

# LECTURE #29

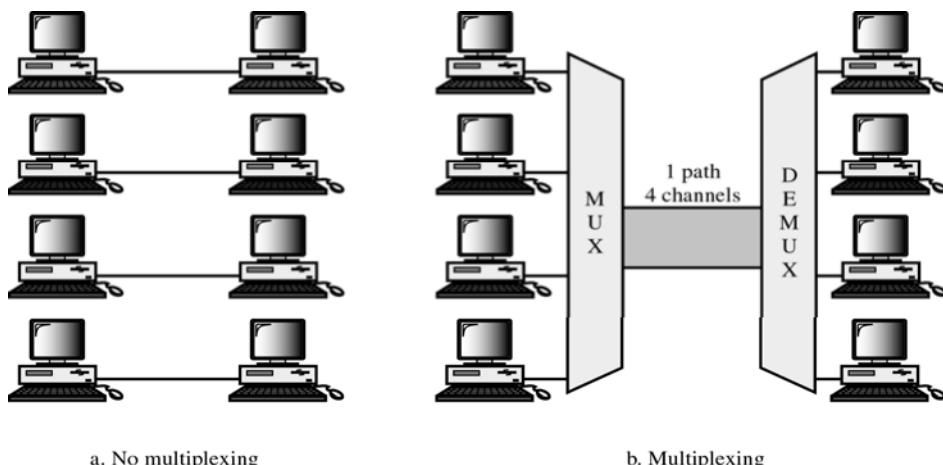
## **Introduction**

- Whenever the TX capacity of a medium linking 2 devices is greater than the TX needs of the devices, the link can be shared
- Example: Large Water pipe can carry water to several separate houses at once
- Multiplexing is the set of techniques that allows simultaneous TX of multiple signals across a single data link
- As data communication usage increases, traffic also increases
- We can add a new line each time a new channel is needed
- Or we can install higher capacity links and use each to carry multiple signals
- All current TX media i.e. Coax, Optical fiber have high available BWs
- Each of these has carrying capacity far in excess of that needed for one signal
- If TX capacity of a link is greater than the TX needs of devices attached to it, the excess capacity is wasted

## **Multiplexing**

**Set of techniques that allows the simultaneous transmission of multiple signals across a single data link”**

In the multiplexed system, ‘n’ devices share the capacity of one link



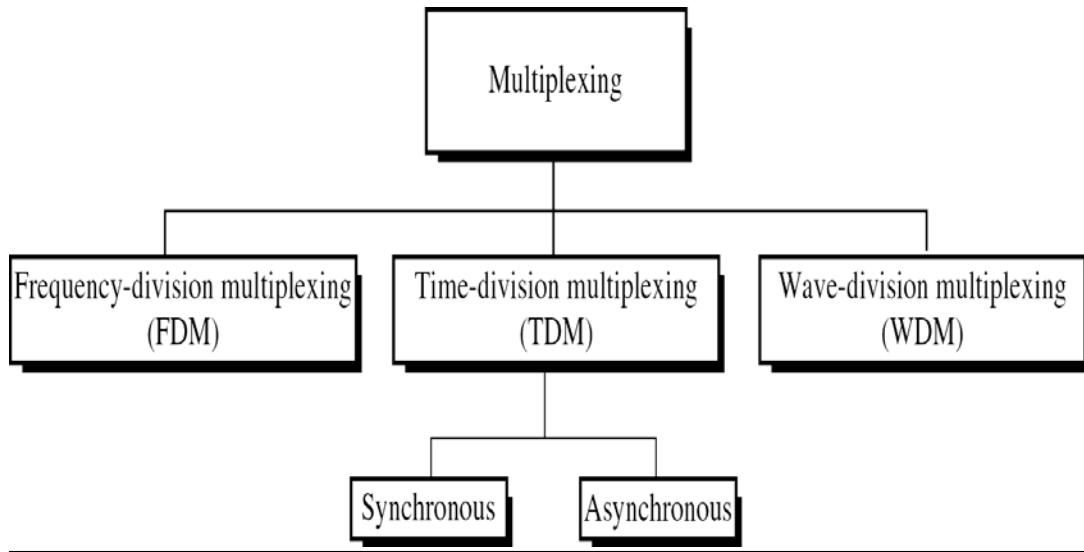
- Fig. shows two possible ways of linking 4 pairs of device
- In fig. (a), each pair has its own link. If full capacity of each link is not utilized, it will be wasted
- In fig. (b), TX b/w pairs are multiplexed . The same 4 pairs share the capacity of single link
- Fig. (b) shows the basic format of a Multiplexed system
- The 4 devices on left direct their TX streams to a **MUX**, which combines them into a single stream
- At the receiving end, that stream is fed into a **DEMUX**, which separates the stream back into its component transmissions and directs them to their intended devices

•**Path:** Physical Link

•**Channel:** A portion of the path that carries TX b/w a given pair of devices

•One path can have many channels

- Categories of Multiplexing

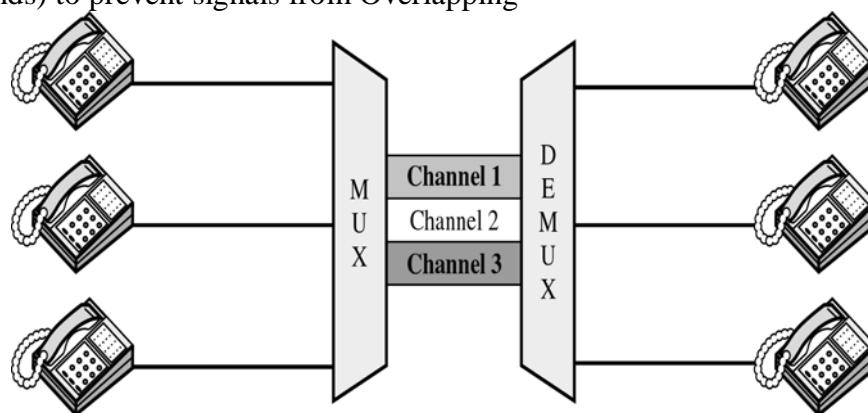


### Y FDM

- o An analog technique that can be applied when BW of the link is greater than the combined BW of the signals to be TX
- o Signals generated by each sending device modulate difference carrier frequencies
- o These modulated signals are then combined into a single Composite signal that can be transported by the link
- o Carrier frequencies are separated by enough BW to accommodate the modulated signal
- o These BW ranges are the channels through which the various signals travel

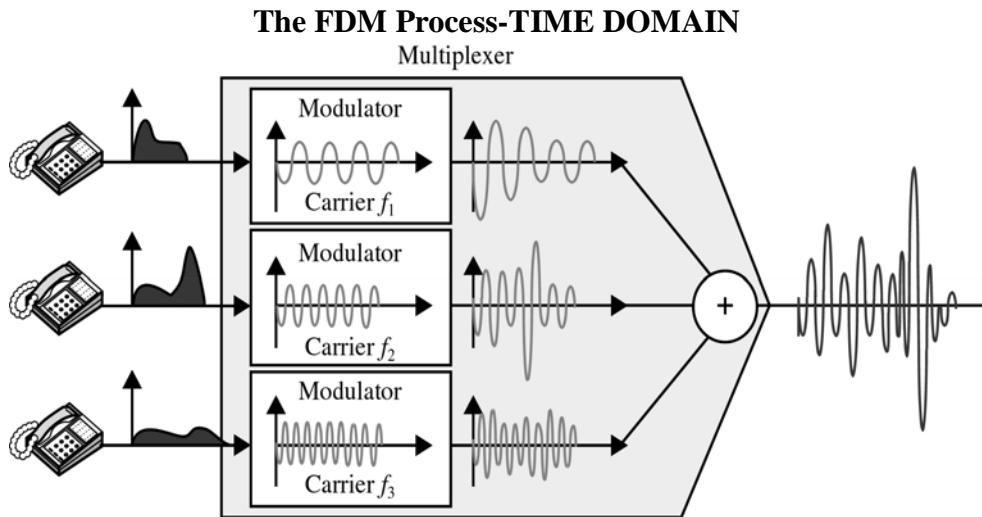
#### FDM (Guard Bands)

- GUARD BANDS:** Channels must be separated by strips of unused BW (guard bands) to prevent signals from Overlapping



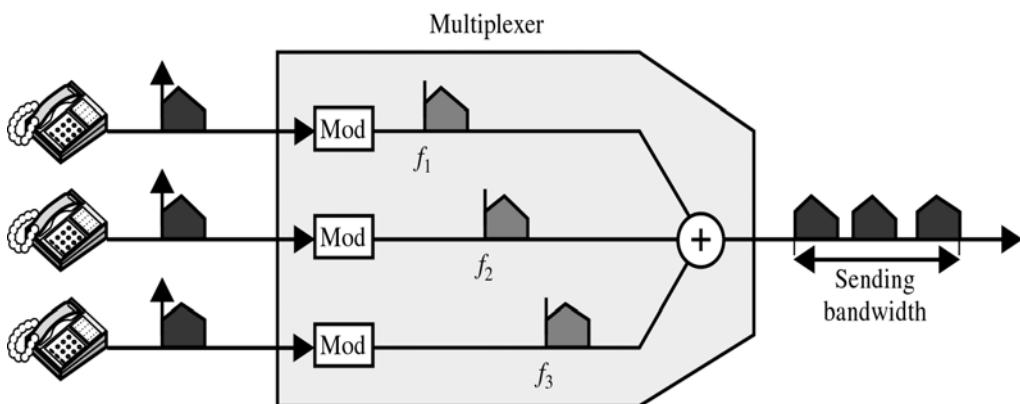
- o In fig. the TX path is divided into 3 parts, each representing a channel to carry one TX
- o As an analogy, imagine a point where 3 narrow streets merge to form a 3-lane highway

- Each of these streets correspond to a lane of the highway
- Each car merging on to the highway from one of these streets still has its own lane and can travel w/o interfering with cars from other lanes



- Figure shows a Time domain fdm
- FDM is an analog process and we show it here in using Telephones as I/p & o/p devices
- Each telephone generates a signal of similar frequency range
- Inside the MUX, these similar signals are modulated on to different carrier frequencies
- The resulting modulated signals are then combined into a single composite signal that is sent over a media link that has enough BW to accommodate it

### The FDM Process-Freq domain

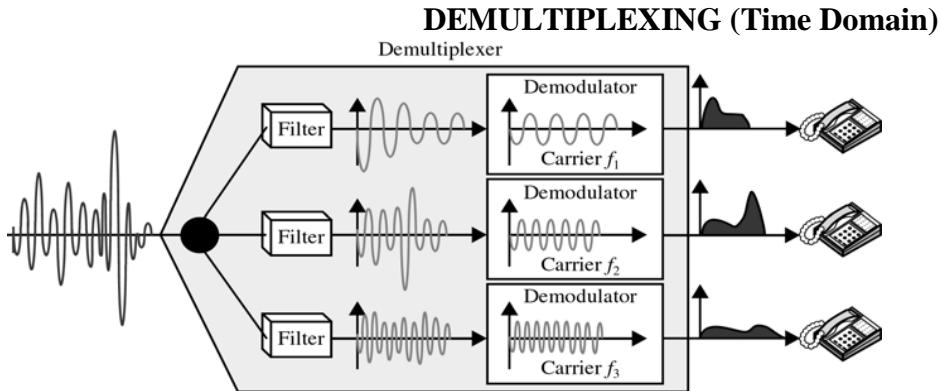


- Fig is freq domain representation of FDM process
- In FDM, signals are modulated onto separate carrier frequencies ( $f_1, f_2, f_3$ ) using either FM or AM
- Modulating one signal into the other results in a BW of at least twice the original

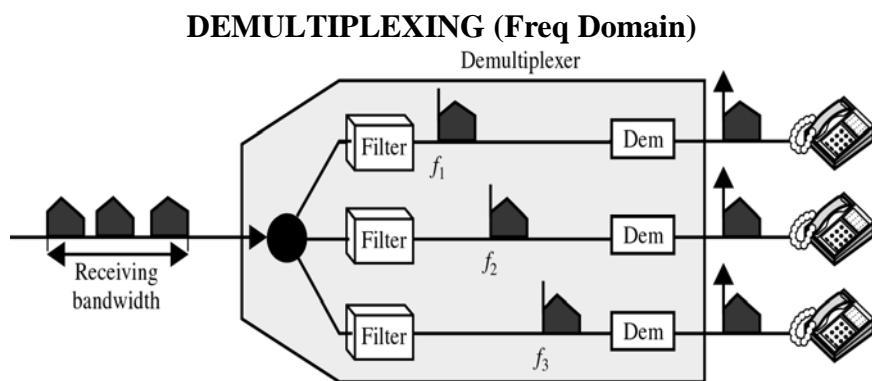
- In fig, the BW of resulting composite signal is more than 3 times the BW of each input signal
- Plus extra BW to allow for necessary GUARD BANDS

### - DEMULTIPLEXING

- DEMUX uses a series of filters to decompose multiplexed signal into its constituent signals
- Individual signals are then passed to a demodulator that separates them to the carriers and passes them to the waiting receivers



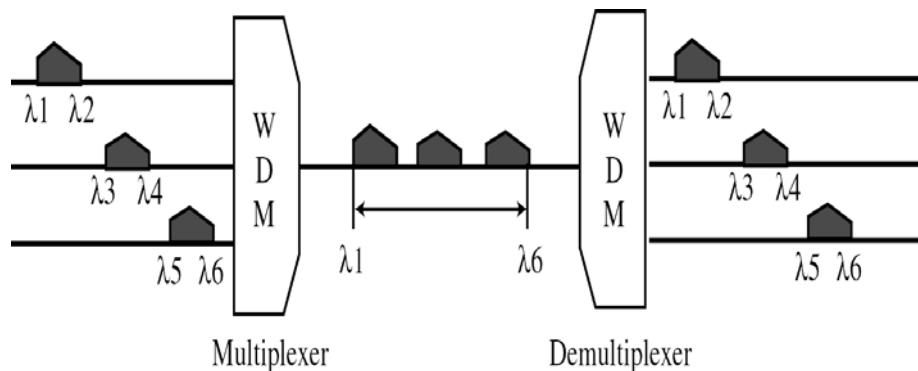
This figure is the time domain representation of the FDM MUX again using 3 telephones as the communication devices



This figure is the time domain representation of the FDM MUX again using 3 telephones as the communication devices

### Wave Division Multiplexing (WDM)

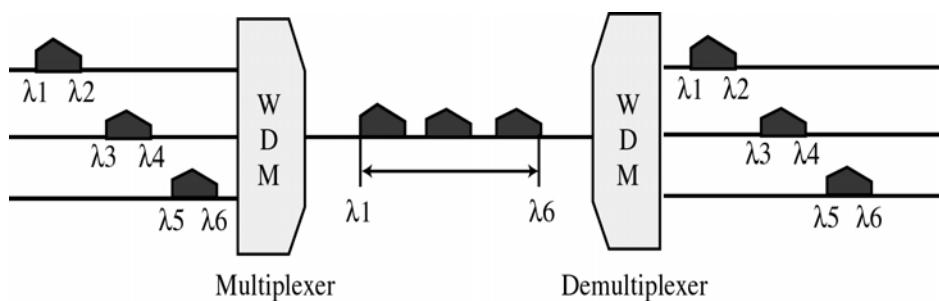
- It is conceptually the same as FDM except that multiplexing and demultiplexing involves light signals TX through fiber optic channels
- Idea is the same: We are combining different signals of the different frequencies
- However in this case frequencies are very high



- WDM MUX and DEMUX
- Very narrow bands of light from different sources are combined to make a wider band of light
- At the receiver are separated by DEMUX

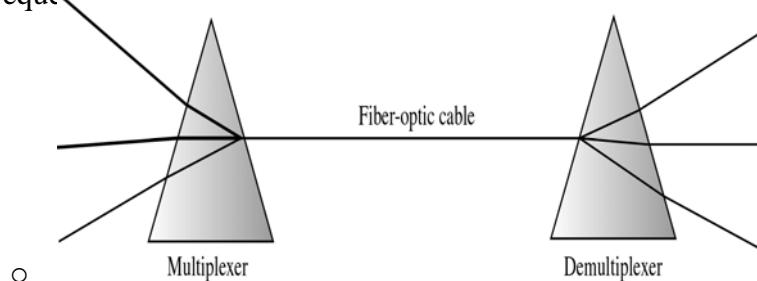
#### Y Mechanism of WDM

- Although the technology is very complex, the idea is very simple
- We want to combine multiple sources into one single light at the the MUX and do the reverse at the DEMUX



#### PRISM

- Combining and Splitting of light sources is easily handled by a PRISM
- From Physics, a prism can deflect the light depending upon the angle of incidence and the frequency

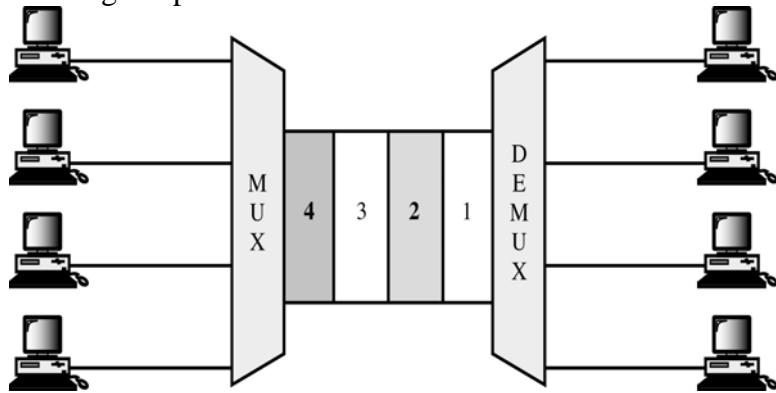


- Using this technique, a MUX can be made to combine several input beams of light each containing a narrow band of frequencies into one o/p beam of a wider band of frequencies
- The DEMUX can also be made to reverse the process

#### TDM

- TDM is a digital process that can be applied when the data rate capacity of the TX medium is greater than the data rate required by the sending and receiving devices

- In such case, multiple transmissions can occupy a single link by subdividing them and Interleaving the portions



In fig, same link is used as in FDM. However, here the link is shown sectioned by time rather than frequency In TDM fig, portions of signals 1, 2, 3 and 4 occupy the link sequentially

### **Y Implementation of TDM**

TDM can be implemented in two ways:

- Synchronous TDM
- Asynchronous TDM

### **Y Synchronous TDM**

- The term synchronous has a different from that used in other areas of telecommunication
- Here synchronous means that MUX allocates exactly the same time slot to each device at all device whether or not the device has any thing to transmit

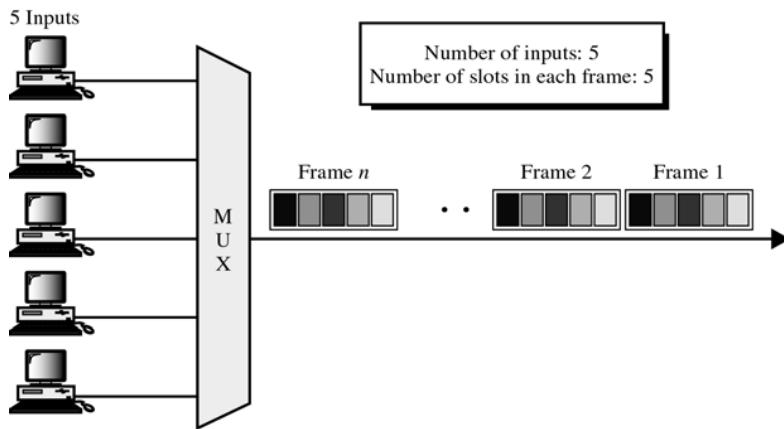
### **Y Synchronous TDM Example**

- Time slot A for example is assigned to device A alone and cannot be used by any other device
- Each time its allocated time slot comes up a device has the opportunity to send a portion of its data
- If a device is unable to transmit or does not have data to send time slot remains empty

### **Y Synchronous TDM Frames**

- Time slots are grouped into Frames
- A frame consists of one complete cycle of Time slots including one or more slots dedicated to each sending device
- In a system with 'n' I/p lines, each frame has atleast 'n' slots with each slot allocated to carrying data from a specific I/p line
- If all the I/p devices sharing a link are transmitting at the same data rate, each device has 1 timeslot per frame
- However it is possible to accommodate varying data rates
- A TX with two slots per frame will arrive twice as quickly as one with 1 slot per frame

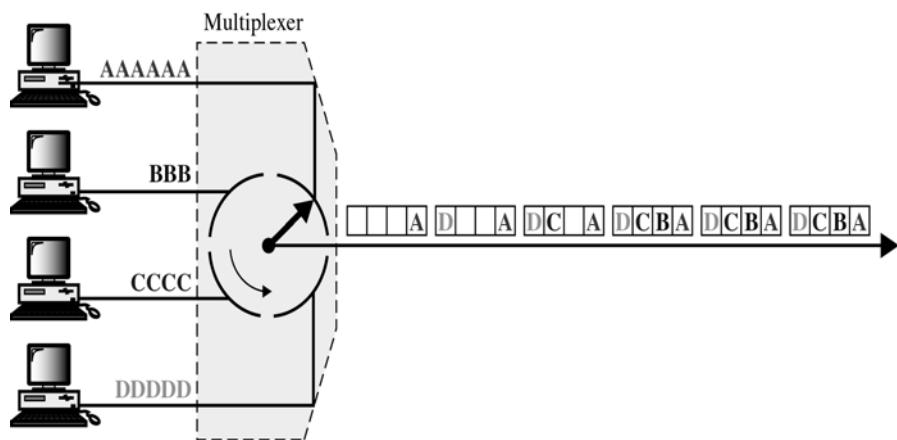
- The time slots dedicated to a given device occupy the same location in each frame and constitute that device's channel



- In figure, we have 5 I/p lines Multiplexed onto a single path using synchronous TDM
- In this example all of the I/p's have the same data rate, so the number of time slots in each frame is equal to the number of I/p lines

## Interleaving

- Synchronous TDM can be compared to a very fast rotating switch
- As the switch opens in front of a device, the device has the opportunity to send a specified amount of data on to the path
- The switch moves from device to device at a constant rate and in a fixed order
- This process is called INTERLEAVING
- Interleaving can be done by BITS, BYTES or by any other DATA UNIT
- In other words MUX can take one byte from each device, then another byte from each device and so on
- In a given system interleaved units will always be of the same size



- Fig., shows interleaving and frame building
- In the example we interleave the various TXs by character (equal to 1 byte each) but the concept is the same for data units of any length
- Each device is sending a different message
- The MUX interleaves the different and forms them into FRAMES before putting them onto the link
- At the receiver the DEMUX decomposes each frame by extracting each character

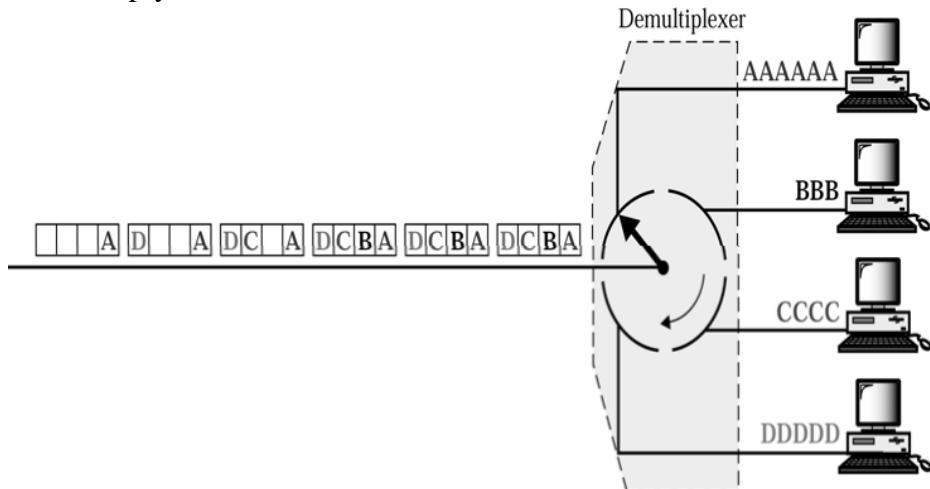
- As a character is removed from a frame, it is passed to the appropriate receiving device

### **Weakness of Synchronous TDM Figure**

- Both figures point out major weakness in Synchronous TDM
- By assigning each timeslot to a specific I/p line, we end up with empty slots whenever not all the lines are active
- In figure only the first three frames are completely filled, The last 3 frames have a collective 6 empty slots
- Having 6 empty slots out of 24 means that a quarter of a capacity of the link is wasted
- **Framing Bits**
- Because the time slots order in a synchronous TDM does not vary from frame to frame, very little overhead information need to be included in each frame
- The order of receipt tells the DEMUX where to direct each time slot so no ADDRESSING is necessary

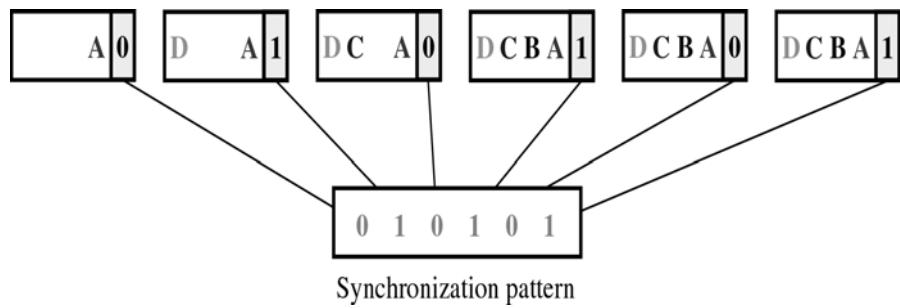
### **Demultiplexing Process**

- Demultiplexer decomposes each frame by extracting each data unit in turn
- Weakness of synchronous TDM
  - Waste of empty slots



### **Framing Bits**

- Various factor however can cause timing inconsistencies.
- For this reason one or more synchronization bits are added to the beginning of each frame
- These bits called Framing bits follow a pattern frame to frame that allows a DEMUX to synchronize with the incoming stream so that it can separate time slots accurately
- This synch info consist of one bit /frame alternating b/w 0 and 1.

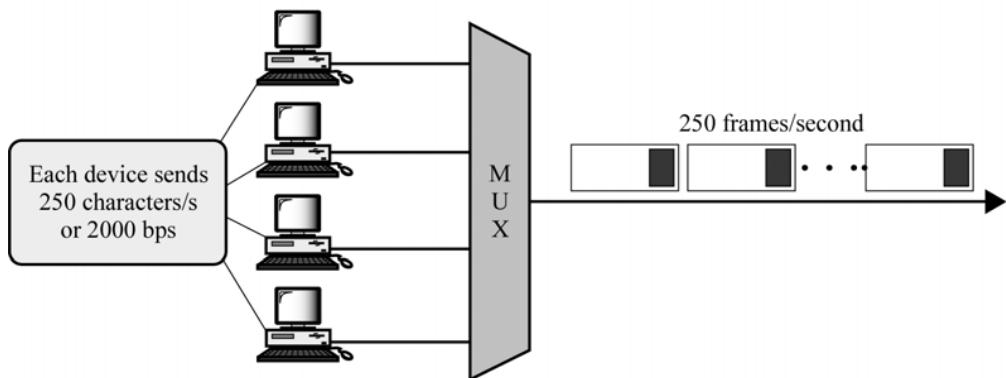


### Synchronous TDM Example

$$8250 \text{ bps} = 250 \text{ frames/second} \times 33 \text{ bits/frame}$$

or

$$8250 \text{ bps} = 4 \times 2000 \text{ bps} + 250 \text{ synchronization bps}$$



### Summary

- Multiplexing
- Frequency division multiplexing
- Wave division multiplexing
- Time division multiplexing

### Reading Sections

- Section 8.1, 8.2, 8.3, 8.4

"Data Communications and Networking" 4th Edition by Behrouz A. Forouzan

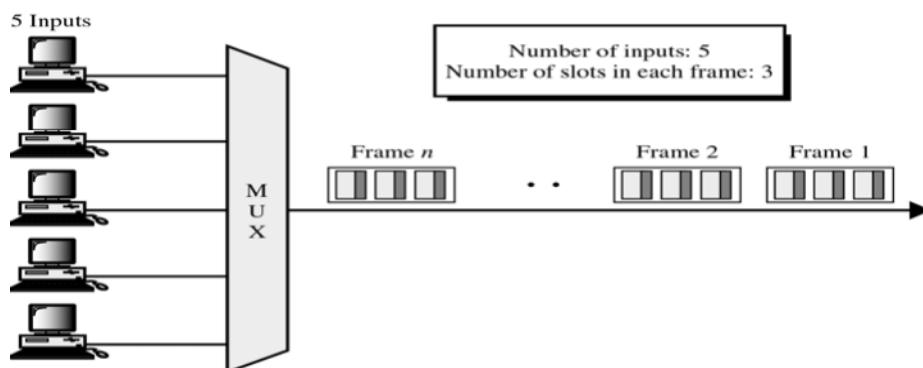
## **LECTURE #30**

### **Bit Stuffing**

- It is possible to connect devices of different data rates to synchronous TDM
- For Example, device A uses one time slot, while the faster device B uses two slots
- The Time slot length is FIXED
- Therefore data rates should be integer multiples of each other
- For example, we can accommodate a device that is 5 times faster than the other device by giving it five slots to one for each of the other devices
- We cannot accommodate a device which is five and a half times faster using this method because we cannot introduce half a time slot into a frame
- When the speeds are not integer multiples of each other, they can be made to behave as if they were
- This is done by a technique known as BIT STUFFING
- In bit stuffing, MUX adds extra bits to a device
- For Example, if we have one device with a bit rate of 2.75 times that of other devices, we can add enough bits to raise this rate to 3 times that of others
- The extra bits are then discarded by the Demultiplexer

### **Asynchronous TDM**

- Synchronous TDM does not guarantee full utilization of the timeslots
- Because the time slots are fixed and pre assigned, whenever a connected device is not transmitting, the corresponding slot is empty and much of the channel capacity is wasted
- For Example, imagine that we have multiplexed the o/p of 20 identical computers onto a single line
- Using synchronous TDM, the speed of that line must be at least 20 times the speed of each i/p line
- But what if only 10 computers are in use at a time?
- Half of the capacity of the line is wasted
- Asynchronous TDM or Statistical TDM is designed to avoid this type of waste
- Asynchronous means flexible or Not fixed
- In an asynchronous system, if we have 'n' input lines, the frame contains no more than 'm' slots, where m is less than n

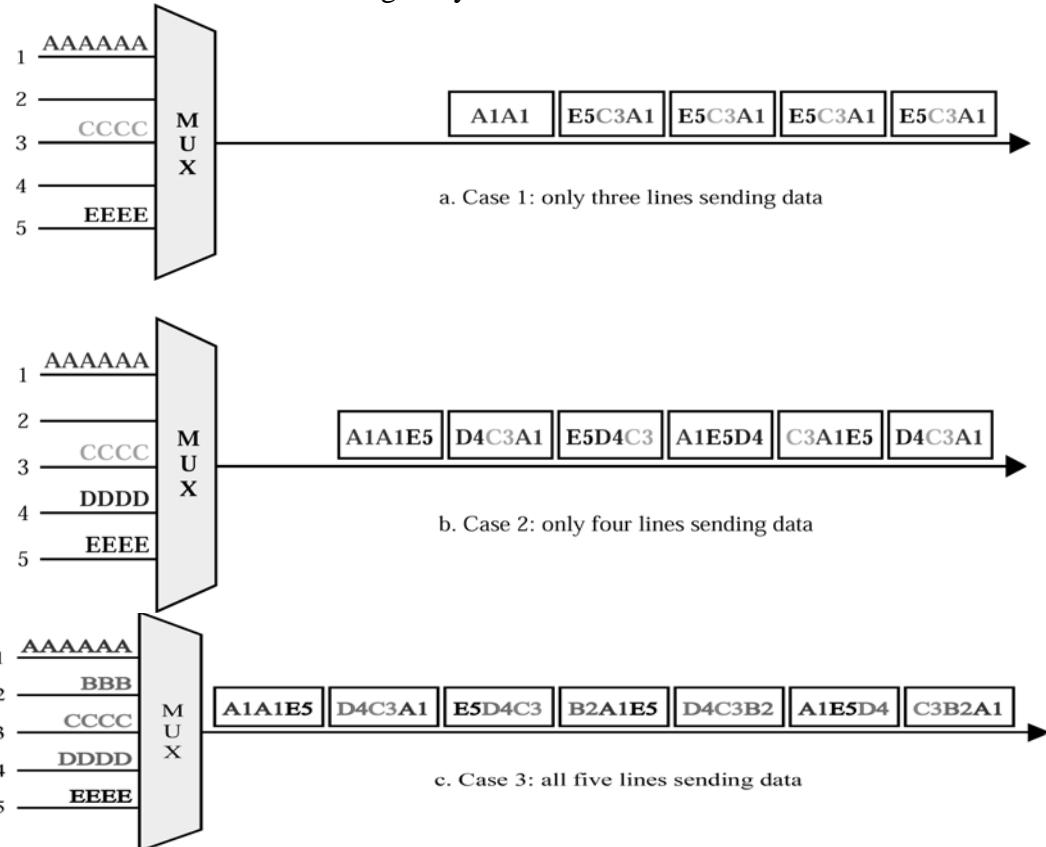


- In this way asynchronous TDM supports the same number of I/p lines as synchronous TDM with a lower capacity link
- A slot is available to any device that wants to send data
- MUX scans I/p lines, accepts data until a frame is filled and then sends the frame across the link

## - Advantages of Asynchronous TDM

- Two major advantages:
- Ability to allocate time slots dynamically
- Lower ratio of time slots to I/p lines

Above two factors greatly reduce the likelihood of a waste



- Fig. shows a system with 5 I/p lines sharing a link using Asynchronous TDM
- Frame size is 3 slots per frame
- Fig shows how MUX handles 3 levels of traffic
- In the first case, only 3 of the 5 computers have data to send
- In the second case, 4 lines are sending data
- In the third case, all devices are sending data
- In each case, MUX scans the devices in order from 1 to 5 filling time slots as it encounters data to be sent

### Y Asynchronous TDM Figure 1

- In the first case, the 3 active i/p lines correspond to the 3 slots in each frame
- For the first 4 frames, the I/p is symmetrically distributed among all the devices.
- By the 5<sup>th</sup> frame however, devices 3 and 5 have completed their transmission but device 1 still has two characters to go

### Y Asynchronous TDM Figure 2

- The MUX picks up the A from device 1, scans down the line without finding another transmission and returns to device 1 to pick up the last A
- There being no data to fill the final slot, the MUX then fills the 5<sup>th</sup> frame with only 2 slots filled
- Compare with Synchronous TX: 6 frames of 5 slots each would be required=30 slots, 14 slots used only
- In second case, there is one more I/p line than there are slots in each frame
- This time MUX scans from 1 to 5 and fills up a frame before each of the lines are checked
- The first frame contains data from device 1, 3, and 4

### **Y Asynchronous TDM Figure 3**

- MUX continues the scan and puts first portion of 5<sup>th</sup> device into the first slot of next frame and so on
- When the number of active senders does not equal the number of slots in a frame, the time slots are not filled symmetrically
- Device 1 occupies the first slot in the first frame, 2<sup>nd</sup> slot in second frame and so on
- In the third case, frames are filled as shown above
- All 5 I/p lines are active
- In this case device 1 occupies the 1<sup>st</sup> slot in the first frame, the 3<sup>rd</sup> slot in the second frame and so on

## **Aspects of Asynchronous TDM**

### **Y Addressing and Overhead**

- Case 2 & 3 above show a major weakness of Asynchronous TDM
- How does the DEMUX know which slot belongs to which output line?
- As opposed to Synchronous TDM, in this case, data from a given device might be in the first slot of one frame and in the third of the next
- Therefore, each time slot must carry an address telling the DEMUX how to direct data
- This address is for local use only attached by the MUX and detached by the DEMUX
- In the figure above address is specified by a digit
- Adding address bits to each time slot increase the overhead of an Asynchronous system and limits its efficiency
- Addresses usually consist of only a small number of bits
- Need for Addressing makes Asynchronous TDM inefficient for bit or byte interleaving
- Imagine bit interleaving with each bit carrying an address
- One bit of data plus 3 bits of address
- Asynchronous TDM is efficient only when the size of the time slot is kept relatively large

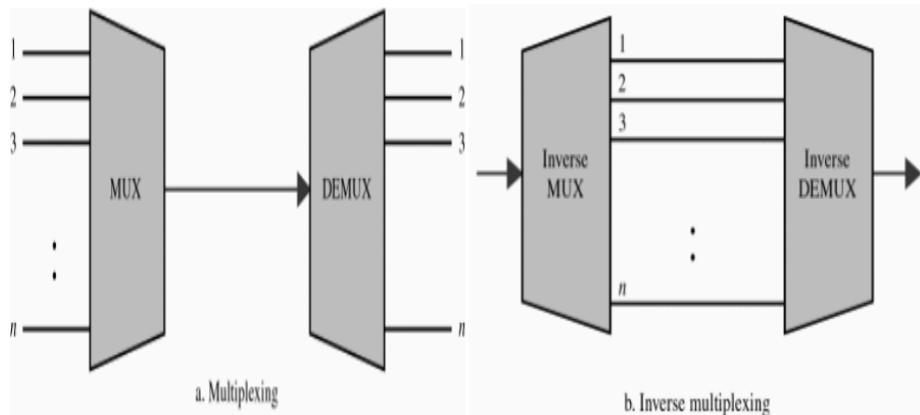
## **Inverse Multiplexing**

- Opposite of Multiplexing
- Takes data from one high speed line and breaks it into portions that can be sent over several lower speed lines simultaneously

### **Why do we need Inverse Multiplexing?**

- An organization wants to send data, voice and video each of which requires a different data rate
- To send voice it needs 64Kbps,
- To send data, it needs 128 Kbps link

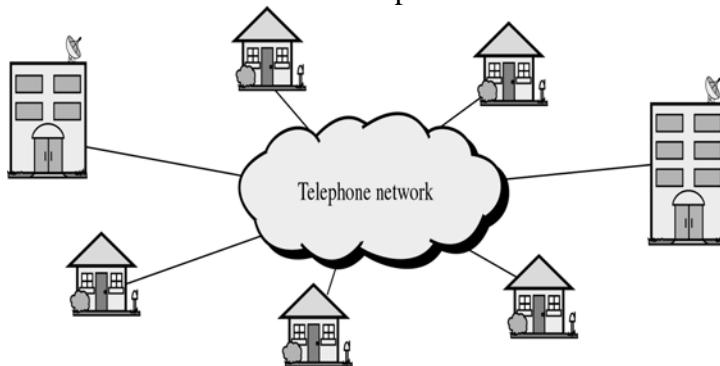
- To send video it may need 1.544 Mbps link
- It can lease a 1.544 Mbps line from a common carrier and only use it fully for sometime
- Or it can lease several separate channels of lower data rates
- Voice can be sent over any of these channels
- Data & Video can be broken into smaller portions using Inverse Multiplexing and TX



### - Multiplexing Application

#### THE TELEPHONE SYSTEM

- Multiplexing has long been used as an essential tool in the Telephone industry
- A country's telephone system may include various carriers that offer local and long-distance service
- These various carriers form a Telephone Network I.e. PTCL



•Each subscriber is connected to the telephone network as a service line

#### Summary

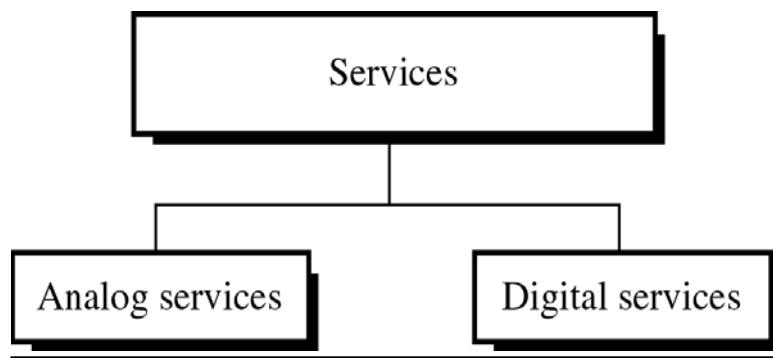
- Time Division Multiplexing
- Asynchronous TDM
- Inverse Multiplexing
- The Telephone System

#### Reading Sections

- Section 8.4,8.5 "Data Communications and Networking" 4th Edition by Behrouz A. Forouzan

# **LECTURE #31**

## **Telephone Services**

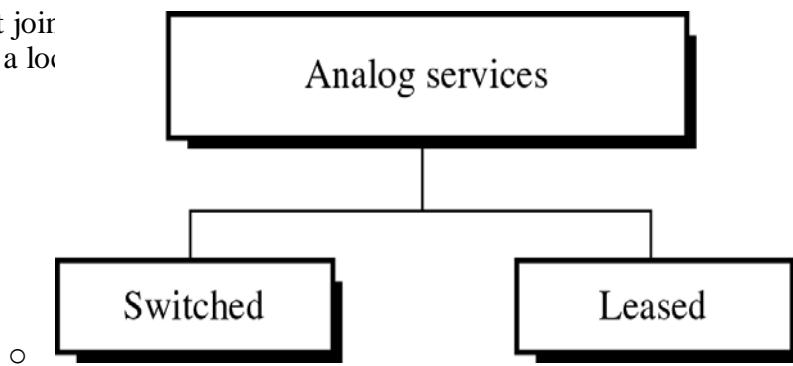


### **- Common carrier Services & Hierarchies**

- Telephone companies began by providing their subscribers with ANALOG services using ANALOG networks
- Later digital services were introduced
- Nowadays carriers are even thinking about changing their service lines digital too
- Soon the entire n/w will be digital
- For now both types of services are available and both FDM and TDM are in use

### **- Analog Services**

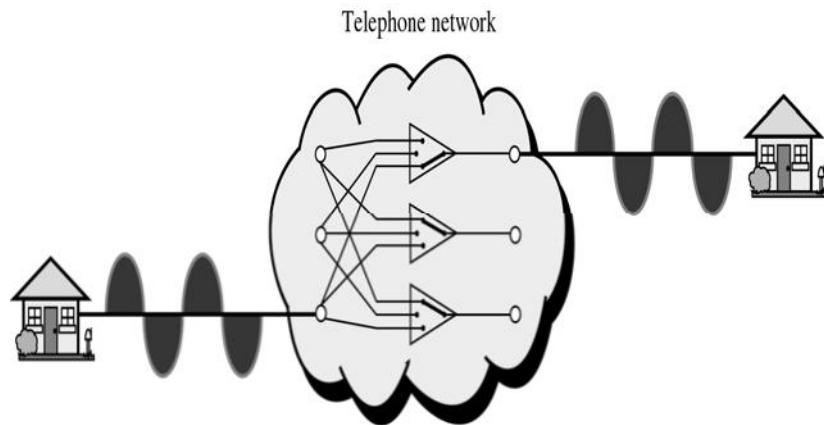
- Of the many analog services available to telephone subscribers, two are really important:
  - Switched Analog Services
  - Leased Analog Services
- Familiar dial up service most often encountered when using a home telephone
- Uses twisted pair cable to connect subscriber's phone to the network via exchange
- This connection is called LOCAL LOOP
- The n/w it joir
- Signal on a lo



## **Switched Analog Services**

With switched lines, caller dials a number and call is conveyed to a series of switches at the exchange

The switch connects two lines for the duration of the call



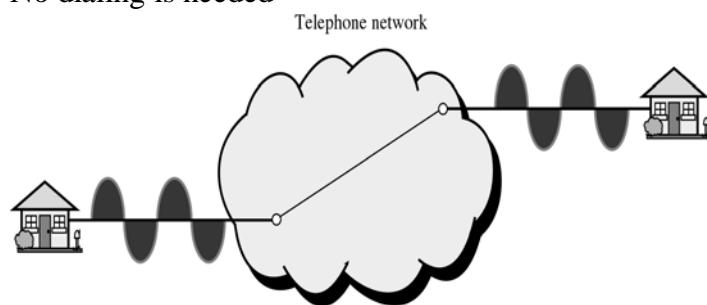
## **Analog Leased Service**

Offers customers the opportunity to lease a line, sometimes called a Dedicated Line

Dedicated line is permanently connected to the other customer

Although the connection still passes through switches in the exchange, the customer experiences it as a single line because switch is always closed

No dialing is needed

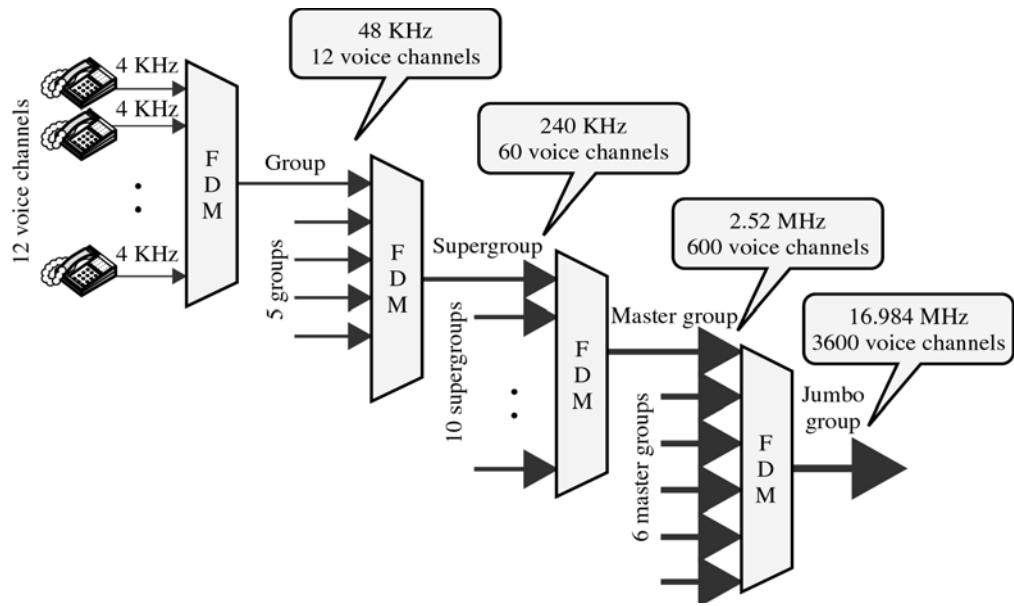


## **Conditioned Lines**

- Another service offered by the carriers
- Conditioning means improving the quality of a line by lessening attenuation, signal distortion or delay distortion
- Conditional lines are Analog but their quality makes them suitable for digital data communication when connected to Modems

## **Analog Hierarchy**

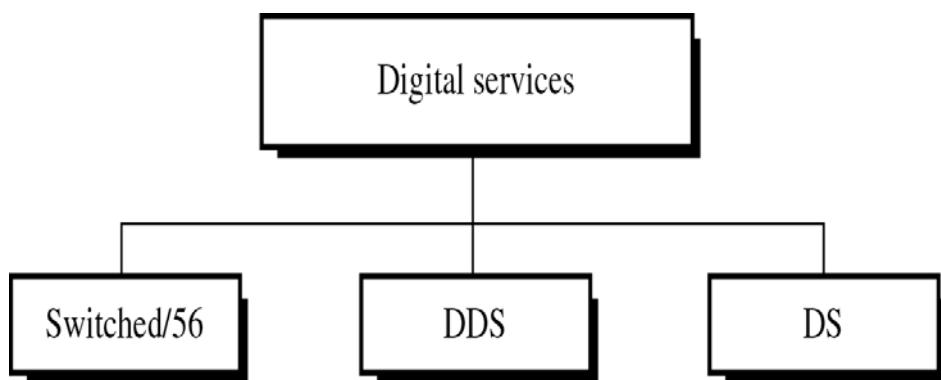
- To maximize efficiency, telephone companies multiplex signal from lower BW lines onto higher BW lines
- In this way many switched or leased lines can be combined into fewer but bigger channels
- FDM is used for analog lines



### - Digital Services

- o Digital services are largely offered nowadays
- o Digital services are less sensitive than analog ones
- o Telephone line acts as an antenna and pick up noise both in case of analog and digital TX
- o In analog, both noise and signal are analog
- o In digital, signal can easily be separated

### Types of Digital Services



#### Switched / 56 Service

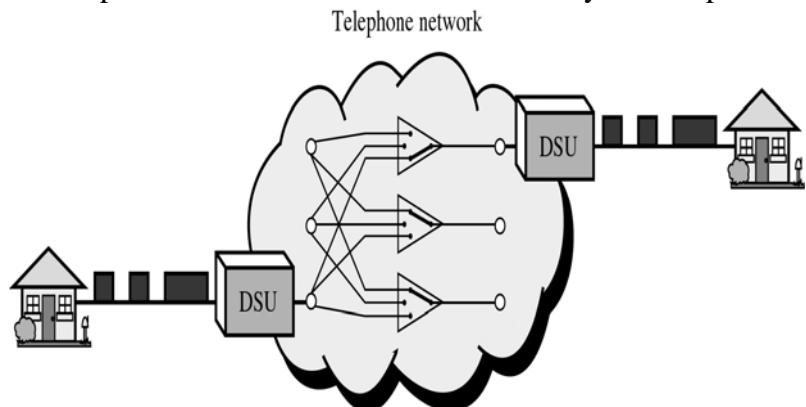
It is digital version of Analog switched line

Allows data rates of up to 56 Kbps

Both parties must subscribe to the service

A caller with normal telephone service cannot connect to a telephone or computer with this service even with a Modem

Because this service is already digital, subscribers do not need modems  
They need DSU (Digital service unit)  
DSU changes the rate of digital data created by the subscriber's device to 56 Kbps and encodes it in the format used by service provider



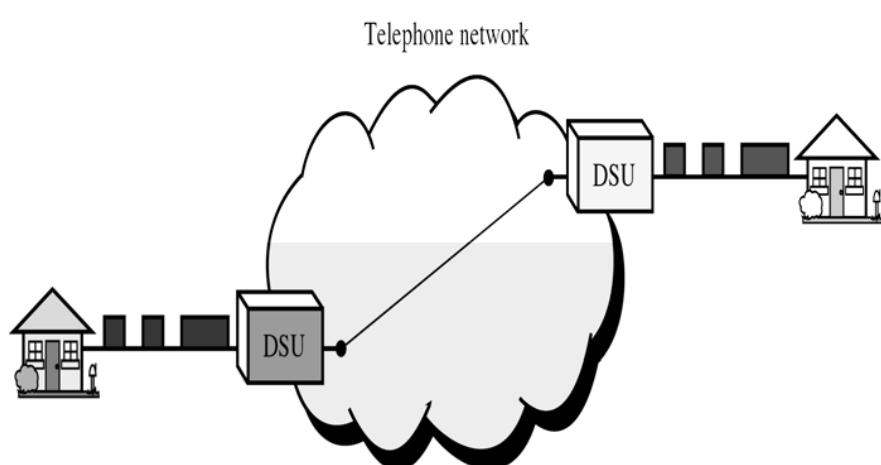
DSU is often included in dialing process  
DSU is more expensive than Modem  
So why would a subscriber pay for switched/56 service and DSU?  
Because digital line has better speed, better quality and less susceptibility to noise  
Bandwidth on Demand

### Digital Data Service (DDS)

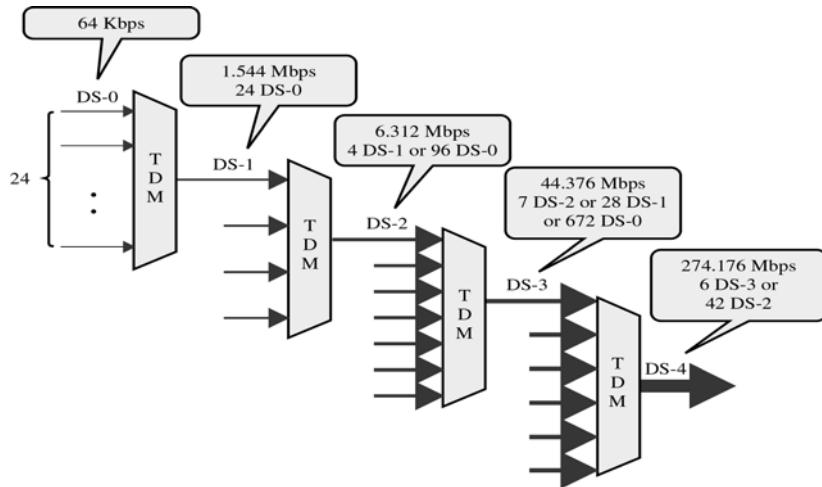
Digital version of analog leased line  
Max. data rate of 64 Kbps  
Like switched/56, DDS requires a DSU  
DSU for this service is cheaper than switched /56 DSU because it does not need a dial pad

### Digital Signal (DS) Service

Telephone companies saw the need to develop a hierarchy of digital services much like those used for Analog services  
The next step was DS service  
DS is a hierarchy of digital signals



DS-0 resembles DDS. It is single digital channel of 64 Kbps  
 DS-1 is 1.544 Mbps service =  $24 * 64 \text{ Kbps} + 8 \text{ Kbps}$  of overhead  
 DS-2 -  $76.312 \text{ Mbps} = 96 * 64 \text{ Kbps} + 168 \text{ Kbps}$   
 DS-3 -  $744.376 \text{ Mbps} = 672 * 64 \text{ Kbps} + 1.368 \text{ Mbps}$  of overhead  
 DS-4 -  $7274.176 \text{ Mbps} = 4032 * 64 \text{ Kbps} + 16.128 \text{ Mbps}$  overhead



## T Lines

- DS-0 DS1 are the names of the Services
- To implement those services, telephone companies use T-lines
- These are the lines with capacities matched to the data rates of DS-0 to DS-4

<i>Service Line</i>	<i>Rate (Mbps)</i>	<i>Voice channels</i>
DS-1	1.544	24
DS-2	6.312	96
DS-3	44.376	672
DS-4	274.176	4032

## Summary

- The Telephone System
- Analog Services
- Digital Services
- Fiber To The Curb (FTTC)

## Reading Sections

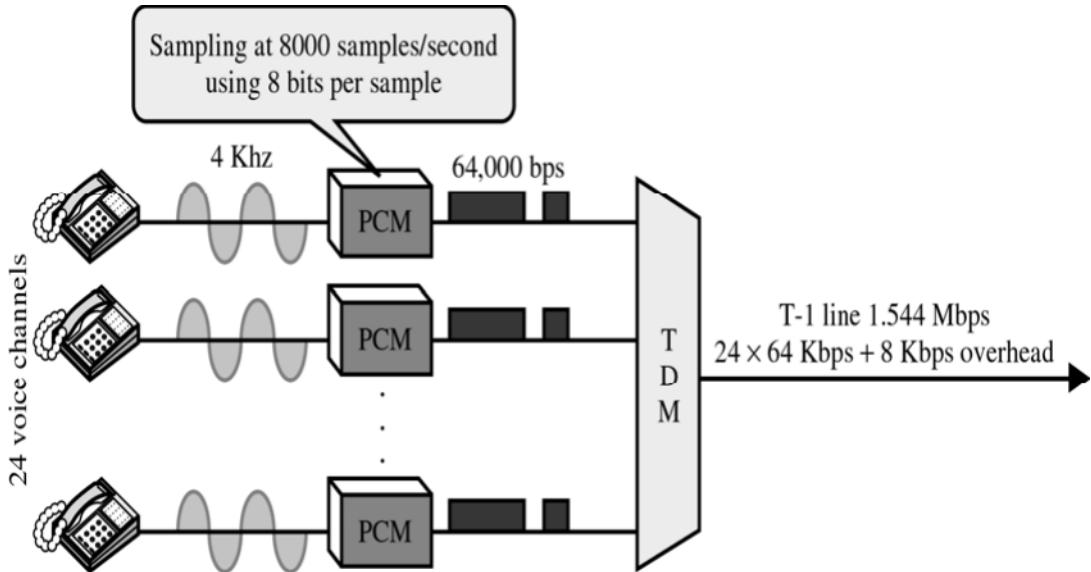
- Section 8.5, ‘‘Data Communications and Networking’’ 4th Edition by Behrouz A. Forouzan

# **LECTURE #32**

## **T Lines**

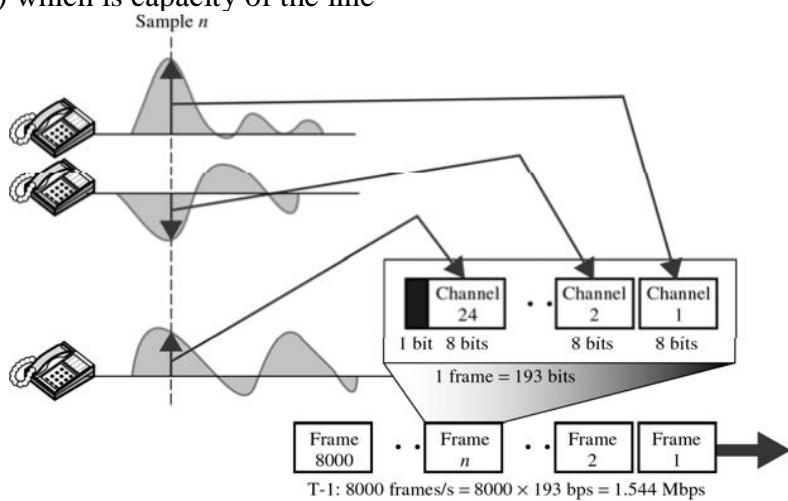
### **- T Lines for Analog Lines**

- T Lines are digital lines designed for digital data
- However they can also be used for analog transmission (Telephone connections)
- Analog signals are first sampled and the Time Multiplexed



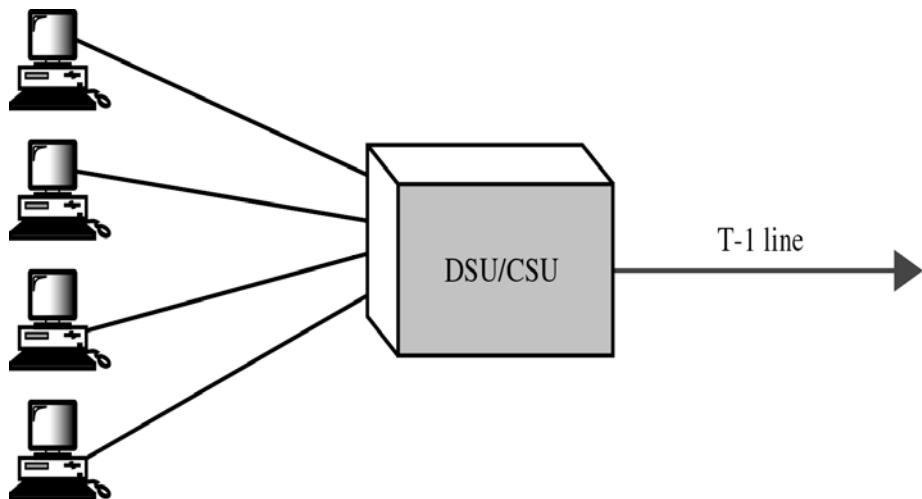
### **Y T 1 Frame (Figure)**

- DS-1 requires 8 Kbps of overhead
- To understand this overhead, let's examine format of a 24-voice channel frame
- Frame used on T-1 line is usually 193 bits divided into 24 slots of 8 bits each + 1 bit for synchronization ( $24 \times 8 + 1 = 193$ )
- 24 segments are interleaved in one frame
- If a T-1 carries 8000 frames, the data rate is 1.544 Mbps ( $193 \times 8000 = 1.544$  Mbps) which is capacity of the line



#### Y Fractional T Lines (Figures)

- Many subscribers do not need the entire capacity of the T line
- For example, a small business may need only one-fourth of the capacity of a T-1 line
- If four business of same size lie in same building, they can share a T-1 line
- DSU/CSU allow the capacity of T-1 line to be interleaved into 4 channels



#### E-Lines

- Europeans use a version of T-lines called E-lines
- Two are conceptually identical but capacities vary

Line	Rate(Mbps)	Voice channels
E-1	2.048	30
E-2	8.448	120
E-3	34.368	480
E-4	139.264	1920

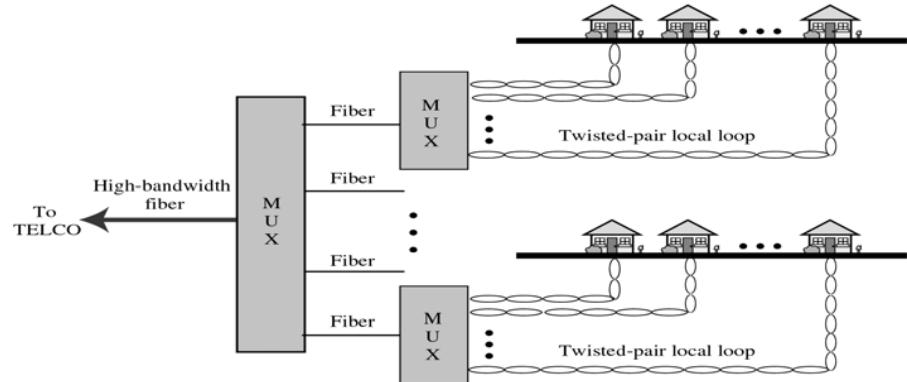
#### Fiber to the Curb (FTTC)

- Optical fiber has many advantages:
  - Noise Resistance
  - High BW Capacity
- **Very Expensive**
- Telephone & Cable companies have devised FTTC to employ optical fiber while keeping the expense down
- Optical Fiber is the medium from the Central Office of the telephone company or from main office of a cable company to the Curb
- The medium from the Curb to the subscriber's premises is less expensive twisted pair or coaxial cable

#### Y FTTC in Telephone Network

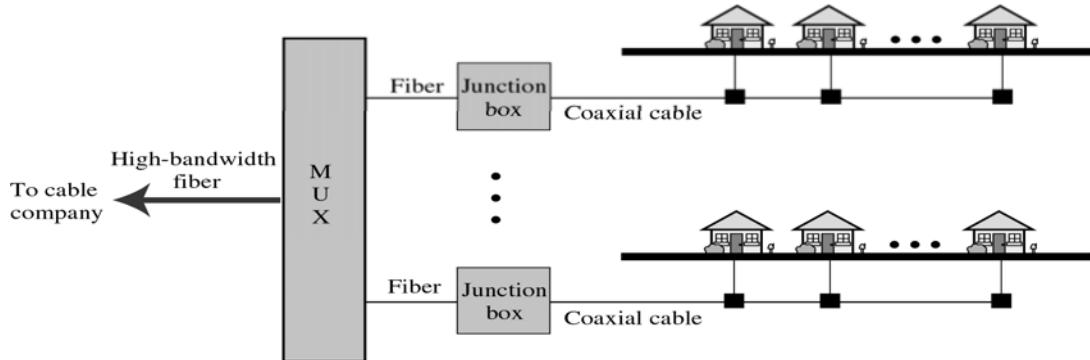
- Optical Fiber is used to connect and multiplex different voice channels
- Copper twisted pair coming from individual premises is multiplexed in junction boxes and converted to optical signals. Optical signals are

multiplexed using WDM at switching office to create wider BW optical



#### Y FTTC in Cable TV Network

- Cable TV uses optical fibers to connect and multiplex different cable channels
  - Coaxial cables coming from individual premises are multiplexed in the junction box and converted to the optical signals
- Optical signals are multiplexed using WDM at switching office to create wider BW optical signals

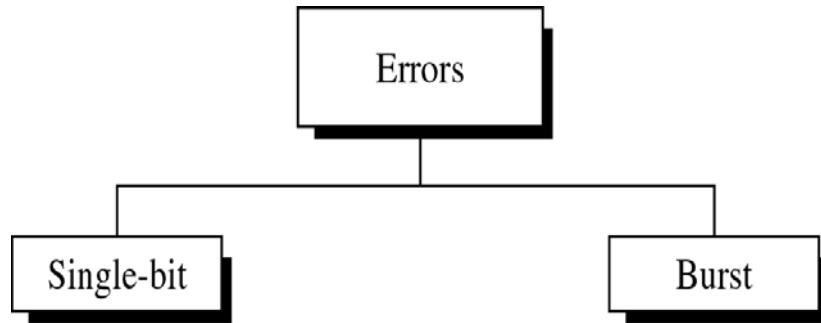


## ERROR DETECTION AND CORRECTION

### Introduction

- Networks must be able to transfer data from one device to another with complete accuracy
- A system that cannot guarantee that the data received by one device is identical to the data transmitted by another device is essentially useless
- Anytime the data is TX from source to destination, it gets corrupted on the way
- Many factors including NOISE can alter or wipe out one or more bits of a given data unit
- Reliable systems must have a mechanism for detecting and correcting such ERRORS
- Error detection and Correction is implemented either at the data link layer or at the Transport layer

### Types of Errors



- Whenever an EM signal flows from one point to the other, it is subject to unpredictable interference from heat, magnetism and other forms of electricity
- This interference can change the shape or timing of the signal
- If the signal is carrying encoded binary data, such changes can alter the meaning of the data

### **Summary**

- T-Lines in Analog Transmission
- Fractional T-Lines
- E-Lines
- Fiber To The Curb (FTTC)
- Error Detection and Correction

### **Reading Sections**

Section 8.5, 8.7, 9.1,

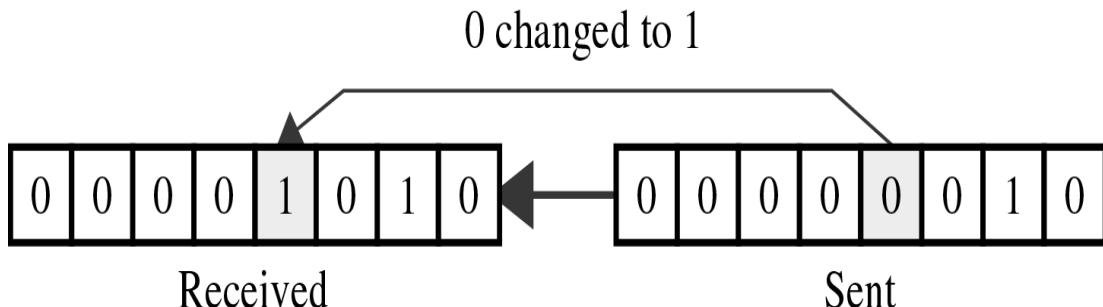
“Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

## **LECTURE #33**

### **Types of Errors**

#### **- Single Bit & Burst**

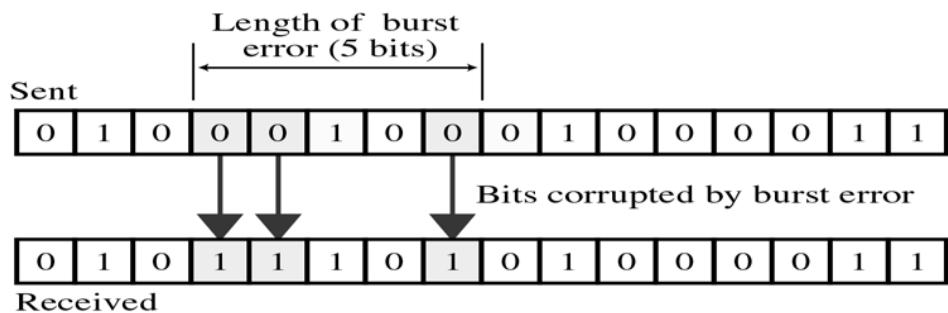
- In a single-bit error, a 0 is changed to a 1 or a 1 to a 0
- In a burst error multiple bits are changed
- For Example, a 0.01 second burst of impulse noise on a TX with a data rate of 1200 bps might change all or some of 12 bits of information



- The term single bit error means that only one bit of a given data unit (such as a byte, character, or a packet) is changed from 1 to 0 or from 0 to 1
- Figure shows the effect of a single bit error on a data unit
- ASCII character 000000010 (ASCII STX) is sent but 00001010 (ASCII LF) is received
- Single bit errors are the least likely type of error in serial data TX
- To see Why? Imagine a sender sends data at 1Mbps
- This means that each bit lasts only  $1/1,000,000$  seconds or 1 microsecond
- For single bit error to occur, the noise must have a duration of 1 microsecond which is very rare, noise lasts much longer than that
- However, single bit error can occur if we are sending data using parallel TX
- For Example, if 8 wires are used to send all of the eight bits of a byte at the same time and one of the wires is noise, one bit can be corrupted in each byte

#### **- Burst Errors**

- The term burst error means that two or more bits in the data unit have changed from 1 to 0 or from 0 to 1



- Figure shows the effect of a burst error on a data unit
- In this case 0100010001000011 was sent but 0101110101000011 was received
- Note that a burst error does not necessarily mean that error occurs in consecutive bits

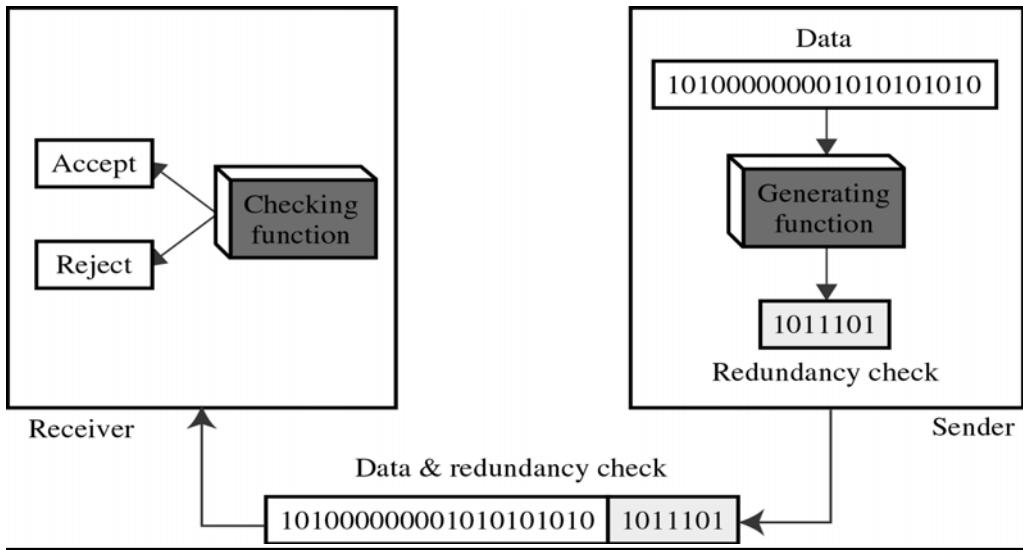
- The length of the burst is measured from the first corrupted bit to the last corrupted bit
- Some bits in b/w may not have been corrupted
- Burst error is most likely to happen in a serial TX
- The duration of the noise is normally longer than the duration of a bit which means that when noise affects data, it affects a set of bits
- The number of bits affected depends on the data rate and duration of noise
  - For example, if we are sending data at 1 Kbps, a noise of 1/100 seconds can affect 10 bits
  - If we are sending data at 1 Mbps, the same noise can affect 10,000 bits

## Error Detection

- Even if we know what type of errors can occur, will we recognize one when we see it?
- If we have a copy of the intended TX for comparison, of course we will
- But what if we don't have a copy of the original
- Then we will have no way of knowing we have received an error until we have decoded the TX and failed to make sense of it
- For a device to check for errors this way will be Costly and Slow
- We don't need a machine that decodes every thing and then sits and decides whether a specific word makes sense or not
- We need a mechanism that is Simple and Completely objective

## Y Redundancy

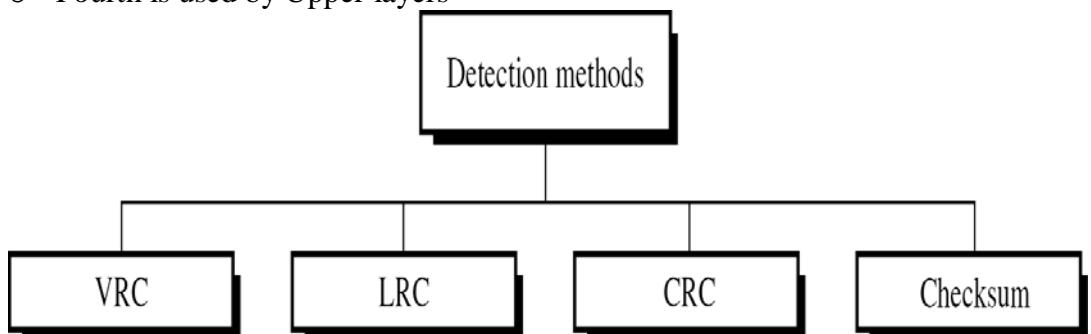
- One error detection mechanism that would satisfy these requirements would be to send every data unit twice
- The receiving device would then be able to do a bit-for-bit comparison b/w two TXs
- Any discrepancy will indicate an error and an appropriate error correction mechanism could be set in place
- This system will be completely Accurate because the odds of error affecting the same bits in both version will be infinitesimally small
- But this system will be extra ordinarily SLOW
- Not only will the TX time double, but the time it takes to compare two data units is also added up
- The concept of including extra information in the TX solely for the purpose of comparison is a good one
- But instead of repeating the entire data stream, a shorter group of bits may be appended to the end of each unit
- This technique is called REDUNDANCY because the extra bit are redundant to the information and are discarded as soon as the accuracy of TX has been determined



- Fig shows the process of using redundant bits to check the accuracy of data unit
- Once the data stream has been generated, it passes through a device that analyzes it and adds on an appropriately coded redundancy check
- The data unit now enlarged by several bits (7) travels over the link to the receiver
- The receiver puts the entire stream through a checking function
- If the received bit stream passes the checking criteria, the data portion of the data unit is accepted and the redundant bits are discarded

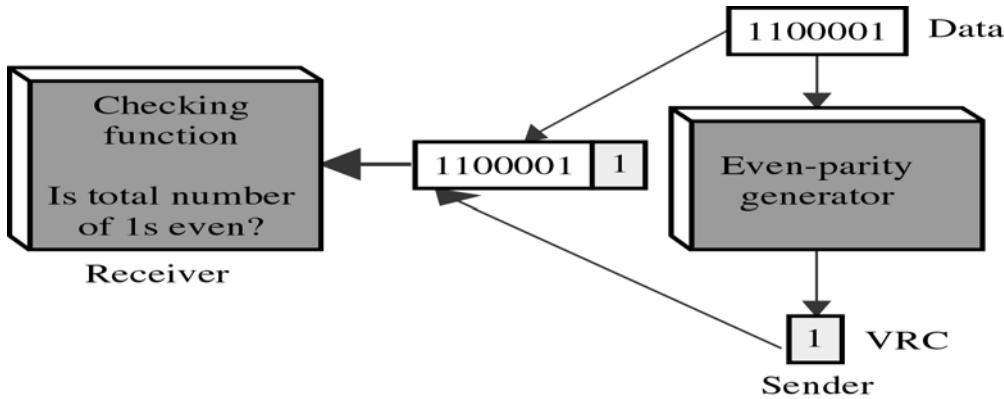
#### – Types of Redundancy Checks

- There are 4 types of redundancy checks used in data communication:
  - Vertical Redundancy Check (VRC)
  - Longitudinal Redundancy Check (LRC)
  - Cyclic Redundancy Check (CRC)
  - Checksum
- First 3 are normally implemented in the physical layer for use in data link layer
- Fourth is used by Upper layers



#### Y Vertical Redundancy Check (VRC)

- Most common and least expensive
- Also called Parity Check
- A redundant bit called parity bit is appended to every data unit so that total number of 1's in the unit becomes even including the parity bit



- We want to TX the binary data unit 1100001
- Adding together the number of 1's gives us 3, an odd number
  - Before TX, we pass the data unit through a parity generator, which counts the 1's and appends the parity bit (1) to the end
  - The total number of 1's is now 4, an even number
  - The system now transfers the entire expanded across the network link
  - When it reaches its destination, the RX puts all 8 bits through an even parity checking function
  - If the RX sees 11100001, it counts four ones, an even number and the data unit passes
  - But what if the data unit has been damaged in transit?
  - What if instead of 11100001, receiver sees 11100101?
  - Then when the parity checker counts the 1's, it gets 5 an odd number
  - The receiver knows that an error has occurred somewhere and therefore rejects the whole unit
  - Some systems may also use ODD parity checking
  - The principal is the same as even parity

./ Example 9.1, 9.2, 9.3

### **Example 9.1**

Sender wants to send “world” In ASCII, the five characters are coded as:

110111 w	1110111 <u>0</u>
1101111 o	1101111 <u>0</u>
1110010 r	1110010 <u>0</u>
1101100 l	1101100 <u>0</u>
1100100 d	1100100 <u>1</u>

### **Example 9.2**

- | Suppose “world” is received by the receiver without being corrupted:

- Receiver counts the 1's and passes it

1110111 w	1110111 <u>0</u>	6
1101111 o	1101111 <u>0</u>	6
1110010 r	1110010 <u>0</u>	4
1101100 l	1101100 <u>0</u>	4
1100100 d	1100100 <u>1</u>	4

### **Example 9.3**

- Suppose “world” is received by the receiver corrupted:

- Receiver counts the 1's and rejects it

1110111	w	1111111	0	7
1101111	o	1101111	0	6
1110010	r	1110110	0	5
1101100	l	1101100	0	4

- **Performance of VRC**

- VRC can detect all single bit errors
- Can also detect Burst errors as long as the total number of bits changed is ODD (1,3,5 etc)

#### **Examples**

- We have an Even Parity data unit where the total number of 1's including the parity bit is ‘6’ : 1000111011
- If 3 bits change value resulting parity will be odd and an error will be detected: 1111111011: 1's = 9
- If 2 bits change value resulting parity will still be even and error will not be detected: 1110111011: 1's = 8
  - VRC cannot detect errors when the total number of bits changed are even
  - VRC checker will return a result of 1 and the data unit will be rejected
  - The same hold true for any odd number of errors
  - In the second case, VRC checker will check parity and will return an even number although the data unit contains two errors
- VRC cannot detect error when the number of bits changed is even
- If any two bits change in TX, the changes cancel each other and the data unit will pass a parity check even though the data unit is damaged
- Same holds true for any even number of errors

#### **Summary**

- Types of Errors
- Error Detection Techniques
- Redundancy
- Types of Redundancy Checks

#### **Reading Sections**

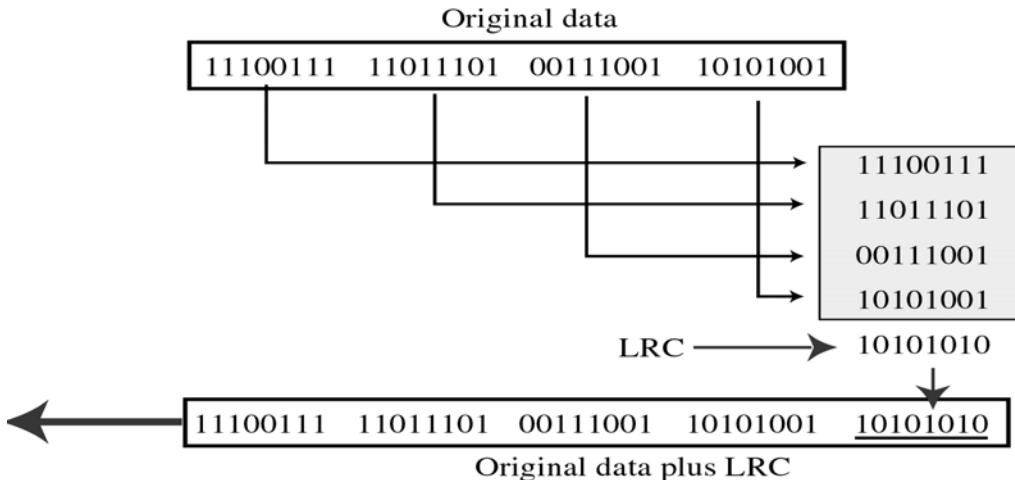
- Section 9.1, 9.2, 9.3  
“Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #34**

## **Error Detection And Correction Methods**

### **- Longitudinal Red Check(LRC)**

- In LRC, a block of bits is organized in a table (rows and columns)
- For example instead of sending 32 bits, we organize them in a table made of 4 rows and 8 columns



- We then calculate the Parity bit for each column and create a new row of 8 bits which are the parity bits for the whole block
- Note that the first parity bit in the 5<sup>th</sup> row is calculated based on all the first bits
- The second parity bit is calculated based on all the second bits and so on
- We then attach the 8 parity bits to the original data and send them to the receiver

### **Example 9.4**

Suppose the following block is sent:

10101001  
00111001  
11011101  
11100111

10101010 (LRC)

- It is hit by a burst of length 8 and some bits are corrupted:

10100011  
10001001  
11011101  
11100111

10101010 (LRC)

- Receiver checks LRC, some of bits do not follow even parity rule and whole block is discarded

10100011  
10001001  
11011101  
11100111

10101010 (LRC)

- **Performance of LRC**

- Burst errors can be detected more often
- As shown in the last example, an LRC of ‘n’ bits can easily detect a burst error of ‘n’ bits
- A burst error of more than ‘n’ bits is also detected by LRC with a very high probability
- One pattern of errors remain elusive
- If two bits in one data unit are changed and two bits in exactly the same place in another data unit are also damaged

For Example:

- Original data units

11110000  
11000011

- Changed data units

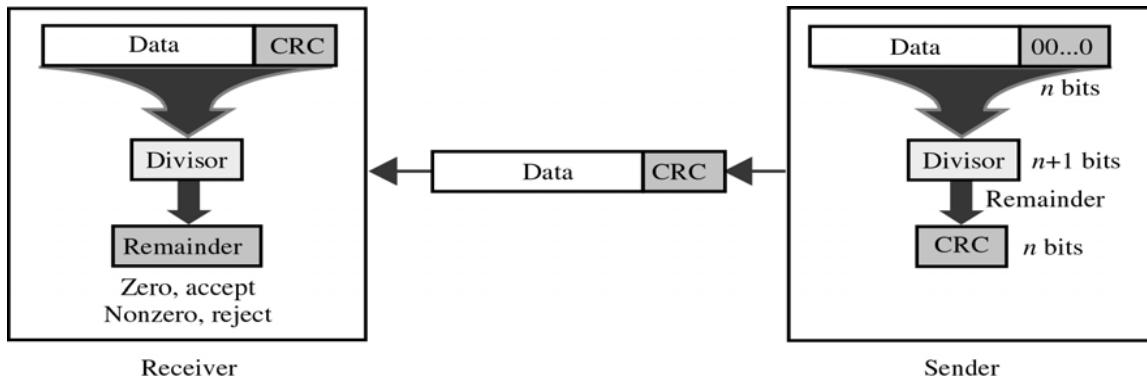
01110001  
01000010

- **Cyclic Redundancy Check (CRC)**

- Most powerful of checking techniques
- VRC and LRC are based on Addition
- CRC is based on Binary Division
- A sequence of redundant bits called CRC or CRC remainder is appended to the end of the data unit, so that the resulting data unit becomes exactly divisible by a second predetermined binary number
- At its destination, the data unit is divided by the same number
- If at this step, there is no remainder, the incoming data unit is assumed to be intact and is therefore accepted
- A remainder indicates that a data unit has been damaged and therefore must be rejected

### **Qualities of CRC**

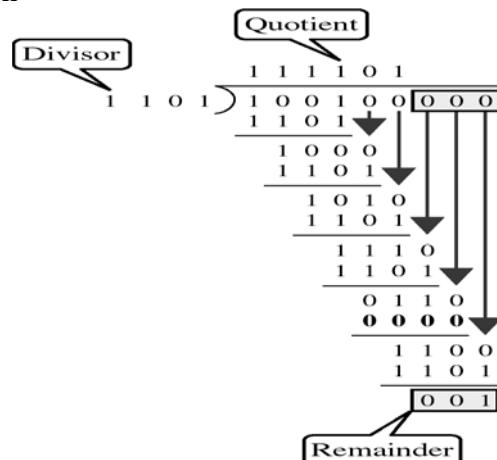
- To be valid the CRC must have two qualities:
  - It must have exactly one less bit than the divisor
  - Appending it to the end of the data must make the resulting bit sequence exactly divisible by the divisor



- **First** a string of  $n$  0's is appended to the data unit
- The number ' $n$ ' is one less than the number of bits in the predetermined divisor, which is  $n+1$  bits
- **Secondly**, newly elongated data unit is divided by the divisor using a process called binary division. The remainder resulting from this division is the CRC
- **Third**, the CRC of ' $n$ ' bits replaces the appended 0's at the end of the data unit
- Note that CRC may consist of all zeros
- The data unit arrives at the receiver followed by the CRC
- The receiver treats the whole string as a unit and divides it by the same divisor that was used to find the CRC remainder
- If string arrives without an error, the CRC checker yields a remainder of zero and data unit passes
- If the string has been changed in transit, the division yields a non-zero remainder and the data unit does not pass

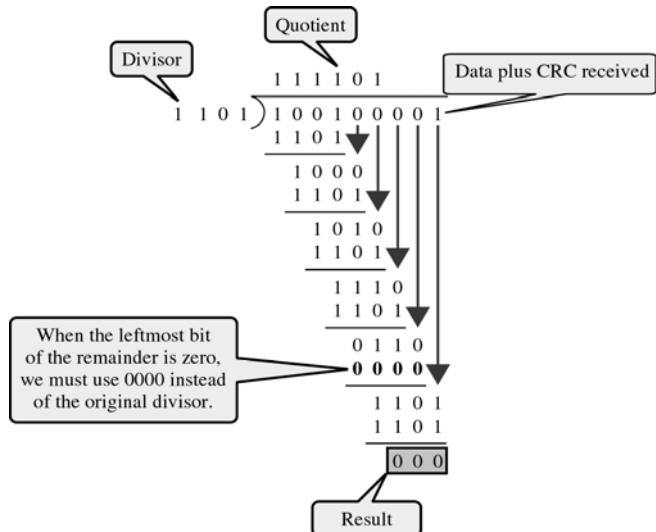
### The CRC Generator

- Uses Modulo-2 Division



### The CRC Checker

- Functions exactly like CRC Generator

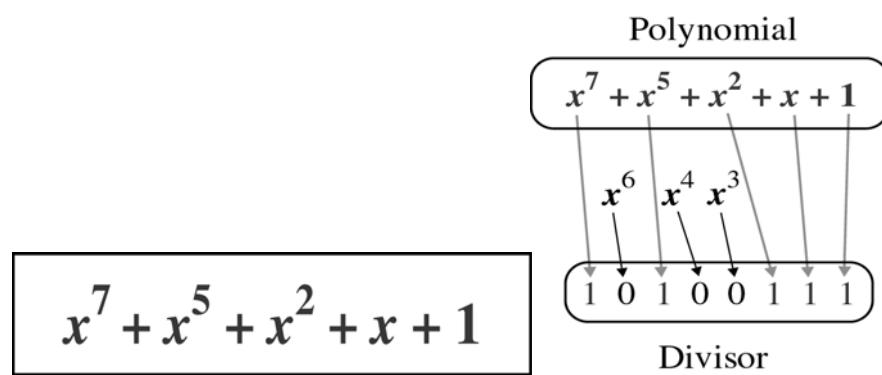


## Polynomials

- CRC generator (the divisor) is most often represented not as a string of 1's and 0's but as an algebraic polynomial
- The polynomial format is useful for two reasons:
  - It is short
  - Can be used to prove the concept mathematically

## Selection of a Polynomial

- A polynomial should have the following properties:
  - It should not be divisible by 'x'
  - It should be divisible by 'x+1'
- The first condition guarantees that all burst errors of a length equal to the degree of the polynomial are detected
- The 2<sup>nd</sup> guarantees that all burst errors affecting an odd number of bits are detected



## Popular Polynomials for CRC

CRC-12

$$x^{12} + x^{11} + x^3 + x + 1$$

CRC-16

$$x^{16} + x^{15} + x^2 + 1$$

CRC-ITU-T

$$x^{16} + x^{12} + x^5 + 1$$

CRC-32

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

### - Performance of CRC

- o CRC can detect all burst errors that affect an odd number of errors
- o CRC can detect all burst errors of length less than or equal to the degree of the polynomial
- o CRC can detect with a very high probability burst errors of length greater than the degree of the polynomial

### Example 9.6

- The CRC-12 ( $x^{12} + x^{11} + x^3 + x + 1$ ) has a degree of 12
- It will detect
  - All burst errors affecting odd no. of bits
  - All burst errors with a length equal to or less than 12
  - 99.97 % of the time burst errors with a length of 12 or more

### Summary

- Types of Redundancy Checks
- Longitudinal Redundancy Check (LRC)
- Cyclic Redundancy Check (CRC)

### Reading Section

- Section 9.4, 9.5  
“Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #35**

## **Error Correction And Detection Method**

### **- CHECKSUM**

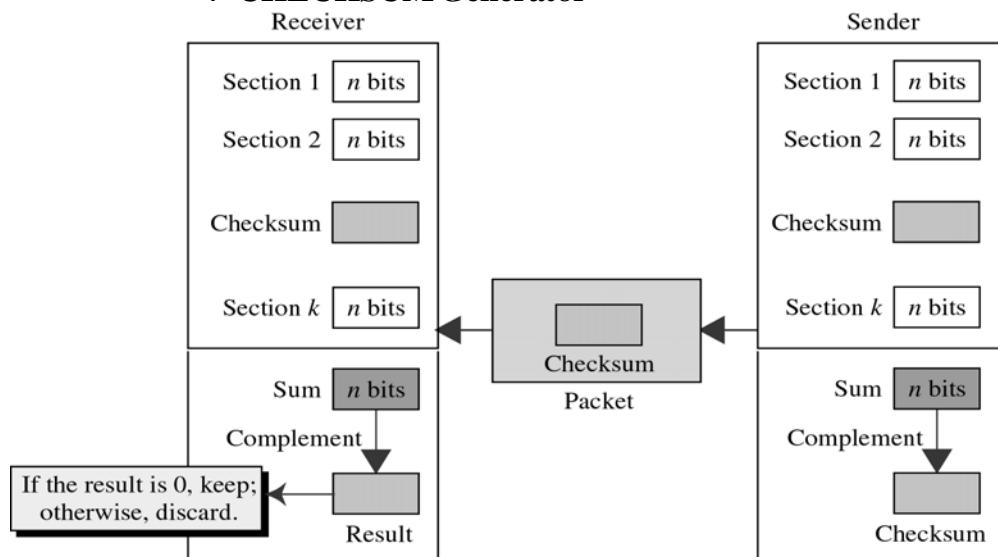
- Error detection method used by the Higher Layers
- Like VRC, LRC, CRC, Checksum is also based on the concept of redundancy

### **One's Complement**

Finding one's complement

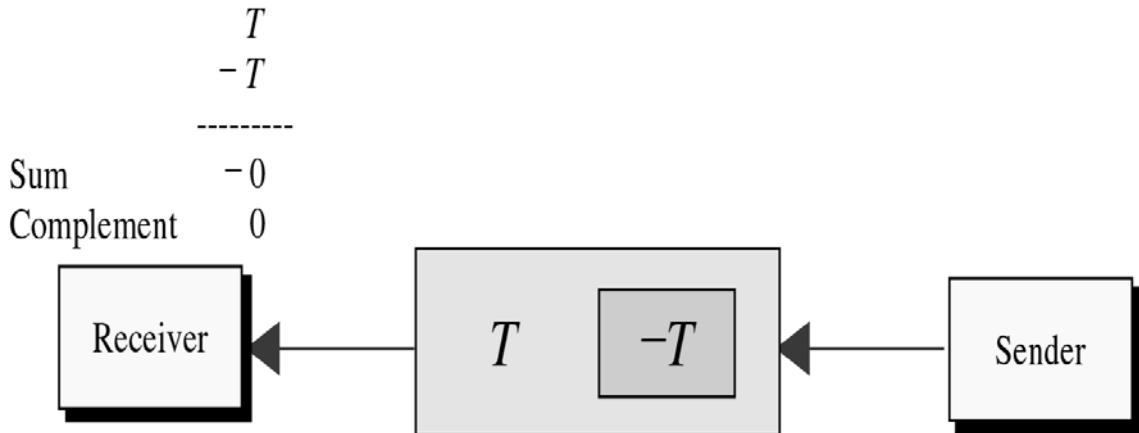
- Invert every 1 to 0 and 0 to 1
- A and  $-A$  are one's complement of each other
- $+A = 1010 \rightarrow -A = 0101$
- $+0 = 0000 \rightarrow -0 = 1111$
- Error detection method used by the Higher Layers
- Like VRC, LRC, CRC, Checksum is also based on the concept of redundancy

### **Y CHECKSUM Generator**



- The sender subdivides data units into equal segments of ' $n$ ' bits(16 bits)
- These segments are added together using one's complement
- The total (sum) is then complemented and appended to the end of the original data unit as redundancy bits called CHECKSUM
- The extended data unit is transmitted across the network
- The receiver subdivides data unit as above and adds all segments together and complement the result
- If the intended data unit is intact, total value found by adding the data segments and the checksum field should be zero
- If the result is not zero, the packet contains an error & the receiver rejects it

### **Checksum Figure**



- **Performance of Checksum**
  - Detects all errors involving an odd number of bits
  - Detects most errors involving an even number of bits
  - One pattern remains elusive

### Examples

#### Example 9.7

- Suppose a block of 16 bits need to be sent: 10101001 00111001  
10101001  
00111001  
-----  
11100010 Sum  
00011101 Checksum
- Sent pattern:  
10101001 00111001 00011101  
**checksum**

#### Example 9.8

- Examples of no error and a burst error

Segment 1	10101001	Segment1	10101111
Segment 2	00111001	Segment2	11111001
Checksum	00011101	Checksum	00011101
-----			
Sum	11111111	Sum	11000110
<b>Complement</b>	<b>00000000</b>	<b>Complement</b>	<b>00111001</b>

- **Error is invisible if a bit inversion is balanced by an opposite bit inversion in the corresponding digit of another segment**

Segment1      10111101

Segment2	00101001
Checksum	00011001
-----	
Sum	11111111

→ The error is undetected

## ERROR CORRECTION

- Mechanisms that we have studied all detect errors but do not correct them
- Error correction can be done in two ways:
  - Receiver can ask Sender for Re- TX
  - Receiver can use an error-detecting code, which automatically correct certain errors
- Error correcting code are more sophisticated than error detecting codes
- They require more redundancy bits
- The number of bits required to correct multiple –bit or burst error is so high that in most cases it is inefficient
- Error correction is limited to 1, 2 or 3 bit

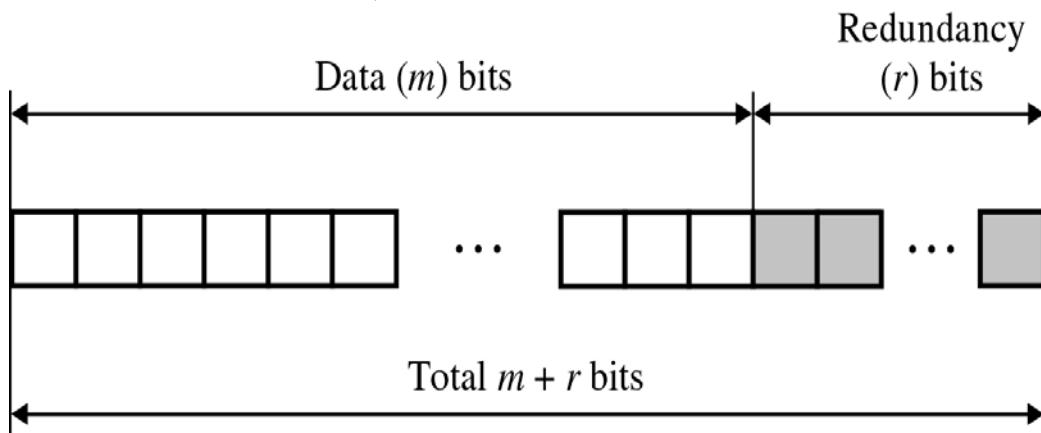
### • Single-bit Error Correction

- Simplest case of error correction
  - Error correction requires more redundancy bits than error detection
  - One additional bit can detect single-bit errors
    - Parity bit in VRC
    - One bit for two states: error or no error
  - To correct the error, more bits are required
    - Error correction locates the invalid bit or bits
    - 8 states for 7-bit data: no error, error in bit 1, and so on
    - Looks like three bits of redundancy is adequate
    - What if an error occurs in the redundancy bits?

## Hamming Code

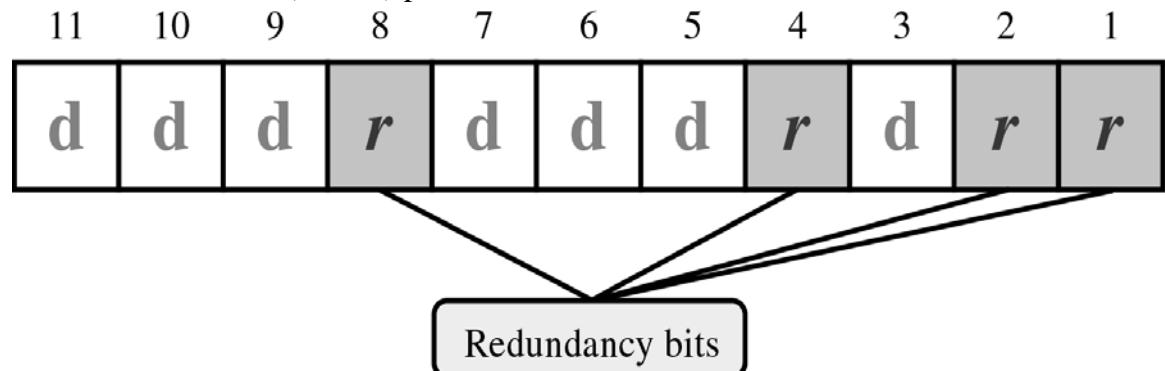
### Redundancy Bits (r)

- r must be able to indicate at least  $m+r+1$  states
- $m+r+1$  states must be discoverable by r bits
- Therefore,  $2^r \geq m+r+1$
- If  $m=7$ ,  $r=4$  as  $2^4 \geq 7+4+1$



### **Hamming Code**

- Each r bit is the VRC bit for one combination of data bits
- r1(r2) bit is calculated using all bit positions whose binary representation includes a 1 in the first(second) position, and so on



### **Summary**

- Checksum
- Single-Bit Error Correction
- Hamming Code

### **Reading Sections**

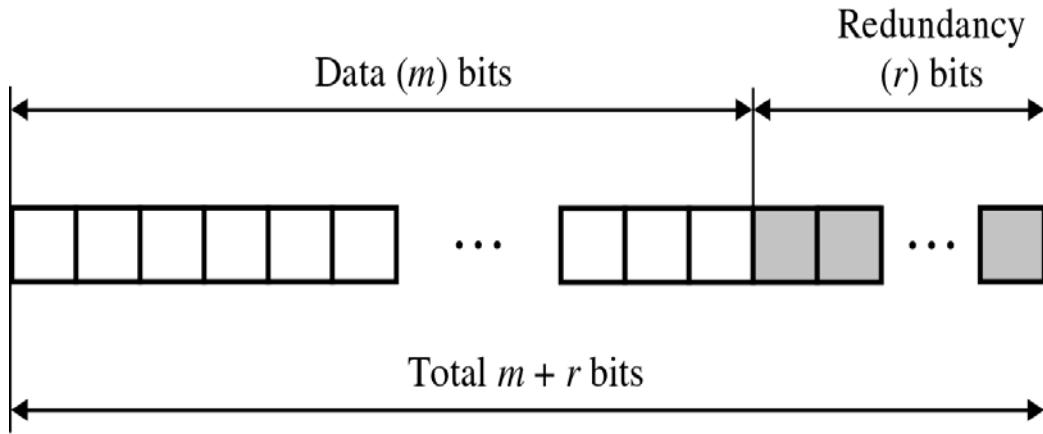
- Section 9.6, 9.7, “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

## **LECTURE #36**

### **Hamming Code**

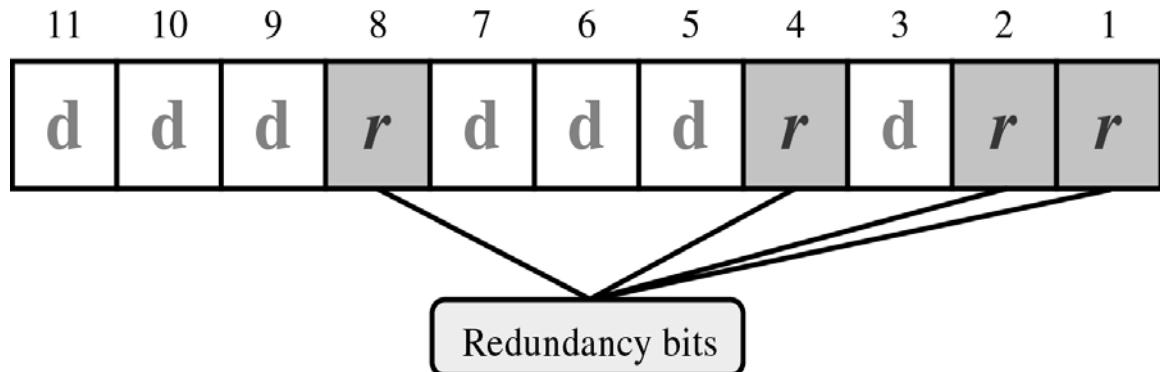
#### **Redundancy Bits (r)**

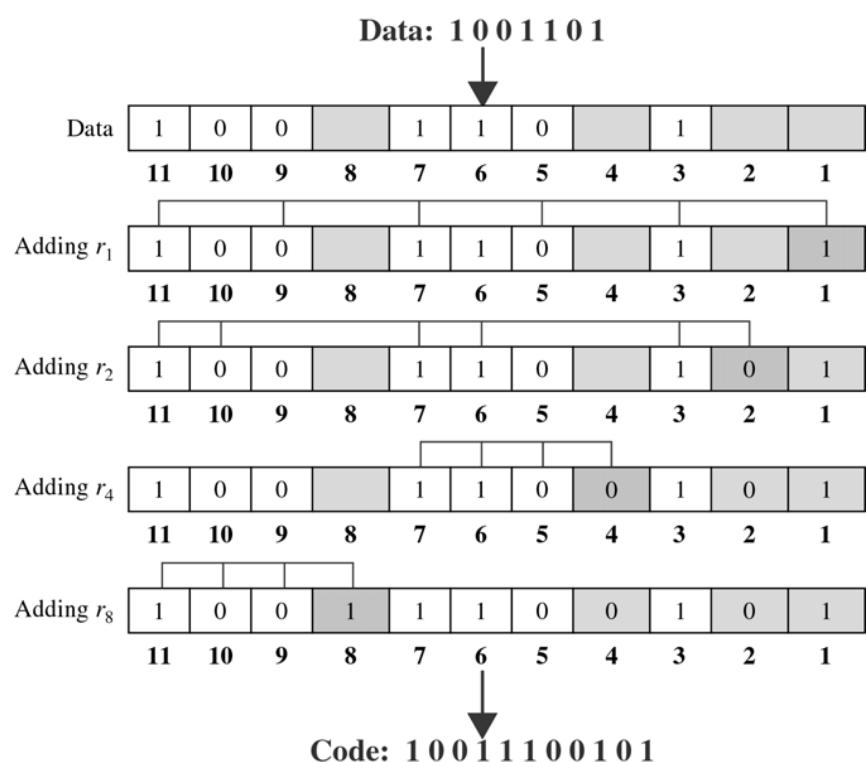
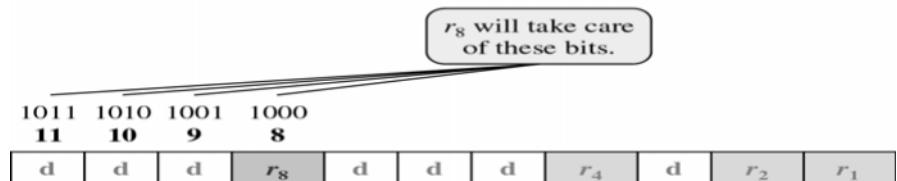
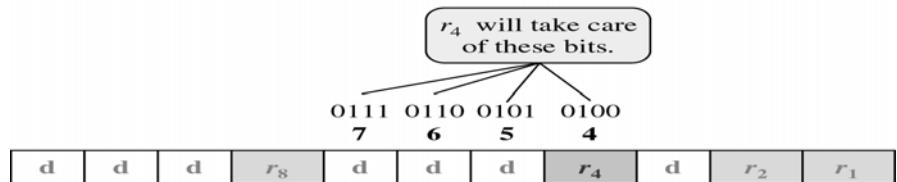
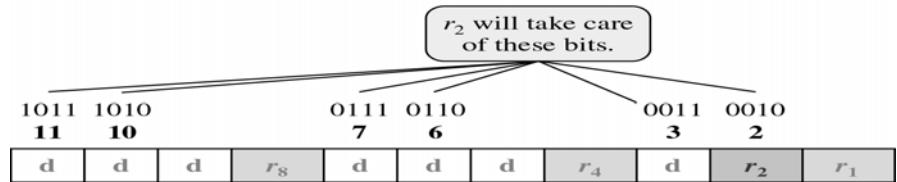
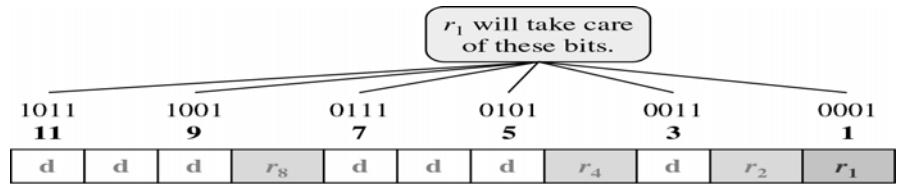
- r must be able to indicate at least  $m+r+1$  states
- $m+r+1$  states must be discoverable by r bits
- Therefore,  $2^r \geq m+r+1$
- If  $m=7$ ,  $r=4$  as  $2^4 \geq 7+4+1$

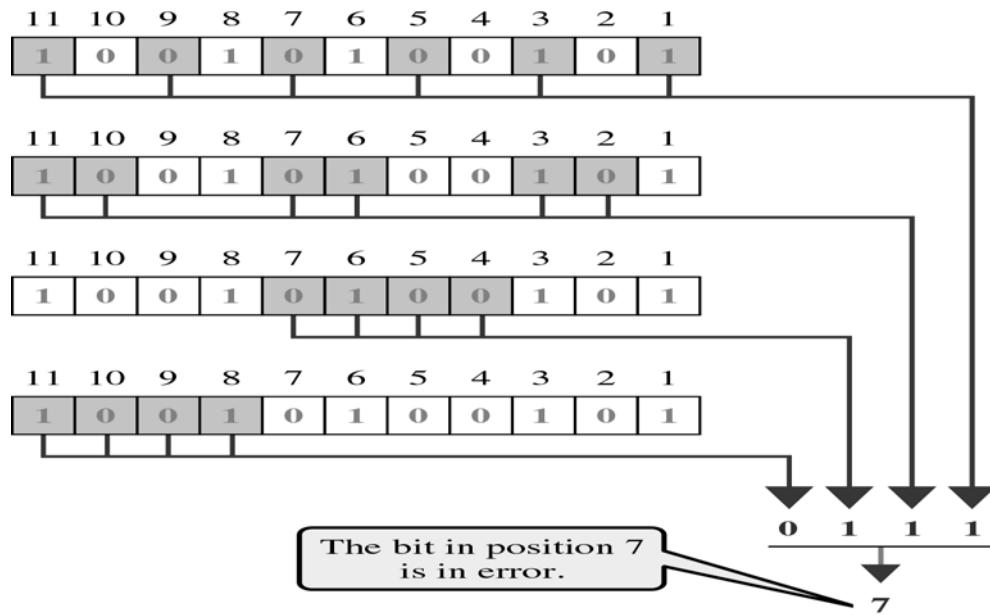


#### **Hamming Code**

- Each r bit is the VRC bit for one combination of data bits
- r1(r2) bit is calculated using all bit positions whose binary representation includes a 1 in the first(second) position, and so on





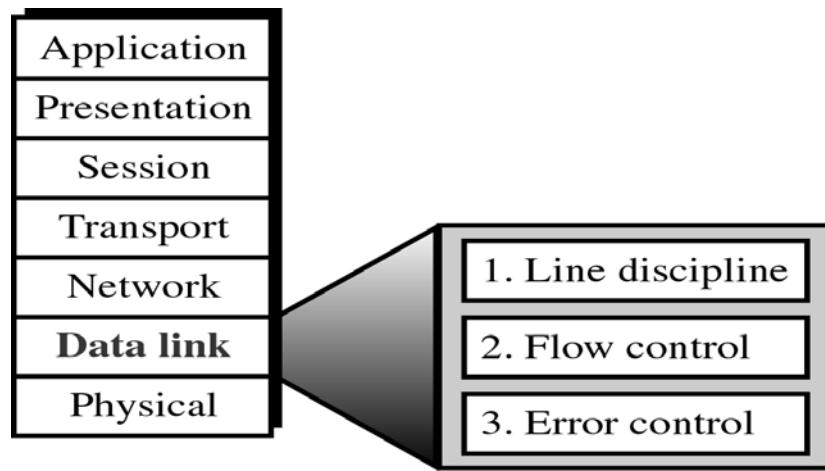


## Data Link Layer

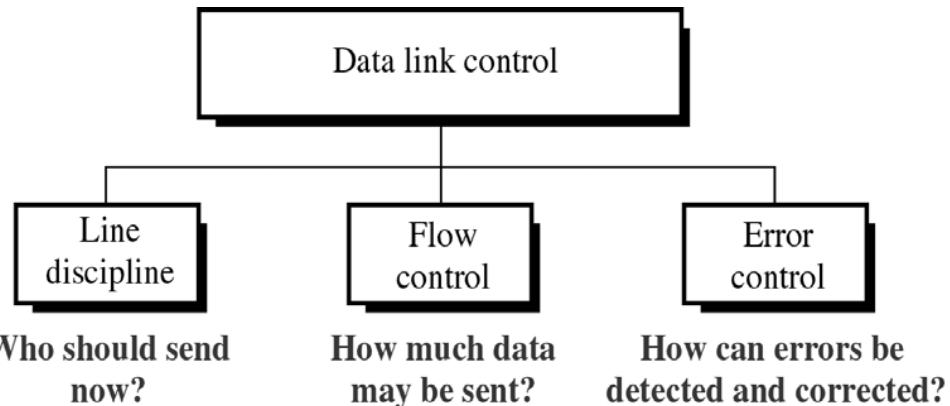
### Introduction

- Unless accurately received by a 2<sup>nd</sup> device, a signal TX over a wire is a waste of electricity
- With TX alone, we can put a signal to the line, but we have
  - no way of controlling which of several devices attached to that line will receive it
  - no way of knowing if the intended receiver is ready and able to receive it
  - No way of keeping a second device from TX at the same time
- Communication requires at least 2 devices working together:
  - Sender
  - Receiver
- Even such a basic arrangement requires great deal of coordination
- For Example, in **Half Duplex TX**, it is essential that only one device TX at a time
- If both device TX, the signals will collide leaving nothing on the line but Noise
- The coordination of half-duplex TX is a part of a procedure called Line Discipline, which is one of the functions included in the second layer of OSI Model, the data link layer
- In addition to Line Discipline, the most important functions in the data link layer are Flow Control and Error Control
- Collectively these functions are called Data Link Control

- Data Link Layer & its Functions



- Data Link Control



Y **Line Discipline:**

- Coordinates the link systems, which device can send and when it can send?

Y **Flow Control:**

- The amount of data that can be sent before the receiving acknowledgement
- It also provides the receiver's acknowledgement for frames received intact and so is linked to error control

Y **Error Control:**

- Means Error detection and Correction
- It allows the receiver to inform the sender of any frames lost or damaged in TX and coordinates Retransmission of those frames by the sender

Y **Line Discipline**

- How efficient the system is, no device in it should be allowed to transmit until that device has the evidence that the intended receiver is:
  - able to receive
  - is prepared to accept the TX
- What if the Rx device does not expect a transmission or is busy

- With no way of determining the status of the intended receiver, the transmitting device may waste its time sending data to a non-functioning receiver
- The Line Discipline functions of the data link layer oversee:
  - the establishment of links and
  - The right of a particular device to transmit at a given time

- **Ways to do Line Discipline**

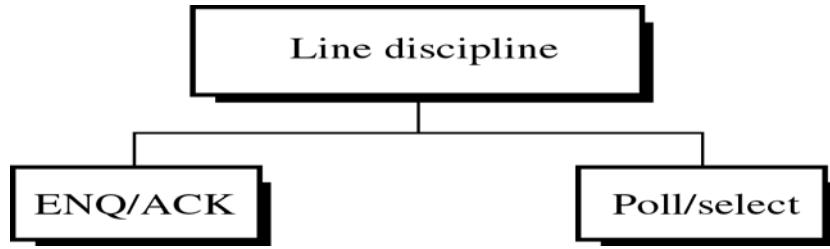
Line Discipline is done in 2 ways:

- Enquiry / Acknowledgement (ENQ/ACK)**

—Used in Peer-to-Peer Communication

- Poll / Select**

—Primary-Secondary communication



## Summary

- Hamming Code
- Data Link Control
- Line Discipline
  - ENQ/ACK
  - POLL/SELECT
- Flow Control

## Reading Sections

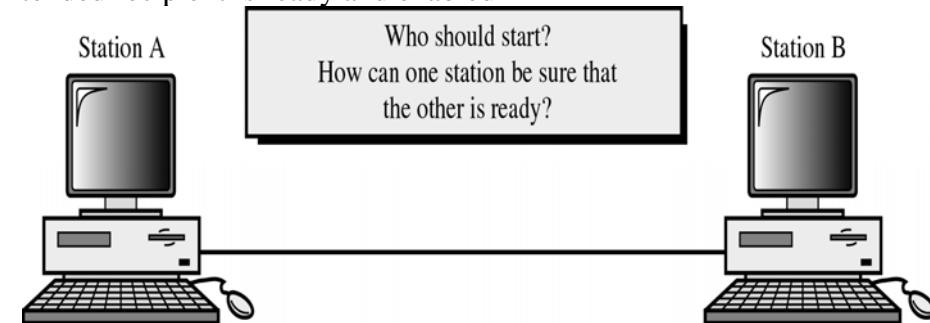
Section 9.7, 10.1 “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #37**

## **Line Discipline**

### **Enquiry / Acknowledgement (ENQ/ACK)**

- Used primarily in systems where there is no question of wrong receiver getting the transmission
- In other words when there is a Dedicated Link b/w the two devices so that the only device that can receive data is the intended one
- Enquiry / Acknowledgement (ENQ/ACK)
- ENQ/ACK coordinates who may start a transmission and whether or not the intended recipient is ready and enabled

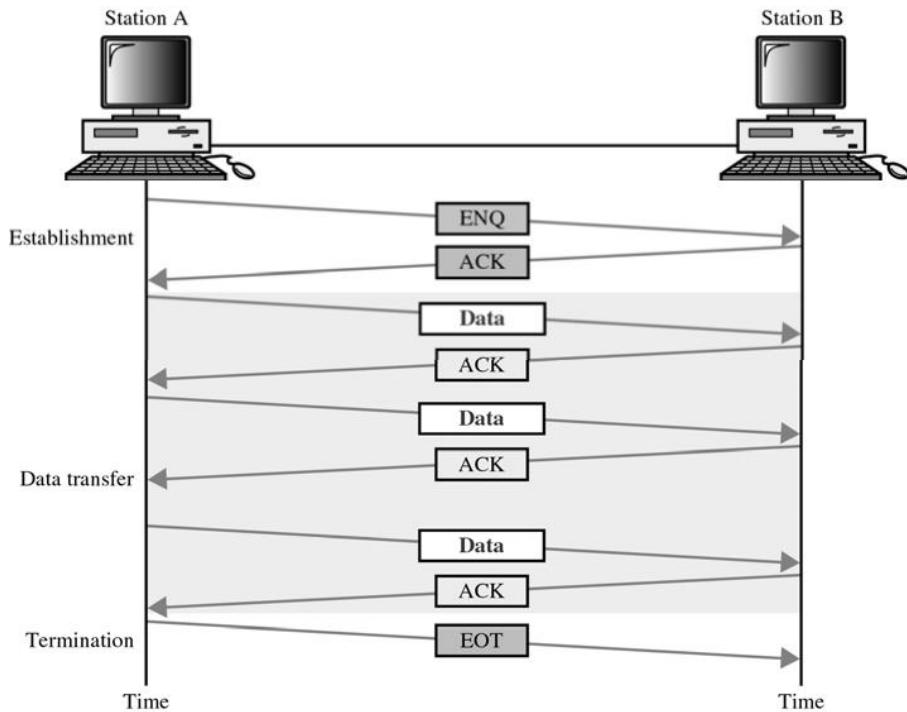


- Using ENQ/ACK, a session can be initiated by either station on a link as long as both are of equal rank- a printer for example cannot initiate communication with a CPU
- Enquiry / Acknowledgement (ENQ/ACK)
- In both half duplex and full duplex TX, the initiating device establishes the session
- In half-duplex, the initiator then sends its data while the responder waits
- The respondent may take over the link when the initiator is finished or has requested a response
- In full duplex, both the devices can TX simultaneously once the link has been established

### **How It Works?**

- The receiver must answer either with an acknowledgement (ACK) frame if it is ready to receive or with a negative acknowledgement (NAK), if it is not
- By requiring a response, even if the answer is negative, the initiator knows that his enquiry was in fact received even if the receiver is currently unable to receive
- If neither an ACK or a NAK is received within a specified time limit, the initiator assumes that an ENQ frame was lost in transit, it disconnects and sends a replacement
- An initiating system ordinarily makes 3 such attempts before giving up
- If the response to the ENQ is negative for 3 attempts, the initiator disconnects and begins the process again at another time

• If the response is positive the initiator is free to send its data



- The initiator first transmits a frame called an enquiry (ENQ) asking if the receiver is available to receive data
- Once all of its data have been transmitted, the sending system finishes with an End of Transmission (EOT) frame

### - Poll / Select

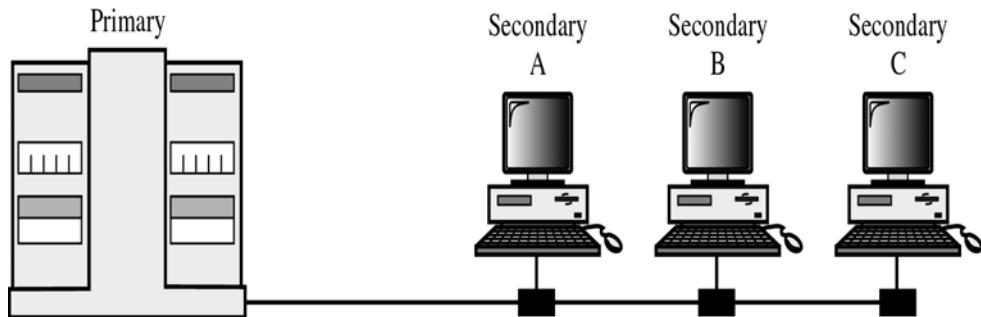
Primary-Secondary communication

- This method works with topologies where one device is designed as a Primary station and the other devices are Secondary stations
- Multipoint systems must coordinate several nodes, not just two
- The questions are not only Are you Ready? But also Which of the device has the right to use the channel
- Whenever a multipoint link consists of a primary device and multiple secondary devices using a single TX line , all exchanges must be made through the primary device even when the ultimate destination is a secondary device
- The primary device controls the link and the secondary device follow its instruction
- It is up to the primary to determine which device is allowed to use the channel at a given time
- The primary therefore is always the initiator of the session

• If the primary wants to receive data, it asks the secondary if they have anything to send, This is called **POLLING**

• If the primary wants to send data, it tells the target secondary to get ready to receive, This function is called **SELECTING**

Who has the right to the channel?

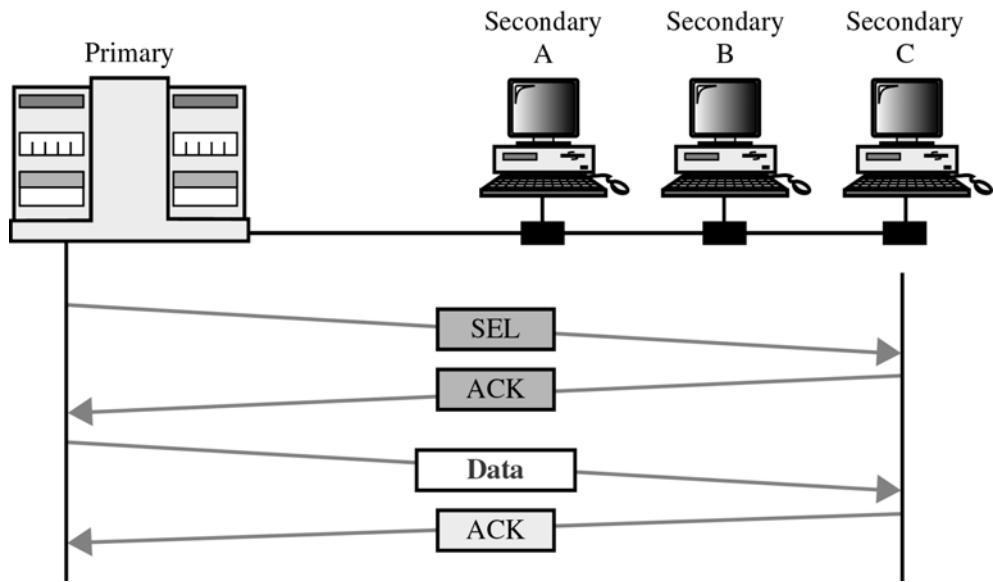


## ADDRESSING

- For point-to-point configuration, there is no need for addressing
- Any TX put onto the link by one device can be intended only for the other
- For the primary device in a multipoint topology to be able to identify and communicate with a specific secondary device, there must be some addressing convention
- For this reason, every device on the link has an address that can be used for identification
- In any transmission, this address will appear in a specified portion of each frame, called the Address Field or Header depending upon the protocol
- If the TX comes from a secondary device, the address indicates the originator of the data

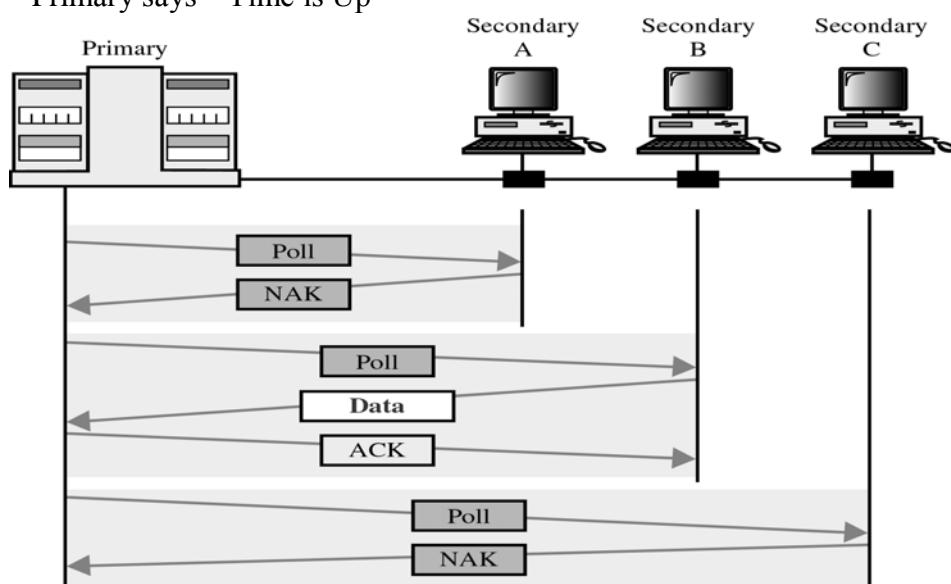
## Y SELECT

- The select mode is used whenever the primary device has something to send
- Primary control the link and if primary is not sending or receiving data, it knows that the link is available
- If it has something to send, it sends it
- As a frame makes its way to the intended device, each of the other devices check the address field
- Only when the device recognizes its own address, does it open the frame and read the data
- In case of a SEL frame, the enclosed data consists of an alert that data is forthcoming
- What it does not know, however is if the target device is ready to receive (ON)
- So the primary must alert the secondary to the upcoming TX and wait for an acknowledgement of the secondary ready status
- Primary send a SEL frame, one field of which includes the address of the intended RX
- If the secondary is awake and running, it returns an ACK frame to the primary
- The primary then sends one or more data frames, each addressed to the intended secondary



### Y POLL

- Used by the primary device to receive transmissions from the secondary devices
- The secondaries are not allowed to TX data until asked
- By keeping control with the primary, the multipoint system guarantees that only one TX can occur at a time
- When the primary is ready to receive, data , it must ask (POLL) each device in turn if it has anything to send
- When the first secondary is approached, it responds either with a NAK frame if it has nothing to send or with data if it does
- If the response is negative, primary then polls the next secondary
- When the primary has received data, it acknowledges by sending an ACK
  - Two possibilities for termination:
  - A secondary sends all its data and finishes with EOT frame
  - Primary says “ Time is Up”



- **Flow Control -Definition**
  - 2<sup>nd</sup> aspect of data link control is Flow Control
  - In most protocols, flow control is a set of procedures that tells the sender how much data it can transmit before it must wait for an ACK from the receiver
  - The flow of data must not be allowed to overwhelm the receiver

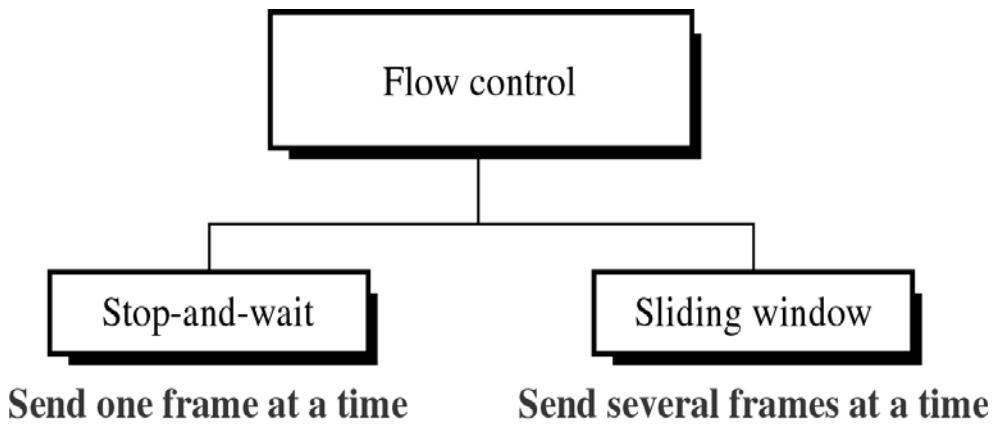
### **Summary**

- Line Discipline
  - ENQ/ACK
  - POLL/SELECT
- Flow Control
  - Stop-and-Wait
  - Sliding Window

### **Reading Sections**

Section 10.1,10.2 “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

## **LECTURE #38**



### **Flow Control- Explanation**

- 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 TX device send fewer frames or stop temporarily

- **Flow Control-Buffer**

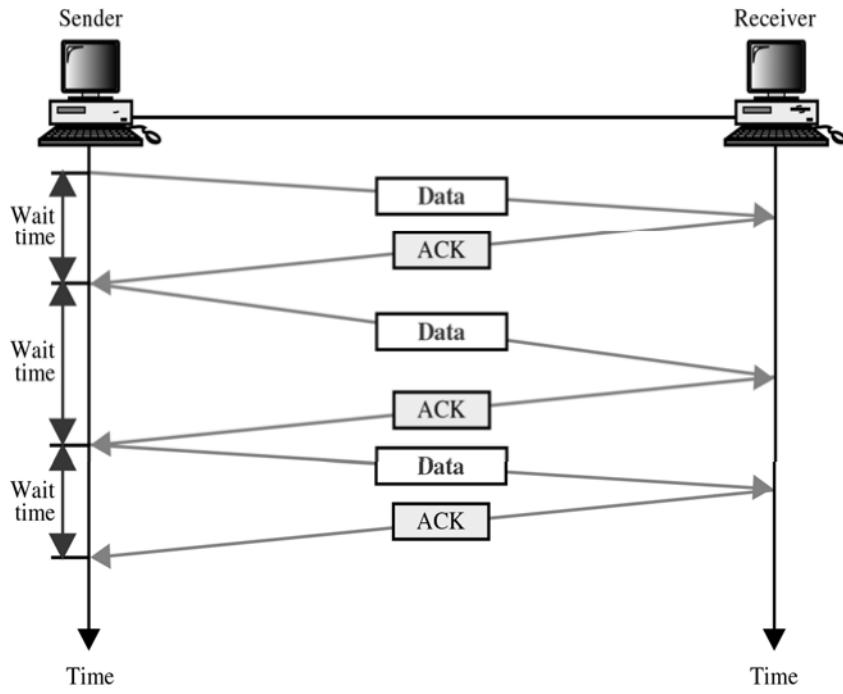
- Incoming data must be processed and checked before it can be used
- The rate of such processing is often slower than the rate of TX
- So, each receiving device has a block of memory called BUFFER, reserved for storing incoming data until it is processed
- If the buffer begins to fill up, the receiver must be able to tell the sender to halt the TX until it is once again able to receive

### **Methods for Flow Control**

- Two methods have been developed to control the flow of data across communication links :
  - Y Stop and Wait
  - Y Sliding Window

#### **Y Stop and Wait**

In this method, the sender waits for an ACK after every frame it sends



- Only when an ACK has been received, is the next frame sent
- This process of alternately sending and waiting repeats until the sender transmits an EOT frame

Example: Officer giving dictation to the Typist, He says a word, typist says OK, he says the next word, typist says OK and so on

- **Advantages of Stop and Wait**

- \*SIMPLICITY**

—Each frame is checked and acknowledged before the next frame is sent

- **Disadvantages of Stop and Wait**

- \*INEFFICIENT ( Slow)**

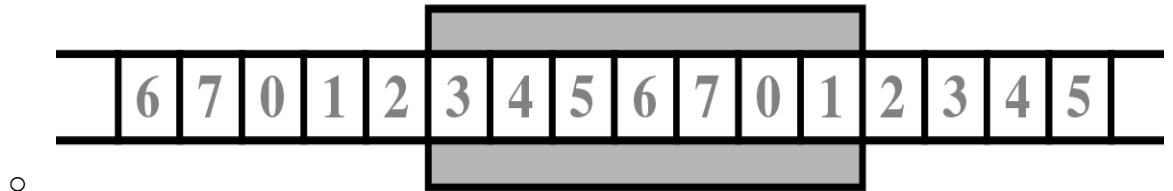
- Each frame must travel all the way to the receiver and an ACK must travel all the way back before the next frame can be sent
- If the distance b/w devices is long, the time spent waiting for ACKs between each frame can be significantly long

- **Sliding Window**

- In this method, sender can transmit several frames before needing an ACK
- Frames can be sent one right after another meaning link can carry several frames at once and its capacity can be used efficiently
- The receiver uses a single ACK to confirm the receipt of multiple data frames
- Sliding Window refers to imaginary boxes at both the sender and the receiver
- This window can hold frames at either end and provides the upper limit on the number of frames that can be sent before requiring an ACK
- Frames may be ACK at any point w/o waiting for the window to fill up and may be TX as long as the window is not yet Full
- To keep track of which frames have been transmitted and which received, sliding window introduces an identification scheme based on the size of the window
- The frames are numbered modulo-n means from 0 to n-1
- If n=8, frames are numbered 0,1,2,3,4,5,6,7,0,1,2,3,4,5,6,7,0,....

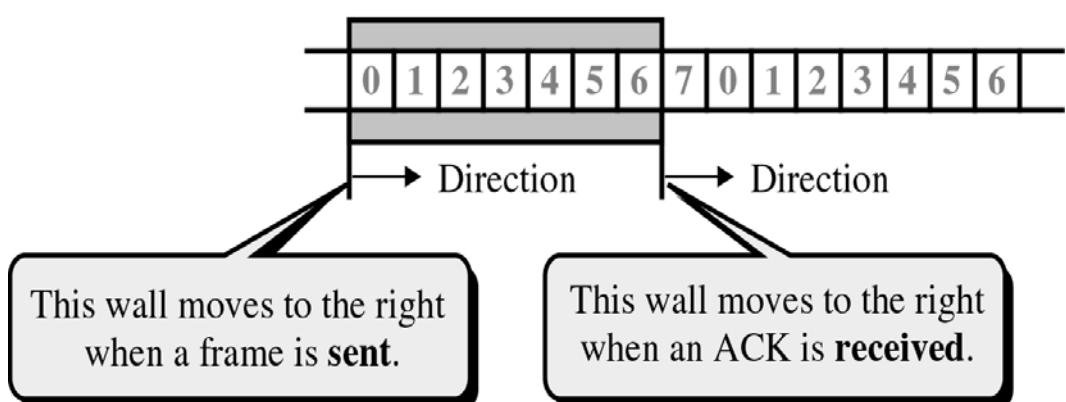
- When the receiver sends the ACK, it includes the number of the next frame it expects to receive

## Window



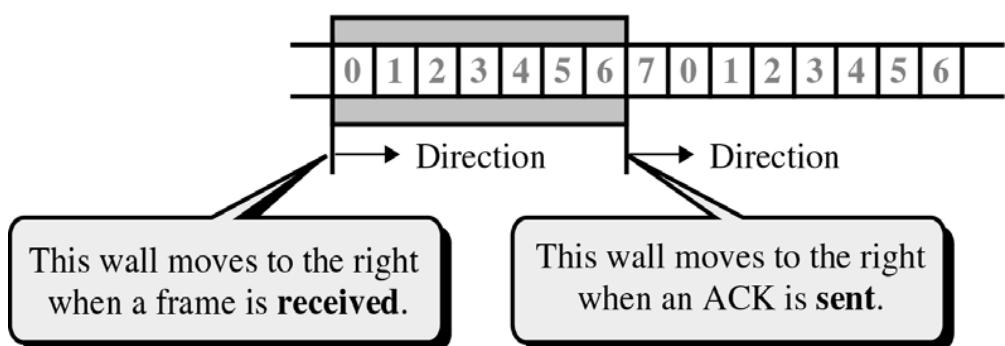
- For example, to ACK the receipt of a string of frames ending in frame 4, the receiver sends an ACK with number 5
- The window can hold  $n-1$  frames at either end, therefore a max of  $n-1$  frames may be sent before an ACK is required

### Sender window



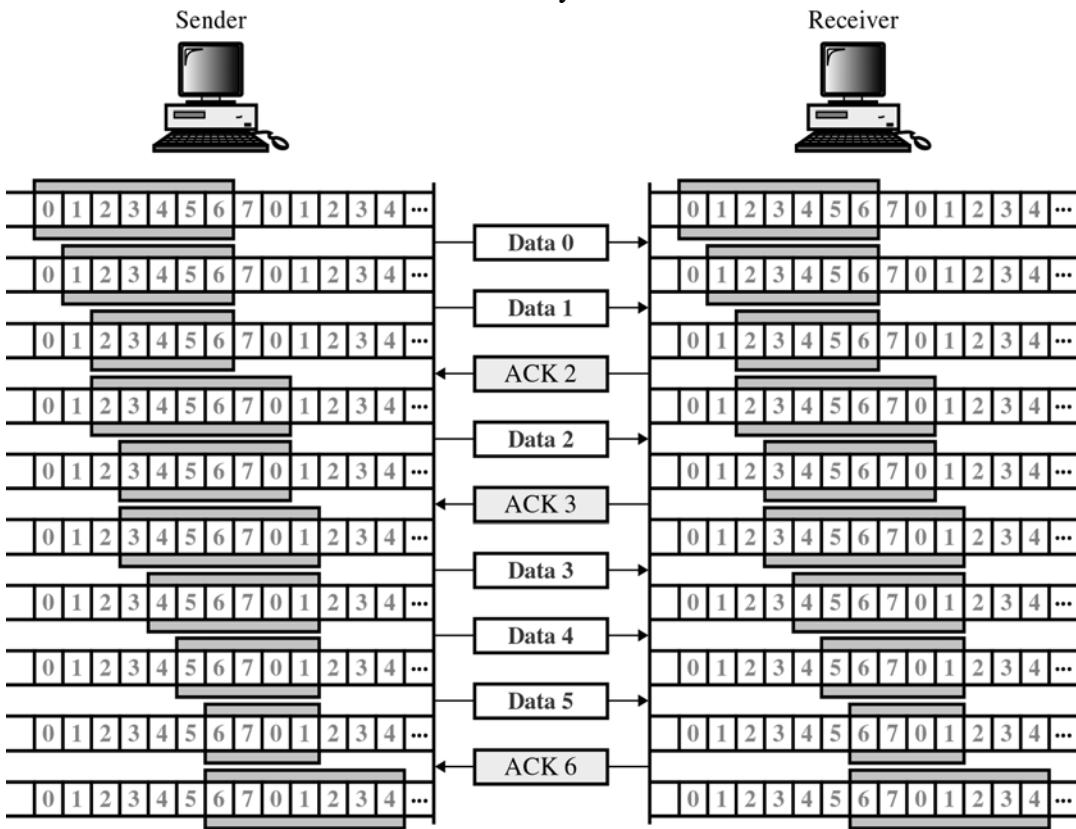
- At the beginning of a TX, sender's window contains  $n-1$  frames
- As frames are sent out, the left boundary of window moves inward, shrinking the size of the window
- When an ACK is received, the window expands to allow in a number of new frames equal to the number of frames acknowledged by that ACK

### Receiver window



- At the beginning of TX, the receiver window contains  $n-1$  spaces for frames
- As new frames come in, the size of the receiver window shrinks

- The receiver window therefore does not show the frames that are received but the frames that may still be received before an ACK is sent



- ERROR CONTROL**

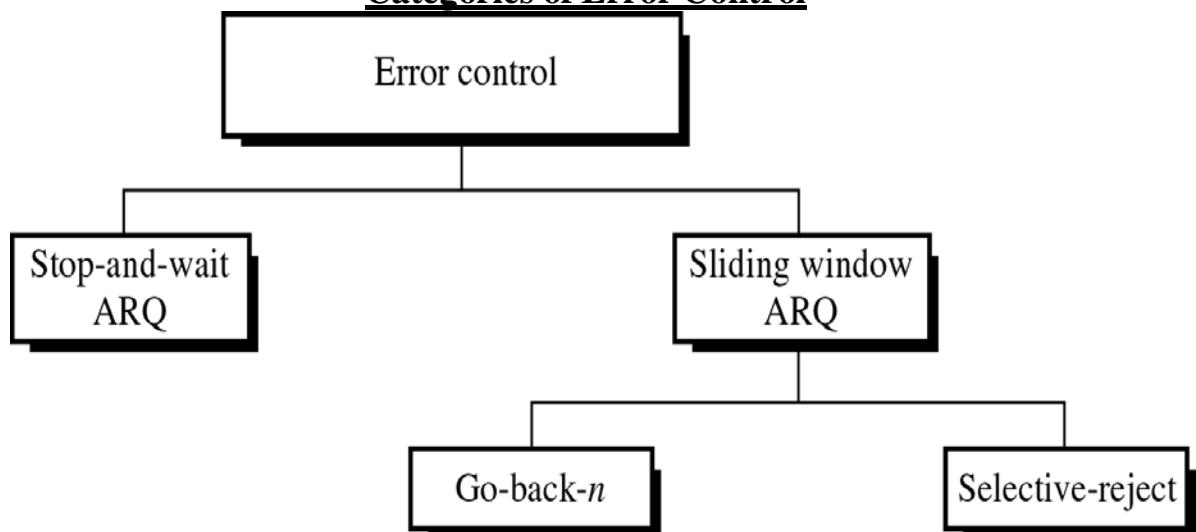
Refers primarily to error detection and correction

- AUTOMATIC REPEAT REQUEST (ARQ)**

Error control in data link layer is implemented simply:

- Anytime an error is detected in an exchange, a negative acknowledgement (NAK) is returned and the specified frames are retransmitted

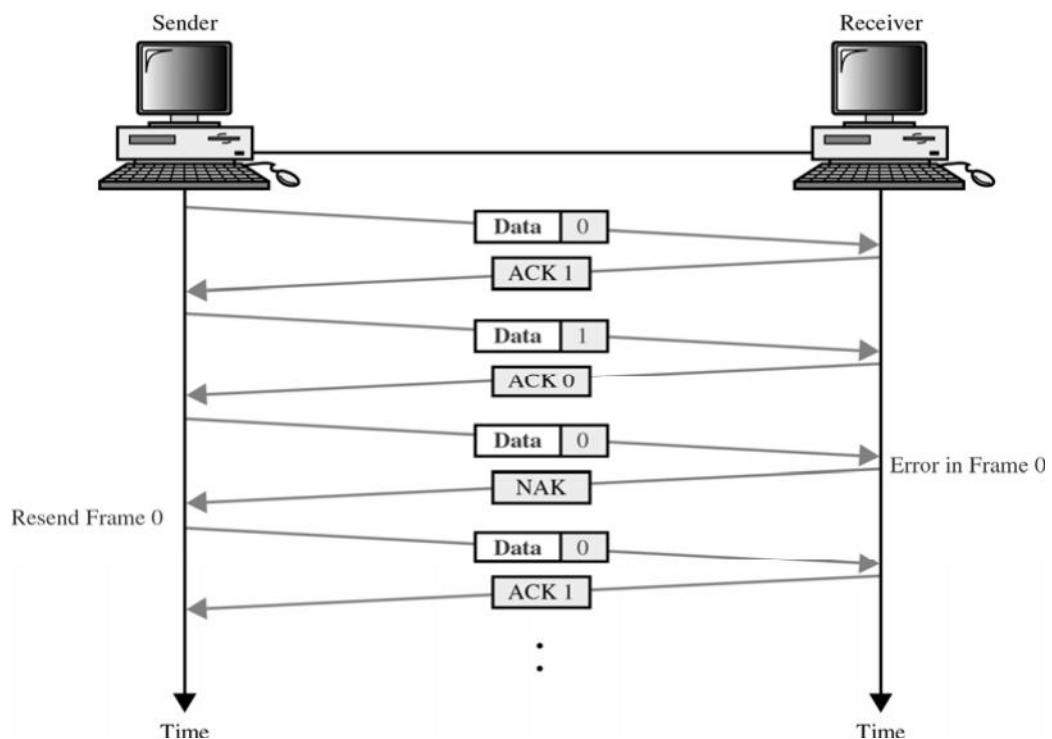
### Categories of Error Control



### Stop and Wait ARQ

- It is a form of stop-and-wait flow control extended to include retransmission of data in case of Lost or Damaged frames
- For retransmission to work, 4 features are added to the basic flow control mechanism
  - Sending device keeps a copy of the last frame transmitted until it receives the ACK for that frame
  - Both data and ACK frames are numbered 0 and 1 alternately
  - A data 0 frame is acknowledged by a ACK 1 frame indicating that the receiver has received data 0 and is now expecting data 1
  - Sending device keeps a copy of the last frame transmitted until it receives the ACK for that frame
  - Both data and ACK frames are numbered 0 and 1 alternately
    - A data 0 frame is acknowledged by a ACK 1 frame indicating that the receiver has received data 0 and is now expecting data 1
      - If an error is discovered in a data frame an NAK frame is returned
      - NAK frames which are not sent tell the sender to retransmit the last frame
      - The sending device is equipped with Timer. If an expected ACK is not received within an allotted time period, the sender assumes that the last frame sent is lost and resends the frame

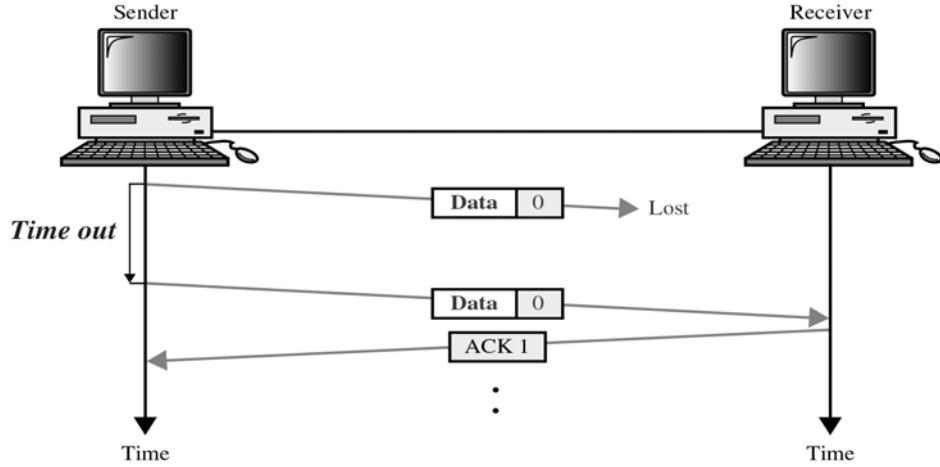
- **Damaged Frame**



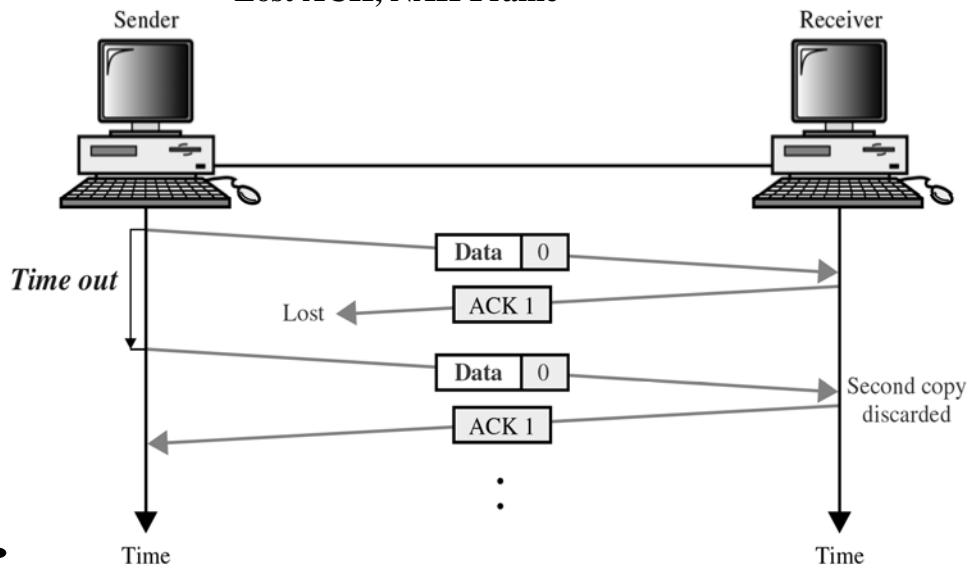
- **Lost Frame**

Any of the 3 frame types can be lost in transit:

- Y Lost Data Frame
- Y Lost ACK Frame
- Y Lost NAK Frame



- **Lost ACK, NAK Frame**



### Summary

- Flow Control
  - Stop-and-Wait-
  - Sliding Window
- Error Control

### Reading Sections

Section 10.2, 10.3 “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #39**

## **SLIDING WINDOW ARQ**

Among several popular mechanisms for error control two protocols are important:

- Y Go-back-n ARQ
- Y Selective Reject ARQ

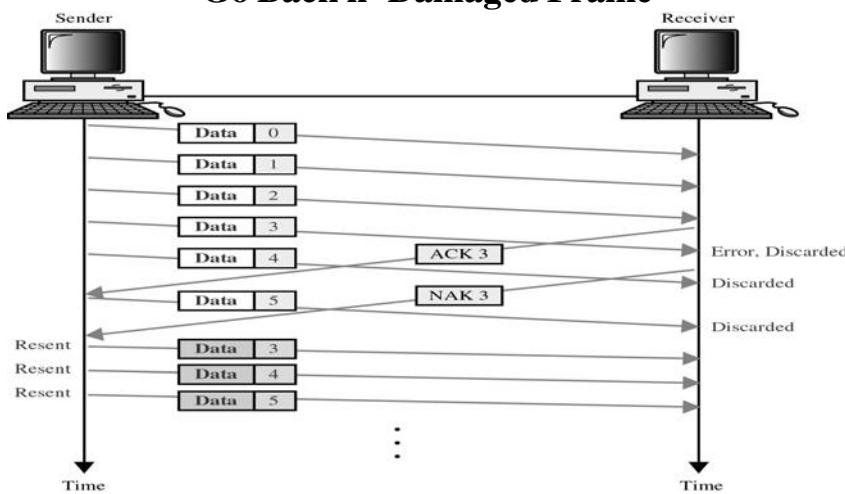
Three features are added to sliding window flow control to allow for the retransmission of the lost or the damaged frames:

- o The sending device keeps copies of the transmitted frames until all of them have been acknowledged
- o In addition to ACK frames, receiver also has the option of NAK frames, if data has been received damaged
- o Because sliding window is a continuous TX mechanism, both ACK and NAK frames must be numbered for identification
  - ACK frames carry the number of the next frame expected
    - ACK 5 tells sender that all frames up to frame 5 are received
    - NAK frames carry the number of the damaged frame itself
    - If data frames 4 and 5 are damaged, NAK 4 and NAK 5 must be sent
- o Like sender in stop-and wait ARQ, the sliding window ARQ is also equipped with a timer in the sender to deal with lost ACKs

### **• Go Back n ARQ**

In Go Back n ARQ, if one frame is lost or damaged, all frames sent since last frame acknowledged are retransmitted

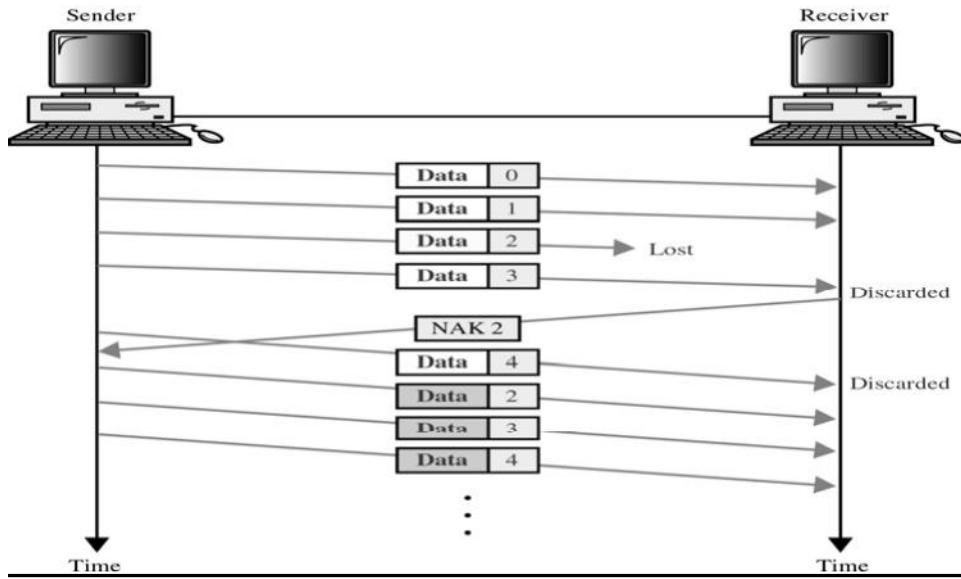
#### **Go Back n- Damaged Frame**



### **Y Go Back n- Lost Data Frame**

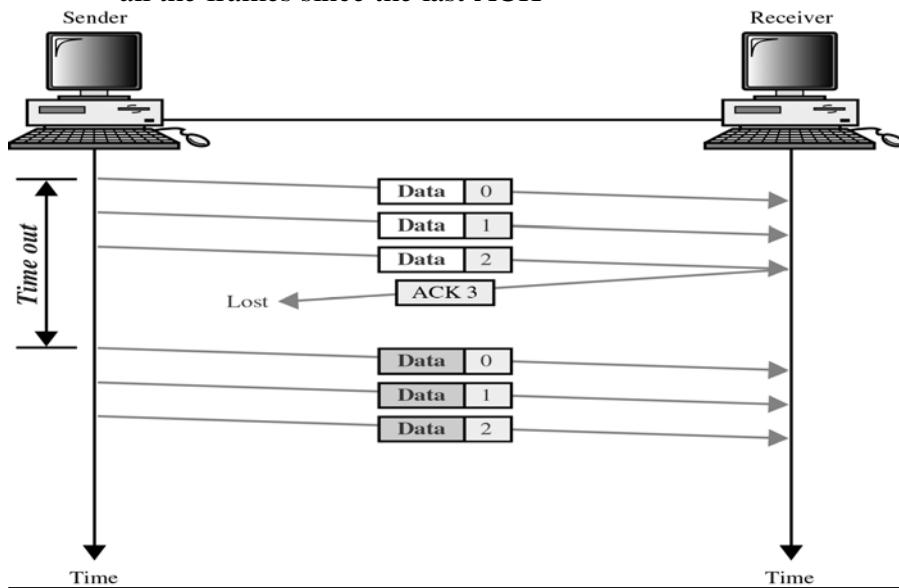
- o Sliding window requires that data frames be transmitted sequentially

- If one or more frames are so noise corrupted that they become lost in transit, the next frame to arrive at the receiver will be out of sequence



#### Y Go Back n- Lost ACK

- When the window capacity is reached and all frames allowed have been sent, the sender starts a Timer
- If an ACK is not received before that Timer expires, sender retransmits all the frames since the last ACK



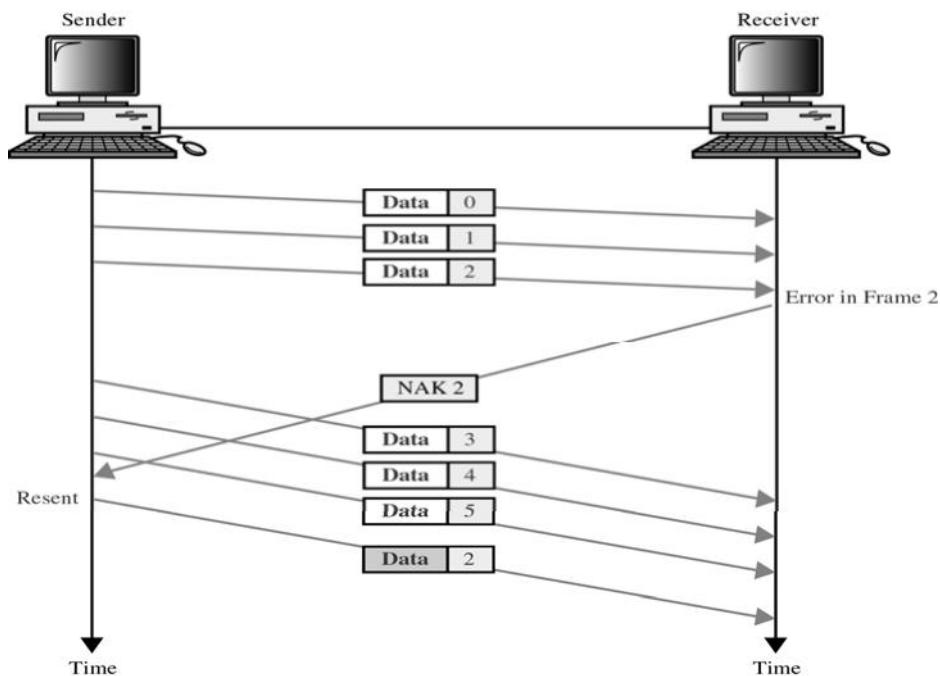
#### ● Selective Reject ARQ

- In selective-reject ARQ, only the specific damaged or lost frame is retransmitted
- If a frame is corrupted in transit, a NAK is returned and the frame is resent out of sequence

- The Rx device must be able to sort frames and insert the retransmitted frame into the proper place

The selective reject ARQ differs from Go Back n in the following ways:

- The Rx device must contain sorting logic to enable it to reorder frames received out of sequence
- Sending device must contain a searching mechanism that allows it to find and select only the requested frame for retransmission
- Selective Reject ARQ
- A buffer in the receiver must keep all previously received frames on hold until all retransmissions have been stored
- To avoid selectivity, ACK number, like NAK numbers must refer to frame received instead of next expected frame
- A smaller window size is required because of this added complexity



#### Y Selective Reject ARQ-Lost Frame

Lost ACK/NAK are treated exactly in the same way as by Go Back n

#### **Selective Reject ARQ vs Go Back n**

- Although retransmitting only specific damaged or lost frames may seem more efficient than resending all the frames

- Because of the complexity of sorting and storage required by the receiver and extra logic needed by sender to select specific frames for retransmission, selective reject ARQ is EXPENSIVE and not often used
- Selective reject gives better performance but in practice it is usually discarded in favor of go-back-n for simplicity of implementation

## **Protocols**

**Protocol:** Set of rules or conventions for executing a particular task

**Protocol in Data Comm.:** Set of rules or specifications used to implement one or more layers of the OSI Model

**Example:** EIA 232-D interface is a protocol used at the physical layer in the OSI Model  
Protocols

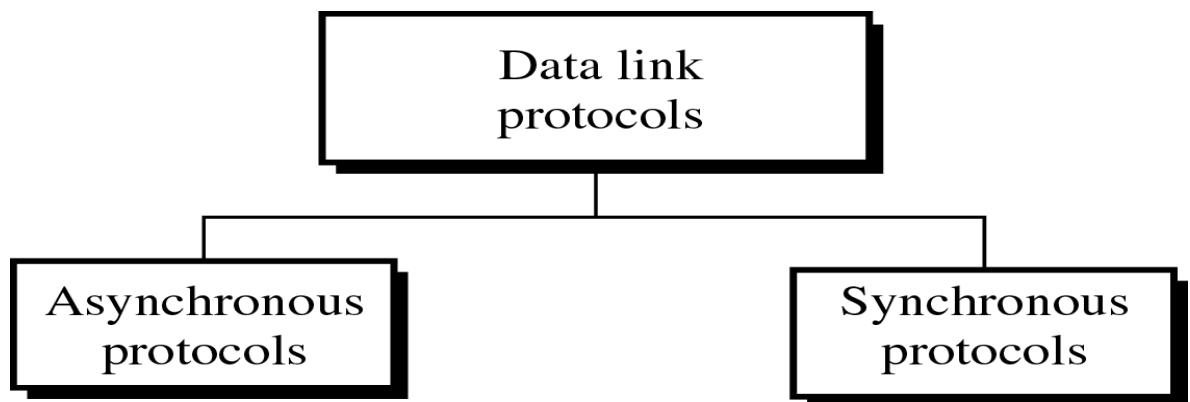
**Data Link Protocols:** Set of specifications used to implement the data link layer

Data link protocols contain rules for:

- Line Discipline
- Flow Control
- Error Control

## **Categories of Data Link Protocols**

Data link protocols can be divided into two sub-groups:



**Asynchronous Protocols:** Treat each character in a Bit stream independently

**Synchronous Protocols:** Take the whole bit stream and chop it into characters of equal Size

Y **Asynchronous Protocols**  
Employed mainly in Modems

**Inherent Slowness is a disadvantage:** Requires addition of start and stop bits and extended spaces b/w frames, so these are mainly replaced with High-speed synchronous mechanisms

### **Not Complex and Inexpensive to Implement**

Transmission does not require timing coordination; Timing is done by using extra bits

- **Different Asynchronous Protocols**

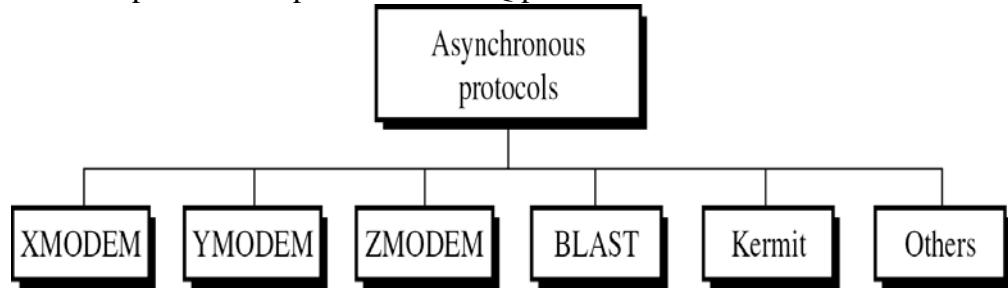
A variety of Asynchronous protocols have been developed

We will discuss some of the important ones

#### **/ XMODEM**

In 1979 Ward Christiansen designed a File transfer protocol for Telephone-line communication b/w PCs called XMODEM

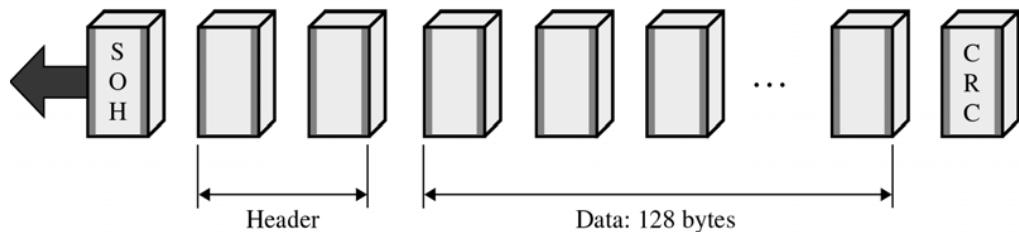
Half Duplex and Stop-and-Wait ARQ protocol



### **XMODEM**

- The first field is a One Byte start of header (SOH) field
- The second field is a two-byte Header.
  - The first header byte , the Sequence number carries the Frame number
  - The second header byte is used to check the validity of the sequence number
- The fixed data field holds 128 bytes of data
- The last field CRC checks for errors in the data field only

Each character contains start and stop bits (dark portion of the box). Characters are separated from each other by gaps.  
The header consists of two bytes: sequence number and its one's complement.



### **Transmission in XMODEM**

- TX begins with sending of a NAK frame from the receiver to the sender
- Each time, the sender sends a frame, it must wait for an ACK before sending next frame

- If NAK is received instead of ACK, the last frame is sent again
- A frame is also resent if no response arrives from a receiver after a fixed time period
- A sender can also receive cancel (CAN) to abort TX

## Control Frames in XMODEM

### Control frames from the receiver

- ACK: Acknowledgement
- NAK: Error or start of transmission
- CAN: Aborts the transmission

## Summary

- Error Control
  - Stop-and-Wait ARQ
  - Sliding Window ARQ
    - Go-back- $n$
    - Selective Reject
- Data Link Protocols

## Reading Sections

- Section 10.3, 11.1,  
“Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #40**

## **YMODEM**

YMODEM is similar to X-MODEM with only the following major differences:

- 1024-Byte data unit
- Two CANs to abort Transmission
- ITU-T CRC-16 for Error Checking
- Multiple files can be sent simultaneously

## **ZMODEM**

- Newer Protocol
- Combines features of
  - XMODEM
  - and
  - YMODEM

## **BLAST**

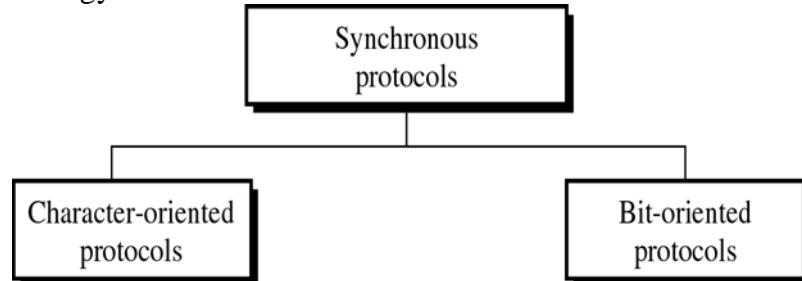
- Blocked Asynchronous Transmission
- More powerful than XMODEM
- Full Duplex
- Sliding Window Flow Control
- Allows transfer of Data and Binary Files

## **KERMIT**

- Designed at Columbia University
- Most Widely used Asynchronous Protocol
- File Transfer protocol is similar in operation to XMODEM, with sender waiting for an NAK before it starts TX
- Kermit allows the transmission of control characters as Text

## **Synchronous Protocols**

Speed of synchronous TX makes it a better choice over Asynchronous T for LAN, MAN and WAN technology



## **Classes of Synchronous Protocols**

Synchronous Protocols can be divided into two main classes:

- Y Character – Oriented Protocols
- Y Bit – Oriented Protocols

## **Character – Oriented Protocols**

- Also called **Byte- Oriented Protocol**
- These protocols interpret a transmission frame or packet as a succession of characters, each usually composed of one byte
- All control information is in the form of an existing character encoding system

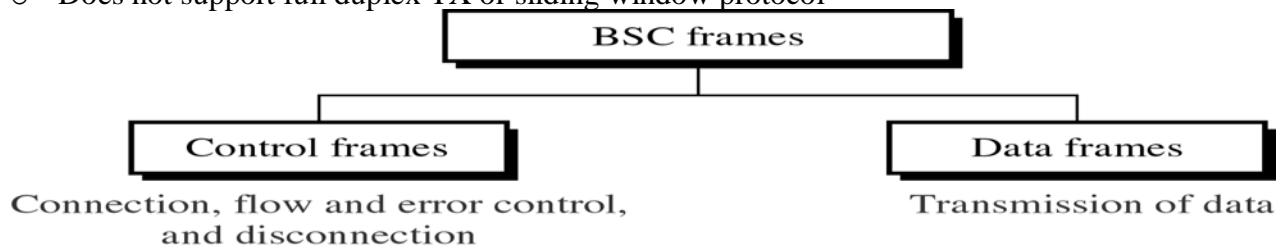
## **Bit – Oriented Protocols**

These protocols interpret a transmission frame or packet as a succession of individual bits, made meaningful by their placement in the frame  
 Control information can be one or multiple bits depending on the information embodied in the pattern

- Character –Oriented Protocols are not as efficient as bit – oriented protocols and are seldom used
- They are easy to comprehend and employ the same logic as bit-oriented protocols
- Their study will provide the basis for studying the other data link layer protocols
- In all data link protocols, control information is inserted in the data frame as separate control frames or as addition to existing data frames
- In character oriented protocols, this info is in the form of code words taken from existing character sets such as ASCII
- IBM's BSC is the best known character oriented protocol

## **Binary Synchronous Communication (BSC)**

- Developed by IBM in 1964
- Usable in both point-to-point and multiple communications
- It supports half-duplex TX using stop-and-wait ARQ flow control
- Does not support full duplex TX or sliding window protocol



### **BSC FRAMES**

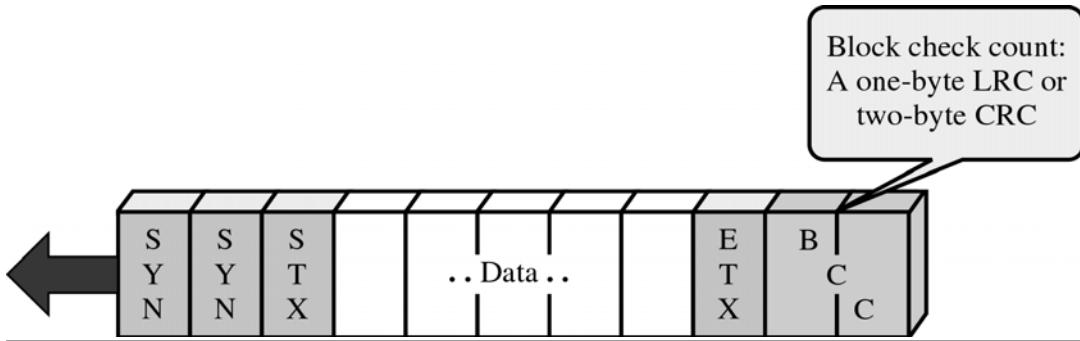
- BSC protocol divides a transmission into frames

### **CONTROL FRAMES**

- If a frame is used strictly for control purposes, it is called a Control frame
- Control frames are used to exchange information b/w communicating devices for example, to establish the connection, to control the flow, to request error correction etc

### **DATA FRAMES**

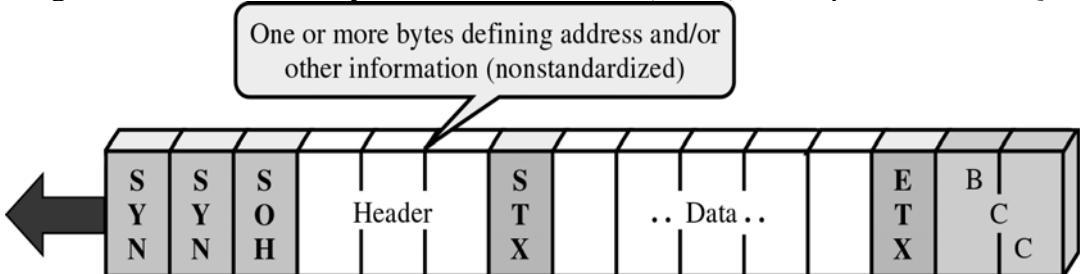
- If a frame contains part or all of the message itself, it is called a Data Frame
- Data frames are used to transmit information, but may also contain control information applicable to that information



- Figure shows the format of a simple data frame
- Arrow shows the direct of TX
- The frame begins with two or more synch. (SYN) characters
- These characters alert the receiver to the arrival of a new frame and provide a bit pattern used by the receiving device to sync itself with that of the sending device
- After the two synch characters, comes a start of text (STX) character
- This character signals to the receiver that the control information is ending and the next byte will be data
- Data or text can consist of varying number of characters
- An end of text (ETX) indicates the end of text
- Finally, the Block Check Count (BCC) are included for error correction
- A BCC field can be a one-character LRC or a two –character CRC

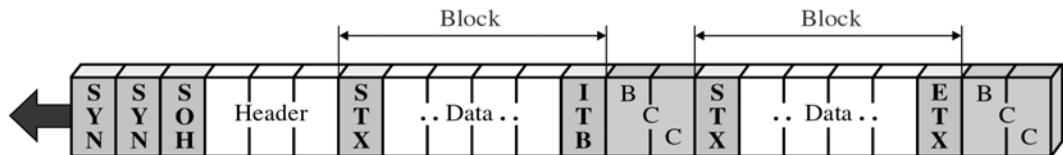
### Data Frame with Header

- A frame as simple as above is seldom used
- Usually we need to include the address of the receiving device, the address of the sending device and the identity number of the frame (0 or 1) for stop-and-wait ARQ



- All the above information is included in a special field called Header
- Header begins with start of the header (SOH) character
- The header comes after the SYNs and before the STX character
- Everything received after the SOH field but before STX character is the Header information

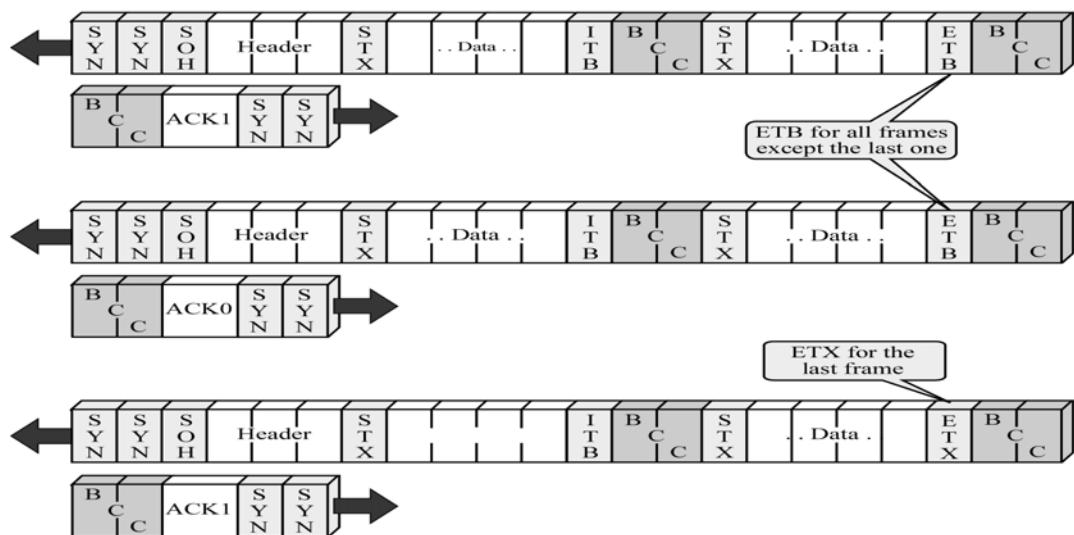
### Multiblock Frame



The probability of an error in the block of text increases with the length of the frame

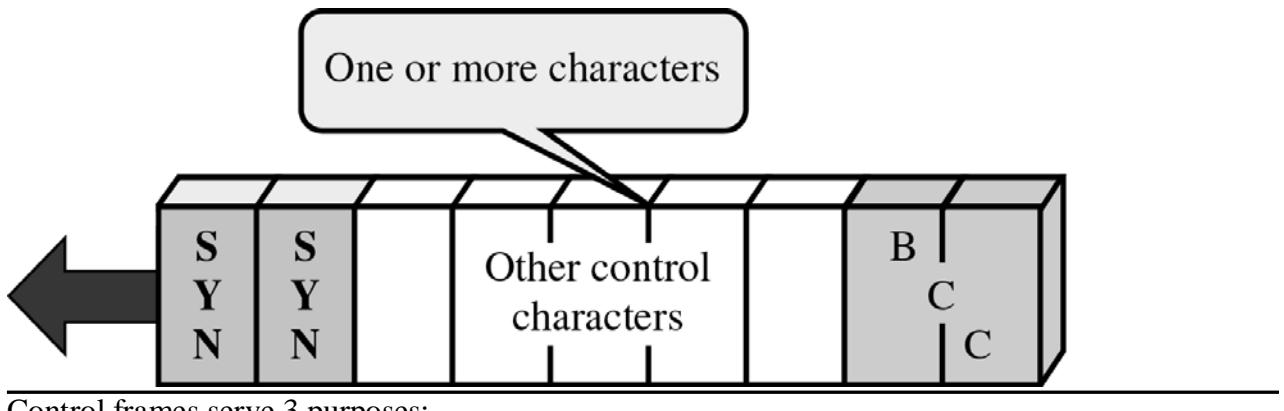
- The more bits in a frame, the more are the chances of an error
- For this reason, text in a message is often divided b/w several blocks
- Each block starts with STX and ends with ITB, intermediate text block except the last one
- Last block ends with ETX character
- After the ETX has been reached, and the last BCC checked, the receiver sends a single ACK for the entire frame
- The figure includes only 2 blocks but actual frames can have more than two
- Some messages may be too long to fit in a frame
- Several frames can carry continuation of a single message
- To let the Rx know that the end of frame is not the end of transmission, the ETX character in all the frames but the last one is replaced by an End of Transmission block (ETB)
- The receiver must acknowledge each frame separately

## Multi Frames



## Control Frames

- A control frame is used by one device to send commands to or to get information from another device
- A control frame contains control characters but no data
- It carries information specific to the functioning of the data link layer itself



Connection establishment								
<b>Bid</b> Point-to-point connection request.			<b>Poll</b> Primary polls secondary.					
<b>Select</b> Primary selects secondary.			<b>Positive response to select or bid</b> Ready to receive data.					
<b>Negative response to select or bid</b> Not ready to receive data.			<b>Negative response to poll</b> Not ready to send data.					
Flow and error control								
<b>Positive ACK of even frames</b> Frame number 0 received.			<b>Positive ACK of odd frames</b> Frame number 1 received.					
<b>Negative ACK of frames</b> Error in the frame received.			<b>Wait &amp; ACK</b> ACK of previous frame, not ready to receive more.					
<b>Reverse interrupt</b> Request for interruption, urgent data to send.			<b>Temporary delay</b> Temporarily delayed but does not relinquish the line.					
Connection termination								
<b>End of transmission</b> Station finished sending data.								

## Summary

- Data Link Protocols
- Asynchronous Protocols
- Synchronous Protocols

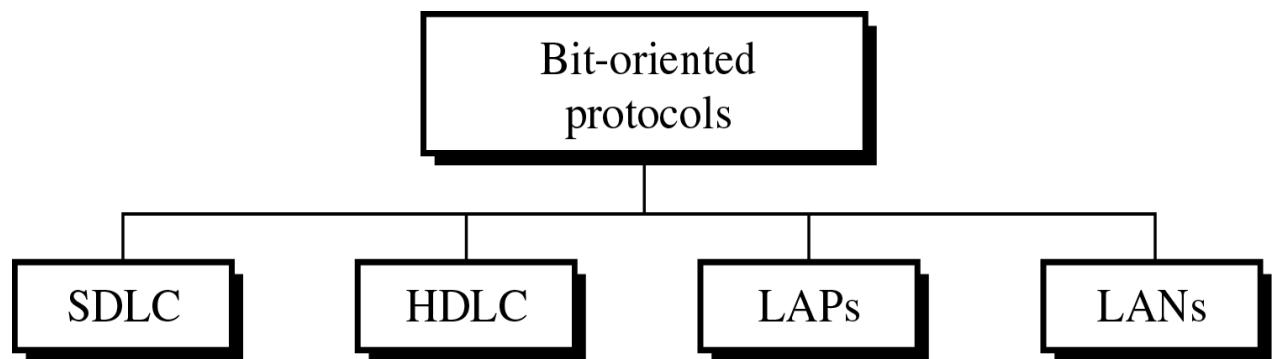
## Reading Sections

- Section 11.1, 11.2
- “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

## **LECTURE #41**

### **Bit-Oriented Protocols**

- In character-oriented protocols, bits are grouped into predefined patterns forming characters
- By comparison, bit-oriented protocols can pack more information into shorter frames



A lot of bit-oriented protocols have been developed over the years:

- One of these HDLC is the design of the ISO and has become the basis for all bit-oriented protocols in use today
  - In 1975, IBM gave Synchronous Data Link Control (SDLC)
  - In 1979, ISO answered with High Level Data Link Control (HDLC)
- Since 1981, ITU-T has developed a series of protocols called Link Access Protocols
- LAPs: LAPB, LAPD, LAPM, LAPZ etc. all based on HDLC

HDLC is basis for all protocols, so we will study it in detail:

### **High Level Data Link Control (HDLC)**

- Bit-oriented data link protocol designed for:
  - Full Duplex and Half Duplex
  - Point-to-point And Multipoint Links

### **Characterization of HDLC**

HDLC can be characterized by:

- Station Types
- Configurations
- Response Modes

### **STATION TYPES (1)**

HDLC differentiates b/w 3 types of stations:

- Primary Station

- Secondary Station
- Combined Station

## **STATION TYPES (2)**

### **Primary Station**

- Primary station works in the same way as primary devices in the discussion of flow control
- The primary is a device in point-to-point or multipoint line configuration that has complete control of the link

## **STATION TYPES (3)**

### **Secondary Station**

- The primary sends commands to the secondary stations
- A primary issues commands and a secondary issues responses

## **STATION TYPES (4)**

### **Combined Station**

- A combined station can both command and respond
- A combined station is one of a set of connected peer devices programmed to behave either as a primary or as a secondary depending on the nature and the direction of the transmission

### **Configuration (1)**

- Configuration refers to the relationship of the hardware devices on a link
- Primary , secondary and combined stations can be configured in three ways:

- Y Unbalanced Configuration
- Y Symmetrical Configuration
- Y Balanced Configuration

### **Configuration (2)**

#### **Unbalanced Configuration**

- Also called Master/Slave Configuration
- One device is a primary and others are secondary
- Unbalanced configuration can be point to point if only two devices are involved
- Most of the times it is multipoint with one primary controlling several secondaries

#### **Configuration (3)**

#### **Symmetrical Configuration**

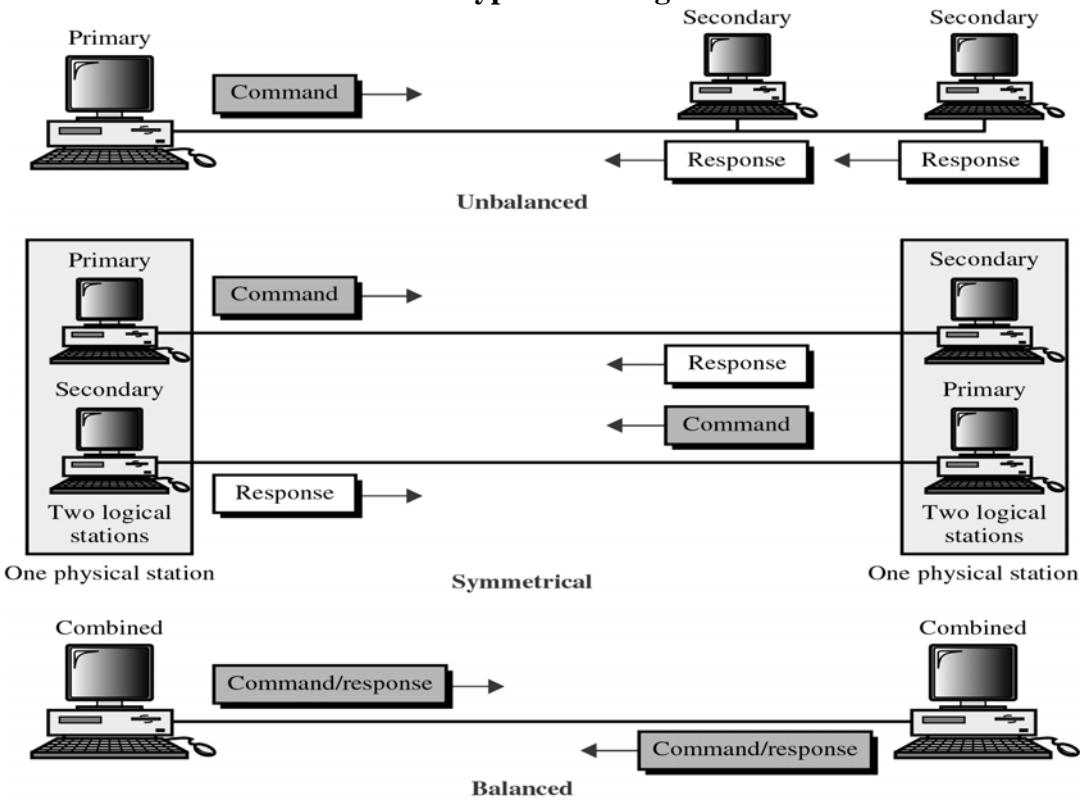
- Each physical station on a link consists of two logical stations, one a primary and the other a secondary
- Separate lines link the primary aspect of one physical station to the secondary aspect of another physical station

## Configuration (4)

### Balanced Configuration

- Both stations in a point-to-point topology are of combined type
- HDLC does not support balanced multipoint

### Station Types & Configurations



### Modes

- A mode in HDLC is the relationship b/w two devices involved in an exchange
- The mode describes who controls the link
- HDLC supports 3 modes of communication b/w stations:
  - Normal Response Mode (NRM)
  - Asynchronous Response Mode (ARM)
  - Asynchronous Balanced Mode (ABM)

	NRM	ARM	ABM
Station type	Primary & secondary	Primary & secondary	Combined
Initiator	Primary	Either	Any

### Normal Response Mode (NRM)

- Refers to the standard primary-secondary relationship
- Secondary device must have permission from primary device before transmitting
- Once permission has been granted, the secondary may initiate a response transmission of one or more frames containing data

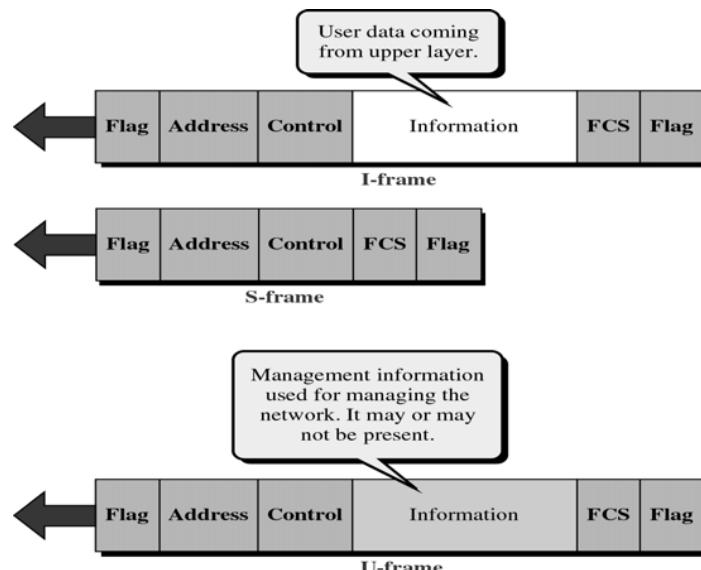
### **Asynchronous Response Mode (ARM)**

- A secondary may initiate a TX w/o permission from the primary whenever the channel is idle
- ARM does not alter the primary secondary relationship in any other way
- All transmissions from the primary still go to the secondary and are then relayed to the other devices

### **Asynchronous Balanced Mode (ABM)**

- All stations are equal and therefore only combined stations connected in point-to-point are used
- Either combined station may initiate TX with the other combined station w/o permission

### **HDLC Frames**



HDLC defines 3 types of Frames:

- Information Frames (I-Frames)
- Supervisory Frames (S-Frames)
- Unnumbered Frames(U-Frames)

**I-Frames** are used to transport user data and control information relating to user data

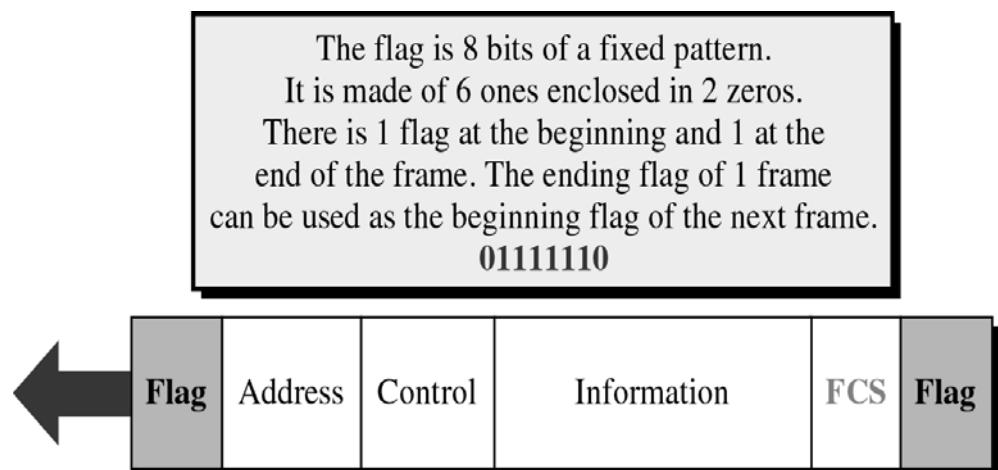
**S-Frames** are used only to transport control information

**U-Frame** are reserved for System Management

Each frame in HDLC may contain up to six fields

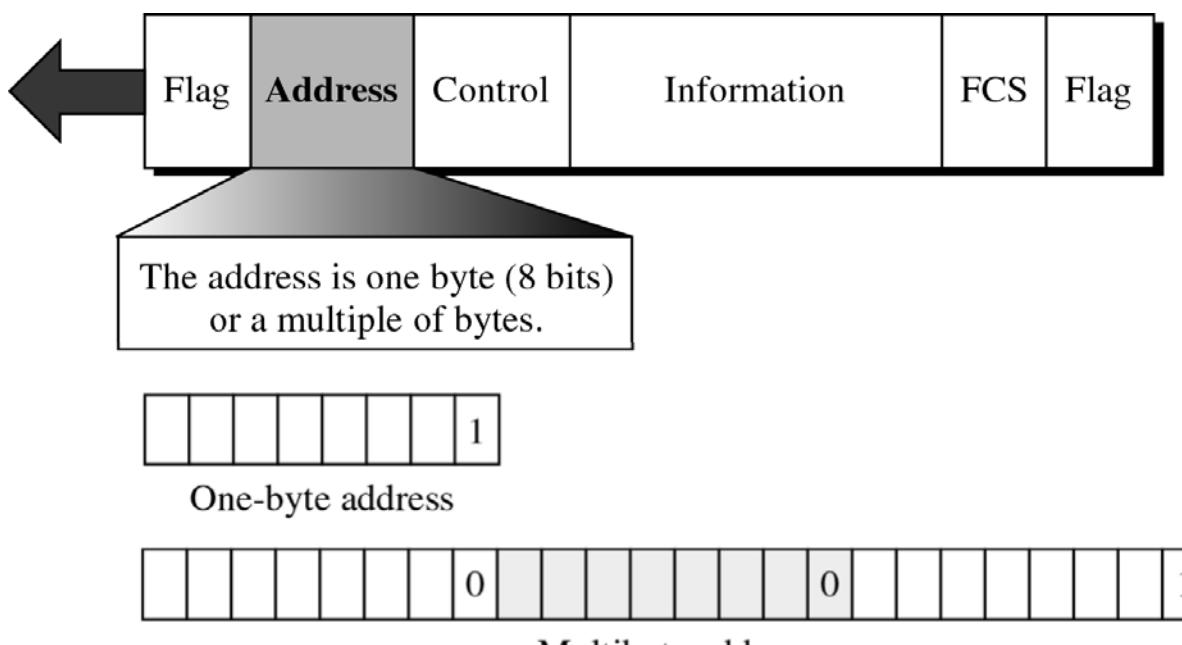
- A beginning Flag Field
- An address field
- A control field
- An information Field
- A frame check sequence (FCS)
- An ending Flag Field

## Flag Field



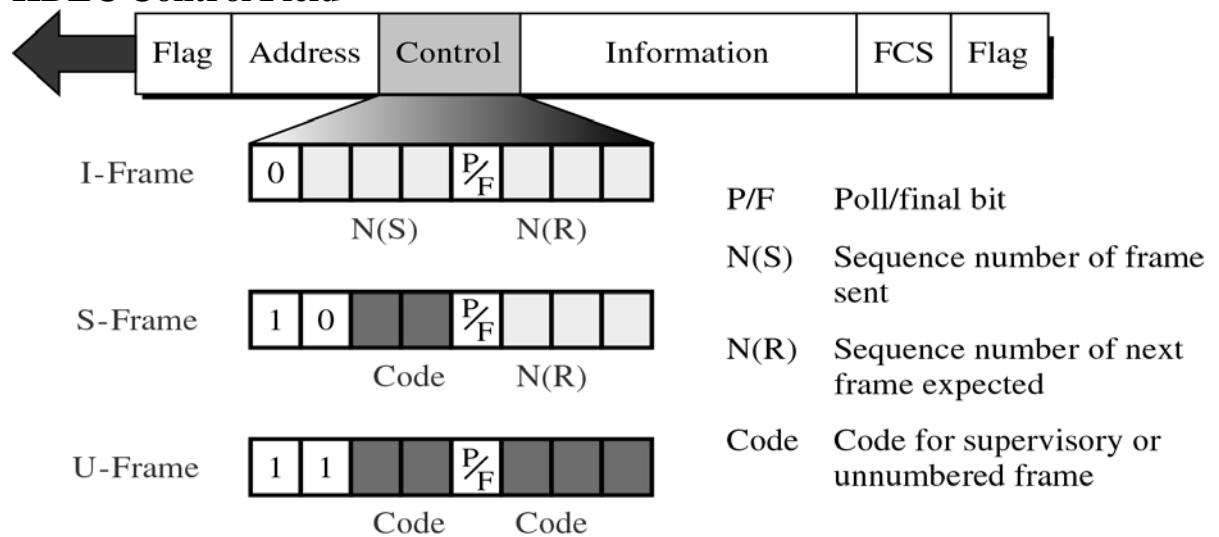
- The flag field of an HDLC frame is an 8-bit sequence with a bit patter 01111110 that identifies both the beginning and the ending of the of a frame
- It serves as a Synchronization pattern for the receiver
- Fig. shows placement of 2 flag fields in an I-Frame

## HDLC Address Field



- The second field of HDLC frame contains the address of the secondary station that is either the originator or the destination of the frame
- If a primary station creates Frame it includes a ‘To’ address and if a secondary creates the frame, it contains a ‘From’ address
- Can be of one byte or several bytes depending upon the network
- If the address field is only 1 byte, the last bit is always a 1
- If the address is of several bytes, all bytes but the last one will end with 0 , and the last will end with a 1
- Ending each intermediate byte with 0 indicates to the receiver that there are more address bytes to come

## HDLC Control Field

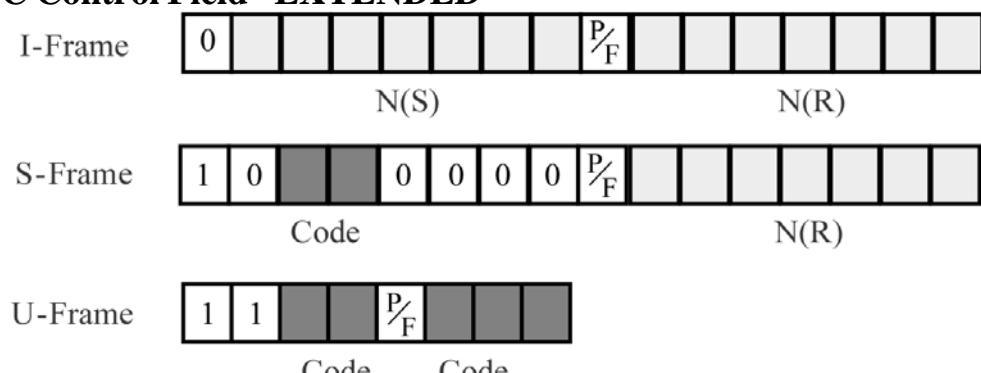


- The control field is a one or two byte segment of the frame used for flow management
- The two byte case is called the Extended Mode

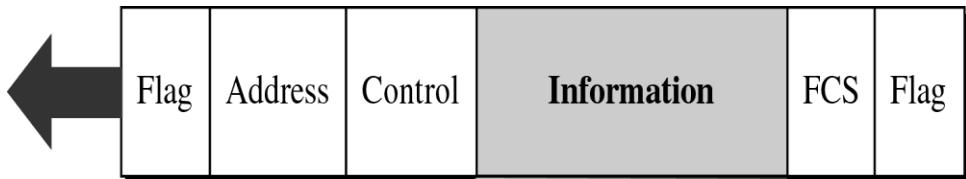
Control fields differ depending on the frame type:

- If the control field is a 0, the frame is an I-Frame
- If the first bit is 1 and the second bit is a 0 , it is S-Frame
- If both first and second bits are 1's, it is U-Frame
- **P/F bit** is a single bit with dual purpose
- It has meaning only when it is '1' and it can mean Poll or Final
- When the frame is sent by a primary to secondary, it means POLL
- When the frame is sent by a secondary to a primary, it is FINAL

## HDLC Control Field –EXTENDED

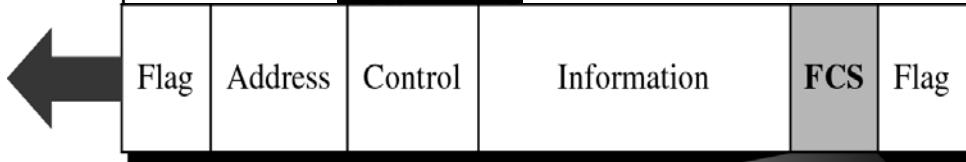


- Control field in the I-Frame and S-Frame is two bytes long to allow seven bits of sending and receiving sequence
- However the control field in the U-Frame is still one byte



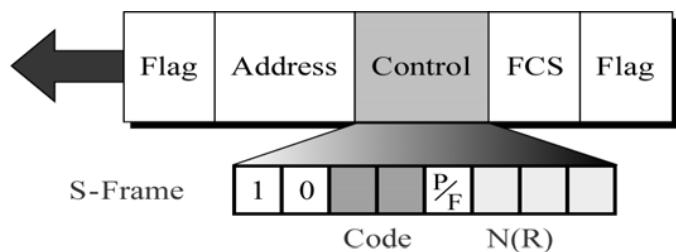
It contains user data in an I-Frame.  
It does not exist in an S-Frame.  
It contains management information in a U-frame.

- Information field contains the user's data in an I-Frame and Network Management information in a U-Frame
- An S-Frame has no information field
- Its length can vary from one network to another but remains fixed within each network
- It is possible to send Control information in the information field of the I-Frame along with data.
- This process is called **Piggybacking**

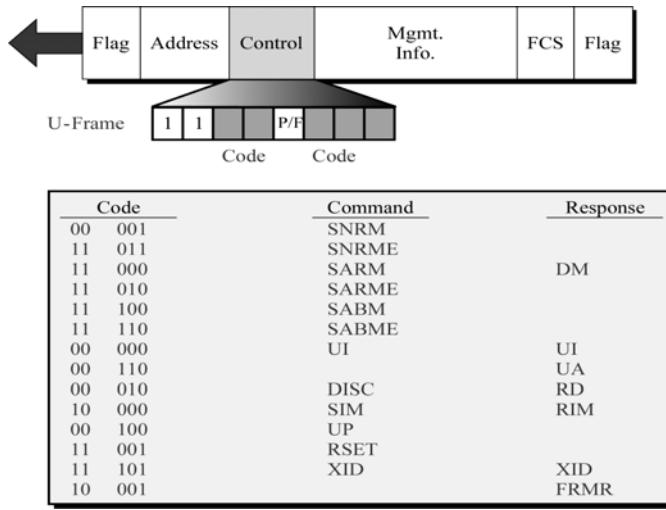


Frame check sequence is the error detection field.  
It can be a two-byte or a four-byte CRC.

- The FCS is HDLC's error detection field
- 
- It can contain a two- or four byte CRC



Code	Command
00	RR Receive ready
01	REJ Reject
10	RNR Receive not ready
11	SREJ Selective-reject



## Link Access Procedures

### Y LAPB

—Link access procedure, balanced

### Y LAPD

—Link access procedure for D- channel

### Y LAPM

—Link access procedure for modems

## Summary

- Synchronous Protocols
  - Bit-Oriented Protocols
- HDLC
- Link Access Protocols (LAPs)

## Reading Sections

- Section 11.4, 11.5
- “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #42**

## **Local Area Network**

A local area network is a data communication system that allows a number of independent devices to communicate directly with each other in a limited geographical area

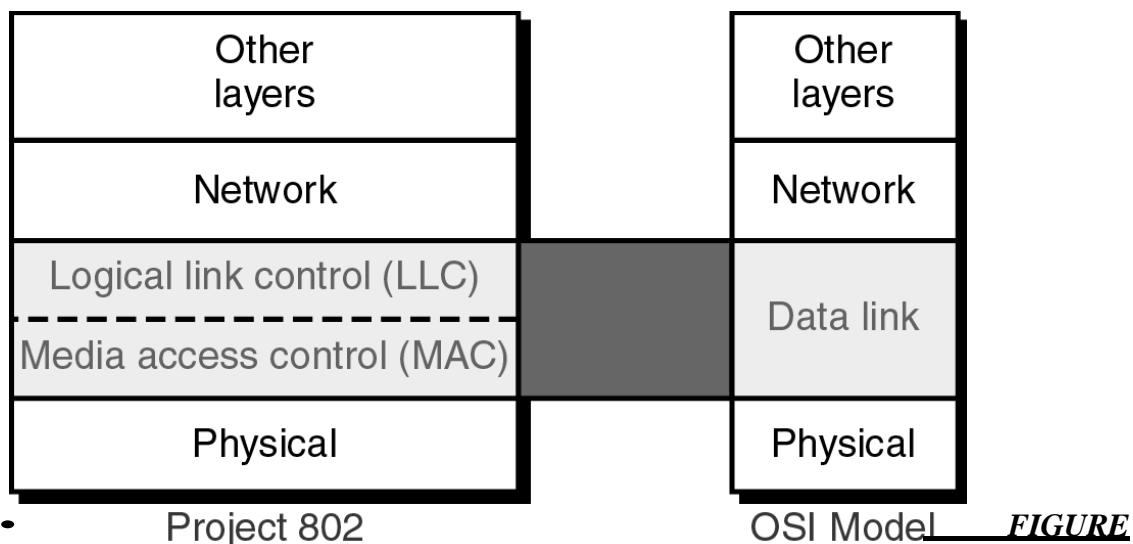
### **Architectures for LANS**

Dominated by 4 architectures:

- Ethernet, Token Bus, Token Ring -7 Standards of IEEE and a part of its Project 802
- Fiber Distributed Data Interface (FDDI) -7 ANSI Standard

### **LAN compared with OSI**

- In 1985, the computer society of IEEE started a project called PROJECT 802 to set standards to enable intercommunication b/w equipment from a variety of manufacturers
- This project does not seek to replace any part of the OSI Model
- Instead it is a way of specifying functions of physical layer , the data link layer and up to some extent the network layer to allow for interconnectivity of major LAN protocols
- The relationship of IEEE project 802 to the OSI Model is shown in the figure:



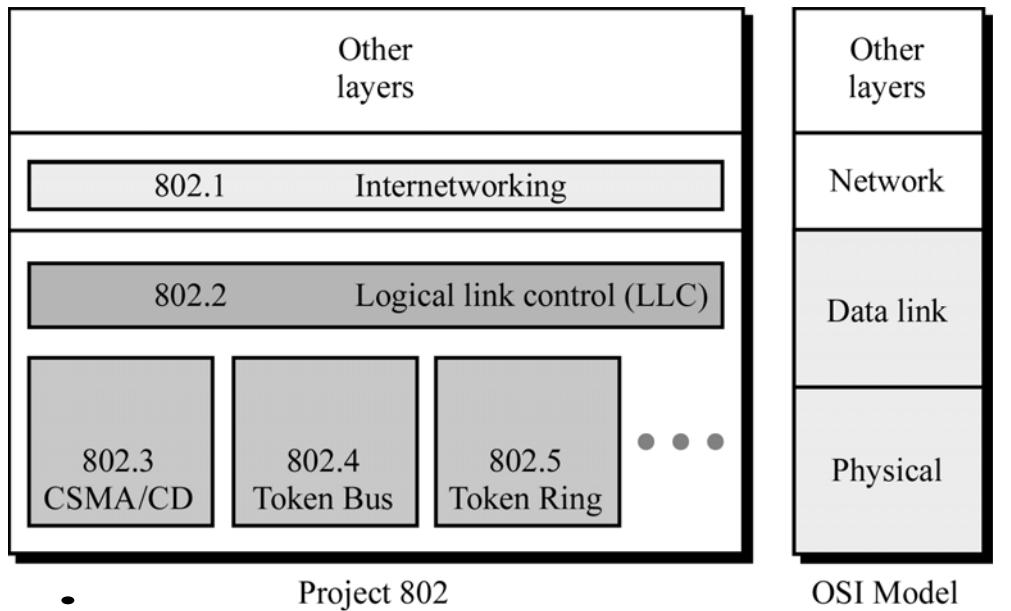
**FIGURE**

- IEEE has divided the data link layer into two sub-layers:  
**Logical Link Control (LLC)**  
**Medium Access Control (MAC)**
- LLC is non-architecture specific i.e. it is the same for all IEEE-defined LANs
- The MAC sub layer on the other hand contains a number of distinct modules; each carries proprietary info specific to the LAN product being used
- LAN compared with OSI -Figure

- In addition to the two sub-layers , Project 802 contain a section governing internetworking
- This section assures the compatibility of different LANs and MANs across protocols and allows data to be exchanged across otherwise incompatible networks

## PROJECT 802

- Strength of Project 802 is Modularity
- By subdividing the functions necessary for LAN management, the designers were able to standardize those that can be generalized and isolate those that must remain specific



### Y IEEE 802.11

- Is the section of Project 802 devoted to internetworking issues in LANs and MANs
- Although not yet complete, it seek to resolve the incompatibilities b/w network architectures w/o requiring modifications in existing addressing, access, and error recovery mechanisms

### Y LLC

In general, IEEE project 802 model takes the structure of an HDLC frame and divides into two sets of functions:

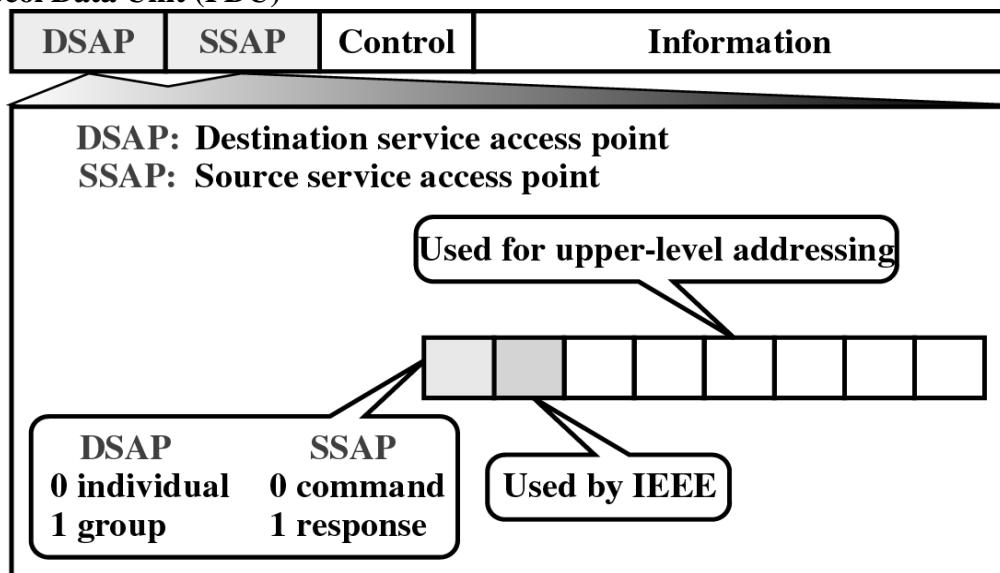
- One set contains the end-user portions of the frame: The logical address, control information and data
- These functions are handled by IEEE 802.2 LLC protocol
- LLC is upper of data link layer and is common to all LAN protocols

### Y MAC

- The second set of functions, the MAC sub-layer , resolves the contention for the shared media
- It contains Synchronization Flag, Flow and Error control specifications as well as the physical address of next station to receive & route a packet

- MAC protocols are specific to LAN using them (Ethernet, Token Ring, Token Bus etc)
- Protocol Data Unit (PDU)
- The data unit in the LLC level is called the Protocol Data unit (PDU)
- The PDU contains 4 fields familiar from HDLC:
  - A destination service access point (DSAP)
  - A source service access point (SSAP)
  - A control field
  - An Information field

### **Protocol Data Unit (PDU)**



### **DSAP and SSAP**

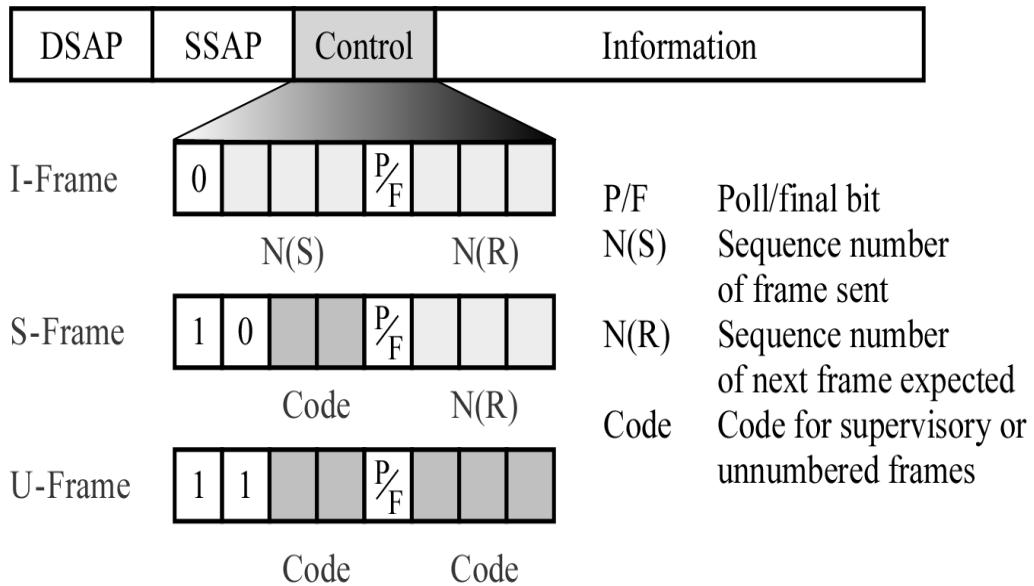
- The DSAP and SSAP are addresses used by LLC to identify the protocol stacks on the receiving and sending machines that are generating and using data
- The first bit of the DSAP indicates whether the frame is intended for an individual or a group
- The first bit of SSAP indicates whether the communication is a command or a response PDU

### **CONTROL**

- The control field of PDU is identical to the control field in HDLC.
- As in HDLC, PDU frames can be I-frames, S-frames, or U-Frames and carry all of the codes and the information that the corresponding HDLC frame carry

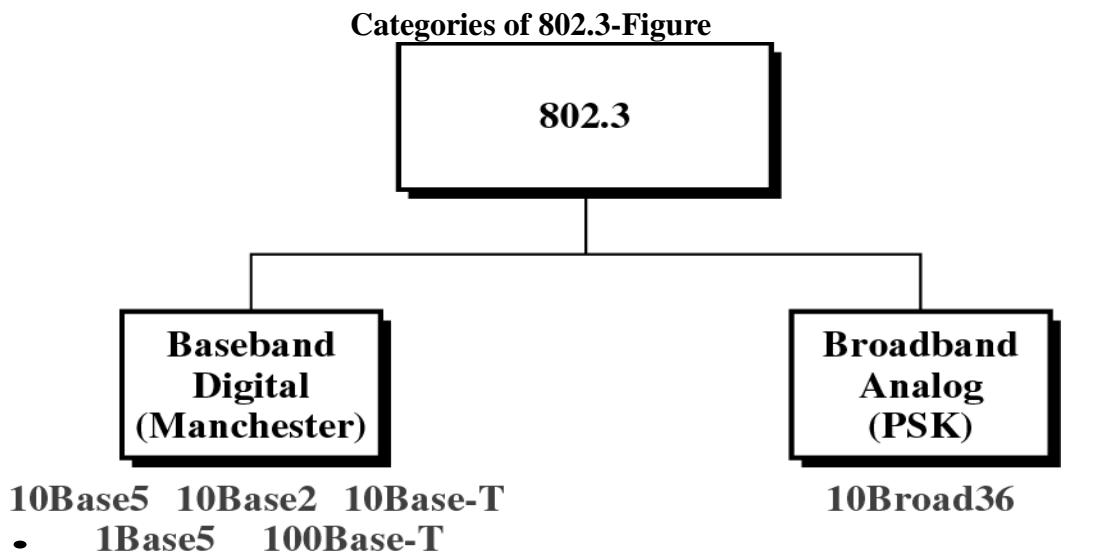
**•The PDU has no flag fields, no CRC, and no station address**

**•These fields are added in the lower sub-layer i.e. MAC Layer**



## ETHERNET

IEEE 802.3 supports a LAN standard originally developed by Xerox and later extended by a joint venture b/w Digital Equipment Corporation, Intel Corporation and Xerox. This is called ETHERNET



**IEEE 802.3 defines two categories:**

- BASEBAND
- BROADBAND

The word “base” specifies a digital signal  
 The word “broad” specifies an analog signal

IEEE divides the base band category into 5 standards:

–10 Base 5 , 10 Base 2, 10 Base-T, 1 Base 5, 100 Base-T

- The first number (10, 1, or 100) indicates the data rate in Mbps.
- The last number or letter (5, 2, 1, or T) indicates the maximum cable length or the type of cable
- IEEE defines only one specification for the broadband category:  
    –10 Broad 36
- Again the first number (10) indicates the data rate.
- The last number defines the maximum cable length
- Max. cable length restriction can be changed using networking devices i.e. Repeaters or Bridges

### **Access Method: CSMA/CD**

- **Multiple Access:** Multiple users access to a single line
- **Carrier Sense:** A device listens to the line before it transmits
- **Collision Detection:** Extremely high voltage indicates a collision

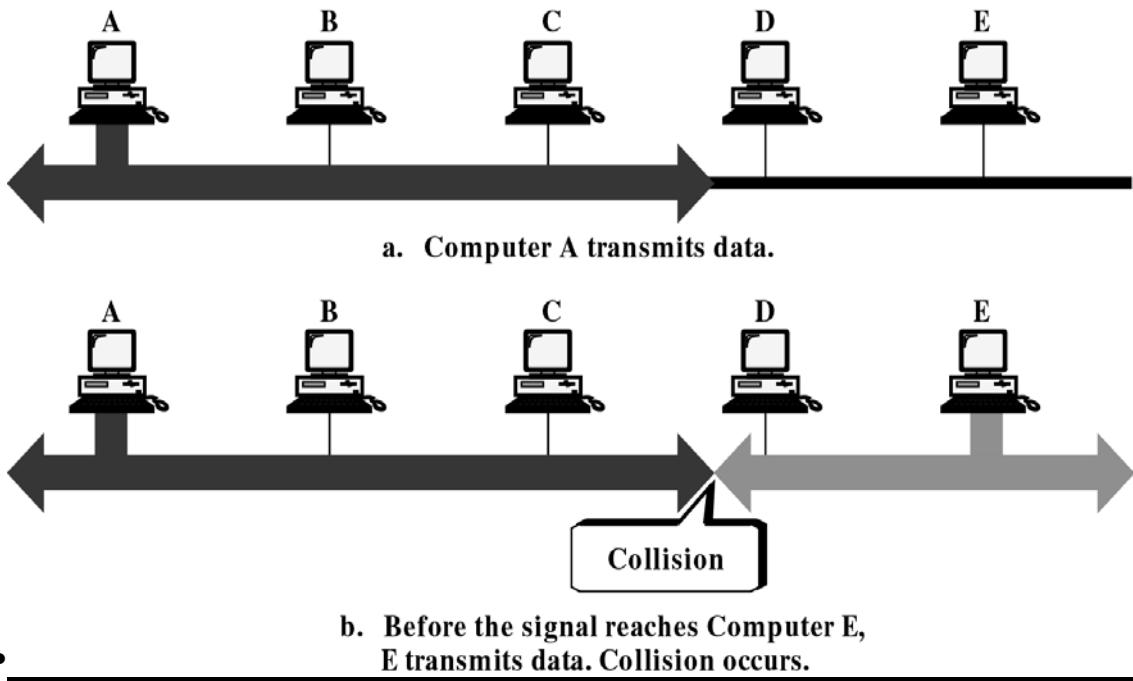
### **Need for a Access Method**

- Whenever multiple users have unregulated access to a single line, there is a danger of signals overlapping and destroying each other
- Such overlaps which turn signals to Noise are called **COLLISIONS**
- As traffic increases on multiple-access link, so do collisions
- A LAN therefore needs a mechanism to coordinate traffic, minimize the number of collisions and maximizes the number of frames that are delivered successfully
- The access mechanism used in Ethernet is called **Carrier Sense Multiple Access with Collision Detection (CSMA/CD)**

#### **•MA-7 CSMA -7 CSMA/CD**

- **CSMA/CD**
- The original design was a multiple access method in which every workstation had equal access to the link
- In multiple access, there was no provision for traffic coordination
- Access to the line was open to any node at any time
- Any device wishing to transmit sent its data and then relied on ACKs to know if it had reached its destination
- In a CSMA system, any device wishing to transmit must first listen for existing traffic on the line
- A device must listen by checking for voltage
- If no voltage is detected, the line is considered idle and the TX is initiated
- CSMA cuts down on the number of collisions but does not eliminate them
- If a system transmits after checking the line and another system transmits during this small interval, collisions can still occur
- The final step is the addition of Collision Detection (CD)
- In CSMA/CD, the station wishing to transmit first listens to make certain the link is free, then transmits its data, then listens again

- During the data transmission, the station checks for the extremely high voltages that indicate a collision
- If a collision is detected, the station quits the current TX and waits a predetermined amount of time for the line to clear, then sends its data again



- 

### **Y Addressing**

- Each station on the Ethernet network such as a PC, workstation or printer has its own Network Interface Card (NIC)
- The NIC usually fits inside the station and provides the station with a 6-byte physical address
- The number on the NIC is unique

### **Y Data Rate**

- Ethernet LANs can support data rates between 1 and 100 Mbps

### **Implementation**

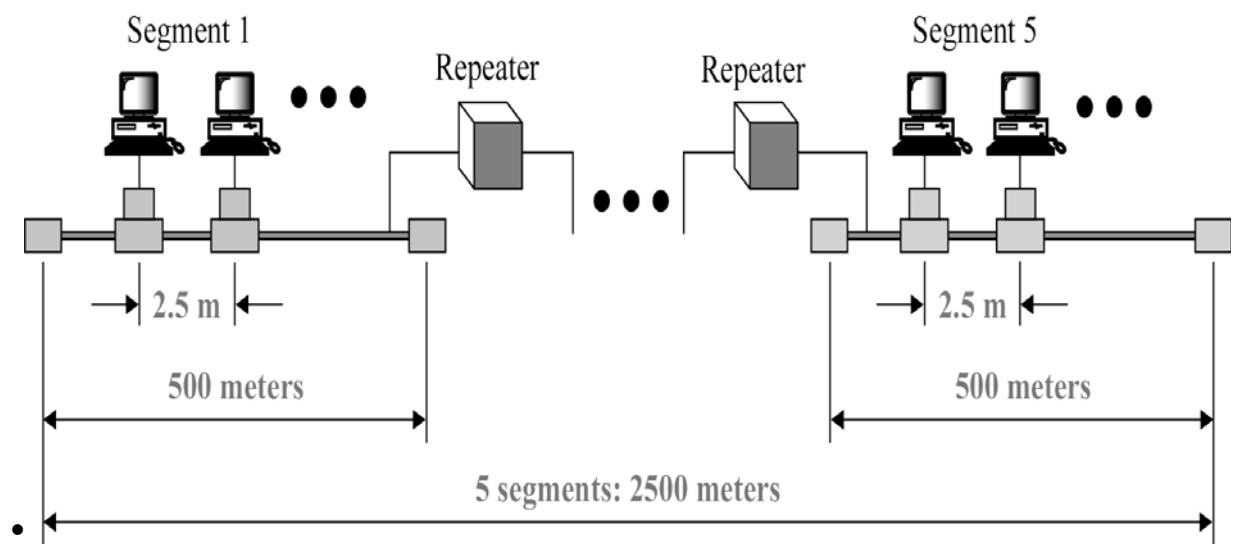
- In IEEE 802.3 standard, the IEEE defines types of cables, connections and signals that are to be used in each of the five different Ethernet implementations
- Each frame is transmitted to every station on the link but read only by the station to which it is addressed

### **10 Base 5: Thick Ethernet**

- The first of the physical standards defined in IEEE 802.3 model is called 10 Base 5, Thick Ethernet or Thicknet
- The name is derived from the size of the cable which is roughly the size of a garden hose and cannot be bent with hands
- 10 Base 5 is a bus topology LAN that uses base band signaling and has a max. segment length of 500 meters

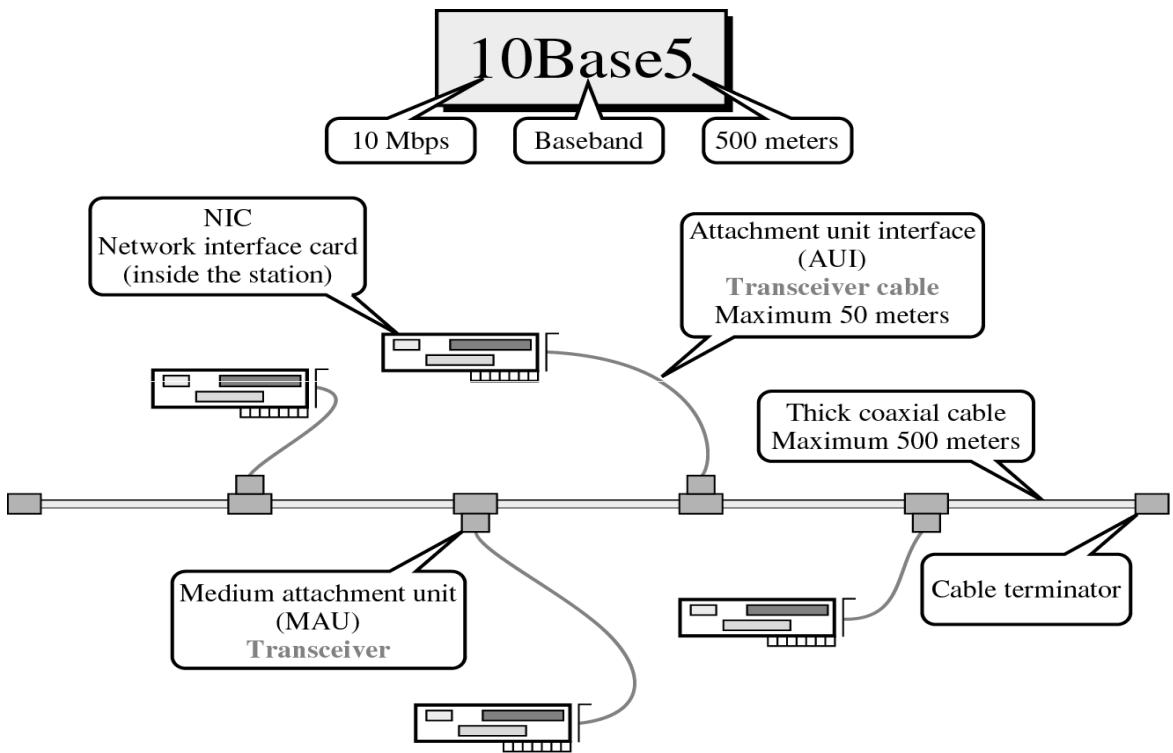
## Size Limitations of 10 Base 5

- Networking devices such as Repeaters and Bridges are used to overcome the size limitation of LAN
  - In Thicknet, a LAN can be divided into segments by connecting devices
  - In this case, the length of each segment is limited to 500 meters
  - However to reduce collisions, the total length of the bus should not exceed 2500 meters (5 segments)
  - Also the standard demands that each station be separated from each other 2.5 meters
  - 200 stations per segment and 1000 stations total



## **Topology of 10 Base 5**

The physical connectors and cables utilized by 10 base 5 include coaxial cable, Network Interface Card, Transceivers and Attachment Unit Interface (AUI) cables



#### Y **RG-8 Cable**

RG-8 cable (Radio Government) is a thick coaxial cable that provides the backbone of IEEE 802.3 standard

#### Y **TRANSCEIVER**

Each station is attached by an AUI cable to an intermediary device called a Medium Attachment Unit (MAU) or a Transceiver

Transceiver performs the CSMS/CD function of checking for voltages and collisions on the line and may contain a small buffer

#### Y **AUI Cables**

Each station is linked to its corresponding transceiver by an **AUI cable** also called the **Transceiver cable**

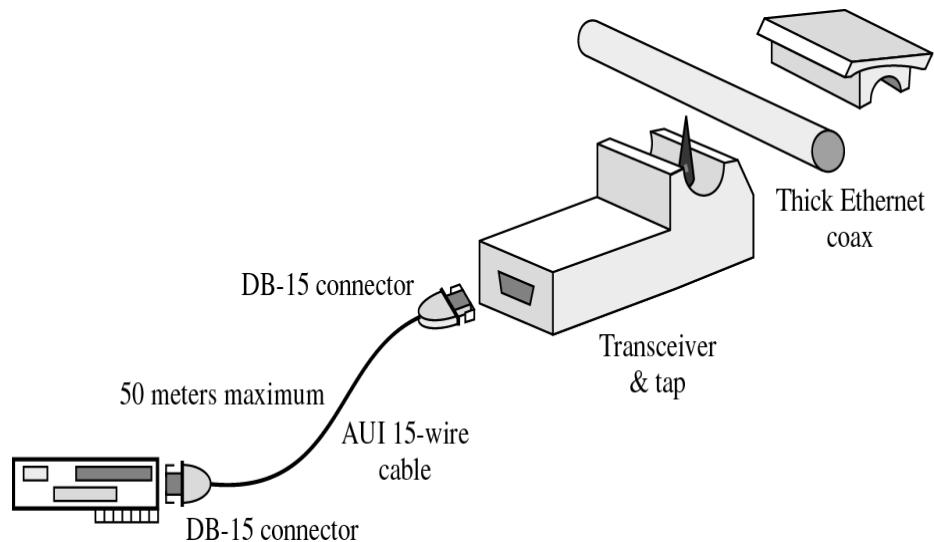
An AUI is a 15 wire cable with plug that performs the physical layer interface functions b/w the station and the transceiver

An AUI has a max. Length of 50 meters and it terminates in a 15-pin DB- 15 connector

#### Y **TRANSCEIVER TAP**

Each transceiver contains a connecting mechanism called a TAP because it allows the transceiver to tap into the line at any point

The TAP is a thick cable sized well with a metal spike in the center



- The spike is attached to wires inside the transceiver
- When the cable is pressed into the well, the spike pierces the jacket and sheathing layers and makes an electrical connection b/w the transceiver and the cables
- This kind of connector is often called a **VAMPIRE TAP** because it bites the cable

### **Summary**

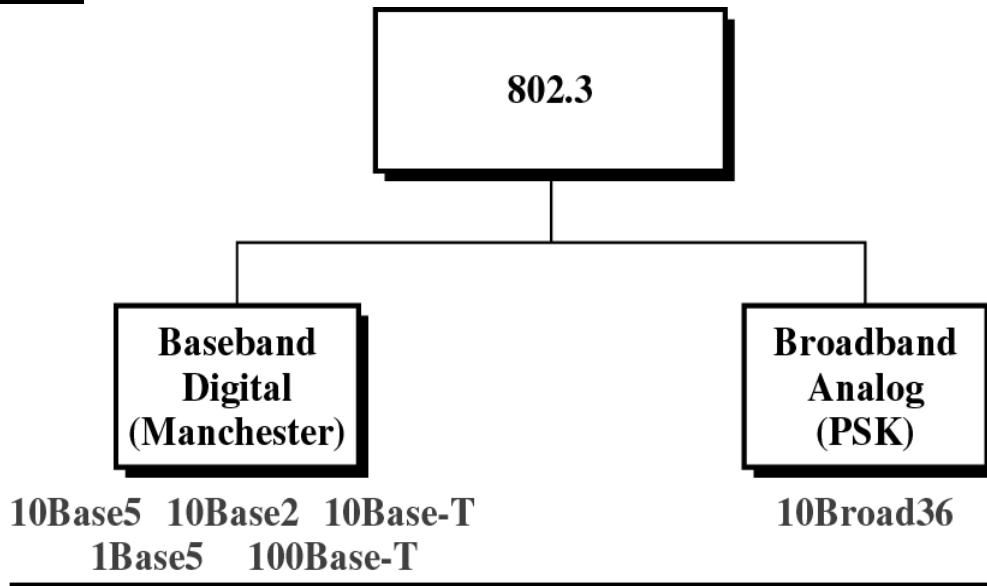
- Local Area Networks (LANs)
- Project 802
- Protocol Data Unit (PDU)
- Ethernet

### **Reading Sections**

- Section 12.1, 12.2
- “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

## **LECTURE #43**

### **Ethernet**



### **10 Base 2: Thin Ethernet**

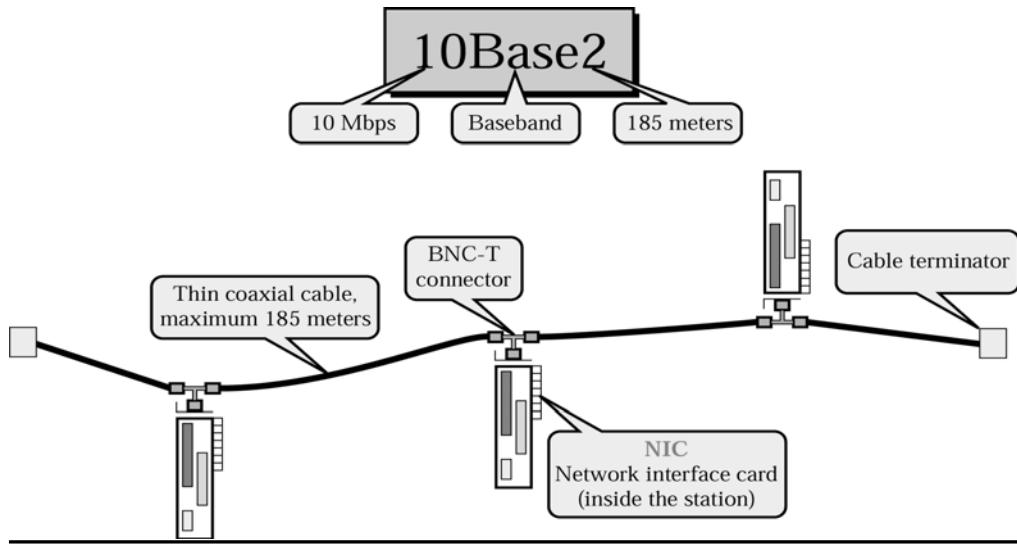
- The second Ethernet Implementation defined by IEEE 802 series is called 10 Base 2, Thin Ethernet, Thin net
- Also called Cheap net because it provides an inexpensive alternative to 10 Base 5 Ethernet, with the same data rate
- Like 10 Base 5, 10 Base 2 is a Bus Topology LAN
- 10 Base 2: Thin Ethernet
  - The advantages of thin Ethernet are :
    - reduced cost and
    - ease of installation

Because the cable is lighter weight and more flexible than that used in Thicknet

- The disadvantages are:
  - Short Range (185 m as opposed to 500 m)
  - Smaller Capacity (thinner cable accommodates fewer stations)

#### **- Physical Topology of 10 base 2**

- The connectors and cables utilized are: NICs, Thin coaxial cable and BNC-T connectors
- In this technology, the transceiver circuitry is moved to the NIC, and transceiver tap is replaced by connector that splices directly into the cable
- No AUI cables are needed



#### Y NIC

- o NICs in Thin Ethernet provide the same functionality as those in Thicknet plus the functions of the Transceiver
- o A 10 Base 2 NIC not only provides the station with an address but also checks for voltages on the link

#### Y RG-58 Cable

- o These cables are relatively easy to install and move around
- o Especially inside the buildings where cable must be pulled through the walls and the ceilings

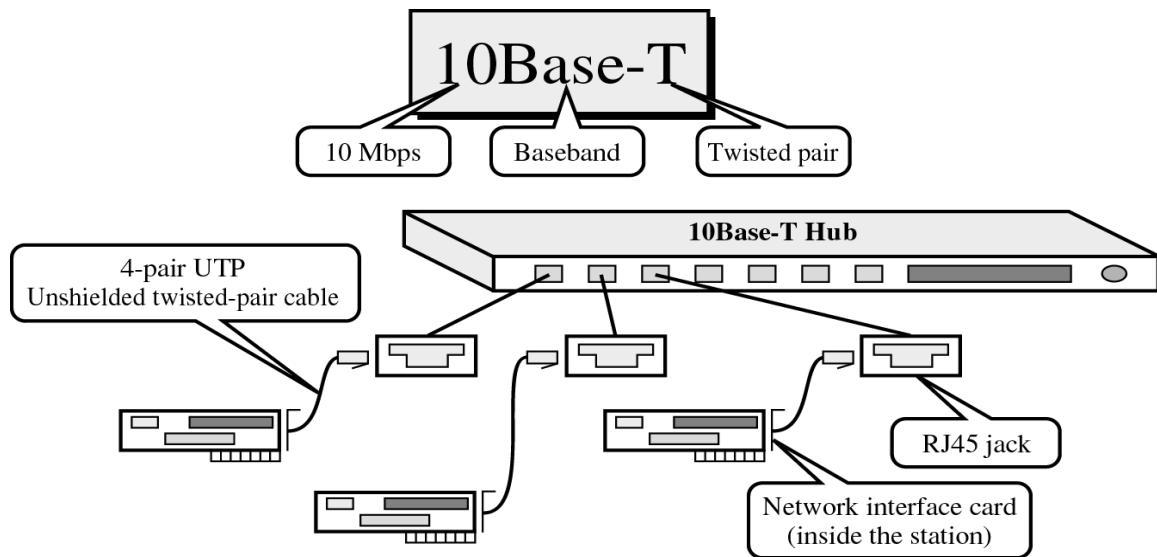
#### Y BNC-T

The BNC-T connector is a T-shaped device with three ports:

- One for the NIC
- One each for the input and output ends of the cable

### **10 base T: Twisted Pair Ethernet**

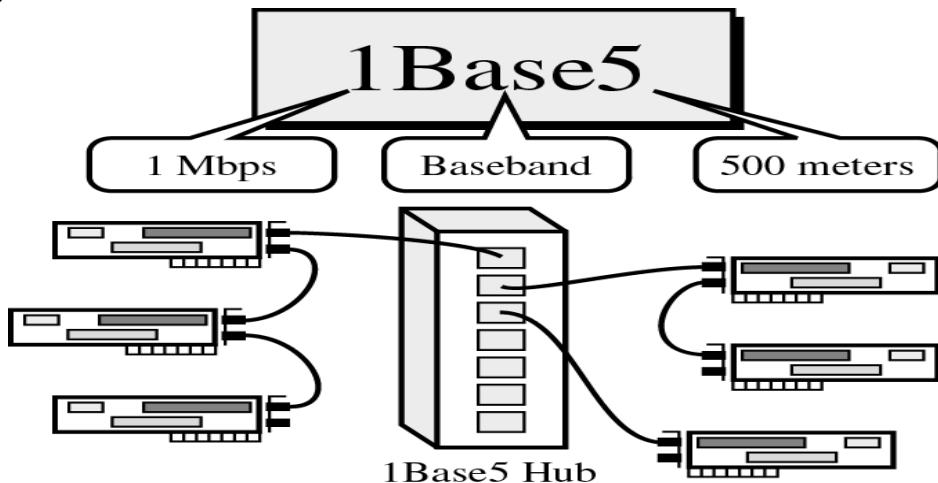
- o Most popular standard defined in IEEE 802.3 series is 10 Base T also called Twisted Pair Ethernet
- o It is a Star topology LAN that uses Unshielded Twisted pair (UTP) cable instead of coaxial cable
- o It supports a data rate of 10 Mbps and has a maximum length of 100 meters



- Instead of individual transceivers, 10 Base T Ethernet places all of its networking operations in an intelligent Hub with a port for each station
- Stations are linked into the hub by four pair RJ-45 cable terminating at each end in a male-type connector much like a Telephone jack
- The hub fans out any transmitted frame to all of its connected stations
- Each station contains a NIC
- A 4-pair UTP of not more than 100 meters connects the NIC to the appropriate port in 10 Base T Hub
- The weight & flexibility of the cable and the convenience of RJ-45 jack and plug makes it the easiest to install and reinstall

### **1 Base 5: STAR LAN**

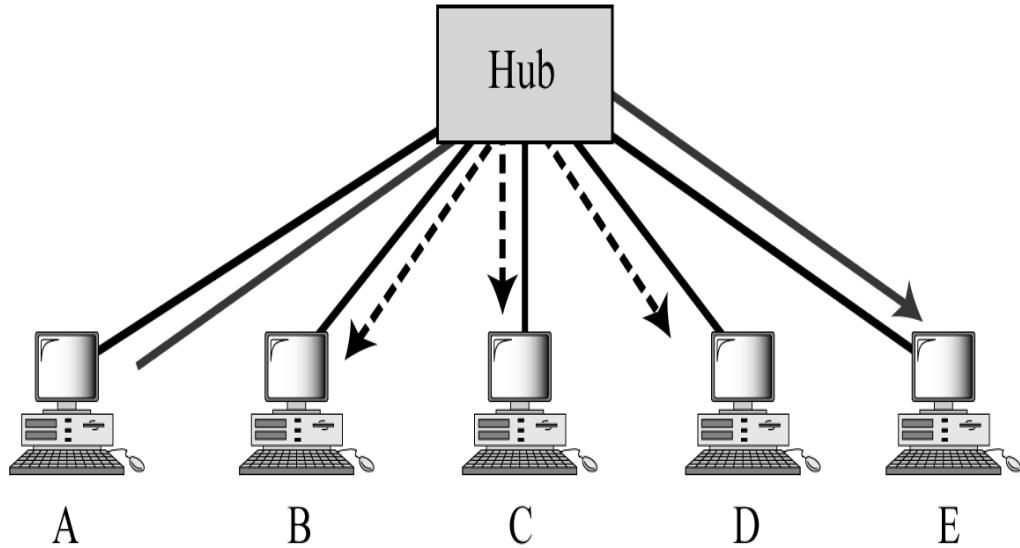
- Star LAN is infrequently used nowadays because of its slow speed of approx. 1Mbps
- Range can be increased by **DAISY CHAINING**
- Like 10 Base T, Star LAN uses Twisted pair cable to connect stations to a central hub
- Star LAN allows 10 devices to be linked with only the lead device connected to the hub



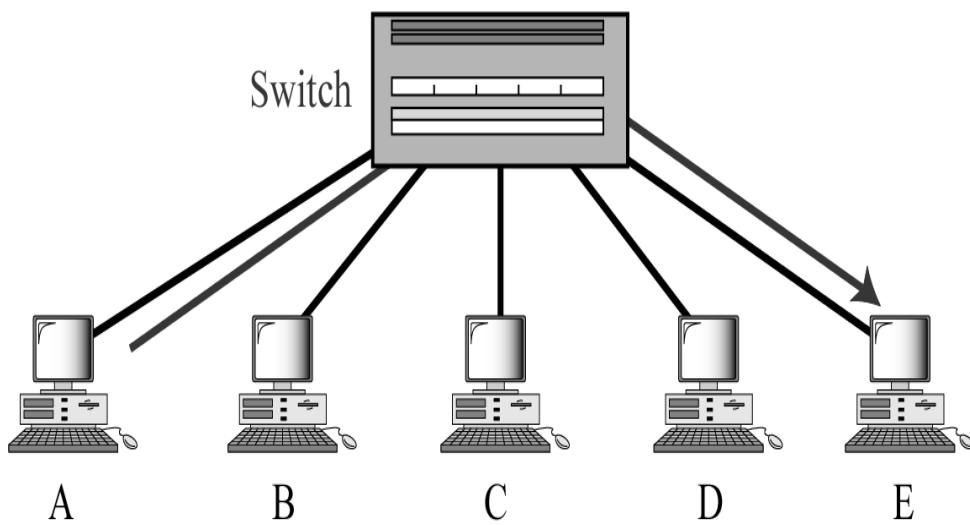
### **Switched Ethernet**

- An attempt to improve the performance of 10BASE-T
- $N \times 10$  Mbps with N devices

Only station E is the actual destination, but all the stations receive the frame sent from A.

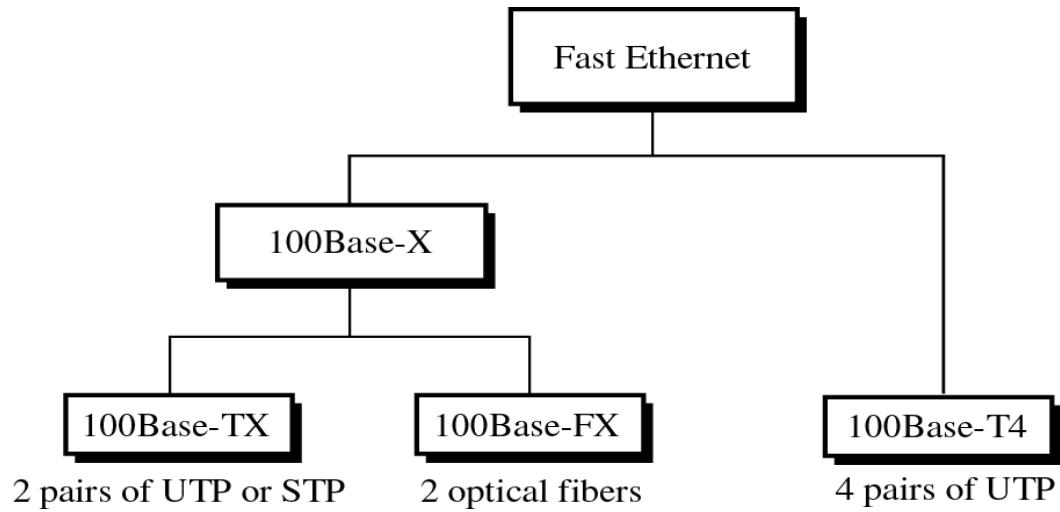


Only station E receives the frame, so the rest of the media is free for another transmission.

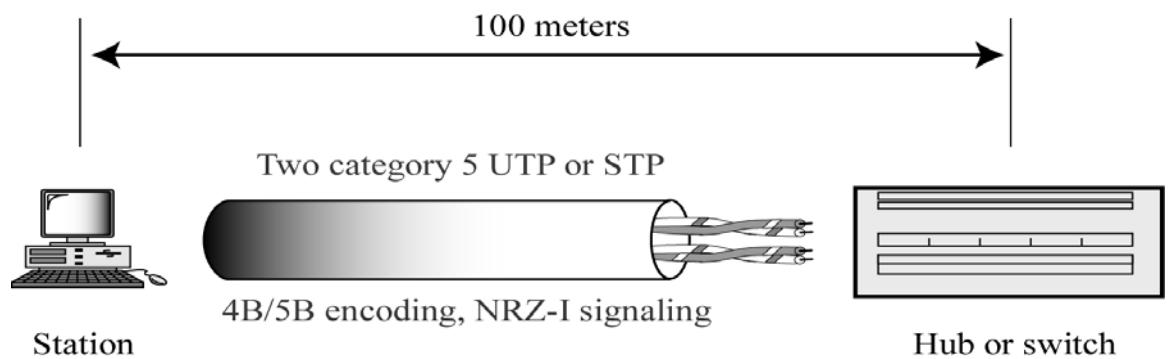


### Fast Ethernet

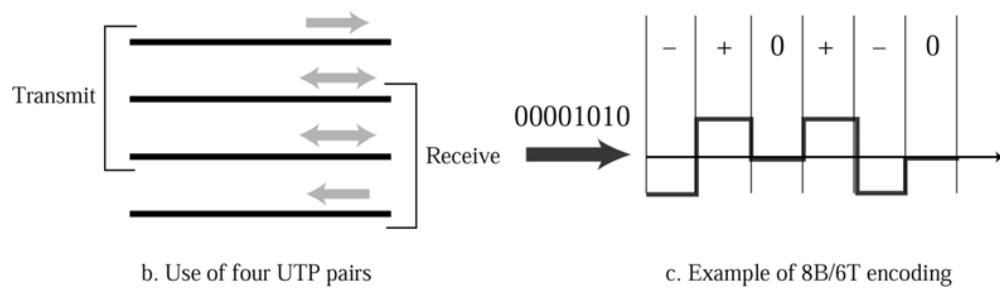
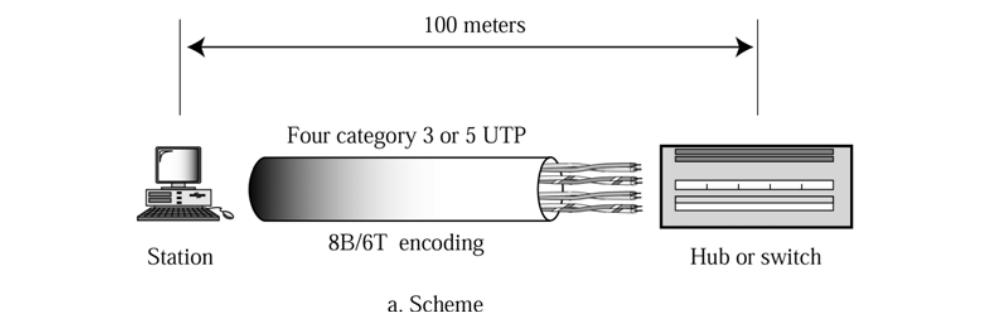
- Data rate can be increased by decreasing the collision domain
- If 10 Mbps  $\Rightarrow$  100 Mbps, then
  - 2500 m  $\Rightarrow$  250 m



- 100 Base-TX and 100 Base-FX**

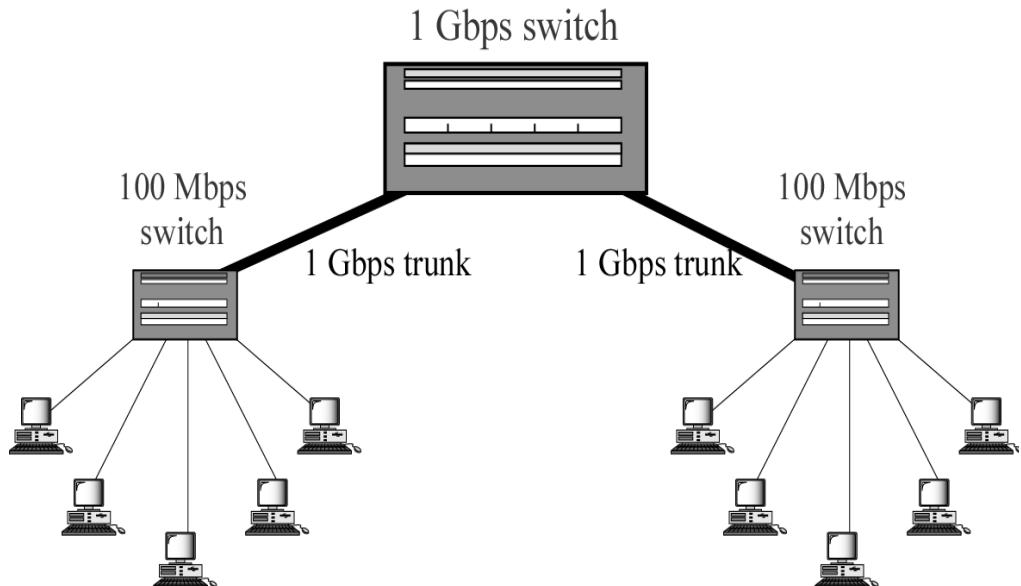


- 100 Base-T4**



## Gigabit Ethernet

- Usually serves as a backbone
- Four implementations
  - 1000Base-SX, 1000Base-LX: Optical fiber, 550-5000 m
  - 1000Base-CX (STP), 1000Base-T (UTP): 25 m



## Token Bus

- Other LANs are not suitable for this purpose
- Token Bus has no commercial application in data communications
- Token Ring allows each station to send one frame per turn
- Access method: Token passing

## Summary

- Ethernet
  - Implementations
- Other Ethernet Networks
- Token Bus/Ring

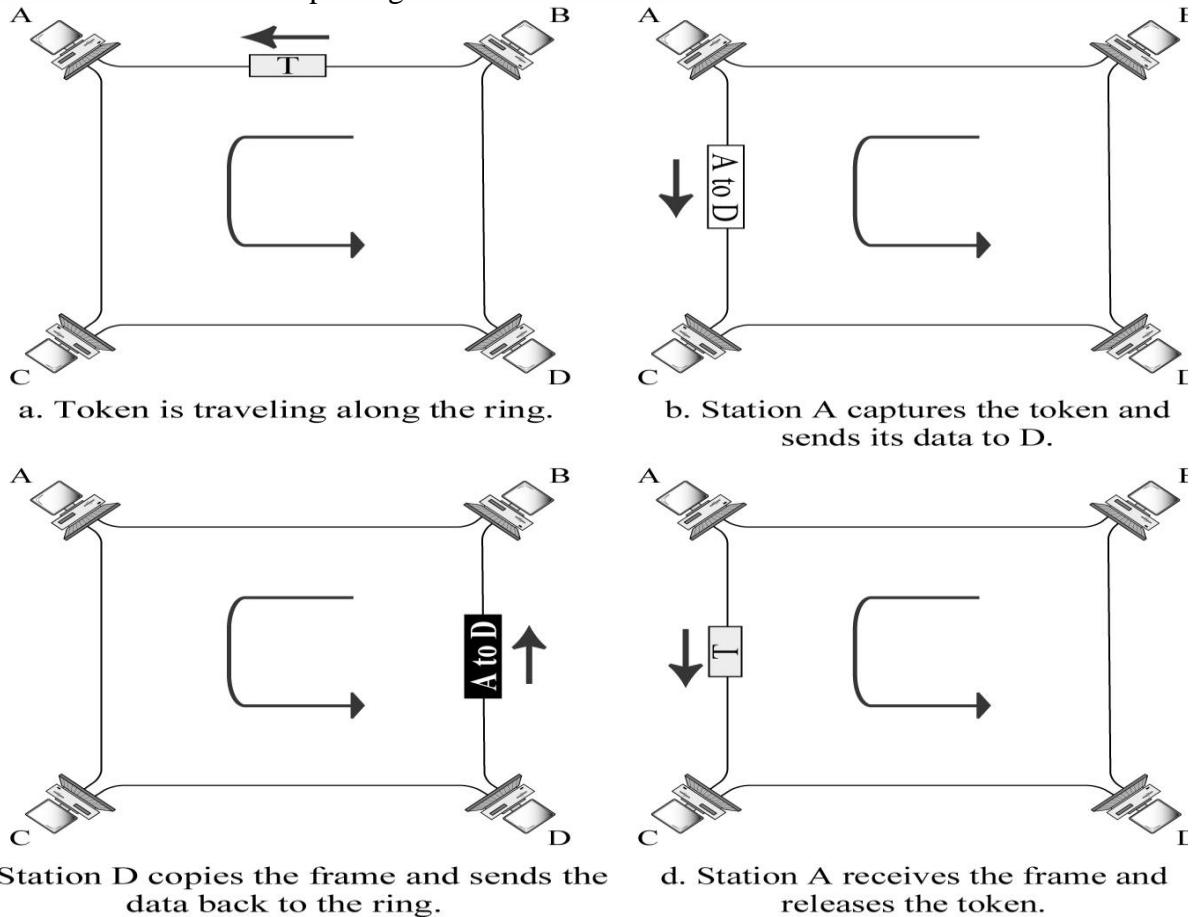
## Reading Sections

- Section 12.2, 12.3, 12.4, 12.5
- “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #44**

## **Token Ring**

- Token Ring allows each station to send one frame per turn
- Access method: Token passing



## **Access Control of Token Ring**

Y Priority and reservation

—Each station has a priority code

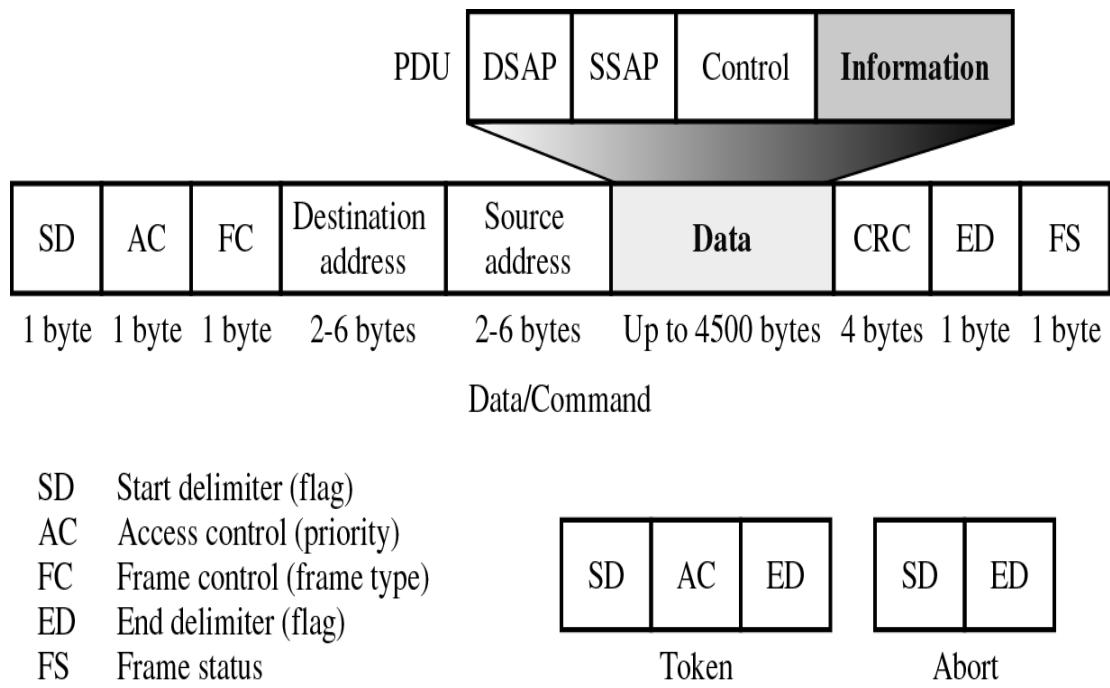
Y Time limits

—Token Ring imposes a time limit to keep traffic moving

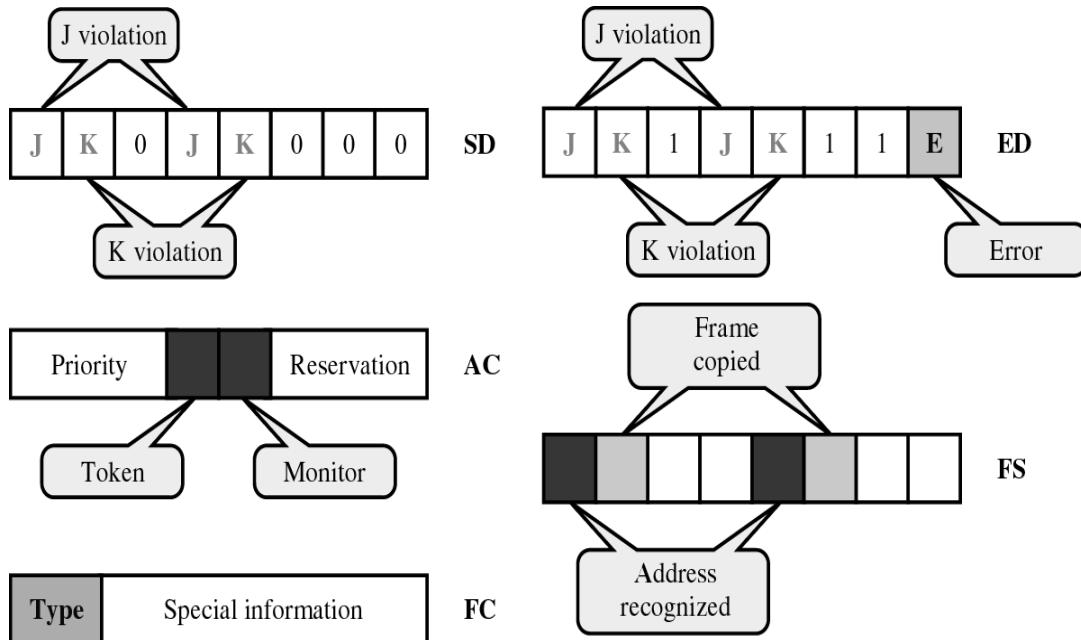
- **Monitor Stations**

- One station is designated as a monitor station to handle several problems
- Monitor station generates a new token when it is lost
- Monitor station removes recirculating data frames

## **Token Ring Frame**

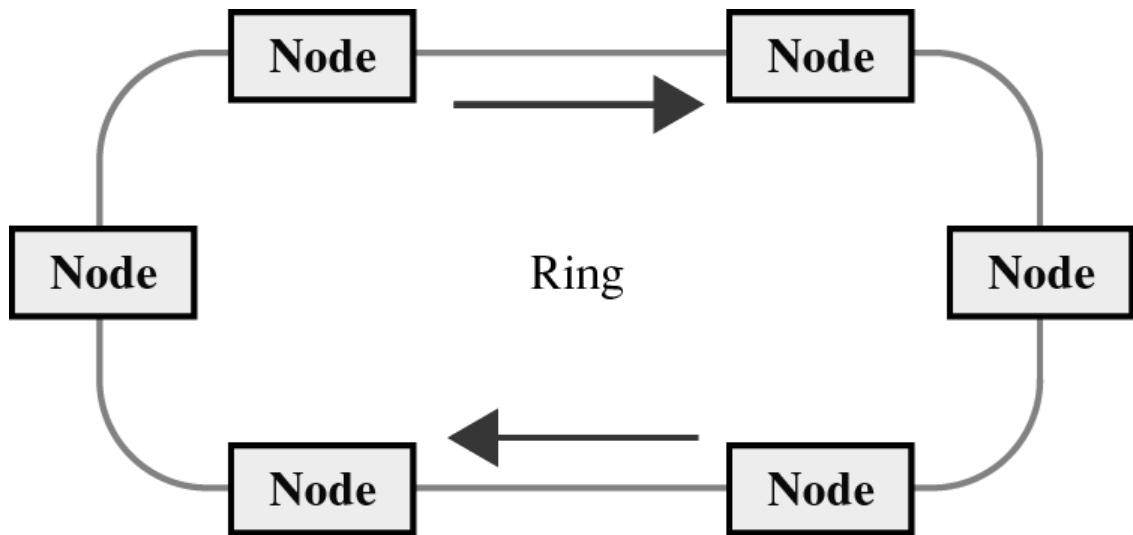


## Data Frame Fields

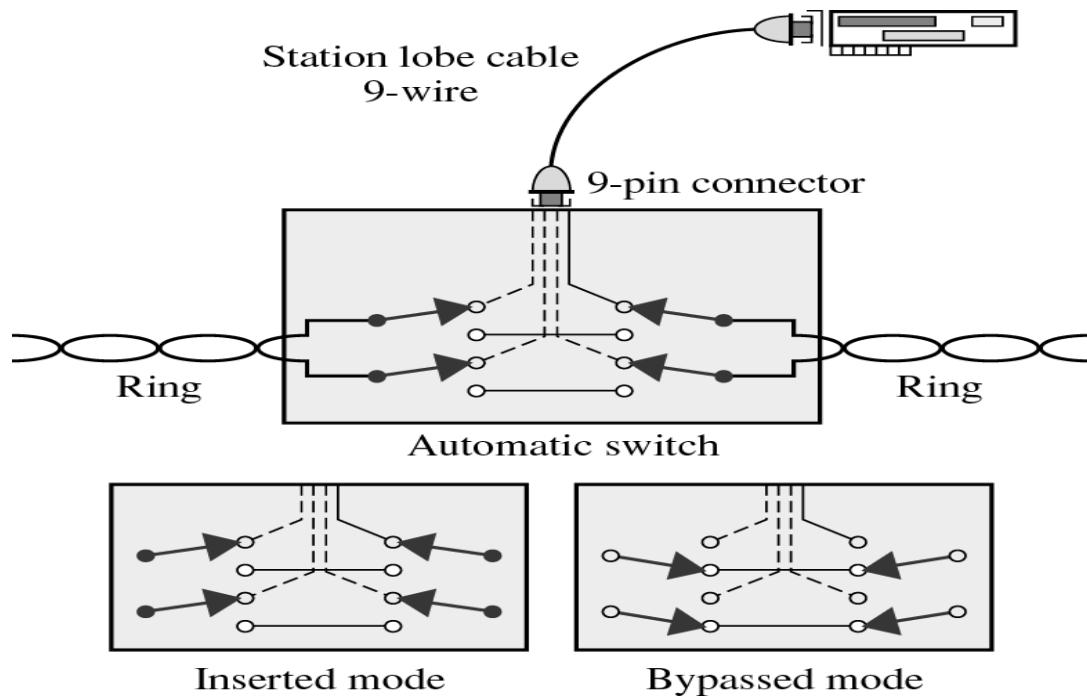


## Implementation

- Each station in the Token Ring regenerates the frame
- One disabled or disconnected node could stop the traffic flow around the entire network
  - Each station is connected to an automatic switch

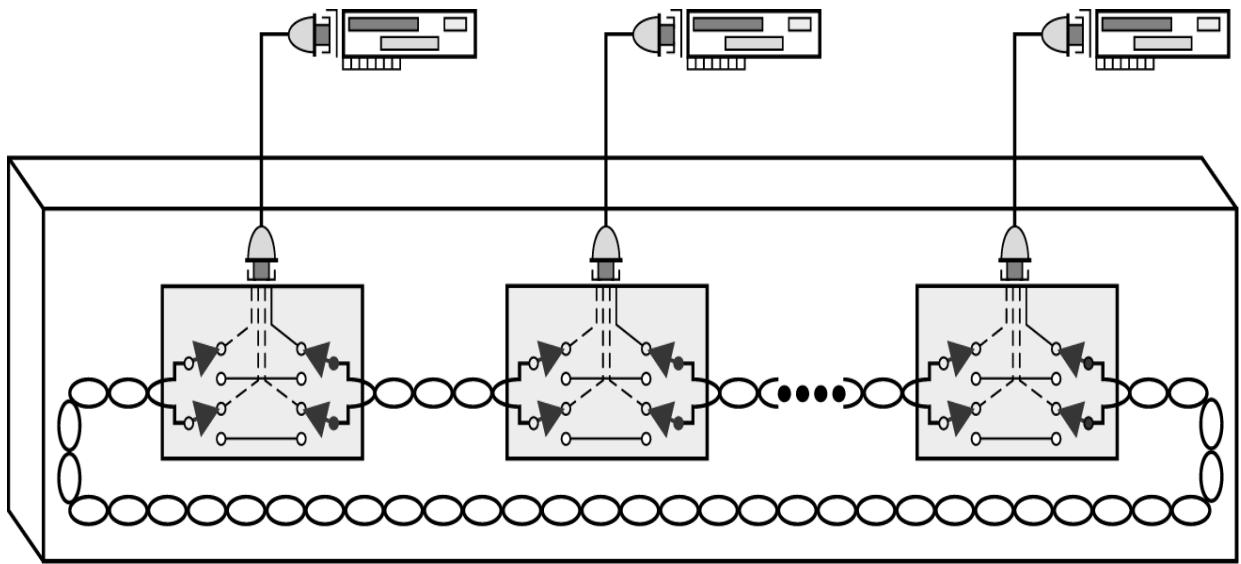


**Token Ring Switch**



#### Y Multi station Access Unit (MAU)

- o Individual automatic switches are combined in to a hub
- o One MAU can support up to 8 stations
- o Although it looks like a star, it is in fact a ring



Multistation access unit  
MAU

- **FDDI**

- o Fiber Distributed Data Interface standardized by ANSI and the ITU-T
- o 100 Mbps LAN protocol
- o CDDI: Copper version of FDDI
- o Access method: Token passing

#### Y FDDI Time Registers

Time registers

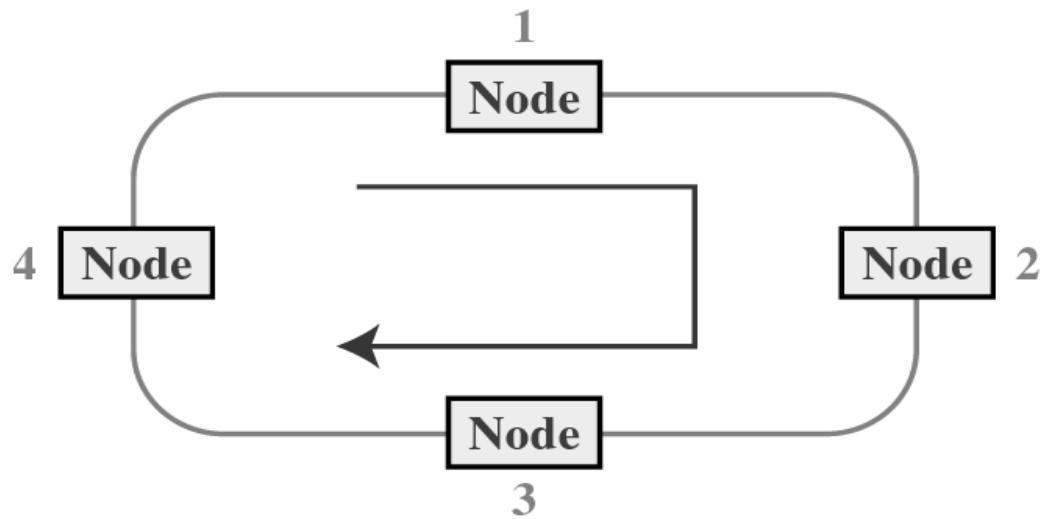
- Values are set when the ring is initialized and do not vary
- SA (Synchronous Allocation)
- TTRT* (Target Token Rotation Time)
- AMT* (Absolute Maximum Time)

#### Y FDDI Timers

Timers

- Each station contains two timers
- TRT* (Token Rotation Timer) : Incrementing
- THT* (Token Holding Timer) : Decrementing

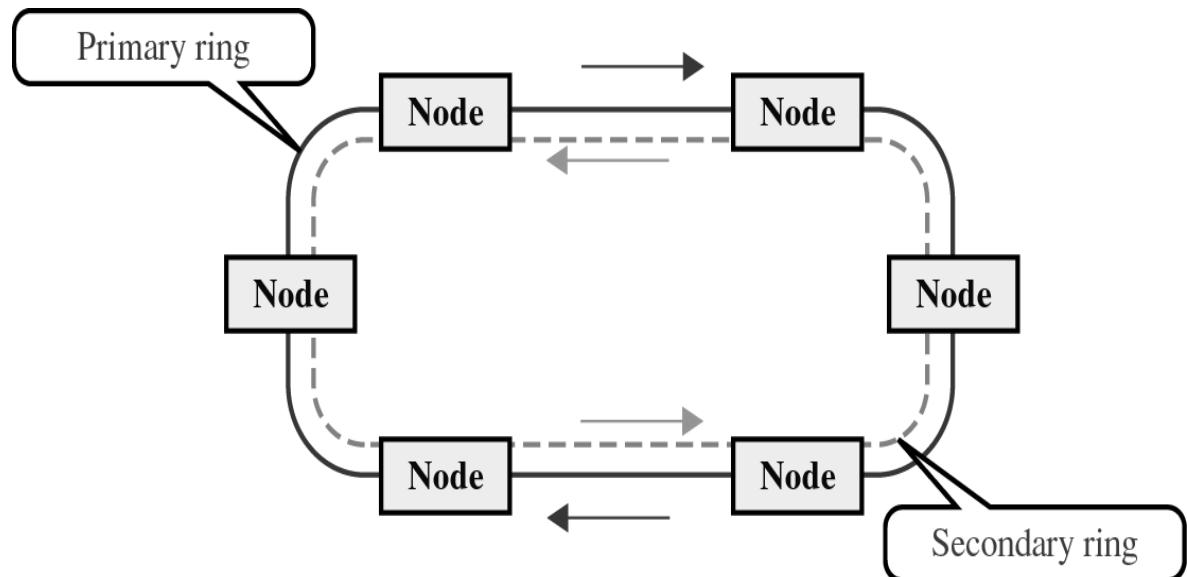
## Station Procedure



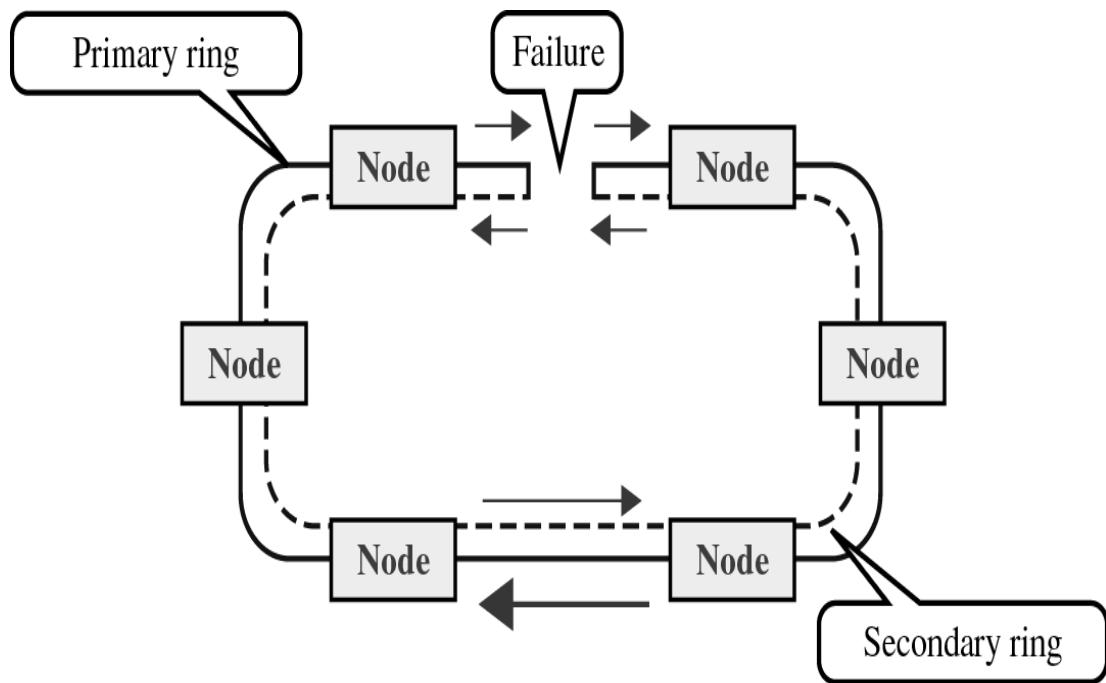
## Station Procedure

- THT is set to the difference between TTTRT and TRT
- $THT = TTTRT - TRT$
- TRT is reset to zero ( $TRT = 0$ )
- The station sends S-frames during the time in SA
- The station sends A-frames as long as  $THT \geq 0$
- Release the token

## Dual Ring (Figure-1)



## Dual Ring (Figure-2)



### Summary

- Token Ring
- FDDI

### Reading Sections

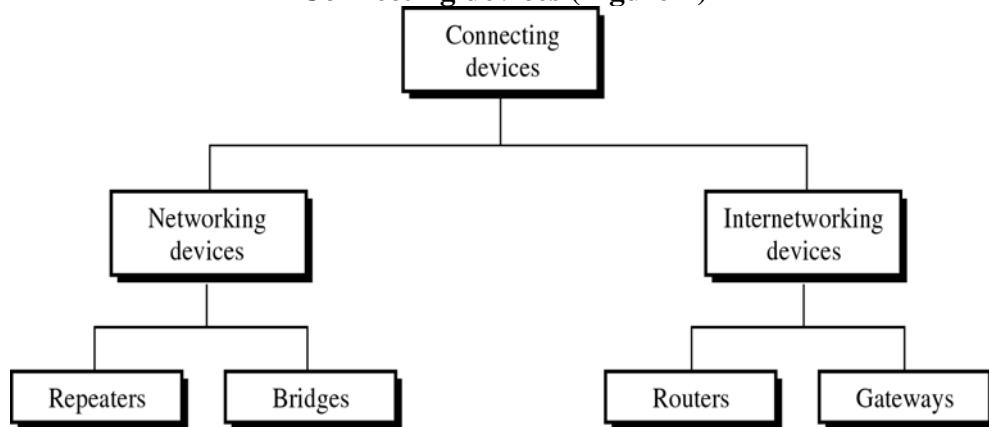
- Section 12.5, 12.6, 12.7
- “Data Communications and Networking” 4th Edition by Behrouz A. Forouzan

# **LECTURE #45**

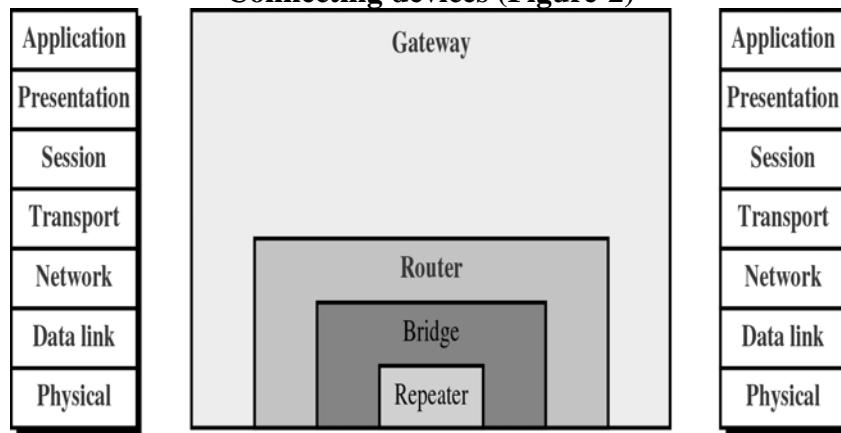
## **Internet**

- An *internet* is a generic term used to mean an interconnection of individual networks
- To create an internet, we need networking devices called routers and gateways
- An internet is different from the Internet
- Internet is the name of a specific worldwide network

**Connecting devices (Figure-1)**

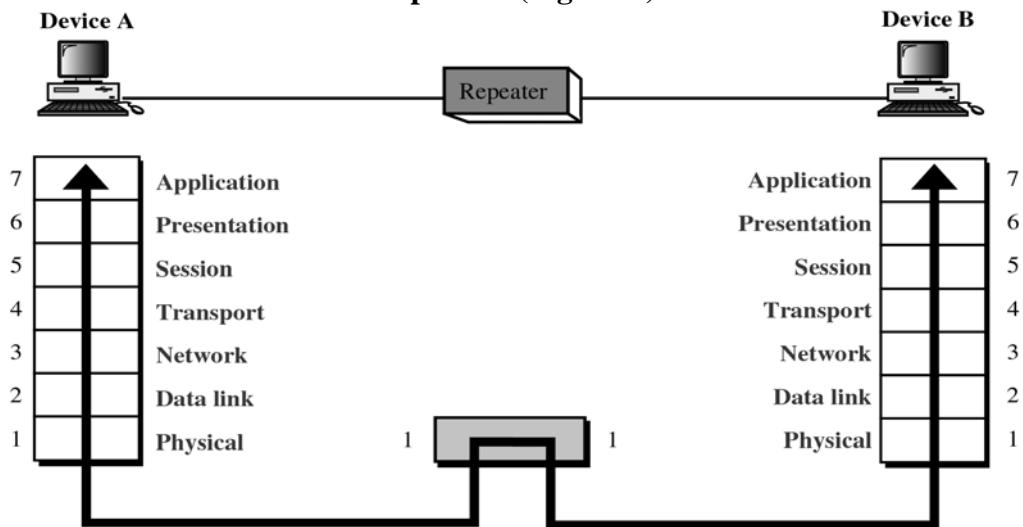


**Connecting devices (Figure-2)**



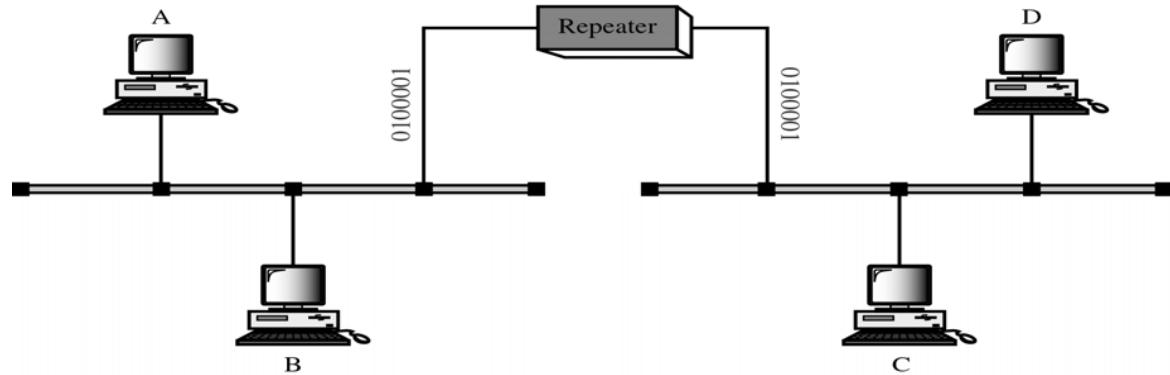
## Repeaters

**Repeaters (Figure-1)**

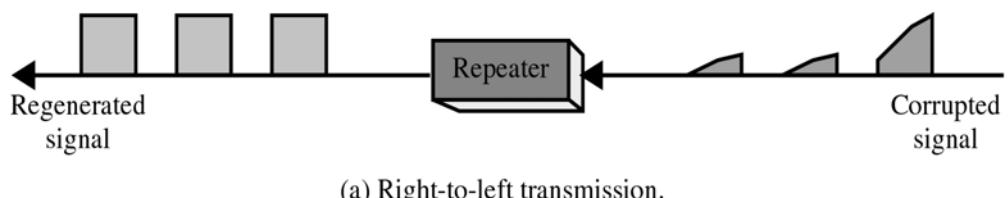


- A repeater allows us to extend only the physical length of a network
- Repeater is a regenerator, not an amplifier

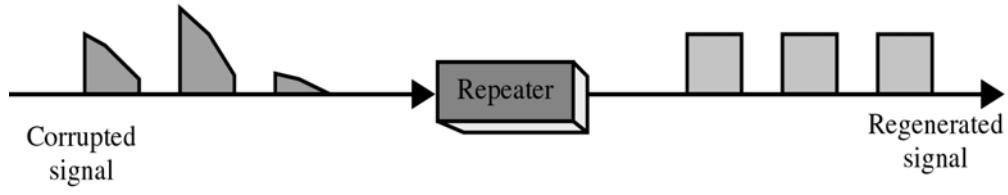
**Repeaters (Figure-2)**



**Repeaters (Figure-3)**

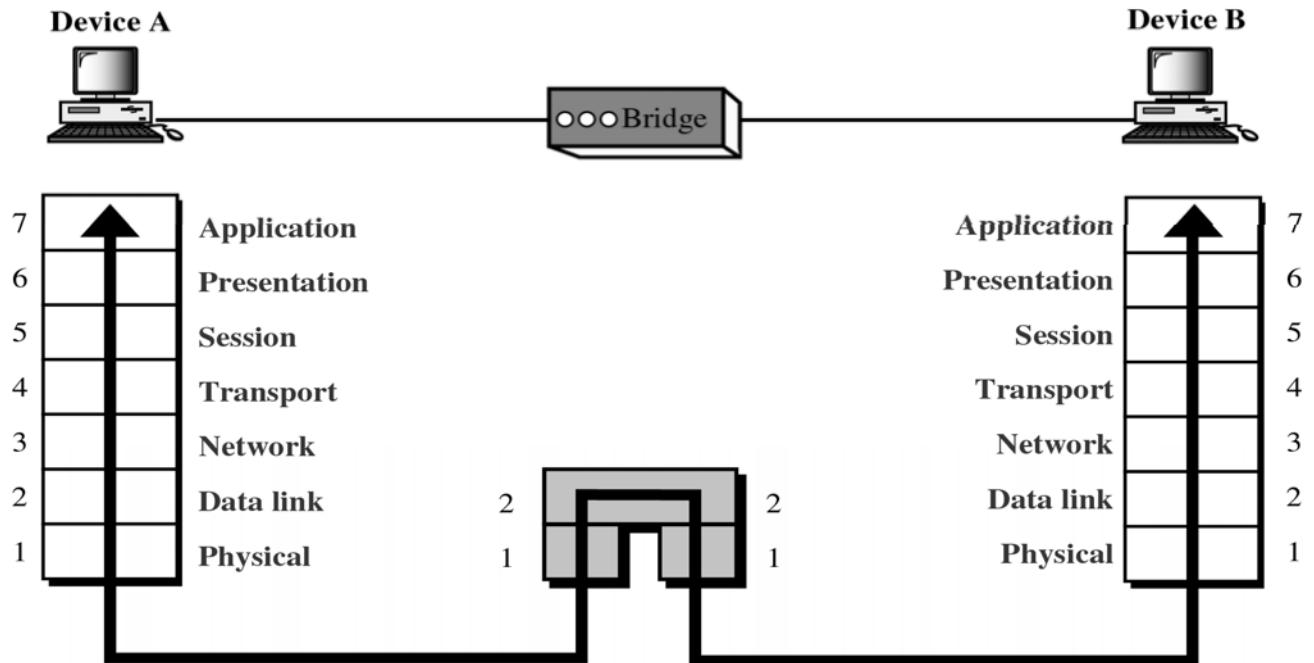


(a) Right-to-left transmission.

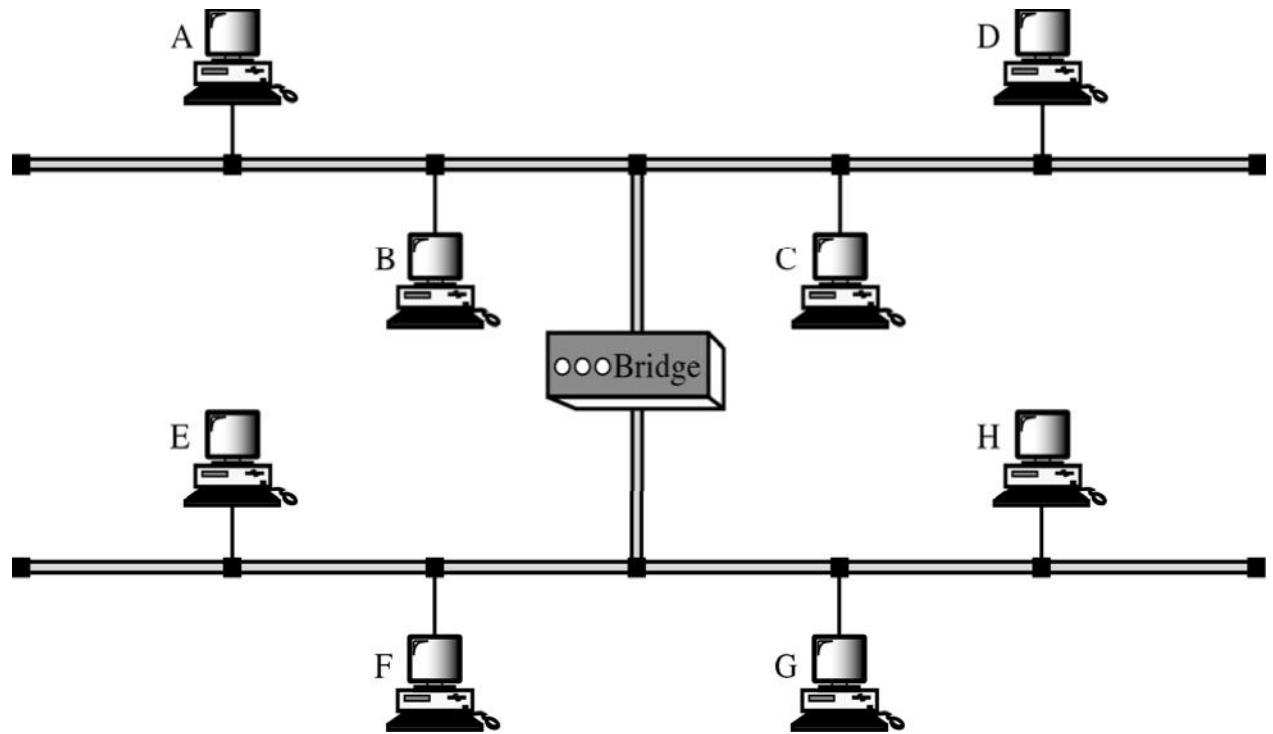


(b) Left-to-right transmission.

## Bridges

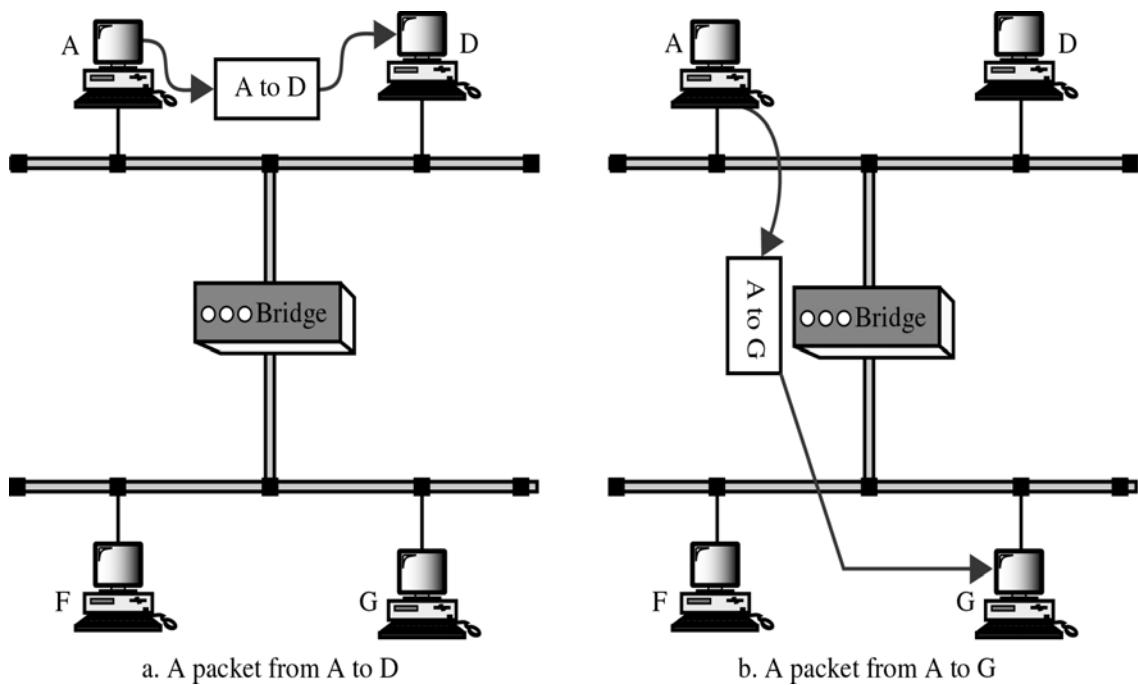


- Bridges can divide a large network into smaller segments

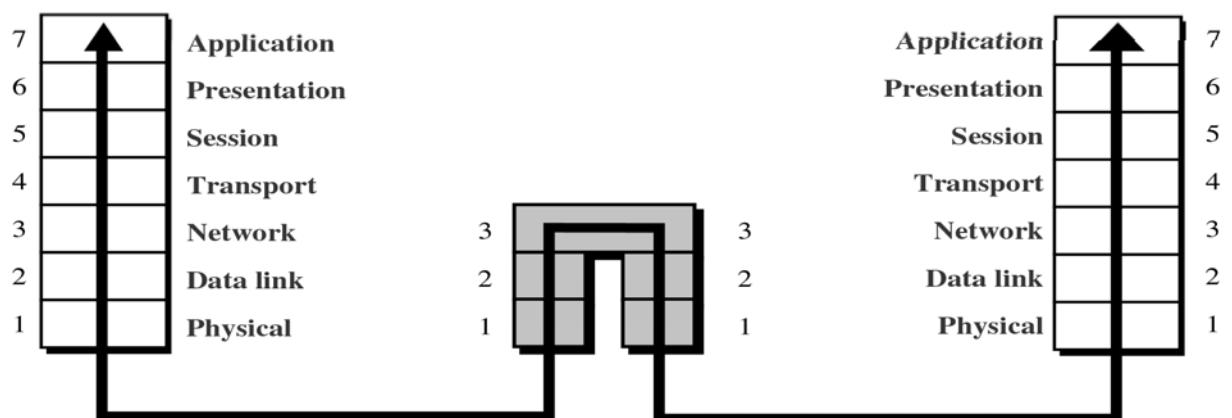


When a frame enters a bridge, the bridge:

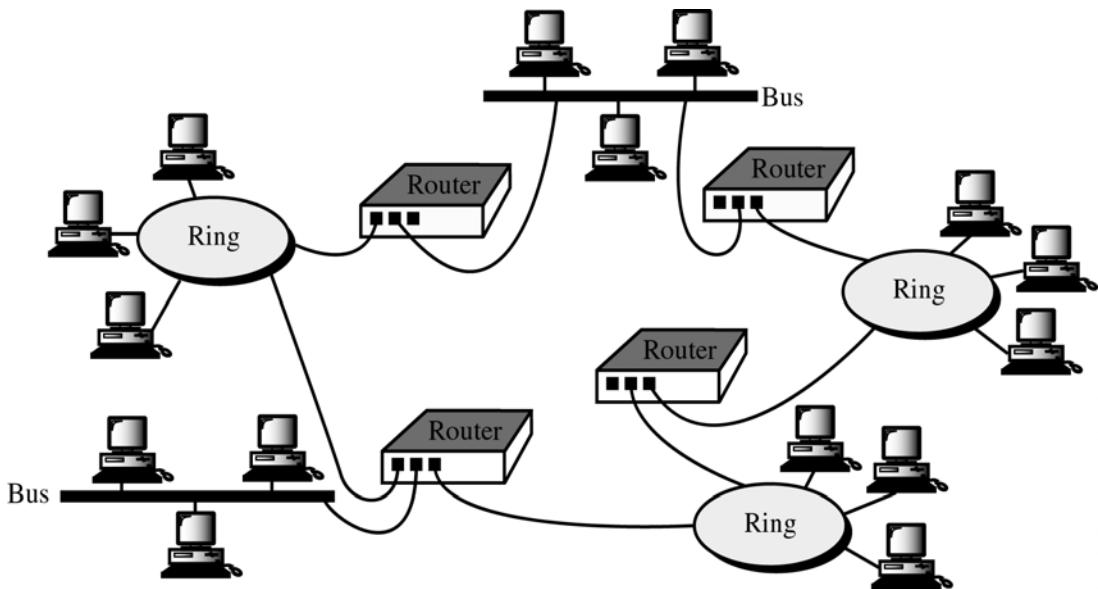
- not only regenerates the signal
- but check the address of the destination and forwards the new copy only to the segment to which the address belongs



## Routers

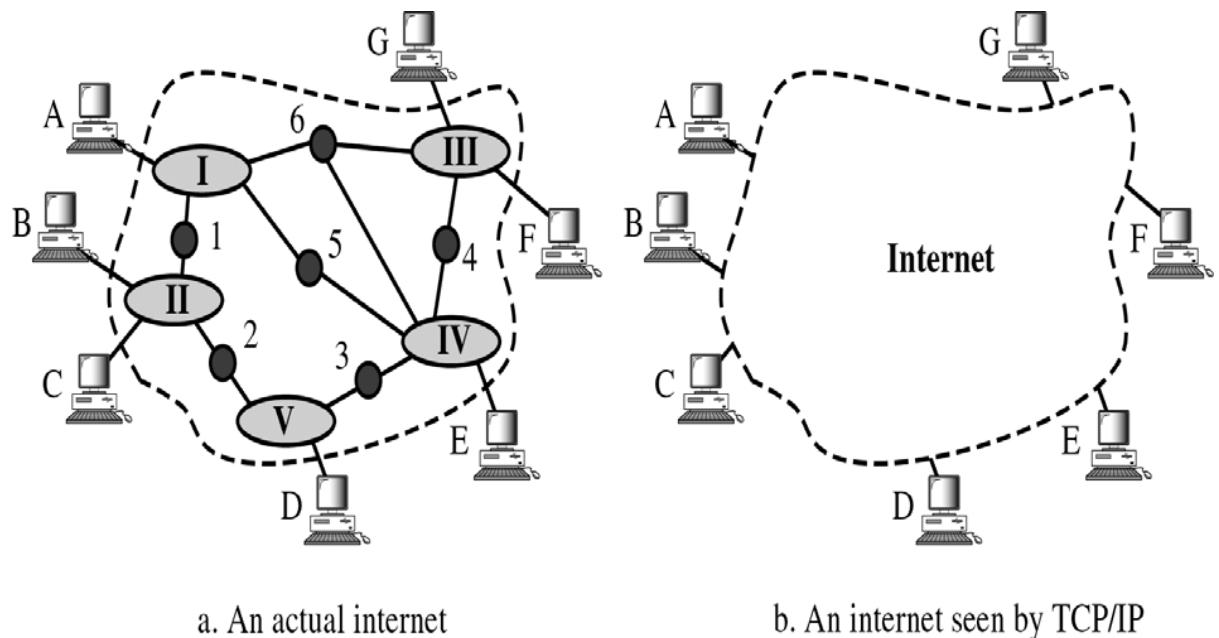


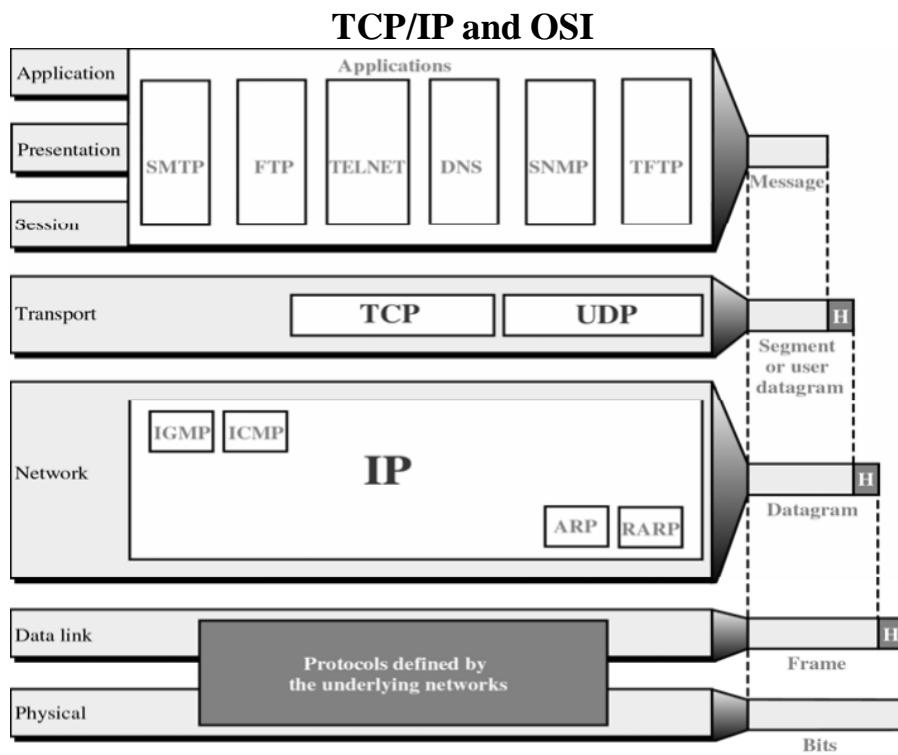
Routers relay packets among multiple interconnection networks



### TCP/IP

- Transmission Control Protocol/ Internet Protocol
- 1969
- ARPA(Advanced Research Project Agency) net





### **TCP/IP Protocols**

- Network Layer
  - IP (Internetworking Protocol )
  - ARP(Address Resolution Protocol)
  - RARP(Reverse Address Resolution Protocol)
  - ICMP(Internet Control Message Protocol)
  - IGMP(Internet Group Message Protocol)
- Transport Layer
  - TCP (Transmission Control Protocol)
  - UDP (User Datagram Protocol)

### **Summary**

- Internet work
- Connecting Devices
  - Repeaters
  - Bridges
  - Routers
- TCP/IP Protocol Suite

### **Reading Sections**

- Section 21.1, 21.2, 21.3, 24.1, 24.2,  
“Data Communications and Networking” 4th Edition by Behrouz A. Forouzan