



Data Communication and Computer Networks

Chapter Two

Data Link Layer



DATA LINK LAYER.

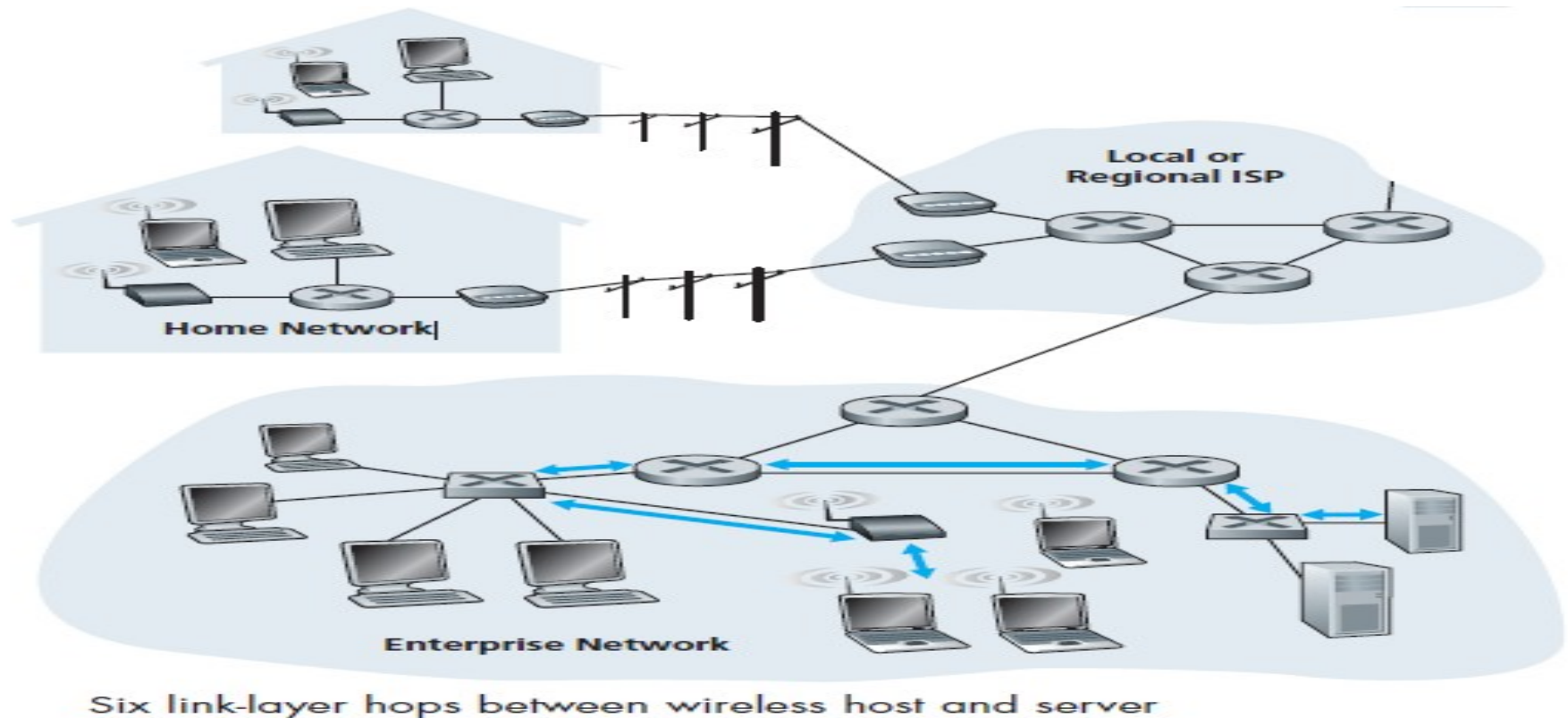
Fundamental Concept



The Data Link layer prepares network data for the physical network.

Contents

- Introduction
- Data link layer services (framing, error control, flow control)
- MAC address

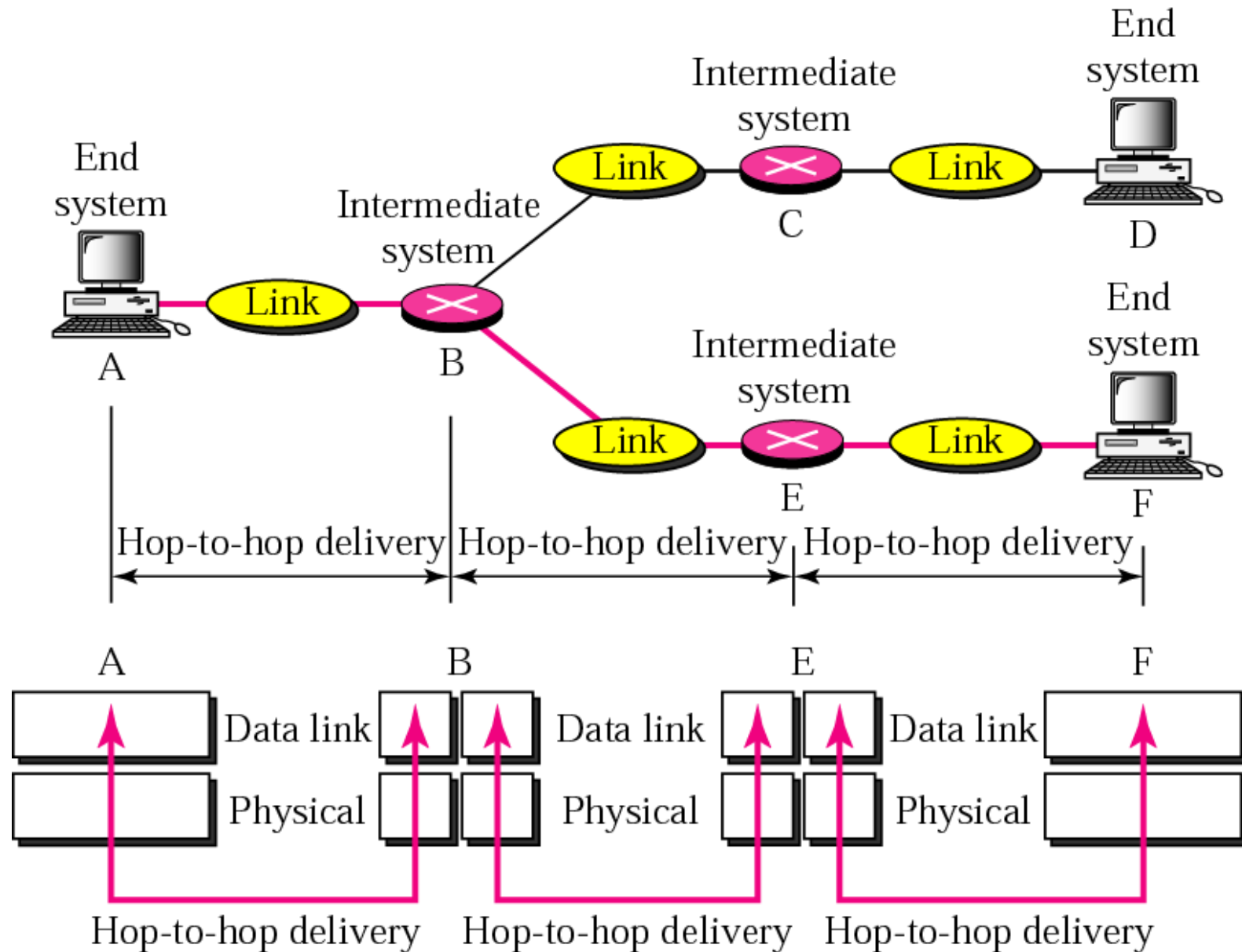


Introduction to Data link layer

- Data link layer is the second layer in OSI and Internet protocol stacks.
- The data link layer is responsible for moving frames from one hop (node) to the next.
- As the main principles of layered architecture the data link layer is responsible in order to provide services to the upper layer.
- The layer accepts data from the physical layer and performs the following activities on it.
 - Providing a well-defined service interface to the network layer.
 - Framing
 - Link access
 - Flow control
 - Error detection and correction

******Reading assignment about physical layer**

Hop-to-hop (node-to-node) delivery by data link layer



Data Link Layer Responsibilities

1. **Framing.** The data link layer divides the stream of bits received from the physical layer into manageable data units called frames.

❖ Separates a **message** from one source to a destination, by adding a sender address and a destination address.

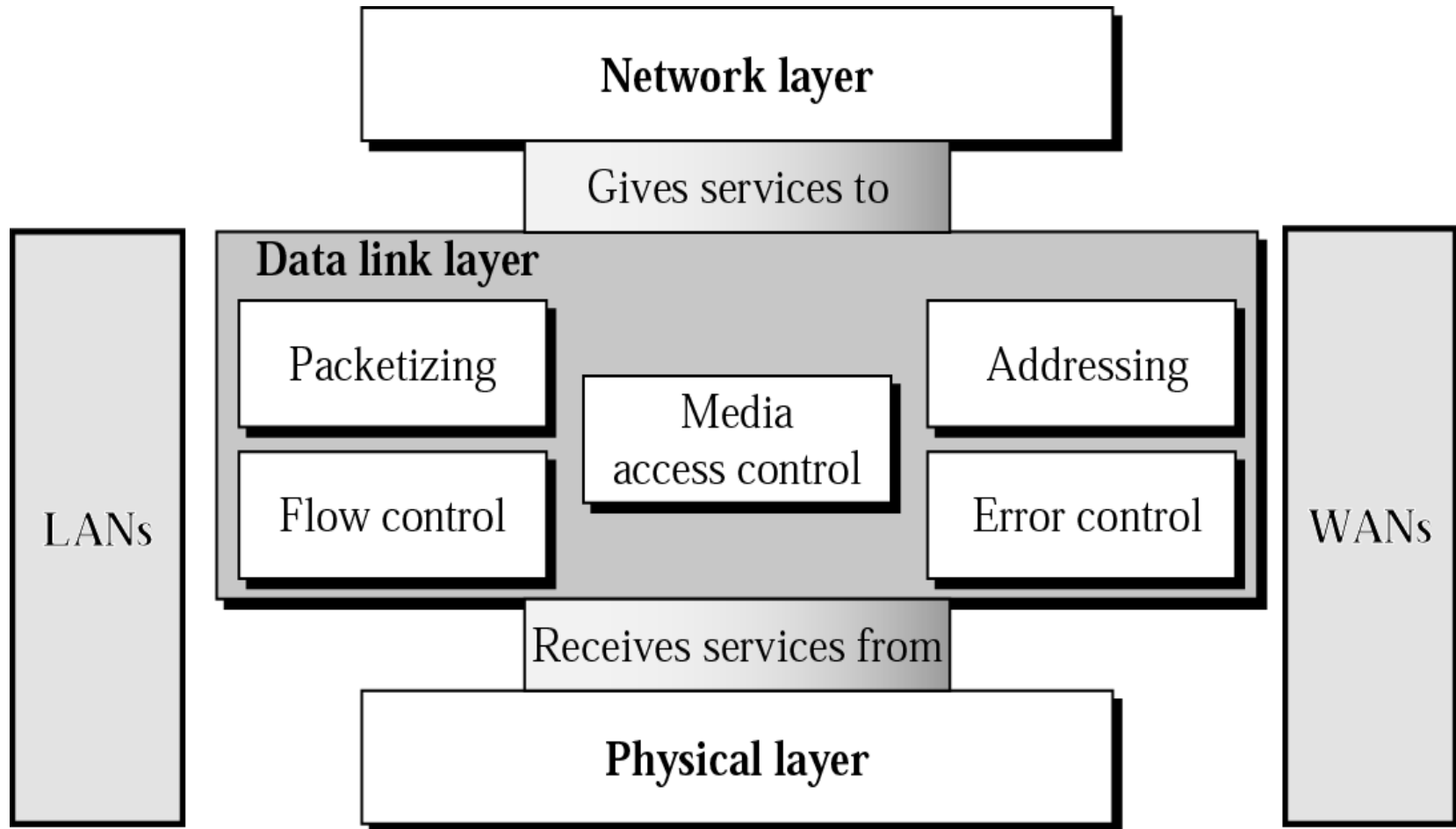
2. **Physical addressing.** If frames are to be distributed to different systems on the network, the data link layer adds a header to the frame to define the sender and/or receiver of the frame.

□ If the frame is intended for a system outside the

Contd.

3. **Flow control**. If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.
4. **Error control**. The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames.
 - ▢ It also uses a mechanism to recognize duplicate frames.
 - ▢ Error control is normally achieved through a **trailer** added to the end of the frame.
5. **Access control(link access)**. When two or more devices are connected to the same link, data link layer protocols are

Introduction cont....



Introduction cont.....

- ❖ In the sender end a packet from the upper layer is fragmented into a set of frames and in the destination end a collection of frames are merged together to create a packet for the upper layer.
- ❖ Each frame contains
 - ❑ A frame header,
 - ❑ A payload field for holding the packet, and
 - ❑ A frame trailer

Layer 2 frame structure

Start Frame (Flag)	Header		Data	Trailer	Stop Frame (Flag)
	Address	Type/ Length		FCS	

Where Is the Link Layer Implemented?

- The link layer is implemented in a network adapter (an interface for network for any device in the network), also known as a network interface card (NIC).
- At the heart of the network adapter is the link-layer controller, usually a single, special-purpose chip that implements many of the link-layer services (framing, link access, error detection, and so on).
- On the sending side, the controller takes a packet that has been created and stored in host memory by the higher layers of the protocol stack, encapsulates the datagram in

Where Is the Link Layer Implemented?

- On the receiving side, a controller receives the entire frame, and extracts the network-layer datagram.
- If the link layer performs error detection, then it is the sending controller that sets the error-detection bits in the frame header and it is the receiving controller that performs error detection.

Services Provided to the Network Layer

- The principal service is transferring data from the network layer on the source machine to the network layer on the destination machine .
- Three services that are commonly provided are:
 - Unacknowledged connectionless service
 - Acknowledged connectionless service
 - Acknowledged connection-oriented service
 - ❖ Connection established
 - ❖ Frames are transmitted
 - ❖ Connection are released

Fragmentation

- Every **packet** from **network layer** is fragmented into a number of frames data link layer.
- Why?
 - The buffer size of the receiver may be limited.
 - With smaller frames, errors are detected sooner, and a smaller amount of data needs to be retransmitted.
 - To prevent monopoly of a shared medium by one station (i.e. it is to implement sharing of expensive resources among different users)
- The data link layer compute a checksum for each frame, and include the checksum in the frame when it is transmitted.
- **Frames can be of fixed or variable size**

Framing

Framing cont'd

1. **Fixed-Size Framing**:- In **fixed-size framing**, there is no need for defining the boundaries of the frames; the size itself can be used as a **delimiter**.

- ❖ example the ATM wide-area network, which uses frames of fixed size called cells.

2. **Variable-Size Framing**:- It is prevalent in **local area networks** and we need a way to define the **end of the frame** and the **beginning of the next**.

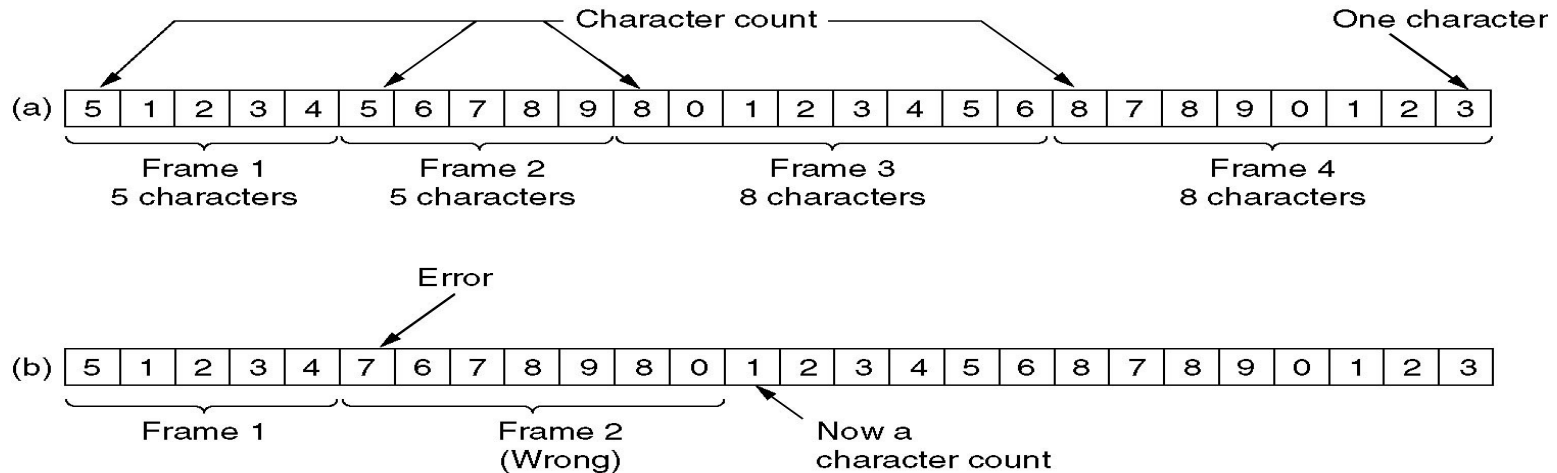
- ❖ Historically, two approaches were used for this purpose: a **character-oriented** approach and a **bit-oriented** approach.

- We will look at four methods:

1. Byte count.
2. Flag bytes with byte stuffing.

Byte count

- Uses a field in the header to specify the number of bytes in the frame.
- When the data link layer at the destination sees the byte count, it knows how many bytes follow and hence where the end of the frame is.



A byte stream. (a) Without errors. (b) With one error.

- Byte count algorithm is unable to locate the correct start of the next frame when error happen in the middle.

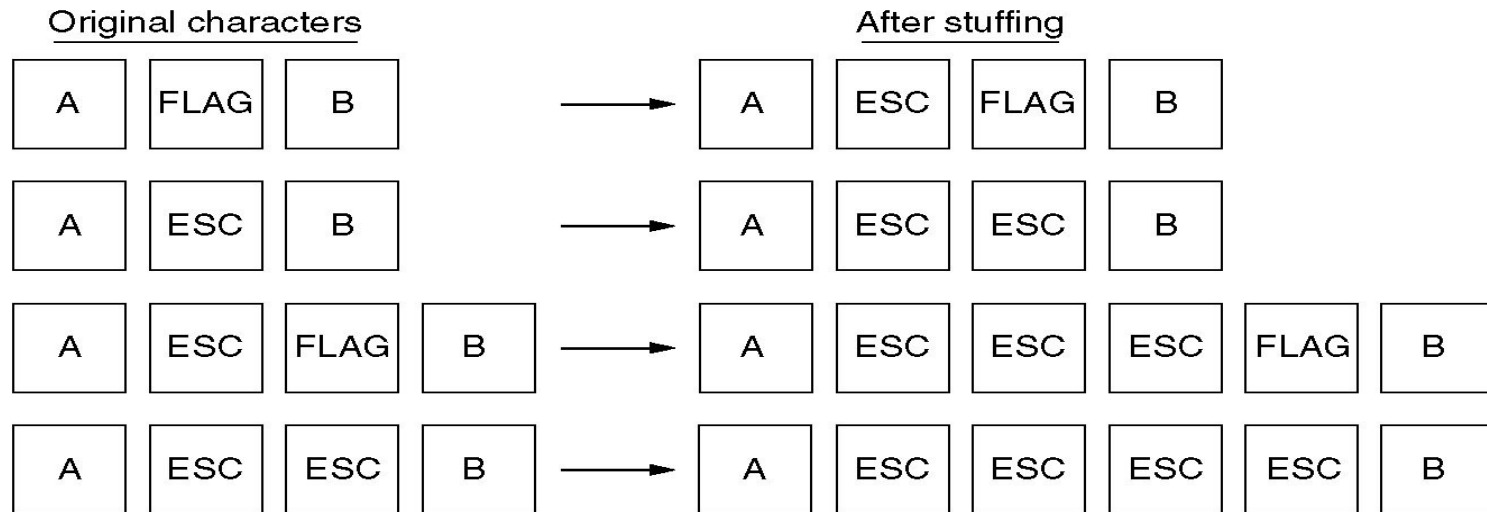
Flag bytes with byte stuffing

- Byte stuffing is the process of adding 1 extra byte whenever there is a flag or escape character in the text.
- One way to solve this problem is to have the sender's data link layer insert a special escape byte (ESC) just before each "accidental" flag byte in the data.
- Thus, a framing flag byte can be distinguished from one in the data by the absence or presence of an escape byte before it.
- In the receiving end removes the escape byte and flag bytes before giving the data to the network layer.
- This technique is called byte **stuffing**.
- **What happens if the text contains one or more normal escape characters followed by a flag?**

Flag bytes with byte stuffing cont....



(a)



(b)

(a) A frame delimited by flag bytes. (b) Four examples of byte sequences before and after byte stuffing.

Flag bits with bit stuffing

- 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 mistaken the pattern 0111110 for a flag.
- It was developed for the very popular data link protocol called **Highlevel Data Link Control (HDLC)**.
- Each frame begins and ends with a special bit pattern, **01111110** or **0x7E** in hexadecimal. This pattern is a flag byte.
- Whenever the sender's data link layer encounters five consecutive 1s in the data, it automatically stuffs a 0 bit into the outgoing bit stream.
- It also ensures a minimum density of transitions that help the physical layer maintain synchronization.
- When the receiver sees five consecutive incoming 1 bits, followed by a 0 bit, it automatically destuffs (i.e., deletes) the 0 bit.

Flag bits with bit stuffing cont....

(a) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

(b) 0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0



Stuffed bits

(c) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

Bit stuffing.

a) The original data.

b) The data as they appear on the line.

c) The data as they are stored in the receiver's memory after de stuffing.

❖ Note that even if we have a 0 after five 1s, we still stuff a 0. The 0 will be removed by the receiver.

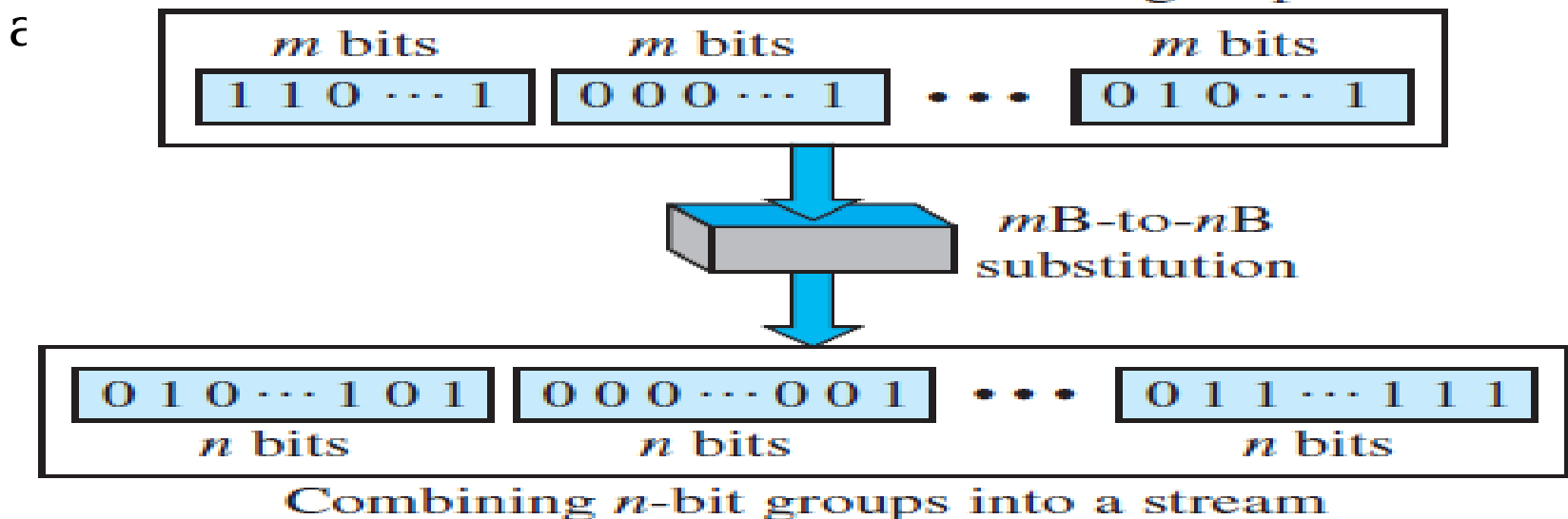
Physical layer coding violations

- The last method of framing is to use a shortcut from the physical layer.
- The encoding of bits as signals often includes redundancy to help the receiver. This redundancy means that some signals will not occur in regular data.

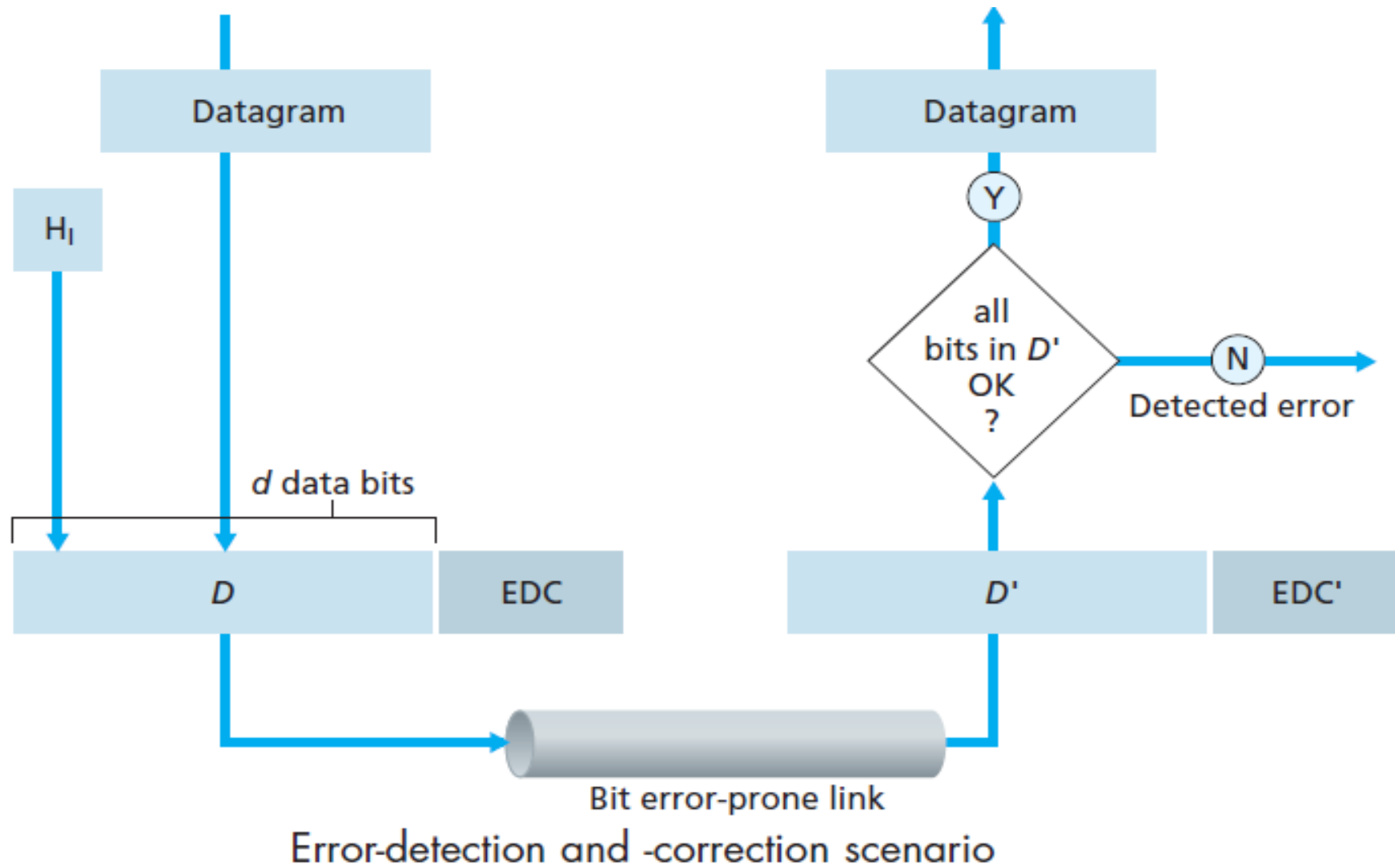
For example, in the 4B/5B line code 4 data bits are mapped to 5 signal bits to ensure sufficient bit transitions.

- We can use some reserved signals to indicate the start and end of frames. This technique uses a “coding violations”

Division of a stream into m -bit groups



Error Control



Error Control cont.....

- Having solved the problem of marking the start and end of each frame, we come to the next problem: how to make sure all frames are eventually delivered to the network layer at the destination and in the proper order.
- Typically, the protocol calls for the receiver to send back special control frames bearing positive or negative acknowledgements about the incoming frames.
- Sometimes because of malfunctioning hardware or a faulty communication channel feedback is not possible.
- This possibility is dealt with by introducing timers into the data link layer. When the sender transmits a frame, it generally also starts a timer.

Error control cont....

- Normally, the frame will be correctly received and the acknowledgement will get back before the timer runs out, in which case the timer will be canceled.
- The whole issue of managing the timers and sequence numbers so as to ensure that each frame is ultimately passed to the network layer at the destination exactly once.
- The common techniques for error control are some or all of the followings:
 - Error detection
 - Positive acknowledgment
 - Retransmission after timeout
 - Negative acknowledgment and retransmission

Flow Control

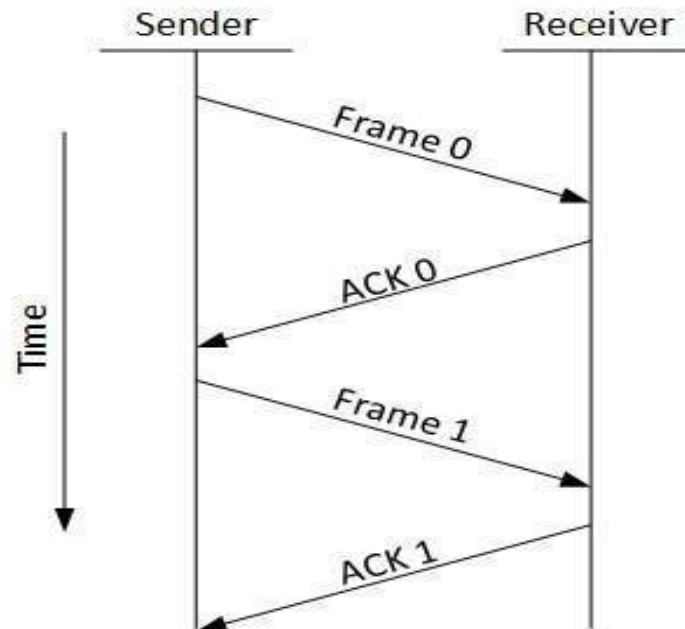
- Flow control **coordinates** the amount of data that can be sent before **receiving** an acknowledgment and is one of the most **important duties** of the data link layer.
- It is a **set of procedures** that tells the sender how much data it can transmit before it must wait for an acknowledgment from the receiver.
- The flow of data must not be allowed to **overwhelm** the receiver. Any receiving device has a **limited speed** at which it can **process incoming data** and a **limited amount of memory** in which to store incoming data.
- The receiving device must be able to **inform the sending device** before those limits are reached and to request that the transmitting device send fewer frames or **stop temporarily**. Incoming data must be checked and processed before they can be used.

Flow control **cont'd.....**

- The rate of such processing is often slower than the rate of transmission. For this reason, each receiving device has a block of memory, called a *buffer*, reserved for storing incoming data until they are processed. If the buffer begins to fill up, the receiver must be able to tell the sender to **halt transmission** until it is once again able to receive.
- Different methods
 - Stop-and-Wait Flow Control
 - Sliding-Window Flow Control

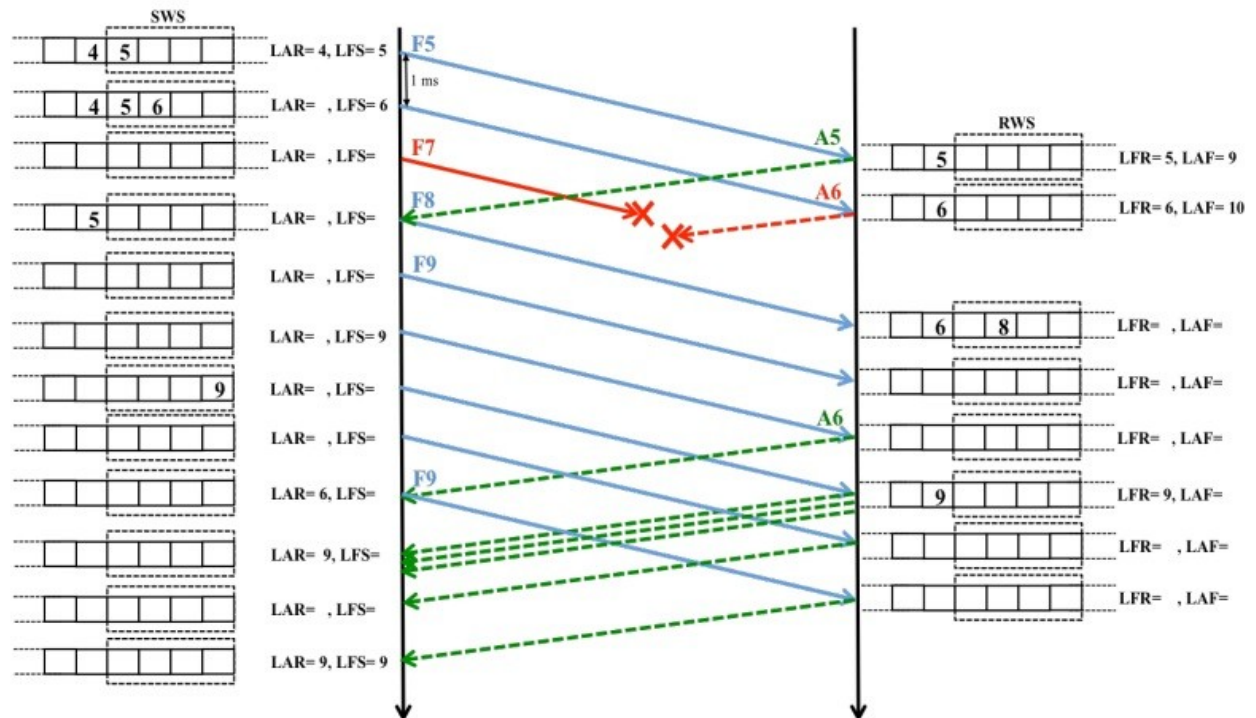
Stop-and-Wait Flow Control

- It is the simplest form of flow control in data link layer. It works in the following way:
 - First the source transmits frame.
 - The destination receive the message and replies with acknowledgement
 - Source waits for ACK before sending next frame
 - Only one frame can be in transit at a time
 - Destination can stop flow by not sending ACK
 - Works well for a few fram



Sliding-Window Flow Control

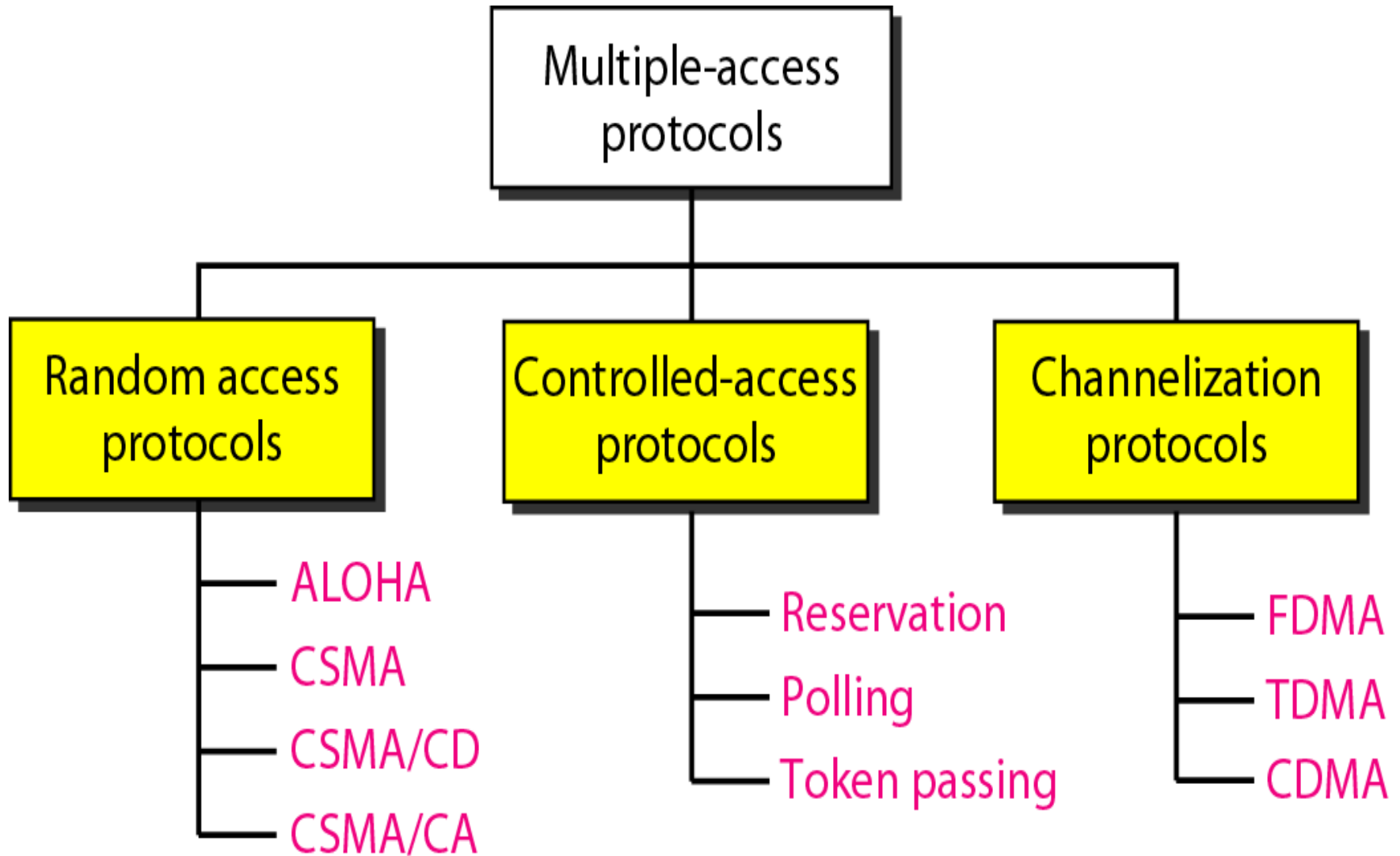
- In this technique of flow control the sender is allowed to send to the destination. The algorithm works in the following way:
 - Receiver has buffer W long
 - Transmitter can send up to W frames without ACK
 - Each frame is numbered
 - ACK includes
 - Sequence num
 - Frames are
 - Efficient
 - Many frame



Media Access control

- The data link layer can further be divided into two layers: the upper sub-layer that is responsible for flow and error control is called the logical link control (LLC) layer; the lower sub-layer that is mostly responsible for multiple access resolution is called the media access control (MAC) layer
- When nodes or stations are connected and use a common link, called a multipoint or broadcast link, we need a multiple-access protocol to coordinate access to the link. The problem of controlling the access to the medium is similar to the rules of speaking in an assembly. The procedures guarantee that the right to

Contd.



Random Access

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

Contd.

- In a random access method, each station has the right to the medium without being controlled by any other station.
- However, if **more than one station** tries to send, there is an access conflict-**collision**-and the frames will be either **destroyed or modified**.
- To avoid access conflict or to **resolve it** when it happens, each station follows a procedure that answers the following questions:
 - ▮ **When can the station access the medium?**
 - ▮ **What can the station do if the medium is busy?**
 - ▮ **What can the station do if the medium is idle?**

Contd.

- ▮ The random access methods have evolved from a very interesting protocol known as **ALOHA**, which used a very simple procedure called **multiple access (MA)**.
- ▮ The method was improved with the addition of a procedure that forces the station to sense the medium before transmitting.
 - ❖ This was called **carrier sense multiple access**.
 - ❖ This method later evolved into two parallel methods: **carrier sense multiple access with collision detection (CSMA/CD)** and **carrier sense multiple access with collision avoidance (CSMA/CA)**.
 - ❖ *CSMA/CD* tells the station what to do when a collision is detected.
 - ❖ *CSMA/CA* tries to avoid the collision.
- ▮ The simplest CSMA scheme is for a station to **sense the medium,**

Persistent

- **Persistent:-***wait if busy and transmit only when the media becomes idle again (not transmission after a triggered timer expire)*
- When a station has the data to send, it first listens the channel to **check if anyone** else is transmitting data or not.

- If it senses the channel **idle**, station starts transmitting the data and if it is busy waits channel until it become idle by **Non-Persistent** continuously sensing.

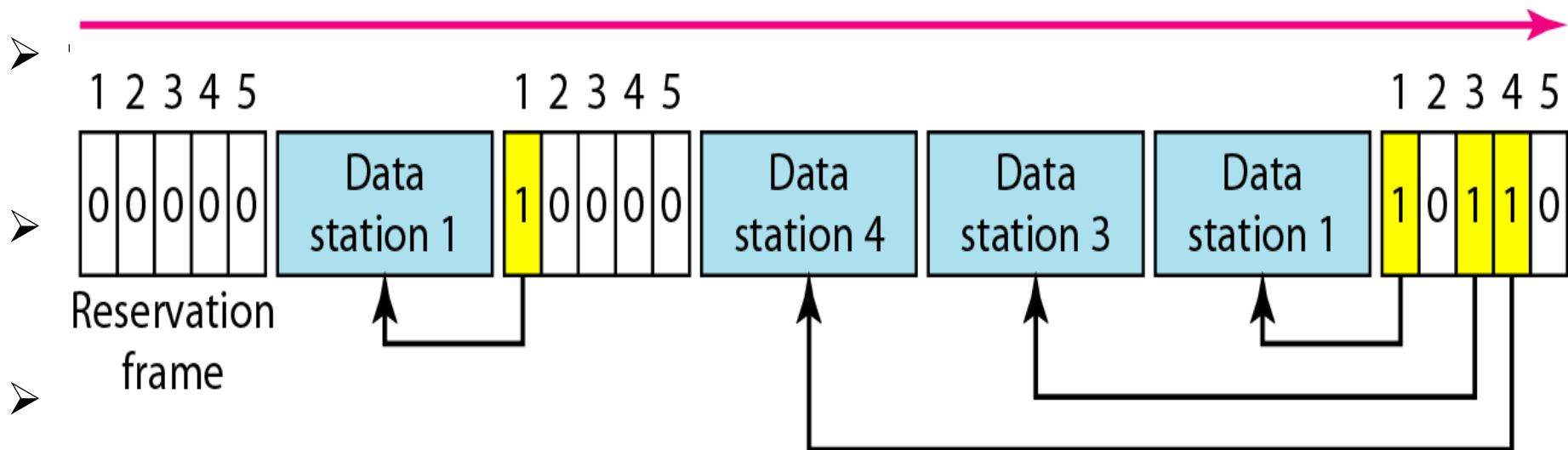
- In this protocol, before sending the data, the station senses the channel and if the channel is idle it **starts transmitting** the

Controlled access

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

Reservation

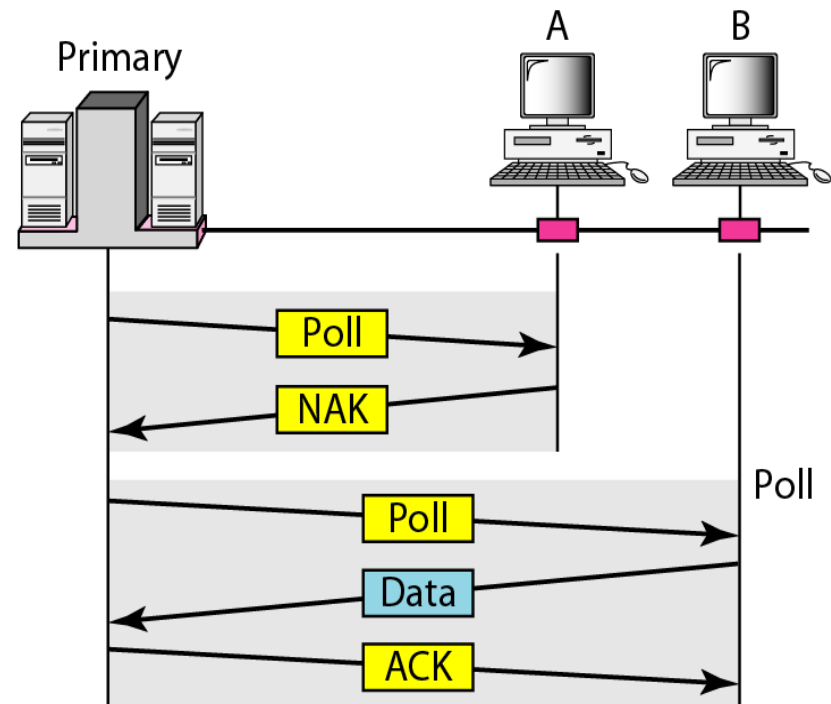
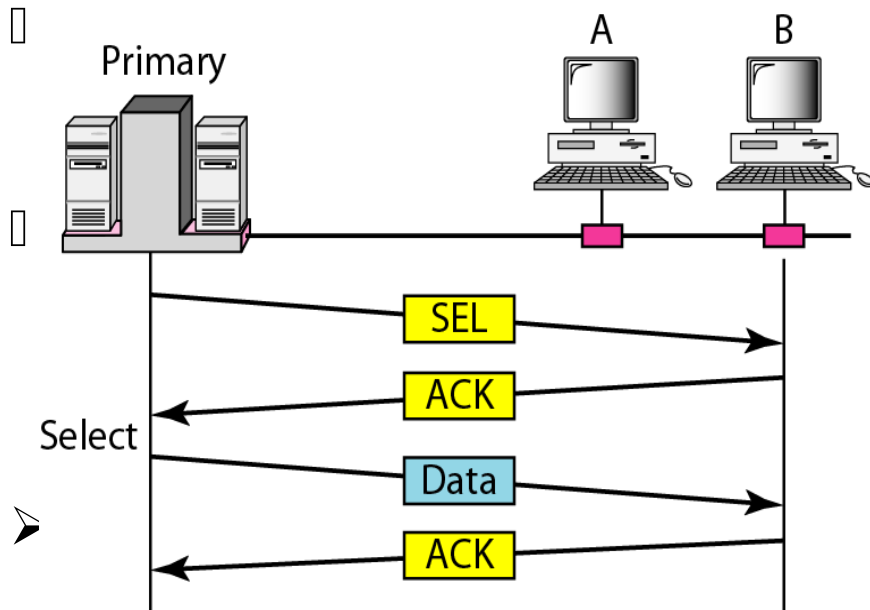
- In the reservation method, a station needs to make a **reservation** before sending data.



only stations 1, 3, and 4 have made reservations. In the second

Polling

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



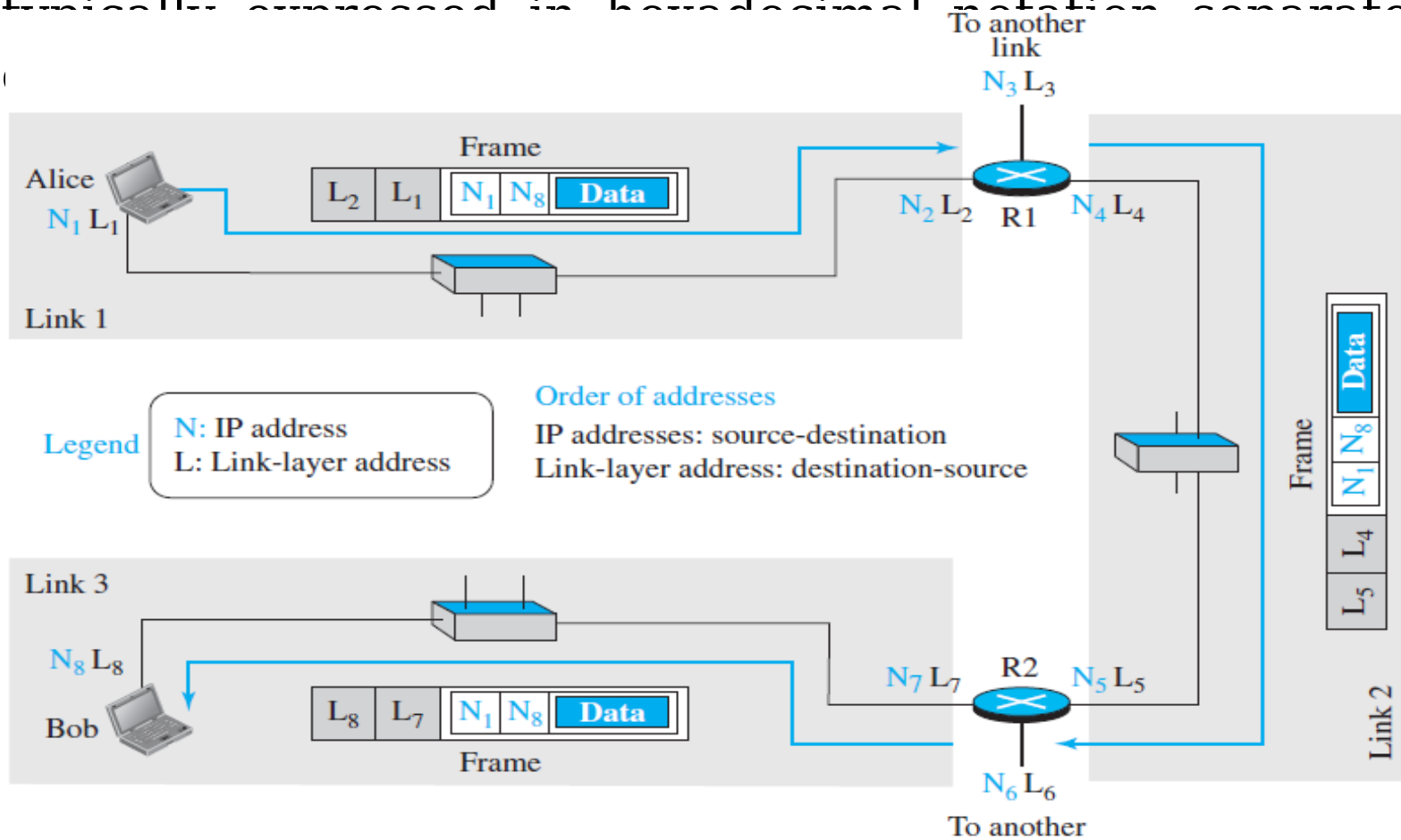
function.

Token Passing

- In the token-passing method, the stations in a network are organized in a logical ring(for each station *predecessor* and a *successor*).
- The predecessor is the station which is logically before the station in the ring; the successor is the station which is after the station in the ring.
- The right to this access has been passed from the predecessor to the current(one that is accessing the channel now) station.
- The right will be passed to the successor when the current station has no more data to send.
- But how is the right to access the channel passed from one station to another?
- The possession of the token gives the station the right to access the channel and send its data.
- When a station has some data to send, it waits until it receives the token from its predecessor.
- When the station has no more data to send, it releases the token,

MAC (Physical) Addresses

- It is an address used to establish a point-to-point connection between nodes in the network.
- For most LANs, the MAC address is 6 bytes long, giving 2^{48} possible MAC addresses. These 6-byte addresses are typically expressed in hexadecimal notation generated by



MAC Address cont...

- When the packet passes from network to 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.
- Some link layer protocols define three types of addresses:
 - unicast

- Unicast

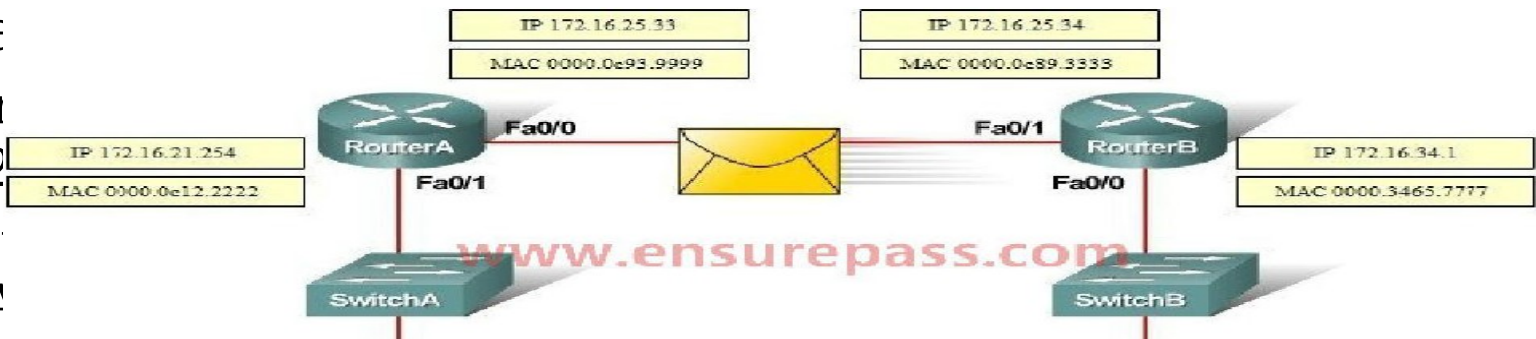
12

- Multicast

LA

- Broadcast

all

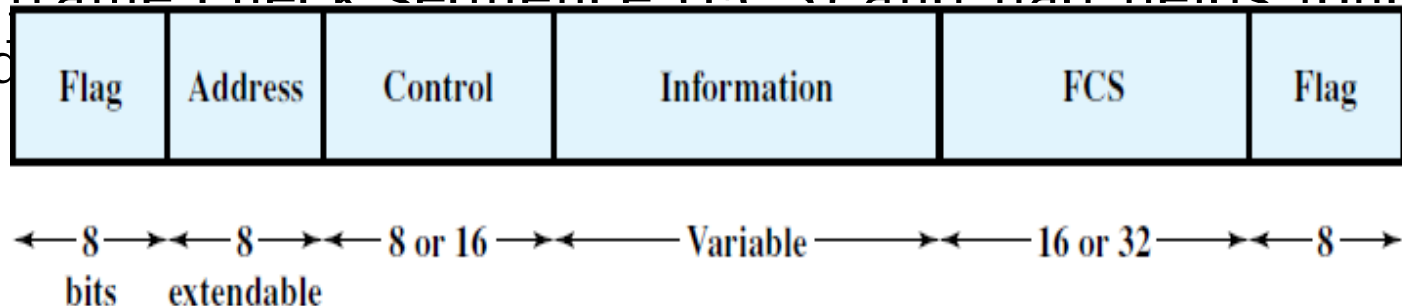


```

Interface: 192.168.92.1 --- 0x10
Internet Address      Physical Address      Type
192.168.92.254        00-50-56-f1-b3-56     dynamic
192.168.92.255        ff-ff-ff-ff-ff-ff     static
224.0.0.22            01-00-5e-00-00-16     static
224.0.0.252           01-00-5e-00-00-fc     static
239.255.255.250       01-00-5e-7f-ff-fa     static
255.255.255.255       ff-ff-ff-ff-ff-ff     static
    
```

Frame Structure

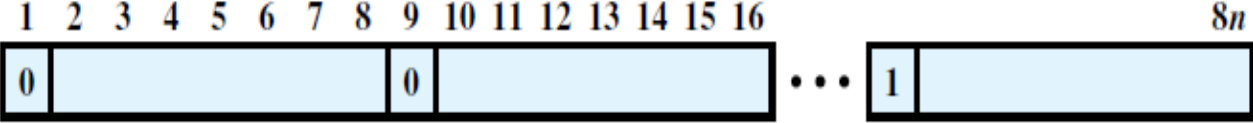
- The most important data link control protocol is High-level Data Link Control (HDLC). HDLC uses synchronous data transmission.
- All transmissions are in the form of frames, and a single frame format suffices for all types of data and control exchanges.
- The flag, address, and control fields that precede the information field are known as a **header**.
- The frame check sequence (FCS) and flag fields following the c



- **Flag Fields:** delimit the frame at both ends with the unique pattern 01111110.
- On both sides of the user-network interface, receivers are continuously hunting for the flag sequence to synchronize

Frame Structure cont....

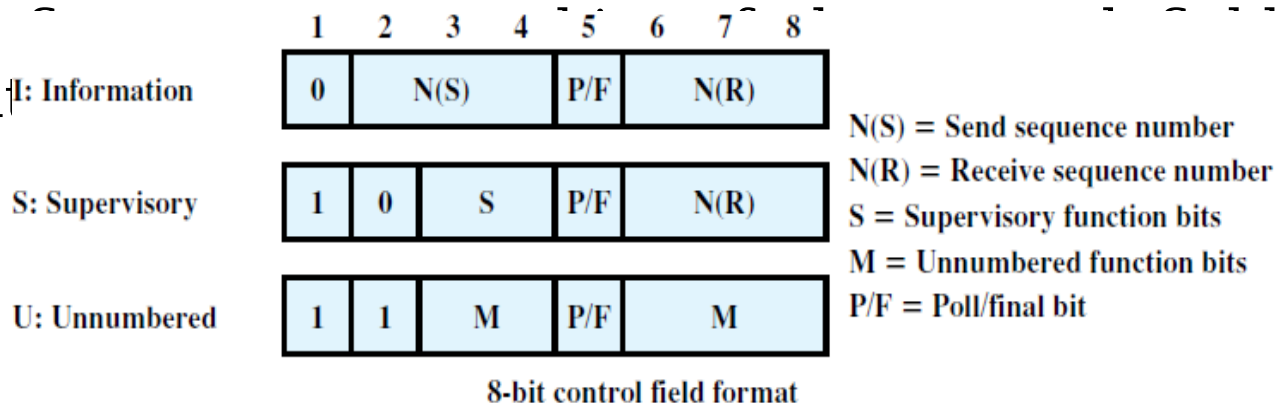
- There is no assurance that the pattern 01111110 will not appear somewhere inside the frame, thus destroying synchronization.
- To avoid this problem, a procedure known as bit stuffing is used. For all bits between the starting and ending flags, the transmitter inserts an extra 0 bit after each occurrence of five 1s in the frame.
- **Address Field:** identifies the secondary station that transmitted or is to receive the frame.
- The address field is usually **8 bits long** but, by prior agreement, an extended format may be used in which the actual address length is a multiple of 7 bits.
- The leftmost bit of each octet is **1 or 0 according** as it is or is not the last octet of the address field. The remaining 7 bits of each octet form part of the address.

- The all-s  reted as the formats.
- Extended address field

Frame Structure cont.....

- **Control Field:** HDLC defines three types of frames, each with a different **control field format**.
- **Information frames (I-frames)** carry the data to be transmitted for the user. Additionally, flow and error control data, using the **automatic repeat request (ARQ)** mechanism, are piggybacked on an information frame.
- **Supervisory frames (S-frames)** provide the ARQ (Automatic Repeat Request) mechanism when piggybacking is not used.
- **Unnumbered frames (U-frames)** provide supplemental link control functions.

➤ The **control field** serves to identify the type of frame and the sequence number of the next frame to be received.



Frame Structure cont.....

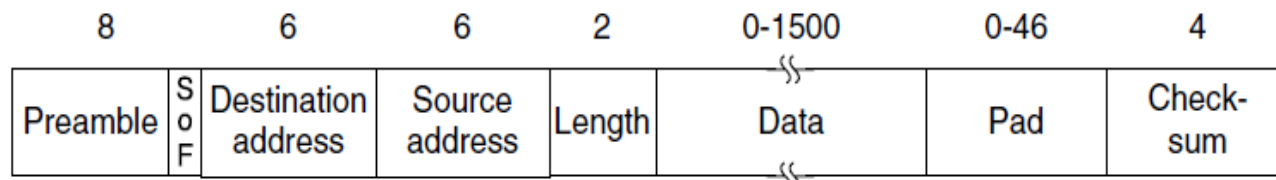
- All of the control field formats contain the poll/final (P/F) bit. Its use depends on context.
- Typically, in command frames, it is referred to as the P bit and is set to 1 to solicit (poll) a response frame from the peer HDLC entity.
- In response frames, it is referred to as the F bit and is set to 1 to indicate the response frame transmitted as a result of a soliciting command.
- **Frame Check Sequence Field (FCS):** is an error detecting code calculated from the remaining bits of the frame, exclusive of flags.
- The normal code is the 16-bit CRC.
- An optional 32-bit FCS, using CRC-32, may be employed if the frame length or the line reliability dictates this choice.

Ethernet

- Ethernet is by far the most prevalent wired LAN technology, and it is likely to remain so for the foreseeable future.
- There are many reasons for Ethernet's success.
 - First, Ethernet was the first widely deployed high-speed LAN.
 - Second, other technologies (i.e. token ring, fiber distributed data interface (FDDI), and ATM) were more complex and expensive than Ethernet.
 - Third, the most compelling reason to switch to another LAN technology (such as FDDI or ATM) was usually the higher data rate of the new technology.

Ethernet Frame Structure

- Preamble: each containing the bit pattern 10101010 (with the exception of the last byte, in which the last 2 bits are set to 11).
- This last byte is called the ***Start of Frame*** delimiter for Ethernet. The last two 1 bits tell the receiver that the rest of the frame is about to start.
- The next two parts are destination and source addresses expressed in 48 bits.
- Next comes the *Type* or *Length* field, depending on whether the frame is Ethernet or IEEE 802.3. Ethernet uses a *Type* field to tell the receiver what to do with the frame.
- The minimum valid frame size in Ethernet is ≥ 64 Bytes, excluding the preamble field in the frame. Sometimes this size may not be met by smaller data filed. Therefore, the Pad field is used to fill out the frame to the minimum size.



Ethernet Technologies

- Ethernet comes in *many* different flavors, with confusing acronyms such as 10BASE-T, 10BASE-2, 100BASE-T, 1000BASE-FX, and 10GBASE-T.
- The first part of the acronym refers to the speed in Megabit or Gigabit per second of the standard: 10, 100, 1000, or 10G.
- BASE refers to baseband Ethernet, meaning that the physical media only carries Ethernet traffic; almost all of the 802.3 standards are for baseband Ethernet.
- The final part of the acronym refers to the physical media itself; Ethernet is both a link-layer *and* a physical-layer specification.
 - “T” refers to twisted-pair copper wires
 - “2” refers two types of coaxial cable

“FX” refers to fiber optic cables

Switch

- The role of the switch is to receive **incoming link-layer** frames and forward them onto outgoing links.
- We'll see that the switch itself is **transparent** to the hosts and routers in the subnet (i.e. the host and router are unaware that a switch will be receiving the frame and forwarding it).
- **Filtering** is the switch function that determines whether a frame should be forwarded to some interface or should just be dropped.
- **Forwarding** is the switch function that determines the interfaces to which a frame should be directed.
- These tasks are done with a **switch table**. The switch table contains entries for some, but not necessarily all, of the hosts and routers on a LAN. An entry in the switch table contains (1) a MAC address, (2) the switch interface that leads toward that MAC address and (3) the time at which the entry was placed

Address	Interface	Time
62-FE-F7-11-89-A3	1	9:32
7C-BA-B2-B4-91-10	3	9:36
....		

Properties of Switches

- **Elimination of collisions**
- **Heterogeneous links**
- **Management:** using different commands found in the switch we have easily detect problems that are happen in data transmission. Some of the commands are:
 - Show interface: it is used to see the detail information about the switch interfaces.
 - Show interface trunk: it is used to see whether the switch interface is a trunk port or not and show allowed VLAN's on the given trunk port (when the port is a trunk).
 - Show vlan: used to see VLAN detail information in the switch.
 - Interface range fa0/1-fa0/20: used to perform different management activities in the switch interfaces defined in the range.
 - Line vty 0 4: used to define remote login access to the users using a telnet.

NEXT CHAPTER 3

Network Layer

Quiz 1

1. Assume that the sender and receiver agreed to use the check sum error detection mechanisms and the data units are given as following.

I.10011001

II.11100010

II .00100100

A. Define what is check sum.

B. Check if the receiver receive error free ,message or not

C. What happens if the text contains one or more normal escape characters followed by a flag?