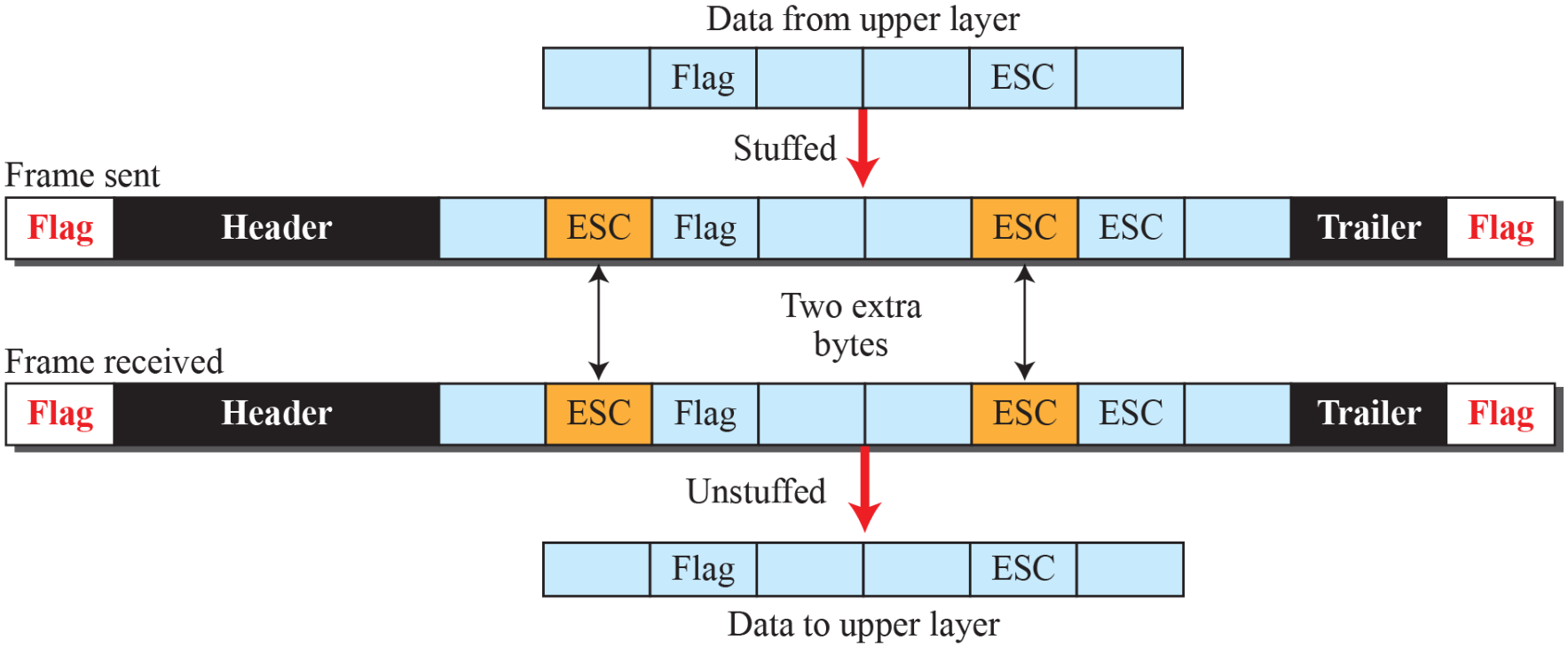


Figure 5.5: Byte stuffing and unstuffing



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.6: A frame in a bit-oriented protocol

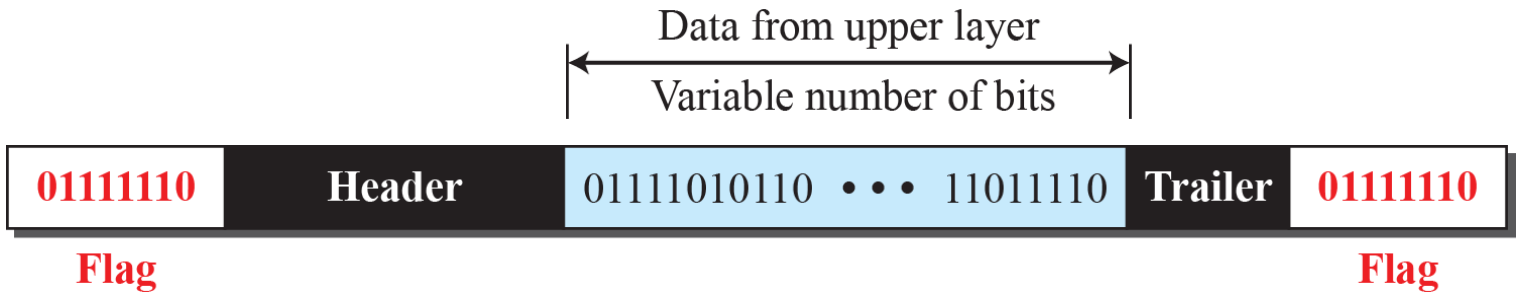
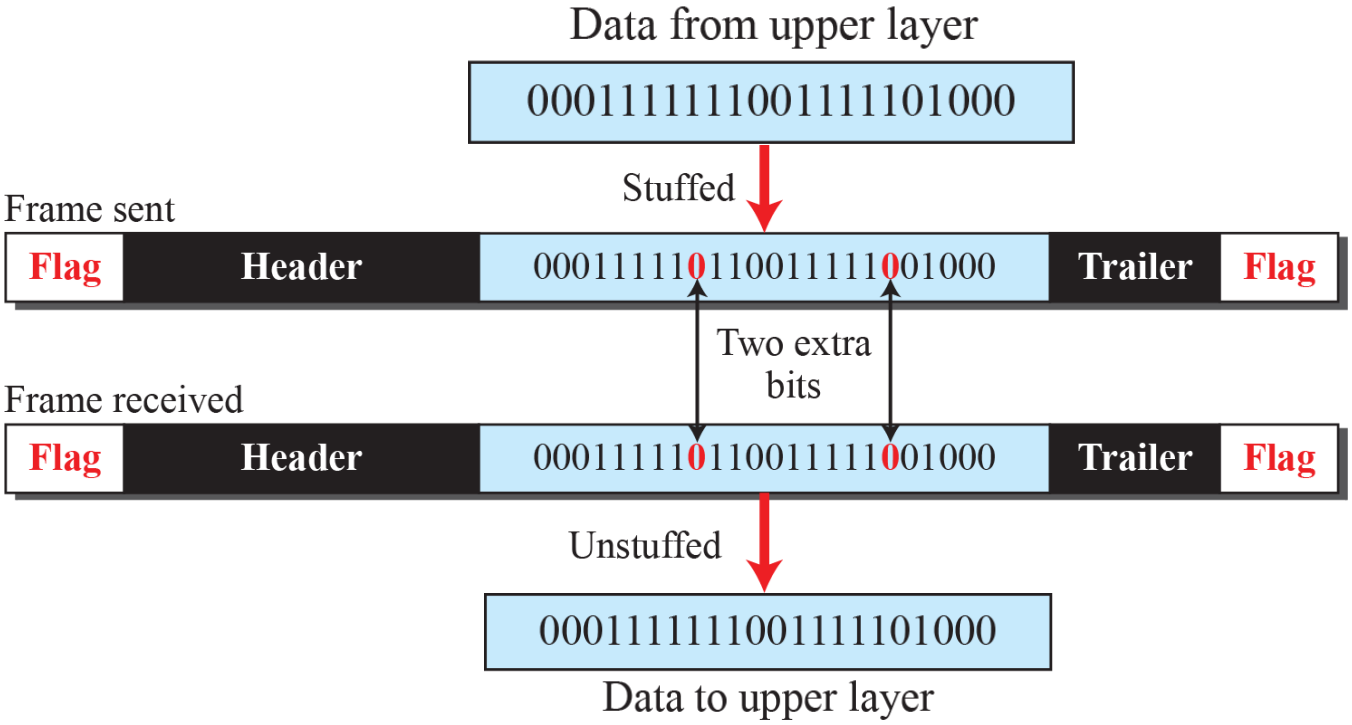


Figure 5.7: Bit stuffing and unstuffing



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5.2.2 Flow and Error Control

❑ Flow Control

- *We covered flow and error control in Transport TCP protocol in Lecture-3.*
- *One of the responsibilities of the data-link control sublayer is flow and error control at the data-link layer.*

5.2.3 Error Detection and Correction

❑ Error Control

- *At the data-link layer, if a frame is corrupted between the two nodes, it needs to be corrected before it continues its journey to other nodes.*
- *However, most link-layer protocols simply discard the frame and let the upper-layer protocols handle the retransmission of the frame.*
- *Some wireless protocols, however, try to correct the corrupted frame.*

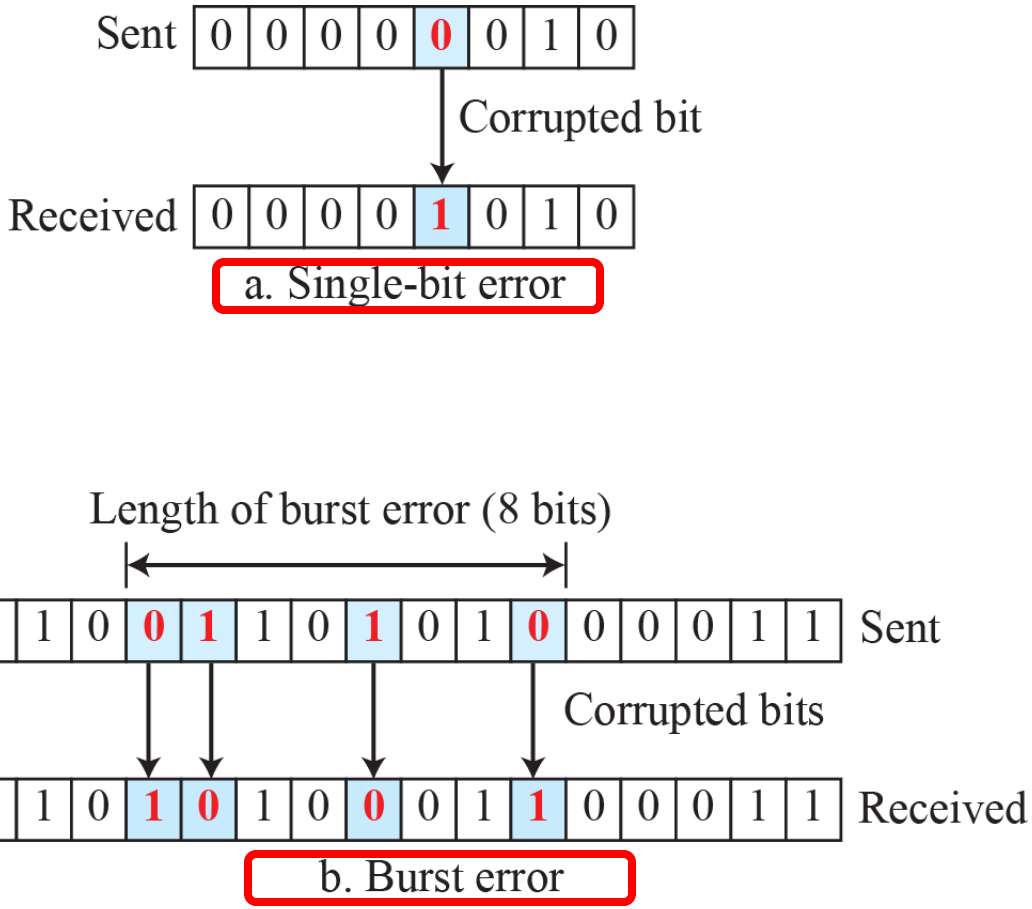
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5.2.3 Error Detection and Correction

- **Introduction**
  - Types of Errors
  - Redundancy
  - Detection versus Correction
  - Coding
- **Block Coding**
  - Error Detection
  - Hamming Distance
  - Minimum Hamming Distance for Error Detection
- **Linear Block Codes**
  - Parity-Check Code
- **Cyclic Codes**
  - Cyclic Redundancy Check
  - Polynomials
  - Requirement
  - Performance
  - Advantages of Cyclic Codes
- **Checksum**
  - Concept
  - Internet Checksum
  - Algorithm
  - Other Approaches to the Checksum

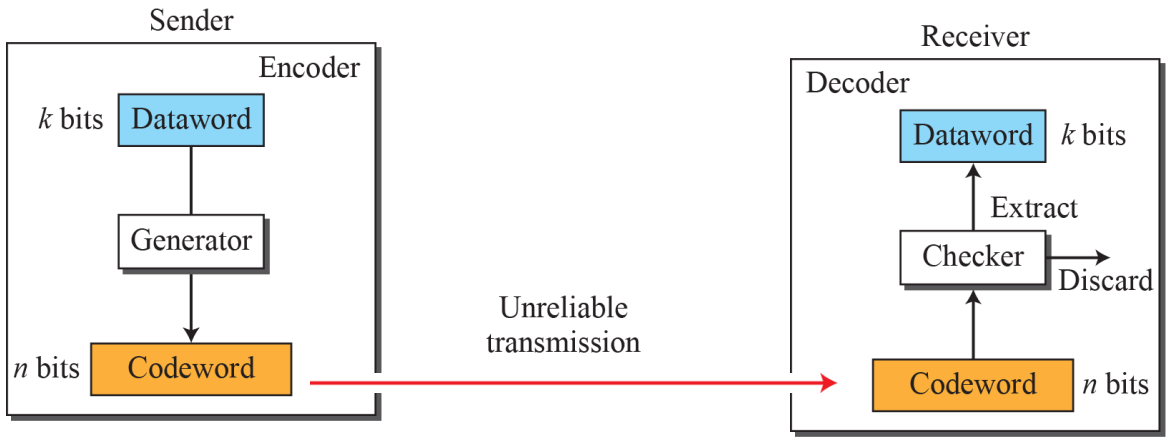
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.8: Bit Errors - Single-bit and burst error



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.9: Process of error detection in block coding



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Example 5.1

Let us assume that
DATAWORD = k bits =2;
CODEWORD = n bits = 3
Table 5.1 shows the list of ***datawords*** *and* ***codewords****.*
we will see how to derive a ***codeword*** *from a* ***dataword****.*

Table 5.1: A code for error detection in Example 5.1

Datawords	*Codewords*	*Datawords*	*Codewords*
00	000	10	101
01	011	11	110

Assume the sender encodes the dataword 01 as 011 and sends it to the receiver. Consider the following cases:

1. The receiver receives 011. It is a valid codeword. The receiver extracts the dataword 01 from it.
2. The codeword is corrupted during transmission, and 111 is received (the leftmost bit is corrupted). This is not a valid codeword and is discarded.
3. The codeword is corrupted during transmission, and 000 is received (the right two bits are corrupted). This is a valid codeword. The receiver incorrectly extracts the dataword 00. Two corrupted bits have made the error undetectable.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Example 5.2

Hamming Distance: *In information theory, the Hamming distance between two strings of equal length is the number of positions at which the corresponding symbols are different. In other words, it measures the minimum number of substitutions required to change one string into the other, or the minimum number of errors that could have transformed one string into the other.*

1. The Hamming distance $d(000, 011)$ is 2 because $(000 \oplus 011)$ is 011 (two 1s).
2. The Hamming distance $d(10101, 11110)$ is 3 because $(10101 \oplus 11110)$ is 01011 (three 1s).

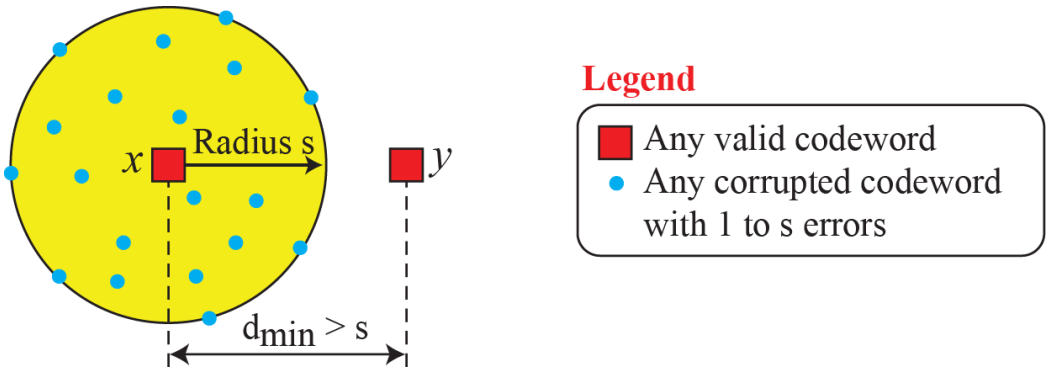


Figure 5.10: ***Geometric concept explaining d_{min} in error detection***

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Datawords	Codewords	Datawords	Codewords
00	000	10	101
01	011	11	110

Example 5.3

- *The minimum Hamming distance for our first code scheme (Table 5.1) is* ***dmin=2.***
- *This code guarantees* ***detection of only a single error.***
- *For example, if the third codeword (101) is sent and one error occurs, the received codeword does not match any valid codeword.*
- *If two errors occur, however, the received codeword may match a valid codeword and the errors are not detected.*
- *(dmin= d = s + 1 or s= 1).*

Example 5.4

- *A code scheme has a Hamming distance* ***dmin = 4****.*
- *This code guarantees the detection of up to three errors*
- *(dmin= d = s + 1 or s= 3).*

Example 5.5

- *The code in Table 5.1 is a linear block code because the result of XORing any codeword with any other codeword is a valid codeword.*
- *For example, the XORing of the second and third codewords creates the fourth one.*

Example 5.6

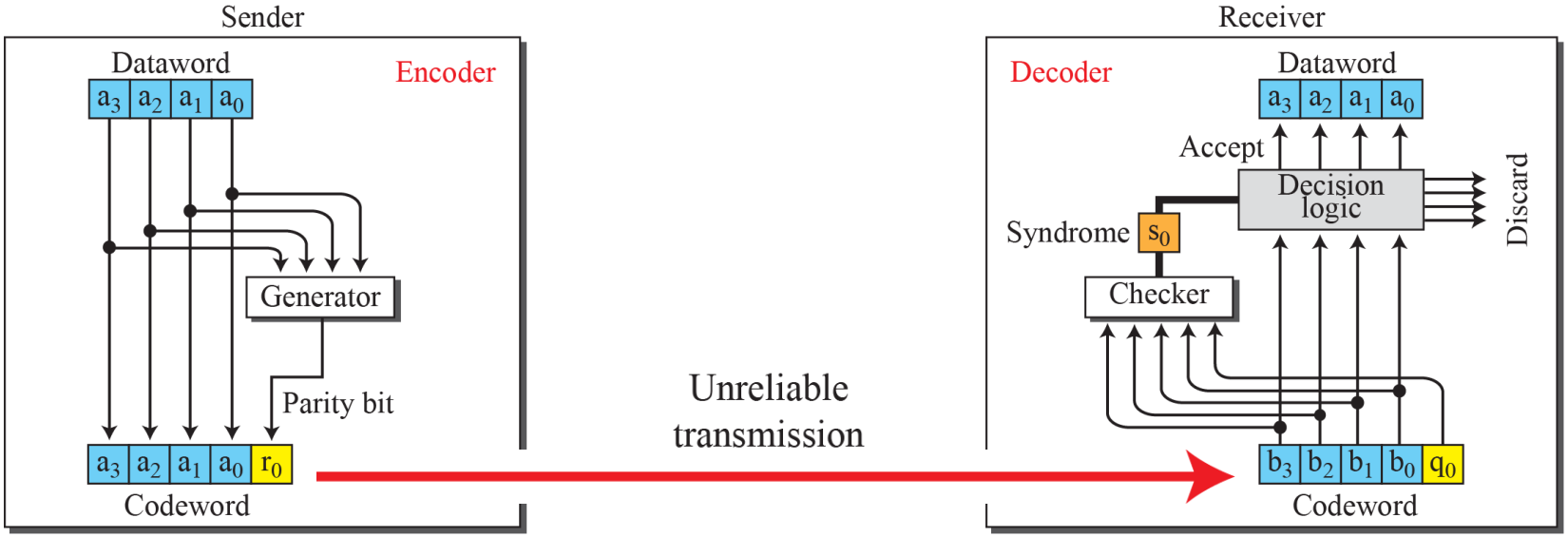
- *In our first code (Table 5.1), the numbers of 1s in the nonzero codewords are 2, 2, and 2. So the minimum Hamming distance is* $\mathbf{d}_{min} = 2$.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Table 5.2: Simple EVEN parity-check code C(5, 4)

Datawords	Codewords	*Datawords*	Codewords
0000	00000	1000	10001
0001	00011	1001	10010
0010	00101	1010	10100
0011	00110	1011	10111
0100	01001	1100	11000
0101	01010	1101	11011
0110	01100	1110	11101
0111	01111	1111	11110

Figure 5.11: Encoder and decoder for simple EVEN parity-check code



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Example 5.7

*Let us look at some transmission scenarios. Assume the sender sends the dataword 1011. The codeword created from this dataword is 10111, which is sent to the receiver. We examine **five** cases of **EVEN** parity:*

1. No error occurs; the received codeword is 10111. The syndrome is 0. The dataword 1011 is created.
2. One single-bit error changes a_1. The received codeword is 10**0**11. The syndrome is 1. No dataword is created.
3. One single-bit error changes r_0. The received codeword is 1011**0**. The syndrome is 1. No dataword is created. Note that although none of the dataword bits are corrupted, no dataword is created because the code is not sophisticated enough to show the position of the corrupted bit.
4. An error changes r_0 and a second error changes a_3. The received codeword is **0**011**0**. The syndrome is 0. The dataword 0011 is created at the receiver. Note that here the dataword is wrongly created due to the syndrome value. The simple parity-check decoder cannot detect an even number of errors. The errors cancel each other out and give the syndrome a value of 0.
5. Three bits—a_3, a_2, and a_1—are changed by errors. The received codeword is **010**11. The syndrome is 1. The dataword is not created. This shows that the simple parity check, guaranteed to detect one single error, can also find any odd number of errors.

- *Both **even** parity and **odd** parity can only **detected odd number of inversions** in the dataword.*
- *It cannot detect even number of inversions in the dataword.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Table 5.3: **A Cyclic Redundancy Check (CRC) code with C(codeword=7, Dataword=4)**

Dataword	Codeword	Dataword	Codeword
0000	0000000	1000	1000101
0001	0001011	1001	1001110
0010	0010110	1010	1010011
0011	0011101	1011	1011000
0100	0100111	1100	1100010
0101	0101100	1101	1101001
0110	0110001	1110	1110100
0111	0111010	1111	1111111

*The **cyclic redundancy check (CRC)** is a technique used to detect errors in digital data. CRC is a hash function that detects accidental changes to raw computer data commonly used in digital telecommunications networks and storage devices such as hard disk drives. This technique was invented by W. Wesley Peterson.*

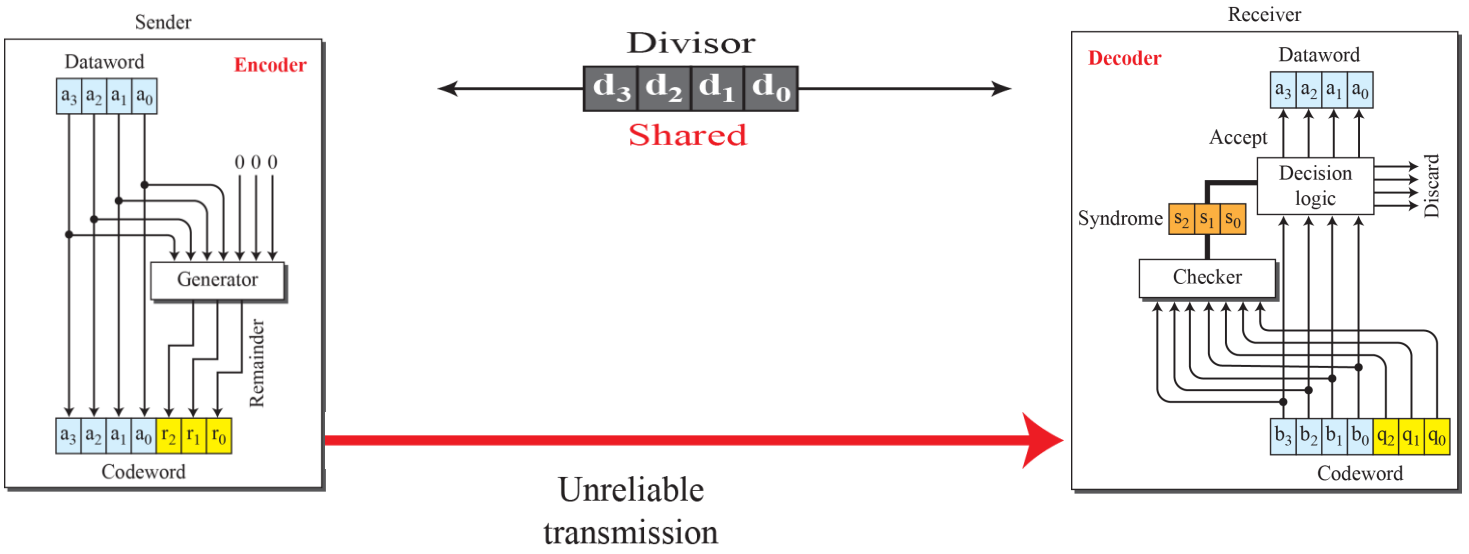
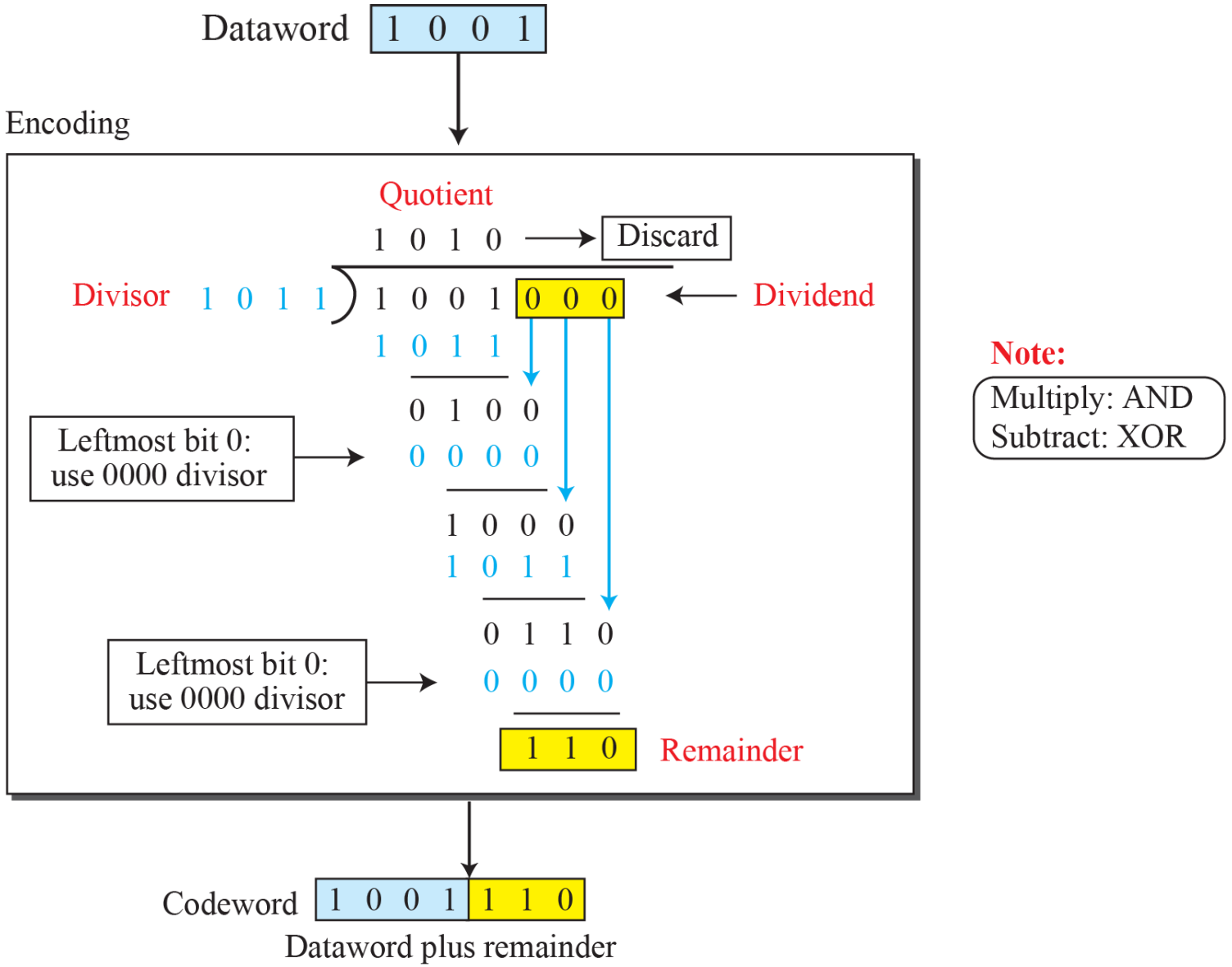


***Figure 5.12:** CRC encoder and decoder*

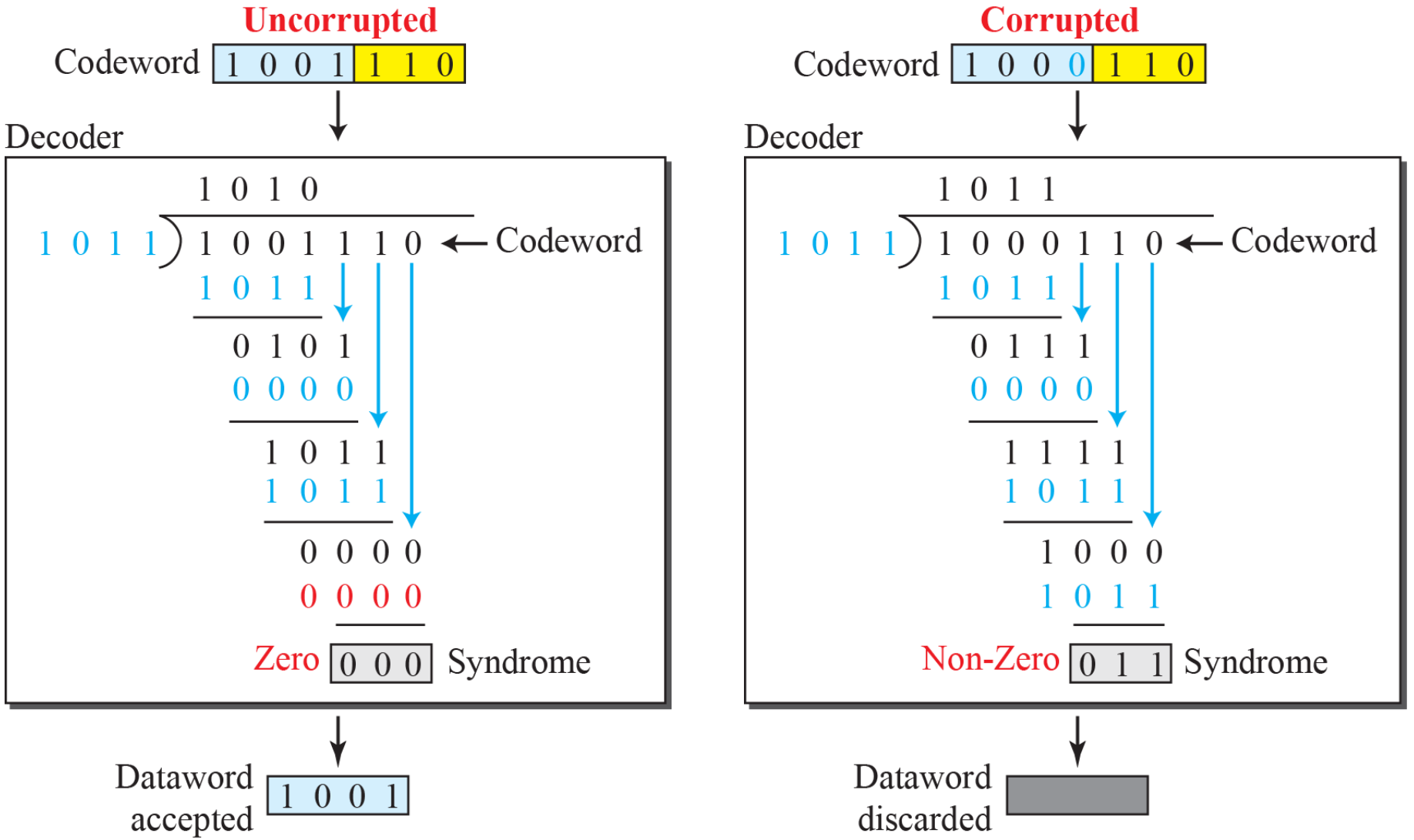
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.13: Division in CRC encoder



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.14: Division in the CRC decoder for two cases



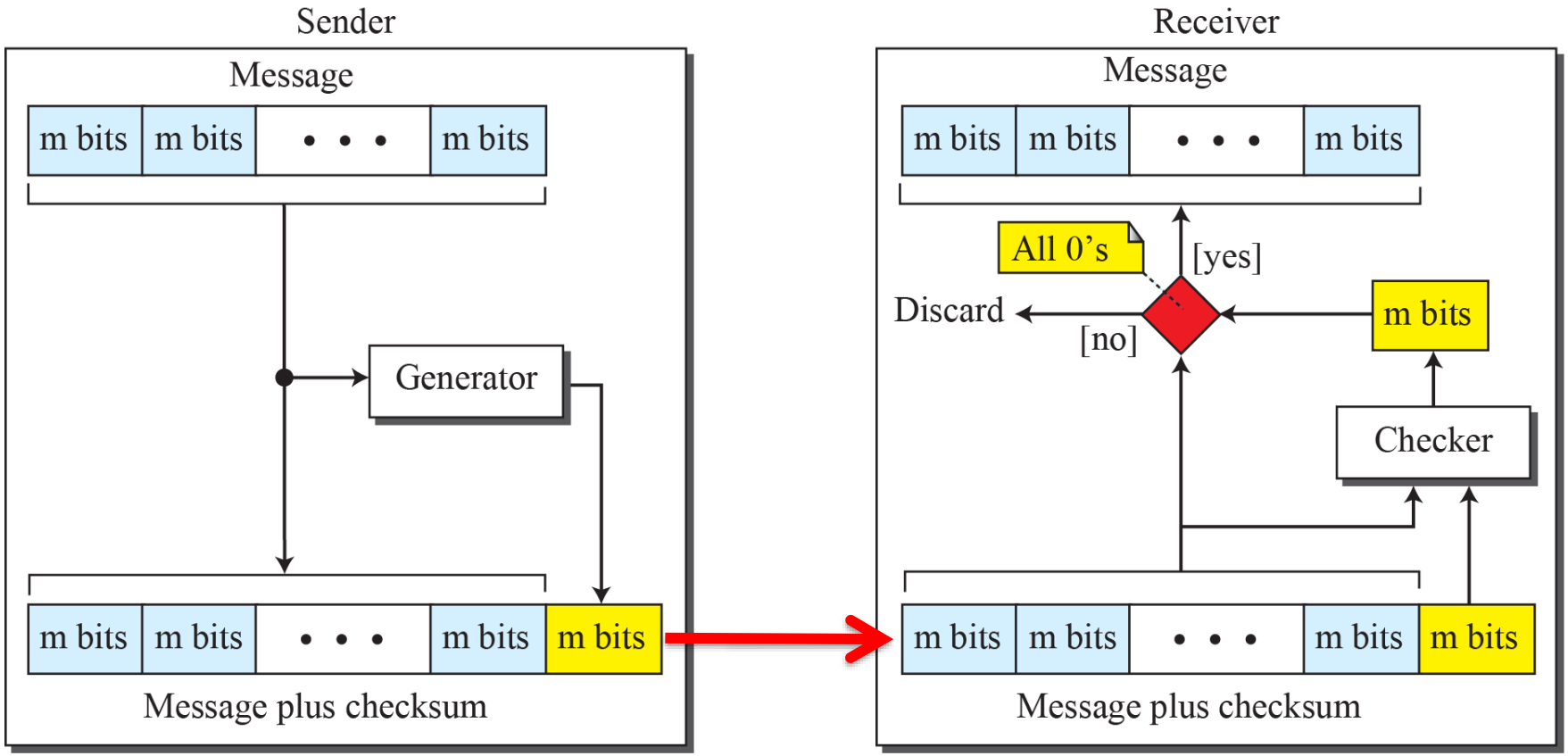
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Table 5.4: Standard polynomials

Name	*Binary*	*Application*
CRC-8	100000111	ATM header
CRC-10	11000110101	ATM AAL
CRC-16	1000100000100001	HDLC
CRC-32	100000100110000010001110110110111	LANs

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.15: Checksum



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Example 5.8

- *Suppose the message is a **list of five 4-bit numbers** that we want to send to a destination.*
- *In addition to sending these numbers, we send the sum of the numbers.*
- *For example, if the set of numbers is (7, 11, 12, 0, 6), we send (7, 11, 12, 0, 6, **36**), where **36** is the sum of the original numbers.*
- ***Receiver adds the five numbers and compares the result with the sum.***
- *If the **two are the same**, the receiver **assumes no error**, accepts the five numbers, and discards the sum. Otherwise, there is an error somewhere and the message not accepted.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Example 5.9

In the previous example, the decimal number **36 in binary is $(100100)_2$.**
To change it to a 4-bit number we add the extra leftmost bit to the right four bits as shown below.

$$(10)_2 + (0100)_2 = (0110)_2 \rightarrow (6)_{10}$$

- *Instead of sending **36 as the sum**,*
- *we can send **6 as the sum** (7, 11, 12, 0, 6, **6**).*
- *The receiver can add the first five numbers in one's complement arithmetic*
- *If the result is 6, the numbers are accepted; otherwise, they are rejected.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

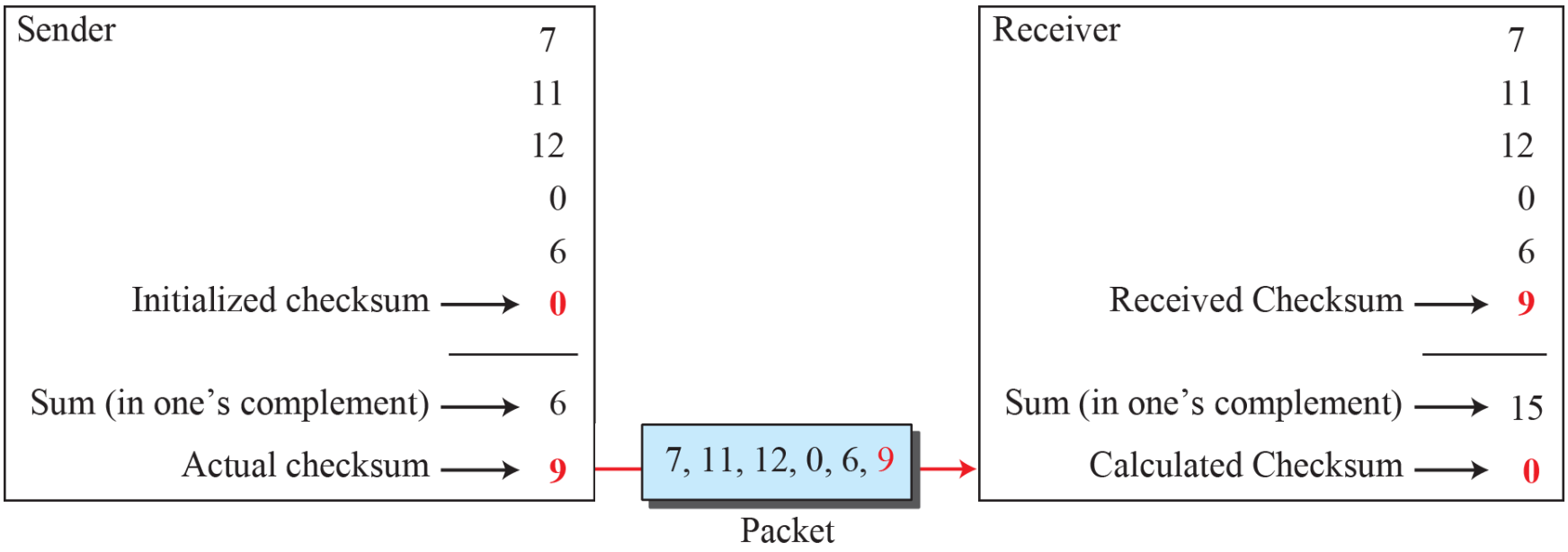
Example 5.10

Let us use the idea of the checksum in Example 5.9.

- *The sender **adds all five numbers in one's complement** to get the sum = 6.*
- *The sender then complements the result to get the checksum = **9**, which is 15 − 6.*
- ***Note that 6 = $(0110)_2$ and 9= $(1001)_2$; they are complements of each other.***
- *The sender sends the five data numbers and the checksum (7, 11, 12, 0, 6, **9**).*
- *If there is no corruption in transmission, the receiver receives (7, 11, 12, 0, 6, **9**) and adds them in one's complement to get 15.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.16: Example 5.10



(7, 11, 12, 0, 6, 0)	=36
	(100100)
	10 + 0100 =0110
	Complement of 0110
	1001 ==9
(7, 11, 12, 0, 6, 9)	

(7, 11, 12, 0, 6, 9)	=45
	(101101)
	10 + 1101 =1111
	Complement of 1111
	0000 ==0
(7, 11, 12, 0, 6, 0)	

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Table 5.5: Procedure to calculate the traditional checksum

Sender	*Receiver*
1. The message is divided into 16-bit words.	1. The message and the checksum is received.
2. The value of the checksum word is initially set to zero.	2. The message is divided into 16-bit words.
3. All words including the checksum are added using one's complement addition.	3. All words are added using one's complement addition.
4. The sum is complemented and becomes the checksum.	4. The sum is complemented and becomes the new checksum.
5. The checksum is sent with the data.	5. If the value of the checksum is 0, the message is accepted; otherwise, it is rejected.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5.2.4 Two DLC Protocols

- *After finishing all issues related to the DLC sublayer, we discuss two DLC protocols that actually implemented those concepts.*
- *The first, LLC, is the base of many protocols that have been designed for LANs.*
- *The second, Point to-Point, is a protocol derived from HDLC and is used for point-to-point links.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

5.2.4 (continued)

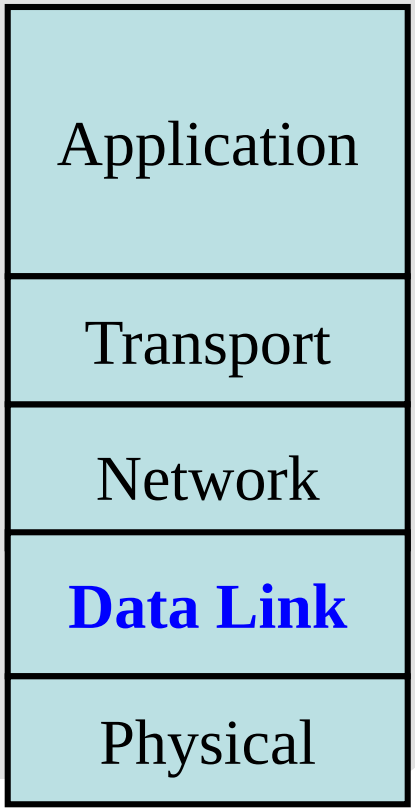
- **HDLC**
  - *Configuration and Transfer Modes*
  - *Frames*
- **Point-to-Point Protocol (PPP)**
  - *Services*
  - *Framing*
  - *Transition Phases*
  - *Multiplexing*
  - *Multilink PPP*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

MONASH University
Information Technology

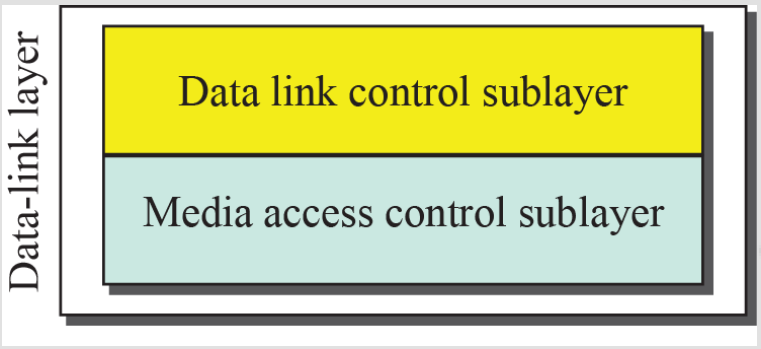
Data Link Control Protocols

Reference: Chapter 7
Data and Computer Communications
Eighth Edition
by William Stallings
Lecture slides by Lawrie Brown



Data Link Control Protocols

- **need layer of logic above Physical**
- **to manage exchange of data over a link**
  - frame synchronization
  - flow control
  - error control
  - frame addressing
  - control and data on same link
  - link management



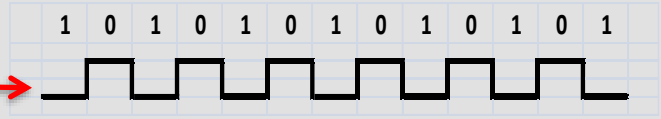
SUB-1 LOGICAL LINK CONTROL (LLC)

- ✓Frames the network layer packet
- ✓Identifies the network layer protocol
- ✓Provides flow control
- ✓Provides error control
- ✓Link management

SUB-2 MEDIA ACCESS CONTROL (MAC)

- ✓Frame addressing (synchronisation)
- ✓Marks the beginning and ending of the frame

The frame is encoded into a signal as **BITS (physical layer 1)**

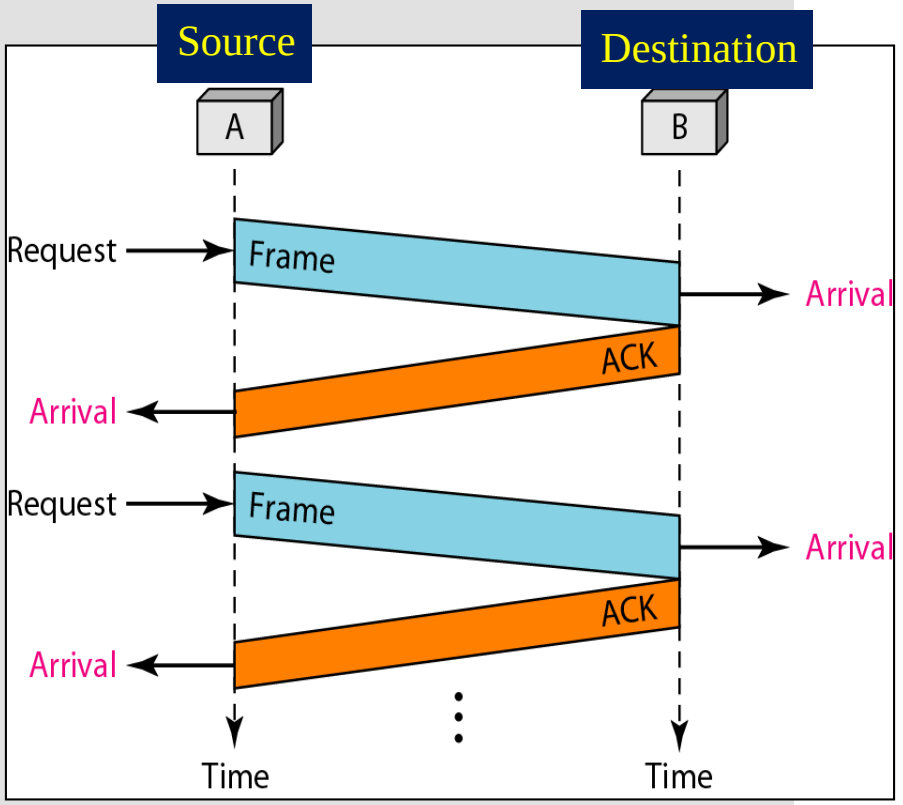


Flow Control

- **ensure sending entity does not overwhelm receiving entity**
  - by preventing buffer overflow
- **influenced by:**
  - transmission time
    - > time taken to emit all bits of a frame onto the medium
  - propagation time
    - > time for a bit to traverse the link between source and destination.
- **assume here no errors but different delays**
  - Point-to-Point links – fixed
  - Circuit switched, ATM networks - variable

Stop and Wait

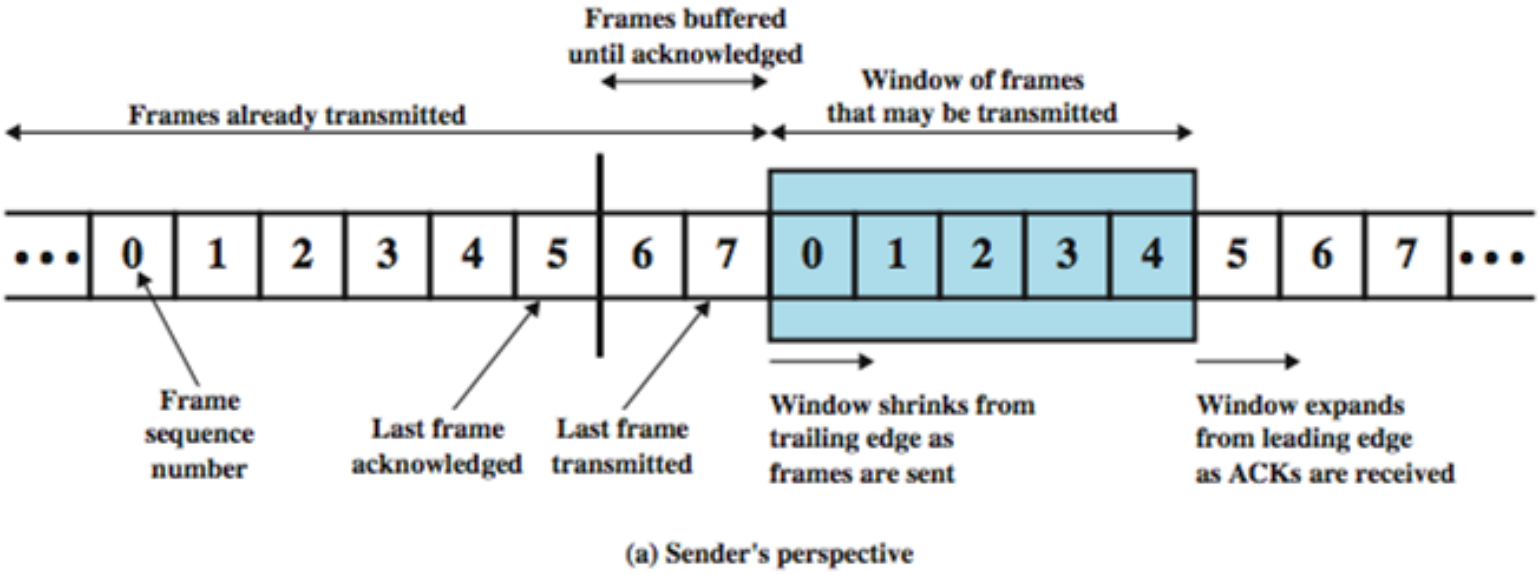
1. source transmits frame
2. destination receives frame and replies with acknowledgement (ACK)
3. source waits for ACK before sending next frame
4. destination can stop the flow by not sending ACK
5. works well for a few large frames
6. Stop and wait becomes inadequate if large block of data is split into small frames



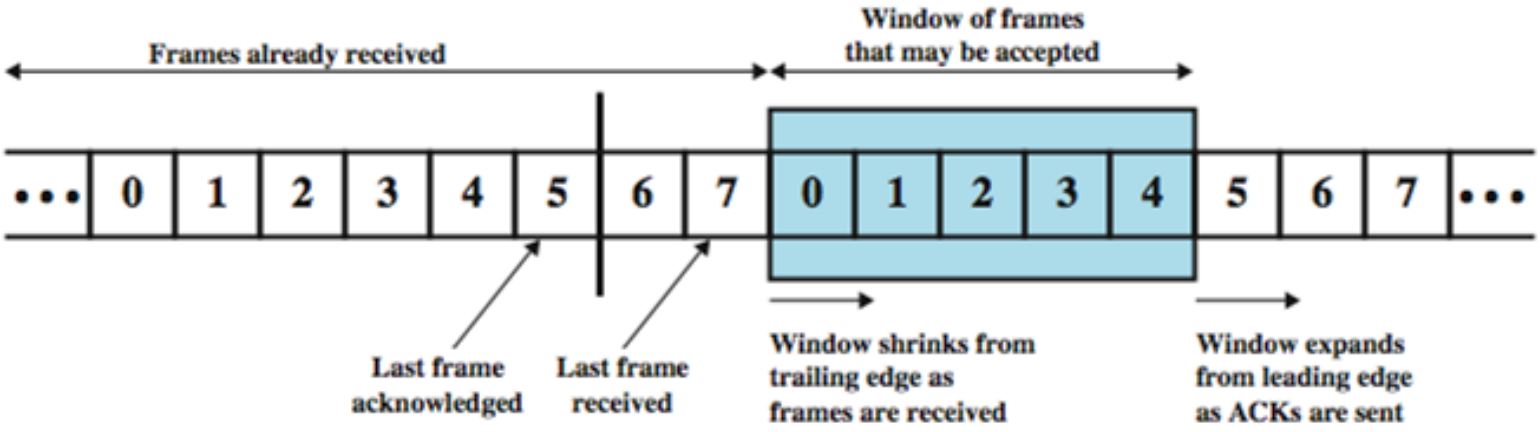
Sliding Windows Flow Control

- allows multiple numbered frames to be in transit
- receiver should have buffer space for W frames
- transmitter sends up to W frames without ACK
- ACK includes number of next frame expected
- sequence number is bounded by size of field (k)
  - frames are numbered modulo 2^k
  - giving max window size W of up to $2^k - 1$
- receiver can stop further transmission with ACK RNR (Receive Not Ready)
- To resume it must send a normal acknowledge to resume
- For full-duplex link, ACK's can piggyback on frames

Sliding Window Diagram

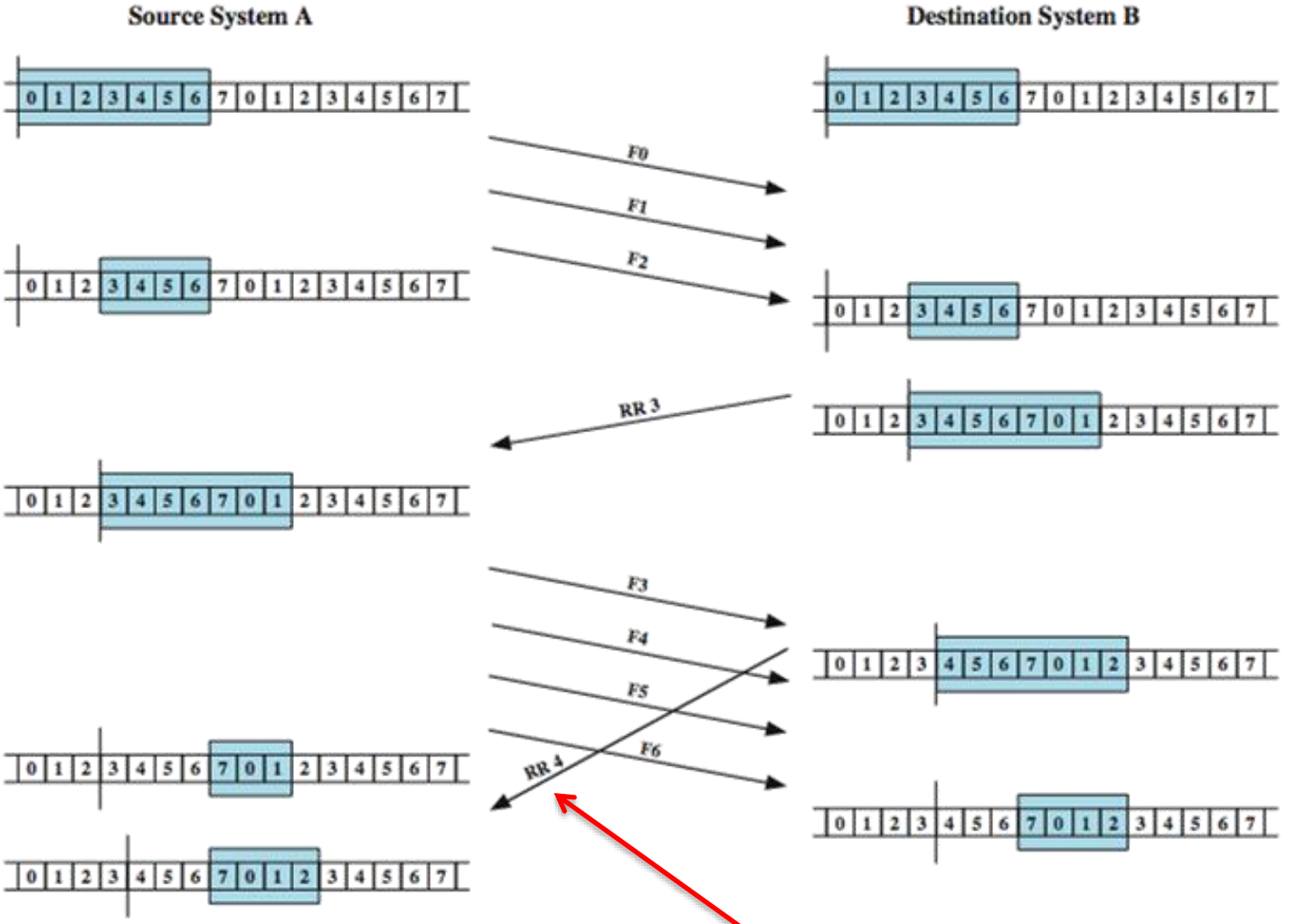


(a) Sender's perspective



(b) Receiver's perspective

Sliding Window Example



RR4 (receive ready for frame 4)

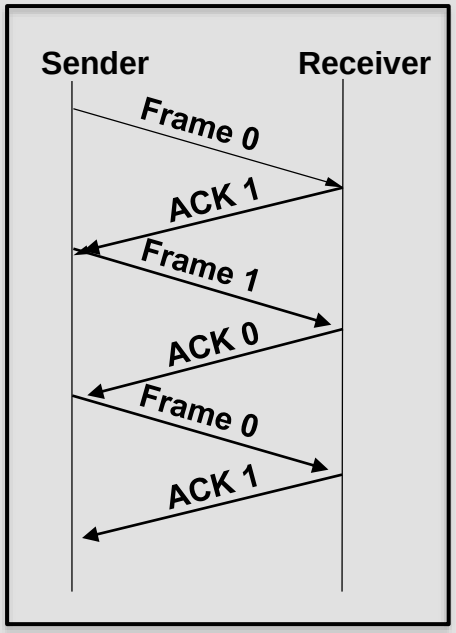
Error Control

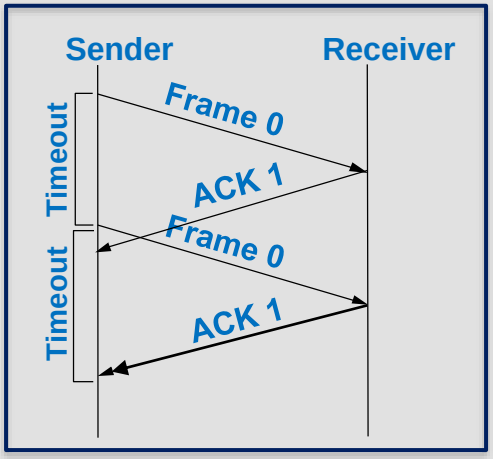
- **detection and correction of errors such as:**
  - lost frames
  - damaged frames
- **common techniques for error control is by using:**
  - error **detection**
  - **positive** acknowledgment
  - **retransmission** after timeout
  - **negative** acknowledgement & retransmission

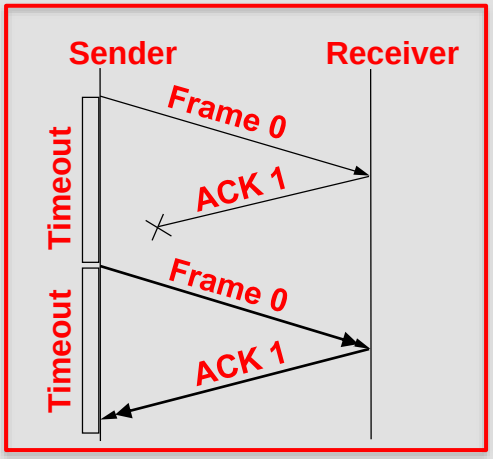
Automatic Repeat Request (ARQ)

- collective names for such error control mechanisms are referred to as ARQ:
- The effect of ARQ is to convert unreliable data link → into an reliable one
- Three versions of ARQ methods:
  - **Stop-and-wait ARQ**
  - **Go-back-N ARQ**
  - **Selective-reject ARQ** (selective retransmission)

Stop and Wait

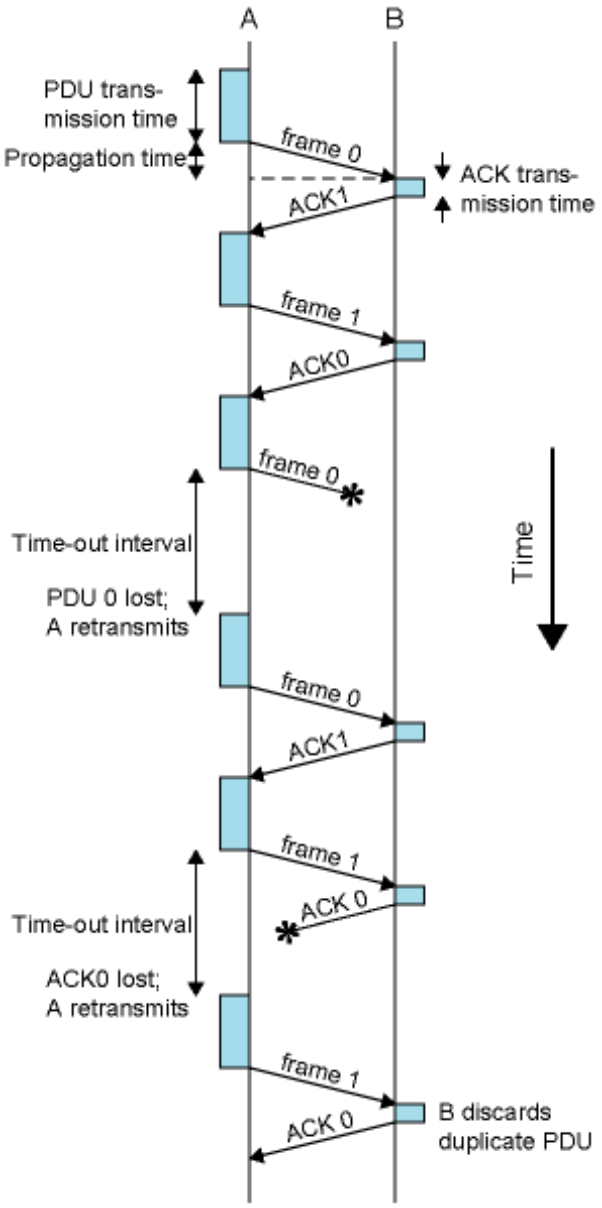






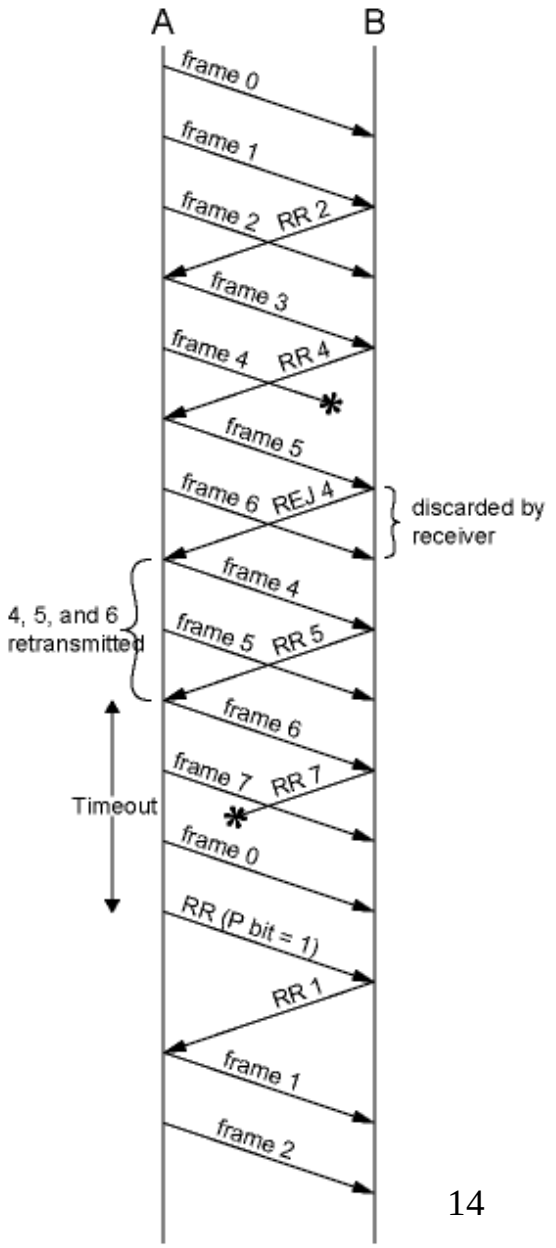
Stop and Wait

- **example with both types of errors**
- **pros and cons**
  - simple
  - inefficient



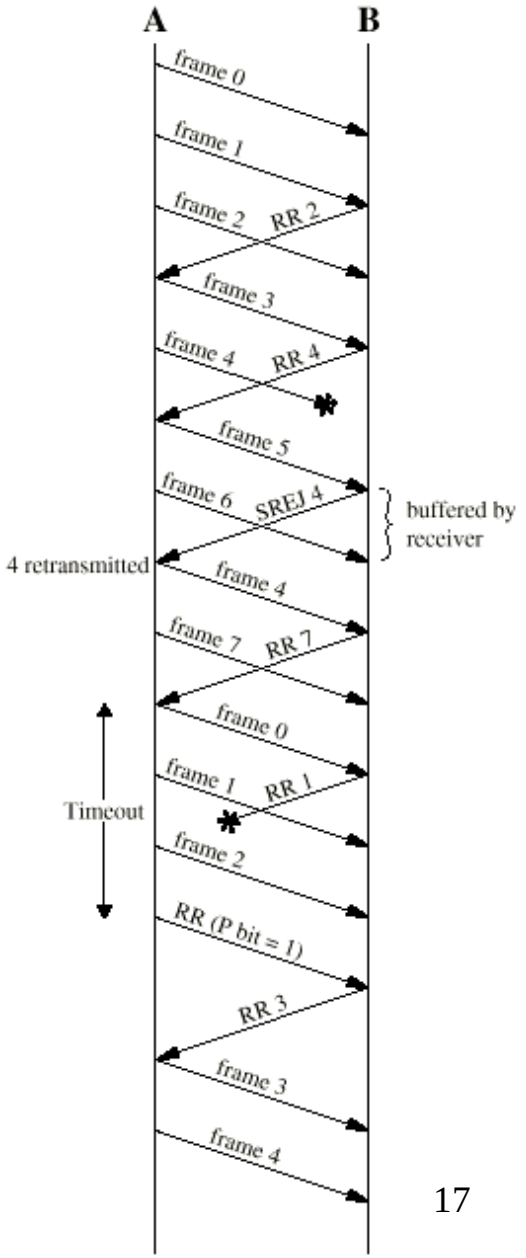
Go Back N

- **based on sliding window**
- **if no error, ACK as usual**
- **use window to control number of outstanding frames**
- **if error in frame-N, reply with rejection REJ (-ve ACK)**
  - discard that frame and all future frames until error frame received correctly
  - transmitter must go back to **N** and retransmit that frame and all subsequent frames
- **The go-back-N technique accounts for:**
  - ✓ Damaged frame.
  - ✓ Lost Frame
  - ✓ Damaged ACK
  - ✓ Damaged Reject



Selective Reject (SREJ)

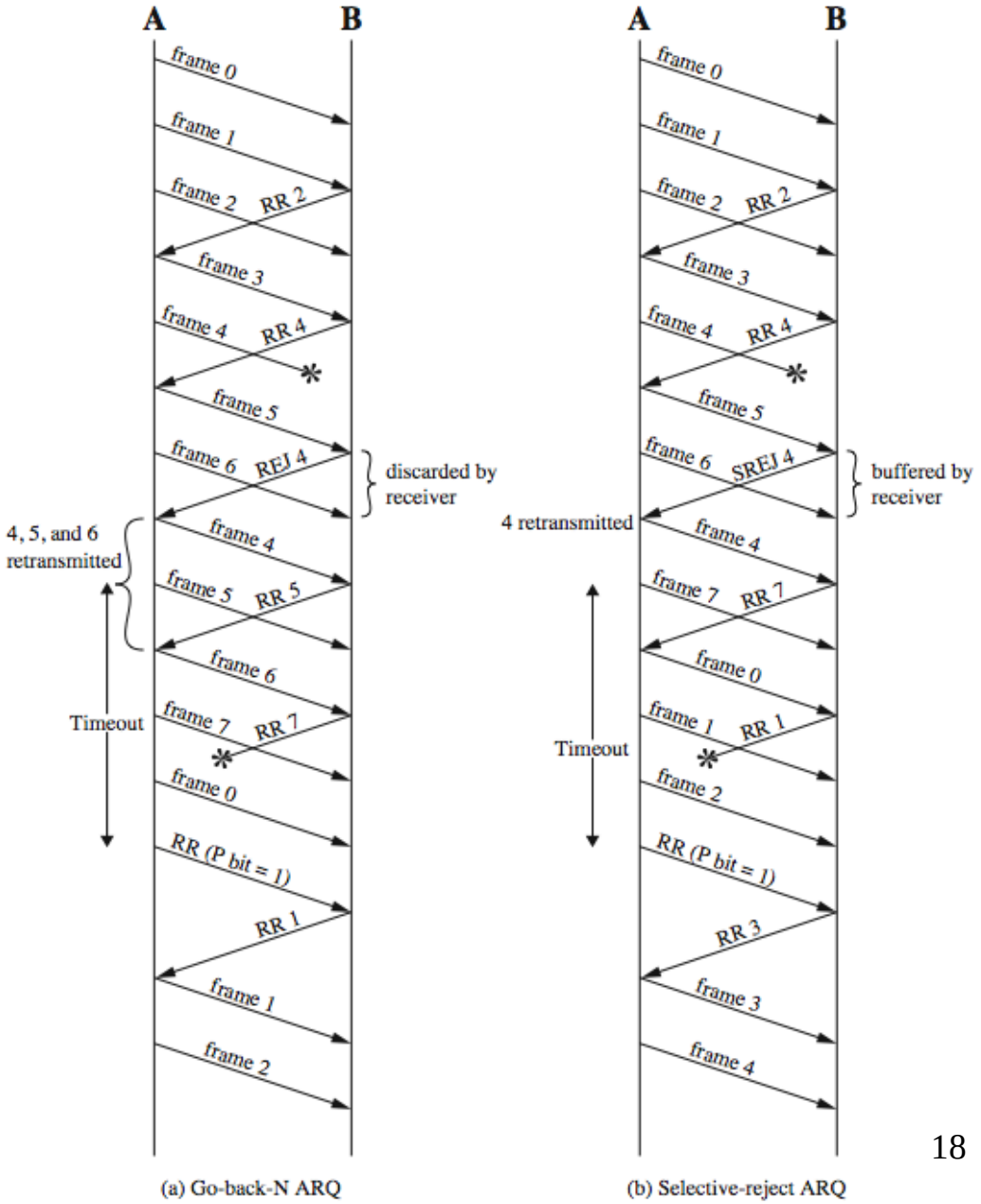
- also called selective retransmission
- only rejected frames are retransmitted
- subsequent frames are accepted by the receiver and buffered
- minimizes retransmission
- receiver must maintain large enough buffer
- more complex logic in transmitter
- hence less widely used
- useful for satellite links with long propagation delays



Go Back N

vs.

Selective Reject

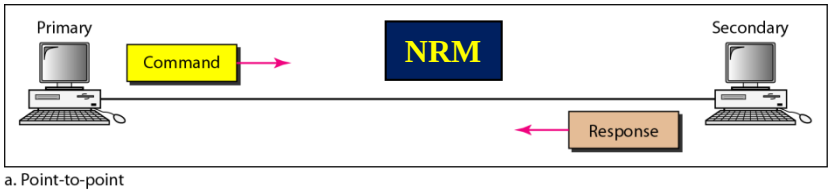


High Level Data Link Control (HDLC)

- **an important data link control protocol, specified as ISO 3009, ISO 4335**
- **Station 3 types:**
  - Primary - controls operation of link (One-Pri, many-Secondary)- **commands**
  - Secondary - under control of primary station - **responses**
  - Combined - combined stations issues **commands and responses**
- **Link configurations 2 types:**
  - Unbalanced - **1** primary, **multiple** secondary (unbalanced because primary controls secondary stations) (support – full and half duplex)
  - Balanced - 2 combined stations(primary & Secondary) (support – full and half duplex)
- **Data transfer modes 3 types:**
  - Normal Response Mode (NRM)
  - **Asynchronous Balanced Mode (ABM) – most widely used!**
  - Asynchronous Response Mode (ARM)

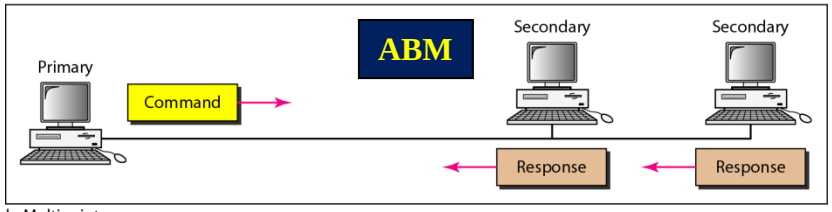
HDLC Transfer Modes

- **Normal Response Mode (NRM)**
  - Unbalanced config, primary initiates transfer
  - Secondary sends only when permitted by primary
  - No communication between secondaries
  - Typically used in multipoint lines, e.g. host + terminals



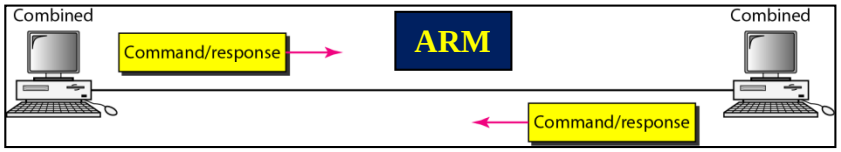
a. Point-to-point

- **Asynchronous Balanced Mode (ABM)**
  - balanced config, either station initiates transmission, has no polling overhead, Most widely used, requires combined stations, **Best mode for point-to-point lines.**



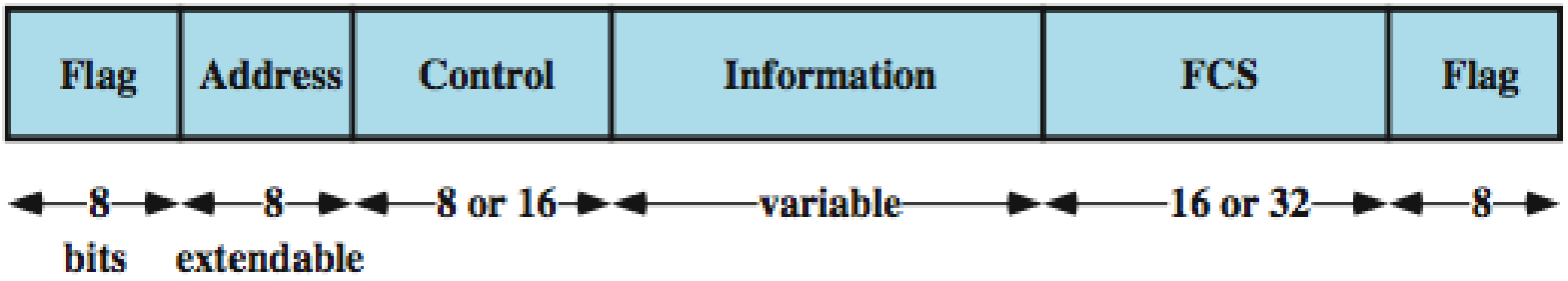
b. Multipoint

- **Asynchronous Response Mode (ARM)**
  - unbalanced config, secondary may initiate and transmit without permission from primary, rarely used
  - Primary is responsibility for the line, initialization, error recovery, and logical disconnection



HDLC Frame Structure

- synchronous transmission of frames
- single frame format used for both data and control exchange

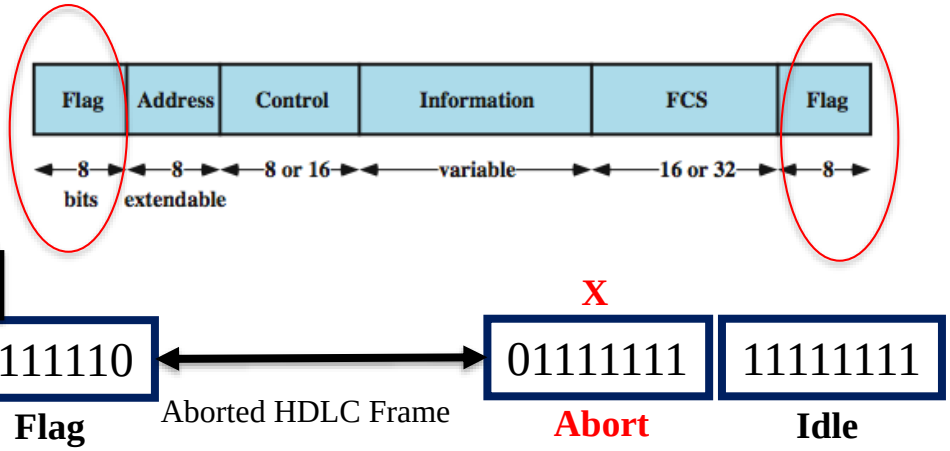


(a) Frame format

Flag Fields and Bit Stuffing

- **delimit frame at both ends with 01111110 sequence**
- **receiver hunts for flag sequence to synchronize**
- **bit stuffing used to avoid confusion with data containing flag sequence 01111110**

- 0 inserted after every sequence of five 1s
- if receiver detects five 1s it checks next bit
- if next bit is 0, it is deleted (was stuffed bit)
- if next bit is 1 and seventh bit is 0, accept it as flag sequence
- if sixth and seventh bits 1, sender is indicating abort



Original Pattern:

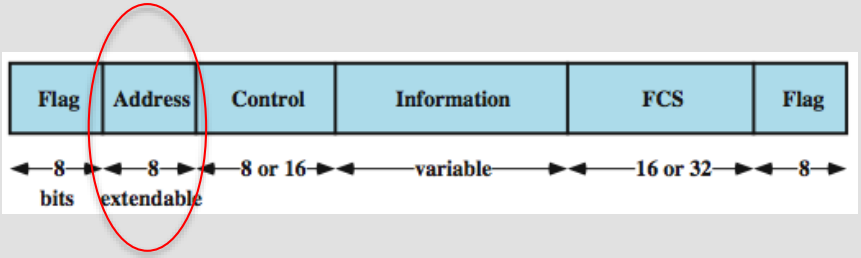
111111111111011111101111110

After bit-stuffing

111110111110110111110101111010

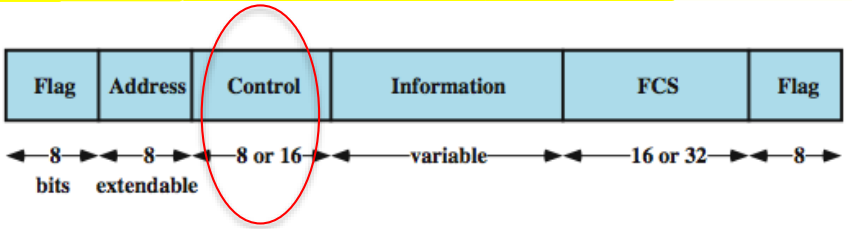
Address Field

- identifies the secondary station if it transmitted or is to receive the frame
- usually 8 bits long

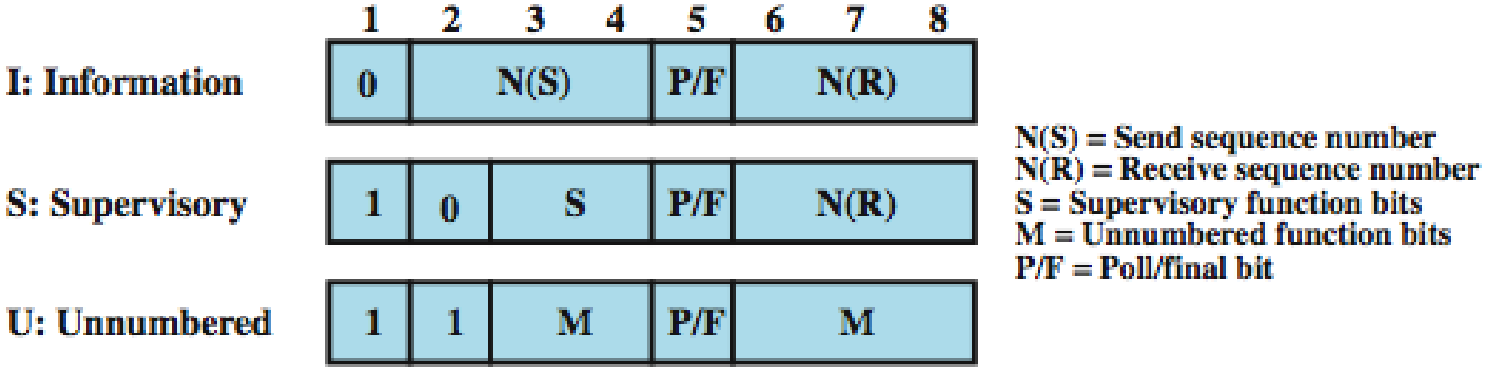


- may be extended to multiples of 7 bits
  - Leftmost Bit of each octet is (1) or (0) – T_x or R_x
  - The remaining 7 bits of each octet form the address.
- all ones address 11111111 is broadcast

Control Field (8 bit)



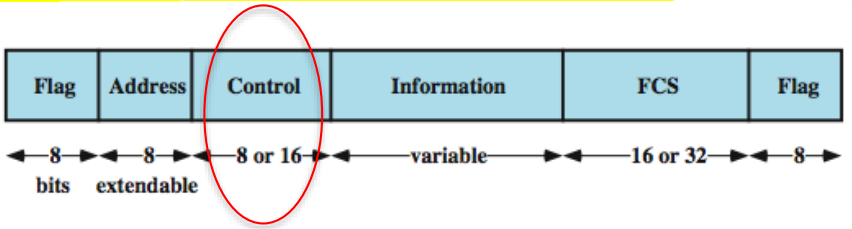
- **different for different frame type**
  - **I: Information** - carry data to be transmitted for the user (next layer up)
    - > Additionally - Flow and error control with ARQ **with piggybacked** on information frames
  - **S: Supervisory** - ARQ **when piggyback not used**
  - **U: Unnumbered** – provides supplementary link control functions
- **Starting 1-2 bits of control field identify frame type**



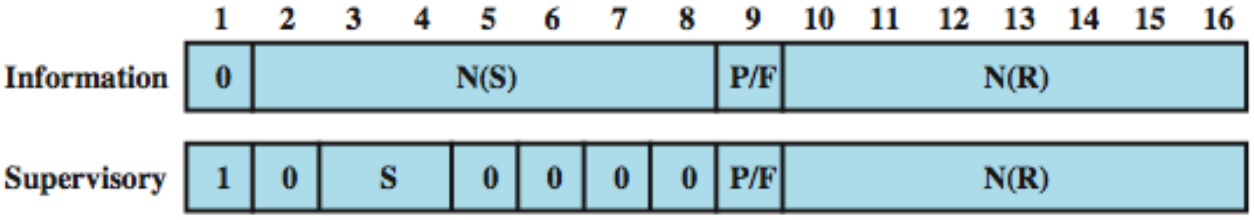
N(S) = Send sequence number
N(R) = Receive sequence number
S = Supervisory function bits
M = Unnumbered function bits
P/F = Poll/final bit

(c) 8-bit control field format

Control Field (16 bit)



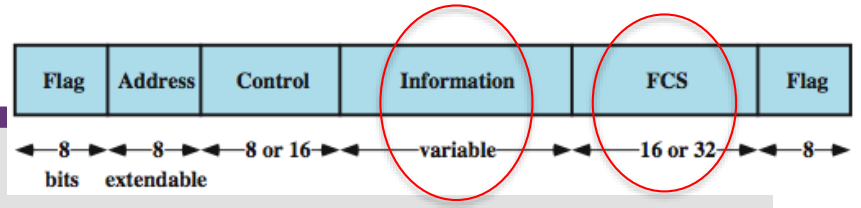
- **use of Poll (P) / Final (F) bit depends on context**
  - in command frame it is referred as Poll - P bit, set to1 to solicit (poll) response from peer
  - in response frame it is referred as F bit set to 1 to indicate response to soliciting command
- **sequence number usually 3 bits**
  - can extend to 7 bits as shown below



N(S) = Send sequence number
N(R) = Receive sequence number
S = Supervisory function bits
M = Unnumbered function bits
P/F = Poll/final bit

(d) 16-bit control field format

Information & FCS Fields



- **Information Field**
  - The information field is present only in Information I-frames and some Unnumbered U-frames
  - must contain integral number of octets
  - variable length
- **Frame Check Sequence Field (FCS)**
  - used for error detection
  - Used for line reliability
  - used and depends on line quality and reliability
  - either 16 bit CRC or 32 bit CRC used for FCS

HDLC Operation

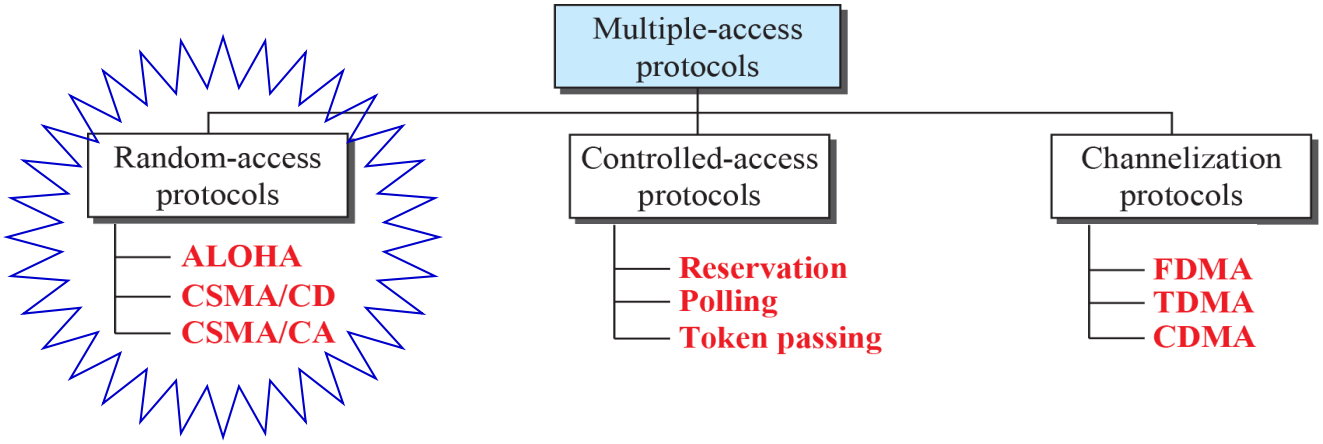
- **consists of exchange of:**
  - Information **(I-Frame)**
  - Supervisory **(S-Frame)** and
  - Unnumbered **(U-Frame)**
- **have three phases**
  - **First** initialization
    - by either side, set mode (NRM, ABM, or ARM) & sequence **of 3-bit or 7-bit** agreed
    - Logical connection is established.
  - **Second** data transfer
    - with flow and error control
    - using both **I** & **S**-frames (RR, RNR, REJ, SREJ)
    - (RR) RecieveReady, (RNR) RecieveNotReady,
    - (REJ) Reject, and (SREJ) SelectiveReject
    - Frames with sequence # of modulo 8 (**3-bit**) **or** modulo-128 (**7-bit**)
  - **Third** disconnect
    - when ready to disconnect or connection fault noted

5-3 MULTIPLE ACCESS PROTOCOLS

- *We said that the data-link layer is divided into two sublayers:*
  - ***data link control (DLC) and***
  - ***media access control (MAC)***
- *We discussed DLC in the previous section;*
- *we talk about MAC in this section.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.28: Taxonomy of multiple-access protocols



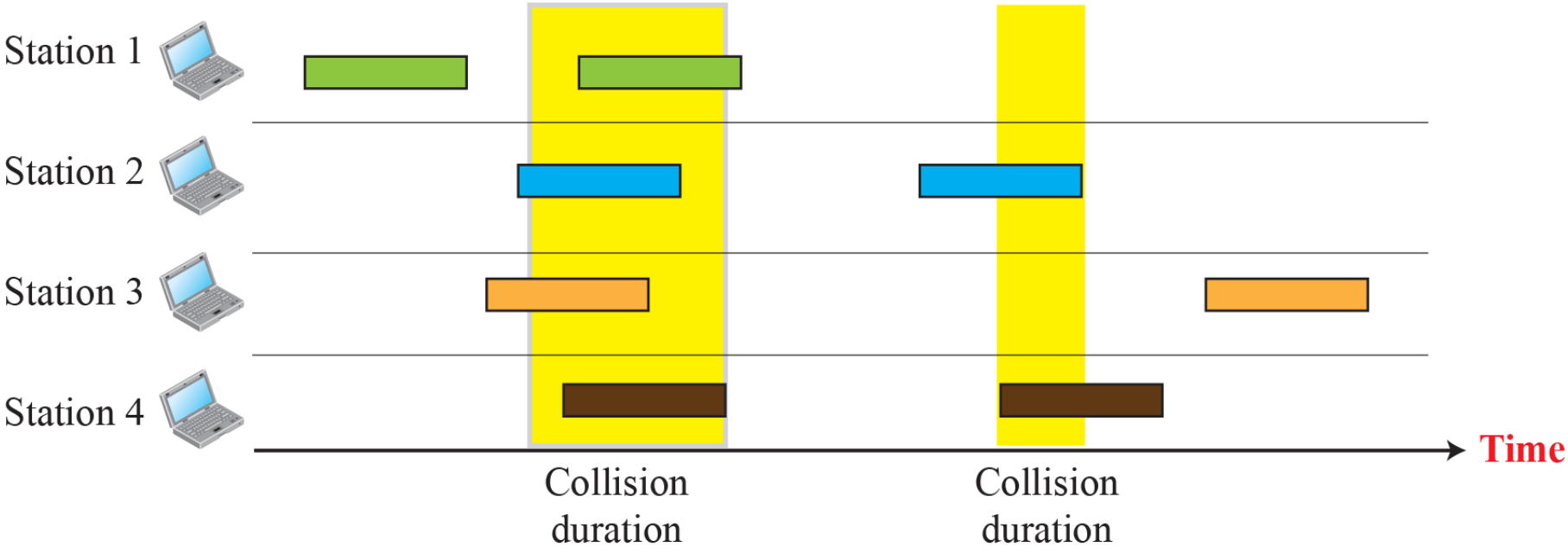
- ***ALOHA***
  - ***Pure ALOHA***
  - ***Slotted ALOHA***
- ***CSMA***
  - ***Vulnerable Time***
  - ***Persistence Methods***
- ***CSMA/CD***
  - ***Minimum Frame Size***
  - ***Procedure***
  - ***Energy Level***
  - ***Throughput***
  - ***Traditional Ethernet***
- ***CSMA/CA***

Random Access

- *In random-access or contention methods, no station is superior to another station and none is assigned the control over another.*
- *At each instance, a station that has data to send uses a procedure defined by the protocol to make a decision on whether or not to send.*
- *This decision depends on the state of the medium (idle or busy).*
- *In other words, each station can transmit when it desires on the condition that it follows the predefined procedure, including the testing of the state of the medium.*

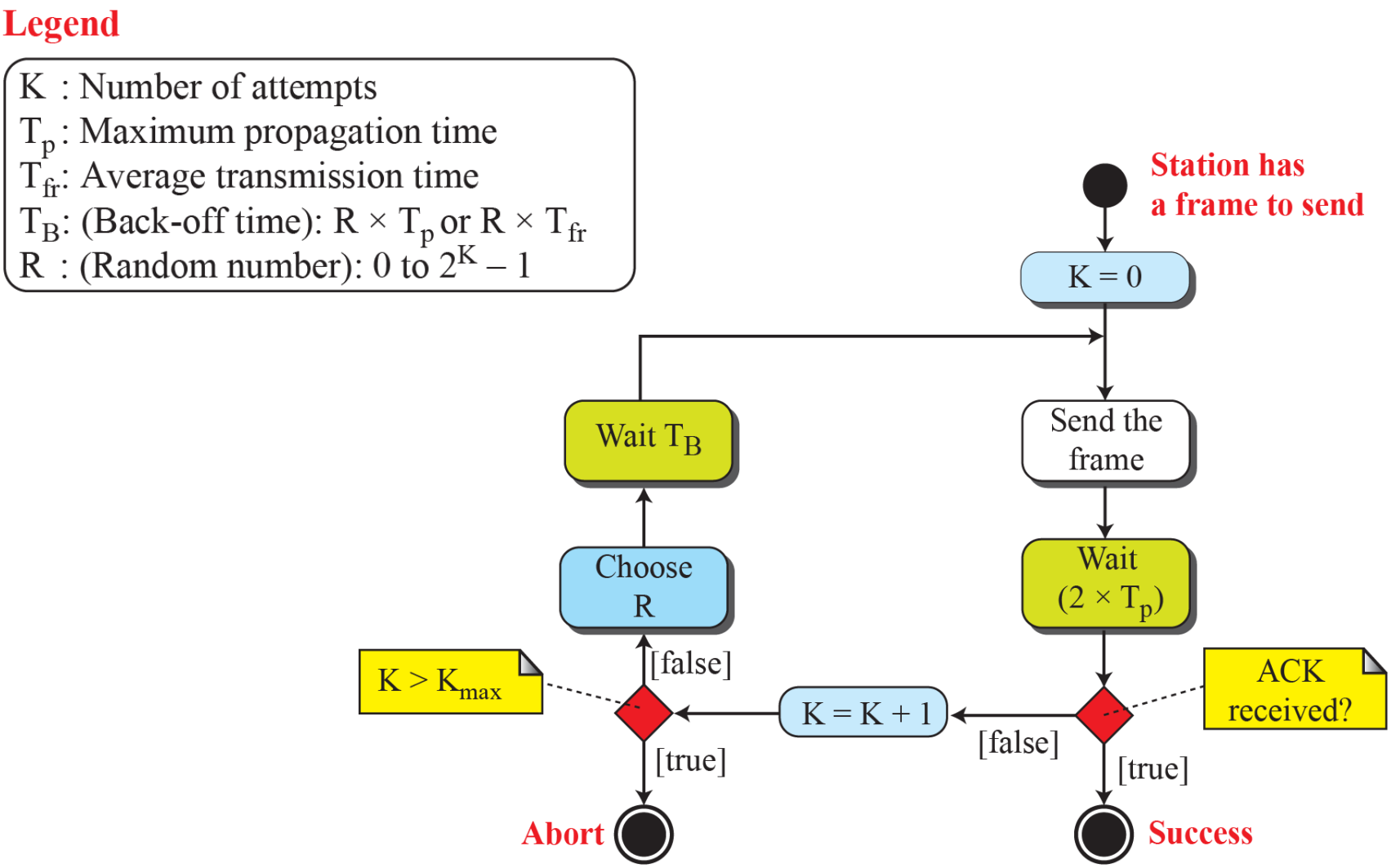
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.29: Frames in a pure ALOHA network



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.30: Procedure for pure ALOHA protocol



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Example 5.11

- 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
- Max propagation time $= T_p = (600 \times 10^3) / (3 \times 10^8) = 2$ ms.

- For K = 2, the range of random number = R = {0, 1, 2, 3}
- T_B is R x T_B = { 0, 2, 4, 6} millisec.

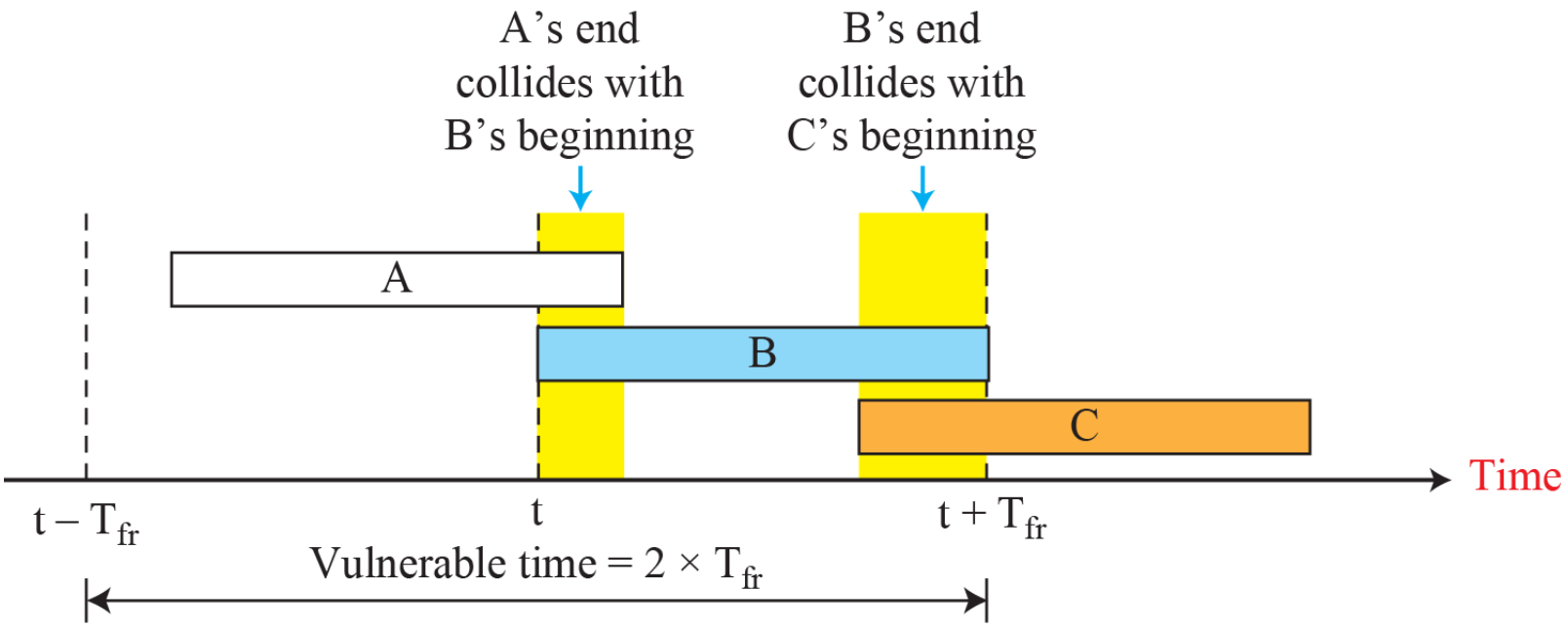
This means that T_B can be 0, 2, 4, or 6 ms, based on the outcome of the random variable R.

Legend

K : Number of attempts
T_p : Maximum propagation time
T_{fr}: Average transmission time
T_B: (Back-off time): R × T_p or R × T_{fr}
R : (Random number): 0 to $2^K - 1$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.31: Vulnerable time for pure ALOHA protocol



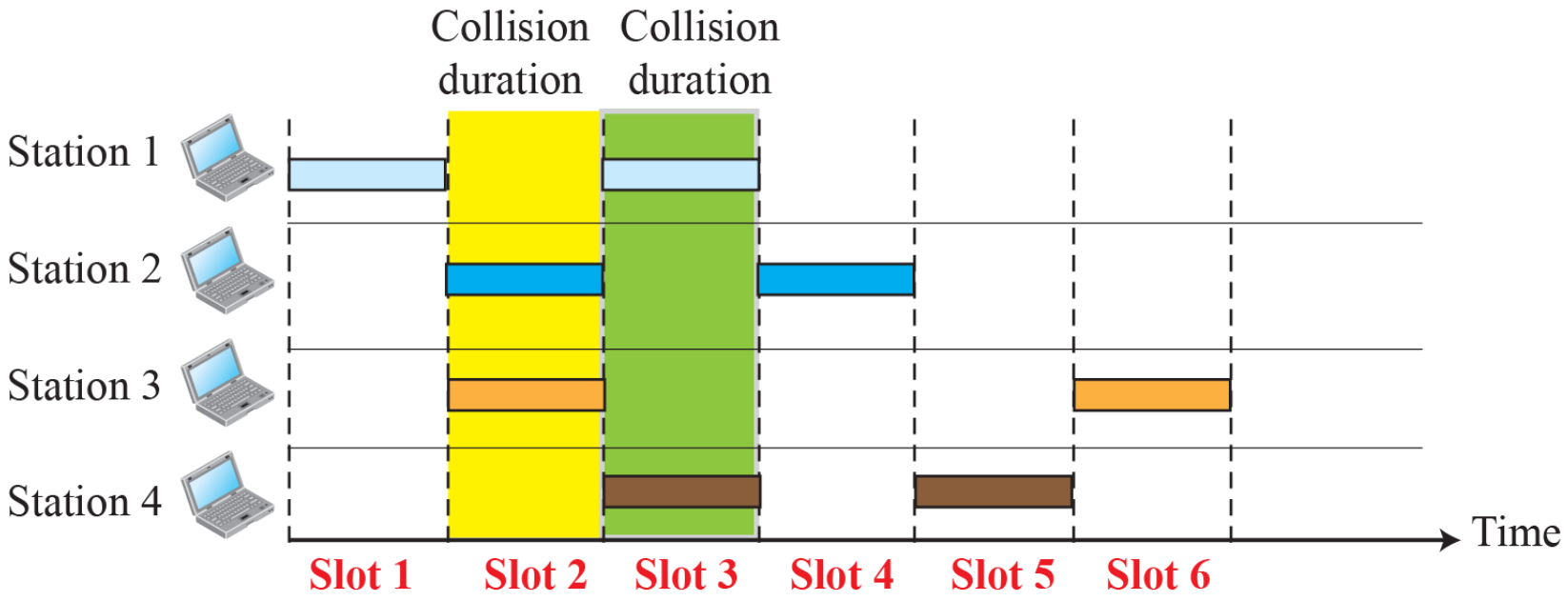
A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the requirement to make this frame collision-free?

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 period (1 ms) that this station is sending.

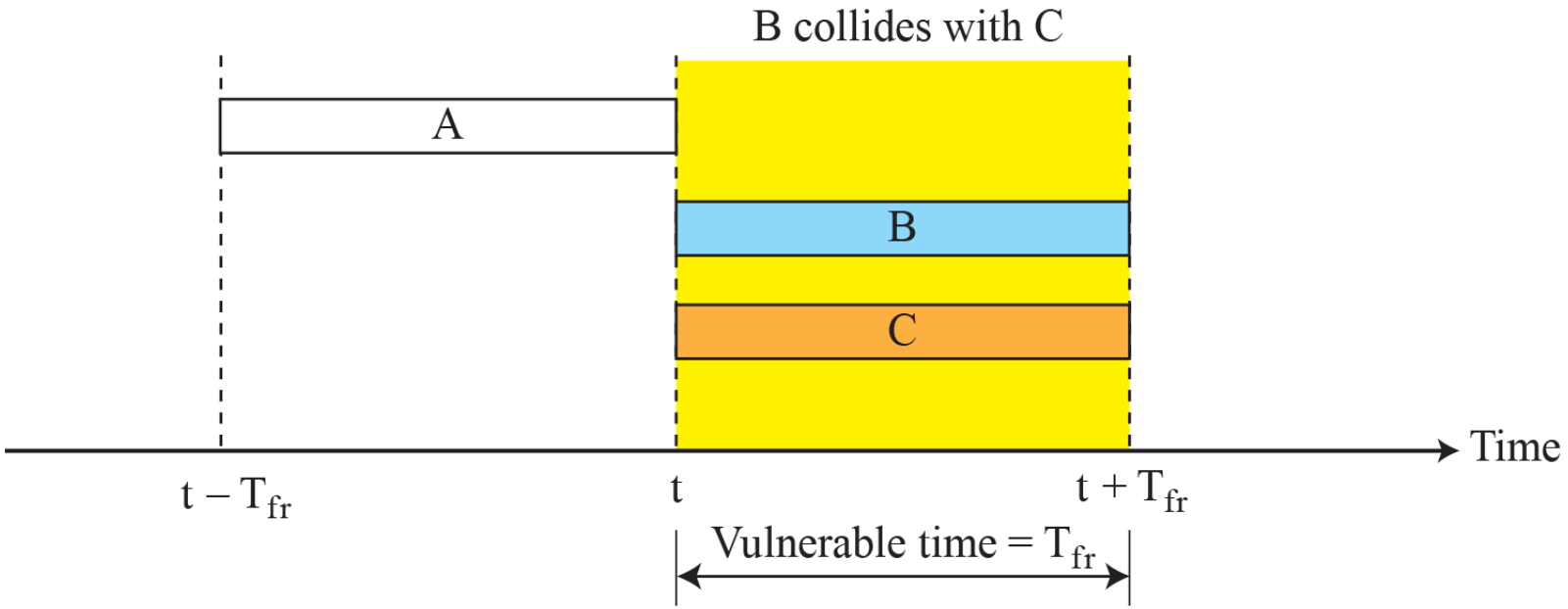
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.32: Frames in a slotted ALOHA network



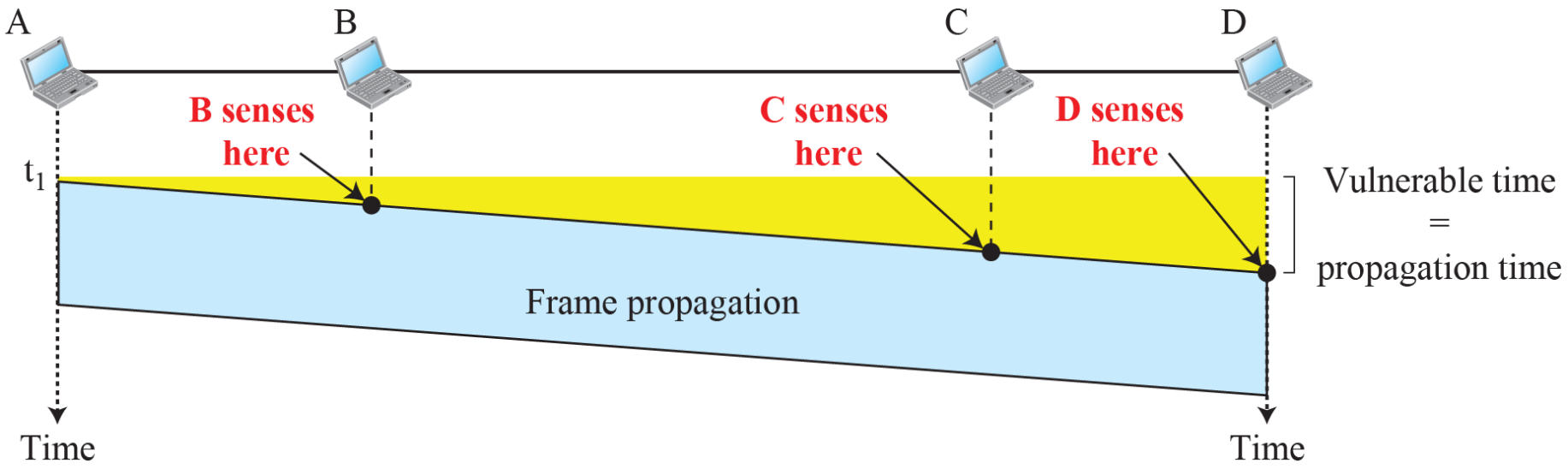
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.33: Vulnerable time for slotted ALOHA protocol



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

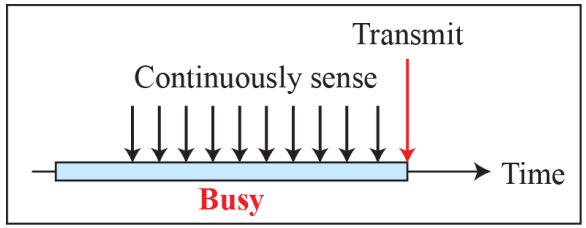
Figure 5.35: Vulnerable time in CSMA



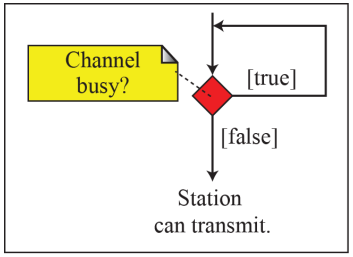
The vulnerable time for CSMA is the ***propagation time*** *Tp*. This is the time needed for a signal to propagate from one end of the medium to the other.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

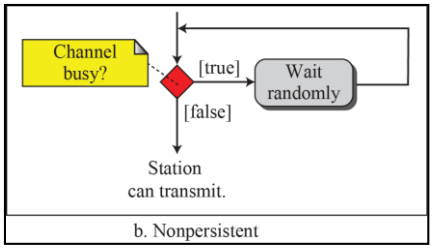
Figure 5.36 & 5.37: Behavior & Flow Diagram of three persistence methods



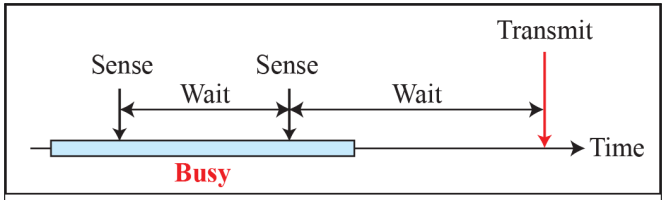
a. 1-persistent



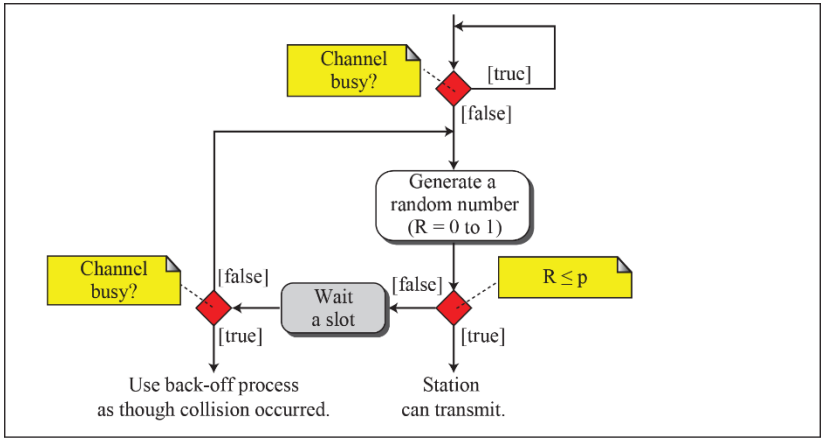
a. 1-persistent



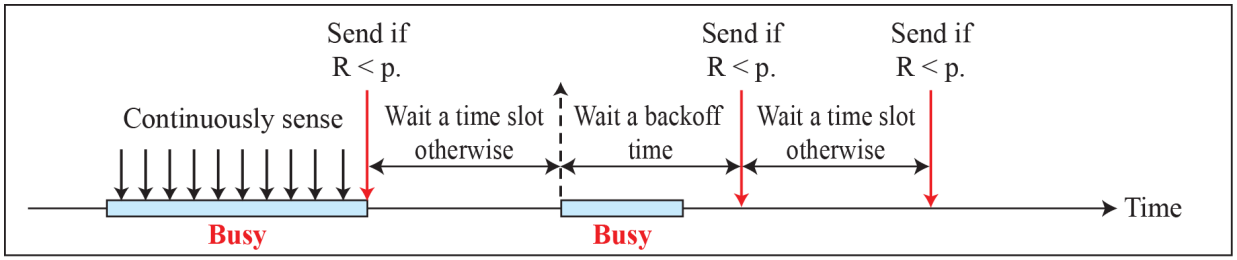
b. Nonpersistent



b. Nonpersistent



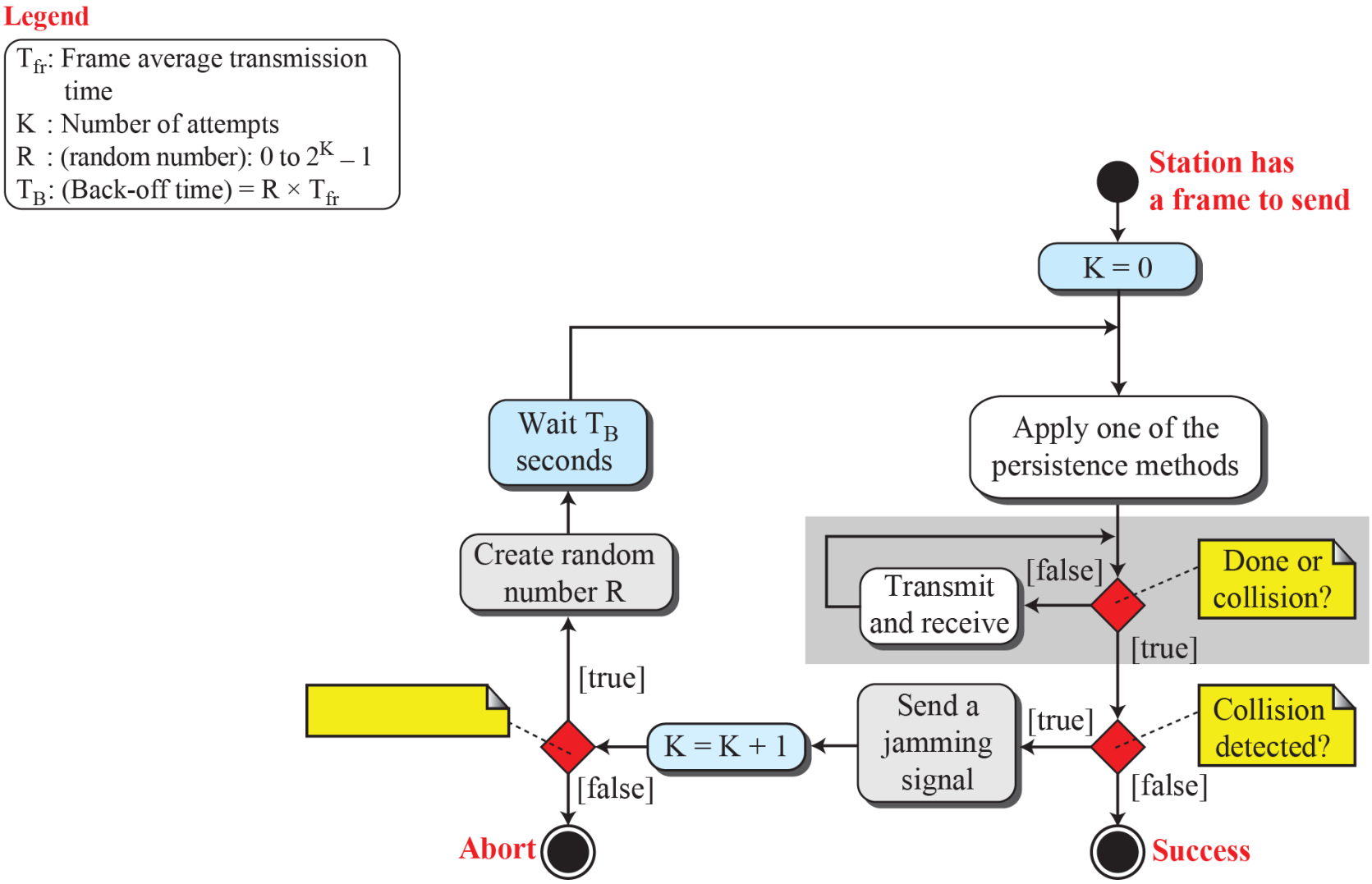
c. *p*-persistent



c. *p*-persistent

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Figure 5.40: Flow diagram for the CSMA/CD



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Next week

Completion of Chapter 5

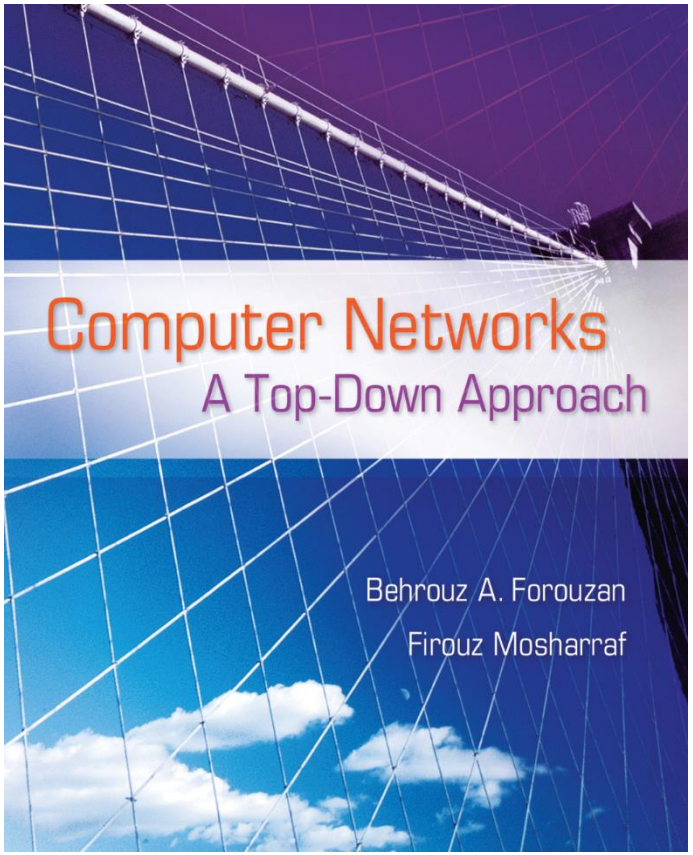
Data-Link Layer: Wired Networks

- 5.5 WIRED LANS: ETHERNET PROTOCOL
- 5.6 OTHER WIRED NETWORKS
- 5.7 CONNECTING DEVICES

Chapter 6

Wireless Networks

- 5.5 WIRED LANS: ETHERNET PROTOCOL
- WIRELESS-LANS
- OTHER WIRELESS-NETWORKS



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Chapter 5: Summary

- *We considered the **data-link layer** as **two sublayers**.*
- *The upper sublayer is responsible for data link control, and the lower sublayer is responsible for resolving access to the shared media.*
- *Data link control (DLC) deals with the design and procedures for communication between two adjacent nodes: node-to-node communication. This sublayer is responsible for framing and error control. Error control deals with data corruption during transmission.*
- *We discussed two link-layer protocols in this chapter: HDLC and PPP.*
- *Many formal protocols have been devised to handle access to a shared link. We categorize them into three groups: random access protocols, controlled access protocols, and channelization protocols. And focused on random access protocols.*
- *At the data-link layer, we use link-layer addressing. Ethernet is the most widely used local area network protocol.*
- *The data-link layer of Ethernet consists of the LLC sublayer and the MAC sublayer.*
- *The MAC sublayer is responsible for the operation of the CSMA/CD access method and framing.*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.