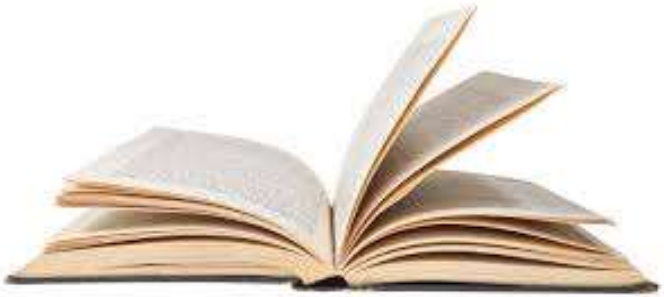
Data Communications and Networking

Lecture 5



Dr. Su Mon Thu
Associate Professor
Department of Computer Engineering and Information Technology
Yangon Technological University

Outlines

- Introduction to Data-Link Layer
- Error Detection and Correction
- Data Link Control (DLC)
- Media Access Control (MAC)

Lecture Objectives

- To introduce
  - Data-Link Layer
  - types of error and concept of redundancy
  - High-Level Data Link Control (HDLC)
- To understand the concepts of link and nodes and discuss link-layer addressing
- To know how error can be detected using block coding and discuss cyclic codes, and checksums
- To discuss services provided by the DLC sublayer, data-link protocols
- To discuss multiple-access protocols

Topic 1: Introduction to Data-Link Layer

Introduction to Data-Link Layer

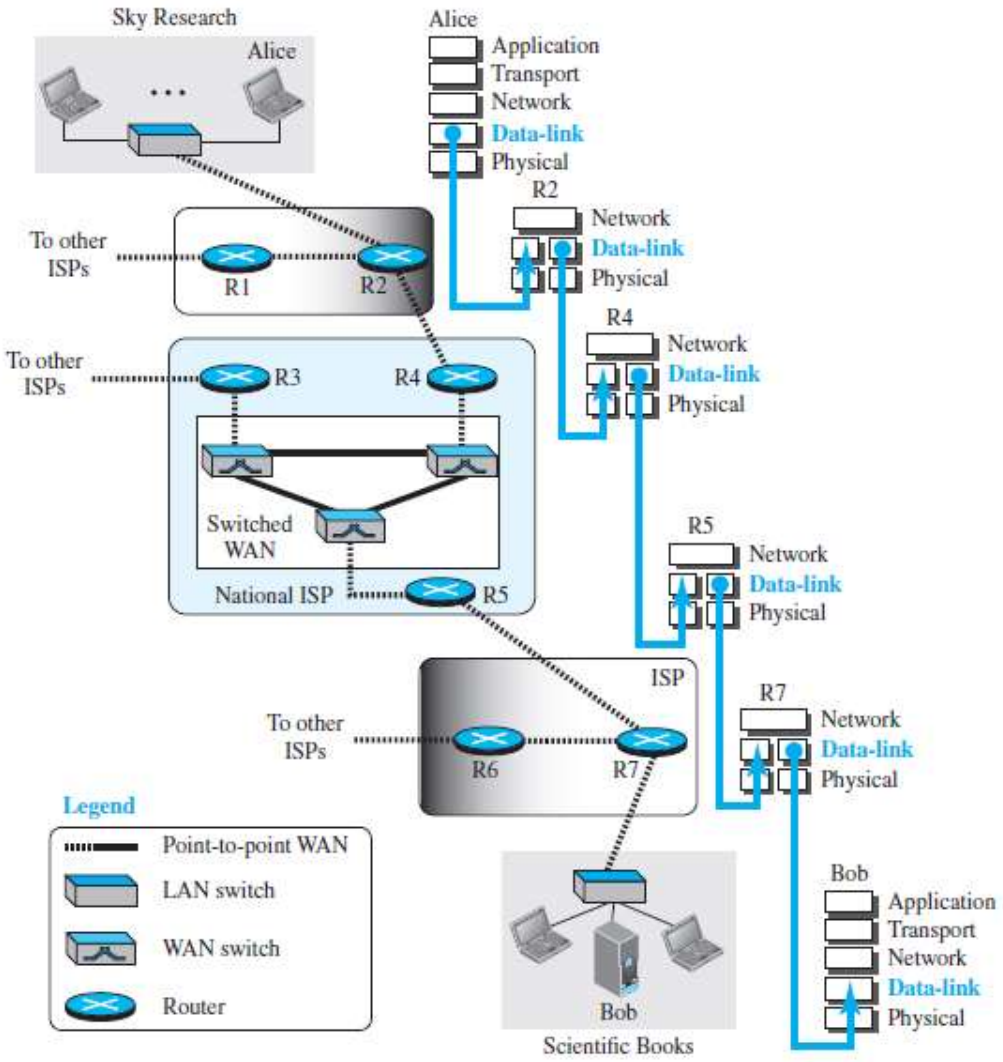


Figure 1. Communication at the Data-Link Layer

Introduction to Data-Link Layer (Continue)

Nodes and Links

- Communication at the data-link layer is node-to-node.
- 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 refers to the two end hosts and the routers as nodes and the networks in between as links.
- Figure 2 is a simple representation of links and nodes when the path of the data unit is only six nodes.

Introduction to Data-Link Layer (Continue)

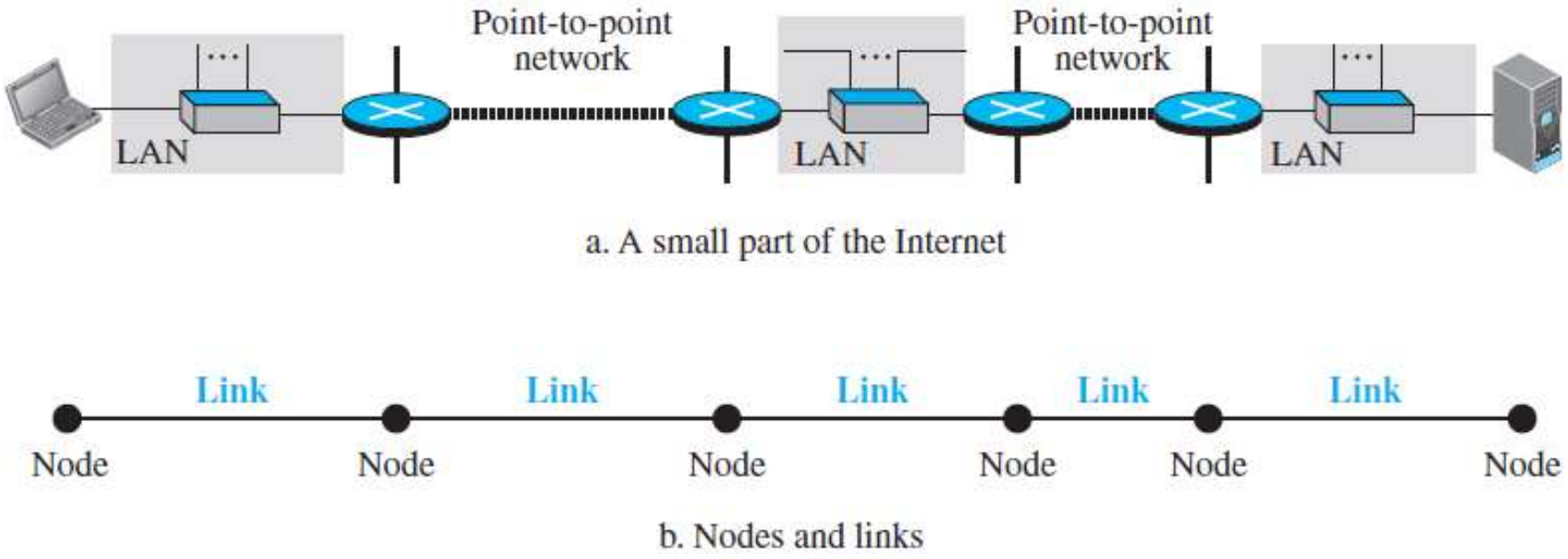


Figure 2. Nodes and Links

Services

- The services provided by a data-link layer are framing, flow control, error control and congestion control.

Introduction to Data-Link Layer (Continue)

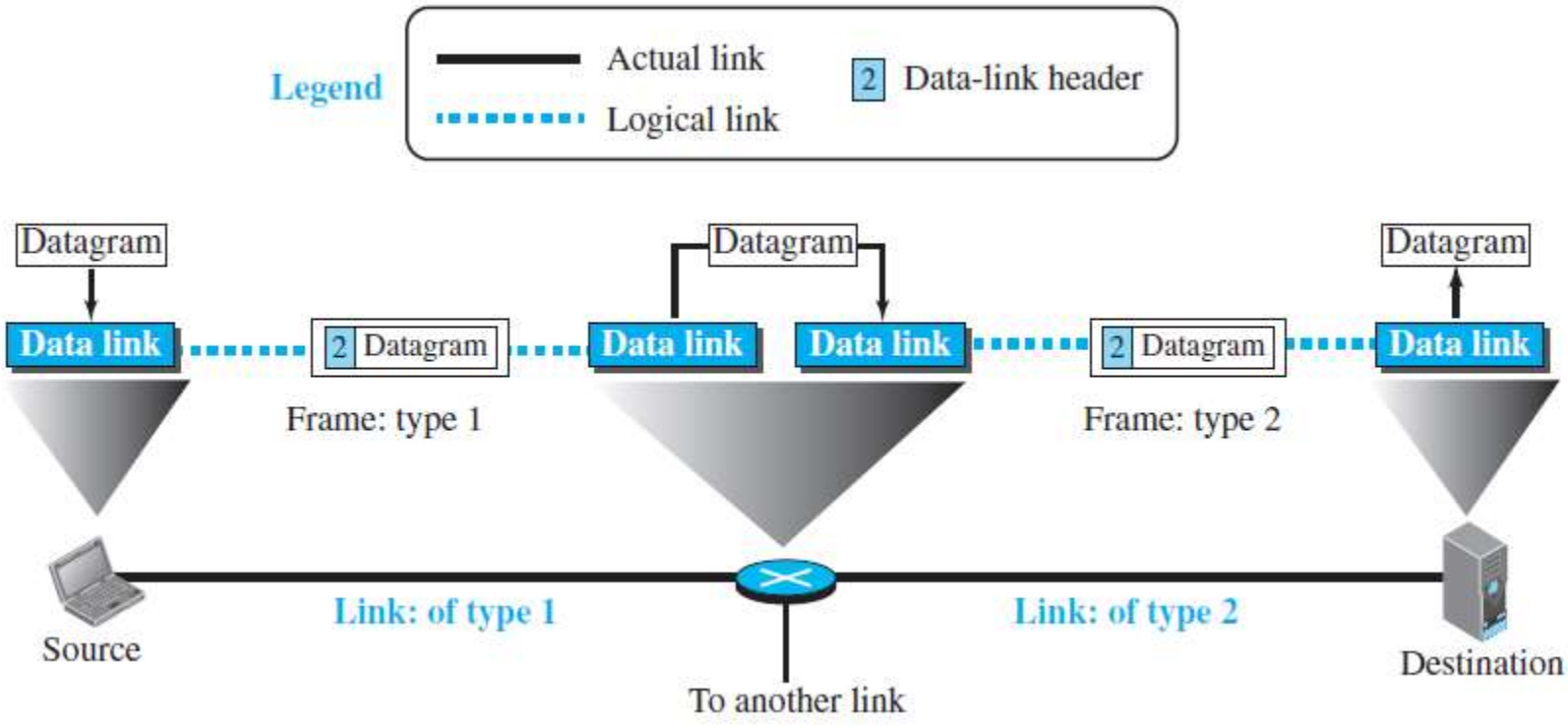


Figure 3. A Communication with Only Three Nodes

Introduction to Data-Link Layer (Continue)

Framing

- The first service provided by the data-link layer is framing.
- The data-link layer at each node needs to encapsulate the datagram in a frame before sending it to the next node.
- The node also needs to decapsulate the datagram from the frame received on the logical channel.
- A packet at the data-link layer is normally called a frame.

Flow Control

- If the producer produces items that cannot be consumed, accumulation of items occurs.
- The sending data-link layer at the end of a link is a producer of frames; the receiving data-link layer at the other end of a link is a consumer.
- If the rate of produced frames is higher than the rate of consumed frames, frames at the receiving end need to be buffered while waiting to be consumed (processed).

Introduction to Data-Link Layer (Continue)

Error Control

- At the sending node, a frame in a data-link layer needs to be changed to bits, transformed to electromagnetic signals, and transmitted through the transmission media.
- At the receiving node, electromagnetic signals are received, transformed to bits, and put together to create a frame.
- Since electromagnetic signals are susceptible to error, a frame is susceptible to error.
- The error needs first to be detected.
- After detection, it needs to be either corrected at the receiver node or discarded and retransmitted by the sending node.

Introduction to Data-Link Layer (Continue)

Congestion Control

- Although a link may be congested with frames, which may result in frame loss, most data-link-layer protocols do not directly use a congestion control to alleviate congestion, although some wide-area networks do.

Two Categories of Links

- There are point-to-point link and broadcast link.
- In a point-to-point link, the link is dedicated to the two devices; in a broadcast link, the link is shared between several pairs of devices.

Two Sublayers

- The data-link layer can be divided into two sublayers: data link control (DLC) and media access control (MAC).

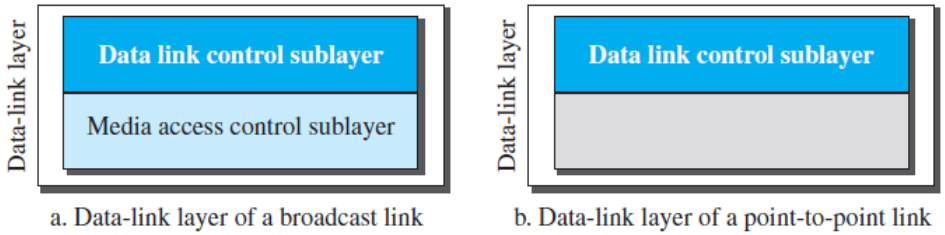


Figure 4. Dividing the data-link layer into two sublayers

Link-layer Addressing

- A link-layer address is sometimes called a link address, sometimes a physical address, and sometimes a MAC address.
- Since a link is controlled at the data-link layer, the addresses need to belong to the data-link layer.
- When a datagram passes from the network layer to the data-link layer, the datagram will be encapsulated in a frame and two data-link addresses are added to the frame header.
- These two addresses are changed every time the frame moves from one link to another.
- Figure 5 demonstrates the concept in a small internet.
- In this figure, there have three links and two routers.

Link-layer Addressing (Continue)

- It also has shown only two hosts: Alice (source) and Bob (destination).
- For each host, there are two addresses, the IP addresses (N) and the link-layer addresses (L).

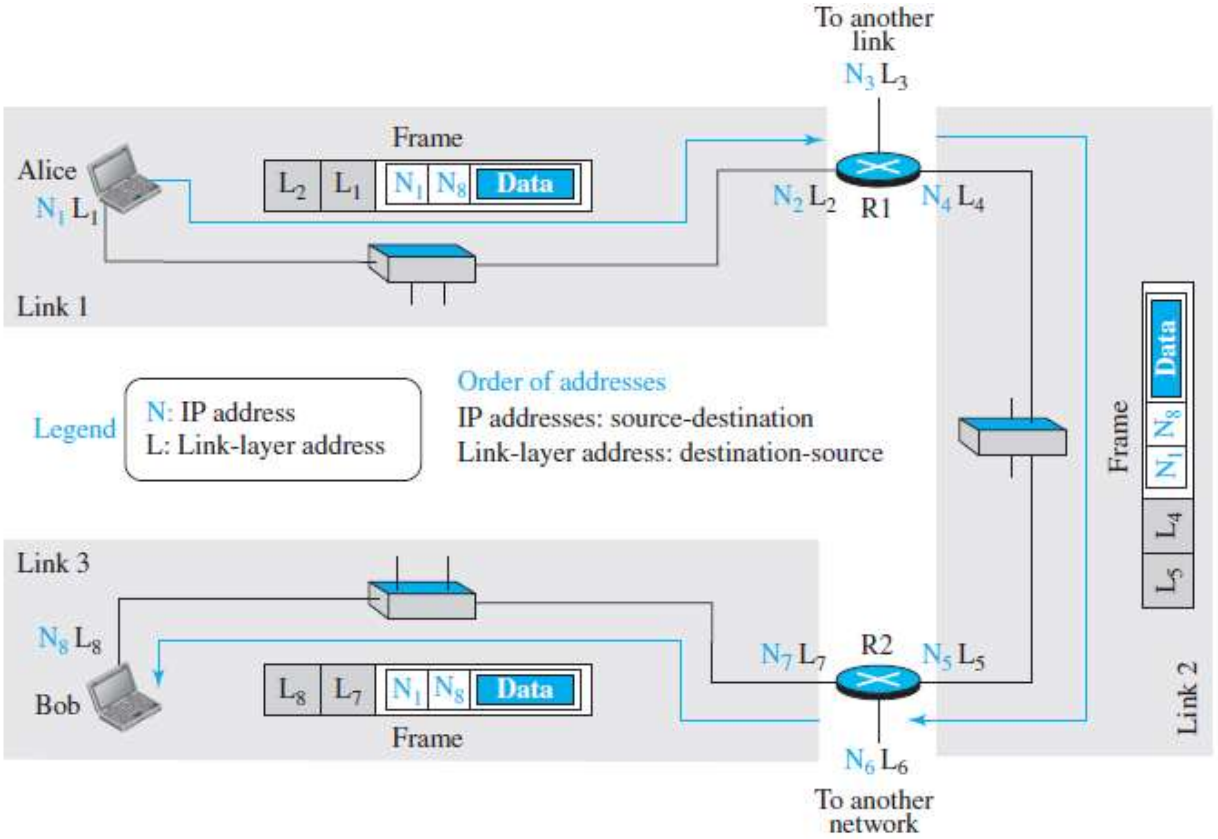


Figure 5. IP Addresses and Link-Layer Addresses in a Small Internet

Link-layer Addressing (Continue)

Three Types of Addresses

- Some link-layer protocols define three types of addresses: unicast, multicast, and broadcast.

Unicast Address

- Each host or each interface of a router is assigned a unicast address.
- Unicasting means one-to-one communication.
- A frame with a unicast address destination is destined only for one entity in the link.

Multicast Address

- Some link-layer protocols define multicast addresses.
- Multicasting means one-to-many communication.

Link-layer Addressing (Continue)

Broadcast Address

- Some link-layer protocols define a broadcast address.
- Broadcasting means one-to-all communication.
- A frame with a destination broadcast is sent to all entities in the link.

Address Resolution Protocol (ARP)

- The ARP protocol is one of the auxiliary protocols defined in the network layer, as shown in figure 6.
- It belongs to the network layer, but it maps an IP address to a logical-link address.
- ARP accepts an IP address from the IP protocol, maps the address to the corresponding link-layer address, and passes it to the data-link layer.

Link-layer Addressing (Continue)

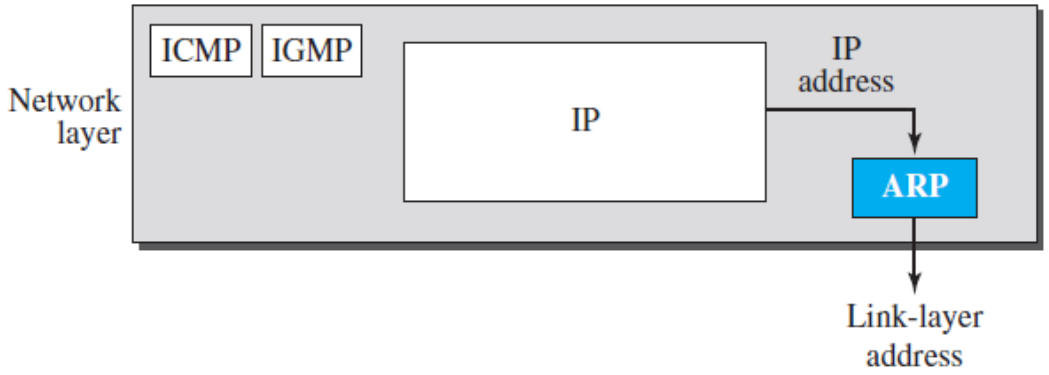


Figure 6. Position of ARP in TCP/IP protocol suite

- Every host or router on the network receives and processes the ARP request packet, but only the intended recipient recognizes its IP address and sends back an ARP response packet.
- The response packet contains the recipient's IP and link-layer addresses.
- The packet is unicast directly to the node that sent the request packet.
- In figure 7a, the system on the left (A) has a packet that needs to be delivered to another system (B) with IP address N2.

Link-layer Addressing (Continue)

- System A needs to pass the packet to its data-link layer for the actual delivery, but it does not know the physical address of the recipient.
- It uses the services of ARP by asking the ARP protocol to send a broadcast ARP request packet to ask for the physical address of a system with an IP address of N2.
- This packet is received by every system on the physical network, but only system B will answer it, as shown in Figure 7b.
- System B sends an ARP reply packet that includes its physical address.
- Now system A can send all the packets it has for this destination using the physical address it received.

Link-layer Addressing (Continue)

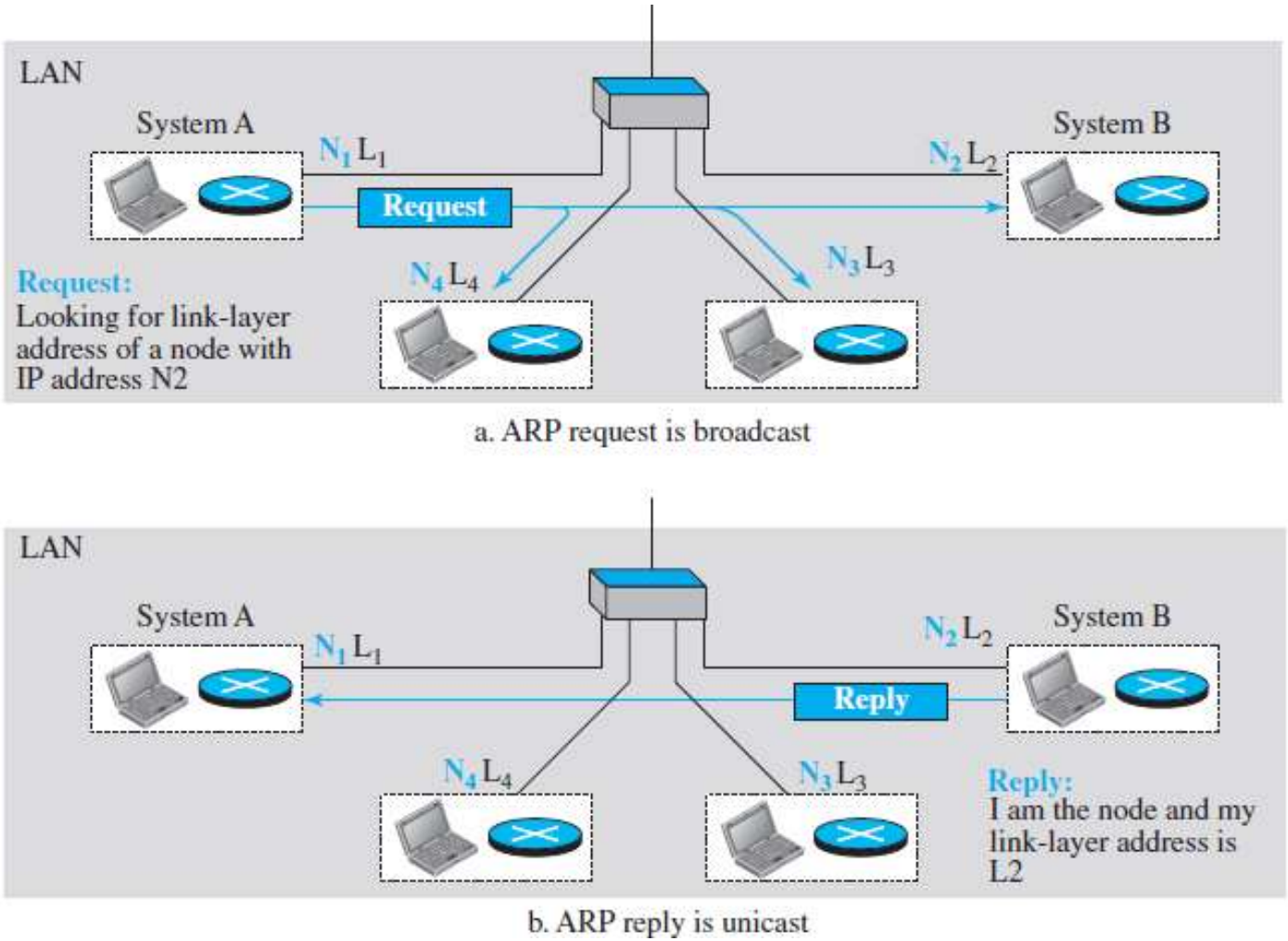


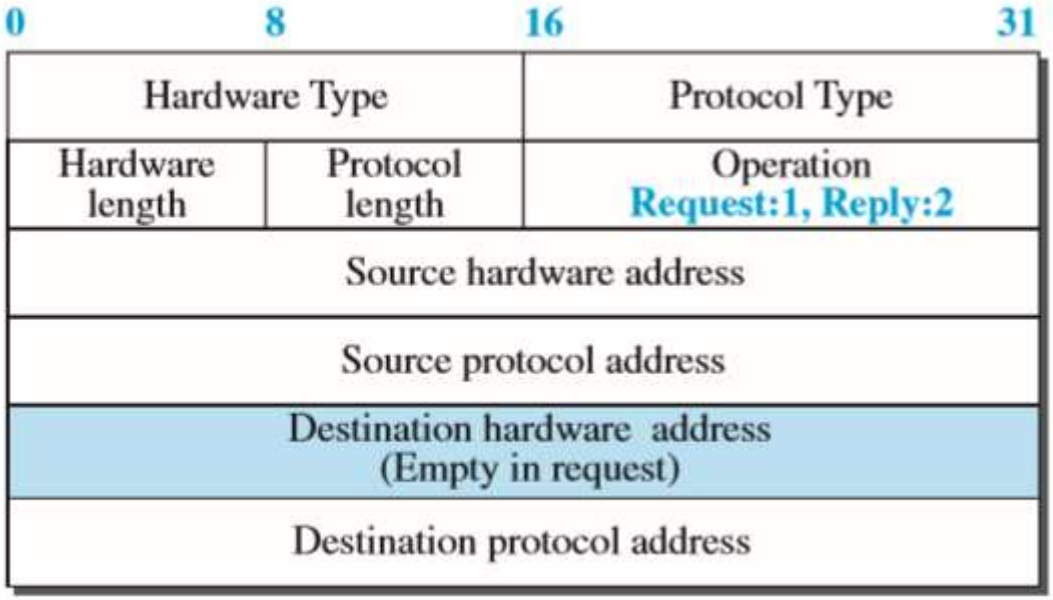
Figure 7. ARP Operation

Link-layer Addressing (Continue)

Packet Format

- Figure 8 shows the format of an ARP packet.
- The hardware type field defines the type of the link-layer protocol; Ethernet is given the type 1.
- The protocol type field defines the network-layer protocol: IPv4 protocol is $(0800)_{16}$.
- The source hardware and source protocol addresses are variable-length fields defining the link-layer and network-layer addresses of the sender.
- The destination hardware address and destination protocol address fields define the receiver link-layer and network-layer addresses.
- An ARP packet is encapsulated directly into a data-link frame.

Link-layer Addressing (Continue)



Hardware: LAN or WAN protocol
Protocol: Network-layer protocol

Figure 8. ARP Packet

Link-layer Addressing (Continue)

Example 1: A host with IP address N1 and MAC address L1 has a packet to send to another host with IP address N2 and physical address L2 (which is unknown to the first host). The two hosts are on the same network. Figure 9 shows the ARP request and response messages.

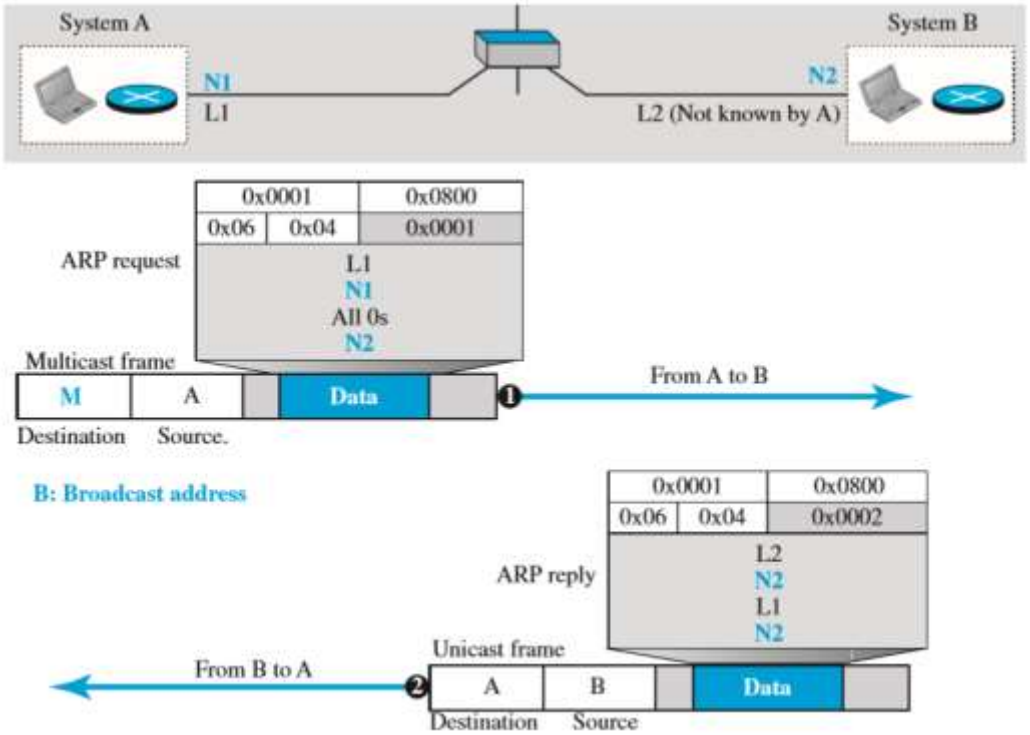


Figure 9. ARP Request and Response Messages

Link-layer Addressing (Continue)

Communication

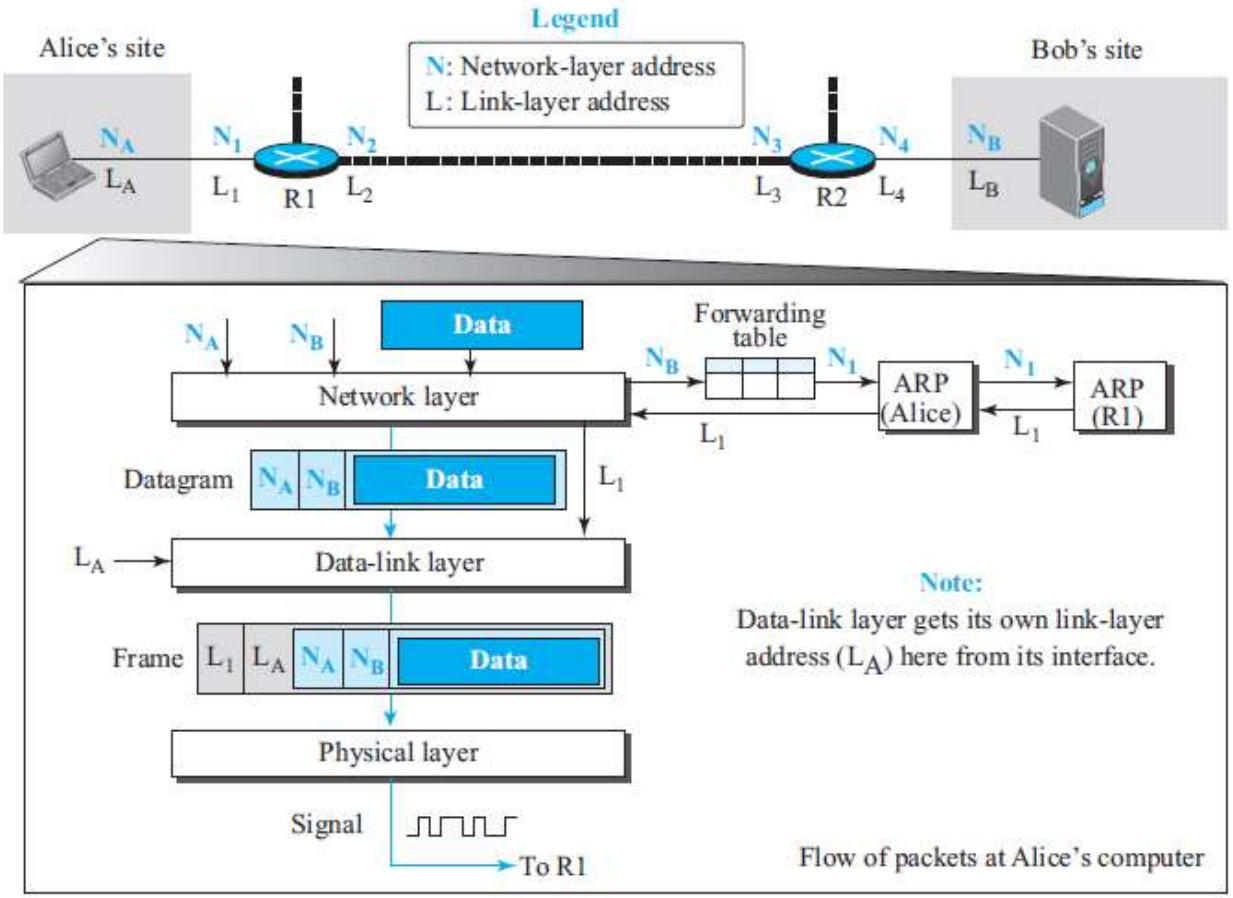


Figure 10. Flow of packets at Alice's computer

Link-layer Addressing (Continue)

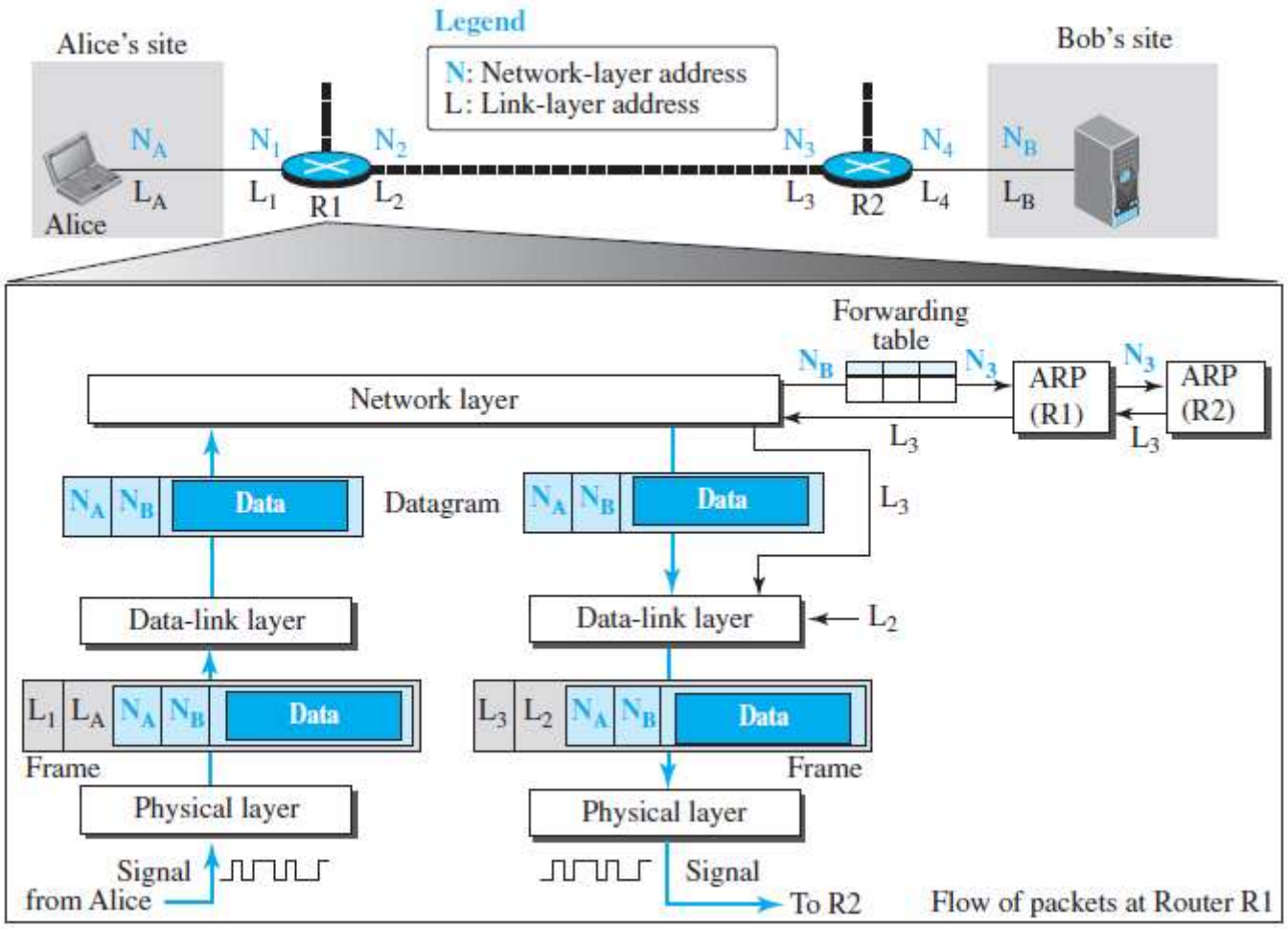


Figure 11. Flow of Activities at Router R1

Link-layer Addressing (Continue)

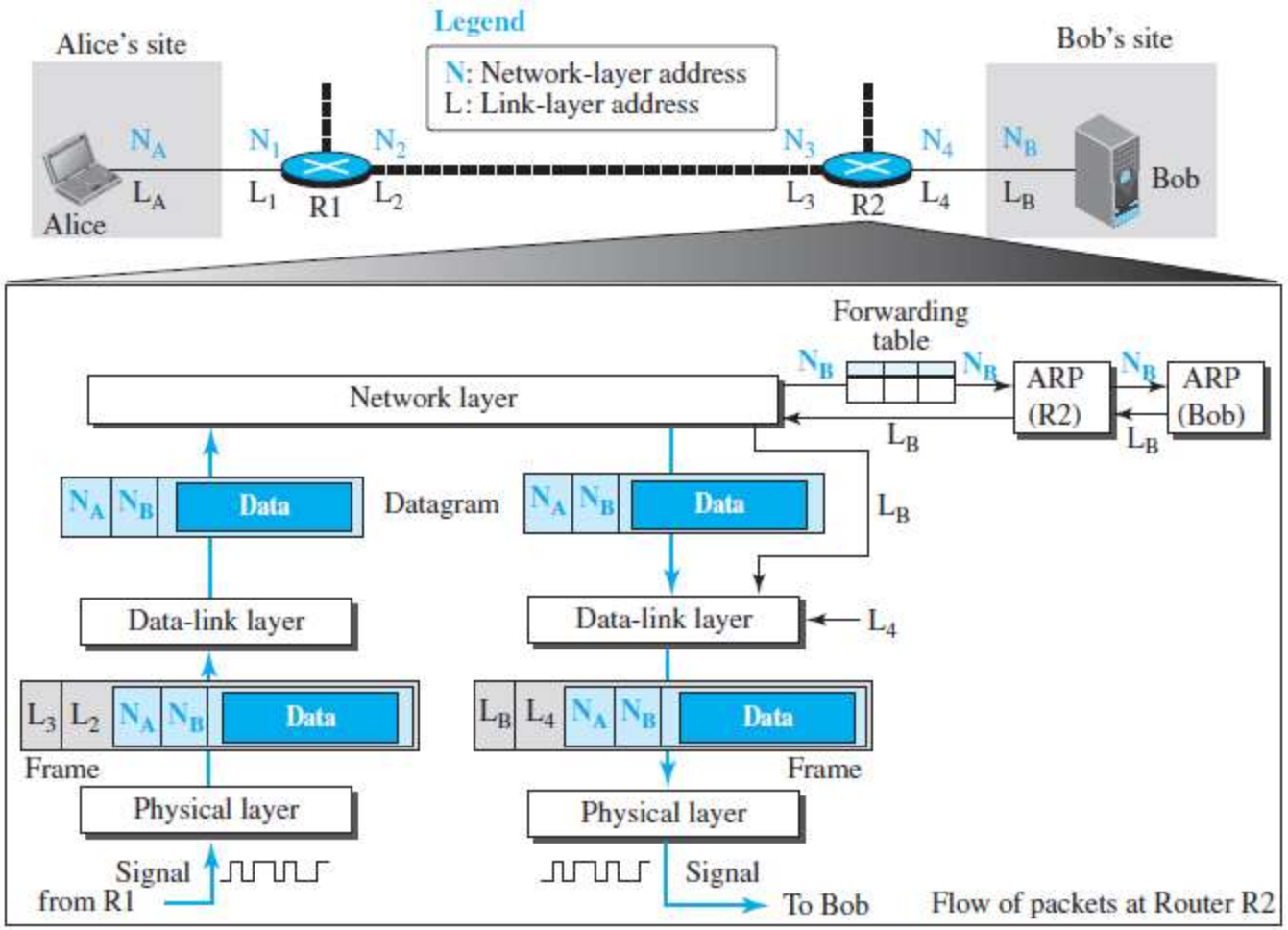


Figure 12. Flow of Activities at Router R2

Link-layer Addressing (Continue)

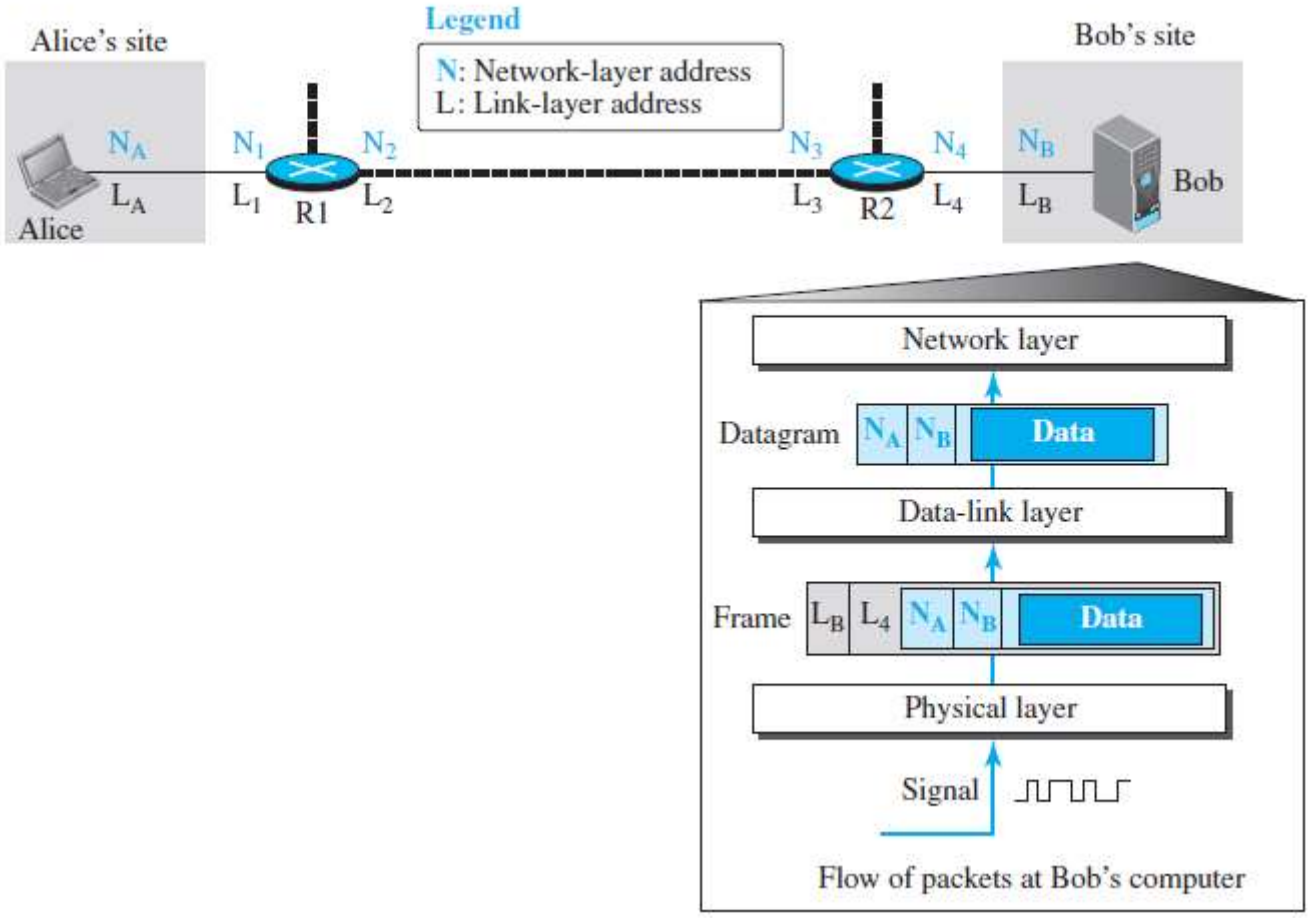


Figure 13. Activities at Bob's site

Topic 2: Error Detection and Correction

Error Detection and Correction: Introduction

Types of Errors

- The term single-bit error means that only 1 bit of a given data unit (such as a byte, character, or packet) is changed from 1 to 0 or from 0 to 1.
- The term burst error means that 2 or more bits in the data unit have changed from 1 to 0 or from 0 to 1.
- Figure 14 shows the effect of a single-bit and a burst error on a data unit.

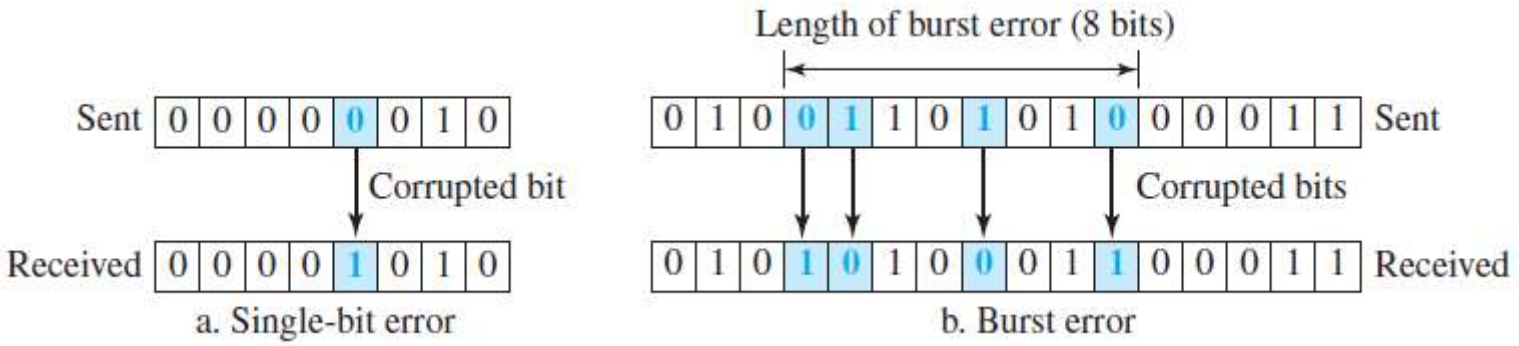


Figure 14. Single-bit and burst error

Introduction (Continue)

Redundancy

- Error detection uses the concept of redundancy, which means adding extra (redundant) bits for detecting errors at the destination.
- These redundant bits are added by the sender and removed by the receiver.

Block Coding

- In block coding, the message is divided into blocks, each of k bits, called datawords.
- The r redundant bits are added to each block to make the length n = k + r.
- The resulting n-bit blocks are called codewords.

Error Detection

- If the following two conditions are met, the receiver can detect a change in the original codeword.
  1. The receiver has (or can find) a list of valid codewords.
  2. The original codeword has changed to an invalid one.
- Figure 15 shows the role of block coding in error detection.

Block Coding (Continue)

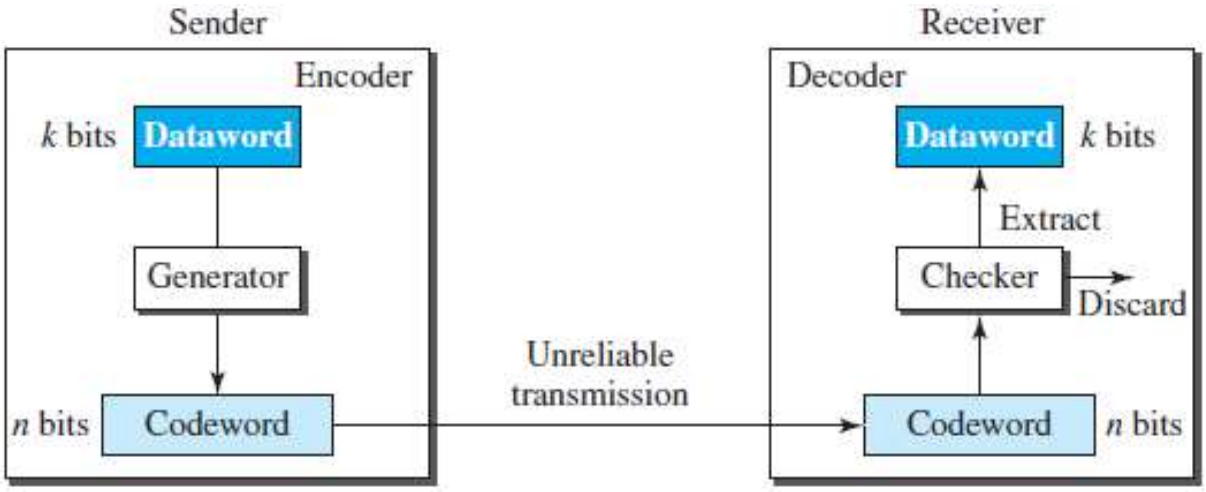


Figure 15. Process of Error Detection in Block Coding

Example 2

Let us assume that k = 2 and n = 3. Table 1 shows the list of datawords and codewords. Later, we will see how to derive a codeword from a dataword.

Table 1. A code for error detection

Dataword	*Codeword*	*Dataword*	*Codeword*
00	000	10	101
01	011	11	110

Block Coding (Continue)

Solution

- 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.

Block Coding (Continue)

Hamming Distance

- The Hamming distance between two words is the number of differences between corresponding bits.
- the Hamming distance between the received codeword and the sent codeword is the number of bits that are corrupted during transmission.
- For example, if the codeword 00000 is sent and 01101 is received, 3 bits are in error and the Hamming distance between the two is d(00000, 01101) = 3.
- In a set of codewords, the minimum Hamming distance is the smallest Hamming distance between all possible pairs of codewords.
- Now let us find the minimum Hamming distance in a code if it wants to be able to detect up to s errors.

Block Coding (Continue)

- If s errors occur during transmission, the Hamming distance between the sent codeword and received codeword is s.
- If the system is to detect up to s errors, the minimum distance between the valid codes must be (s + 1), so that the received codeword does not match a valid codeword.
- To guarantee the detection of up to s errors in all cases, the minimum Hamming distance in a block code must be $d_{min} = s + 1$.

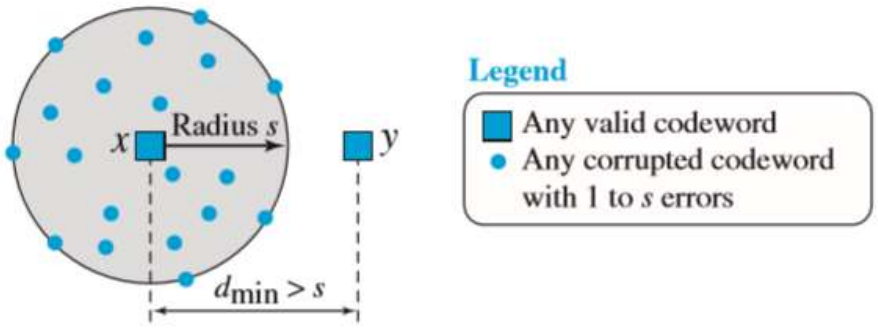


Figure 16. Geometric concept explaining d_{min} in error detection

Block Coding (Continue)

Linear Block Codes

- Almost all block codes used today belong to a subset of block codes called linear block codes.
- The formal definition of linear block codes requires the knowledge of abstract algebra (particularly Galois fields), which is beyond the scope of this book.
- A linear block code is a code in which the exclusive OR (addition modulo-2) of two valid codewords creates another valid codeword.

Parity-Check Code

- The most familiar error-detecting code is the parity-check code.
- This code is a linear block code.
- In this code, a k-bit dataword is changed to an n-bit codeword where n = k + 1.

Block Coding (Continue)

- The extra bit, called the parity bit, is selected to make the total number of 1s in the codeword even.
- Although some implementations specify an odd number of 1s.
- The code in Table 2 is also a parity-check code with k = 4 and n = 5.

Table 2: Simple parity-check code C(5, 4)

Dataword	*Codeword*	*Dataword*	*Codeword*
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**

Block Coding (Continue)

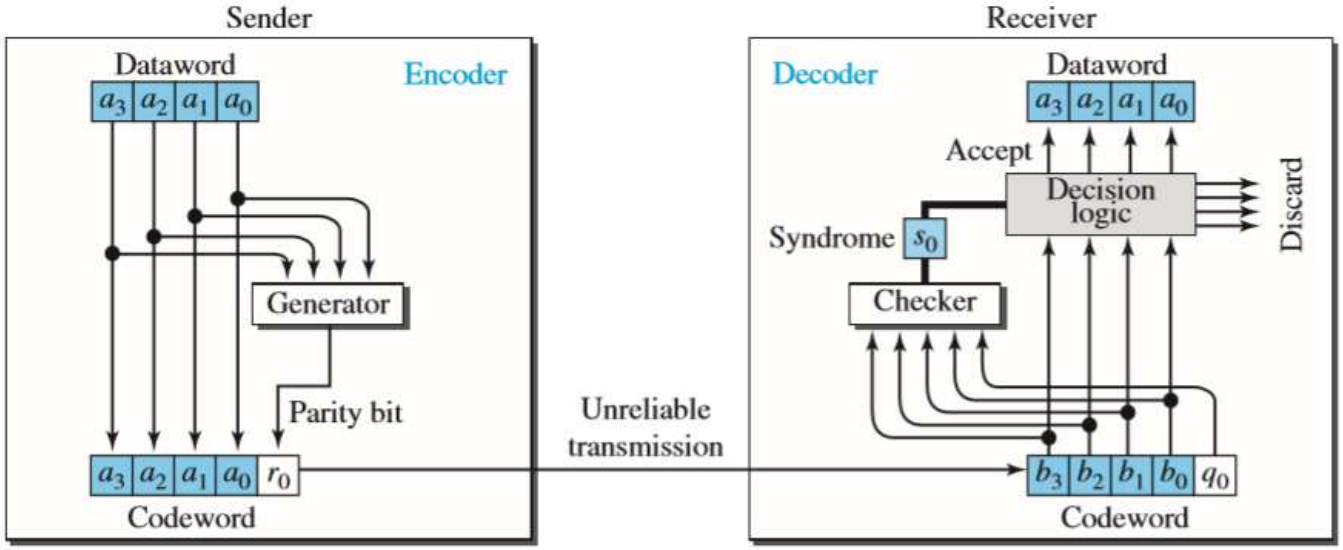


Figure 17. Encoder and Decoder for Simple Parity-Check Code

Cyclic Codes

Cyclic Codes

- In a cyclic code, if a codeword is cyclically shifted (rotated), the result is another codeword.
- For example, if 1011000 is a codeword and it is cyclically left-shift, then 0110001 is also a codeword.
- In this case, if it is called the bits in the first word a0 to a6, and the bits in the second word b0 to b6, it can shift the bits by using the following:

b1 = a0 b2 = a1 b3 = a2 b4 = a3 b5 = a4 b6 = a5 b0 = a6

Cyclic Redundancy Check

- A subset of cyclic codes called the cyclic redundancy check (CRC), which is used in networks such as LANs and WANs.

Cyclic Codes (Continue)

Table 3. A CRC code with C(7, 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

Cyclic Codes (Continue)

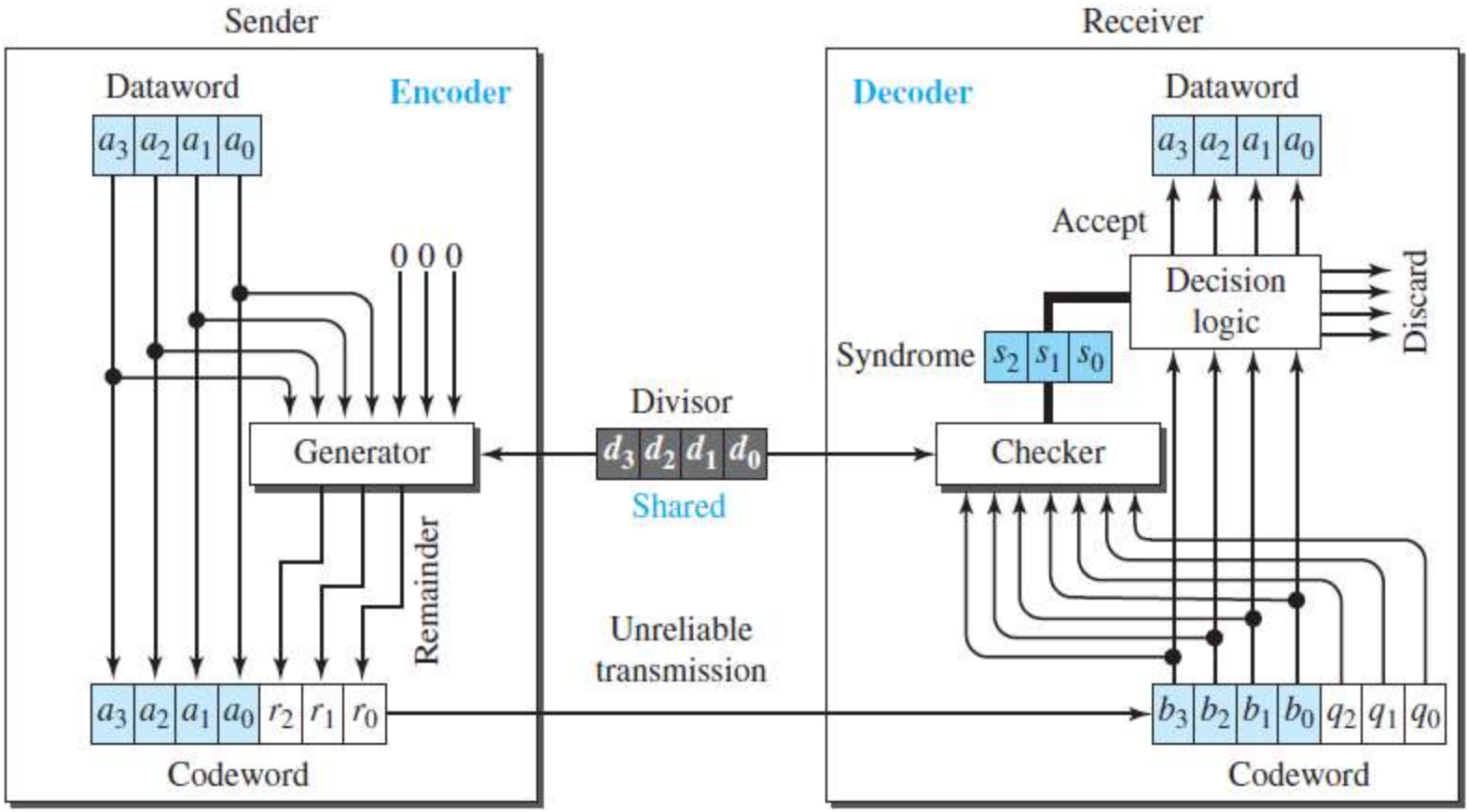


Figure 18. CRC Encoder and Decoder

Cyclic Codes (Continue)

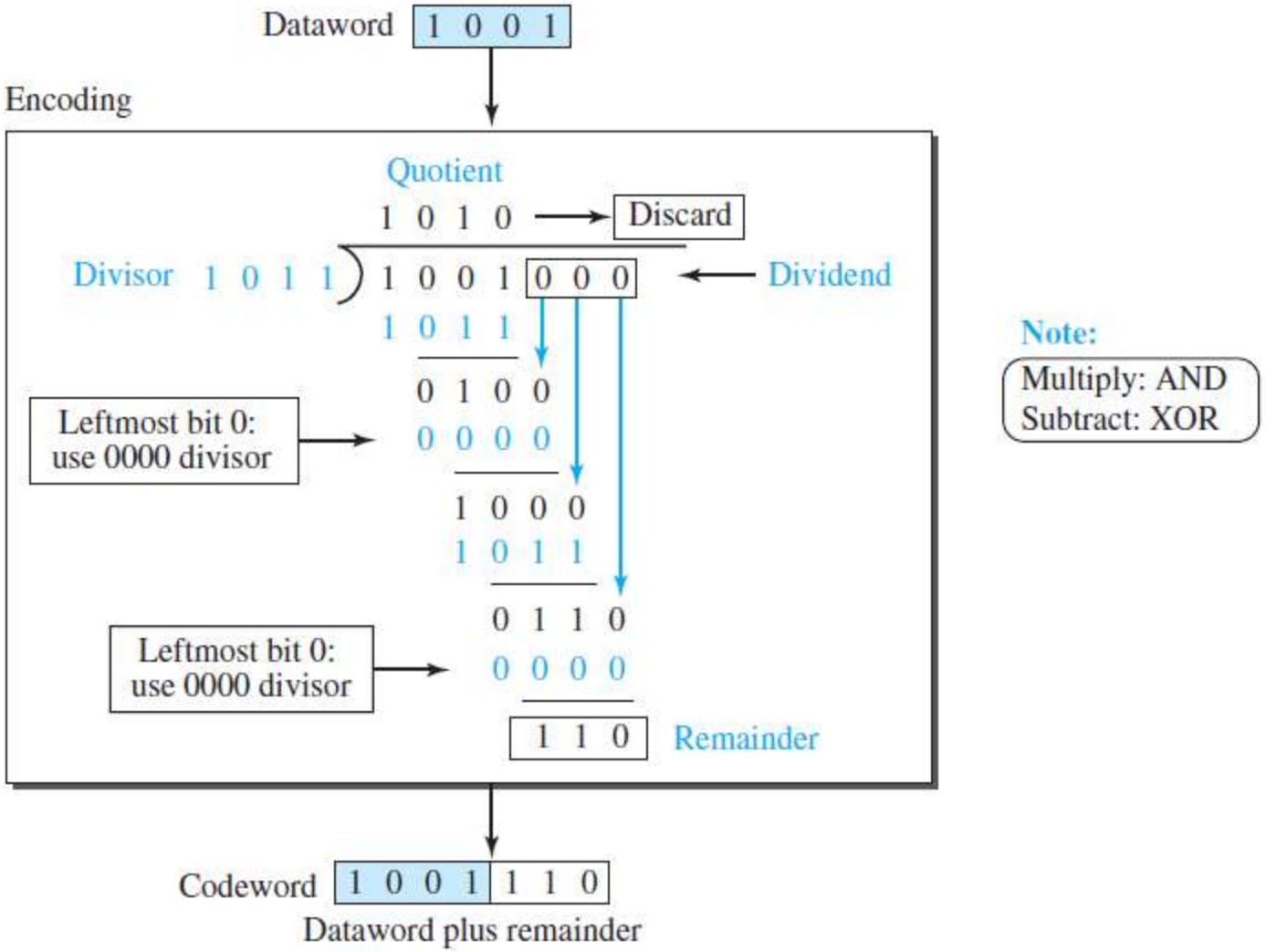


Figure 19. Division in CRC Encoder

Cyclic Codes (Continue)

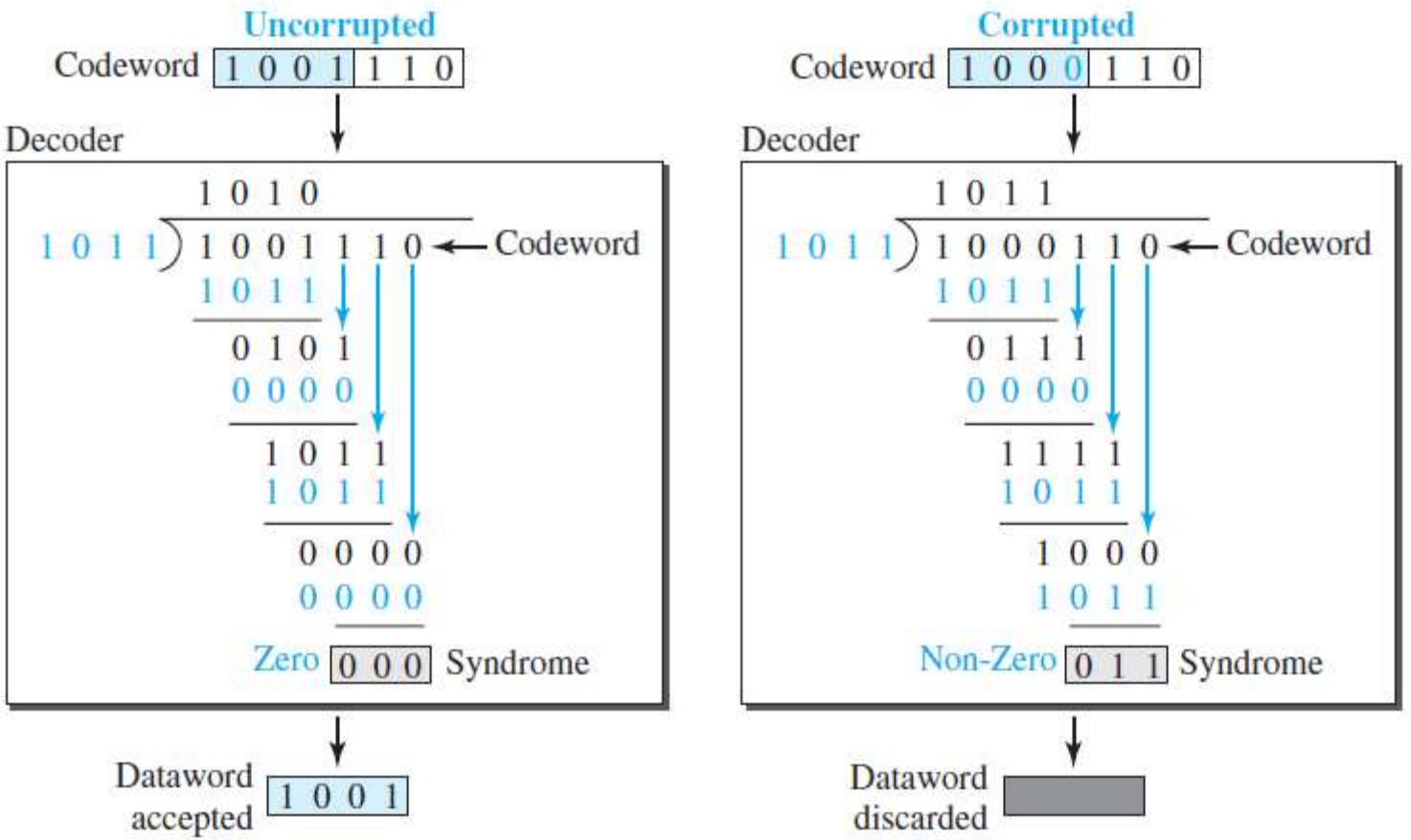


Figure 20. Division in the CRC Decoder for Two Cases

Cyclic Codes (Continue)

Polynomials

- A pattern of 0s and 1s can be represented as a polynomial with coefficients of 0 and 1.
- Figure 21 shows one immediate benefit; a 7-bit pattern can be replaced by three terms.

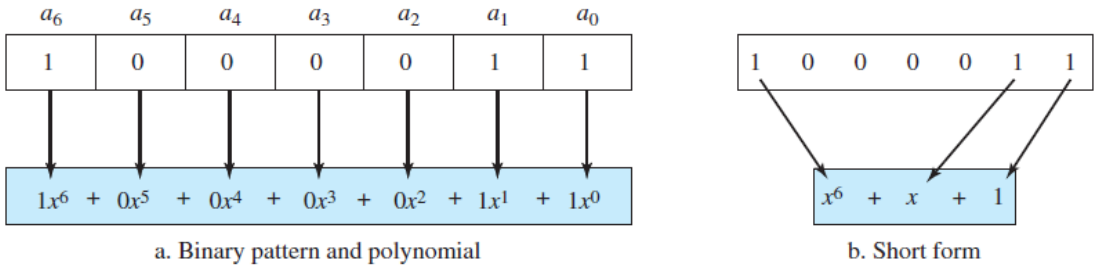


Figure 21. A Polynomial to Represent a Binary Word

- Types of polynominals are degree of a polynomial, adding and subtracting polynomials, multiplying or dividing terms, multiplying two polynomials, dividing one polynomial by another, and shifting.

Cyclic Codes (Continue)

Cyclic Code Encoder Using Polynomials

- The divisor in a cyclic code is normally called the generator polynomial or simply the generator.

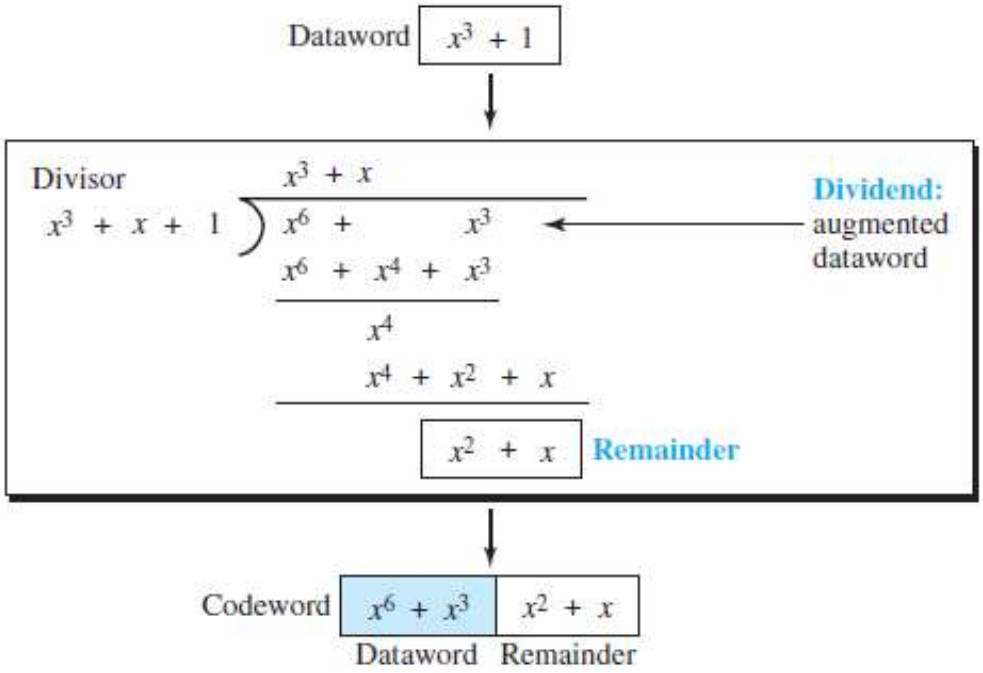


Figure 22. CRC Division using Polynomials

Cyclic Codes (Continue)

Cyclic Code Analysis

- A cyclic code to find its capabilities by using polynomials is analyzed.
- It defines the following, where f(x) is a polynomial with binary coefficients.

Dataword: d(x) Codeword: c(x) Generator: g(x) Syndrome: s(x) Error: e(x)

- In a cyclic code,

1. If s(x)!0, one or more bits is corrupted.
2. If s(x) = 0, either
   a. No bit is corrupted, or
   b. Some bits are corrupted, but the decoder failed to detect them.

Checksum

- Checksum is an error-detecting technique that can be applied to a message of any length.
- At the source, the message is first divided into m-bit units.
- The generator then creates an extra m-bit unit called the checksum, which is sent with the message.
- At the destination, the checker creates a new checksum from the combination of the message and sent checksum.
- If the new checksum is all 0s, the message is accepted; otherwise, the message is discarded as shown in figure 23.

Checksum (Continue)

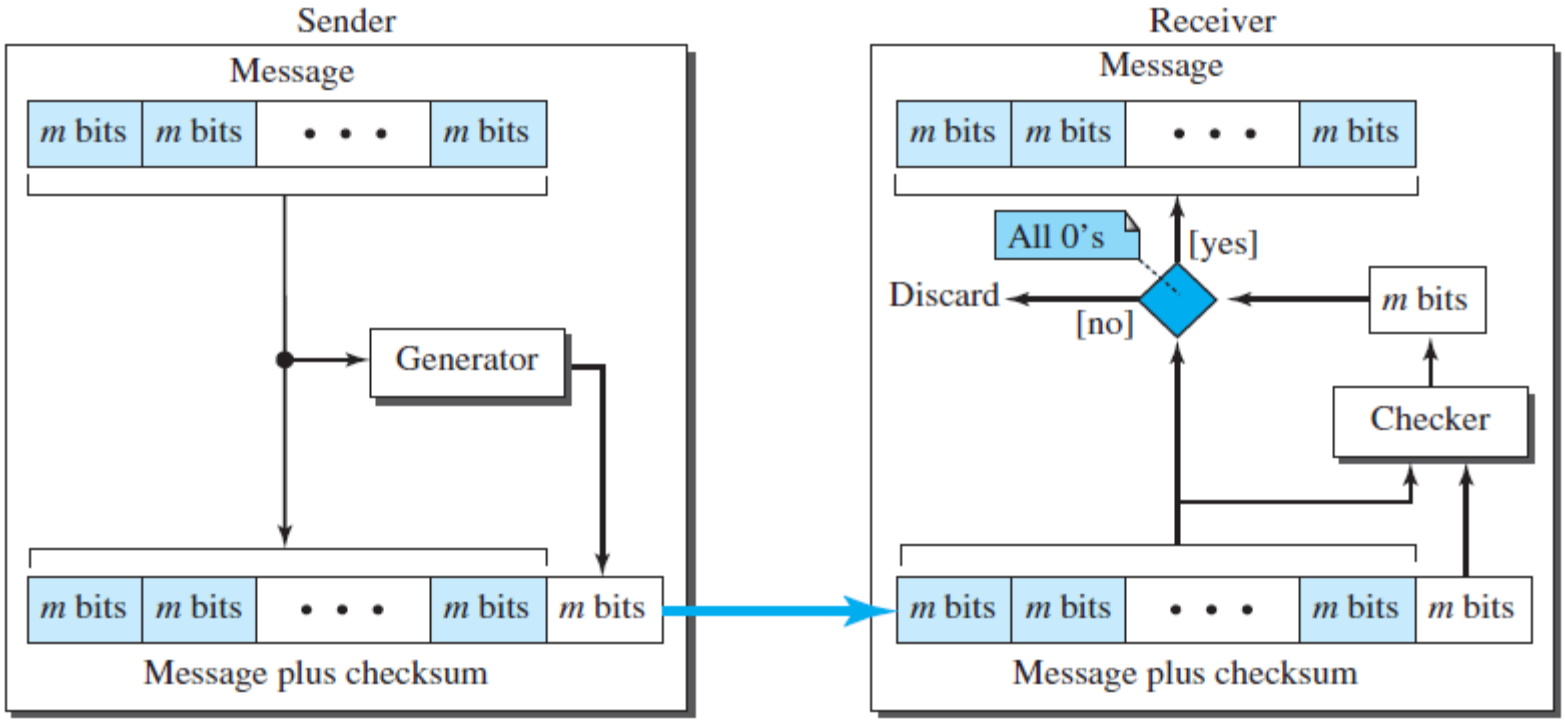


Figure 23. Checksum

Checksum (Continue)

Example 3

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. The 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 is not accepted.

Checksum (Continue)

Example 4

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.

Solution

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

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

Checksum (Continue)

Example 5

Let us use the idea of the checksum in Example 3. 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. The sender complements 15 to get 0. This shows that data have not been corrupted. figure 9 shows the process.

Solution

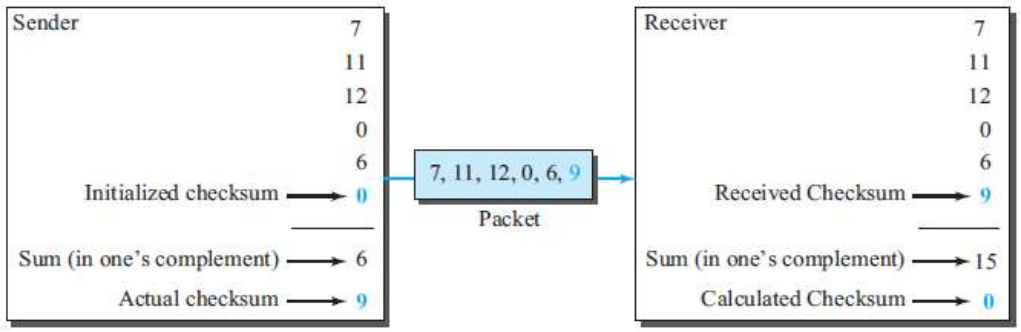


Figure 24. Checksum

Checksum (Continue)

Internet Checksum

- Traditionally, the Internet has used a 16-bit checksum.
- The sender and the receiver follow the steps depicted in Table 4.
- The sender or the receiver uses five steps.
- It can use the flow diagram of figure 25 to show the algorithm for calculation of the checksum.

Table 4: Procedure to Calculate the Traditional Checksum

Sender	*Receiver*
1. The message is divided into 16-bit words.	1. The message and the checksum are 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.

Checksum (Continue)

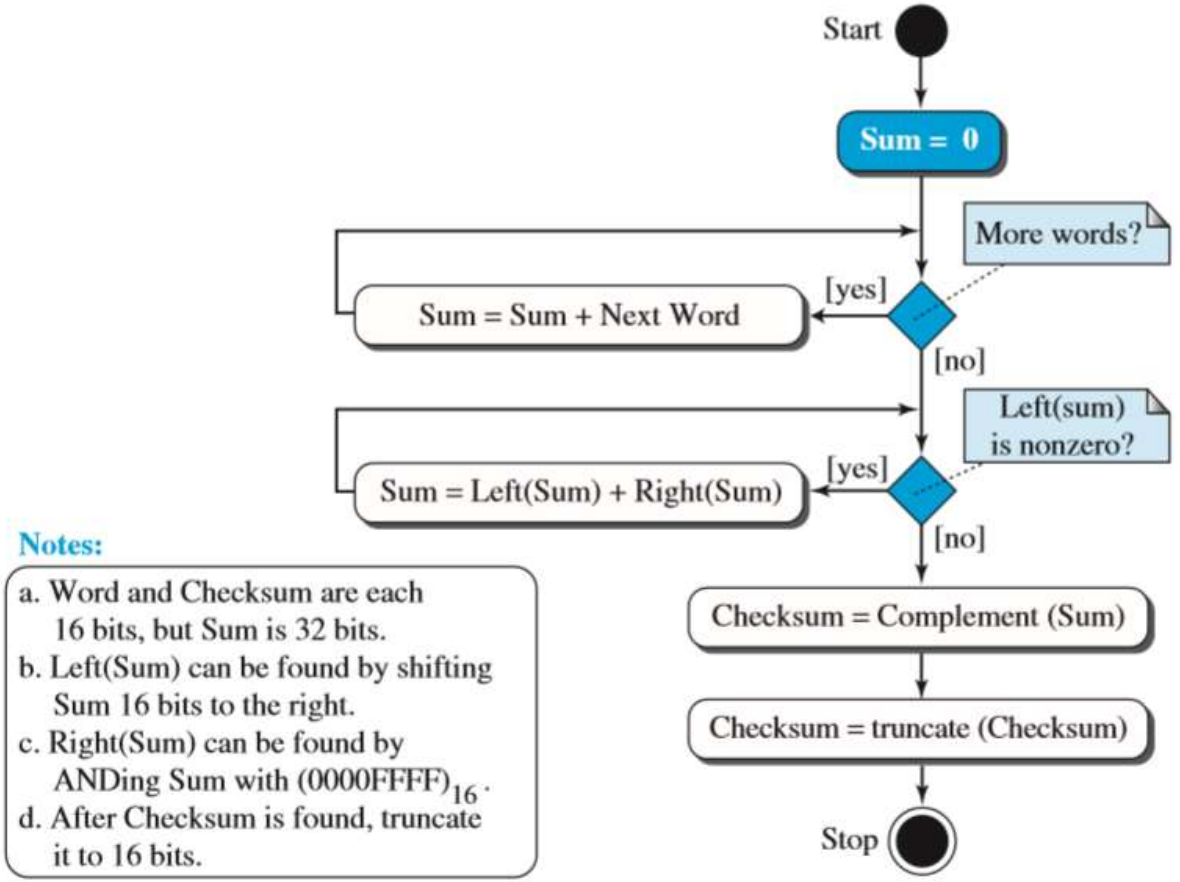


Figure 25. Algorithm to calculate a traditional checksum

Forward Error Correction

- Error detection and retransmission are discussed in the previous sections.
- However, retransmission of corrupted and lost packets is not useful for real-time multimedia transmission because it creates an unacceptable delay in reproducing: it is needed to wait until the lost or corrupted packet is resent.
- It is needed to correct the error or reproduce the packet immediately.
- Several schemes have been designed and used in this case that are collectively referred to as forward error correction (FEC) techniques.

Using Hamming Distance

- To give an example, consider the famous BCH code.
- In this code, if data is 99 bits, it is needed to send 255 bits (extra 156 bits) to correct just 23 possible bit errors.

Forward Error Correction (Continue)

- Most of the time it cannot afford such a redundancy.
- Some examples of how to calculate the required bits in the practice set are given.
- Figure 26 shows the geometrical representation of this concept.

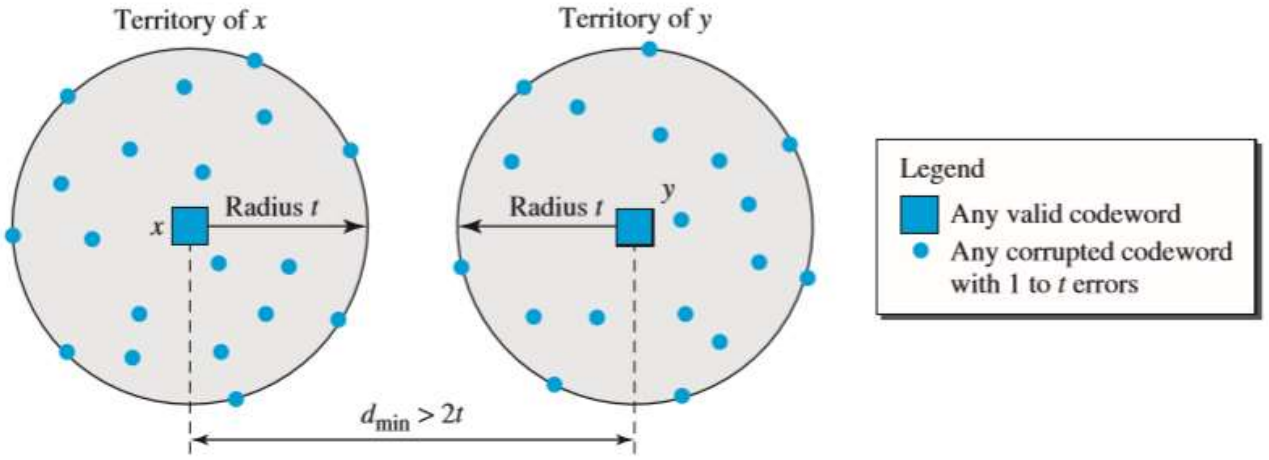


Figure 26. Hamming Distance for Error Correction Checksum

Forward Error Correction (Continue)

Using XOR

- Another recommendation is to use the property of the exclusive OR operation as shown below.

$$R = P_1 \oplus P_2 \oplus \ldots \oplus P_i \oplus \ldots \oplus P_N \longrightarrow P_i = P_1 \oplus P_2 \oplus \ldots \oplus \mathrm{R} \oplus \ldots \oplus P_N$$

- If it applies the exclusive OR operation on N data items (P1 to PN), it can recreate any of the data items by exclusive-ORing all of the items, replacing the one to be created by the result of the previous operation (R).
- This means that it can divide a packet into N chunks, create the exclusive OR of all the chunks and send N + 1 chunks.
- If any chunk is lost or corrupted, it can be created at the receiver site.
- Now the question is what should the value of N be.
- If N = 4, it means that it is needed to send 25 percent extra data and be able to correct the data if only one out of four chunks is lost.

Forward Error Correction (Continue)

Chunk Interleaving

- Another way to achieve FEC in multimedia is to allow some small chunks to be missing at the receiver.
- Figure 27 shows that it can divide each packet into 5 chunks (normally the number is much larger).
- It can then create data chunk by chunk (horizontally), but combine the chunks into packets vertically.
- In this case, each packet sent carries a chunk from several original packets.
- If the packet is lost, it misses only one chunk in each packet, which is normally acceptable in multimedia communication.

Forward Error Correction (Continue)

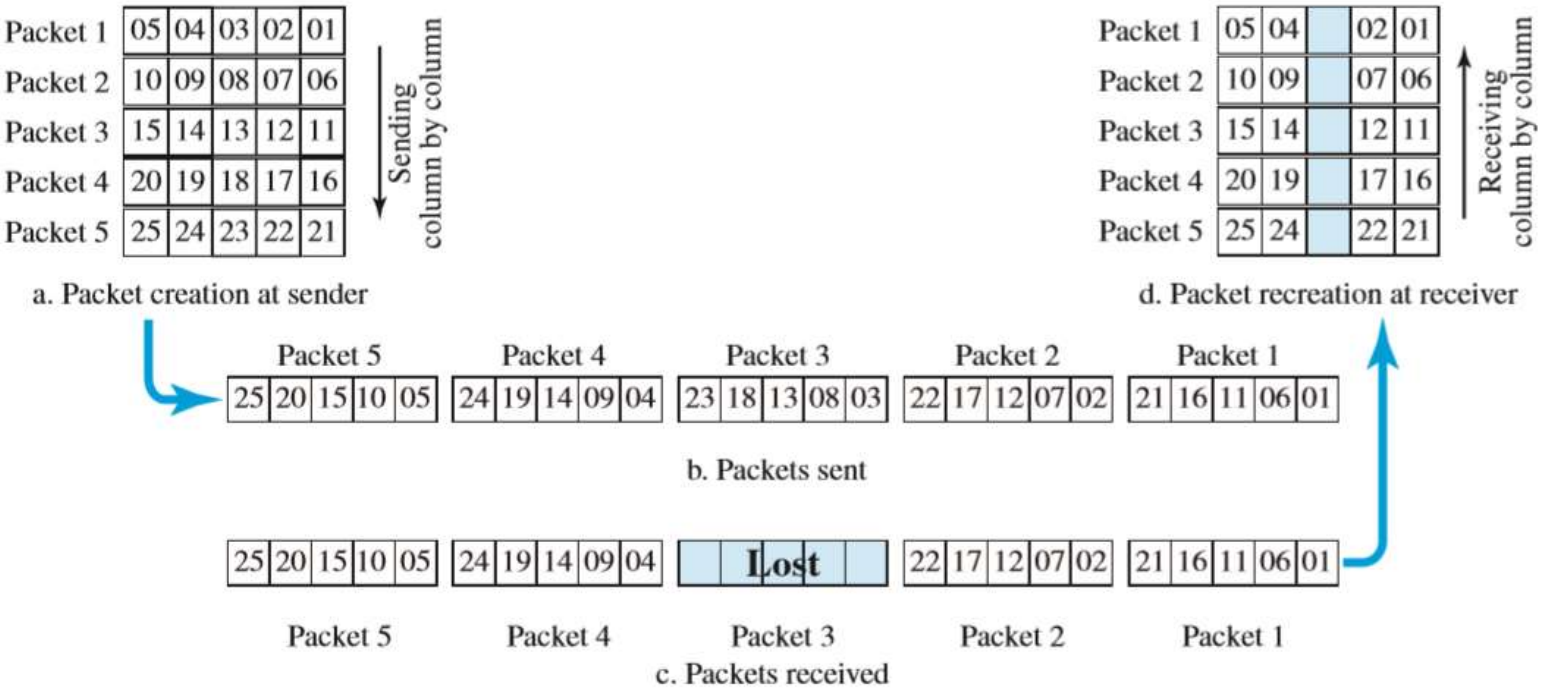


Figure 27. Interleaving

Topic 3: Data Link Control

Data Link Control (DLC) Services

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

Framing

- Framing in the data-link layer separates a message from one source to a destination by adding a sender address and a destination address.
- The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt.
- One reason is that a frame can be very large, making flow and error control very inefficient.
- When a message is carried in one very large frame, even a single-bit error would require the retransmission of the whole frame.
- When a message is divided into smaller frames, a single-bit error affects only that small frame.

DLC Services (Continue)

Character-Oriented Framing

- In character-oriented (or byte-oriented) framing, data to be carried are 8-bit characters from a coding system such as ASCII.
- The header, which normally carries the source and destination addresses, and the trailer, which carries error detection redundant bits, are also multiples of 8 bits.
- To separate one frame from the next, an 8-bit (1-byte) flag is added at the beginning and the end of a frame.
- Figure 28 shows the format of a frame in a character-oriented protocol.

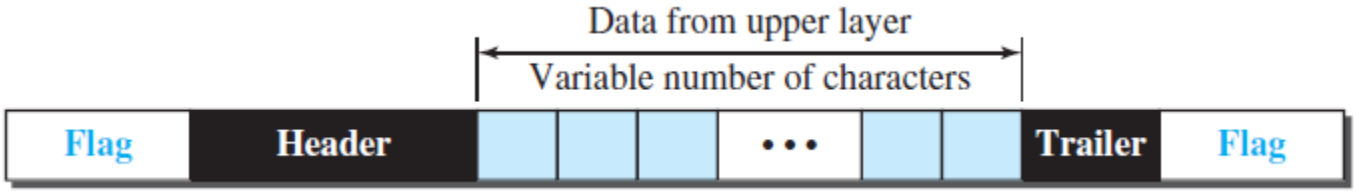


Figure 28. A frame in a character-oriented protocol

DLC Services (Continue)

- In byte stuffing (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag.
- The data section is stuffed with an extra byte.
- This byte is usually called the escape character (ESC).
- Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not as a delimiting flag.
- Figure 29 shows the situation.

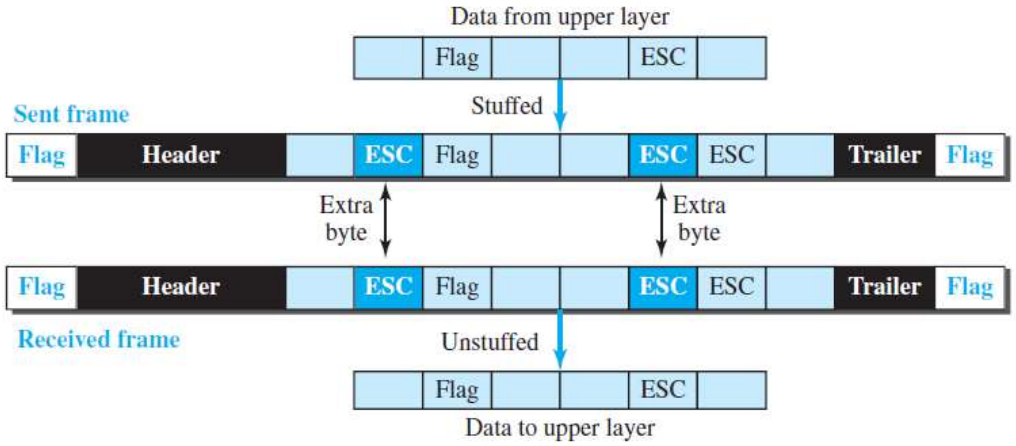


Figure 29. Byte stuffing and unstuffing

DLC Services (Continue)

Bit-Oriented Framing

- In bit-oriented framing, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on.
- However, in addition to headers (and possible trailers), it needs a delimiter to separate one frame from the other.
- Most protocols use a special 8-bit pattern flag, 01111110, as the delimiter to define the beginning and the end of the frame, as shown in figure 30.

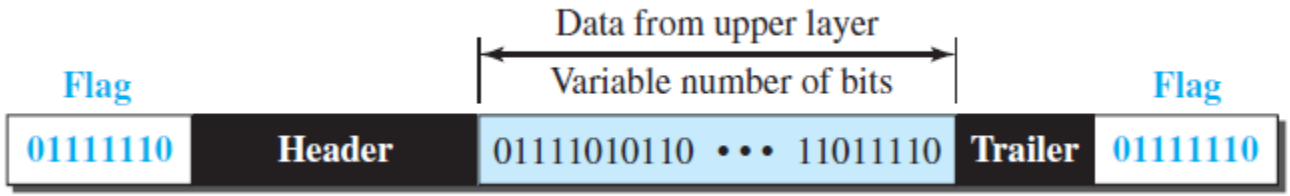


Figure 30. A Frame in a Bit-Oriented Protocol

DLC Services (Continue)

- In bit stuffing, if a 0 and five consecutive 1 bits are encountered, an extra 0 is added.
- This extra stuffed bit is eventually removed from the data by the receiver.

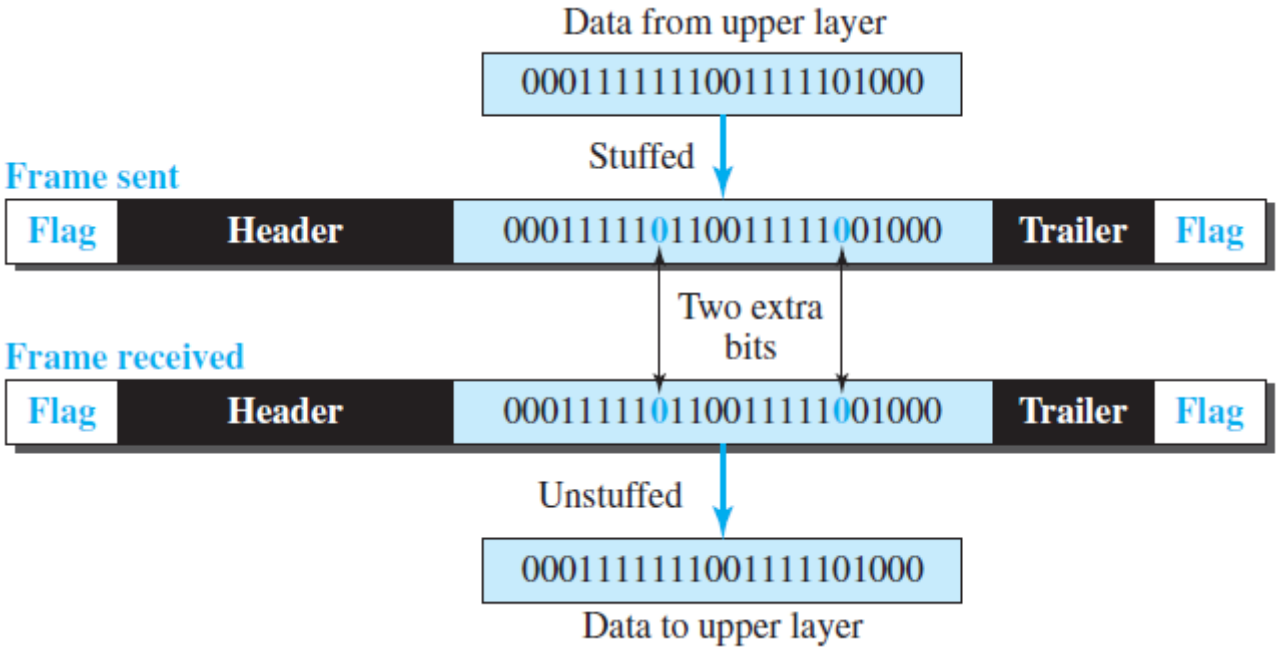


Figure 31. Bit Stuffing and Unstuffing

DLC Services (Continue)

Flow and Error Control

- One of the responsibilities of the data-link control sublayer is flow and error control at the data-link layer.
- Flow control is a technique that allows two stations working at different speeds to communicate with each other.
- In data link layer, flow control restricts the number of frames the sender can send before it waits for an acknowledgment from the receiver.

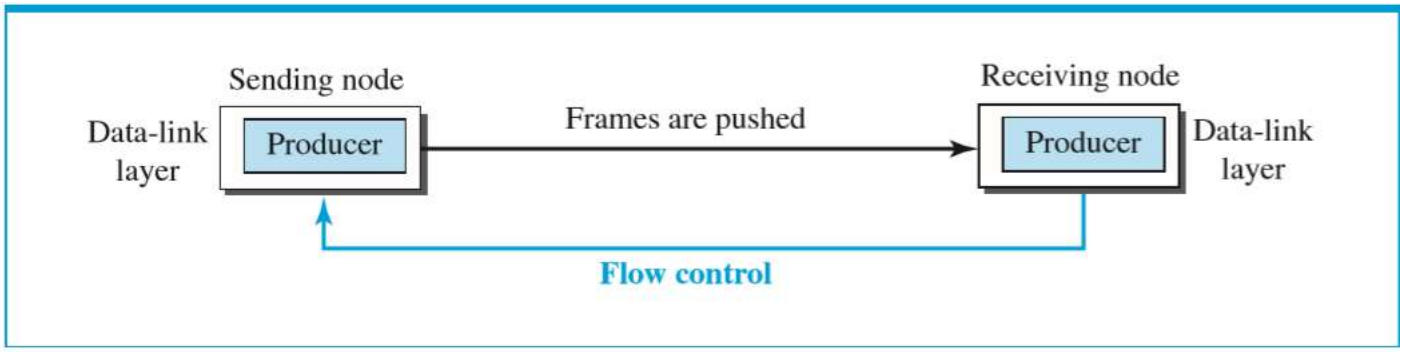


Figure 32. Flow Control at the Data-Link Layer

DLC Services (Continue)

Buffer

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

Error Control

- Error control at the data-link layer is normally very simple and implemented using one of the following two methods.

1. In the first method, if the frame is corrupted, it is silently discarded; if it is not corrupted, the packet is delivered to the network layer. This method is used mostly in wired LANs such as Ethernet.
2. In the second method, if the frame is corrupted, it is silently discarded; if it is not corrupted, an acknowledgment is sent to the sender.

DLC Services (Continue)

Connectionless and Connection-Oriented

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

Connectionless Protocol

- In a connectionless protocol, frames are sent from one node to the next without any relationship between the frames; each frame is independent.

Connection-Oriented Protocol

- In a connection-oriented protocol, a logical connection should first be established between the two nodes (setup phase).
- After all frames that are somehow related to each other are transmitted (transfer phase), the logical connection is terminated (teardown phase).

Data-link Layer Protocols

- Traditionally four protocols have been defined for the data-link layer to deal with flow and error control: Simple, Stop-and-Wait, Go-Back-N, and Selective-Repeat.
- The behavior of a data-link-layer protocol can be better shown as a finite state machine (FSM).
- An FSM is thought of as a machine with a finite number of states.
- The machine is always in one of the states until an event occurs.
- Each event is associated with two reactions: defining the list of actions to be performed and determining the next state.
- One of the states must be defined as the initial state, the state in which the machine starts when it turns on.

Data-link Layer Protocols (Continue)

- It is used rounded-corner rectangles to show states, colored text to show events, and regular black text to show actions.
- A horizontal line is used to separate the event from the actions, although later the horizontal line is replaced with a slash.
- The arrow shows the movement to the next state.

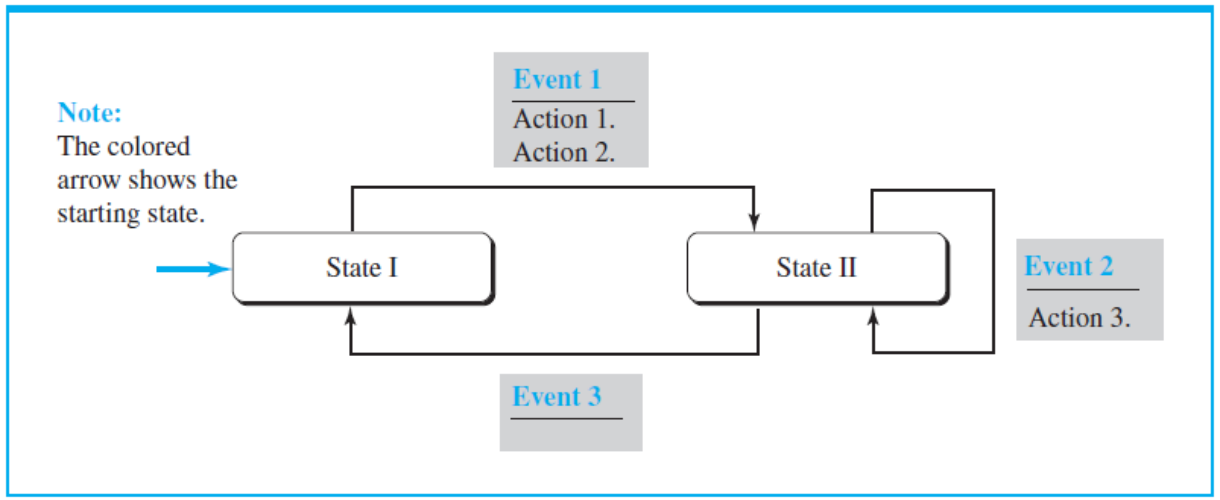


Figure 33. Connectionless and connection-oriented service represented as FSMs

Simple Protocol

- The first protocol is a simple with neither flow nor error control.

Data-link Layer Protocols (Continue)

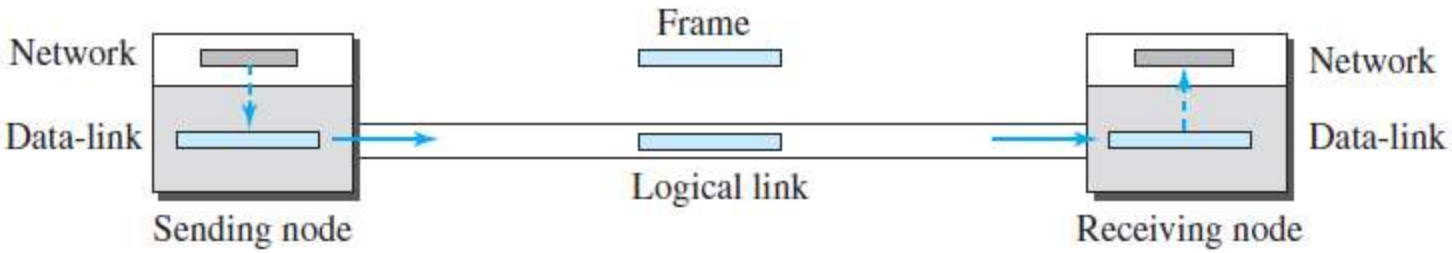


Figure 34. Simple Protocol

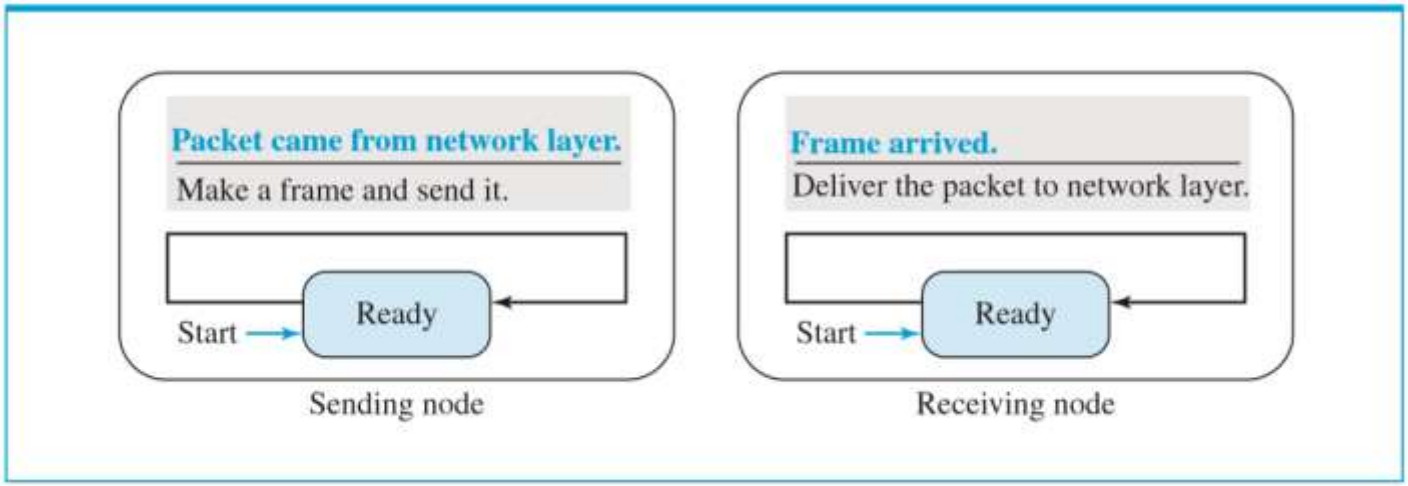


Figure 35. FSMs for the Simple Protocol

Data-link Layer Protocols (Continue)

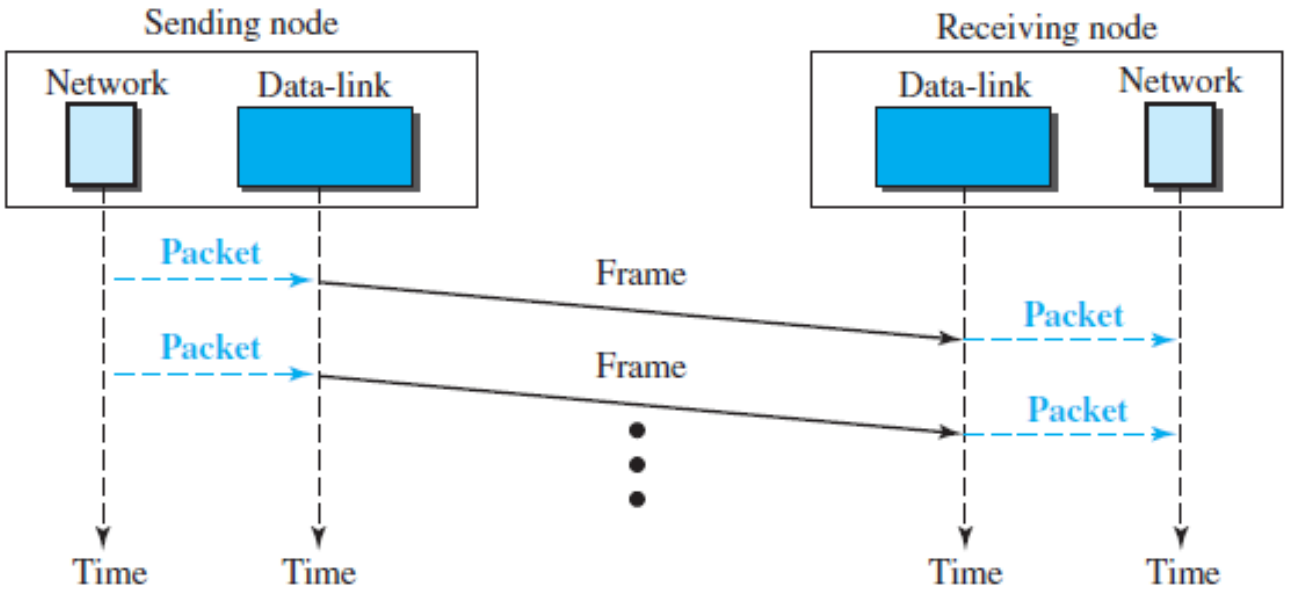


Figure 36. Flow diagram for Simple Protocol

Stop-and-Wait Protocol

- The second protocol is called the Stop-and-Wait protocol, which uses both flow and error control as shown in figure 37.

Data-link Layer Protocols (Continue)

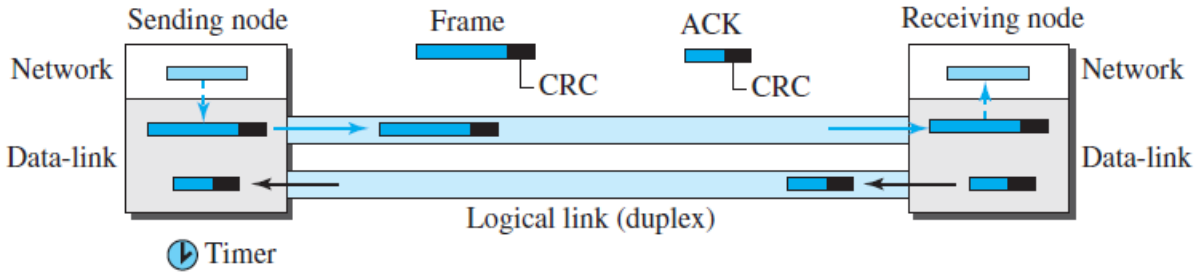


Figure 37. Stop-and-Wait Protocol

- In this protocol, the sender sends one frame at a time and waits for an acknowledgment before sending the next one.
- To detect corrupted frames, we need to add a CRC to each data frame.
- When a frame arrives at the receiver site, it is checked. If its CRC is incorrect, the frame is corrupted and silently discarded.
- Every time the sender sends a frame, it starts a timer.
- If an acknowledgment arrives before the timer expires, the timer is stopped and the sender sends the next frame (if it has one to send).
- If the timer expires, the sender resends the previous frame, assuming that the frame was either lost or corrupted.

Data-link Layer Protocols (Continue)

- This means that the sender needs to keep a copy of the frame until its acknowledgment arrives.
- When the corresponding acknowledgment arrives, the sender discards the copy and sends the next frame if it is ready.
- Figure 37 shows the outline for the Stop-and-Wait protocol.

FSMs

- The FSMs for primitive Stop-and-Wait protocol is shown in figure 38.

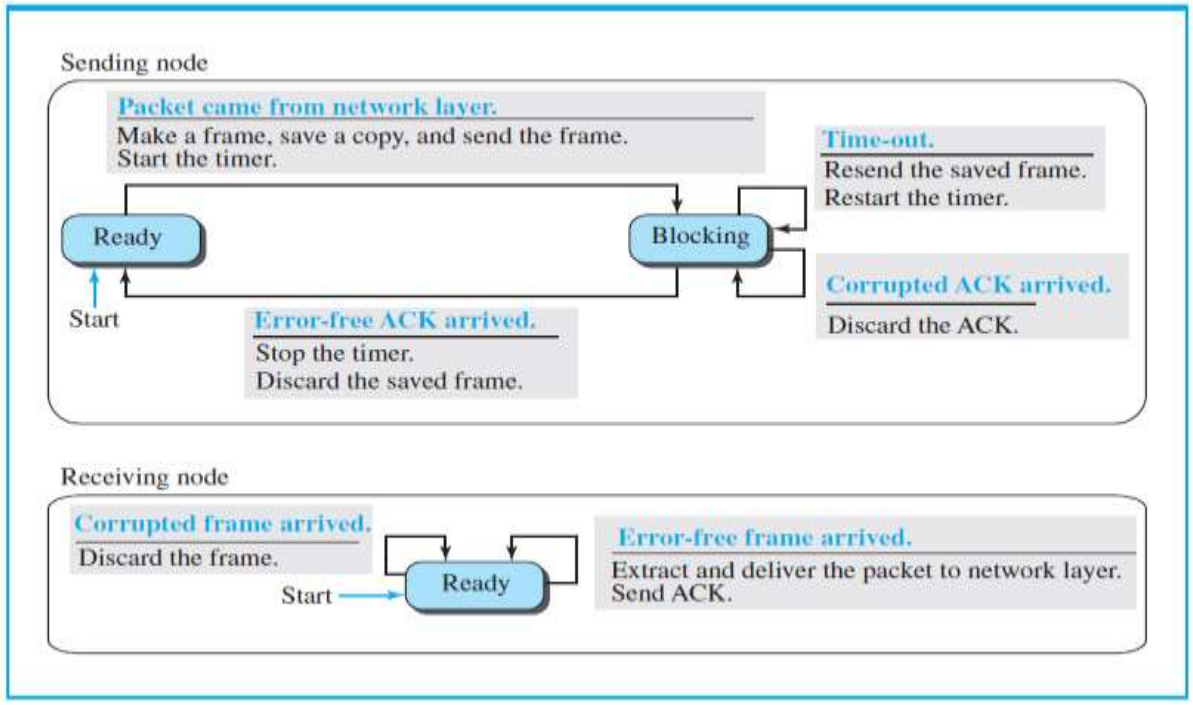


Figure 38. FSM for the Stop-and-Wait Protocol

Data-link Layer Protocols (Continue)

- It describe the sender and receiver states below.

Sender States

- The sender is initially in the ready state, but it can move between the ready and blocking state.
- **Ready State:** When the sender is in this state, it is only waiting for a packet from the network layer. If a packet comes from the network layer, the sender creates a frame, saves a copy of the frame, starts the only timer and sends the frame. The sender then moves to the blocking state.
- **Blocking State:** When the sender is in this state, three events can occur:
  a. If a time-out occurs, the sender resends the saved copy of the frame and restarts the timer.
  b. If a corrupted ACK arrives, it is discarded.
  c. If an error-free ACK arrives, the sender stops the timer and discards the saved copy of the frame. It then moves to the ready state.

Data-link Layer Protocols (Continue)

Receiver

- The receiver is always in the ready state. Two events may occur:

a. If an error-free frame arrives, the message in the frame is delivered to the network layer and an ACK is sent.

b. If a corrupted frame arrives, the frame is discarded.

Example 6

Figure 39 shows an example. The first frame is sent and acknowledged. The second frame is sent, but lost. After time-out, it is resent. The third frame is sent and acknowledged, but the acknowledgment is lost. The frame is resent. However, there is a problem with this scheme. The network layer at the receiver site receives two copies of the third packet, which is not right. In the next section, we will see how we can correct this problem using sequence numbers and acknowledgment numbers.

Data-link Layer Protocols (Continue)

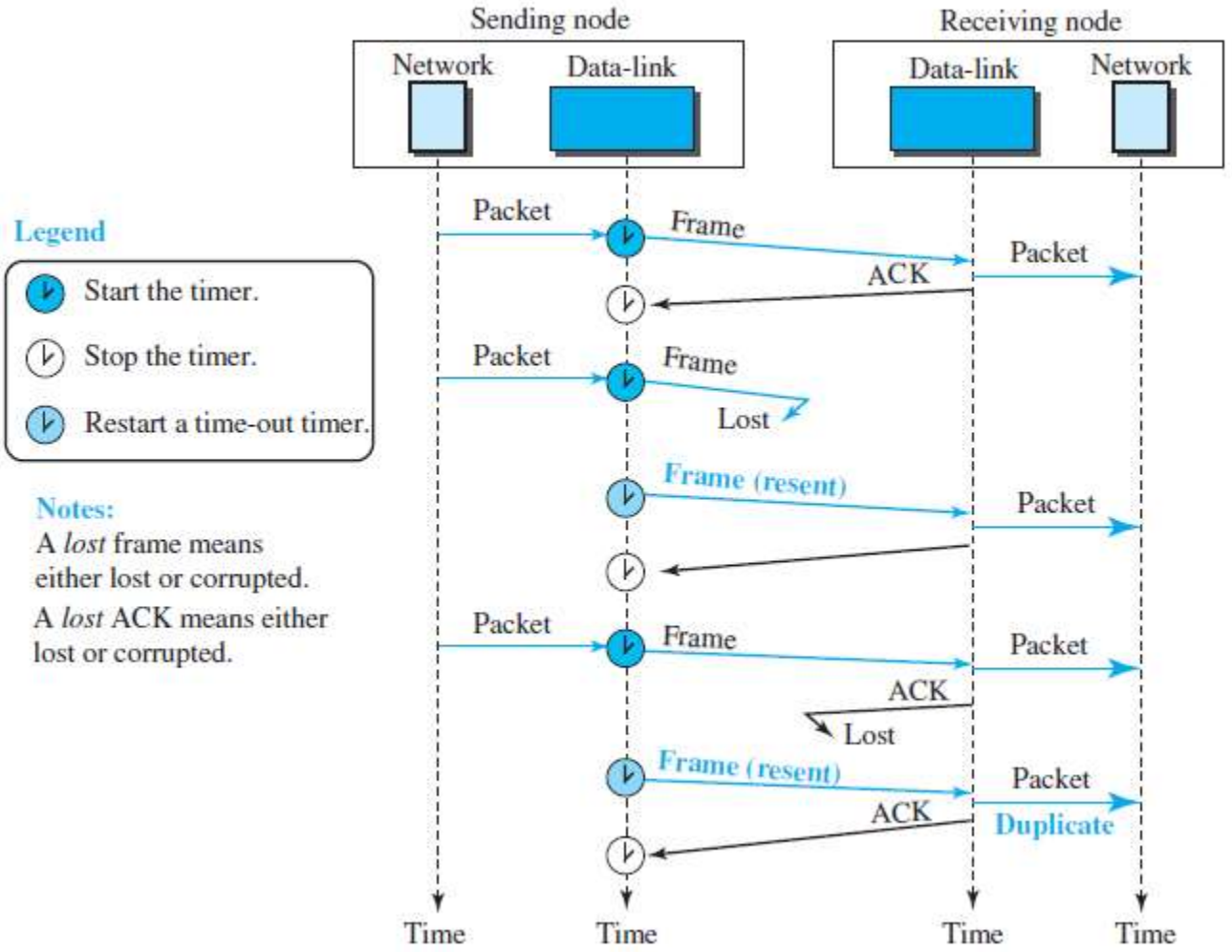


Figure 39. Flow Diagram for Example 6

Data-link Layer Protocols (Continue)

Sequence and Acknowledgment Numbers

- To avoid duplicating packets in receiver site, it is needed to add sequence numbers to the data frames and acknowledgment numbers to the ACK frames.
- The sequence numbers start with 0 (eg. 0, 1, 0, 1,…) and the acknowledgment numbers start with 1 (e.g. 1, 0, 1, 0, …).

Example 7

Figure 40 shows how adding sequence numbers and acknowledgment numbers can prevent duplicates. The first frame is sent and acknowledged. The second frame is sent, but lost. After time-out, it is resent. The third frame is sent and acknowledged, but the acknowledgment is lost. The frame is resent.

Data-link Layer Protocols (Continue)

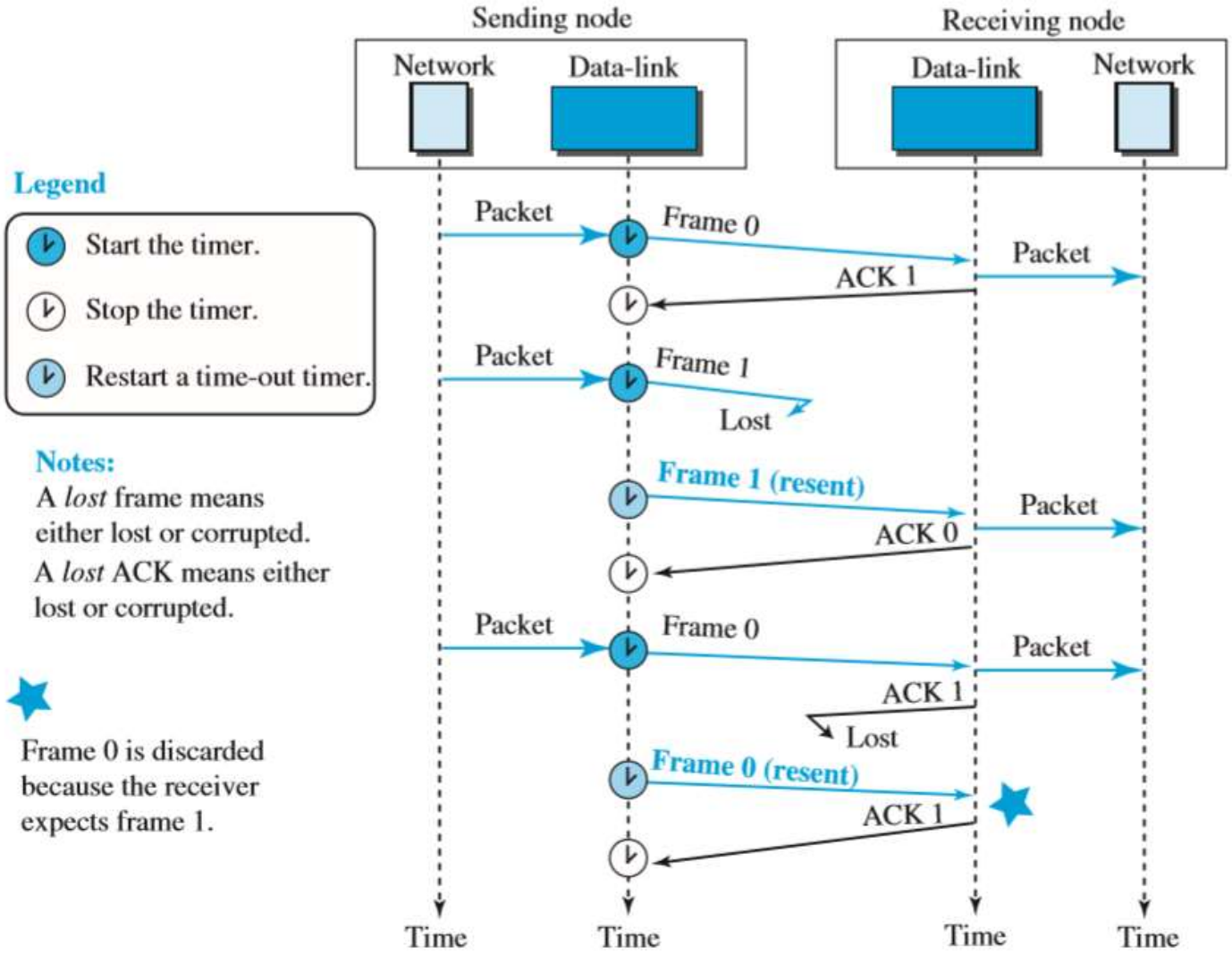


Figure 40. Flow Diagram for Example 7

HDLC

- High-level Data Link Control (HDLC) is a bit-oriented protocol for communication over point-to-point and multipoint links.

Configurations and Transfer Modes

- HDLC provides two common transfer modes that can be used in different configurations: normal response mode (NRM) and asynchronous balanced mode (ABM).
- In normal response mode (NRM), the station configuration is unbalanced.
- It has one primary station and multiple secondary stations.
- A primary station can send commands; a secondary station can only respond.
- The NRM is used for both point-to-point and multipoint links, as shown in Figure 40.

HDLC (Continue)

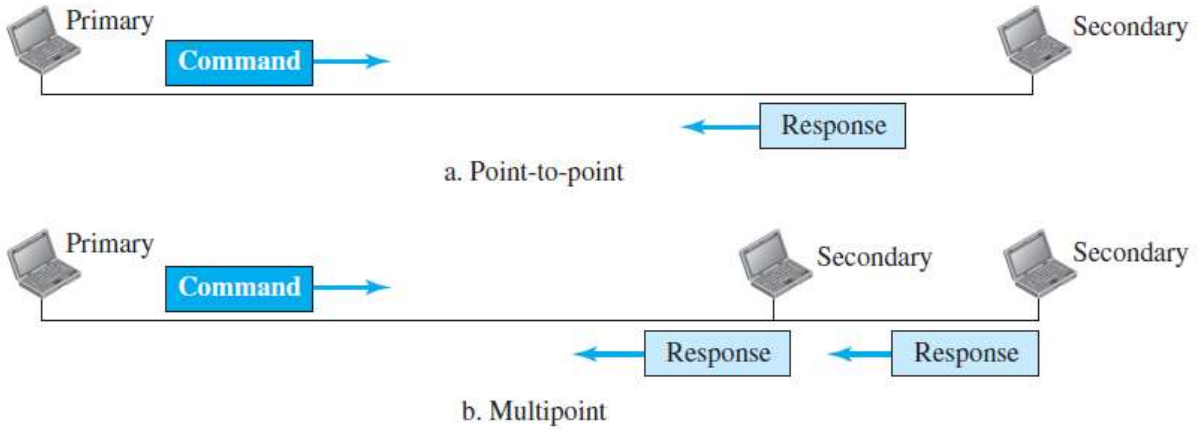


Figure 41. Normal Response Mode

- In ABM, the configuration is balanced.
- The link is point-to-point, and each station can function as a primary and secondary, as shown in figure 42.

Figure 42. Asynchronous Balanced Mode

HDLC (Continue)

Framing

- To provide the flexibility necessary to support all the options possible in the modes and configurations just described, HDLC defines three types of frames: information frames (I-frames), supervisory frames (S-frames), and unnumbered frames (U-frames).
- S-frames are used only to transport control information.
- U-frames are reserved for system management.
- Information carried by U-frames is intended for managing the link itself.
- Each frame in HDLC may contain up to six fields, as shown in Figure: a beginning flag field, an address field, a control field, an information field, a frame check sequence (FCS) field, and an ending flag field.

-In multiple-frame transmissions, the ending flag of one frame can serve as the beginning flag of the next frame.

HDLC (Continue)

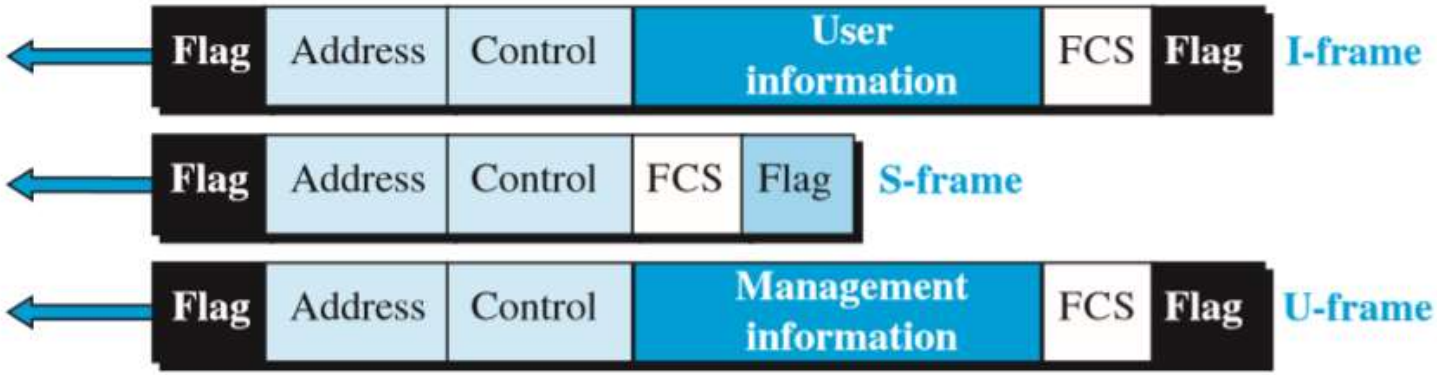


Figure 43. HDLC Frames

Point-to-point Protocol (PPP)

- In computer networking, Point-to-Point Protocol (PPP) is a data link layer (layer 2) communications protocol used to establish a direct connection between two nodes.
- It supports authentication.
- PPP defines the format of the frame to be exchanged between devices.
- It also defines how two devices can negotiate the establishment of the link and exchange of data.
- It defines how network layer data are encapsulated in the data link frame.
- It defines how two devices can authenticate each other.
- Three sets of protocols are defined to make PPP powerful: the link control protocol (LCP), two authentication protocols (password authentication protocol and challenge handshake authentication protocol) and network control protocol.

Point-to-point Protocol (Continue)

Link Control Protocol

- The Link Control Protocol (LCP) is responsible for establishing, maintaining, configuring, and terminating links.
- All LCP packets are carried in the payload field of the PPP frame with the protocol field set to C021 in hexadecimal as depicted in figure 44.

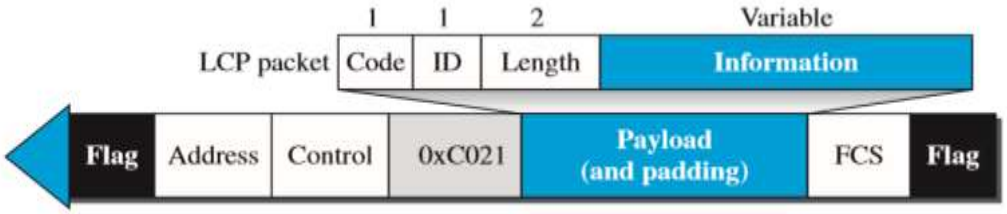


Figure 44. LCP packet Encapsulated in a Frame

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 5: LCP Packets

Point-to-point Protocol (Continue)

Authentication Protocols

- Authentication means validating the identity of a user who needs to access a set of resources.
- PPP has created two protocols for authentication: Password Authentication Protocol and Challenge Handshake Authentication Protocol.

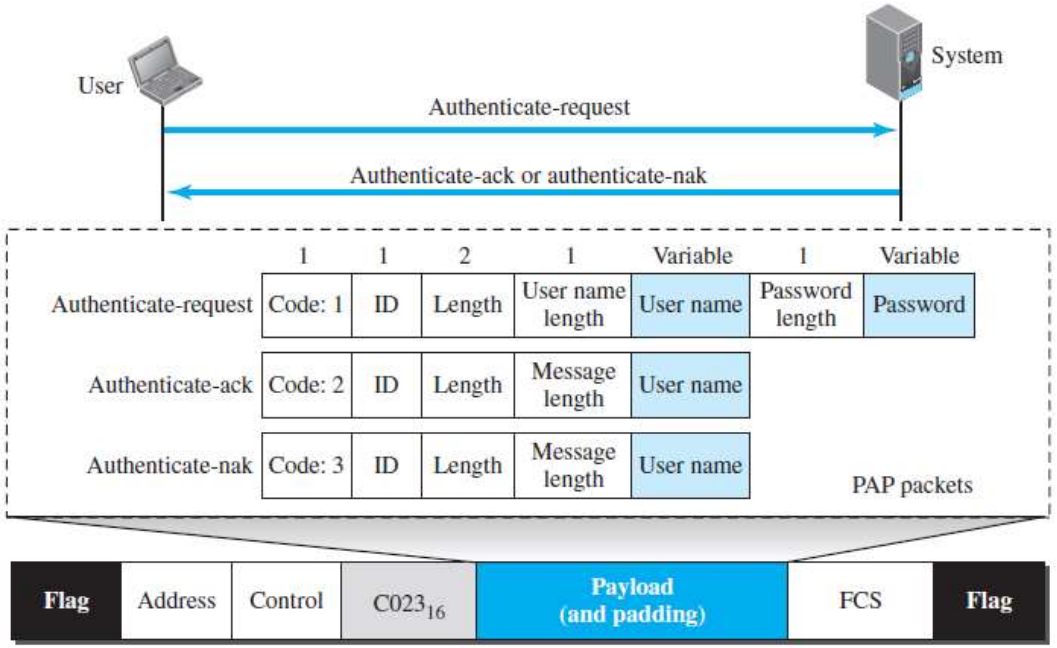


Figure 45. PAP Packets Encapsulated in a PPP Frame

Point-to-point Protocol (Continue)

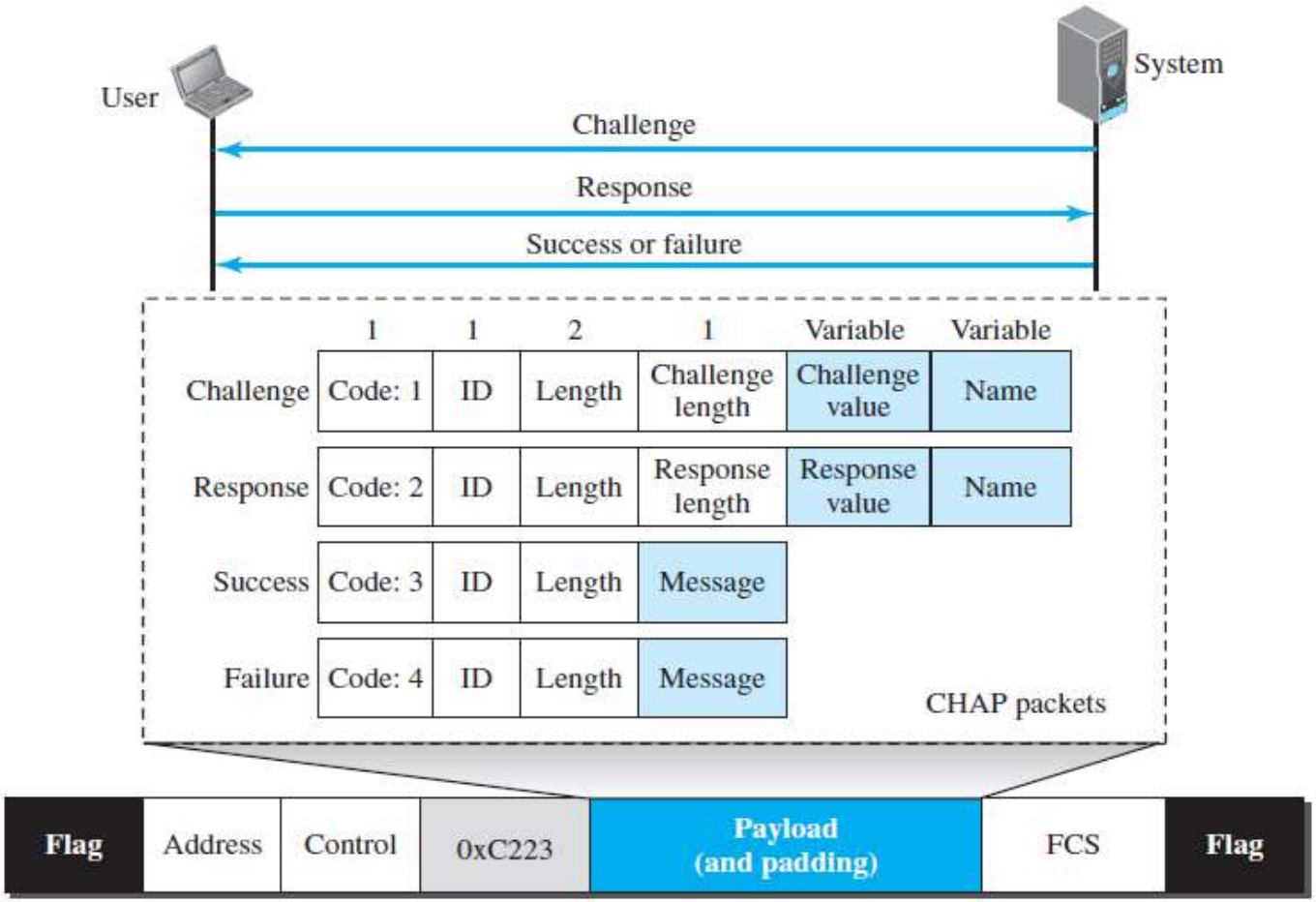


Figure 46. CHAP Packets Encapsulated in a PPP Frame

Point-to-point Protocol (Continue)

Network Control Protocol

- A set of control protocols to allow data from the network layer to be encapsulated into a PPP frame.
- One common protocol: IPCP
- This protocol configures the link used to carry IP packets in the Internet.
- The format of an IPCP packet is shown in Figure 47.

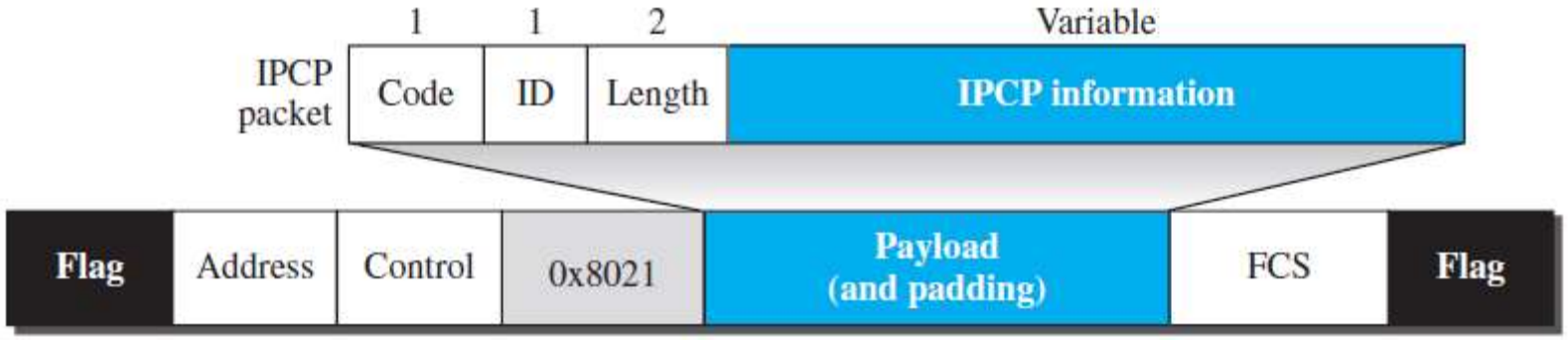


Figure 47. IPCP Packet Encapsulated in PPP Frame

Topic 4: Media Access Control

Media Access Control

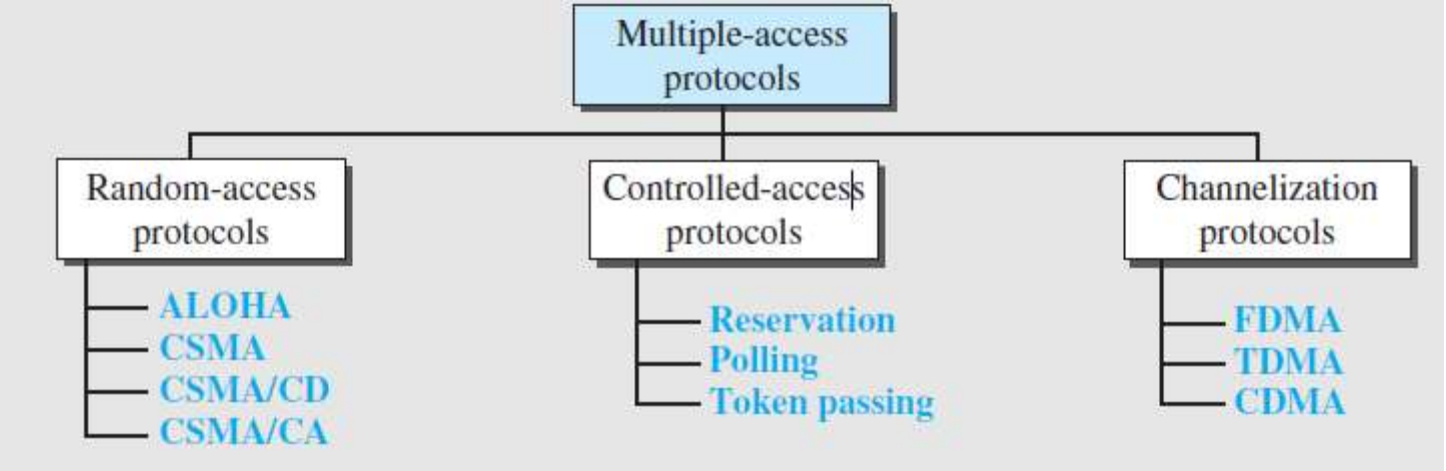


Figure 48. Taxonomy of multiple-access protocols

Random Access

ALOHA

- ALOHA is obvious that there are potential collisions in this arrangement.
- The medium is shared between the stations.
- When a station sends data, another station may attempt to do so at the same time.
- The data from the two stations collide and become garbled.
- The original ALOHA protocol is called pure ALOHA as shown in fig. 49.

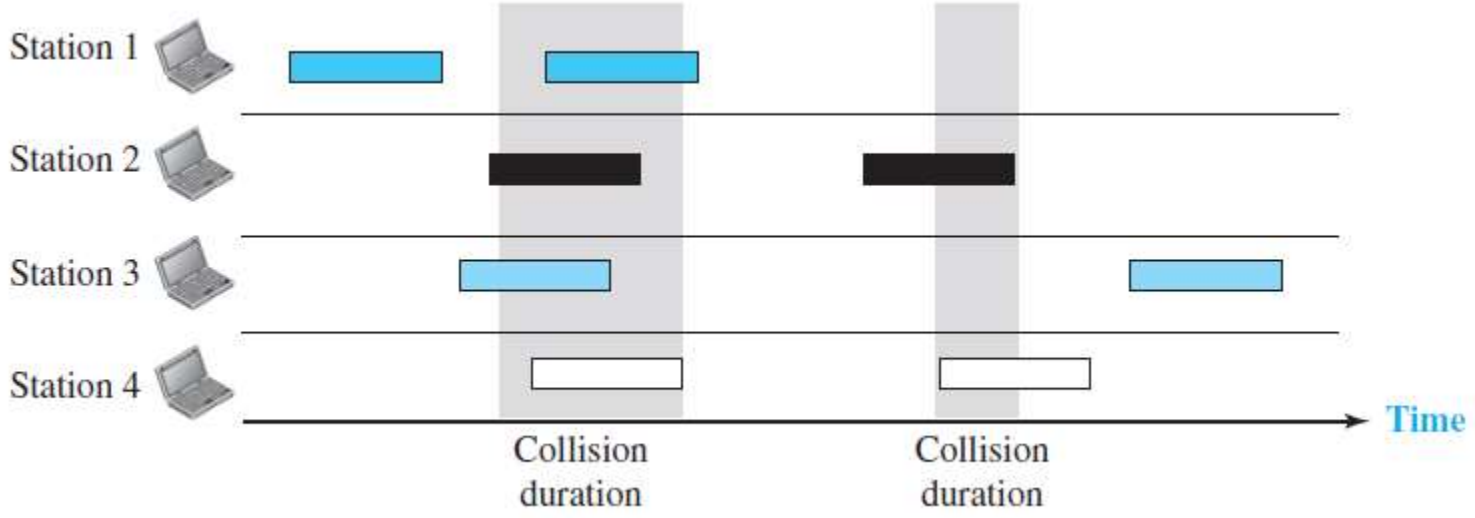


Figure 49. Frames in a Pure ALOHA Network

Random Access (Continue)

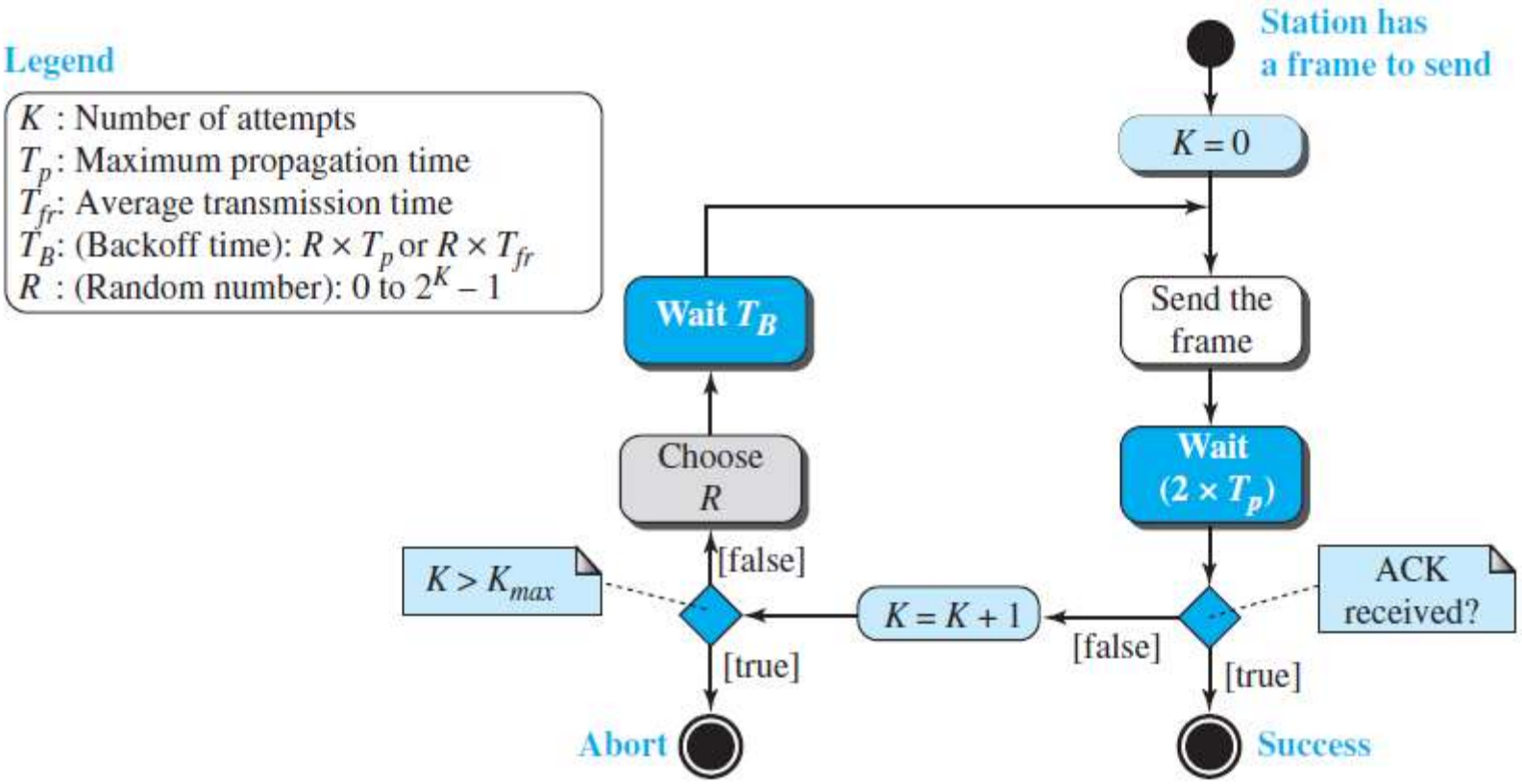


Figure 50. Procedure for pure ALOHA protocol

Random Access

- The throughput for pure ALOHA is $S = G \times e^{-2G}$.
- The maximum throughput is $S_{max} = 0.184$ when $G = (1/2)$.

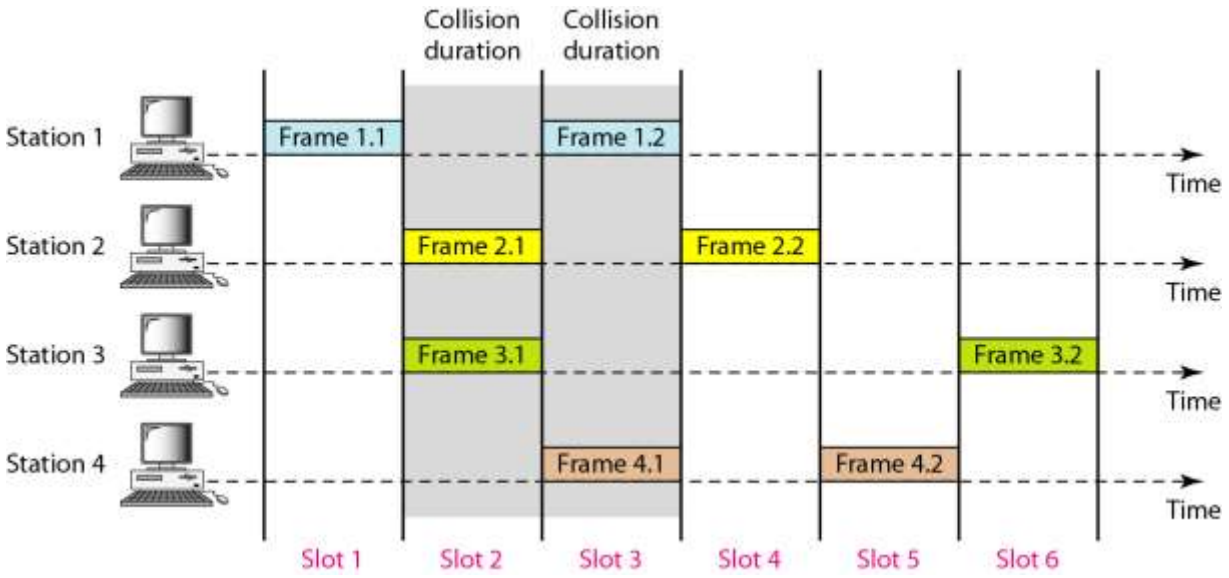


Figure 51. Frames in a slotted ALOHA network

- The throughput for slotted ALOHA is $S = G \times e^{-G}$.
- The maximum throughput is $S_{max} = 0.368$ when $G = 1$.

Random Access (Continue)

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

- CSMA/CA was invented for wireless networks.
- Collisions are avoided through the use of CSMA/CA's three strategies: the interframe space, the contention window, and acknowledgments as shown in figure 53.

Interframe Space (IFS)

- When an idle channel is found, the station does not send immediately. It waits for a period of time called the interframe space (IFS).

Contention Window

- The contention window is and amount of time divided into slots.
- A station that is ready to send chooses a random number of slots as its wait time.
- The number of slots in the window changes according to the binary exponential backoff strategy.
- One interesting point about the contention window is that the station needs to sense the channel after each time slot.

Random Access (Continue)

- However, if the station finds the channel busy, it does not restart the process; it just stops the timer and restarts it when the channel is sensed as idle.
- This gives priority to the station with the longest waiting time as depicted in figure 52.

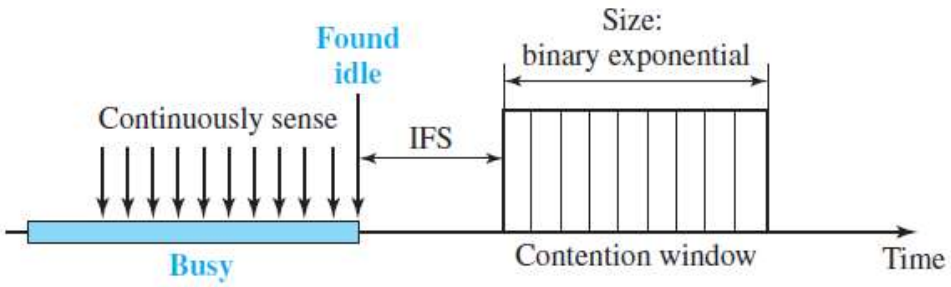


Figure 52. Contention window

Acknowledgment

- With all these precautions, there still may be a collision resulting in destroyed data.
- In addition, the data may be corrupted during the transmission.
- The positive acknowledgment and the time-out timer can help guarantee that the receiver has received the frame.

Random Access (Continue)

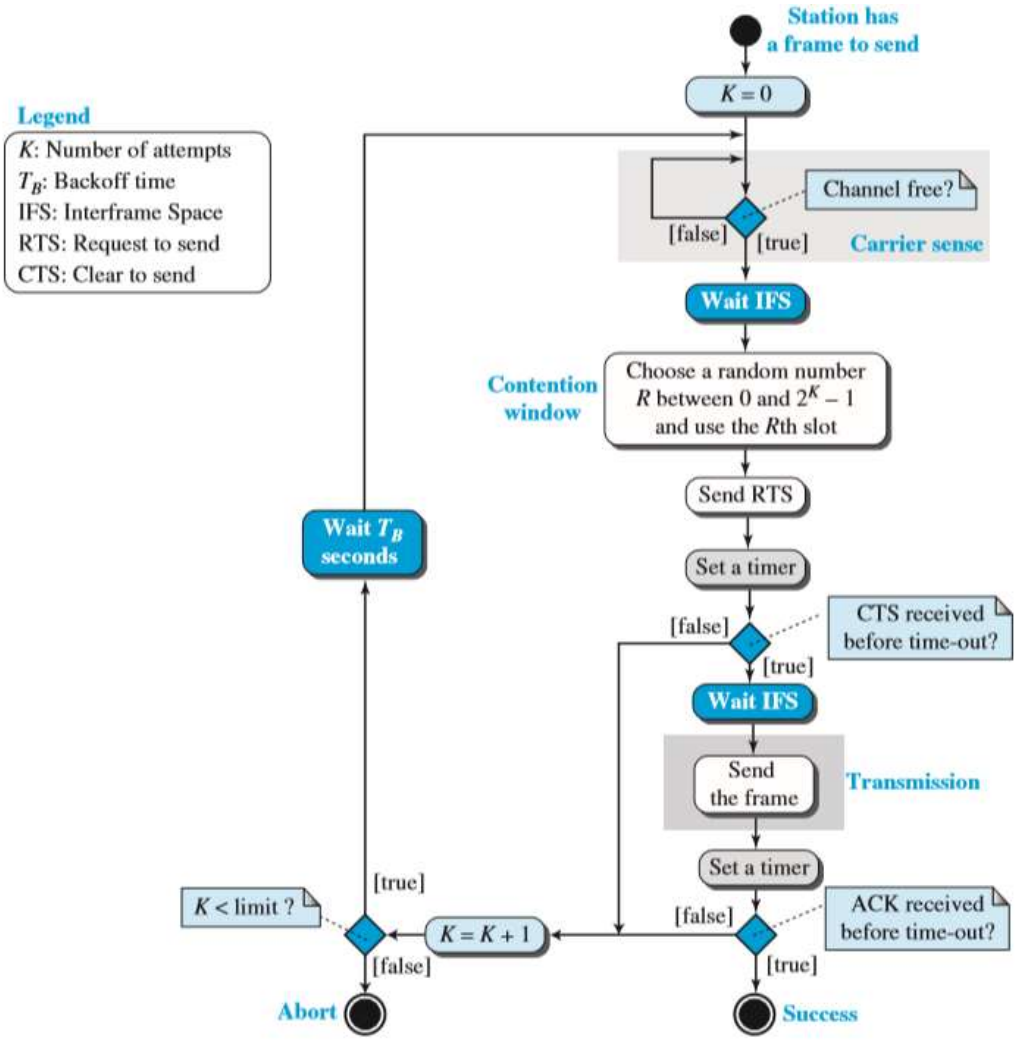


Figure 53. Flow diagram for CSMA/CA

Controlled Access

- In controlled access, the stations consult one another to find which station has the right to send.
- There are three popular controlled-access methods: reservation, polling and token passing.

Reservation

- In the reservation method, a station needs to make a reservation before sending data.
- Time is divided into intervals.
- In each interval, a reservation frame precedes the data frames sent in that interval.
- If there are N stations in the system, there are exactly N reservation minislots in the reservation frame.

Controlled Access (Continue)

- Each minislot belongs to a station.
- When a station needs to send a data frame, it makes a reservation in its own minislot.
- The stations that have made reservations can send their data frames after the reservation frame.

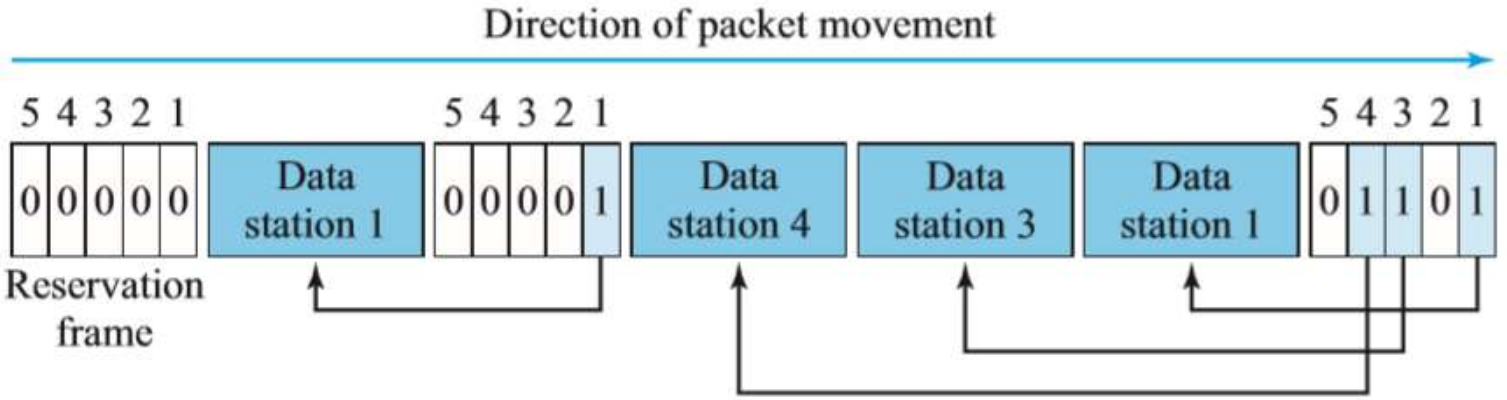


Figure 54. Reservation Access Method

Controlled Access (Continue)

Polling

- Polling works with topologies in which one device is designated as a primary station and the other devices are secondary stations.
- All data exchanges must be made through the primary device even when the ultimate destination is a secondary device.
- The primary device controls the link; the secondary devices follow its instructions.

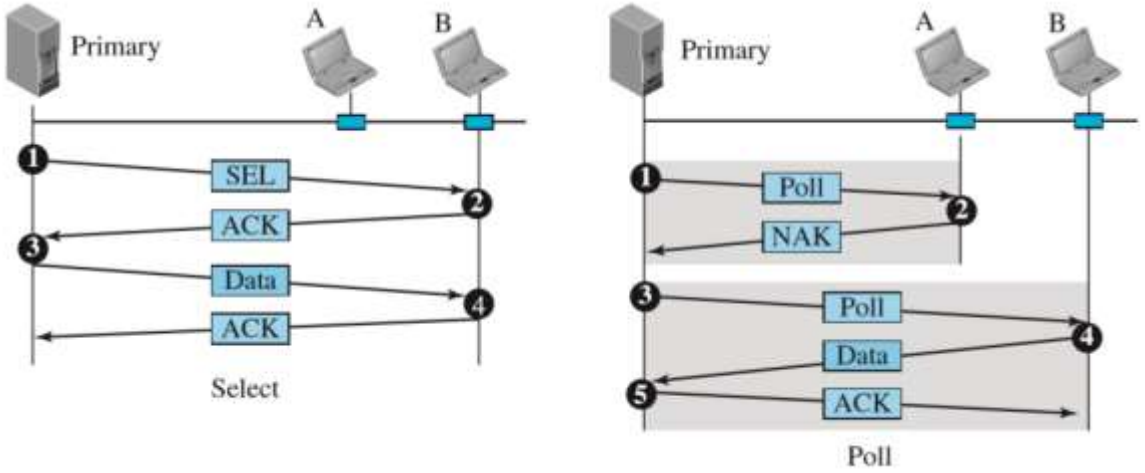


Figure 55. Select and Poll Functions in Polling Access Method

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.
- There are three channelization protocols: FDMA, TDMA, and CDMA.

Frequency-Division Multiple Access (FDMA)

- In frequency-division multiple access (FDMA), the available bandwidth is divided into frequency bands.
- Each station is allocated a band to send its data as illustrated in fig. 56.

Channelization (Continue)

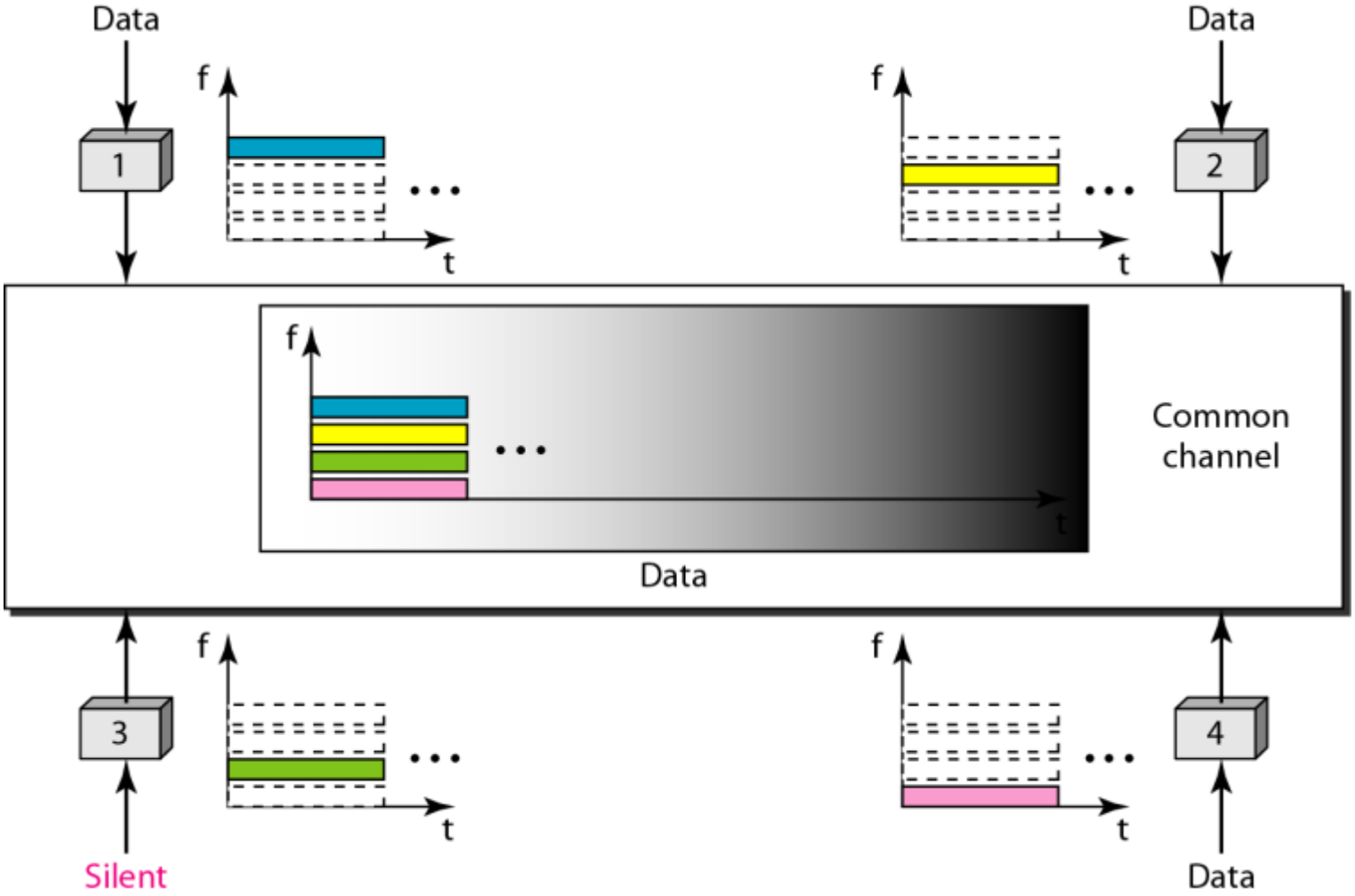


Figure 56. Frequency-division multiple access (FDMA)

Channelization (Continue)

Time-Division Multiple Access (TDMA)

- In time-division multiple access (TDMA), the stations share the bandwidth of the channel in time as shown in figure 57.

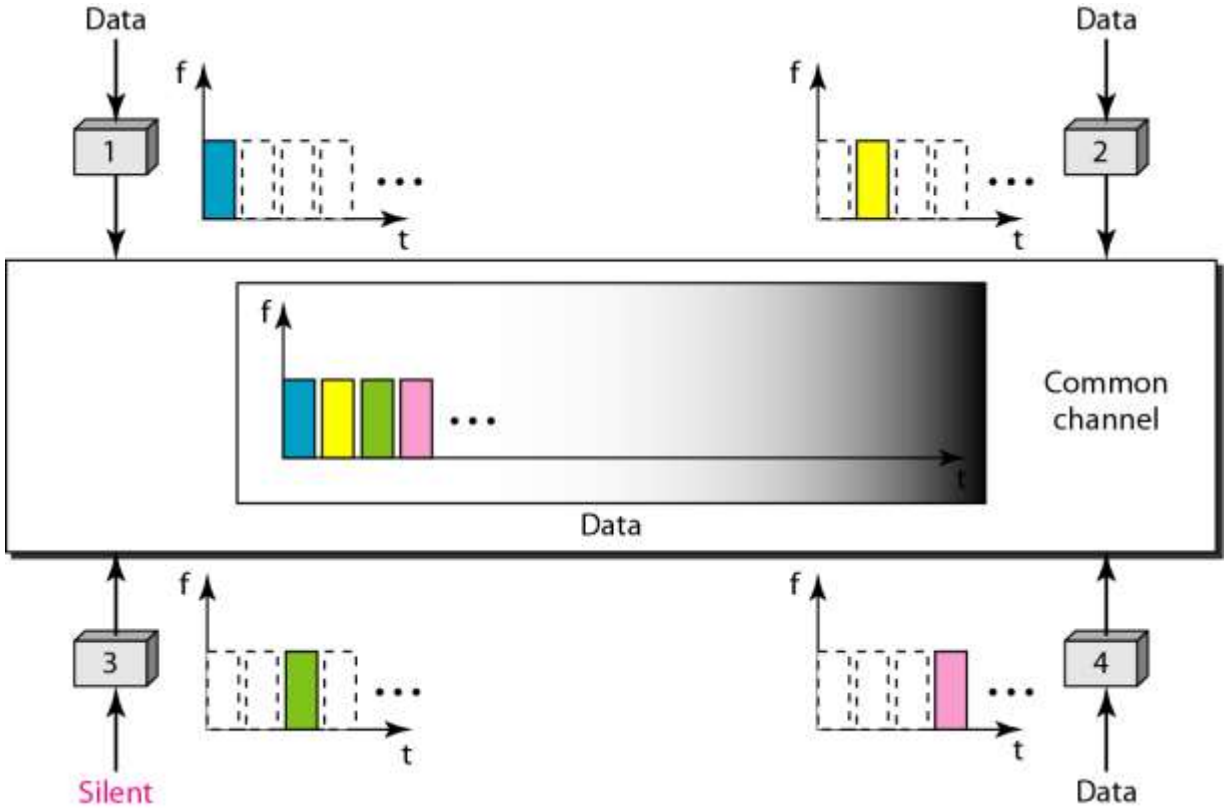


Figure 57. Time-division multiple access (TDMA)

Channelization (Continue)

Code-Division Multiple Access (CDMA)

- In CDMA, one channel carries all transmissions simultaneously.
- It is assumed that it has four stations, four stations, 1, 2, 3, and 4, connected to the same channel.
- The data from station 1 are d_1, from station 2 are d_2, and so on.
- The code assigned to the first station is c_1, to the second is c_1, and so on.
- It is assumed that the assigned codes have two properties.
  1. If each code is multiplied by another, it gets 0.
  2. If each code is multiplied by itself, it gets 4 (the number of stations).

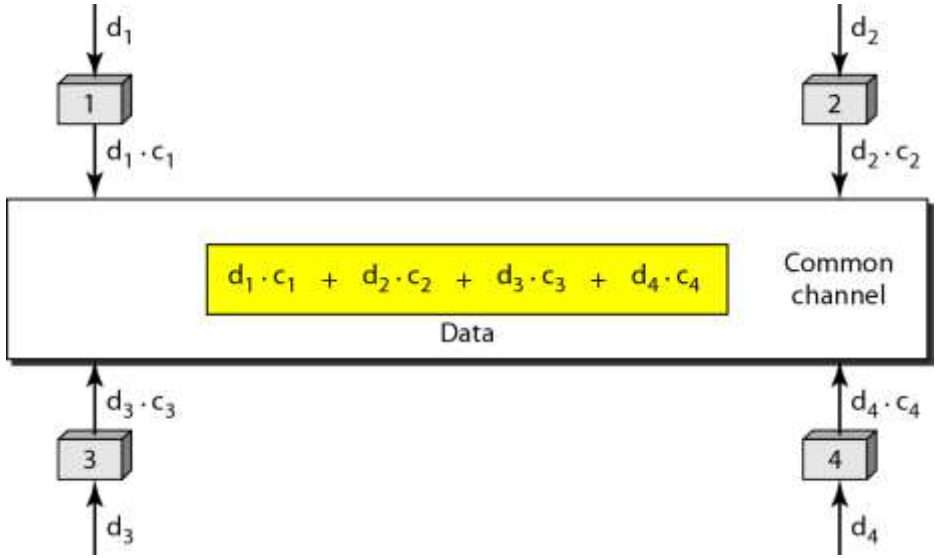


Figure 58. Simple idea of communication with code

Channelization (Continue)

- Each station is assigned a code, which is a sequence of numbers called chips as shown in 58.
- If a station needs to send a 0 bit, it encodes it as -1; if it needs to send a 1 bit, it encodes it as +1.
- When a station is idle, it sends no signal, which is interpreted as a 0 as shown in figure 59.

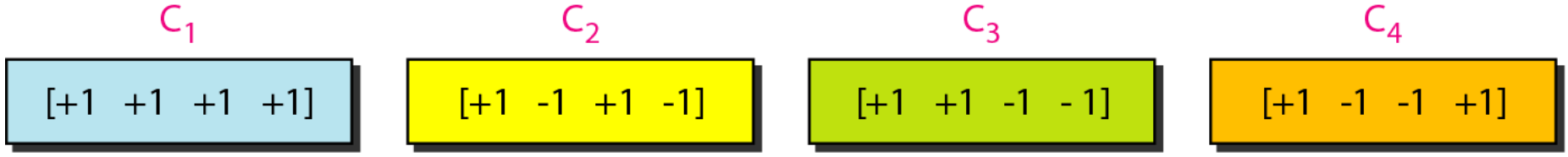


Figure 56. Chip sequences

Figure 59. Data representation in CDMA

Channelization (Continue)

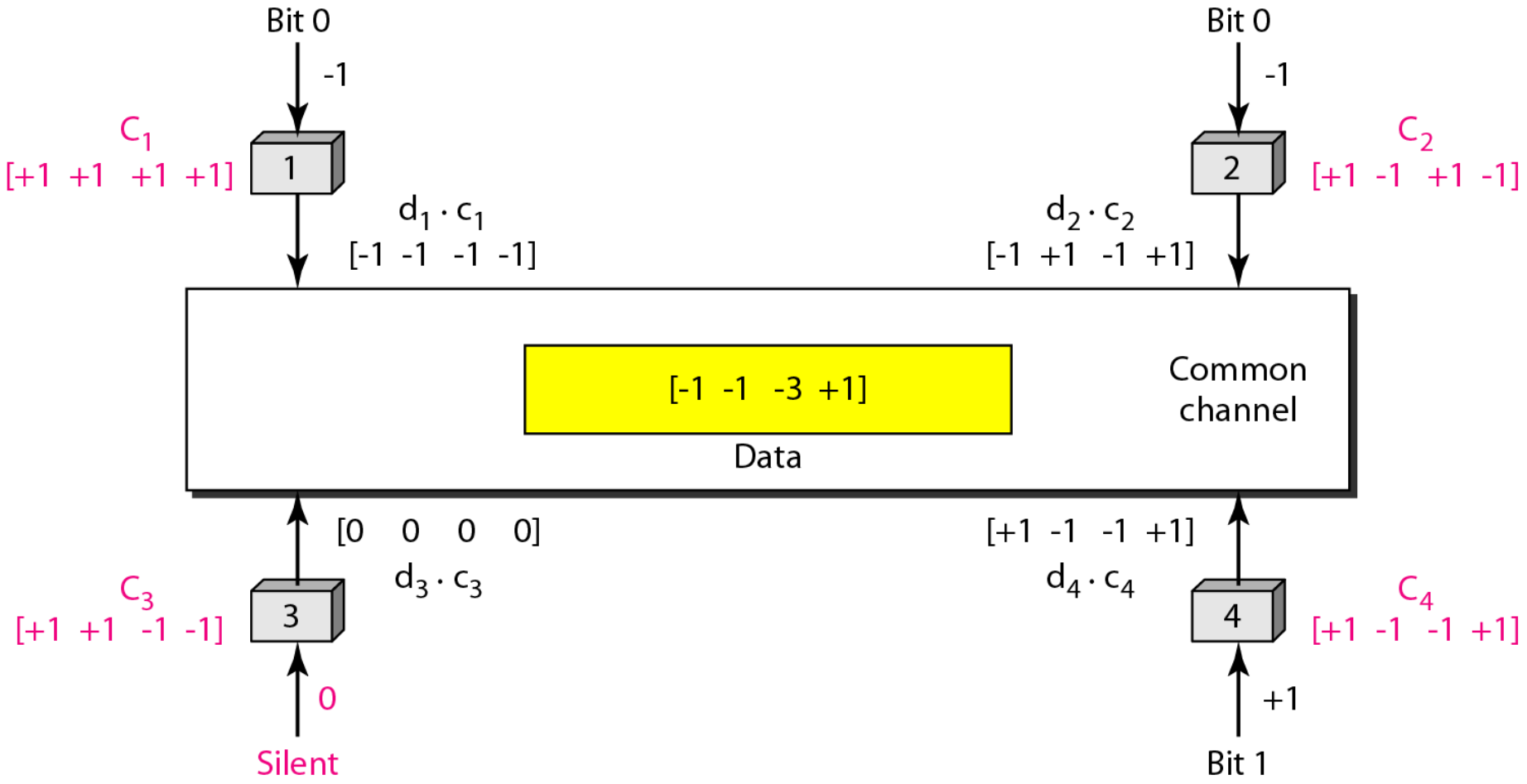


Figure 60. Sharing Channel in CDMA

Next Week Lecture

- Wired LANs: Ethernet
- Other Wired Networks
- Wireless LANs
- Other Wireless Networks
- Connecting Devices and Virtual LANs

Thank You