

Communication between more
than two devices sharing a link

Why Share?

- Creating separate physical link between each transmitter-receiver pair is infeasible
- In most cases, a single transmitter-receiver pair does not utilize the whole bandwidth of a link
- Bandwidth can be shared between multiple pairs to increase line efficiency and reduce cost
- Approaches
 - Multiplexing
 - Create multiple “logical” channel out of a single physical channel
 - Contention based protocols
 - Contend for use of channel when needed, no a-priori agreement or permission given
 - Controlled access protocols
 - Agree beforehand who has right to send when
 - Ex. — Reservation, Polling, Token Passing

Multiplexing

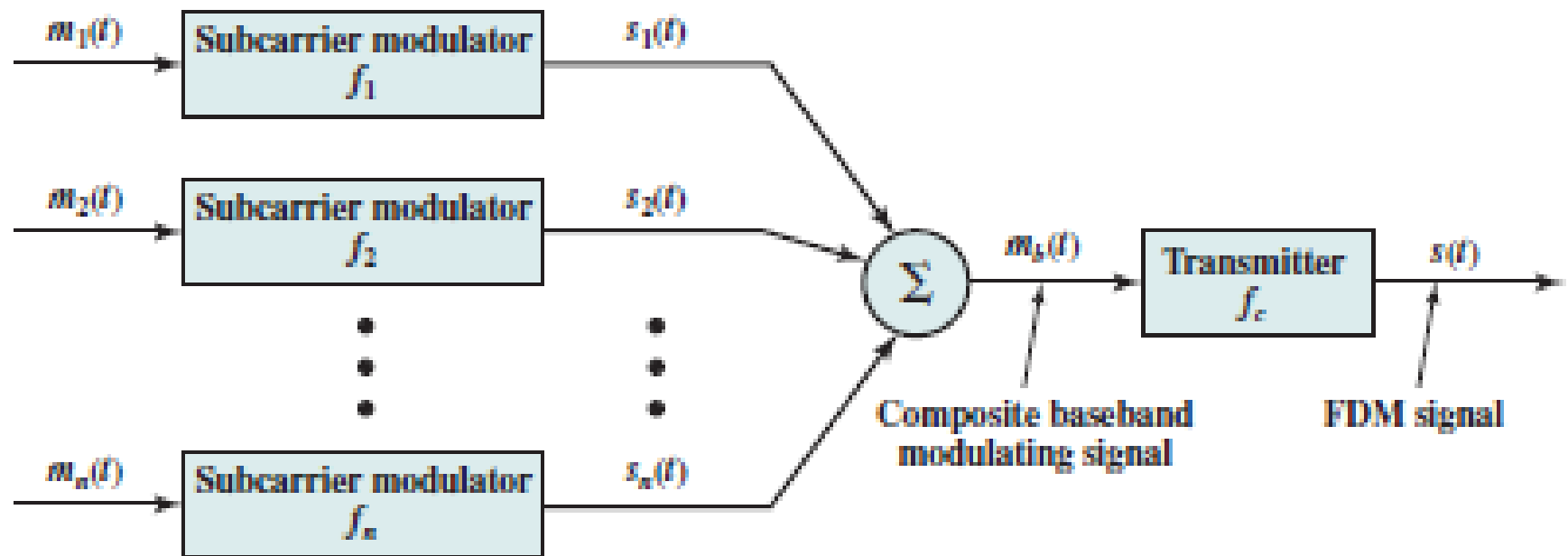
- Create more than 1 channel out of a link
- Each channel has a “share” of the link (in time, frequency, wavelength etc.)
 - Time Division Multiplexing (TDM) – each channel assigned a time slot
 - Frequency Division Multiplexing (FDM) –each channel assigned a frequency band
 - Wavelength Division Multiplexing (WDM) – each channel assigned a wavelength (optical communication only)
- Different sender-receiver pairs can use the different channels



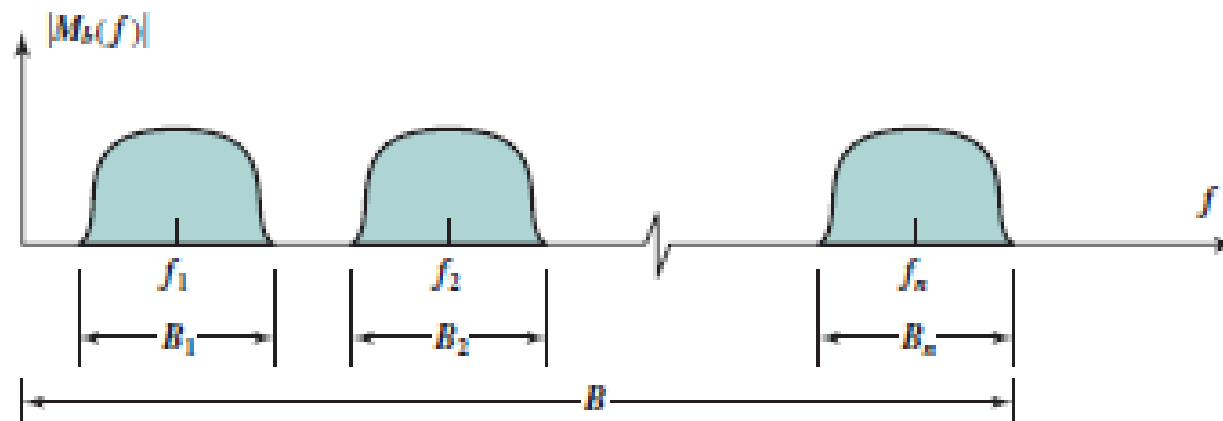
Figure 8.1 Multiplexing

Frequency Division Multiplexing (FDM)

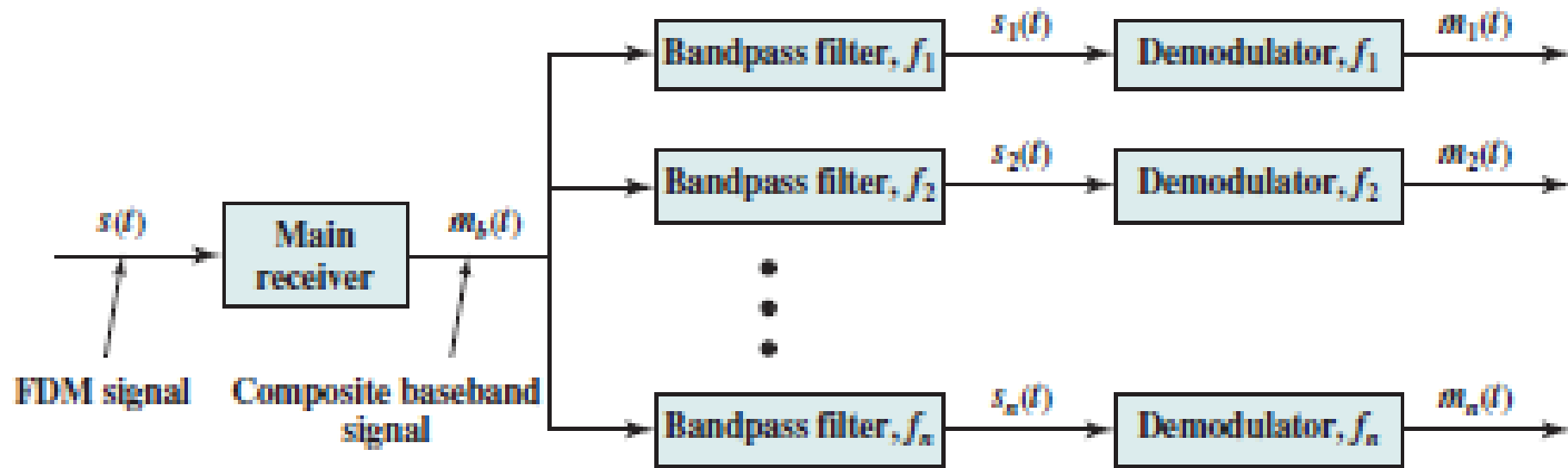
- Possible when useful bandwidth of medium exceeds required bandwidth of channel
- Each signal is modulated to a different carrier frequency
 - So a number of carrier signals used
 - Each called a subcarrier
- Sum the resulting analog, modulated signals to get a composite baseband signal
- Carrier frequencies separated by **guard bands** so signals do not overlap
- Channel allocated even if there is no data



(a) Transmitter



(b) Spectrum of composite baseband modulating signal

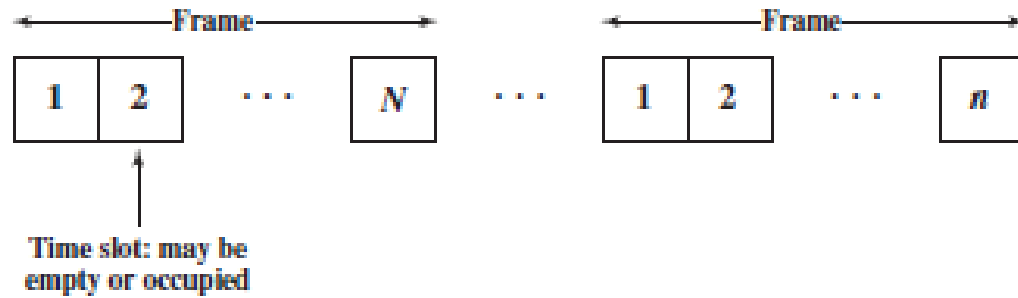
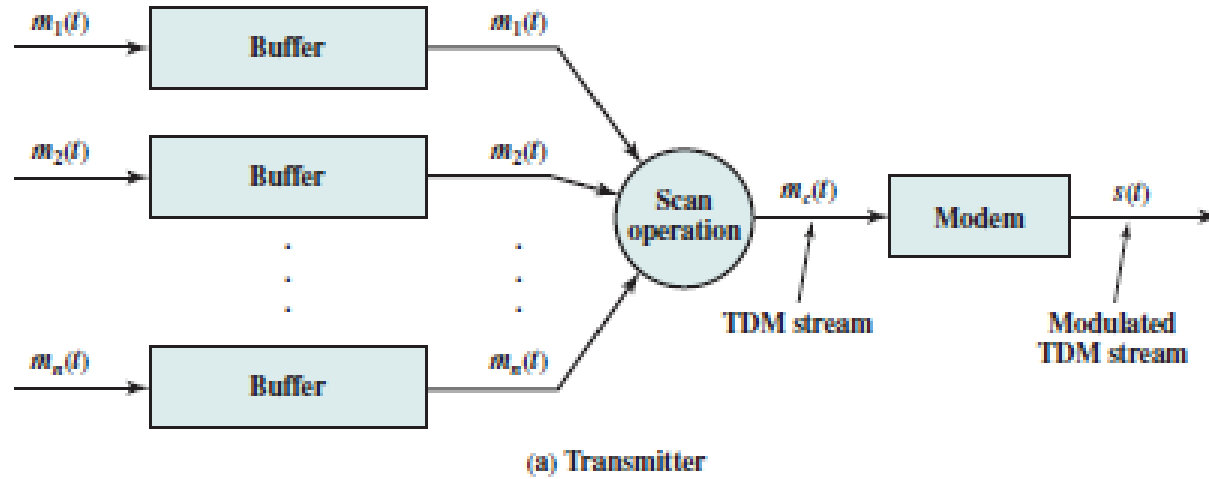


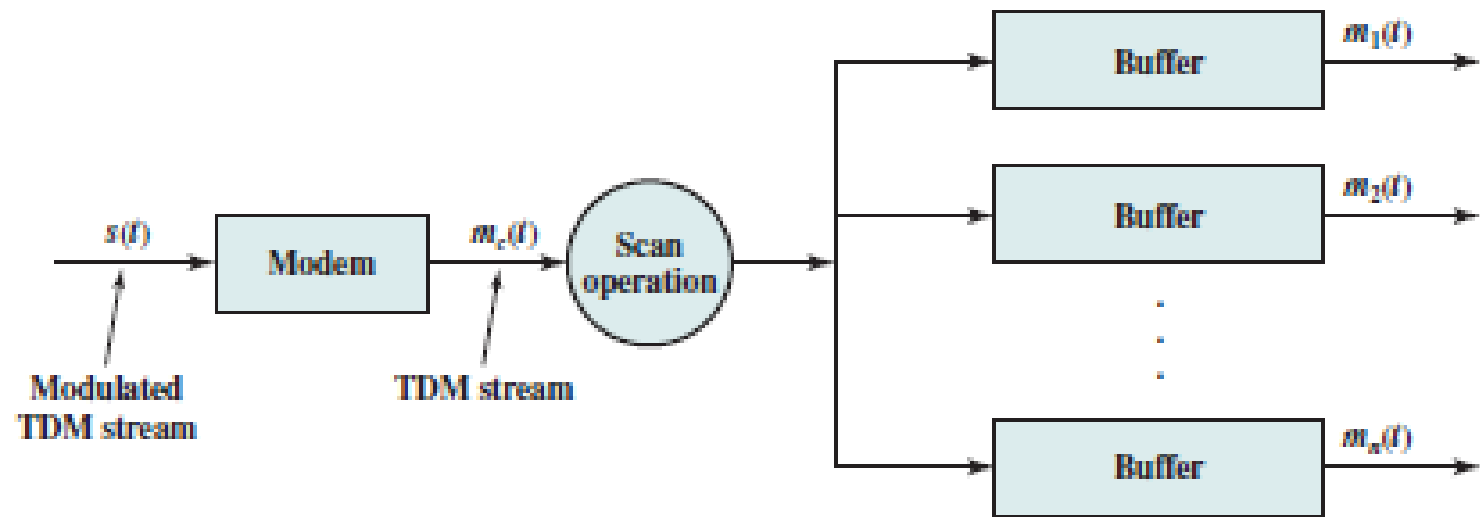
(c) Receiver

- Requires analog signaling & transmission
- Bandwidth = sum of subcarriers + guard bands
- Modulates signals so that each occupies a different frequency band
- Standard for radio broadcasting, analog telephone network, and television (broadcast, cable, & satellite)

Synchronous TDM

- Multiple digital signals interleaved in time
- TDM frame contains time slots
- Time slots pre-assigned to sources and fixed
- Multiplexer scans each source and puts its data in its time slot
- If there is no data for a source, the slot goes free
- Each slot contains 1 bit (*bit multiplexing*) or 1 byte (*byte multiplexing*) or blocks of bytes....
 - Can be anything fixed
- TDM frames transmitted using synchronous or asynchronous transmission





(c) Receiver

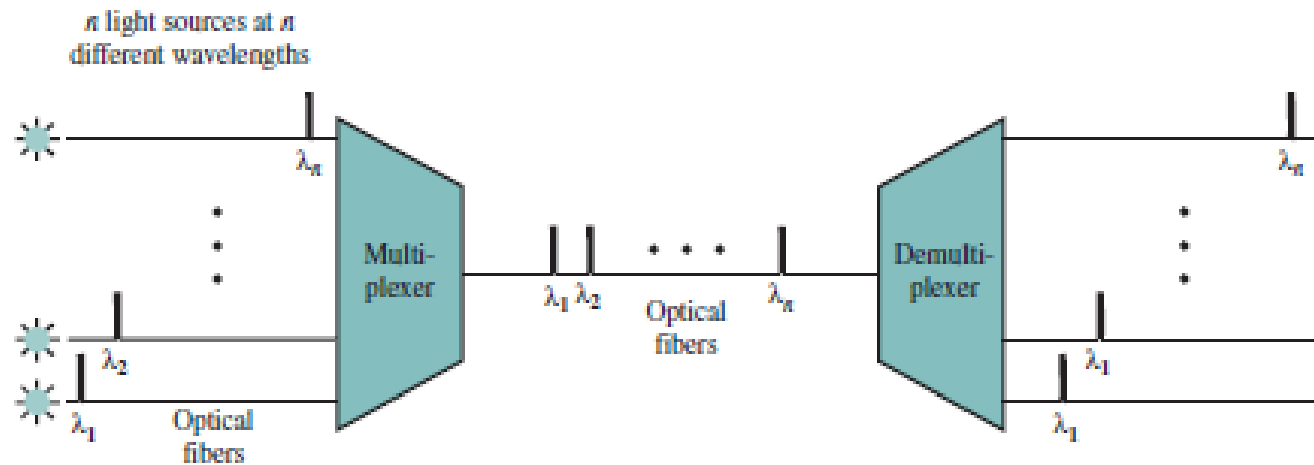
- Synchronization between mux and demux
 - Extra framing bit transmitted with each frame for clock synchronization
 - Framing bits define a *control channel*
- Requires data rate of the medium to be greater than sum of the data rates of signals to be transmitted

Statistical TDM

- In Synchronous TDM many slots are wasted if there is no data to be transmitted in that channel
- Statistical TDM allocates time slots dynamically based on demand
- Data rate capacity required is below the sum of connected capacity
- Frames are more complex, and so is the mux and demux

Wavelength Division Multiplexing (WDM)

- Used only for optical channels
- Each channel is modulated on carriers of a different wavelength (multiple beams of light)



Contention Based Protocols

- Commonly called *Medium Access Control (MAC) Protocols*
- Defines protocols for multiple stations to access medium at multiple points and successfully send data
 - Needs to avoid collision, i.e. two or more stations accessing medium at the same time

ALOHA

- Developed in Univ. of Hawaii in early 70s
 - Interconnected the different islands
- For packet radio networks
- Transmission to and from a central station
- All sources transmit using same frequency, can transmit anytime
- If two sources transmit simultaneously, collision
- If collision, each station waits for a random time, then tries again

- How is collision detected?
 - Acknowledgement frames sent by receiver on a different frequency
 - If no ack received within some time, sender assumes collision
- Simple
- Very inefficient for larger no. of nodes or higher transmission rates

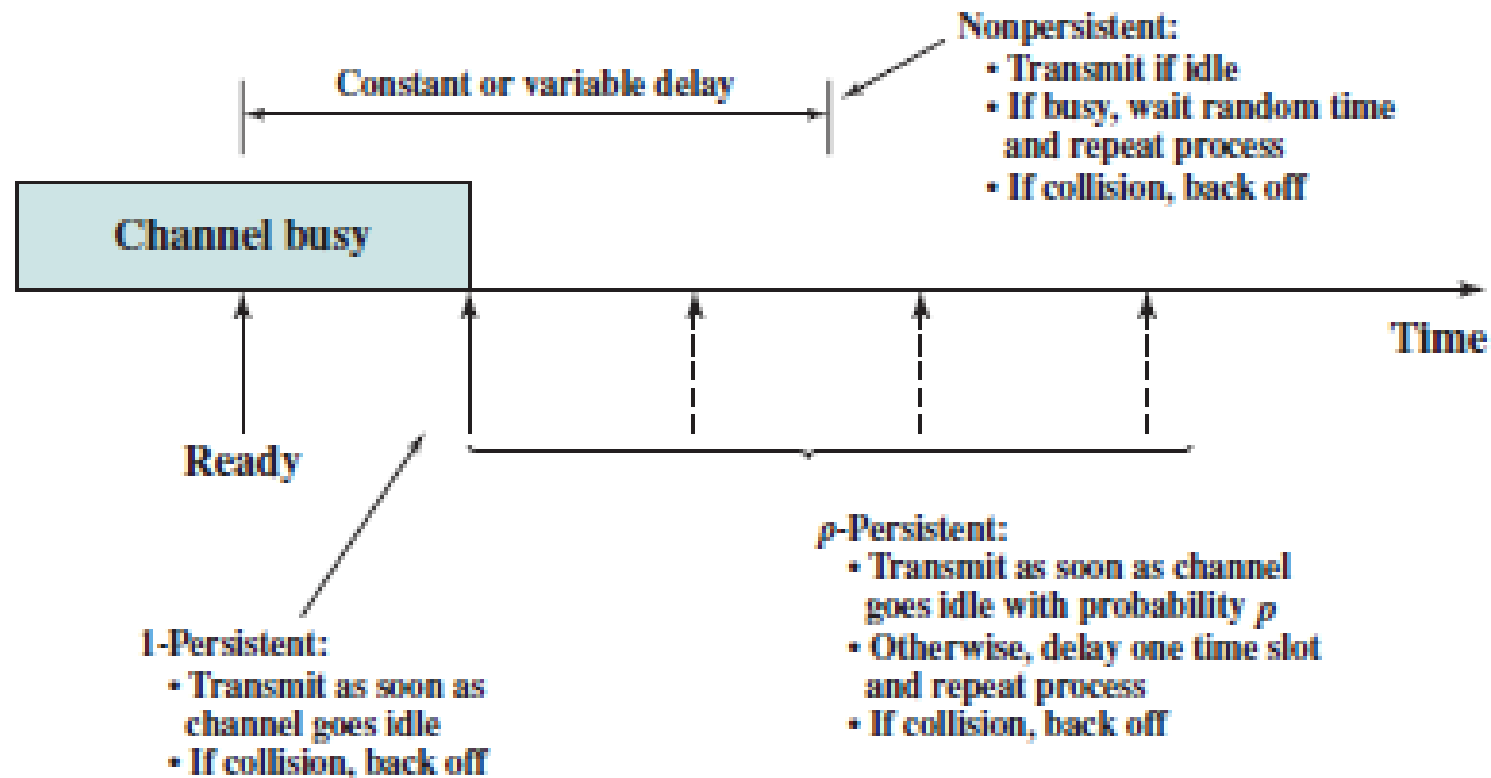
Slotted ALOHA

- Time divided into slots
- Each source should start sending only at the beginning of a slot
- Collision possible only if two sources become ready at the same slot.
- Reduces no. of collisions over ALOHA (why?)

Carrier Sense Multiple Access (CSMA)

- Broad Idea
 - Sense the medium to see if it is used (Carrier Sense)
 - If not (line idle), transmit
 - If yes, wait for some time, then repeat the above
- Types - nonpersistent, 1-persistent, p-persistent
 - Differs mostly in what a sender does when the medium is found to be busy
 - Tradeoff between line utilization and chance of collision

- Non-persistent CSMA
 - If medium is found idle, transmit
 - If medium found busy, wait for a random time drawn from a probability distribution, then sense line again
- 1-persistent CSMA
 - If medium is found idle, transmit
 - If medium found busy, continue to listen until the channel is sensed idle, then transmit immediately
- p-persistent CSMA
 - If medium is found idle, transmit with probability p , and delay 1 time unit with probability $(1 - p)$
 - If delayed, repeat this step after the delay
 - If medium found busy, continue to listen until the channel is sensed idle, then use the rule for sending in idle channel
 - What is a good value of p ?



CSMA/CD (CSMA with Collision Detection)

- Why needed?
 - Suppose two senders detect the channel idle at the same time and start to transmit
 - Both will end up transmitting a full damaged frame
- CSMA/CD
 - Sense medium
 - If medium is idle, transmit
 - If medium is busy, apply standard CSMA (non-persistent, p-persistent,...)
 - If collision detected during transmission (how?), transmit a short jamming signal to indicate collision to all and stop transmitting
 - After sending the jamming signal, wait for a random backoff time (Binary exponential backoff), then repeat above steps

- Binary Exponential Backoff
 - Time divided into slots
 - After k-th collision, pick a slot randomly within the next 2^k slots to try transmitting again
 - So with increased congestion, transmitters choose from a larger range to reduce chance of collision
 - Usually gives up after a certain number of tries

- CSMA/CD requires a minimum packet length L to detect collision (why?)

$$L = 2DR/V \text{ where}$$

D = maximum distance between two machines

R = data rate

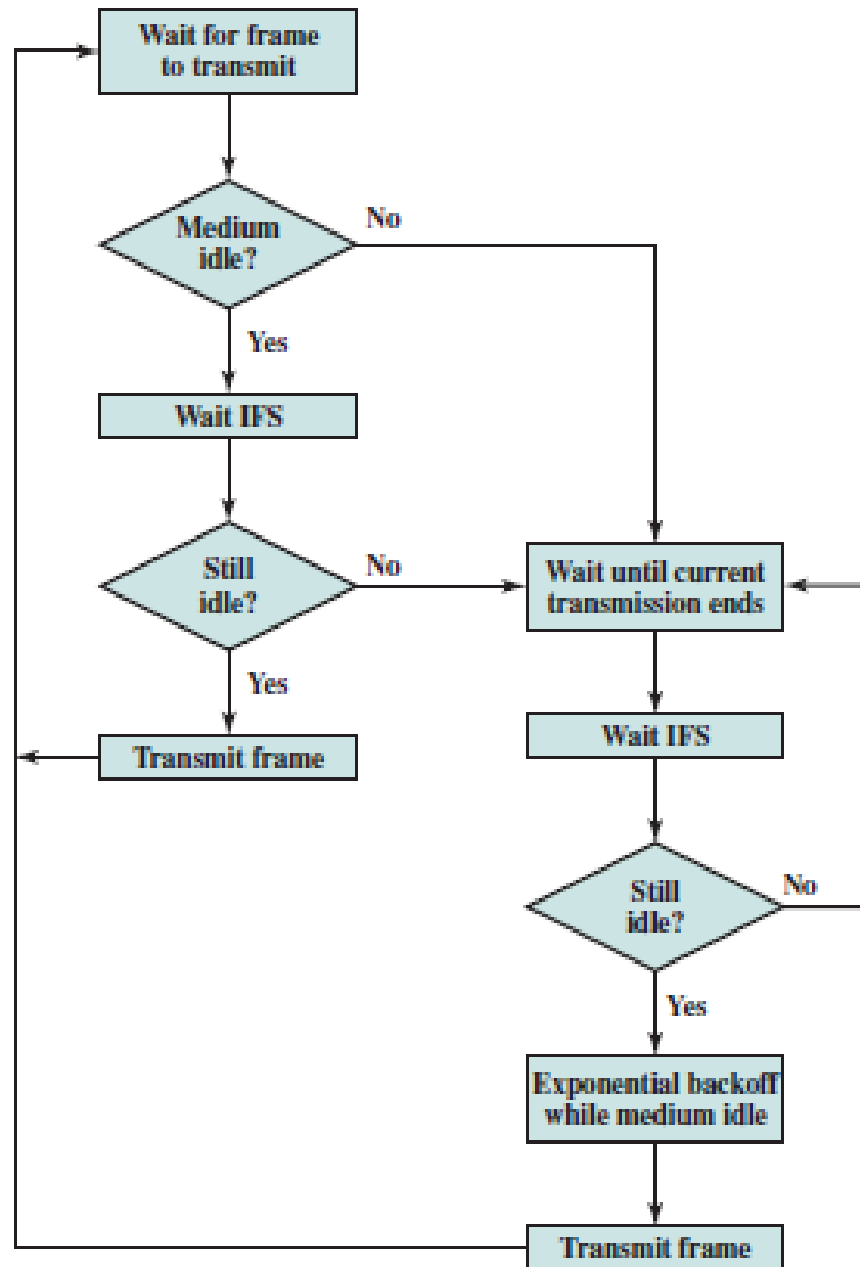
V = propagation delay

CSMA/CA (Collision Avoidance)

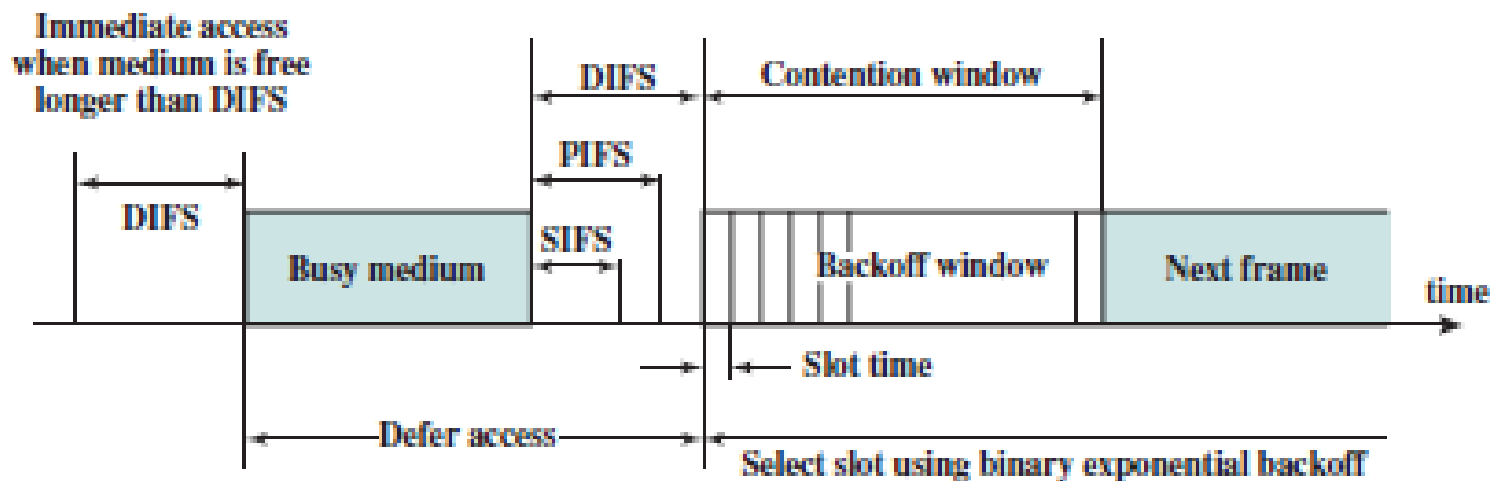
- Problems in collision detection in wireless media
 - Weak received signal
 - Hidden terminal problem
 - Exposed terminal problem
- Collision Avoidance proposed as a solution
- In wireless, a receiver can get a sender's signal only if it is within a certain range of the sender
 - Range depends on different things like transmitter power, frequency range used, atmospheric conditions etc.

Basic Transmission Steps

- Sense medium. If idle, wait for IFS (Inter Frame Space) time. If still idle, transmit.
- If medium is busy, defer transmission and monitor medium
- When medium found free again
 - Defer another IFS time
 - If still idle backoff, a random time and again sense medium
 - If still idle, transmit
 - During backoff time, if medium is found busy, halt the timer, resume when medium is free again
- If no ack received, assume collision



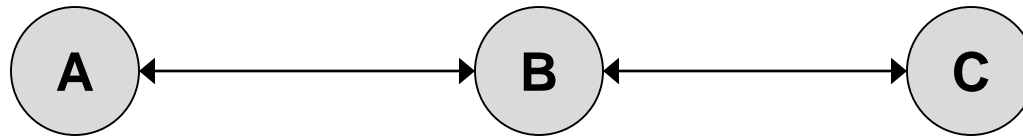
- Different IFS lengths to prioritize different types of processes



(a) Basic access method

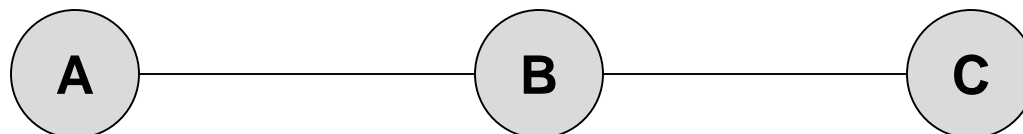
- Problems: Hidden Terminals and Exposed Terminal

Hidden Terminal Problem



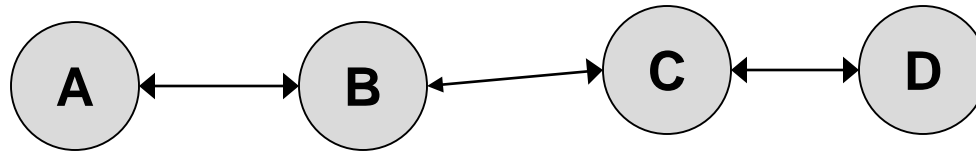
- B can communicate with both A and C
- A and C cannot hear each other (out of range)
- Problem
 - When A transmits to B, C cannot detect the transmission using the carrier sense mechanism
 - If C transmits, collision will occur at node B
- Solution
 - Hidden sender C needs to defer

Solution for Hidden Terminal Problem



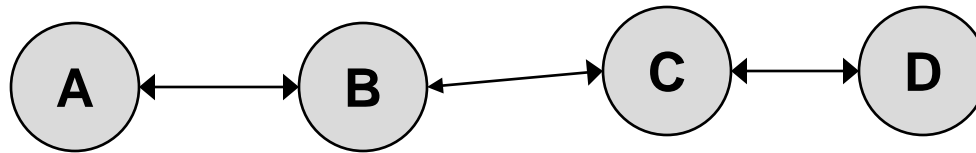
- When A wants to send a packet to B, A first sends a **Request-to-Send (RTS)** to B
- On receiving RTS, B responds by sending **Clear-to-Send (CTS)**
- When C overhears a CTS, it keeps quiet for the duration of the transfer
 - Transfer duration is included in both RTS and CTS
- But RTS/CTS are also frames and they can also collide! Why is this not a problem?

Exposed Terminal Problem



- B can communicate with A and C
- C can communicate with B and D
- A and C cannot hear each other
- Problem
 - When B transmits to A, C detects the transmission using the carrier sense mechanism
 - So C defers transmitting to D
 - But C could have sent to D, so blocked unnecessarily (C is the exposed terminal)

Solution to Exposed Terminal Problem



- B sends **RTS** to A
- A sends **CTS** to B
- C hears **RTS**, but not **CTS**, assumes it is ok to send to D
- Does it fully solve the problem?

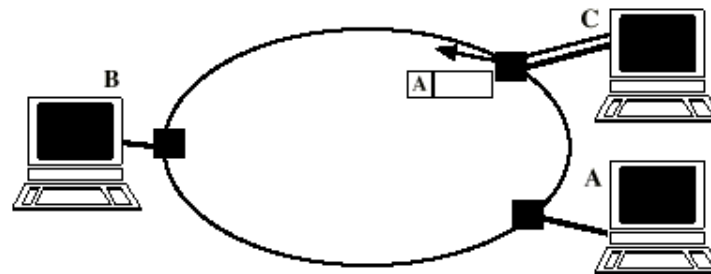
Token Ring

- Repeaters joined by point to point links in closed loop
 - Receive data on one link and retransmit on another
 - Links unidirectional
 - Stations attach to repeaters
- Data in frames
 - Circulate past all stations
 - Destination recognizes address and copies frame
 - Frame circulates back to source where it is removed
- Media access control determines when station can insert frame

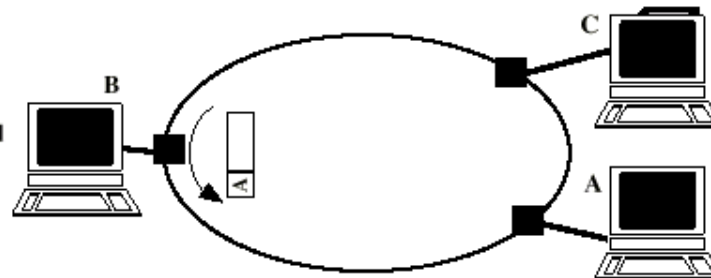
Token Passing

- Special frame called Token
- A m/c can transmit only when it has the token
- If no transmission needed, pass token to neighbor
- If data to transmit, put data and destination in token, change control bit to indicate data frame, and pass to neighbor
- When frame received, if data frame and not destination, pass to neighbor. If data frame and destination, copy and pass to neighbor
- When sender receives back, it changes control bits to make it a token again, and passes to neighbor

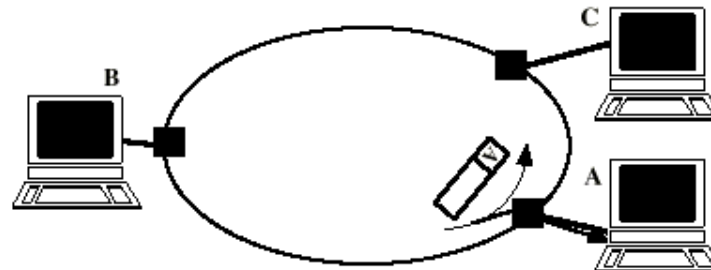
(a) C transmits frame addressed to A



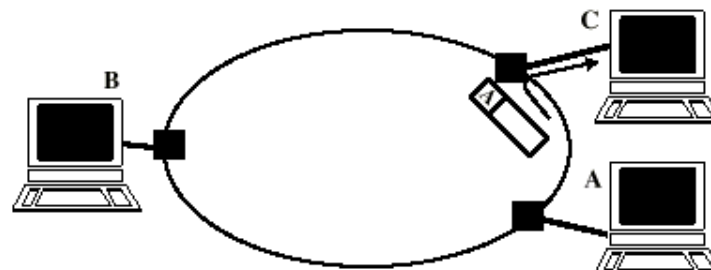
(b) Frame is not addressed to B; B ignores it



(c) A copies frame as it goes by



(d) C absorbs returning frame



- Used in old token ring networks (IEEE 802.4), not used much now
- However, token passing is an important concept in many different contexts

Summary

- We have seen different methods for sharing one medium by more than one sender
- Each sender gets its own chance to send on its own channel to the receiver
 - Logical channels created for each sender-receiver pair pair out of the single physical channel
 - Note that full duplex communication will requires two channels
- Each logical channel appears as a dedicated channel for the sender to send to the receiver, so as if one sender and one receiver on one channel
- We already know from earlier how to make that work