CS 455/555
Intro to Networks and Communications

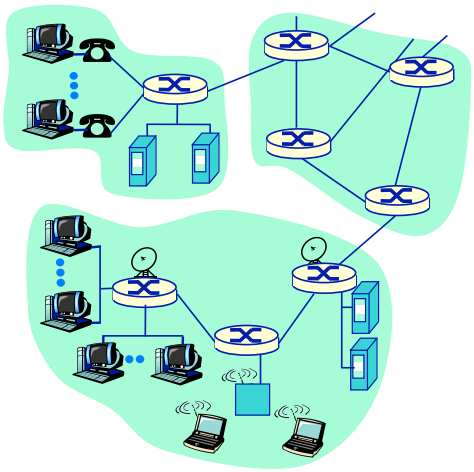
Link Layer

Dr. Michele Weigle
Department of Computer Science
Old Dominion University
mweigle@cs.odu.edu

http://www.cs.odu.edu/~mweigle/CS455-S13

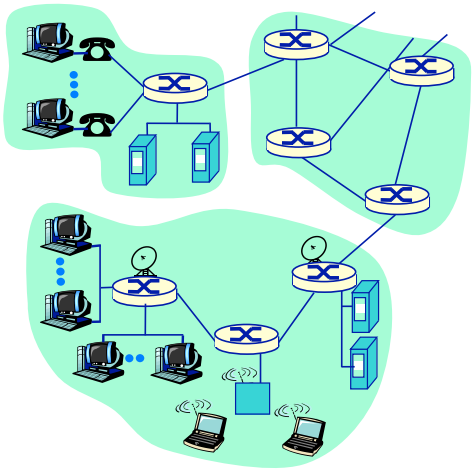
Link Layer
Outline

- Introduction and services
- Error detection and correction
- Multiple access protocols
- Link-layer Addressing
- Ethernet
- Link-Layer Switches



Link Layer
Introduction

- Hosts and routers are ***nodes***
- Communication channels that connect adjacent nodes along communication path are ***links***
  - » wired links
  - » wireless links
  - » LANs
- Layer-2 packet is a ***frame***, encapsulates datagram



data-link layer has responsibility of transferring datagram from one node to adjacent node over a single link

Link Layer
Context

- Datagram may be transferred by different link protocols over different links:
  - » *e.g.*, Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- Each link protocol provides different services
  - » *e.g.*, may or may not provide reliable transport over link

Transportation Analogy

- trip from Norfolk to Florence
  - » car: Norfolk to DC
  - » plane: DC to Pisa
  - » train: Pisa to Florence
- tourist = **datagram**
- transport segment = **communication link**
- transportation mode = **link layer protocol**
- travel agent = **routing algorithm**

Link Layer
Services

- Framing, link access:
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - MAC addresses used in frame headers to identify source, destination
    - different from IP address
- Reliable delivery between adjacent nodes
  - seldom used on low bit error link (fiber, some twisted pair)
  - wireless links: high error rates

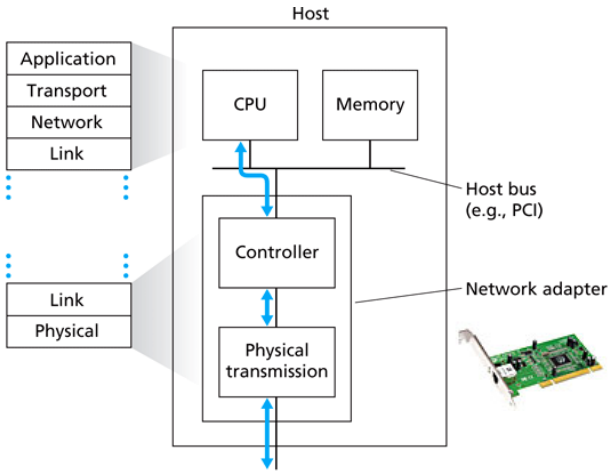
Link Layer
Services (more)

- Flow Control:
  - pacing between adjacent sending and receiving nodes
- Error Detection:
  - errors caused by signal attenuation, noise.
  - receiver detects presence of errors:
    - signals sender for retransmission or drops frame
- Error Correction:
  - receiver identifies and corrects bit error(s) without resorting to retransmission
- Half-duplex and full-duplex
  - with half duplex, nodes at both ends of link can transmit, but not at same time

Link Layer

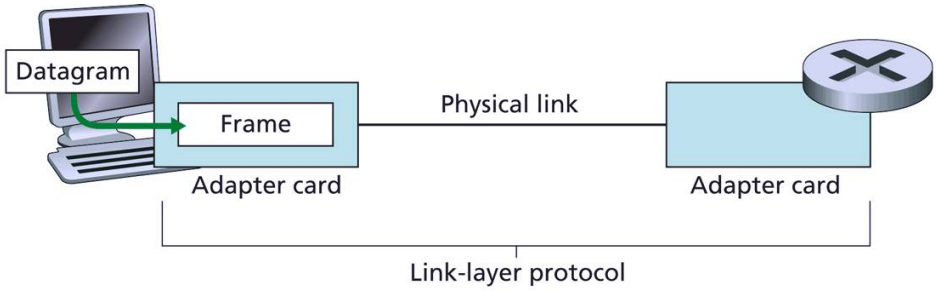
Where is the link layer implemented?

- Link layer implemented in "adapter" (*aka* network interface card NIC)
  - » Ethernet card, PCMCI card, 802.11 card
  - » implements link, physical layer
- Attaches into host's system buses
- Combination of hardware, software, firmware



Link Layer

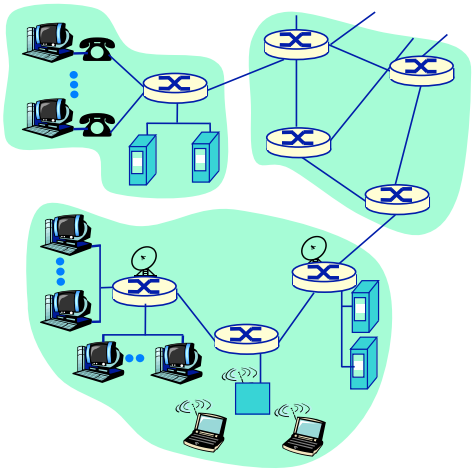
Adapters Communicating



- Sending side:
  - » encapsulates datagram in a frame
  - » adds error checking bits, reliable data transport, flow control, etc.
- Receiving side:
  - » looks for errors, reliable data transport, flow control, etc
  - » extracts datagram, passes to receiving node

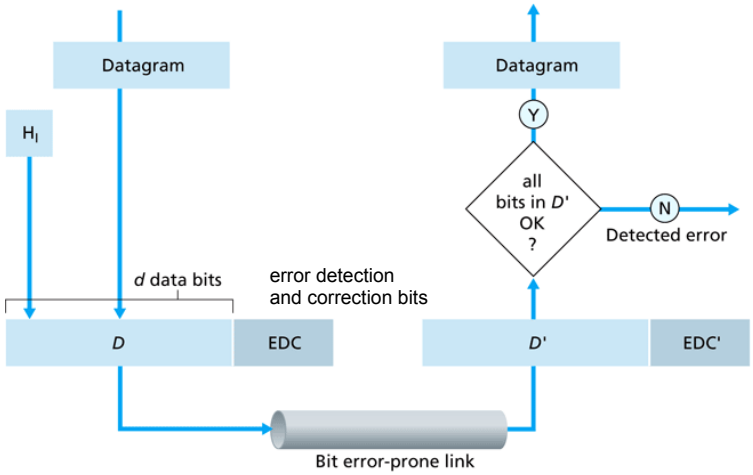
Link Layer
Outline

- Introduction and services
- Error detection and correction
- Multiple access protocols
- Link-layer Addressing
- Ethernet
- Link-Layer Switches



Link Layer
Error Detection

- Error detection not 100% reliable
  - » may miss some errors
  - » larger EDC field yields better detection and correction

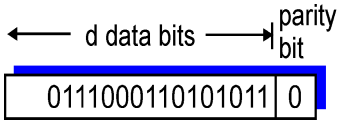


Error Detection
Common Techniques

- Parity Check
  - » illustrate basic ideas behind error detection and correction
- Checksumming
  - » typically used in transport layer
- Cyclic Redundancy Check (CRC)
  - » typically used in the link layer in the adapter

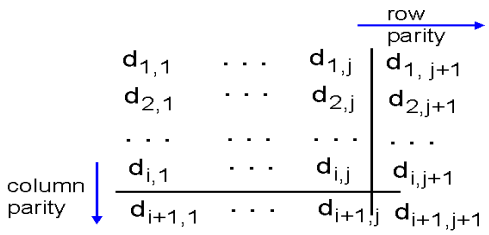
Error Detection
Parity Checking

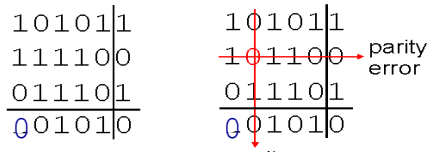
Single Bit Parity:
Detect single bit errors



odd parity

Two Dimensional Bit Parity:
Detect *and correct* single bit errors





no errors

even parity

correctable single bit error

Error Detection
Internet Checksum

Goal: detect "errors" (*e.g.*, flipped bits) in transmitted packet (note: used at transport layer *only*)

Sender:

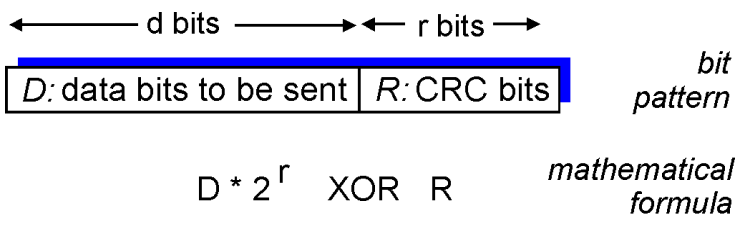
- treat segment contents as sequence of 16-bit integers
- checksum: addition (1's complement sum) of segment contents
- sender puts checksum value into UDP checksum field

Receiver:

- compute checksum of received segment
- check if computed checksum equals checksum field value:
  - » NO - error detected
  - » YES - no error detected. But maybe errors nonetheless?

Error Detection
Cyclic Redundancy Check (CRC)

- View data bits, *D*, as a binary number
- Choose *r*+1 bit pattern (generator), *G*
- Choose *r* CRC bits, *R*, such that <*D*, *R*> exactly divisible by *G* (modulo 2)
- Receiver knows *G*, divides <*D*, *R*> by *G*
  - » if non-zero remainder: error detected!
- Can detect all burst errors less than *r*+1 bits



CRC
Example

Want:

$D \cdot 2^r$ XOR $R = nG$

equivalently:

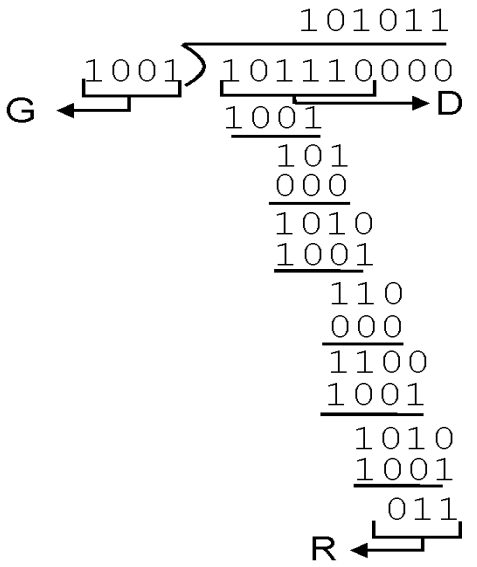
$D \cdot 2^r = nG$ XOR R

equivalently:

if we divide $D \cdot 2^r$ by G, want remainder R

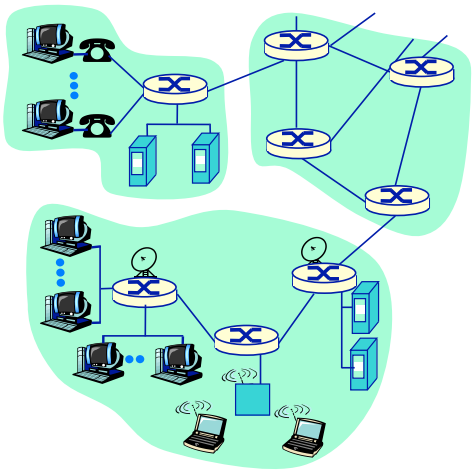
$$R = \text{remainder}\left[\frac{D \cdot 2^r}{G}\right]$$

$d = 6, r = 3$



Link Layer
Outline

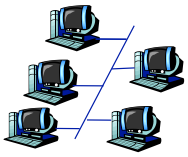
- Introduction and services
- Error detection and correction
- Multiple access protocols
- Link-layer addressing
- Ethernet
- Link-Layer switches



Link Layer
Multiple Access Links and Protocols

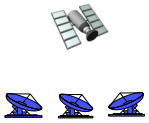
- Point-to-point
  - » PPP for dial-up access
  - » point-to-point link between Ethernet switch and host
- Broadcast (shared wire or medium)
  - » old-fashioned Ethernet
  - » upstream HFC
  - » 802.11 wireless LAN

humans at a cocktail party (shared air, acoustical)



shared wire (e.g., cabled Ethernet)

shared RF (e.g., 802.11 WiFi)



shared RF (satellite)

Link Layer
Multiple Access

- Single shared broadcast channel
- Two or more simultaneous transmissions by nodes: *interference*
  - » *collision,* if node receives two or more signals at the same time
- Multiple access protocol
  - » distributed algorithm that determines how nodes share channel, *i.e.*, determine when node can transmit
  - » communication about channel sharing must use channel itself
    - ❖ no out-of-band channel for coordination

Multiple Access Protocols
Ideal Protocol

Broadcast channel of rate *R* bps

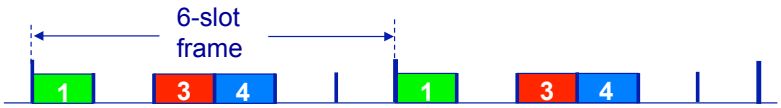
- when one node wants to transmit, it can send at rate *R*
- when *M* nodes want to transmit, each can send at average rate *R/M*
- fully decentralized
  - » no special node to coordinate transmissions
  - » no synchronization of clocks, slots
- simple

Multiple Access Control (MAC) Protocols
Taxonomy

- Channel Partitioning
  - » divide channel into smaller "pieces" (time slots, frequency, code)
  - » allocate piece to node for exclusive use
- Random Access
  - » channel not divided, allow collisions
  - » "recover" from collisions
- "Taking turns"
  - » nodes take turns, but nodes with more to send can take longer turns

Channel Partitioning Protocols
Time Division Multiple Access (TDMA)

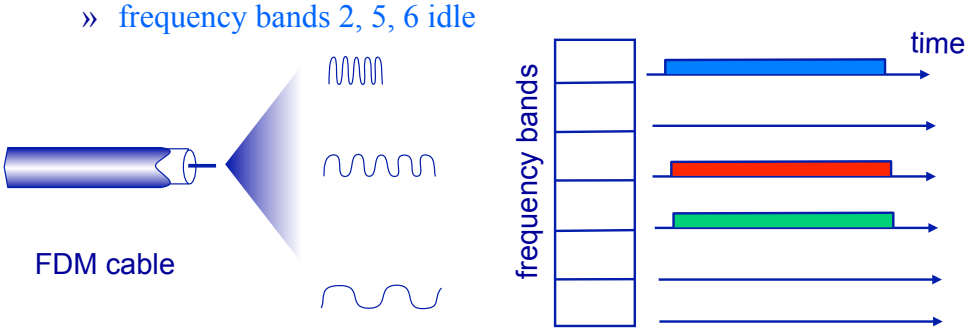
- Access to channel in "rounds"
- Each station gets fixed length slot (length = pkt trans time) in each round
- Unused slots go idle
- Example:
  - » 6-station LAN
  - » 1, 3, 4 have packets
  - » slots 2, 5, 6 idle



Channel Partitioning Protocols
Frequency Division Multiple Access (FDMA)

- Channel spectrum divided into frequency bands
- Each station assigned fixed frequency band
- Unused transmission time in frequency bands go idle

Example:
- » 6-station LAN
- » 1, 3, 4 have packets
- » frequency bands 2, 5, 6 idle



MAC Protocols
Random Access Protocols

- When node has packet to send
  - » transmit at full channel data rate *R*
  - » no *a priori* coordination among nodes
- Two or more transmitting nodes ➜ "collision"
- Random access MAC protocol specifies:
  - » how to detect collisions
  - » how to recover from collisions (*e.g.*, via delayed retransmissions)
- Examples of random access MAC protocols:
  - » slotted ALOHA
  - » ALOHA
  - » CSMA, CSMA/CD, CSMA/CA

Random Access Protocols
Slotted ALOHA

Assumptions:

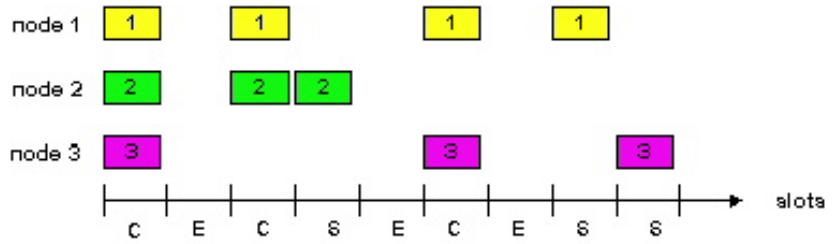
- All frames are *L* bits
- Time divided into equal size slots of *L*/*R* seconds (time to transmit 1 frame)
- Nodes start to transmit only at slot beginning
- Nodes are synchronized
- If 2 or more nodes transmit in slot, all nodes detect collision

Operation:

- When node obtains fresh frame, transmits in next slot
  - » *if no collision:* node can send new frame in next slot
  - » *if collision:* node retransmits frame in each subsequent slot with probability *p* until success

Random Access Protocols

Slotted ALOHA



Pros

- Single active node can continuously transmit at full rate of channel
- Highly decentralized: only slots in nodes need to be in sync
- Simple

Cons

- Collisions, wasting slots
- Idle slots
- Clock synchronization

Slotted ALOHA

Efficiency

Efficiency: long-run fraction of successful slots

- Suppose N nodes with many frames to send, each transmits in slot with probability p
- Probability that given node has success in a slot

  $p\,(1\text{-}p)^{N-1}$
- Probability that any node has a success

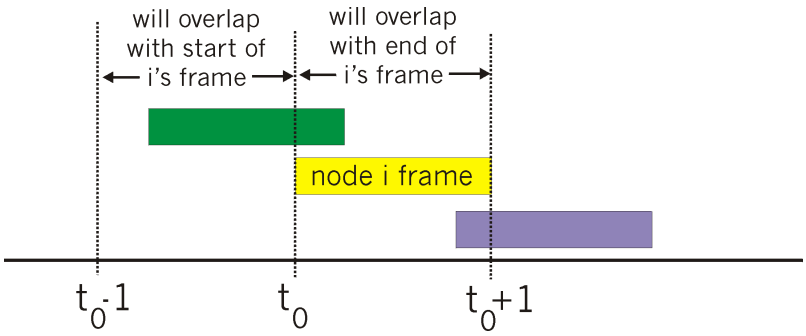
  $Np\,(1\text{-}p)^{N-1}$
- Max efficiency
  - » find p^* that maximizes $Np\,(1\text{-}p)^{N-1}$
- For many nodes, take limit of $Np^*\,(1\text{-}p^*)^{N-1}$ as N goes to infinity
- Max efficiency = $1/e = .37$

At best, channel used for useful transmissions 37% of time!

Random Access Protocols

Pure (unslotted) ALOHA

- Simpler, no synchronization
- When frame first arrives, transmit immediately
- Collision probability increases:
  - » frame sent at t_0 collides with other frames sent in $[t_0\text{-}1, t_0\text{+}1]$



Pure ALOHA

Efficiency

P (success by given node) = P (node transmits) *
P (no other node transmits in $[t_0\text{-}1, t_0]$) *
P (no other node transmits in $[t_0, t_0\text{+}1]$)
$= p\,(1\text{-}p)^{N\text{-}1}\,(1\text{-}p)^{N\text{-}1}$
$= p\,(1\text{-}p)^{2(N\text{-}1)}$

… choosing optimum p and then letting n -> infinity ...

$= 1/(2e) = .18$

even *worse* than slotted ALOHA

Random Access Protocols

CSMA (Carrier Sense Multiple Access)

- Listen before transmit
  - » if channel sensed idle, transmit entire frame
  - » if channel sensed busy, defer transmission for a random amount of time and listen again

- Listen while transmitting
  - » if collision, stop transmitting

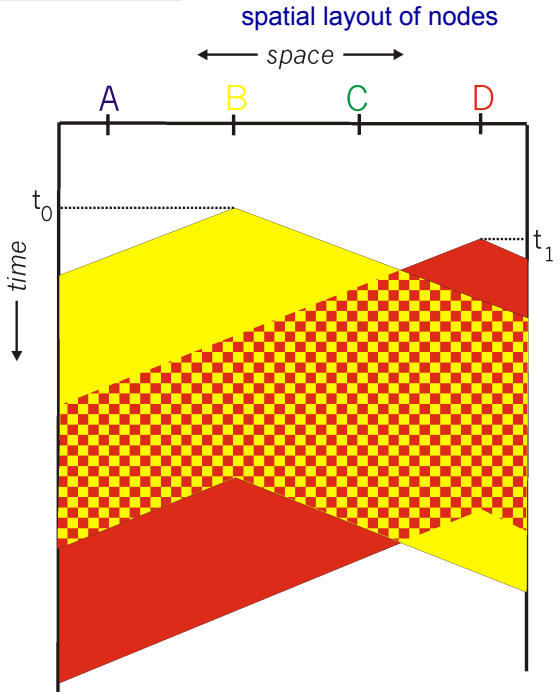
CSMA

Collisions

collisions *can* still occur:
propagation delay means
two nodes may not hear
each other's transmission

collision:
entire packet transmission
time wasted

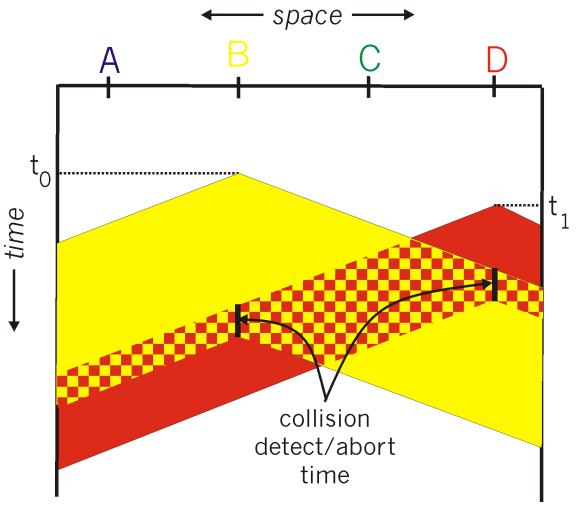
note the role of distance and propagation delay in determining collision probability



CSMA
CSMA/CD (Collision Detection)

- Carrier sensing, deferral as in CSMA
  - » collisions detected within short time
  - » colliding transmissions aborted, reducing channel wastage
- Collision detection
  - » easy in wired LANs
    - ❖ measure signal strengths, compare transmitted, received signals
  - » difficult in wireless LANs
    - ❖ received signal strength overwhelmed by local transmission strength
- Human analogy: the polite conversationalist

CSMA/CD
Collision detection

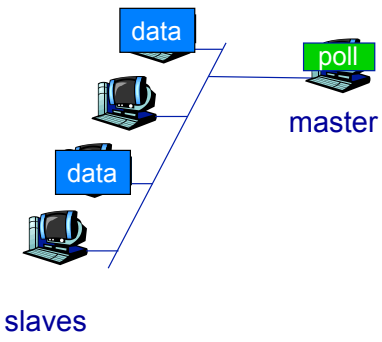


MAC Protocols
"Taking Turns" MAC protocols

- Channel partitioning MAC protocols
  - » share channel efficiently and fairly at high load
  - » inefficient at low load: delay in channel access, 1/*N* bandwidth allocated even if only 1 active node!
- Random access MAC protocols
  - » efficient at low load: single node can fully utilize channel
  - » high load: collision overhead
- "Taking turns" protocols
  - » look for best of both worlds!

"Taking Turns" MAC protocols
Polling

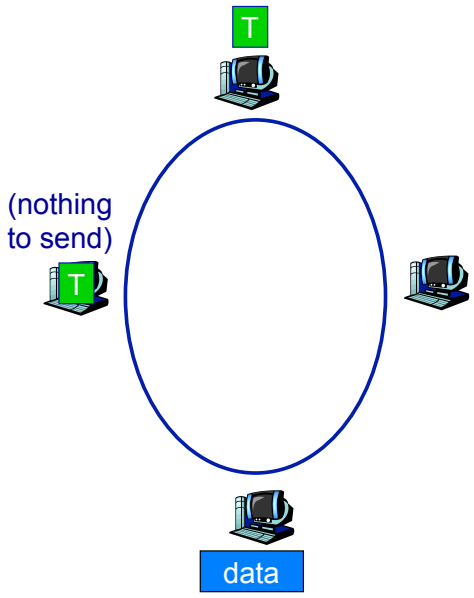
- Master node "invites" slave nodes to transmit in turn
- Typically used with "dumb" slave devices
- Concerns:
  - » polling overhead
  - » latency
  - » single point of failure (master)



"Taking Turns" MAC protocols

Token Passing

- Control *token* passed from one node to next sequentially
- Node can only send when it has the token
- Concerns:
  - » token overhead
  - » latency
  - » single point of failure (token)



MAC Protocols

Summary

- Channel partitioning, by time, frequency, or code
  - » Time Division, Frequency Division
- Random access (dynamic)
  - » ALOHA, S-ALOHA, CSMA, CSMA/CD
  - » carrier sensing: easy in some technologies (wire), hard in others (wireless)
  - » CSMA/CD used in Ethernet
  - » CSMA/CA used in 802.11
- Taking turns
  - » polling from central site, token passing
  - » Bluetooth, FDDI, IBM Token Ring

Link Layer
Outline

- Introduction and services
- Error detection and correction
- Multiple access protocols
- Link-layer Addressing
- Ethernet
- Link-Layer Switches