Module-4

- **Data link control**
  DLC services
  Data link layer protocols
  Point to Point protocol (Framing, Transition phases only).
- **Media Access control**
  Random Access
  Controlled Access
  Channelization
- **Introduction to Data-Link Layer**
  Introduction
  Link-Layer Addressing
  ARP
- **IPv4 Addressing and Sub-netting:**
  Class full and CIDR addressing
  DHCP
  NAT

DATA LINK CONTROL

DLC SERVICES

• The data link control (DLC) deals with procedures for communication between two adjacent nodes i.e. node-to-node communication.
• Data link control functions include
  1) Framing
  2) Flow control
  3) Error control.

Framing

• A frame is a group of bits.
• Framing means organizing the bits into a frame that are carried by the physical layer.
• The data-link-layer needs to form frames, so that each frame is distinguishable from another.
• Framing separates a message from other messages by adding sender-address & destination-address.
• The destination-address defines where the packet is to go.
  The sender-address helps the recipient acknowledge the receipt.
• Q: Why the whole message is not packed in one frame?

Ans: Large frame makes flow and error-control very inefficient.
  Even a single-bit error requires the re-transmission of the whole message.
  When a message is divided into smaller frames, a single-bit error affects only that small frame.

(Our postal system practices a type of framing. The simple act of inserting a letter into an envelope separates one piece of information from another; the envelope serves as the delimiter. In addition, each envelope defines the sender and receiver addresses since the postal system is a many-to-many carrier facility).

Frame Size

• Two types of frames:

Fixed Size Framing

- There is no need for defining boundaries of frames; the size itself can be used as a delimiter.
- For example: ATM WAN uses frames of fixed size called cells.

Variable Size Framing

- We need to define the end of the frame and the beginning of the next frame.
- Two approaches are used: (1) Character-oriented approach
  (2) Bit-oriented approach.

Character Oriented Framing

• Data to be carried are 8-bit characters from a coding system such as ASCII (Figure 11.1).

• The header and the trailer are also multiples of 8 bits.

1) Header carries the source and destination-addresses and other control information.
2) Trailer carries error-detection or error-correction redundant bits.

• To separate one frame from the next frame, an 8- bit (I-byte) flag is added at the beginning and the end of a frame.

• The flag is composed of protocol-dependent special characters.

• The flag signals the start or end of a frame.

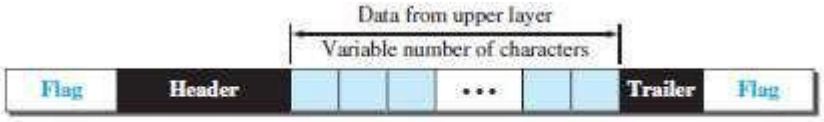


Figure 11.1 *A frame in a character-oriented protocol*

• Problem:

- Character-oriented framing is suitable when only text is exchanged by the data-link-layers.
- However, if we send other type of information (say audio/video), then any pattern used for the flag can also be part of the information.
- If the flag-pattern appears in the data-section, the receiver might think that it has reached the end of the frame.

Solution: A byte-stuffing is used. (Byte stuffing character stuffing)

• In byte 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 called the escape character (ESC), which has a predefined bit pattern.

• When a receiver encounters the ESC character, the receiver
→ removes ESC character from the data-section and
→ treats the next character as data, not a delimiting flag.

• Problem:

- What happens if the text contains one or more escape characters followed by a flag?
- The receiver removes the escape character, but keeps the flag, which is incorrectly interpreted as the end of the frame.

Solution:

- Escape characters part of the text must also be marked by another escape character (Fig 11.2).

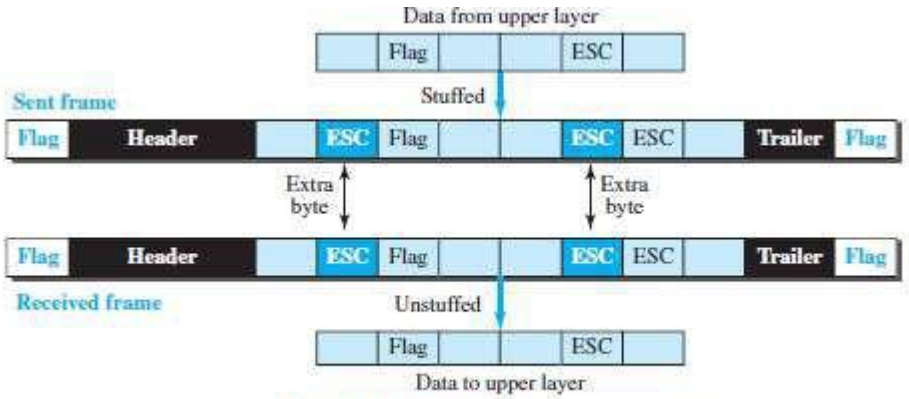


Figure 11.2 *Byte stuffing and unstuffing*

• In short, byte stuffing is the process of adding one extra byte whenever there is a flag or escape character in the text.

Bit Oriented Framing

• The data-section of a frame is a sequence of bits to be interpreted by the upper layer as text, audio, video, and so on.

• However, in addition to headers and trailers, we need 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 (Figure 11.3).

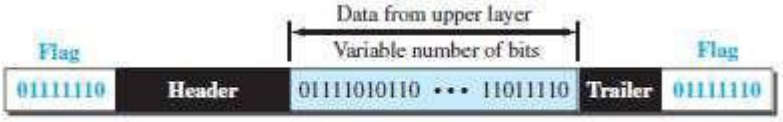


Figure 11.3 *A frame in a bit-oriented protocol*

• Problem:

- If the flag-pattern appears in the data-section, the receiver might think that it has reached the end of the frame.

Solution: A bit-stuffing is used.

• 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 11.4).

• This guarantees that the flag field sequence does not inadvertently appear in the frame.

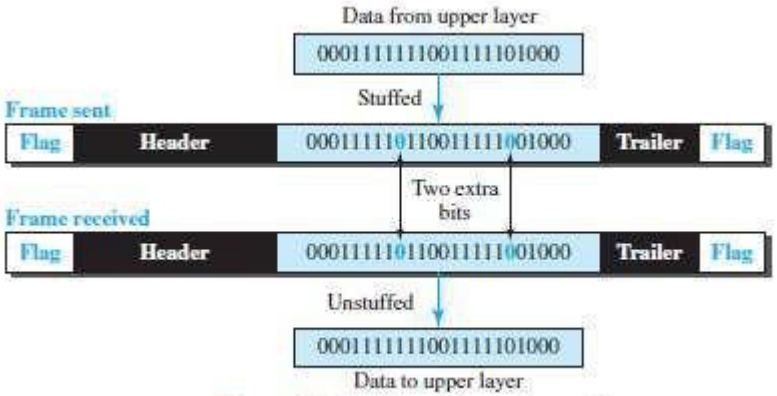


Figure 11.4 *Bit stuffing and unstuffing*

• In short, bit stuffing is the process of adding one extra 0 whenever five consecutive 1s follow a 0 in the data, so that the receiver does not mistake the pattern 0111110 for a flag.

Flow Control and Error Control

• One of the responsibilities of the DLC sublayer is flow and error control at the data-link layer.

Flow Control

• Whenever an entity produces items and another entity consumes them, there should be a balance between production and consumption rates.

• If the items are produced faster than they can be consumed, the consumer can be overwhelmed and may need to discard some items.

• We need to prevent losing the data items at the consumer site.

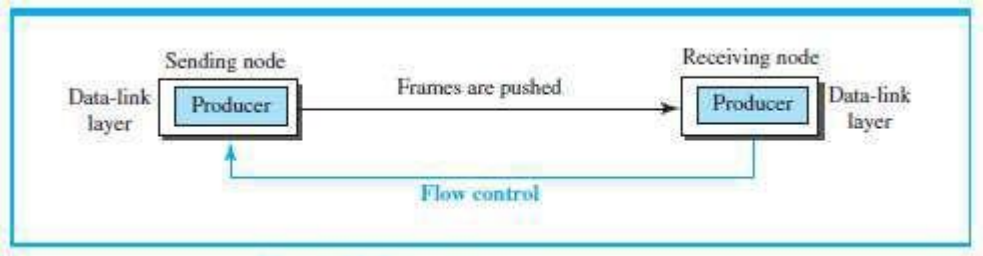


Figure 11.5 *Flow control at the data-link layer*

• At the sending node, the data-link layer tries to push frames toward the data-link layer at the receiving node (Figure 11.5).

• If the receiving node cannot process and deliver the packet to its network at the same rate that the frames arrive, it becomes overwhelmed with frames.

• Here, flow control can be feedback from the receiving node to the sending node to stop or slow down pushing frames.

Buffers

• Flow control can be implemented by using buffer.

• A buffer is a set of memory locations that can hold packets at the sender and receiver.

• Normally, two buffers can be used.

1) First buffer at the sender.
2) Second buffer at the receiver.

• The flow control communication can occur by sending signals from the consumer to the producer.

• When the buffer of the receiver is full, it informs the sender to stop pushing frames.

Error Control

• Error-control includes both error-detection and error-correction.

• Error-control allows the receiver to inform the sender of any frames lost/damaged in transmission.

• A CRC is

→ added to the frame header by the sender and
→ checked by the receiver.

• At the data-link layer, error control is normally implemented using one of the following two methods.

3) First method: If the frame is corrupted, it is discarded;
If the frame is not corrupted, the packet is delivered to the network layer.
This method is used mostly in wired LANs such as Ethernet.
4) Second method: If the frame is corrupted, it is discarded;
If the frame is not corrupted, an acknowledgment is sent to the sender.
Acknowledgment is used for the purpose of both flow and error control.

Combination of Flow and Error Control

• Flow and error control can be combined.

• The acknowledgment that is sent for flow control can also be used for error control to tell the sender the packet has arrived uncorrupted.

• The lack of acknowledgment means that there is a problem in the sent frame.

• A frame that carries an acknowledgment is normally called an ACK to distinguish it from the data frame.

Connectionless and Connection-Oriented

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

Connectionless Protocol

- Frames are sent from one node to the next without any relationship between the frames; each frame is independent.
- The term connectionless does not mean that there is no physical connection (transmission medium) between the nodes; it means that there is no connection between frames.
- The frames are not numbered and there is no sense of ordering.
- Most of the data-link protocols for LANs are connectionless protocols.

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).
- The frames are numbered and sent in order.
- If the frames are not received in order, the receiver needs to wait until all frames belonging to the same set are received and then deliver them in order to the network layer.
- Connection oriented protocols are rare in wired LANs, but we can see them in some point-to-point protocols, some wireless LANs, and some WANs

High-Level Data Link Control (HDLC)

- HDLC is a bit-oriented protocol for communication over point-to-point and multipoint links.
- HDLC implements the ARQ mechanisms.

3.8.1 Configurations and Transfer Modes

- HDLC provides 2 common transfer modes that can be used in different configurations:
    1) Normal response mode (NRM)
    2) Asynchronous balanced mode (ABM).

NRM

- The station configuration is unbalanced (Figure 11.14).
- We have 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 multiple-point links.

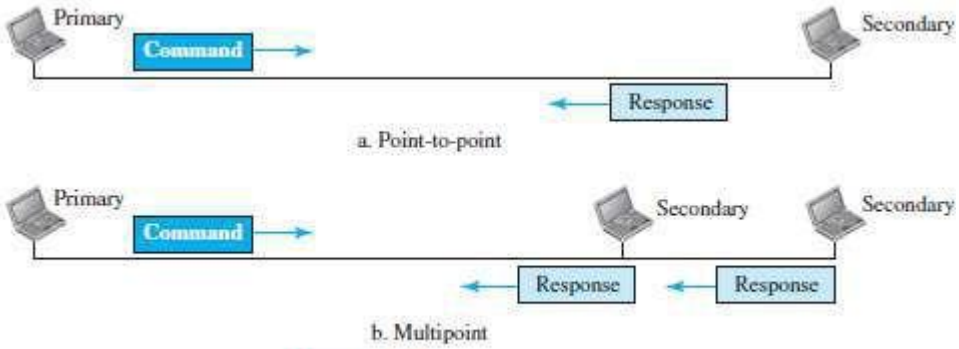


Figure 11.14 *Normal response mode*

ABM

- The configuration is balanced (Figure 11.15).
- Link is point-to-point, and each station can function as a primary and a secondary (acting as peers).
- This is the common mode today.

Figure 11.15 *Asynchronous balanced mode*

3.8.2 Framing

• To provide the flexibility necessary to support all the options possible in the modes and configurations, HDLC defines three types of frames:

1) Information frames (I-frames): are used to transport user data and control information relating to user data (piggybacking).
2) Supervisory frames (S-frames): are used only to transport control information.
3) Unnumbered frames (U-frames): are reserved for system management. Information carried by U-frames is intended for managing the link itself.

• Each type of frame serves as an envelope for the transmission of a different type of message.

3.8.2.1 Frame Format

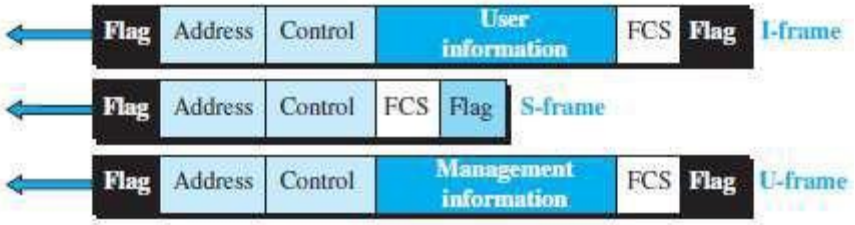


Figure 11.16 *HDLC frames*

• Various fields of HDLC frame are:

1) Flag Field

- This field has a synchronization pattern 01111110.
- This field identifies both the beginning and the end of a frame.

2) Address Field

- This field contains the address of the secondary station.
- If a primary station created the frame, it contains a to-address.
- If a secondary creates the frame, it contains a from-address.
- This field can be 1 byte or several bytes long, depending on the needs of the network.

3) Control Field

- This field is one or two bytes used for flow and error control.

4) Information Field

- This field contains the user's data from the network-layer or management information.
- Its length can vary from one network to another.

5) FCS Field

- This field is the error-detection field. (FCS → Frame CheckSequence)
- This field can contain either a 2- or 4-byte standard CRC.

3.8.2.1.1 Control Fields of HDLC Frames

- The control field determines the type of frame and defines its functionality (Figure 11.17).

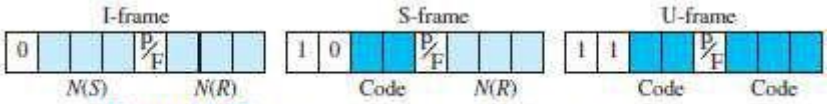


Figure 11.17 *Control field format for the different frame types*

1) Control Field for I-Frames

- I-frames are designed to carry user data from the network-layer.
- In addition, they can include flow and error-control information (piggybacking).
- The subfields in the control field are:
  1) The first bit defines the type.
     If the first bit of the control field is 0, this means the frame is an I-frame.
  2) The next 3 bits N(S) define the sequence-number of the frame. With 3 bits, we can define a sequence-number between 0 and 7
  3) The last 3 bits N(R) correspond to the acknowledgment-number when piggybacking is used.
  4) The single bit between N(S) and N(R) is called the P/F bit.
     The P/F field is a single bit with a dual purpose. It can mean poll or final.
     i) It means poll when the frame is sent by a primary station to a secondary (when the address field contains the address of the receiver).
     ii) It means final when the frame is sent by a secondary to a primary (when the address field contains the address of the sender).

2) Control Field for S-Frames

- Supervisory frames are used for flow and error-control whenever piggybacking is either impossible or inappropriate (e.g., when the station either has no data of its own to send or needs to send a command or response other than an acknowledgment).
- S-frames do not have information fields.
- The subfields in the control field are:
  1) If the first 2 bits of the control field is 10, this means the frame is an S-frame.
  2) The last 3 bits N(R) corresponds to the acknowledgment-number (ACK) or negative acknowledgment-number (NAK).
  3) The 2 bits called code is used to define the type of S-frame itself. With 2 bits, we can have four types of S-frames:
     1) Receive Ready (RR) = 00
     ¤ This acknowledges the receipt of frame or group of frames.
     ¤ The value of N(R) is the acknowledgment-number.
     2) Receive Not Ready (RNR) = 10
     ¤ This is an RR frame with 1 additional function:
       i) It announces that the receiver is busy and cannot receive more frames.
     ¤ It acts as congestion control mechanism by asking the sender to slow down.
     ¤ The value of N(R) is the acknowledgment-number.
     3) ReJect (REJ) = 01
     ¤ It is a NAK frame used in Go-Back-N ARQ to improve the efficiency of the process.
     ¤ It informs the sender, before the sender time expires, that the last frame is lost or damaged.
     ¤ The value of N(R) is the negative acknowledgment-number.
     4) Selective REJect (SREJ) = 11
     ¤ This is a NAK frame used in Selective Repeat ARQ.
     ¤ The value of N(R) is the negative acknowledgment-number.

3) Control Field for U-Frames

- Unnumbered frames are used to exchange session management and control information between connected devices.
- U-frames contain an information field used for system management information, but not user data.
- Much of the information carried by U-frames is contained in codes included in the control field.
- U-frame codes are divided into 2 sections:
  - i) A 2-bit prefix before the P/F bit
  - ii) A 3-bit suffix after the P/F bit.
- Together, these two segments (5 bits) can be used to create up to 32 different types of U-frames.

Example 3.9

Figure 11.18 shows how U-frames can be used for connection establishment and connection release. Node A asks for a connection with a set asynchronous balanced mode (SABM) frame; node B gives a positive response with an unnumbered acknowledgment (UA) frame. After these two exchanges, data can be transferred between the two nodes (not shown in the figure). After data transfer, node A sends a DISC (disconnect) frame to release the connection; it is confirmed by node B responding with a UA (unnumbered acknowledgment).

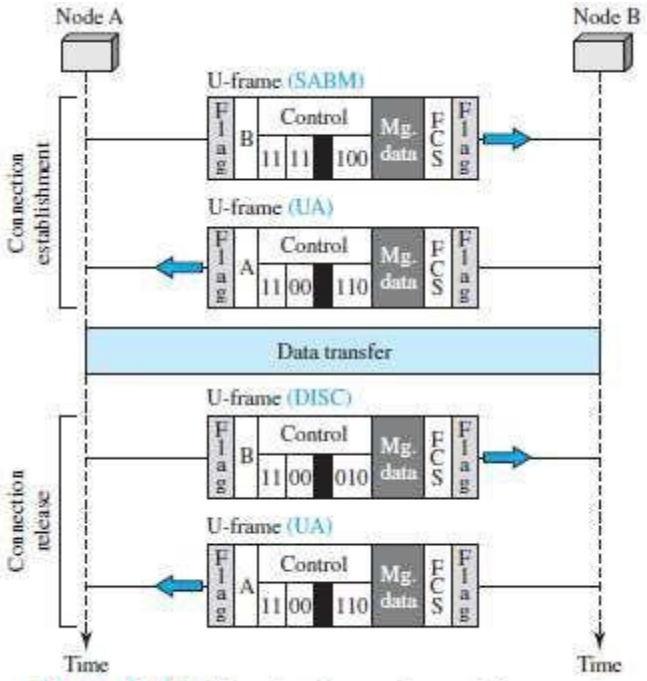


Figure 11.18 *Example of connection and disconnection*

DATA LINK LAYER PROTOCOLS

- Traditionally 2 protocols have been defined for the data-link layer to deal with flow and error control:
  1) Simple Protocol and 2) Stop-and-Wait Protocol.
- The behaviour of a data-link-layer protocol can be better shown as a finite state machine (FSM).
- An FSM is a machine with a finite number of states (Figure 11.6).
- The machine is always in one of the states until an event occurs.
- Each event is associated with 2 reactions:
  1) Defining the list (possibly empty) of actions to be performed.
  2) Determining the next state (which can be the same as the current state).
- One of the states must be defined as the initial state, the state in which the machine starts when it turns on.

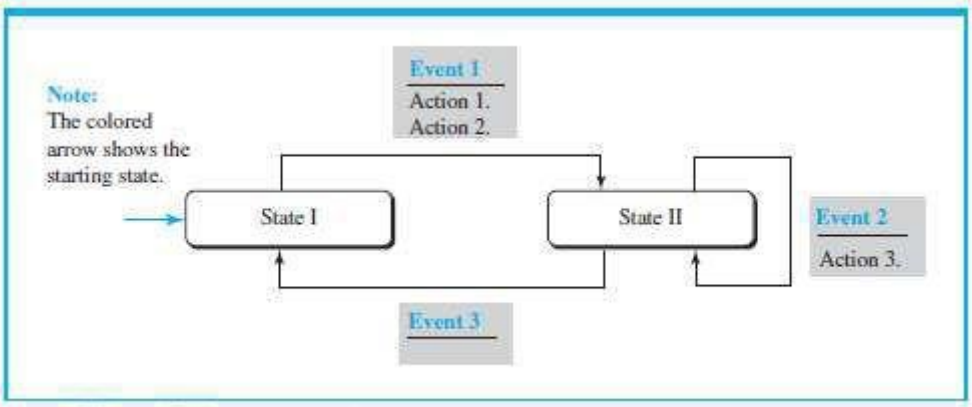


Figure 11.6 *Connectionless and connection-oriented service represented as FSMs*

Simplest Protocol

- Assumptions:
  - The protocol has no flow-control or error-control.
  - The protocol is a unidirectional protocol (in which frames are traveling in only one direction).
  - The receiver can immediately handle any frame it receives.

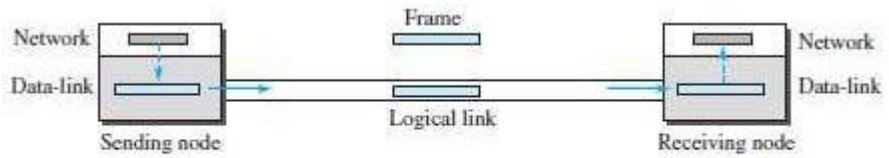


Figure 11.7 *Simple protocol*

Design

- Here is how it works (Figure 11.7):

At Sender

¤ The data-link-layer
→ gets data from its network-layer
→ makes a frame out of the data and
→ sends the frame.

At Receiver

¤ The data-link-layer
→ receives a frame from its physical layer
→ extracts data from the frame and
→ delivers the data to its network-layer.

- Data-link-layers of sender & receiver provide transmission services for their network-layers.
- Data-link-layers use the services provided by their physical layers for the physical transmission of bits.

FSMs

- Two main requirements:
  1) The sender-site cannot send a frame until its network-layer has a data packet to send.
  2) The receiver-site cannot deliver a data packet to its network-layer until a frame arrives.
- These 2 requirements are shown using two FSMs.
- Each FSM has only one state, the ready state.

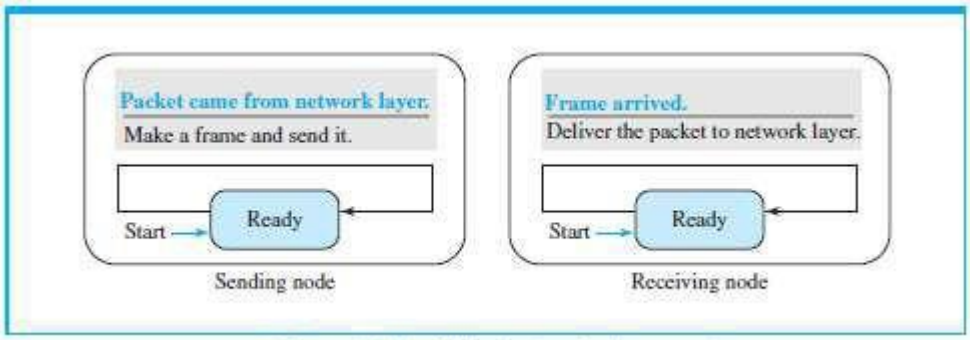


Figure 11.8 *FSMs for the simple protocol*

- Here is how it works (Figure 11.8):

1) At Sending Machine

➢ The sending machine remains in the ready state until a request comes from the process in the network layer.

➢ When this event occurs, the sending machine encapsulates the message in a frame and sends it to the receiving machine.

2) At Receiving Machine

➢ The receiving machine remains in the ready state until a frame arrives from the sending machine.

➢ When this event occurs, the receiving machine decapsulates the message out of the frame and delivers it to the process at the network layer.

Example 3.6

Figure 11.9 shows an example of communication using this protocol. It is very simple. The sender sends frames one after another without even thinking about the receiver.

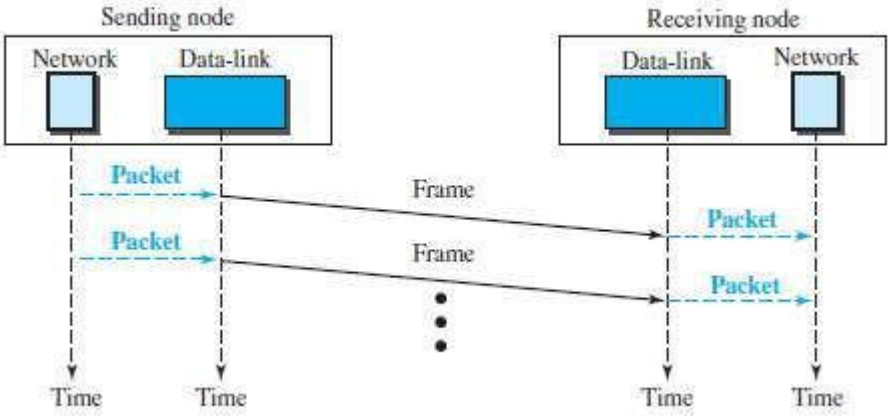


Figure 11.9 *Flow diagram*

Stop & Wait Protocol

- This uses both flow and error control.
- Normally, the receiver has limited storage-space.
- If the receiver is receiving data from many sources, the receiver may
  → be overloaded with frames &
  → discard the frames.
- To prevent the receiver from being overloaded with frames, we need to tell the sender to slow down.

Design

1) At Sender

- The sender
  → sends one frame & starts a timer
  → keeps a copy of the sent-frame and
  → waits for ACK-frame from the receiver (okay to go ahead).
- Then,
  1) If an ACK-frame arrives before the timer expires, the timer is stopped and the sender sends the next frame.
     Also, the sender discards the copy of the previous frame.
  2) If the timer expires before ACK-frame arrives, the sender resends the previous frame and restarts the timer

2) At Receiver

- To detect corrupted frames, a CRC is added to each data frame.
- When a frame arrives at the receiver-site, the frame is checked.
- If frame's CRC is incorrect, the frame is corrupted and discarded.
- The silence of the receiver is a signal for the sender that a frame was either corrupted or lost.

FSMs

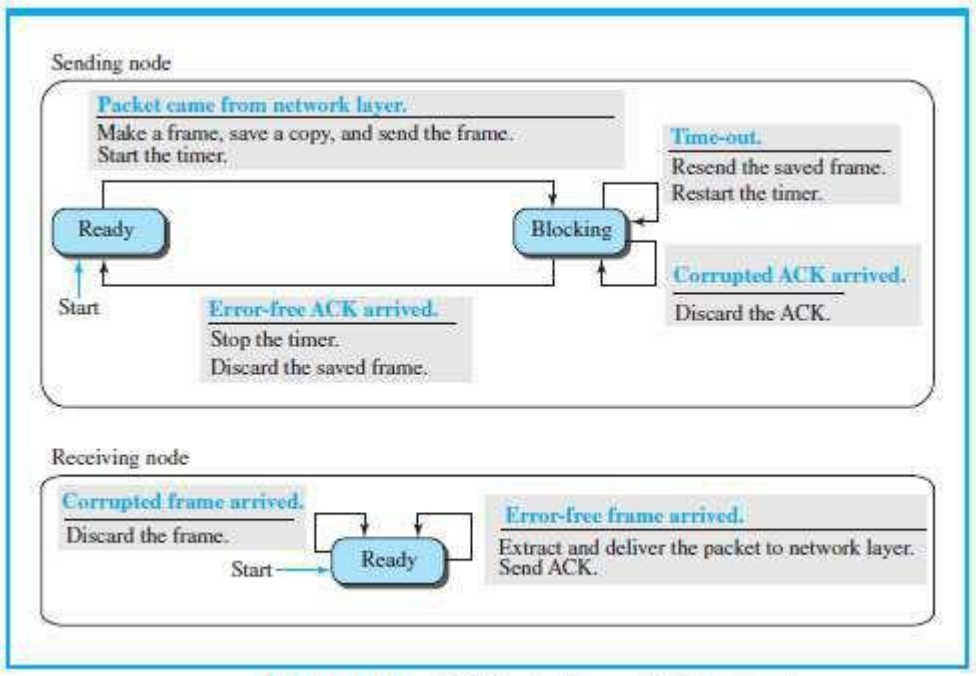


Figure 11.11 *FSM for the Stop-and-Wait protocol*

- Here is how it works (Figure 11.11):

3) Sender States

- Sender is initially in the ready state, but it can move between the ready and blocking state.

i) 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.

ii) 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.

4) 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 3.7

Figure 11.12 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.

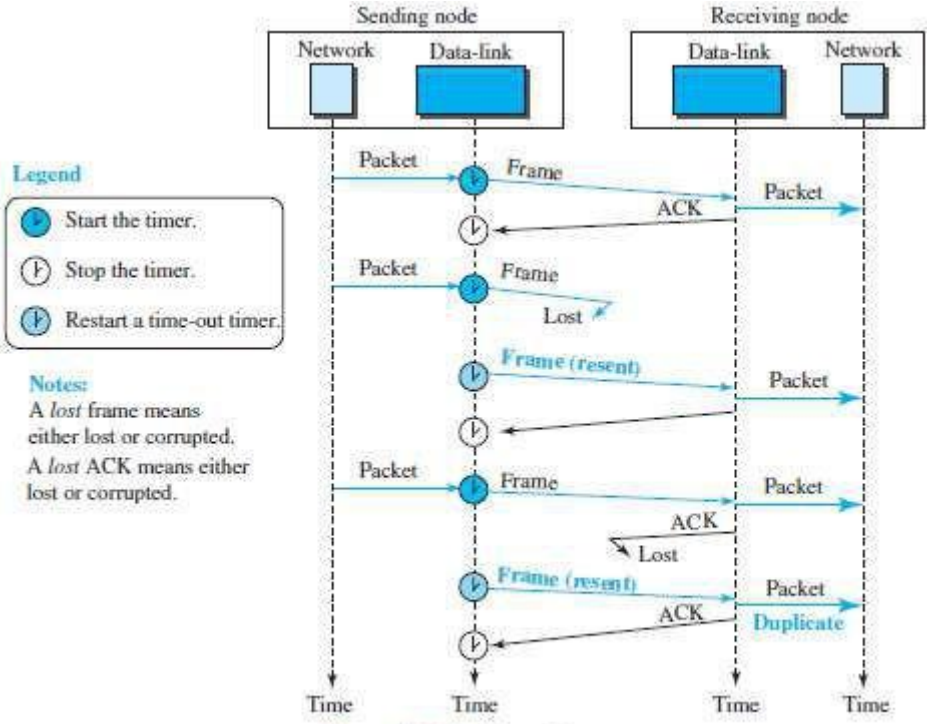


Figure 11.12 *Flow diagram*

Sequence and Acknowledgment Numbers

- Q: How to deal with corrupted-frame?

  Ans: If the corrupted-frame arrives at the receiver-site, then the frame is simply discarded.

- Q: How to deal with lost-frames?

  Ans: If the receiver receives out-of-order data-frame, then it means that frames were lost. .'.

  The lost-frames need to be resent.

- Problem in Stop and Wait protocols:

  5) There is no way to identify a frame.

  6) The received-frame could be the correct one, or a duplicate, or a frame out of order. Solution: 1) Use sequence-number for each data frame.

  2) Use Acknowledgment-number for each ACK frame.

Sequence Numbers

- Frames need to be numbered. This is done by using sequence-numbers.
- A sequence-number field is added to the data-frame.

Acknowledgment Numbers

- An acknowledgment-number field is added to the ACK-frame.
- Sequence numbers are 0, 1, 0, 1, 0, 1, . . .

  The acknowledgment numbers can also be 1, 0, 1, 0, 1, 0, …

- The acknowledgment-numbers always announce the sequence-number of the next frame expected by the receiver.
- For example,

  If frame-0 has arrived safely, the receiver sends an ACK-frame with acknowledgment-1 (meaning frame-1 is expected next).

Example 3.8

Figure 11.13 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.

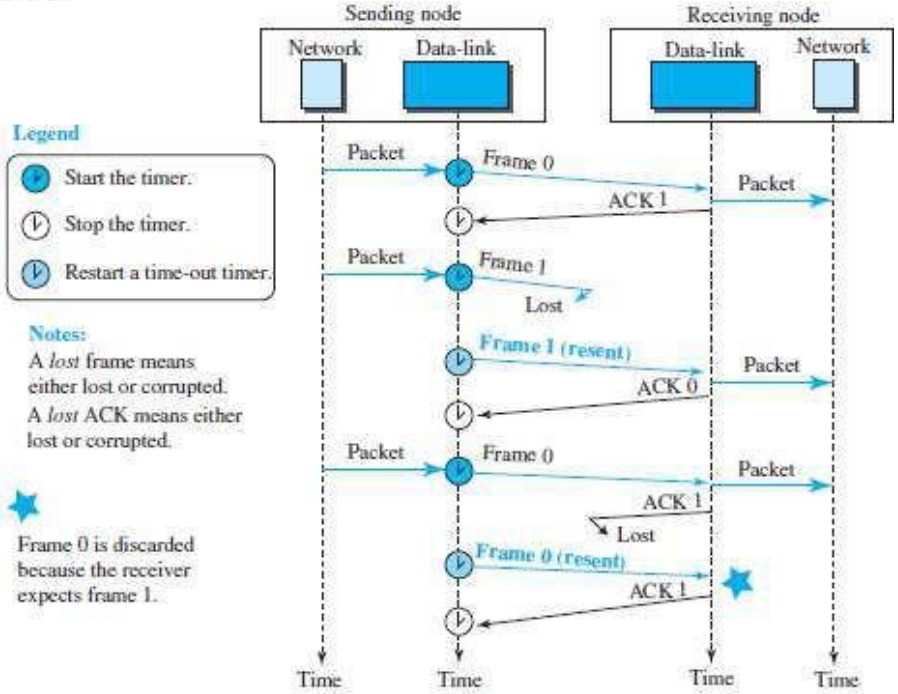


Figure 11.13 *Flow diagram*

Piggybacking

- A technique called piggybacking is used to improve the efficiency of the bidirectional protocols.
- The data in one direction is piggybacked with the acknowledgment in the other direction.
- In other words, when node A is sending data to node B, Node A also acknowledges the data received from node B.

POINT-TO-POINT PROTOCOL (PPP)

- PPP is one of the most common protocols for point-to-point access.
- Today, millions of Internet users who connect their home computers to the server of an ISP use PPP.

Framing

- PPP uses a character-oriented (or byte-oriented) frame (Figure 11.20).

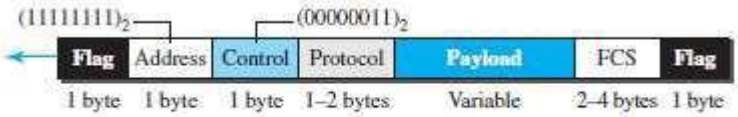


Figure 11.20 *PPP frame format*

- Various fields of PPP frame are:

1) Flag

- This field has a synchronization pattern 01111110.
- This field identifies both the beginning and the end of a frame.

2) Address

- This field is set to the constant value 11111111 (broadcast address).

3) Control

- This field is set to the constant value 00000011 (imitating unnumbered frames in HDLC).
- PPP does not provide any flow control.
- Error control is also limited to error detection.

4) Protocol

- This field defines what is being carried in the payload field.
- Payload field carries either i) user data or ii) other control information.
- By default, size of this field = 2 bytes.

5) Payload field

- This field carries either i) user data or ii) other control information.
- By default, maximum size of this field = 1500 bytes.
- This field is byte-stuffed if the flag-byte pattern appears in this field.
- Padding is needed if the payload-size is less than the maximum size.

6) FCS

- This field is the PPP error-detection field.
- This field can contain either a 2- or 4-byte standard CRC.

Byte Stuffing

- Since PPP is a byte-oriented protocol, the flag in PPP is a byte that needs to be escaped whenever it appears in the data section of the frame.
- The escape byte is 01111101, which means that every time the flag like pattern appears in the data, this extra byte is stuffed to tell the receiver that the next byte is not a flag.
- Obviously, the escape byte itself should be stuffed with another escape byte.

Transition Phases

- The transition diagram starts with the dead state (Figure 11.21).

1) Dead State

- In dead state, there is no active carrier and the line is quiet.

2) Establish State

- When 1 of the 2 nodes starts communication, the connection goes into the establish state.
- In establish state, options are negotiated between the two parties.

3) Authenticate State

- If the 2 parties agree that they need authentication, Then the system needs to do authentication;

4) Open State

- Data transfer takes place in the open state.

5) Terminate State

- When 1 of the endpoints wants to terminate connection, the system goes to terminate state.

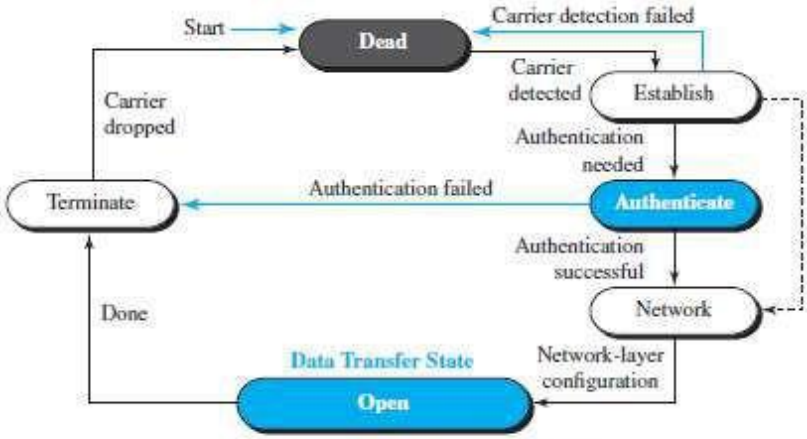


Figure 11.21 *Transition phases*

Multiple Access Protocols

- When nodes use shared-medium, there is a need for multiple-access protocol to coordinate access to medium.
- Analogy:
  - This problem is similar to the rules of speaking in an assembly.
  - It need to ensure
    - → Each people has right to speak.
    - → Two people do not speak at the same time
    - → Two people do not interrupt each other (i.e. Collision Avoidance)
- Many protocols have been designed to handle access to a shared-link (Figure 12.1).
- These protocols belong to a sublayer in the data-link layer called Media Access Control (MAC).
  1) Four random-access protocols (or Contention Methods):
     - i) ALOHA
     - ii) ii) CSMA
     - iii) CSMA/CD
     - iv) CSMA/CA

     These protocols are mostly used in LANs and WANs.
  2) Three controlled-access protocols:
     - i) Reservation
     - ii) ii) Polling
     - iii) Token-passing

     Some of these protocols are used in LANs.
  3) Three channelization protocols:
     - i) FDMA
     - ii) TDMA
     - iii) CDMA

     These protocols are used in cellular telephony.

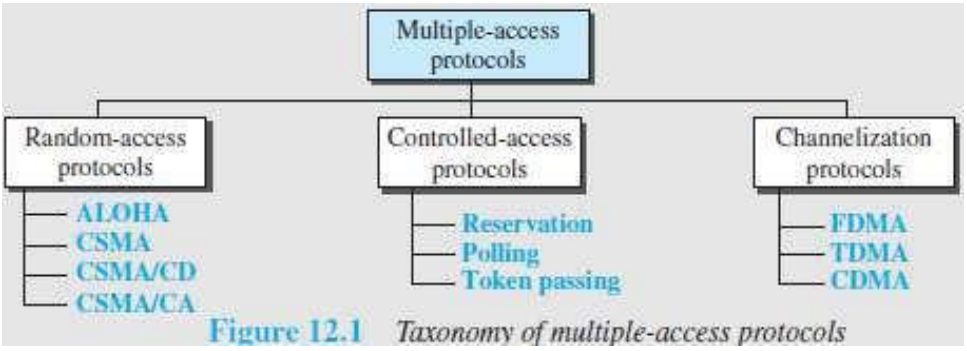


Figure 12.1 *Taxonomy of multiple-access protocols*

RANDOM ACCESS PROTOCOL

- No station is superior to another station.
- No station is assigned control over other station.
- To send the data, a station uses a procedure to make a decision on whether or not to send.
- This decision depends on the state of the medium: idle or busy.
- This is called Random Access because
  - → Transmission is random among the stations.
  - → There is no scheduled-time for a station to transmit.
- This is called Contention Method because
  - → Stations compete with one another to access the medium.
- If more than one station tries to send,
  - there is an access-conflict (i.e. collision) and the frames will be destroyed.
- Each station follows a procedure that answers the following questions:
  1) When can the station access the medium?
  2) What can the station do if the medium is busy?
  3) How can the station determine the success or failure of the transmission?
  4) What can the station do if there is a collision?
- Four random-access protocols (or Contention methods):
  1) ALOHA
  2) CSMA (Carrier Sense Multiple Access)
  3) CSMA/CD (Carrier Sense Multiple Access with Collision-detection)
  4) CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)

ALOHA

- ALOHA was designed for a wireless LAN, but it can be used on any shared medium.
- Since the medium is shared between the stations, there is possibility of collisions.
- When 2 or more stations send the data simultaneously, there is possibility of collision & data loss.

Pure ALOHA

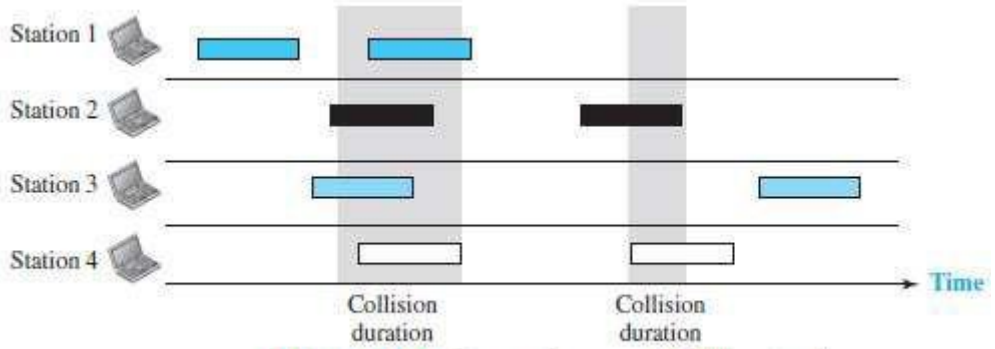


Figure 12.2 *Frames in a pure ALOHA network*

- Here is how it works (Figure 12.2):
  1) The sender sends a frame & starts the timer.
  2) The receiver receives the frame and responds with an acknowledgment.
  3) If the acknowledgment does not arrive after a time-out period, the sender resends the frame. The sender assumes that the frame (or the acknowledgment) has been destroyed.
  4) Since the medium is shared between the stations, there is possibility of collisions.
  5) If two stations try to resend the frames after the time-out, the frames will collide again.
  6) Two methods to deal with collision:

1) Randomness

¤ When the time-out period passes, each station waits a random amount of time before resending the frame. This time is called back-off time T_B.

¤ The randomness will help avoid more collisions.

2) Limit Maximum Retransmission

¤ This method prevents congestion by reducing the number of retransmitted frames.

¤ After a maximum number of retransmission-attempts K_{max}, a station must give up and try later.

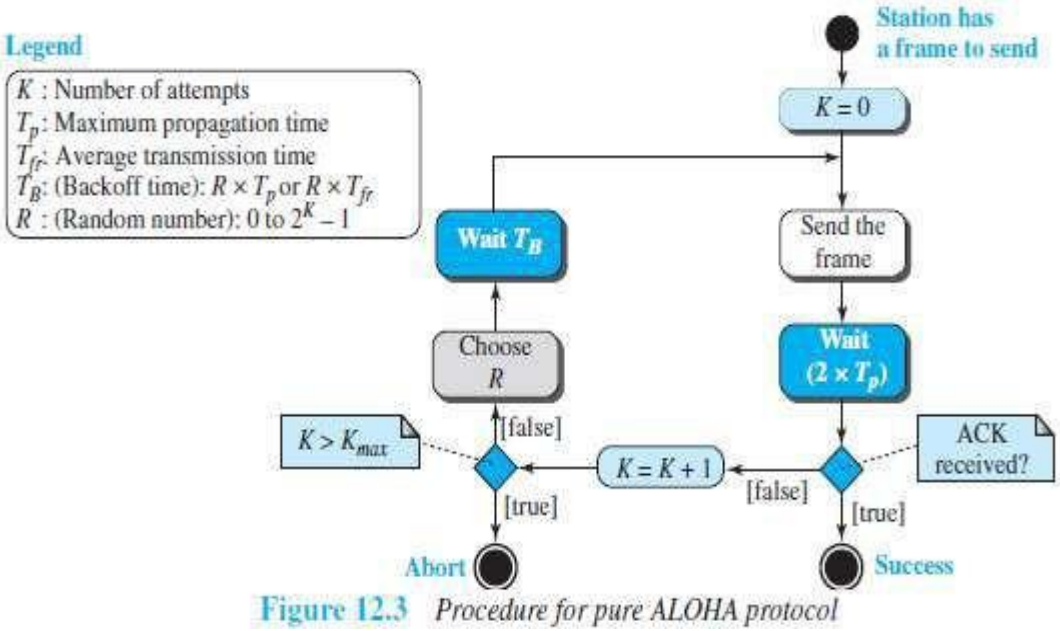


Figure 12.3 *Procedure for pure ALOHA protocol*

Vulnerable Time

- The vulnerable-time is defined as a time during which there is a possibility of collision.

Pure ALOHA vulnerable time $= 2 \times T_{fr}$

where T_{fr} = Frame transmission time

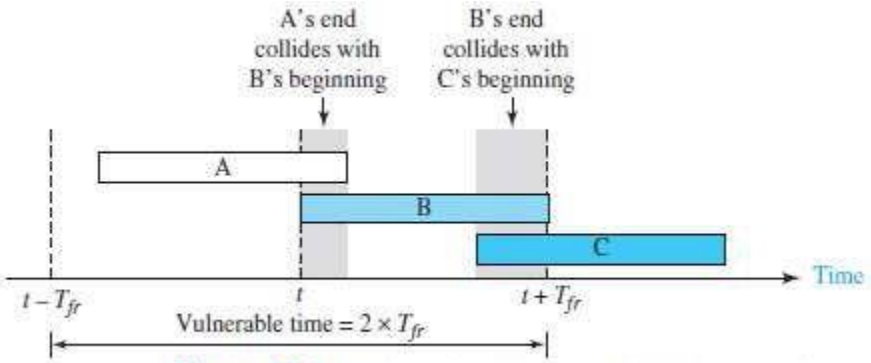


Figure 12.4 *Vulnerable time for pure ALOHA protocol*

- In Figure 12.4,
  - If station B sends a frame between t-T_{fr} and t, this leads to a collision between the frames from station A and station B.
  - If station C sends a frame between t and t+T_{fr}, this leads to a collision between the frames from station A and station C.

Example

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

Solution

Average frame transmission time 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.

Throughput

- The average number of successful transmissions is given by

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

where G = average no. of frames in one frame transmission time (T_{fr})

- For G = 1, the maximum throughput S_{max} = 0.184.
- In other words, out of 100 frames, 18 frames reach their destination successfully.

Example

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces

a. 1000 frames per second? b. 500 frames per second? c. 250 frames per second?

Solution

The frame transmission time is 200/200 kbps or 1 ms.

a. If the system creates 1000 frames per second, or 1 frame per millisecond, then $G = 1$. In this case $S = G \times e^{-2G} = 0.135$ (13.5 percent). This means that the throughput is $1000 \times 0.135 = 135$ frames. Only 135 frames out of 1000 will probably survive.

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

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

Slotted ALOHA

- Slotted ALOHA was invented to improve the efficiency of pure ALOHA.
- The time is divided into time-slots of T_{fr} seconds (Figure 12.5).
- The stations are allowed to send only at the beginning of the time-slot.

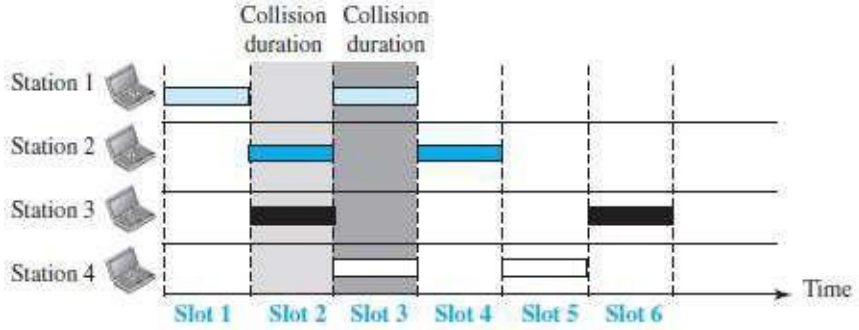


Figure 12.5 *Frames in a slotted ALOHA network*

- If a station misses the time-slot, the station must wait until the beginning of the next time-slot.
- If 2 stations try to resend at beginning of the same time-slot, the frames will collide again (Fig 12.6).

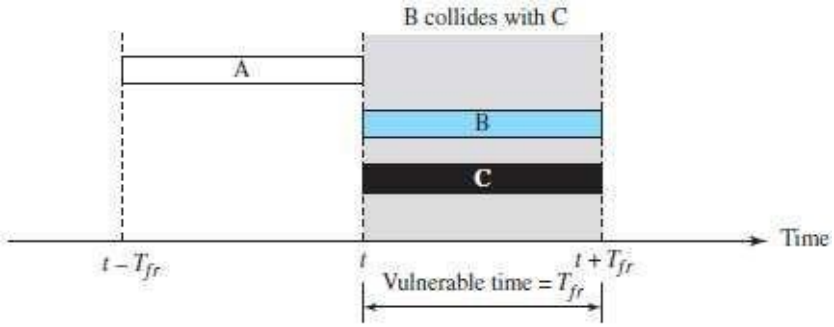


Figure 12.6 *Vulnerable time for slotted ALOHA protocol*

- The vulnerable time is given by:

vulnerable time = T_{fr}

Throughput

- The average number of successful transmissions is given by

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

- For G = 1, the maximum throughput S_{max} = 0.368.
- In other words, out of 100 frames, 36 frames reach their destination successfully.

Example 4.3

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

a. 1000 frames per second.
b. 500 frames per second.
c. 250 frames per second.

Solution

This situation is similar to the previous exercise except that the network is using slotted ALOHA instead of pure ALOHA. The frame transmission time is 200/200 kbps or 1 ms.

a. In this case G is 1. So $S = G \times e^{-G} = 0.368$ (36.8 percent). This means that the throughput is $1000 \times 0.0368 = 368$ frames. Only 368 out of 1000 frames will probably survive. Note that this is the maximum throughput case, percentagewise.
b. Here G is 1/2. In this case $S = G \times e^{-G} = 0.303$ (30.3 percent). This means that the throughput is $500 \times 0.0303 = 151$. Only 151 frames out of 500 will probably survive.
c. Now G is 1/4. In this case $S = G \times e^{-G} = 0.195$ (19.5 percent). This means that the throughput is $250 \times 0.195 = 49$. Only 49 frames out of 250 will probably survive.

CSMA

- CSMA was developed to minimize the chance of collision and, therefore, increase theperformance.
- CSMA is based on the principle "sense before transmit" or "listen before talk."
- Here is how it works:
  1) Each station checks the state of the medium: idle or busy.
  2) i) If the medium is idle, the station sends the data.
     i) If the medium is busy, the station defers sending.
- CSMA can reduce the possibility of collision, but it cannot eliminate it.

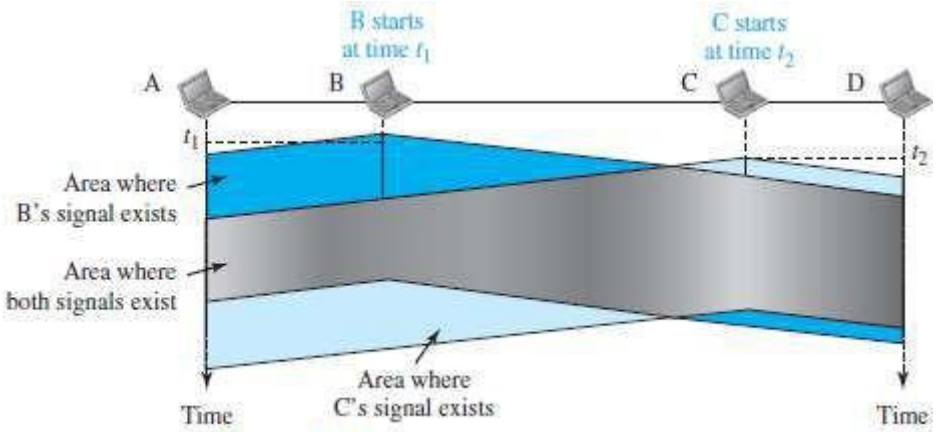


Figure 12.7 *Space/time model of a collision in CSMA*

- The possibility of collision still exists. For example:

  When a station sends a frame, it still takes time
  → for the first bit to reach every station and
  → for every station to sense it.

- For example: In Figure 12.7,
  - At time t1, station B senses & finds the medium idle, so sends a frame.
  - At time t2, station C senses & finds the medium idle, so sends a frame.
  - The 2 signals from both stations B & C collide and both frames are destroyed.

Vulnerable Time

- The vulnerable time is the propagation time T_p (Figure12.8).
- The propagation time is the time needed for a signal to propagate from one end of the medium to the other.
- Collision occurs when

  → a station sends a frame, and
  → other station also sends a frame during propagation time

- If the first bit of the frame reaches the end of the medium, every station will refrain from sending.

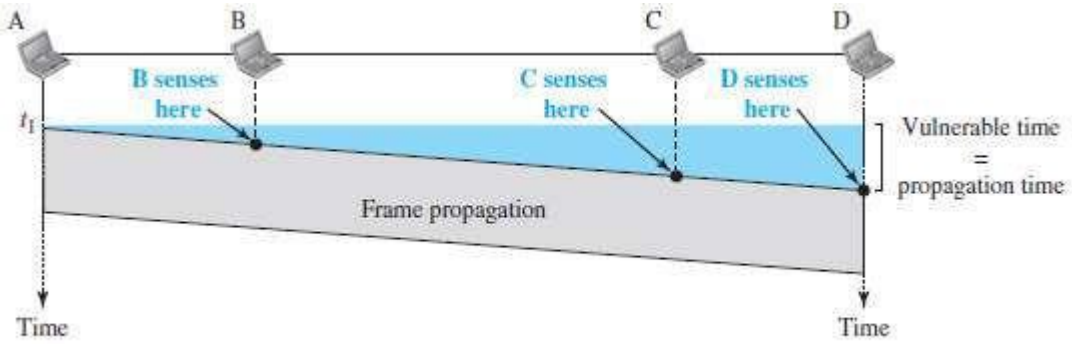


Figure 12.8 *Vulnerable time in CSMA*

Persistence Methods

- Q: What should a station do if the channel is busy or idle? Three methods can be used to answer this question:
  1) 1-persistent method 2) Non-persistent method and 3) p-persistent method

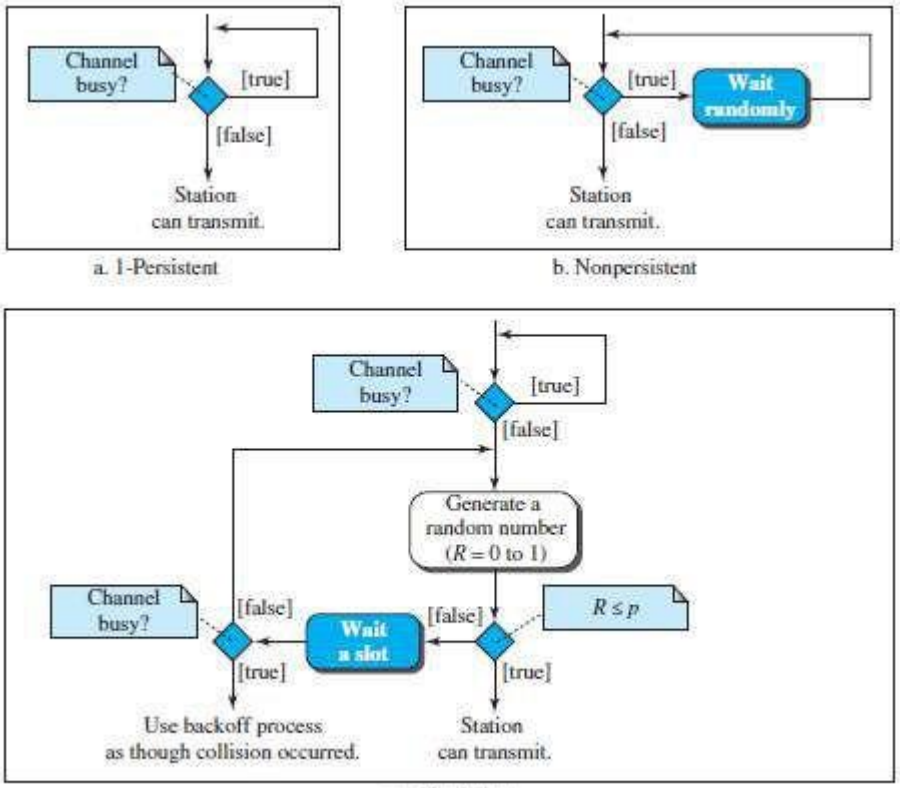


Figure 12.10 *Flow diagram for three persistence methods*

1) 1-Persistent

- Before sending a frame, a station senses the line (Figure 12.10a).
  i) If the line is idle, the station sends immediately (with probability = 1).
  ii) If the line is busy, the station continues sensing the line.
- This method has the highest chance of collision because 2 or more stations:
  → may find the line idle and
  → send the frames immediately.

2) Non-Persistent

- Before sending a frame, a station senses the line (Figure 12.10b).
  i) If the line is idle, the station sends immediately.
  ii) If the line is busy, the station waits a random amount of time and then senses the line again.
- This method reduces the chance of collision because 2 or more stations:
  → will not wait for the same amount of time and
  → will not retry to send simultaneously.

3) P-Persistent

- This method is used if the channel has time-slots with a slot-duration equal to or greater than the maximum propagation time (Figure 12.10c).
- Advantages:
  i) It combines the advantages of the other 2 methods.
  ii) It reduces the chance of collision and improves efficiency.
- After the station finds the line idle, it follows these steps:
  1) With probability p, the station sends the frame.
  2) With probability q=1-p, the station waits for the beginning of the next time-slot and checks the line again.
     i) If line is idle, it goes to step 1.
     ii) If line is busy, it assumes that collision has occurred and uses the back off procedure.

CSMA/CD

- Disadvantage of CSMA: CSMA does not specify the procedure after a collision has occurred. Solution: CSMA/CD enhances the CSMA to handle the collision.
- Here is how it works (Figure 12.12):
  1) A station
     → sends the frame &
     → then monitors the medium to see if the transmission was successful or not.
  2) If the transmission was unsuccessful (i.e. there is a collision), the frame is sent again.

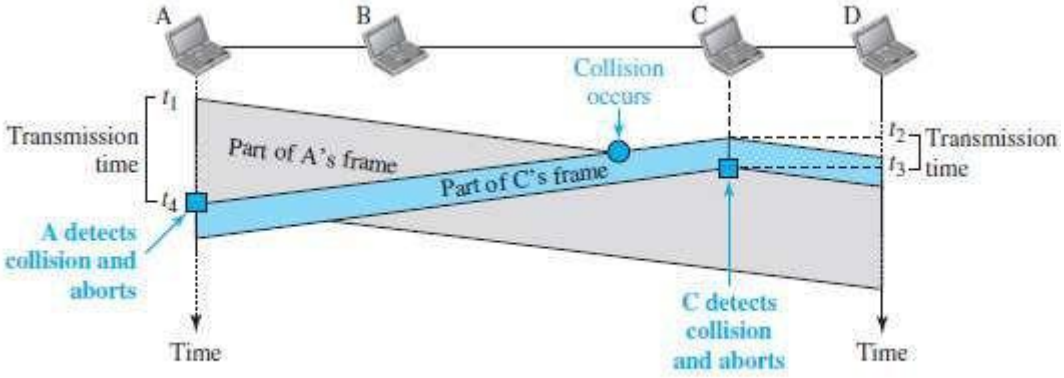


Figure 12.12 *Collision and abortion in CSMA/CD*

- In the Figure 12.11,
  - At time t_1, station A has executed its procedure and starts sending the bits of its frame.
  - At time t_2, station C has executed its procedure and starts sending the bits of its frame.
  - The collision occurs sometime after time t_2.
  - Station C detects a collision at time t_3 when it receives the first bit of A's frame. Station C immediately aborts transmission.
  - Station A detects collision at time t_4 when it receives the first bit of C's frame. Station A also immediately aborts transmission.
- Station A transmits for the duration $t_4 - t_1$. Station C transmits for the duration $t_3 - t_2$.
- For the protocol to work:
  The length of any frame divided by the bit rate must be more than either of these durations.

Minimum Frame Size

- For CSMA/CD to work, there is a need to restrict the frame-size.
- Before sending the last bit of the frame, the sender must
  → detect a collision and
  → abort the transmission.
- This is so because the sender
  → does not keep a copy of the frame and
  → does not monitor the line for collision-detection.
- Frame transmission time T_{fr} is given by
  $T_{fr} = 2T_p$ where T_p=maximum propagation time

Example 4.4

A network using CSMA/CD has a bandwidth of 10 Mbps. If the maximum propagation time (including the delays in the devices and ignoring the time needed to send a jamming signal, as we see later) is 25.6 μs, what is the minimum size of the frame?

Solution

The minimum frame transmission time is $T_{fr} = 2 \times T_p = 51.2$ μs. This means, in the worst case, a station needs to transmit for a period of 51.2 μs to detect the collision. The minimum size of the frame is 10 Mbps × 51.2 μs = 512 bits or 64 bytes. This is actually the minimum size of the frame for Standard Ethernet, as we will see later in the chapter.

Procedure

- CSMA/CD is similar to ALOHA with 2 differences (Figure 12.13):
  2) Addition of the persistence process.
     ¤ It need to sense the channel before sending the frame by using non-persistent, 1-persistent or p-persistent.
  3) Frame transmission.
     i) In ALOHA, first the entire frame is transmitted and then acknowledgment is waited for.
     ii) In CSMA/CD, transmission and collision-detection is a continuous process.

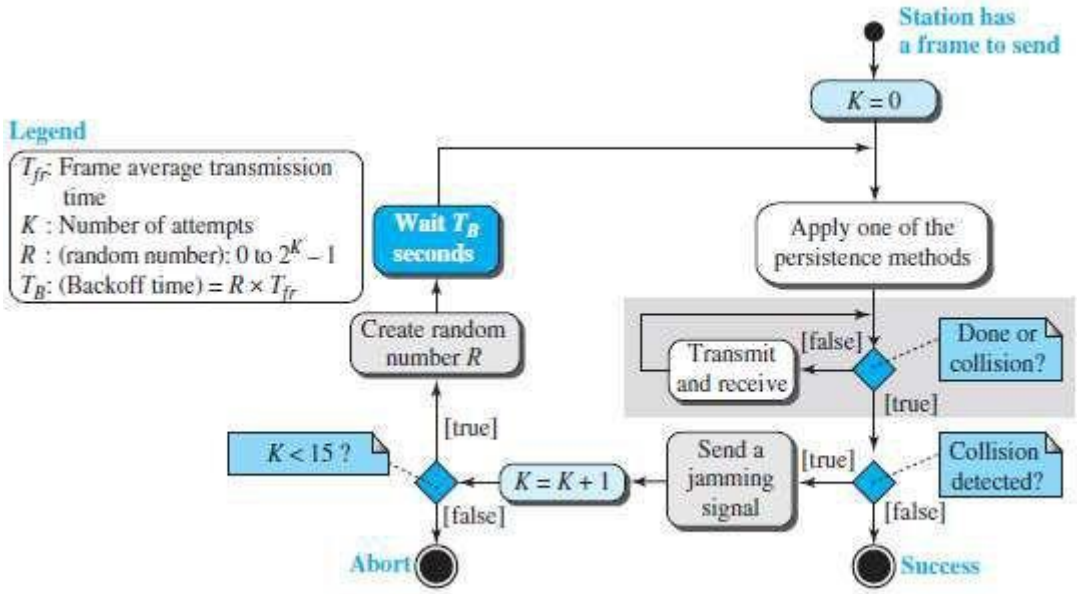


Figure 12.13 *Flow diagram for the CSMA/CD*

- A sender needs to monitor the energy-level to determine if the channel is
  → Idle
  → Busy or
  → Collision mode

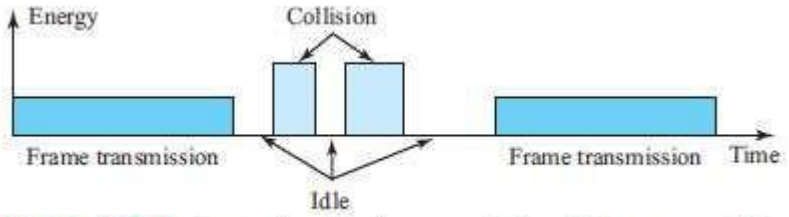


Figure 12.14 *Energy level during transmission, idleness, or collision*

Throughput

- The throughput of CSMA/CD is greater than pure or slotted ALOHA.
- The maximum throughput is based on
  → different value of G
  → persistence method used (non-persistent, 1-persistent, or p-persistent) and
  → 'p' value in the p-persistent method.
- For 1-persistent method, the maximum throughput is 50% when G =1.
- For non-persistent method, the maximum throughput is 90% when G is between 3 and 8.

CSMA/CA

- Here is how it works (Figure 12.15):
  1) A station needs to be able to receive while transmitting to detect a collision.
     i) When there is no collision, the station receives one signal: its own signal.
     ii) When there is a collision, the station receives 2 signals:
        a) Its own signal and
        b) Signal transmitted by a second station.
  2) To distinguish b/w these 2 cases, the received signals in these 2 cases must be different.

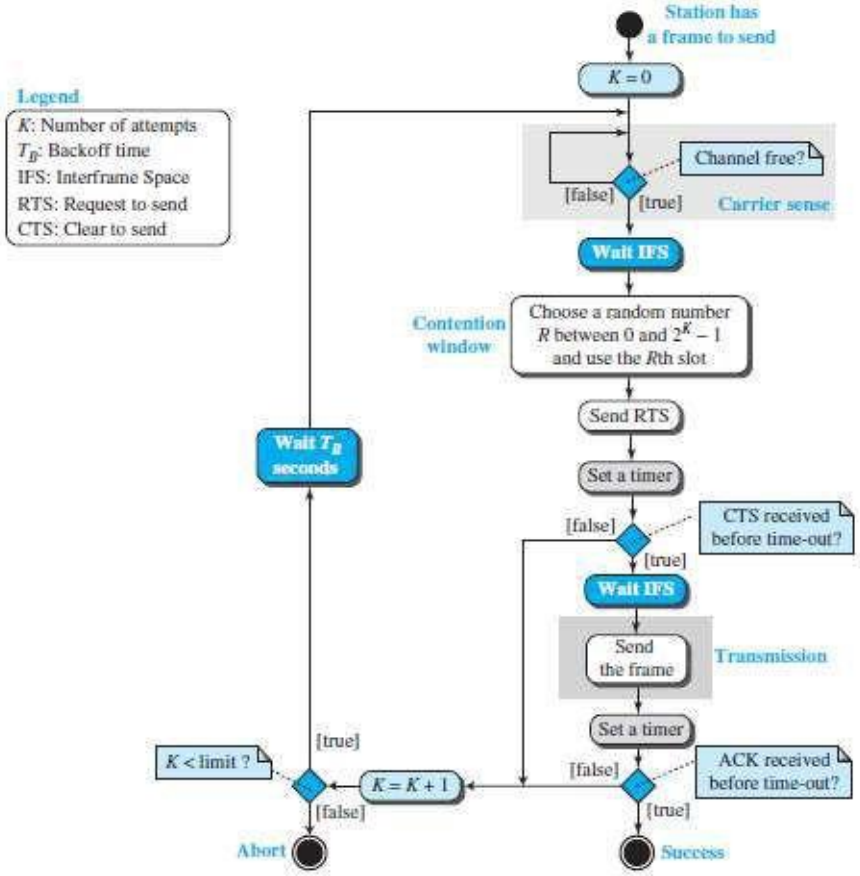


Figure 12.15 *Flow diagram of CSMA/CA*

- CSMA/CA was invented to avoid collisions on wireless networks.
- Three methods to avoid collisions (Figure 12.16):
  1) Inter frame space 2) Contention window and 3) Acknowledgments

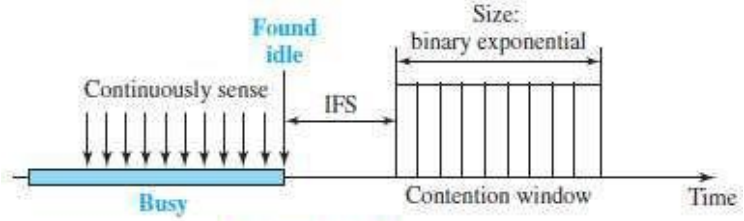


Figure 12.16 *Contention window*

1) Inter frame Space (IFS)

- Collisions are avoided by deferring transmission even if the channel is found idle.
- When the channel is idle, the station does not send immediately.
  Rather, the station waits for a period of time called the inter-frame space or IFS.
- After the IFS time,
  if the channel is still idle,
  then, the station waits for the contention-time &
  finally, the station sends the frame.
- IFS variable can also be used to prioritize stations or frame types.
  For example, a station that is assigned a shorter IFS has a higher priority.

2) Contention Window

- The contention-window is an amount of time divided into time-slots.
- A ready-station chooses a random-number of slots as its wait time.
- In the window, the number of slots changes according to the binary exponential back-off strategy.
- For example:

  At first time, number of slots is set to one slot and
  Then, number of slots is doubled each time if the station cannot detect an idle channel.

3) Acknowledgment

- There may be a collision resulting in destroyed-data.
- In addition, the data may be corrupted during the transmission.
- To help guarantee that the receiver has received the frame, we can use
  i) Positive acknowledgment and
  ii) Time-out timer

Frame Exchange Time Line

- Two control frames are used:
  1) Request to send (RTS)
  2) Clear to send (CTS)
- The procedure for exchange of data and control frames in time (Figure 12.17):
  1) The source senses the medium by checking the energy level at the carrier frequency.
     ii) If the medium is idle,
     then the source waits for a period of time called the DCF interframe space (DIFS);
     finally, the source sends a RTS.
  2) The destination
     → receives the RTS
     → waits a period of time called the short inter frame space (SIFS)
     → sends a control frame CTS to the source.
     CTS indicates that the destination station is ready to receive data.
  3) The source
     → receives the CTS
     → waits a period of time SIFS
     → sends a data to the destination
  4) The destination
     → receives the data
     → waits a period of time SIFS
     → sends a acknowledgment ACK to the source.
     ACK indicates that the destination has been received the frame.

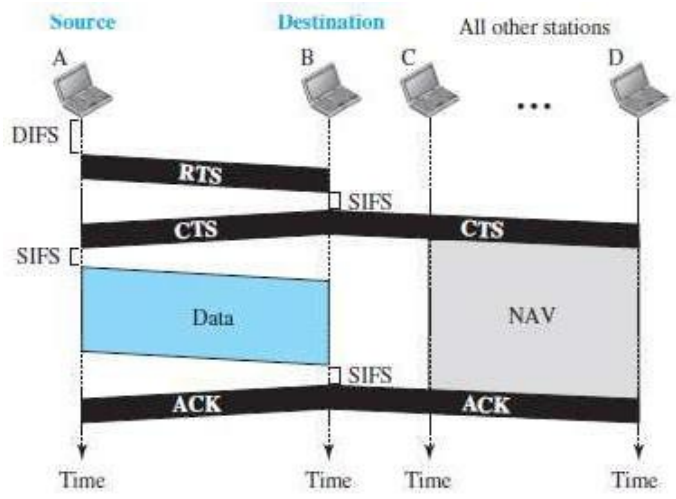


Figure 12.17 *CSMA/CA and NAV*

Network Allocation Vector

- When a source-station sends an RTS, it includes the duration of time that it needs to occupy the channel.
- The remaining stations create a timer called a network allocation vector (NAV).
- NAV indicates waiting time to check the channel for idleness.
- Each time a station accesses the system and sends an RTS frame, other stations start their NAV.

Collision during Handshaking

- Two or more stations may try to send RTS at the same time.
- These RTS may collide.
- The source assumes there has been a collision if it has not received CTS from the destination.
- The back off strategy is employed, and the source tries again.

Hidden Station Problem

- Figure 12.17 also shows that the RTS from B reaches A, but not C.
- However, because both B and C are within the range of A, the CTS reaches C.
- Station C knows that some hidden station is using the channel and refrains from transmitting until that duration is over.

CSMA/CA and Wireless Networks

- CSMA/CA was mostly intended for use in wireless networks.

However, it is not sophisticated enough to handle some particular issues related to wireless networks, such as hidden terminals

CONTROLLED ACCESS PROTOCOLS

- Here, the stations consult one another to find which station has the right to send.
- A station cannot send unless it has been authorized by other stations.
- Three popular controlled-access methods are: 1) Reservation 2) Polling 3) TokenPassing

Reservation

- Before sending data, each station needs to make a reservation of the medium.
- Time is divided into intervals.
- In each interval, a reservation-frame precedes the data-frames.
- If no. of stations = N, then there are N reservation mini-slots in the reservation-frame.
- Each mini-slot belongs to a station.
- When a station wants to send a data-frame, it makes a reservation in its own minislot.
- The stations that have made reservations can send their data-frames.

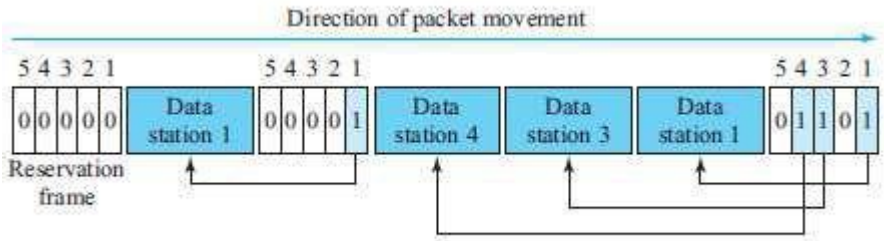


Figure 12.18 *Reservation access method*

- For example (Figure 12.18):
  - 5 stations have a 5-minislot reservation-frame.
  - In the first interval, only stations 1, 3, and 4 have made reservations.
  - In the second interval, only station-1 has made a reservation.

Polling

- In a network,
  One device is designated as a primary station and
  Other devices are designated as secondary stations.
- Functions of primary-device:
  1) The primary-device controls the link.
  2) The primary-device is always the initiator of a session.
  3) The primary-device is determining which device is allowed to use the channel at a given time.
  4) All data exchanges must be made through the primary-device.
- The secondary devices follow instructions of primary-device.
- Disadvantage: If the primary station fails, the system goes down.
- Poll and select functions are used to prevent collisions (Figure 12.19).

1) Select

- If the primary wants to send data, it tells the secondary to get ready to receive; this is called select function.
- The primary
  → alerts the secondary about upcoming transmission by sending select frame (SEL)
  → then waits for an acknowledgment (ACK) from secondary
  → then sends the data frame and
  → finally waits for an acknowledgment (ACK) from the secondary.

2) Poll

- If the primary wants to receive data, it asks the secondaries if they have anything to send; this is called poll function.
- When the first secondary is approached, it responds either
  → with a NAK frame if it has no data to send or
  → with data-frame if it has data to send.

i) If the response is negative (NAK frame), then the primary polls the next secondary in the same manner.
ii) When the response is positive (a data-frame), the primary
  → reads the frame and
  → returns an acknowledgment (ACK frame).

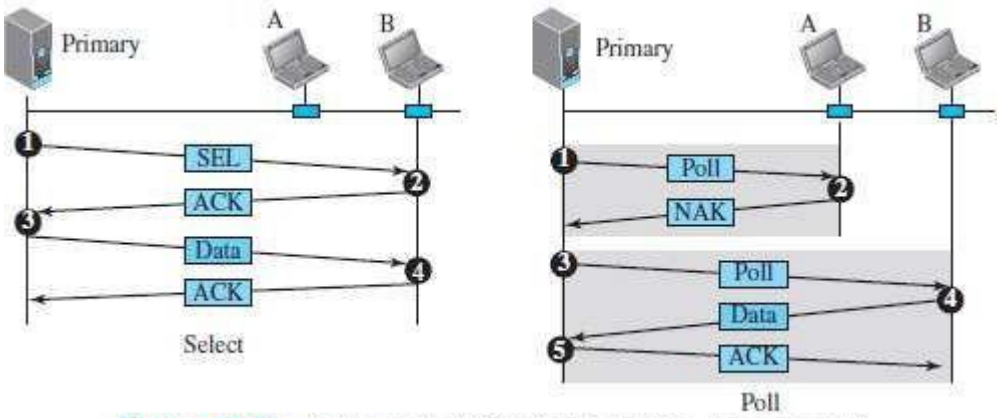


Figure 12.19 *Select and poll functions in polling-access method*

Token Passing

- In a network, the stations are organized in a ring fashion i.e. for each station; there is a predecessor and a successor.
    5) The predecessor is the station which is logically before the station in the ring.
    6) The successor is the station which is after the station in the ring.
- The current station is the one that is accessing the channel now.
- A token is a special packet that circulates through the ring.
- Here is how it works:
    - A station can send the data only if it has the token.
    - When a station wants to send the data, it waits until it receives the token from its predecessor.
    - Then, the station holds the token and sends its data.
    - When the station finishes sending the data, the station
        → releases the token
        → passes the token to the successor.
- Main functions of token management:
    1) Stations must be limited in the time they can hold the token.
    2) The token must be monitored to ensure it has not been lost or destroyed.
        For ex: if a station that is holding the token fails, the token will disappear from the network
    3) Assign priorities
        → to the stations and
        → to the types of data being transmitted.
    4) Make low-priority stations release the token to high priority stations.

Logical Ring

- In a token-passing network, stations do not have to be physically connected in a ring; the ring can be a logical one.
- Four physical topologies to create a logical ring (Figure 12.20):
    1) Physical ring
    2) Dual ring
    3) Bus ring
    4) Star ring

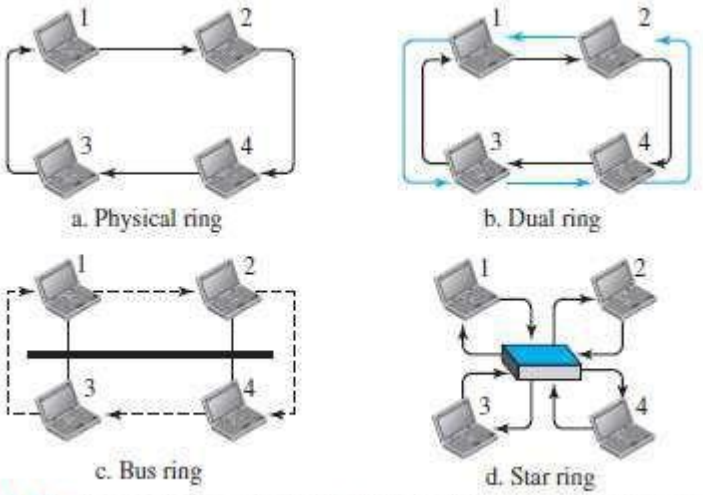


Figure 12.20 *Logical ring and physical topology in token-passing access method*

1) Physical Ring Topology

- When a station sends token to its successor, token cannot be seen by other stations. (Figure 12.20a)
- This means that the token does not have the address of the next successor.
- Disadvantage: If one of the links fails, the whole system fails.

2) Dual Ring Topology

- A second (auxiliary) ring
  - → is used along with the main ring (Figure 12.20b).
  - → operates in the reverse direction compared with the main ring.
  - → is used for emergencies only (such as a spare tire for a car).
- If the main ring fails, the system automatically combines the 2 rings to form a temporary ring.
- After the failed link is restored, the second ring becomes idle again.
- Each station needs to have 2 transmitter-ports and 2 receiver-ports.
- This topology is used in
  - i) FDDI (Fiber Distributed Data Interface) and
  - ii) CDDI (Copper Distributed Data Interface).

3) Bus Ring Topology

- The stations are connected to a single cable called a bus (Figure 12.20c).
- This makes a logical ring, because each station knows the address of its successor and predecessor.
- When a station has finished sending its data, the station
  - → releases the token and
  - → inserts the address of its successor in the token.
- Only the station gets the token to access the shared media.
- This topology is used in the Token Bus LAN.

4) Star Ring Topology

- The physical topology is a star (Figure 12.20d).
- There is a hub that acts as the connector.
- The wiring inside the hub makes the ring i.e. the stations are connected to the ring through the 2 wire connections.
- Disadvantages:
  1) This topology is less prone to failure because If a link goes down,
     then the link will be bypassed by the hub and the rest of the stations can operate.
  2) Also adding and removing stations from the ring is easier.
- This topology is used in the Token Ring LA

CHANNELIZATION PROTOCOLS

- Channelization is a multiple-access method.
- The available bandwidth of a link is shared b/w different stations in time, frequency, or through code.
- Three channelization protocols:
    1) FDMA (Frequency Division Multiple Access)
    2) TDMA (Time Division Multiple Access) and
    3) CDMA (Code Division Multiple Access)

FDMA

- The available bandwidth is divided into frequency-bands (Figure 12.21).
- Each band is reserved for a specific station.
- Each station can send the data in the allocated band.
- Each station also uses a bandpass filter to confine the transmitter frequencies.
- To prevent interferences, small guard bands are used to separate the allocated bands from one another.

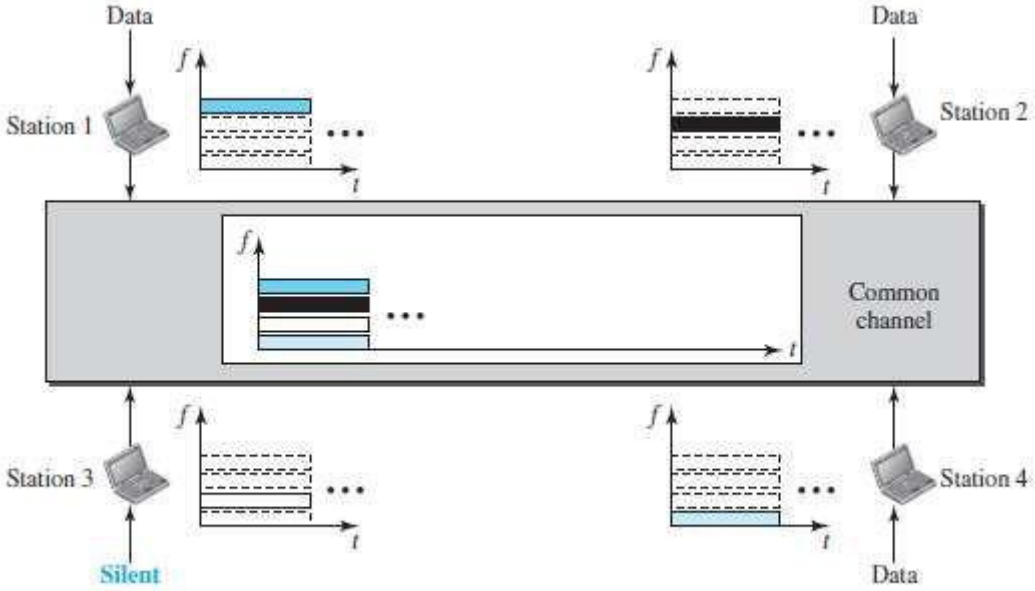


Figure 12.21 *Frequency-division multiple access (FDMA)*

- FDM vs. FDMA

1) FDM

- FDM is a multiplexing method in the physical layer.
- FDM
    → combines individual-loads from low-bandwidth channels and
    → transmits aggregated-load by using a high-bandwidth channel.
- The channels that are combined are low-pass.
- The multiplexer
    → modulates & combines the signals and
    → creates a band pass signal.
- The bandwidth of each channel is shifted by the multiplexer.

2) FDMA

- FDMA is an access method in the data link layer.
- In each station, the data link layer tells the physical layer to make a band pass signal from the data passed to it.
- The signal must be created in the allocated band.
- There is no physical multiplexer at the physical layer.
- The signals created at each station are automatically band pass-filtered.
- They are mixed when they are sent to the common channel.

TDMA

- The stations share the bandwidth of the channel in time (Figure 12.22).
- Each time-slot is reserved for a specific station.

- Each station can send the data in the allocated time-slot.

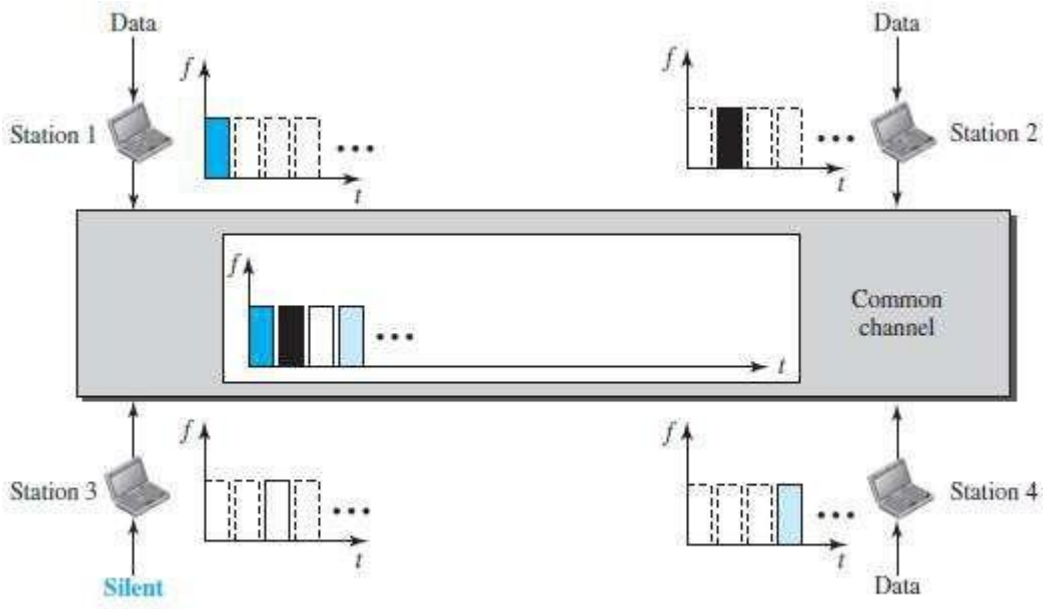


Figure 12.22 *Time-division multiple access (TDMA)*

- Main problem: Achieving synchronization between the different stations.
  i.e. each station needs to know the beginning of its slot and the location of its slot.
  This may be difficult because of propagation delays introduced in the system.
- To compensate for the delays, insert guard-times.
- Normally, synchronization is accomplished by having some synchronization bits at the beginning of each slot.
- TDMA vs. TDM

1) TDM

- TDM is a multiplexing method in the physical layer.
- TDM
  → combines the individual-data from slower channels and
  → transmits the aggregated- data by using a faster channel.
- The multiplexer interleaves data units from each channel.

2) TDMA

- TDMA is an access method in the data link layer.
- In each station, the data link layer tells the physical layer to use the allocated time-slot.
- There is no physical multiplexer at the physical layer.

CDMA

- CDMA simply means communication with different codes.
- CDMA differs from FDMA because
  → only one channel occupies the entire bandwidth of the link.
- CDMA differs from TDMA because
  → all stations can send data simultaneously; there is no timesharing.

(Analogy: CDMA simply means communication with different codes.
For example, in a large room with many people, 2 people can talk privately in English if nobody else understands English. Another 2 people can talk in Chinese if they are the only ones who understand Chinese, and so on).

Implementation

- Let us assume there are 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_2, and so on.
- Assume that the assigned codes have 2 properties.
  If we multiply each code by another, we get 0.
  If we multiply each code by itself, we get 4 (the number of stations).

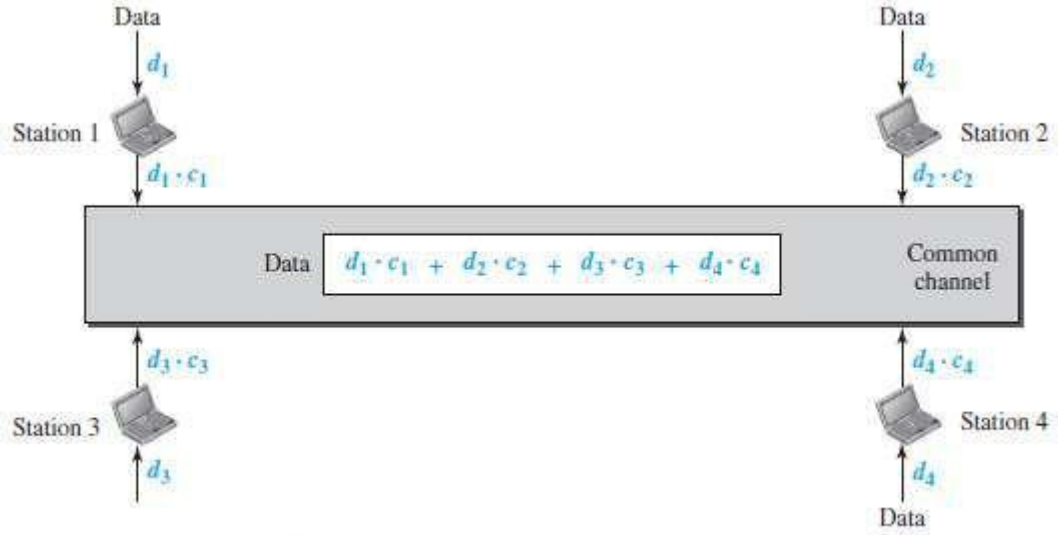


Figure 12.23 *Simple idea of communication with code*

- Here is how it works (Figure 12.23):
  - Station-1 multiplies the data by the code to get $d_1.c_1$.
  - Station-2 multiplies the data by the code to get $d_2.c_2$. And so on.
  - The data that go on the channel are the sum of all these terms.

$$d_1 \cdot c_1 + d_2 \cdot c_2 + d_3 \cdot c_3 + d_4 \cdot c_4$$

  - The receiver multiplies the data on the channel by the code of the sender.
  - For example, suppose stations 1 and 2 are talking to each other.
  - Station-2 wants to hear what station-1 is saying.
  - Station-2 multiplies the data on the channel by c_1 the code of station-1. $(c_1.c_1)=4$, $(c_2.c_1)=0$, $(c_3.c_1)=0$, and $(c_4.c_1)=0$,

Therefore, station-2 divides the result by 4 to get the data from station-1.

$$\begin{aligned}\text{data} &= (d_1 \cdot c_1 + d_2 \cdot c_2 + d_3 \cdot c_3 + d_4 \cdot c_4) \cdot c_1 \\ &= d_1 \cdot c_1 \cdot c_1 + d_2 \cdot c_2 \cdot c_1 + d_3 \cdot c_3 \cdot c_1 + d_4 \cdot c_4 \cdot c_1 = 4 \times d_1\end{aligned}$$

Chips

- CDMA is based on coding theory.
- Each station is assigned a code, which is a sequence of numbers called chips (Figure 12.24).

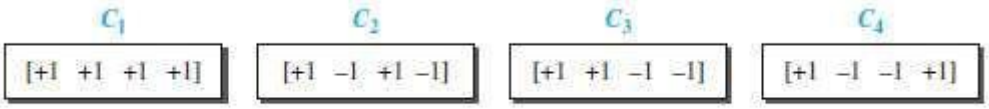


Figure 12.24 *Chip sequences*

- These sequences were carefully selected & are called orthogonal sequences
- These sequences have the following properties:

  Each sequence is made of N elements, where N is the number of stations.

  Multiplication of a sequence by a scalar:

  If we multiply a sequence by a number i.e. every element in the sequence is multiplied by that element.

  For example,

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

  Inner product of 2 equal sequences:

  If we multiply 2 equal sequences, element by element, and add the results, we get N, where N is the number of elements in each sequence.

  For example,

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

  Inner product of 2 different sequences:

  If we multiply 2 different sequences, element by element, and add the results, we get 0.

  For example,

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

  Adding 2 sequences means adding the corresponding elements. The result is another sequence.

  For example,

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

Data Representation

- It follows the following rules for encoding:

  To send a 0 bit, a station encodes the bit as -1

  To send a 1 bit, a station encodes the bit as +1

  When a station is idle, it sends no signal, which is interpreted as a 0.

DATA-LINK LAYER

INTRODUCTION

The Internet is a combination of networks glued together by connecting devices (routers or switches). If a packet is to travel from a host to another host, it needs to pass through these networks.. Communication at the data-link layer is made up of five separate logical connections between the data-link layers in the path.

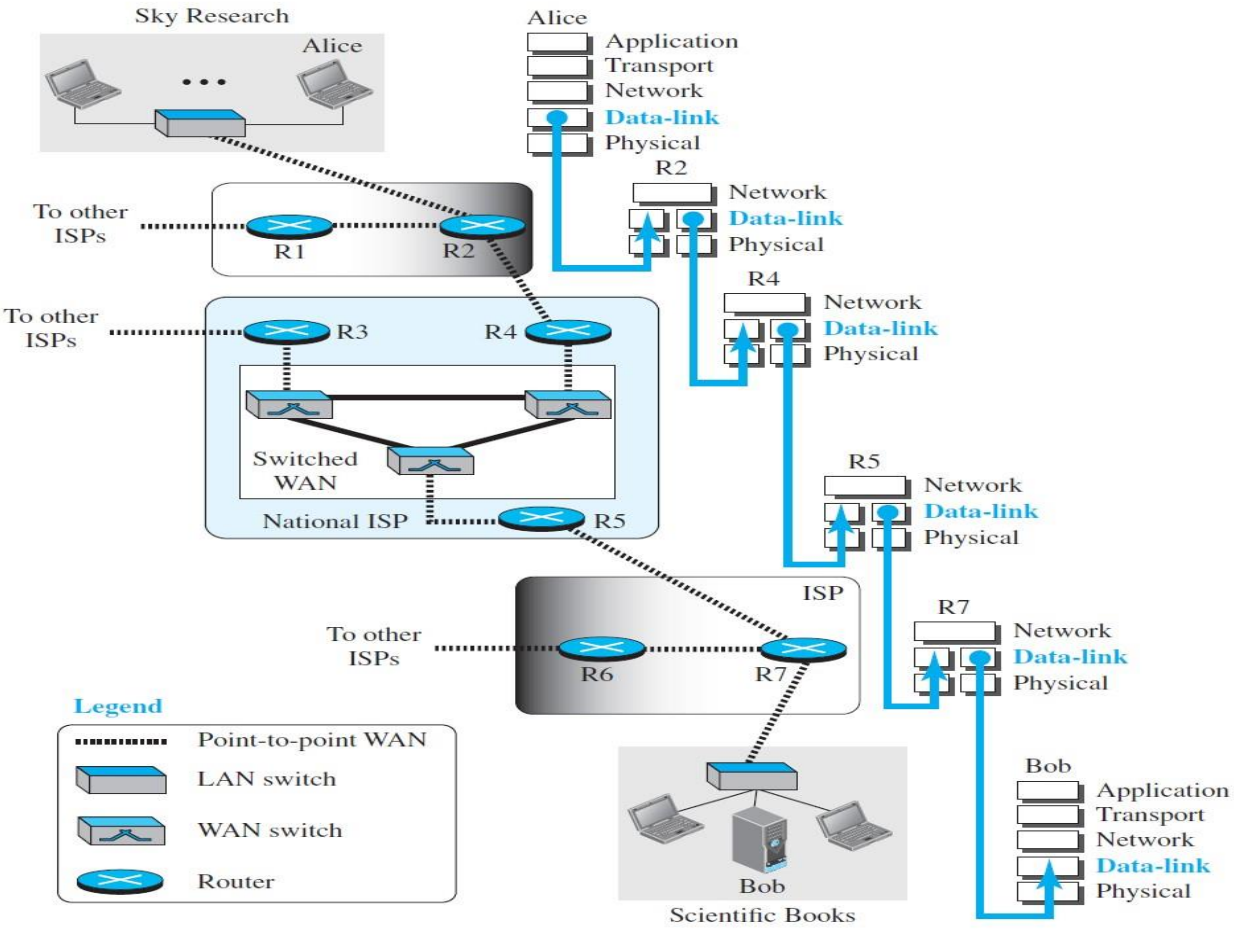


fig: communication at the data-link layer

The data-link layer at Alice's computer communicates with the data-link layer at router R2. The data-link layer at router R2 communicates with the data-link layer at router R4,

and so on. Finally, the data-link layer at router R7 communicates with the data-link layer at Bob's computer. Only one data-link layer is involved at the source or the destination, but two data-link layers are involved at each router. The reason is that Alice's and Bob's computers are

each connected to a single network, but each router takes input from one network and sends output to another network. Note that although switches are also involved in the data-link-layer communication, for simplicity we have not shown them in the figure.

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 is customary to refer to the two end hosts and the routers as nodes and the networks in between as links. Figure shows a simple representation of links and nodes when the path of the data unit is only six nodes.

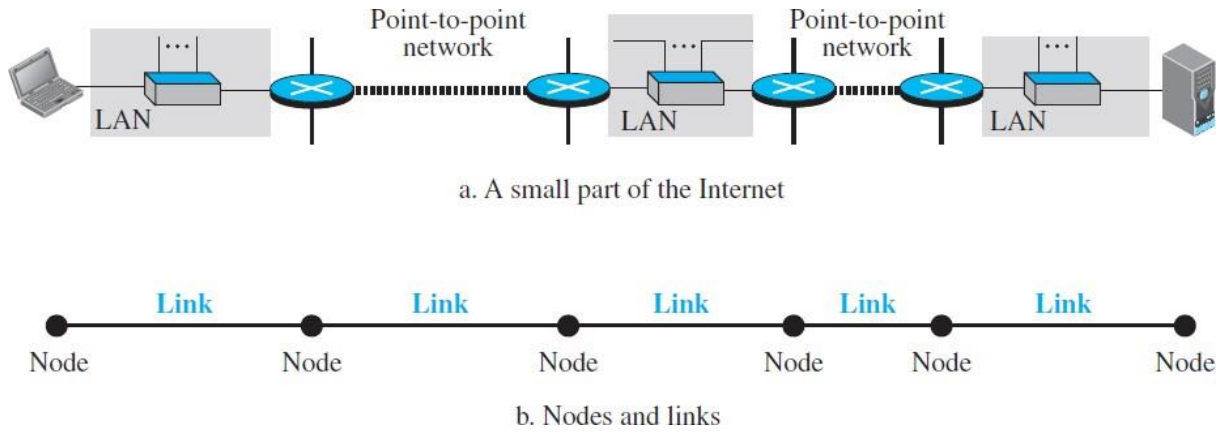


fig: links and nodes

The first node is the source host; the last node is the destination host. The other four nodes are four routers. The first, the third, and the fifth links represent the three LANs; the second and the fourth links represent the two WANs.

Services

The data-link layer is located between the physical and the network layers. The data link layer provides services to the network layer; it receives services from the physical layer.

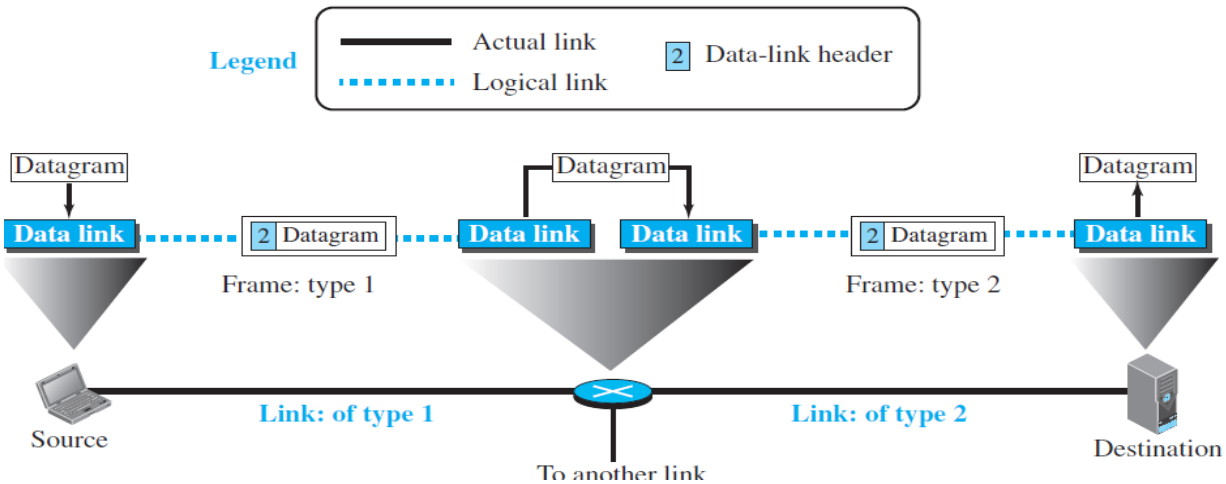


fig: communication with only three nodes

Framing

The first service provided by the data-link layer is framing. The data-link layer at each node needs to encapsulate the datagram (packet received from the network layer) 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. Although we have shown only a header for a frame, frame may have both a header and a trailer. Different data-link layers have different formats for framing.

Flow Control

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). Definitely, we cannot have an unlimited buffer size at the receiving side. We have two choices.

The first choice is to let the receiving data-link layer drop the frames if its buffer is full. The second choice is to let the receiving data-link layer send a feedback to the sending data-link layer to ask it to stop or slow down.

Different data-link-layer protocols use different strategies for flow control. Since flow control also occurs at the transport layer, with a higher degree of importance.

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. Since error detection and correction is an issue in every layer (node-to node or host-to-host).

Congestion Control

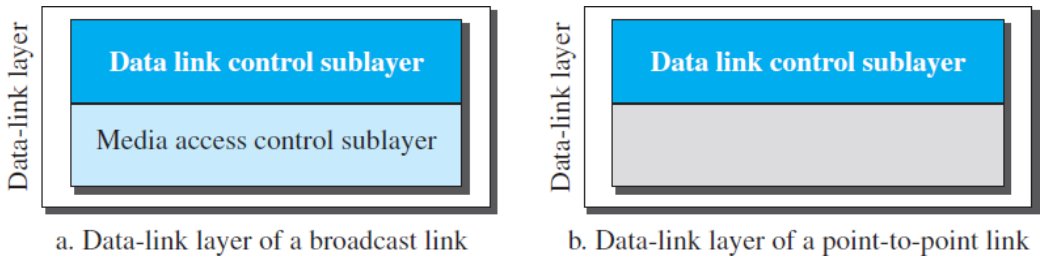
Although a link may be congested with frames, which may result in frame loss, most data-link-layer protocols do not directly use a congestion control to alleviate congestion, although some wide-area networks do. In general, congestion control is considered an issue in the network layer or the transport layer because of its end-to-end nature.

Two Categories of Links

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. For example, when two friends use the traditional home phones to chat, they are using a point-to-point link; when the same two friends use their cellular phones, they are using a broadcast link (the air is shared among many cell phone users).

Two Sublayers

The data-link layer is divided into two sublayers: data link control (DLC) and media access control (MAC). LAN protocols actually use the same strategy. **The data link control sublayer** deals with all issues common to both point-to-point and broadcast links; **the media access control sublayer** deals only with issues specific to broadcast links. In other words, we separate these two types of links at the data-link layer, as shown in fig



a. Data-link layer of a broadcast link

b. Data-link layer of a point-to-point link

fig: dividing the data link layer into two sublayers

LINK-LAYER ADDRESSING

In a connectionless internetwork such as the Internet we cannot make a datagram reach its destination using only IP addresses. The reason is that each datagram in the Internet, from the same source host to the same destination host, may take a different path. The source and

destination IP addresses define the two ends but cannot define which links the datagram should pass through. We need to remember that the IP addresses in a datagram should not be changed. If the destination IP address in a datagram changes, the packet never reaches its destination; if the source IP address in a datagram changes, the destination host or a router can never communicate with the source if a response needs to be sent back or an error needs to be reported back to the source (ICMP).

The above discussion shows that we need another addressing mechanism in a connectionless internetwork: the link-layer addresses of the two nodes. 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 demonstrates the concept in a small internet.

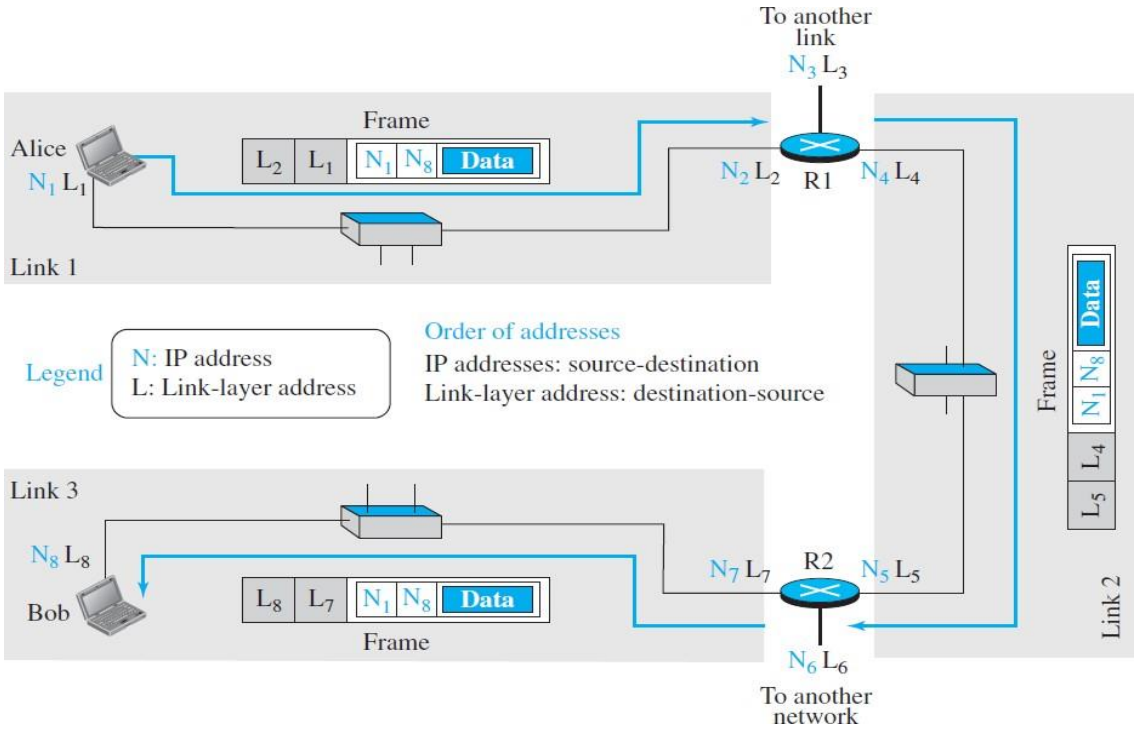


fig: IP address and link layer addresses in a small internet

above Figure shows three links and two routers and also have only two hosts: Alice (source) and Bob (destination). For each host, two addresses, the IP addresses (N) and the link-layer addresses (L) are shown. Note that a router has as many pairs of addresses as the number of links the router is connected to. We have shown three frames, one in each link. Each frame

carries the same datagram with the same source and destination addresses (N1 and N8), but the link-layer addresses of the frame change from link to link.

In link 1, the link-layer addresses are L1 and L2. In link 2, they are L4 and L5. In link 3, they are L7 and L8. Note that the IP addresses and the link-layer addresses are not in the same order. For IP addresses, the source address comes before the destination address; for link-layer addresses, the destination address comes before the source. The datagrams and frames are designed in this way, and we follow the design. We may raise several questions:

❑ The IP address of a router does not appear in any datagram sent from a source to a destination, why do we need to assign IP addresses to routers? The answer is that in some protocols a router may act as a sender or receiver of a datagram. For example, in routing protocols a router is a sender or a receiver of a message. The communications in these protocols are between routers.

❑ Why do we need more than one IP address in a router, one for each interface? The answer is that an interface is a connection of a router to a link. We will see that an IP address defines a point in the Internet at which a device is connected. A router with n interfaces is connected to the Internet at n points. This is the situation of a house at the corner of a street with two gates; each gate has the address related to the corresponding street.

❑ How are the source and destination IP addresses in a packet determined? The answer is that the host should know its own IP address, which becomes the source IP address in the packet. the application layer uses the services of DNS to find the destination address of the packet and passes it to the network layer to be inserted in the packet.

❑ How are the source and destination link-layer addresses determined for each link? Again, each hop (router or host) should know its own link-layer address, The destination link-layer address is determined by using the Address Resolution Protocol.

❑ What is the size of link-layer addresses? The answer is that it depends on the protocol used by the link. Although we have only one IP protocol for the whole Internet, we may be using different data-link protocols in different links.

different link-layer protocols.

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.

Example: The unicast link-layer addresses in the most common LAN, Ethernet, are 48 bits (six bytes) that are presented as 12 hexadecimal digits separated by colons; for example, the following is a link-layer address of a computer.

A2:34:45:11:92:F1

Multicast Address: Some link-layer protocols define multicast addresses. Multicasting means one-to-many communication. However, the jurisdiction is local (inside the link).

Example: the multicast link-layer addresses in the most common LAN, Ethernet, are 48 bits (six bytes) that are presented as 12 hexadecimal digits separated by colons. The second digit, however, needs to be an even number in hexadecimal. The following shows a multicast address: A3:34:45:11:92:F1

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

Example : the broadcast link-layer addresses in the most common LAN, Ethernet, are 48 bits, all 1s, that are presented as 12 hexadecimal digits separated by colons. The following shows a broadcast address: FF:FF:FF:FF:FF:FF

Address Resolution Protocol (ARP)

Anytime a node has an IP datagram to send to another node in a link, it has the IP address of the receiving node. The source host knows the IP address of the default router. Each router except the last one in the path gets the IP address of the next router by using its forwarding table. The last router knows the IP address of the destination host. However, the IP address of the next node is not helpful in moving a frame through a link; we need the link-layer address of the next node. This is the time when the Address Resolution Protocol (ARP) becomes helpful.

The ARP protocol is one of the auxiliary protocols defined in the network layer, as shown in Figure . It belongs to the network layer, 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.

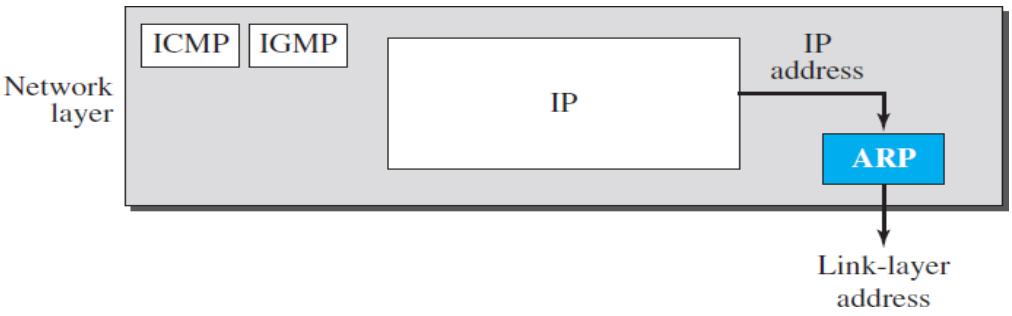


fig: Position of ARP in TCP/IP protocol suite

Anytime a host or a router needs to find the link-layer address of another host or router in its network, it sends an ARP request packet. The packet includes the link-layer and IP addresses of the sender and the IP address of the receiver. Because the sender does not know the link-layer address of the receiver, the query is broadcast over the link using the link-layer broadcast address.

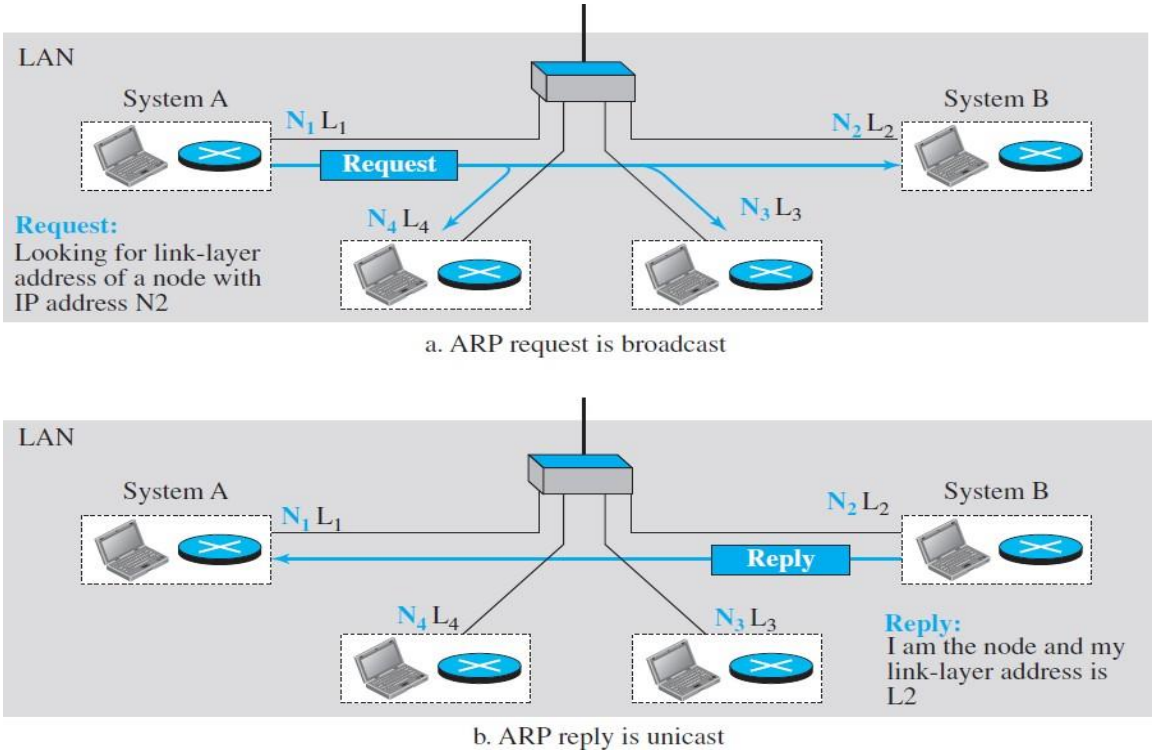


fig: ARP operation

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 (a) the system on the left (A) has a packet that needs to be delivered to another system (B) with IP address N2. 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 (b). 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.

Caching

Let us assume that there are 20 systems connected to the network (link): system A, system B, and 18 other systems. We also assume that system A has 10 datagrams to send to system B in one second.

a. Without using ARP, system A needs to send 10 broadcast frames. Each of the 18 other systems need to receive the frames, decapsulate the frames, remove the datagram and pass it to their network-layer to find out the datagrams do not belong to them.This means processing and discarding 180 broadcast frames.

b. Using ARP, system A needs to send only one broadcast frame. Each of the 18 other systems need to receive the frames, decapsulate the frames, remove the ARP message and pass the message to their ARP protocol to find that the frame must be discarded. This means processing and discarding only 18 (instead of 180) broadcast frames. After system B responds with its own data-link address, system A can store the link-layer address in its cache memory. The rest of the nine frames are only unicast. Since processing broadcast frames is expensive (time consuming), the first method is preferable.

Packet Format

Figure shows the format of an ARP packet.

The hardware type field - defines the type of the link-layer protocol; Ethernet given the type 1. The protocol type field - defines the network-layer protocol: IPv4 protocol is (0x0800).

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. The frame needs to have a field to show that the payload belongs to the ARP and not to the network-layer datagram

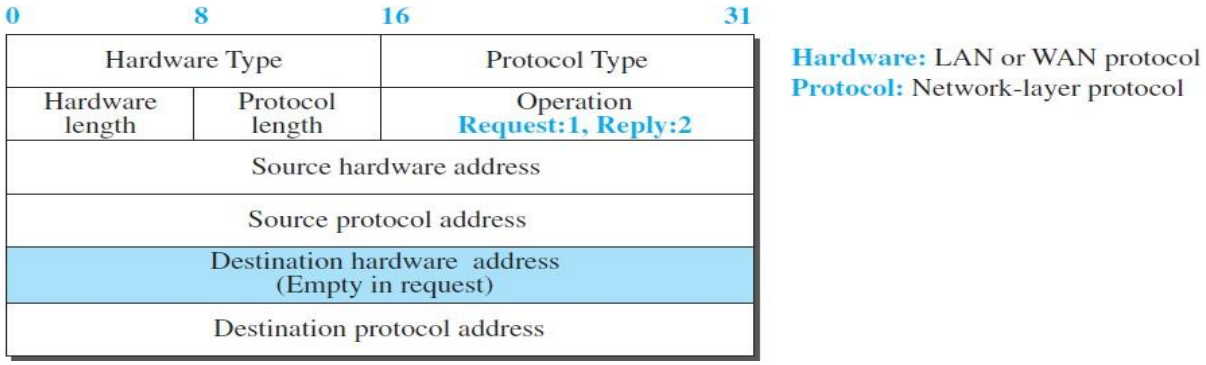
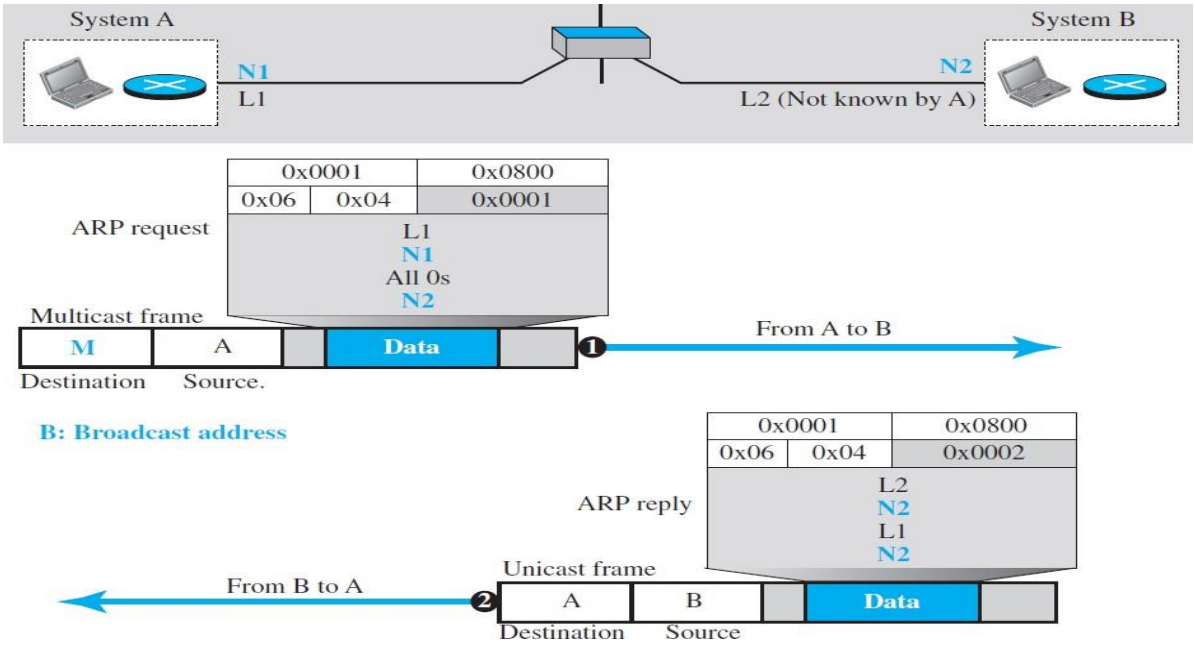


fig: ARP packet

Example : 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 shows the ARP request and response messages



IPv4 Addressing

- IP address is a numeric identifier assigned to each machine on the internet.
- IP address consists of two parts: network ID(NID) and host ID(HID).
  1) NID identifies the network to which the host is connected. All the hosts connected to the same network have the same NID.
  2) HID is used to uniquely identify a host on that network.
- HID is assigned by the network-administrator at the local site.
  NID for an organization may be assigned by the ISP (Internet Service Provider).
- IPv4 uses 32-bit addresses, i.e., approximately 4 billion addresses (2^{32}).
- IP addresses are usually written in dotted-decimal notation. The address is broken into four bytes.
  For example, an IP address of
  10000000 10000111 01000100 00000101
  is written as
  128.135.68.5
- IP address can be classified as
  1) Classful IP addressing &
  2) Classless IP addressing (CIDR → Classless Inter Domain Routing)

3.4.3.1 IPv4 Classful Addressing

- In classful addressing, the address space is divided into five classes: A, B, C, D and E.
- IP address class is identified by MSBs in binary.
- Classes A, B and C are used for unicast addressing. (Figure 3.13).
- Class D was designed for multicasting and class E is reserved.

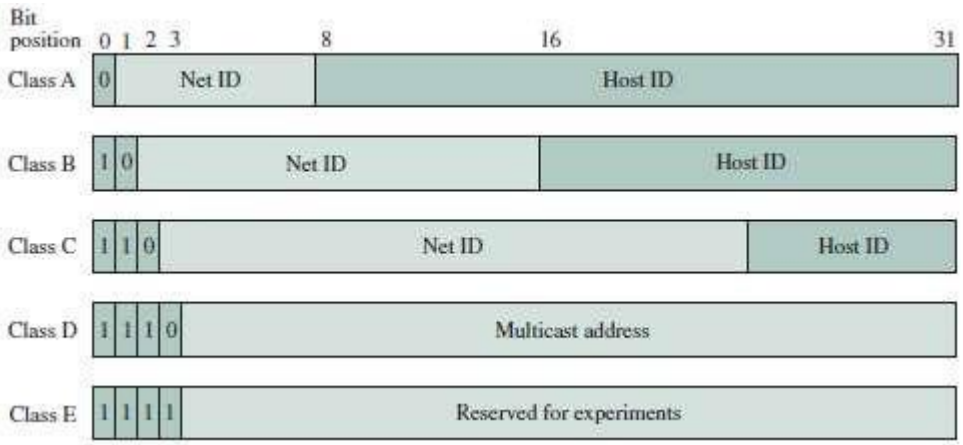


Figure 3.13: The five classes of IP addresses

Class	No. of networks	Max. No. of hosts per network	Designed for
A	126	$2^{24} - 2$	WAN
B	16,382	65,534	Campus networks
C	2^{21}	254	LAN

Table 3.3: Classful Addressing

- Analysis:
  - In c1assful addressing, a large part of the available addresses were wasted, since Class A and B were too large for most organizations (Table 3.3).
  - Class C is suited only for small organization and reserved addresses were sparingly used.

Subnet Addressing

- Problem with classful addressing:
  - Consider an organization has a Class B address which can support about 64,000 hosts.
  - It will be a huge task for the network-administrator to manage all 64,000 hosts.
- Solution: Use subnet addressing.
- Subnetting reduces the total number of network-numbers by assigning a single network-number to many adjacent physical networks.
- Each adjacent physical network is referred to as subnet. (Figure 3.14).
- All nodes on a subnet are configured with a subnet mask. For example: 255.255.255.0.
- The 1's in the subnet-mask represent the positions that refer to the network or subnet-numbers. The 0's represent the positions that refer to the host part of the address.
- The bitwise AND of IP address and its subnet mask gives the subnet number.
- Advantage:
  - The subnet-addressing scheme is oblivious to the network outside the organization.
  - Inside the organization the network-administrator is free to choose any combination of lengths for the subnet & host ID fields.

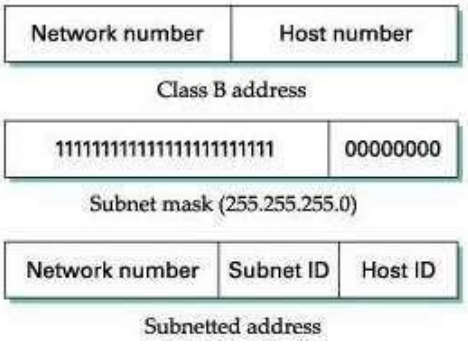


Figure 3.14: Subnet addressing

Question: If a packet with a destination IP address of 150.100.12.176 arrives at site from the outside network, which subnet should a router forward this packet to? Assume subnet mask is 255.255.255.128 (Figure 3.15).

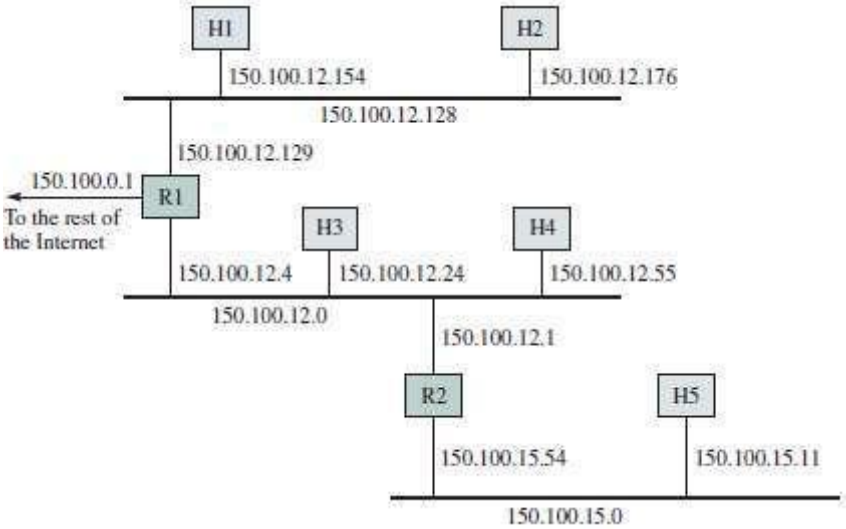


Figure 3.15: Example of address assignment with subnetting

Solution: The router can determine the subnet number by performing a binary AND between the subnet mask and the IP address.

IP address:	10010110 01100100 00001100	10110000(150.100.12.176)
Subnet mask:	11111111 11111111 11111111	10000000(255.255.255.128)
Subnet number:	10010110 01100100 00001100	10000000(150.100.12.128)

This number (150.100.12.128) is used to forward the packet to the correct subnet work inside the organization.

CIDR

- Problem with classful IP addressing:
  - Consider an organization needs about 500 hosts.
  - Obviously, the organization will get a Class B license, even though it has far fewer than 64,000 hosts.
  - At most, over 64,000 addresses can go unused.
  - This results in inefficient usage of the available address-space.
- Solution: Use CIDR (Classless Inter Domain Routing).
  - A single IP address can be used to designate many unique IP addresses. This is called supernetting.
  - A CIDR IP address looks like a normal IP address except that
    → the address ends with a slash followed by a number, called the IP network prefix.
    For ex: 205.100.0.0/22
  - CIDR addresses
    → reduce the size of routing-tables and
    → make more IP addresses available within organizations.

Obtaining a Block of Addresses

- To obtain a block of IP addresses for use within an organization's subnet, a network-administrator contacts the ISP.
- IP addresses are managed under the authority of the ICANN.
- The responsibility of the ICANN (Internet Corporation for Assigned Names and Numbers):
  → to allocate IP addresses,
  → to manage the DNS root servers.
  → to assign domain names and resolve domain name disputes.
  → to allocate addresses to regional Internet registries.

Obtaining a Host Address: DHCP

- Two ways to assign an IP address to a host:
  1) Manual Configuration
  - Operating systems allow system-administrator to manually configure IP address.

  2) Dynamic Host Configuration Protocol (DHCP)
  - DHCP enables auto-configuration of IP address to host.

DHCP Protocol

- DHCP enables auto-configuration of IP address to host.
- DHCP assigns dynamic IP addresses to devices on a network.
- Dynamic address allocation is required
  → when a host moves from one network to another or
  → when a host is connected to a network for the first time.

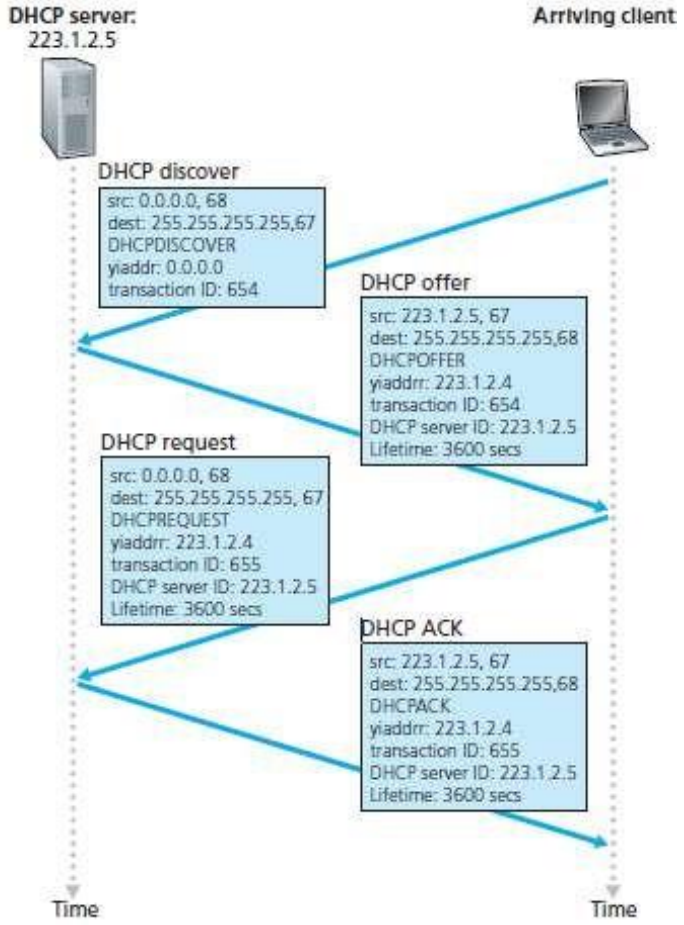


Figure 3.16: DHCP client-server interaction

- Four steps in DHCP protocol (Figure 3.16):

1) DHCP Server Discovery

➢ DHCP server contains a range of unassigned addresses to be assigned to hosts on-demand.

➢ To contact DHCP server, a client broadcasts a DHCPDISCOVER message with destination IP address 255.255.255.255.

2) DHCP Server Offer

➢ DHCP server broadcasts DHCPOFFER message containing
→ client's IP address
→ network mask and
→ IP address lease time (i.e. the amount of time for which the IP address will be valid).

3) DHCP Request

➢ The client sends a DHCPREQUEST message, requesting the offered address.

4) DHCP ACK

➢ The DHCP server acknowledges with a DHCPACK message containing the requested configuration.

"

NAT

• Network Address Translation (NAT) enables hosts to use Internet without the need to have globally unique addresses.
• NAT enables organization to have a large set of addresses internally and one address externally.
• The organization must have single connection to the Internet through a NAT-enabled router.
• NAT allows a single device (such as a router) to act as an agent b/w
    1) Internet (or "public network") and
    2) Local (or "private") network.
• This means only a single, unique IP address is required to represent an entire group of computers.
• Figure 3.17 shows the operation of a NAT-enabled router.

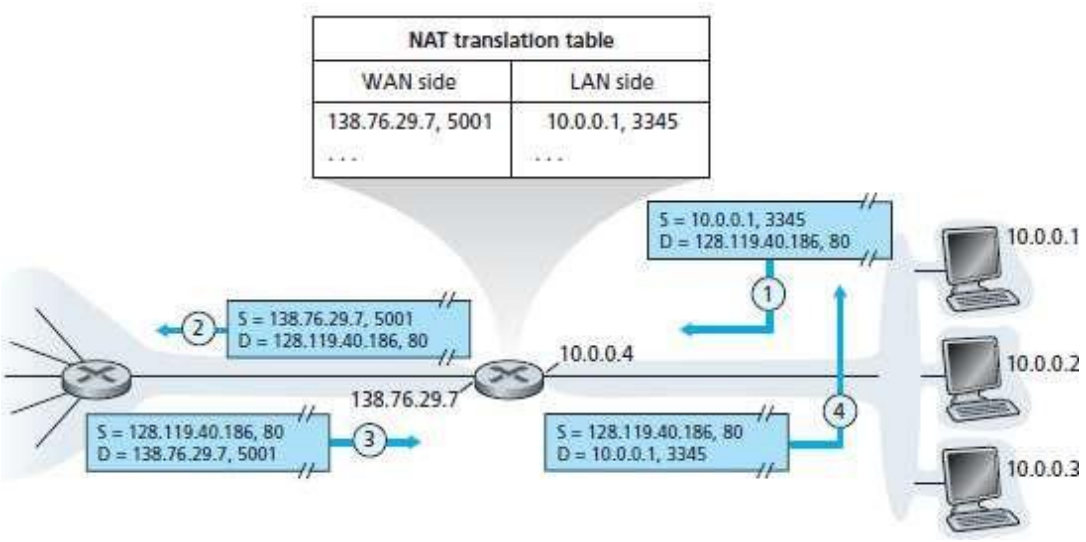


Figure 3.17: Network address translation

• The private addresses only have meaning to devices within a given network.
• The NAT-enabled router does not look like a router to the outside world.
• Instead, the NAT-enabled router behaves to the outside world as a single device with a single IP address.
• In Figure 3.17,
    1) All traffic leaving the home-router for the Internet has a source-address of 138.76.29.7.
    2) All traffic entering the home-router must have a destination-address of 138.76.29.7.
• The NAT-enabled router is hiding the details of the home-network from the outside world.
• At the NAT router, NAT translation-table includes
    1) Port numbers and
    2) IP addresses.
• IETF community is against the use of NAT. This is because of following reasons:
    1) They argue, port numbers are to be used for addressing processes, not for addressing hosts.
    2) They argue routers are supposed to process packets only up to layer 3.
    3) They argue the NAT protocol violates the end-to-end argument.
    4) They argue, we should use IPv6 to solve the shortage of IP addresses.
    5) NAT interferes with P2P applications. If Peer B is behind NAT, Peer B cannot act as a server.