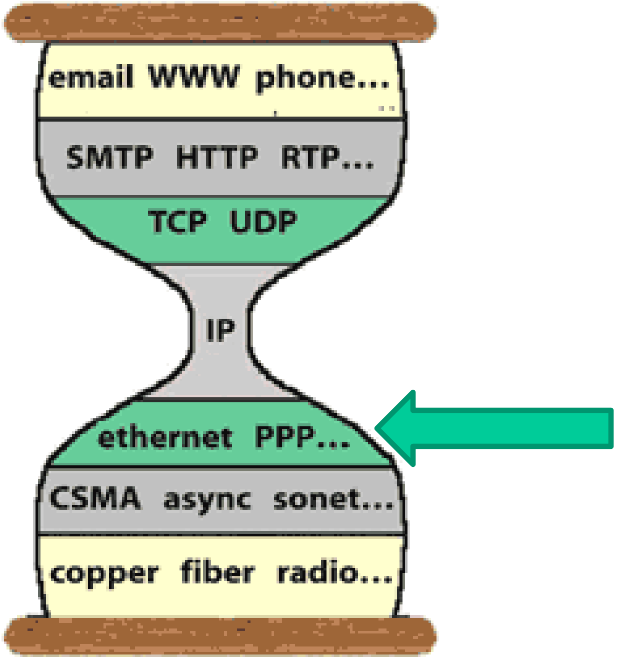
Lecture 15: Link Layer



6.1 Introduction, services

6.2 Error detection, correction

6.3 multiple access protocols

6.4 LANs

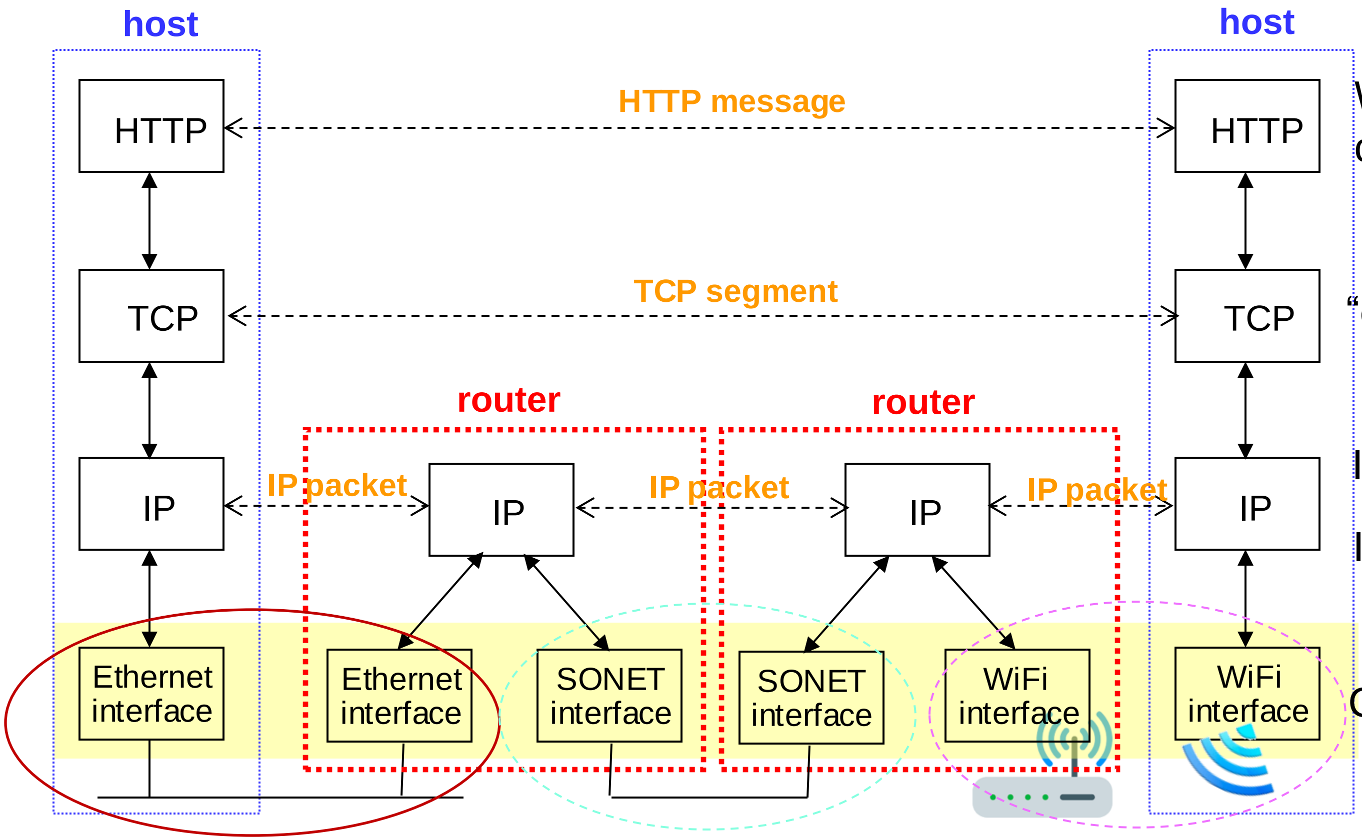
- 6.4.1 Addressing, ARP
- 6.4.2 Ethernet
- 6.4.3 Switches
  - VLANS

6.5 link virtualization: MPLS

6.6 data center networking

6.7 a day in the life of a web request

Where we are in the big picture



Whatever apps want… connection-ish or not

"connect" processes

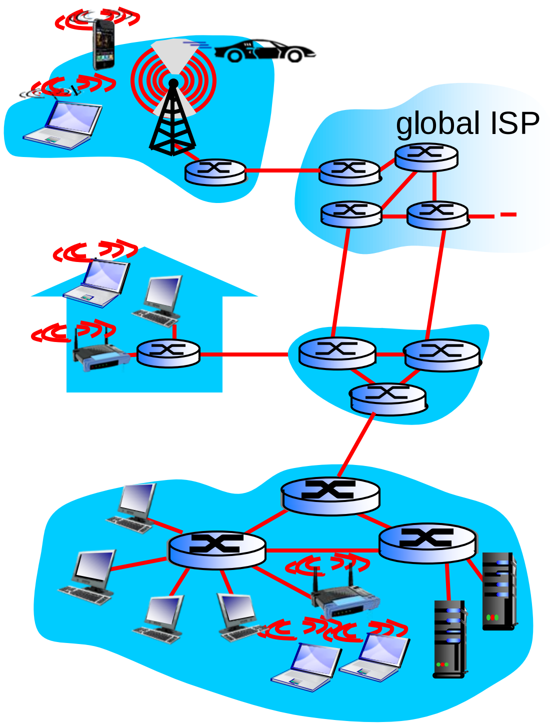
Intra- & inter-domain routing

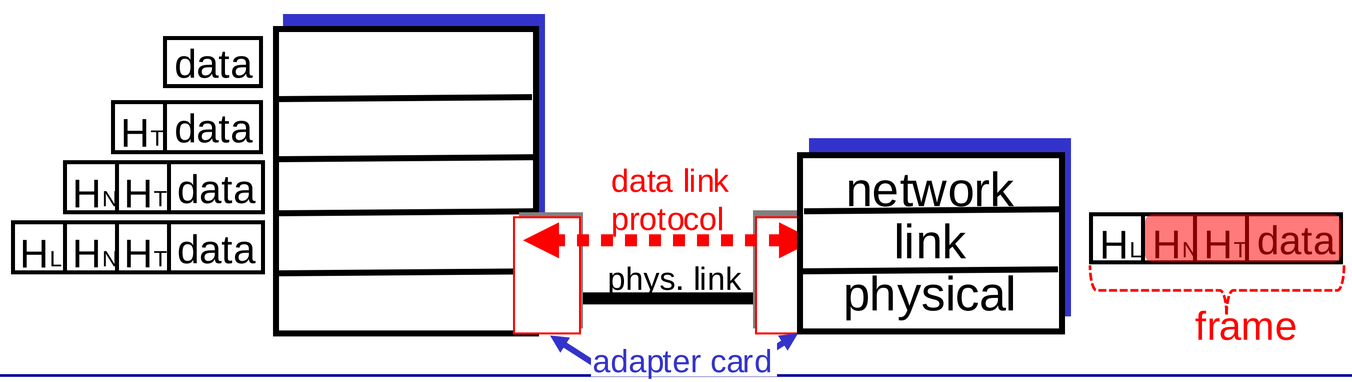
Interconnect *subnets*

Connect physical nodes/NIC

Data Link Layer: overview

- Link layer transfers packets from one node to a *physically connected* node
  - Nodes: routers, hosts
- implementation of various link layer technologies:
  - Ethernet, wireless LANs, LoRA, many others
- *Encapsulate* IP packet in layer-2 frame



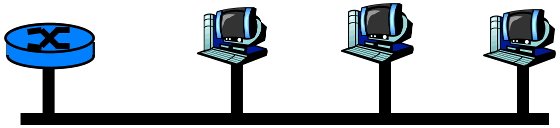


Link Layer: basic concepts

- Link layer address: MAC (Medium Access Control) addresses
- Link type: simplex, Half-duplex, full-duplex

  Multi-access links, e.g, Ethernet, WiFi

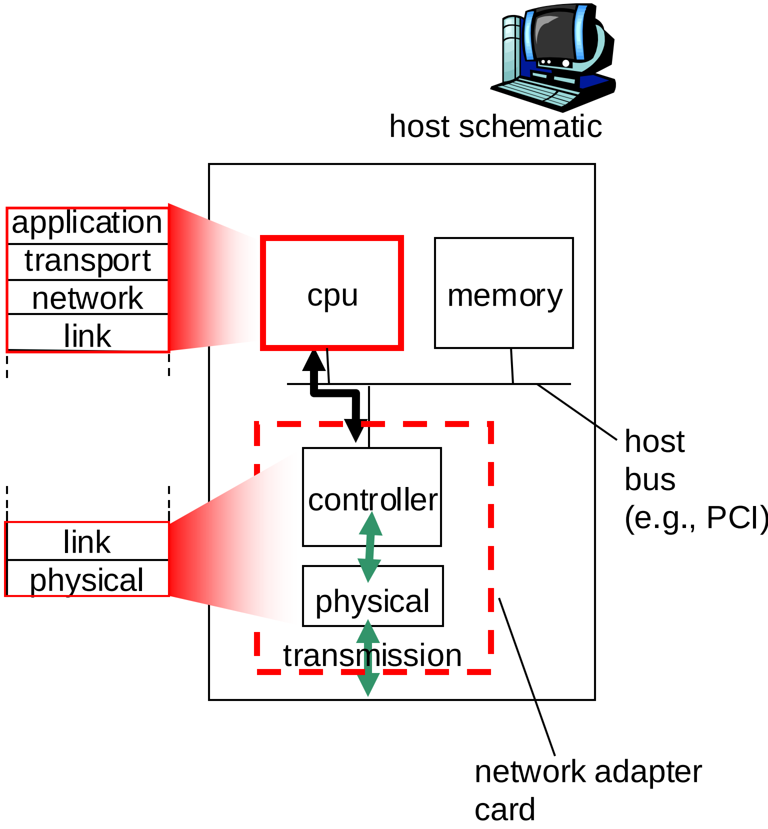
- Link layer functions:
  - Data framing (marking the beginning & end of a data chunk)
  - error detection
  - Channel access protocols



FYI

Where is the link layer implemented?

- Implemented in adaptor (aka *network interface card*, NIC) or on a chip
  - Ethernet card, PCMCI card, 802.11 card
  - implements link & physical layer
- Attached to host's system buses (e.g., PCIe)
- Combination of hardware, software, firmware



Communication between Adaptors

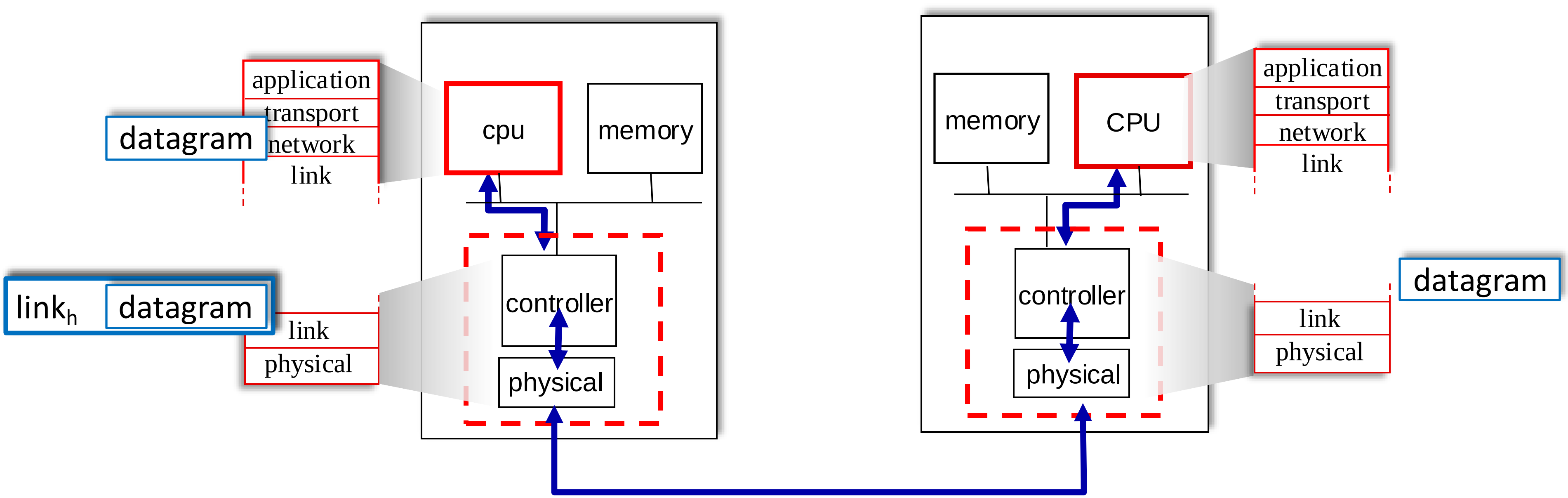
FYI

Sending side:

- Encapsulates IP packet in frame
- Adds error checking bits
- Follows access control protocol to send frame out

Receiving side

- Looks for errors
- If OK, extracts datagram, passes to upper layer at receiving side



important

Data Framing

- For a block of data: different name at different layer
  - at link layer: a data frame
  - at network layer: an IP datagram
  - at transport level: TCP — a segment
- A frame has a header field
  - optionally there may be a **trailer** field as well

- Byte-Oriented Framing Protocol: delineate frame with a byte of special bit sequence: 01111110

Q: What if the bit sequence 01111110 occurs in data stream?

Byte stuffing

important

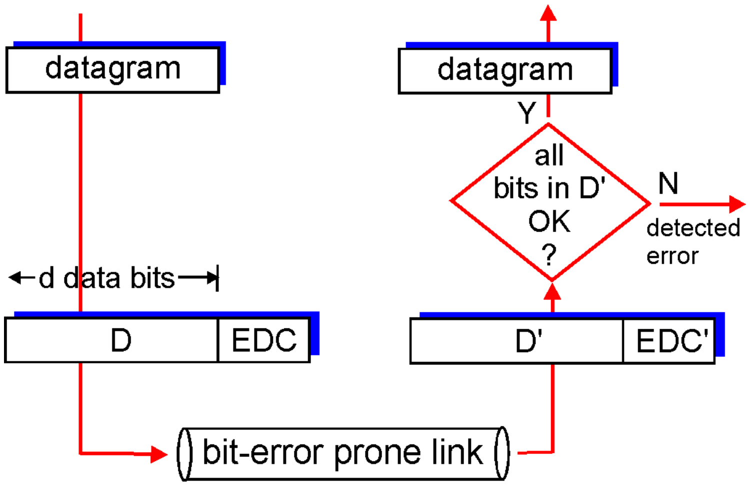
- Sender: adds ("stuffs") extra 01111110 byte after each appearance of 01111110
- Receiver:
  - If single 01111110: flag byte
  - If 2 back-to-back 01111110 bytes: discard first byte, continue data reception
- Example:
  - Original user data: 01111110 01010101 01111110 01111110
  - After byte stuffing (before sending out):

01111110 01111110 01010101 01111110 01111110 01111110 01111110

FYI

Error Detection

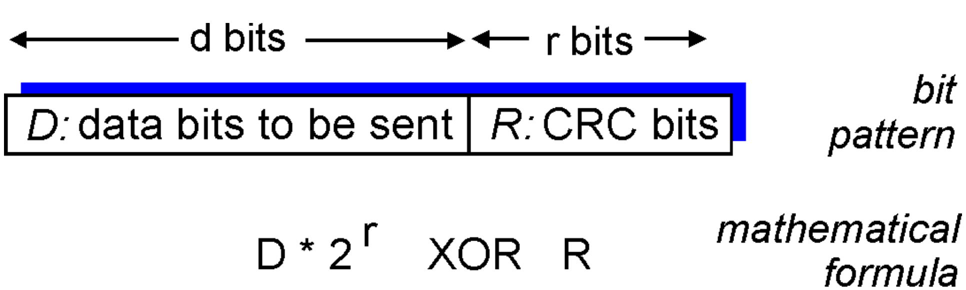
- EDC= Error Detection and Correction bits
- D = Data protected by error checking
- Error detection not 100% reliable!
  - protocol may miss some errors, though rarely
  - larger EDC field offers better detection and correction



Cyclic Redundancy Check (CRC)

FYI

- consider a data frame as a bit sequence D
- choose a (r+1) bit pattern (generator), G
  - known to both sender and receiver
- Goal: Sender chooses r CRC bits, R, such that
  - <D, R> = D * 2^r divisible by G (modulo 2)
  - receiver divides the received bit sequence by G. If non-zero remainder: error detected!
- widely used in practice (Ethernet, 802.11 WiFi)



FYI

CRC example

We want:

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

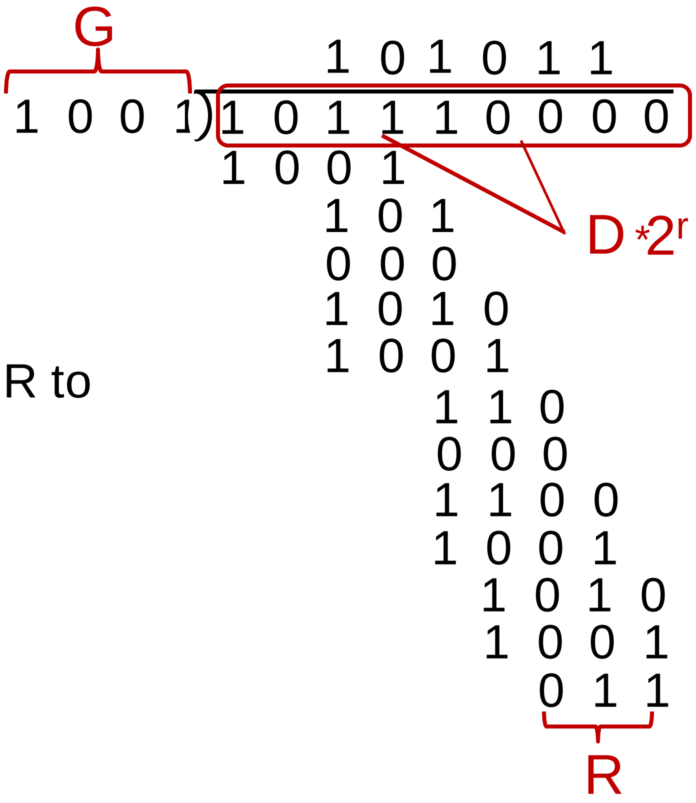
or equivalently:

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

or equivalently:

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

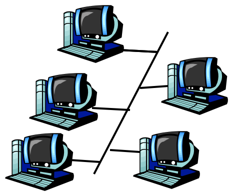
$$R = remainder\left[\frac{D \cdot 2^r}{G}\right]$$



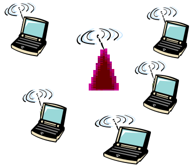
* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/

Multiple-Access Links and Protocols

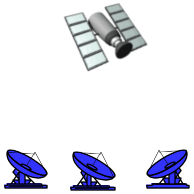
- Sharing a single transmission medium can lead to *collisions*
  - Two or more parties speaking at the same time (intersecting times)
  - Receivers cannot decode frames
- Multi-access protocols “coordinate” *when* a node can speak
  - ”Hard” coordination
  - “Soft” coordination



shared wire (e.g., cabled Ethernet)



shared RF (e.g., 802.11 WiFi)

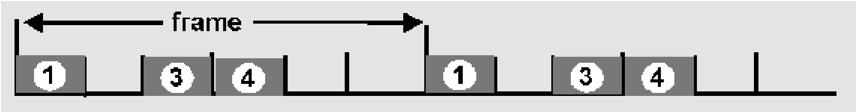


shared RF (satellite)

humans at a cocktail party (shared air, acoustical)

Multiple Access Control

- An ideal solution: given a broadcast channel of rate R bit-per-sec,
  - If only one node wants to send: can send at rate R
  - If M nodes want to send: each can send at rate R/M
  - *simple, no central controller*
    - *no special node to coordinate transmissions*
    - *no synchronization of clocks, slots*
- 3 classes of solutions:
  - Channel partitioning: divide the channel into pieces
    - By time/frequency/code

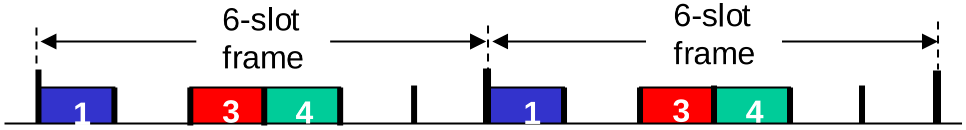


  - Taking turns: coordinated access to avoid collision
  - Random Access: no coordination
    - Try to avoid collisions
    - detect and resolve collisions in case they occur

Channel partitioning MAC protocols: TDMA

TDMA: time division multiple access

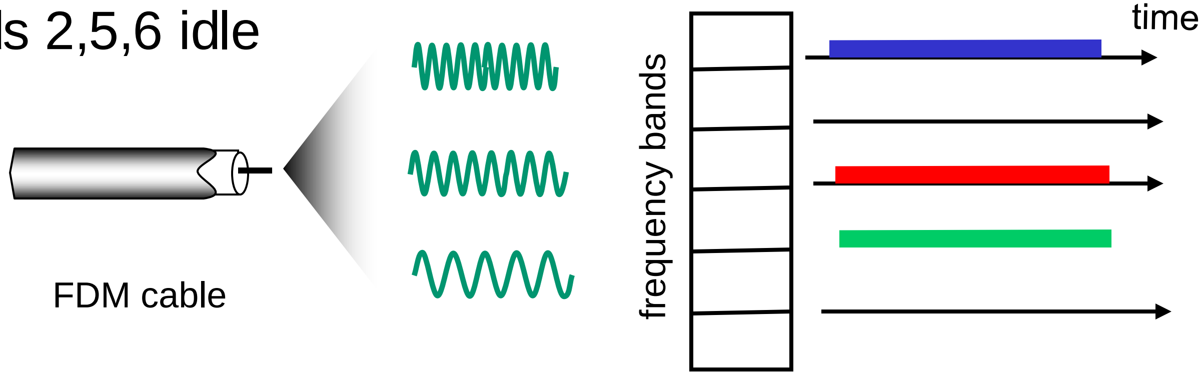
- access to channel in "rounds"
- each station gets fixed length slot (length = packet transmission time) in each round
- unused slots go idle
- example: 6-station LAN, 1,3,4 have packets to send, slots 2,5,6 idle



Channel partitioning MAC protocols: FDMA

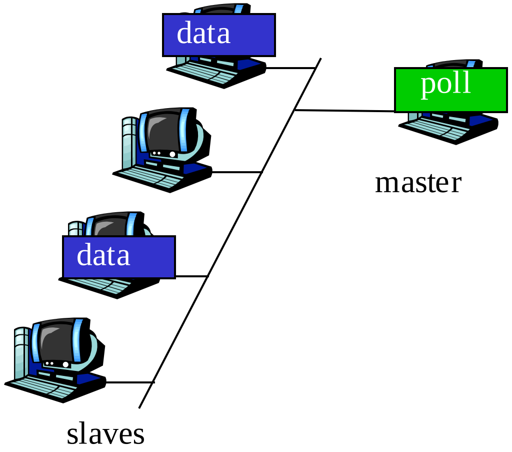
FDMA: frequency division multiple access

- 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 packet to send, frequency bands 2,5,6 idle



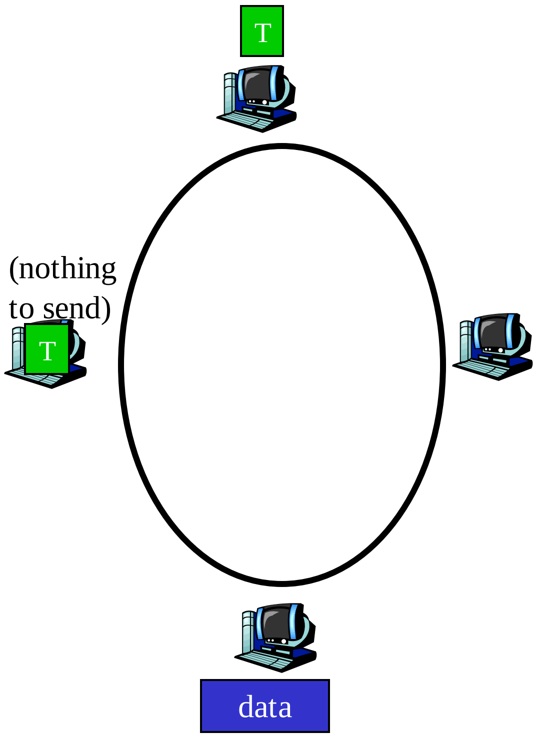
“Taking Turns” MAC protocols

- *On-demand* channel allocation
- Polling:
  - master node asks slave nodes to transmit in turn
  - Concerns
    - polling overhead
    - Latency
    - single point of failure (master)



"Taking Turns" MAC protocols (II)

- Token passing
  - One token message passed from one node to next sequentially
  - whoever gets the token can send one data frame, then pass token to next node
- Concerns:
  - latency
  - single point of failure (the token)
- A master station generates the token and monitors its circulation
  - If token is lost, generate a new one

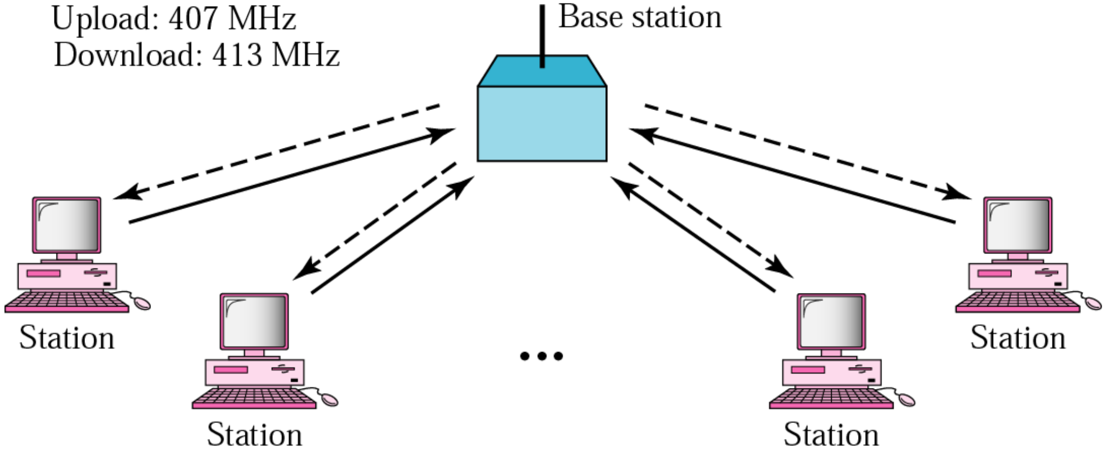


Random Access protocols

- Let a node transmit at full channel data rate R
  - *no a priori coordination among nodes*
  - If collision happens: detect and recover from it
- When collide (2 or more nodes transmitting at the same time), a random access protocol needs to figure out
  - how to detect a collision
  - how to recover from a collision
- Examples of random access MAC protocols:
  - ALOHA, slotted ALOHA
  - CSMA/CD, CSMA/CA
    - CSMA: channel sensing, multiple access
    - CD: collision detection
    - CA: collision avoidance

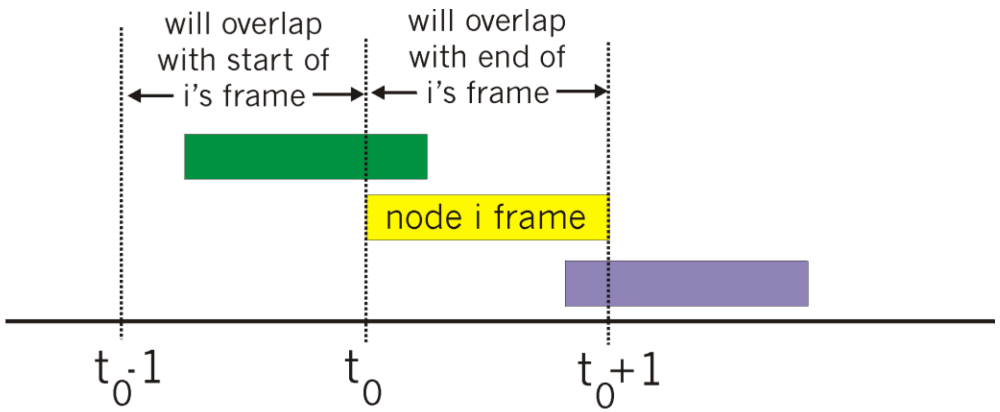
ALOHA History

- Developed by Norm Abramson at Univ. of Hawaii in 1970
  - The world's first wireless packet-switched network
- Why ALOHA
  - mountainous islands → wire-based network infeasible
  - Radio channel → high error rate → centralized control infeasible
- Upload channel: contention-based random access
- Download channel: rebroadcasting all received packets



ALOHA

- If a node has data to send, send the whole frame immediately
  - If collision: retransmits the frame again with the probability *p*
- collision probability: assume all frames of same size, frame sent at t_0 may collide with other frames sent in $[t_0-1, t_0+1]$

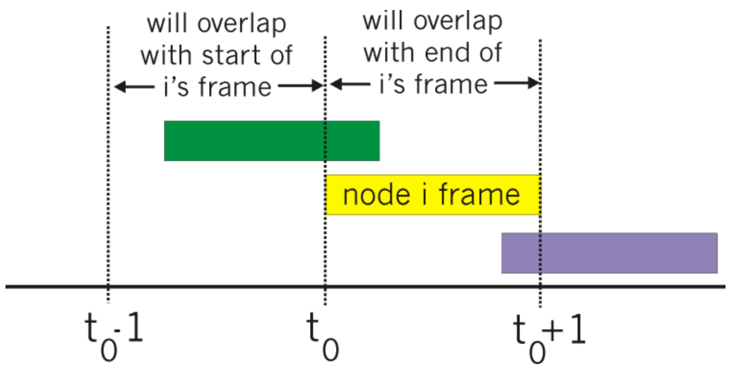


Pure ALOHA Efficiency

- Probability (p) to transmit a frame by one node in $[t_0, t_0 + 1]$ while
  - no other node in the system transmits during $[t_0 - 1, t_0]$ (p1)
  - no other node in the system transmits during $[t_0, t_0 + 1]$ (p2)

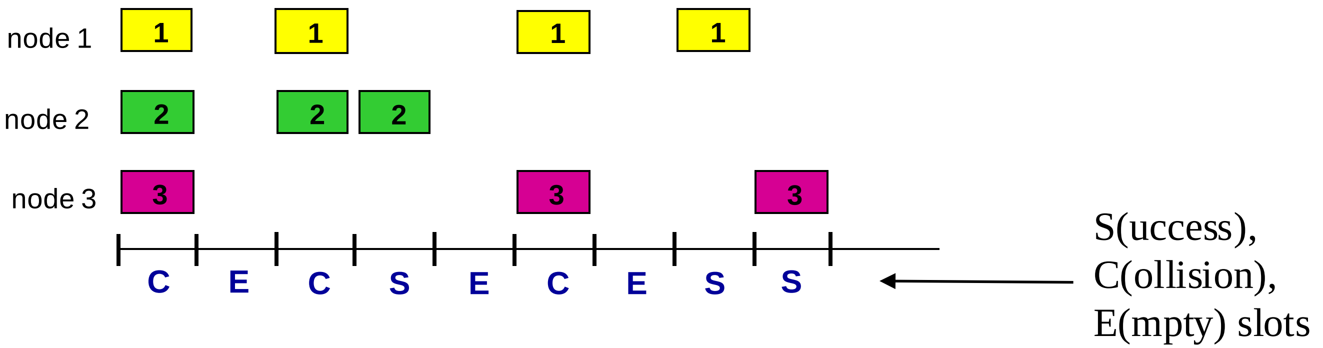
- One node success = $p \cdot p1 \cdot p2 = p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1} = p \cdot (1-p)^{2(N-1)}$

- Any node success = efficiency $= Np \cdot (1-p)^{2(N-1)}$
  - … choosing optimum p as $N \to \infty$
  - $\max efficiency = \frac{1}{2e} = 0.18$



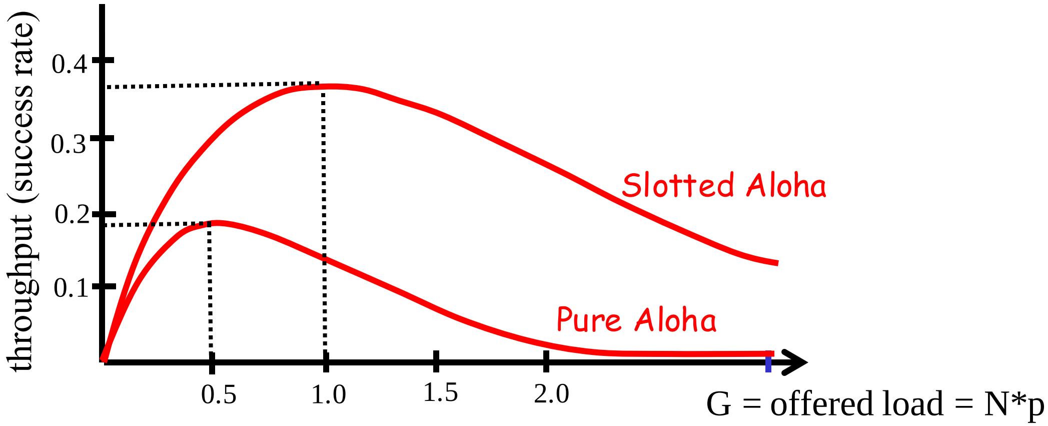
Slotted Aloha

- Assumptions:
  - Divide time into equal size slots (= frame transmission time)
  - clocks in all nodes are synchronized
  - If 2 or more nodes collide in one slot, all nodes detect collision
- Operations: a node transmits only at <u>beginning</u> of next slot
  - If no collision, node can send new frame in next slot
  - If collision, retransmit in each subsequent slots with probability p, until succeed



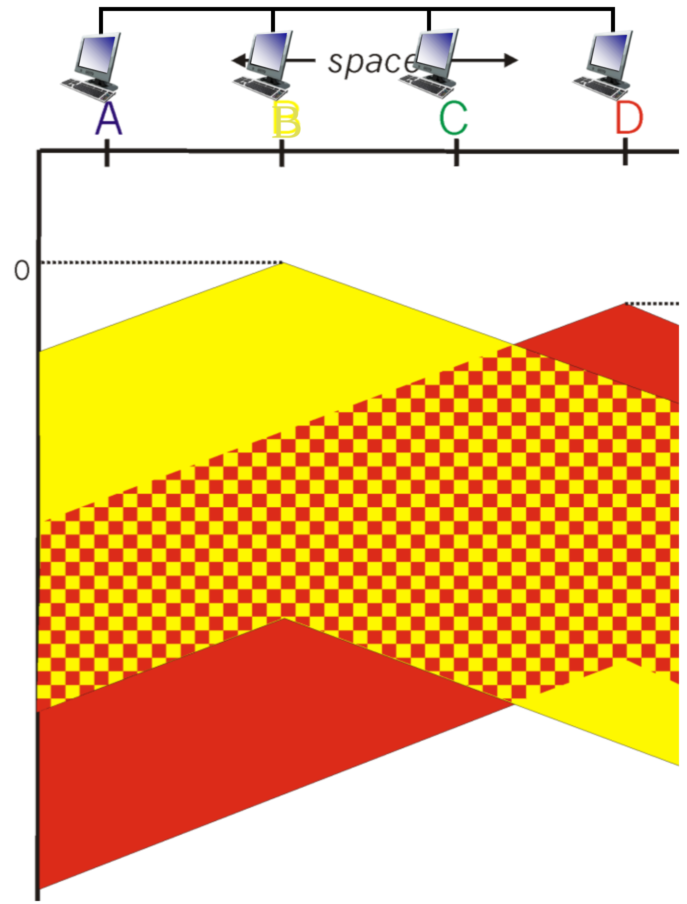
Efficiency of Slotted ALOHA

- Probability (p) to transmit a frame by one node in $[t_0, t_0 + 1]$ while
  - no other node in the system transmits during $[t_0, t_0 + 1]$ (p2)
- One node success = $p \cdot p1 = p \cdot (1-p)^{N-1} = p \cdot (1-p)^{N-1}$
- Any node success = $efficiency = Np \cdot (1-p)^{N-1}$
  - … choosing optimum p and then letting $N \to \infty$
  - $max\ efficiency = \frac{1}{e} = 0.37$



CSMA: Carrier Sense Multiple Access

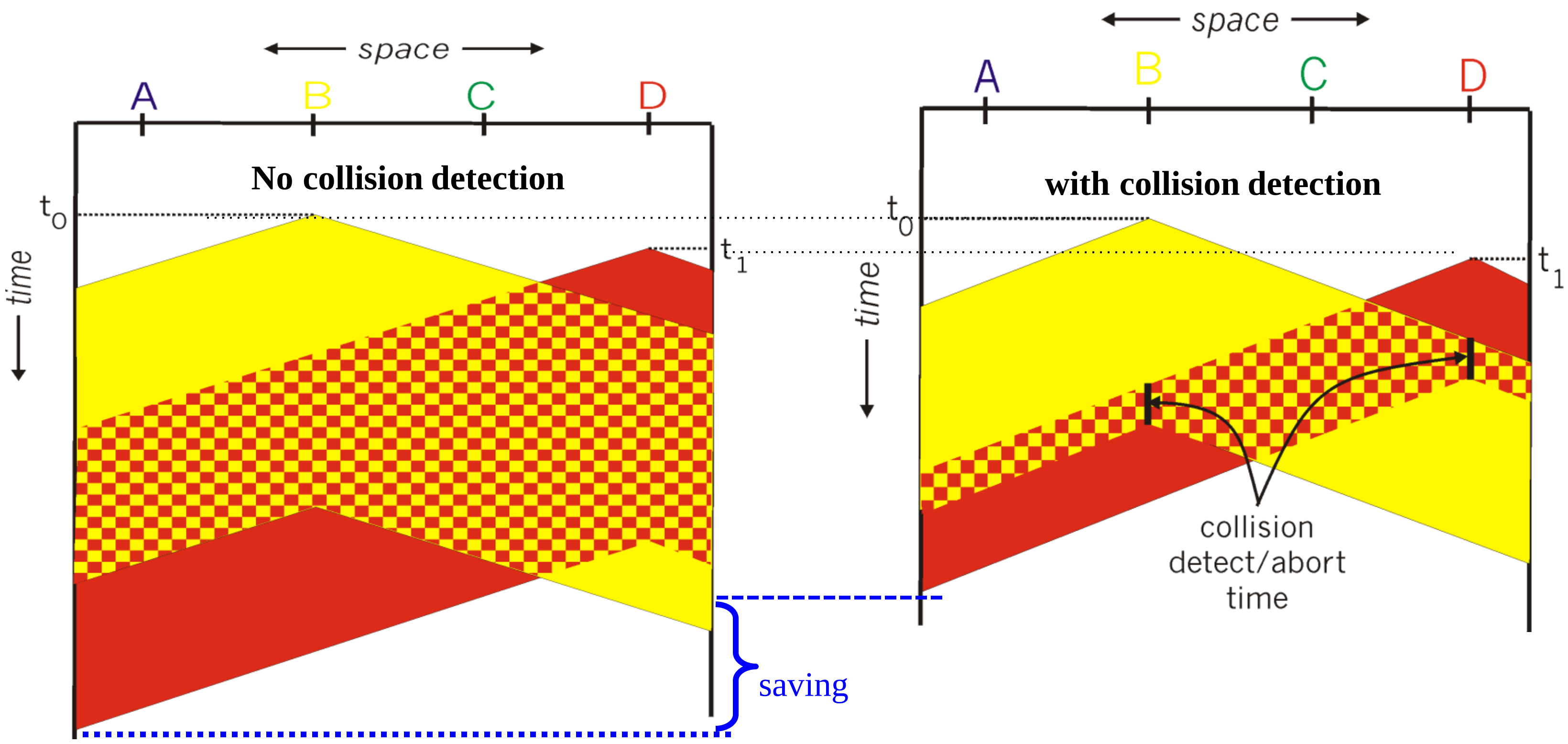
- listen before transmit
  - If channel *sensed* idle: transmit
- If channel sensed busy, wait until it becomes idle; once idle;
  1-persistent CSMA: retry immediately
  p-persistent CSMA: retry immediately with probability p
  Non-persistent CSMA: retry after a random interval
- collisions still possible:
  - Chance of collision goes up with distance between nodes



To cut the loss early: CSMA/CD

CSMA/CD (Collision Detection)

- Collision Detection: compare transmitted with received signals
- Abort collided transmissions

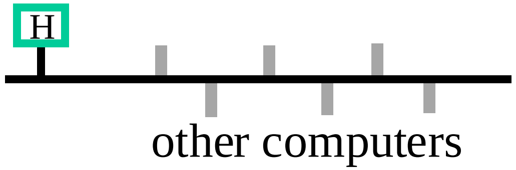


important

Ethernet CSMA/CD Algorithm

1. NIC receives datagram from network layer, creates frame
2. If NIC senses channel idle, starts frame transmission. If NIC senses channel busy, waits until channel idle, then transmits
   - "1-persistent"
3. If NIC transmits entire frame without detecting another transmission, NIC is done with frame!
4. If NIC detects another transmission while transmitting, aborts and *sends jam signal* for a short time period
5. After aborting, NIC enters binary exponential backoff:
   - after m_{th} collision, NIC chooses a value K at random from $\{0,1,2, \ldots, 2^m - 1\}$
     - NIC waits K slots, returns to Step 2
       - 1 slot= transmission time for 512 bits
       - more collisions → much longer backoff interval

An example: host **H** on an Ethernet with data to send, collided 2 times in a row. What's the probability H will choose K=2 for its 3rd try?



binary exponential backoff:

- after m^{th} collision, NIC chooses K randomly from $\{0,1,2, \ldots, 2^m - 1\}$
  - NIC waits K slots, returns to Step 2 (sense channel)

After 1st collision: choose between $[0, 2^1 - 1] = [0, 1]$:

- Wait or no wait: each has 50% chance

After 2nd collision: choose between $[0, 2^2 - 1] = [0, 3]$:

- Randomly pick from 0, 1, 2, or 3 time slots to wait, each gets ¼ chance

important

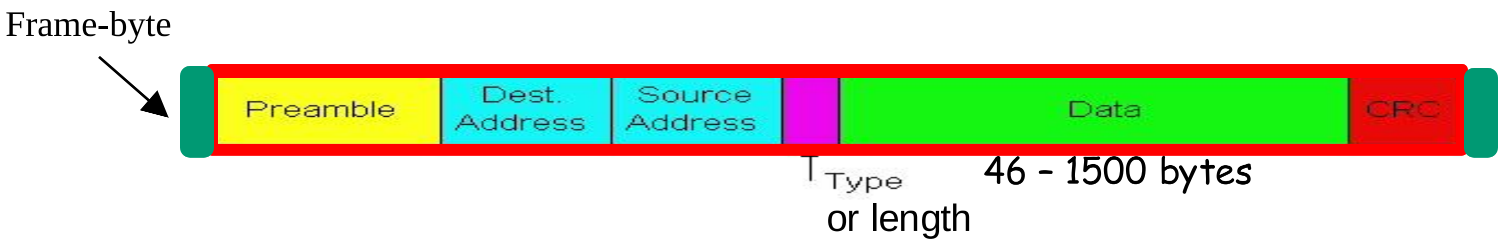
CSMA/CD efficiency

- T_{prop} = maximum propagation delay between any 2 nodes
- T_{trans} = time to transmit a maximum-sized frame

$$\text{efficiency} = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

- Efficiency approaches 1
  - as T_{prop} goes to 0
  - as T_{trans} goes to infinity
- What happens when Ethernet speed changed from 10Mbps to 100Mbps, and to 1Gbps?

Ethernet Frame Structure



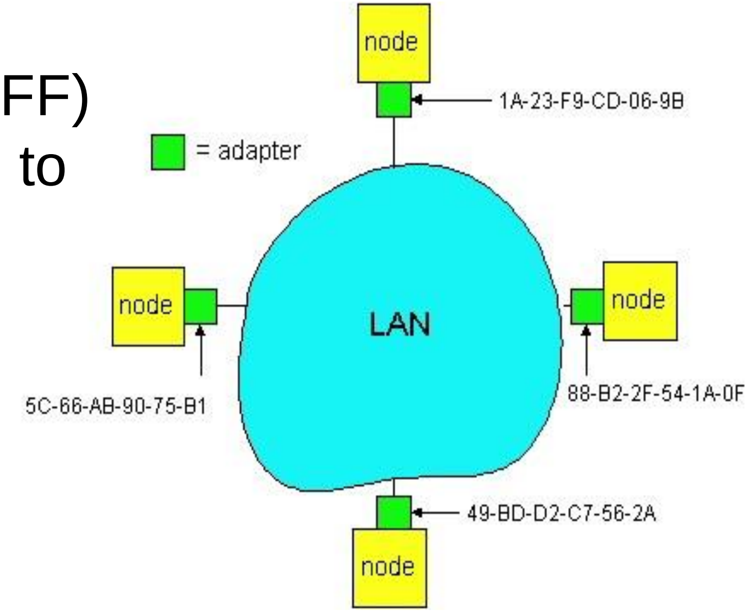
- The sending adapter encapsulates an IP datagram in an Ethernet frame
- Preamble: 8bytes
  - 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
  - used to synchronize receiver, sender clock rates
- Addresses: 6 bytes each (MAC address)
  - If the received frame destination address matches NIC address, or is broadcast address, the adapter passes data to network layer protocol; otherwise, discards frame
- Type: 2 bytes, indicates the higher layer protocol
  - IEEE802.3 changed the “type” field to “length”, defined a separate type field in the data part
- CRC: 4 bytes, added by sender, checked at receiver, if error, drop the frame

Medium Access Control (MAC) Address

- Ethernet & WiFi use 48-bit MAC addresses
  - Each interface on LAN has a unique MAC address

    e.g.: 1A-2F-BB-76-09-AD

    hexadecimal (base 16) notation
    (each "number" represents 4 bits)
- Hard-coded into adapter (software settable in some cases)
  - Blocks: assigned to vendors (e.g., Apple) by IEEE
  - Adapters: assigned by the vendor from its block
- Special addresses
  - Broadcast address (FF-FF-FF-FF-FF-FF)
  - Group addresses (01-80-C2-00-00-00 to 01-80-C2-FF-FF-FF)



MAC Address (more)

important

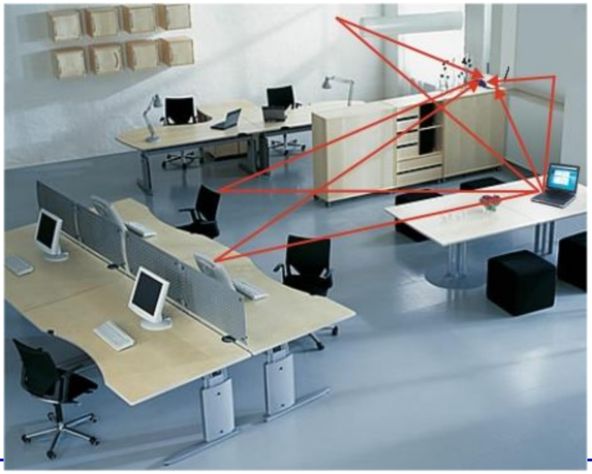
- IEEE controls MAC address allocation
  - ***I**nstitute of **E**lectrical and **E**lectronics **E**ngineers*
- (adaptor) manufacturers buy MAC address blocks from IEEE
  - Assuring uniqueness
- MAC address is flat → portability
  - LAN (local area network) card can move from one LAN to another
- IP address is hierarchical, NOT portable
  - Tied to the network a node is attached to
- Analogy:
  - MAC address: like Social Security Number
  - IP address: like postal address

Wireless channel characteristics

FYI

- decreased signal strength: radio signal attenuates as it propagates through matter
- Interference signals from other sources
  - standardized wireless network frequencies (e.g., 2.4 GHz) shared by other devices (e.g., microwave oven, cordless phone)
- multipath propagation: radio signal reflects off objects around (e.g. walls), arriving at destination at slightly different times

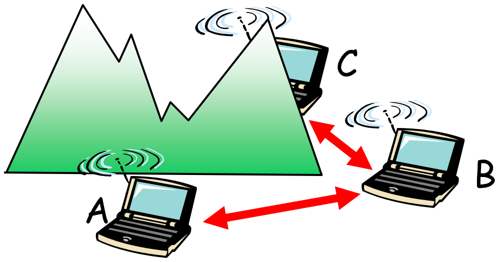
the above make communication across (even a point to point) wireless link much more "difficult"



other problems with wireless channels

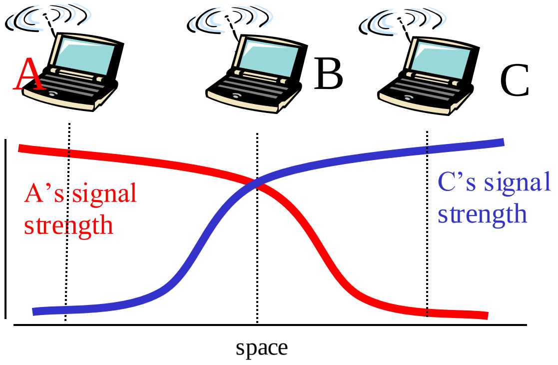
Hidden terminal

- B, A hear each other
- B, C hear each other
- A, C can't hear each other
- A, C may send to B at the same time, cause collision at B



Signal attenuation

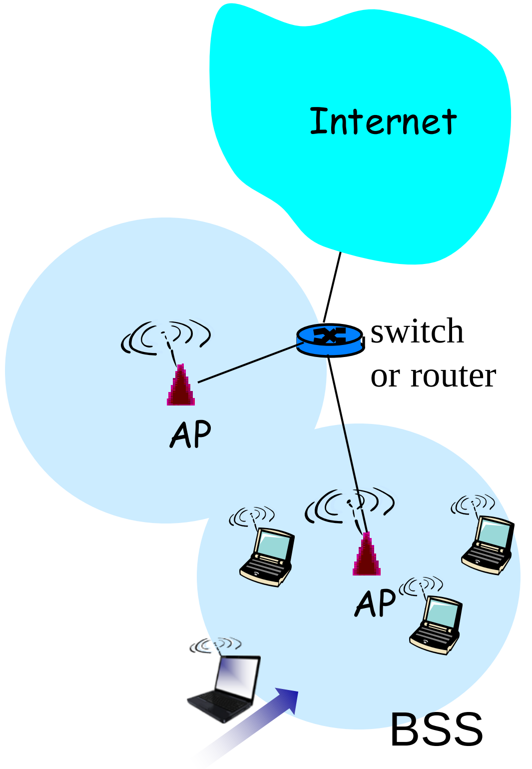
- B, A hear each other
- B, C hear each other
- A, C cannot hear each other → interference at B



IEEE 802.11 (WiFi) LAN architecture

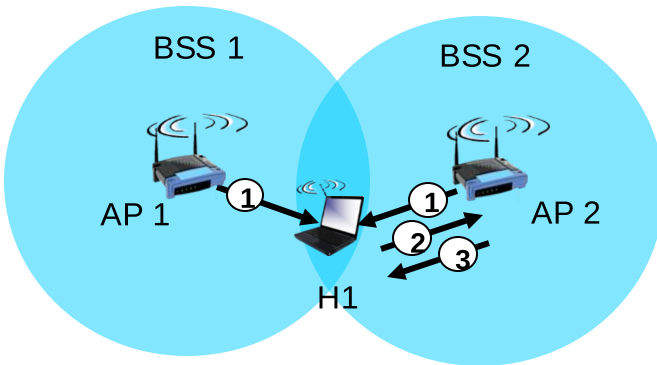
FYI

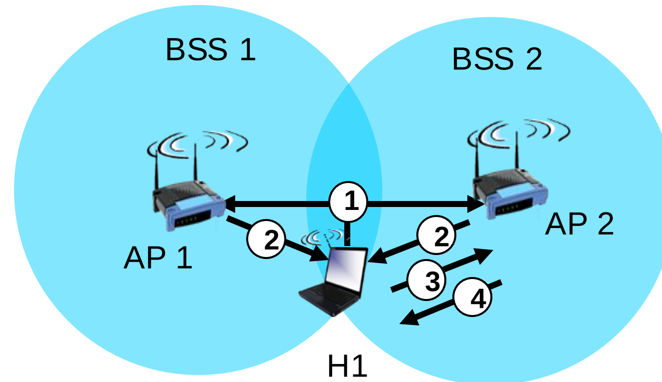
- **AP**: access point (also called base-station)
  - **BSS**: Basic Service Set (aka "cell"), contains wireless hosts and access point (AP)
  - **SSID**: Service Set Identifier
- 802.11: spectrum divided into channels at different frequencies
  - Administrator chooses frequency for an AP
  - If neighbor APs use same channel ⇒ interference
- AP sends *beacon frame* periodically
  - Contain SSID and its own MAC address
- Arriving host: must associate with an AP before transmitting
  - scan channels, listening for *beacon frames*
  - then select an AP to associate with by initiating association protocol
  - then run DHCP to get IP address in AP's subnet



802.11: passive/active scanning

FYI





passive scanning:

(1) beacon frames sent **from APs**

(2) Association Request frame sent from H1 to selected AP

(3) Association Response frame sent from selected AP to H1

active scanning:

(1) H1 *broadcasts* Probe Request frame

(2) APs send Probe Response frames

(3) H1 sends Association Request frame to selected AP

(4) selected AP sends Association Response frame to H1

IEEE 802.11 multiple access

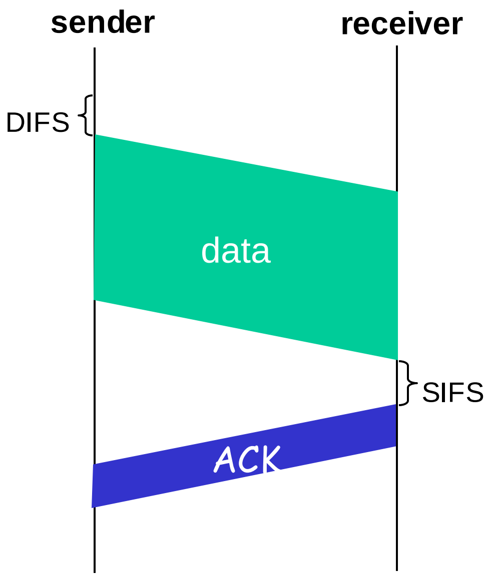
- Similar to Ethernet, CSMA: sense the channel before transmitting
  - avoid collision with ongoing transmission
- Unlike Ethernet:
  - *no collision detection* – once start, transmit a frame to completion
  - *Receiver sends acknowledgment* – enable the sender to find out whether the transmission collided or succeeded
- Why no collision detection?
  - weak received signals (fading) → difficult to receive (sense collisions) when transmitting
  - can't sense all collisions, e.g. due to hidden terminal
- Goal: avoid collisions: **CSMA** / **C**(ollision)**A**(voidance)

IEEE 802.11 MAC Protocol: CSMA/CA

FYI

802.11 sender: channel sensing

1. If sense channel idle for **DIFS** period then transmit *entire* frame
2. Else if sense channel busy
   - start random back-off timer
     - timer counts down while channel busy
   - when timer expires
     - If channel busy, go back to step-2
     - If channel idle, start transmitting frame, then set a timer to wait for ACK
       - If ACK received: success
       - if no ACK, retry



DIFS: Distributed Inter-Frame Spacing
SIFS: Spacing between transmission and ACK

802.11 receiver

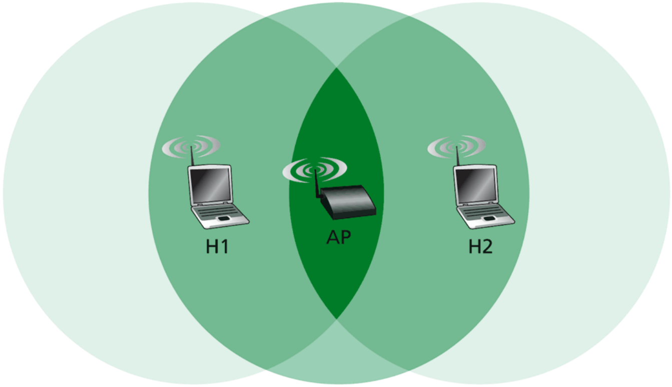
if frame received OK then return ACK after **SIFS**

Active Collision Avoidance

FYI

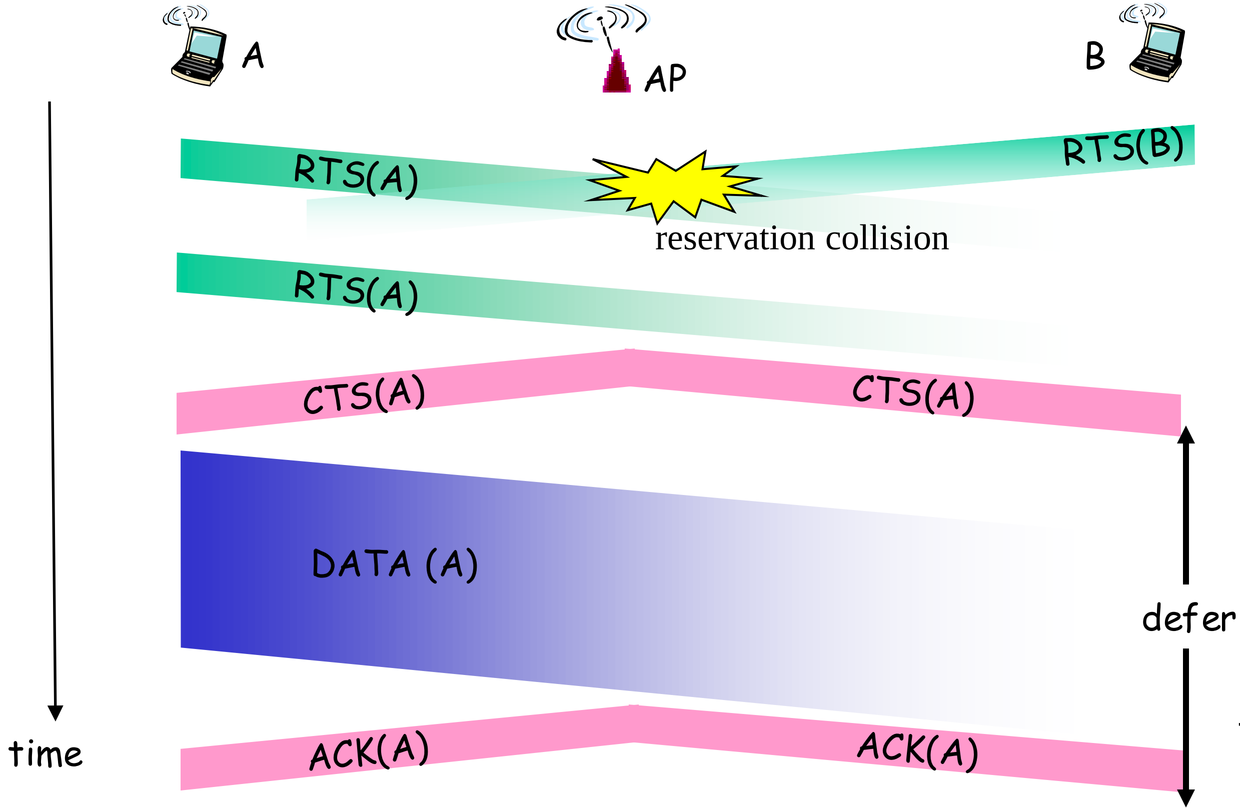
- **idea**: allow sender to "reserve" channel, to avoid collisions of long data frames
- sender first transmits a small request-to-send (RTS) packet to AP using CSMA
  - RTSs may still collide with each other (but they're short)
    - Set a retransmission timer: if no CTS arrival, retry
- AP broadcasts clear-to-send (CTS) in response to RTS
- CTS heard by all nodes within AP's wireless range
  - sender transmits its data frame
  - other stations defer transmissions

Use small packet exchanges to avoid data frame collision



Collision Avoidance: RTS-CTS exchange

FYI



CTS carries data frame size info, so B knows how long to defer its retry of RTS

802.11 frame: addressing

FYI

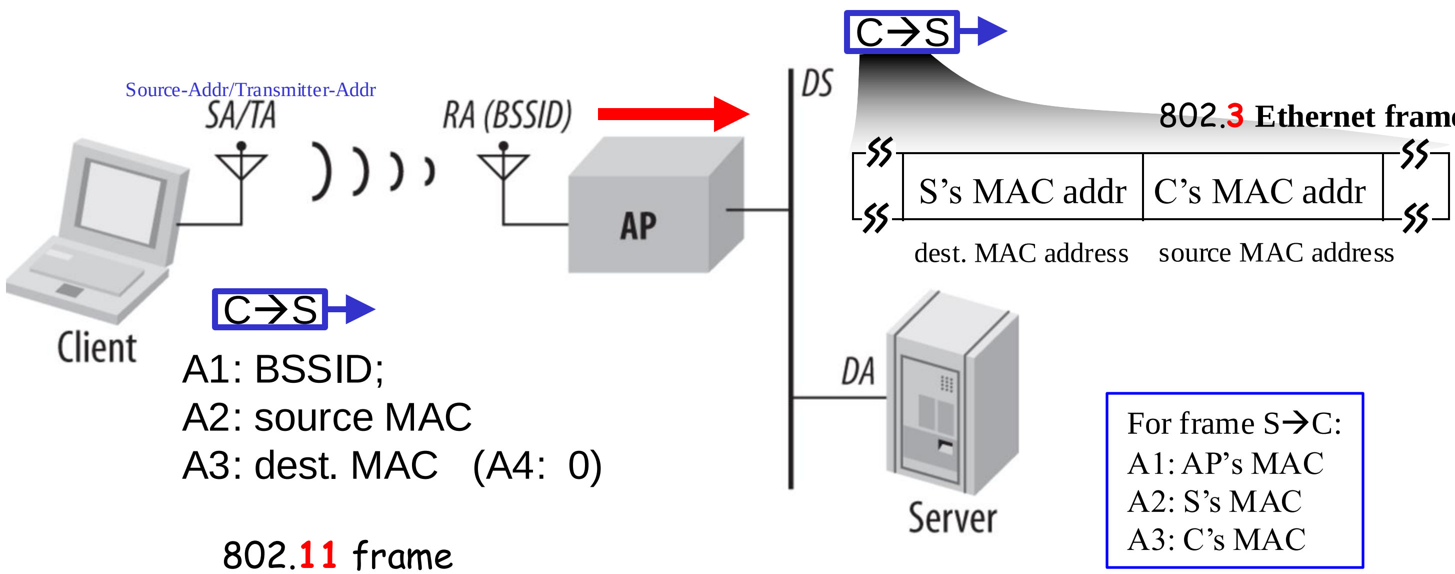
2	2	6	6	6	2	6	0 - 2312	4
frame control	duration	address 1	address 2	address 3	seq control	address 4	payload	CRC

Address 1: MAC address of wireless host or AP to receive this frame

Address 2: MAC address of wireless host or AP transmitting this frame

Address 3: MAC address of next IP node interface connected to AP

Address 4: used only in ad hoc mode (optional)



802.11 frame

Summary of MAC protocols

- channel partitioning
  - Time Division, Frequency Division
- taking turns
  - polling from central site, or token passing
- random access
  - ALOHA, Slotted-ALOHA
  - CSMA: Carrier Sensing in Multi-Access: easy in some case (wire), harder in others (wireless)
  - CSMA/CD used in Ethernet
  - CSMA/CA used in 802.11
  - Why