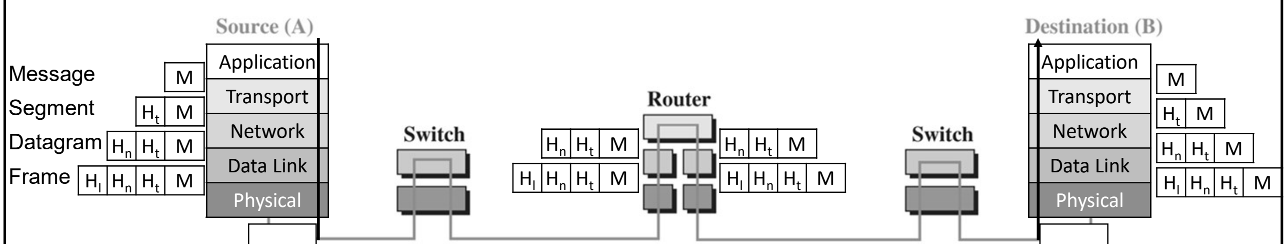
Welcome!

ELEC 8560 – Computer Networks

Data Link Layer

1

Recall: End-to-End Communication via Internet



2

Outline

- Communication at the data link layer
- Data link control
- Media access control
- Addressing and address resolution

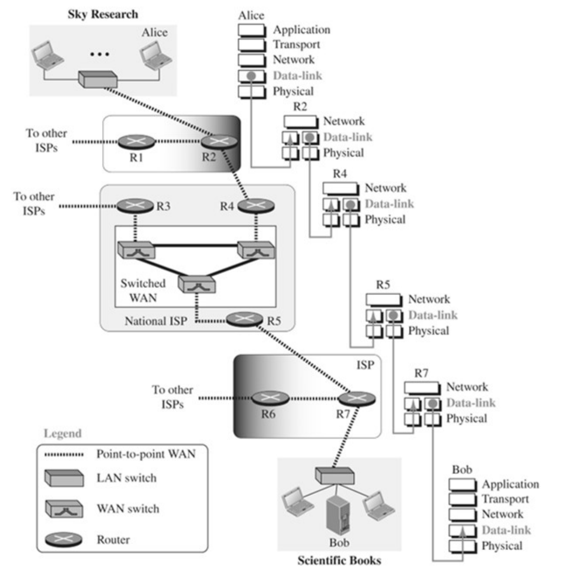
- Recommended reading: Forouzan – Chapter 3
- Extra reading: Kurose and Ross – Chapter 6

Outline

- Communication at the data link layer
- Data link control
- Media access control
- Addressing and address resolution

Communication at the Data Link Layer

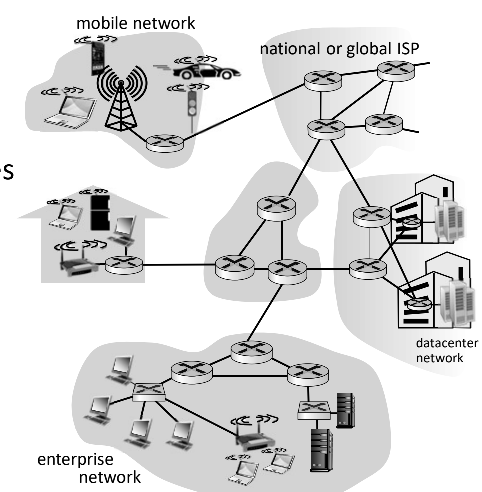
- Data link layer provides services to the network layer and receives services from the physical layer
  - Transport datagrams to adjacent node
  - Sender: encapsulates datagram from network layer into frames, passes to physical layer
  - Receiver: reassemble frames into datagrams, delivers to network layer



Data Flow Diagram

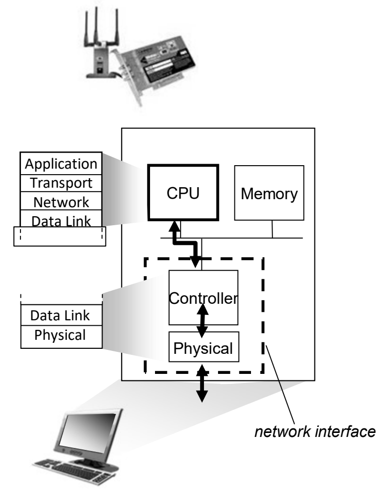
Nodes and Links

- Communication at the data link layer is node-to-node
  - Nodes: hosts and routers
  - Links: connections between nodes
    - Point-to-point link: dedicated to two devices
    - Broadcast link: shared between several pairs
    - Example: landline phones vs. cell phones



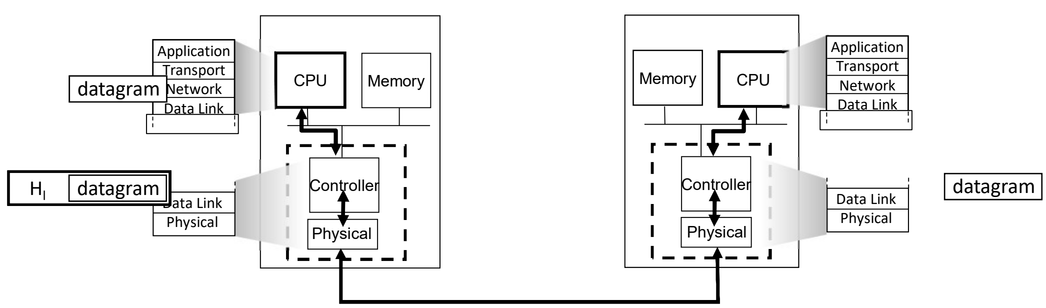
Where is The Link Layer Implemented?

- Data link layer is in each-and-every host
- Data link and physical layers are implemented in Network Interface Card (NIC) or on a chip
- Combination of hardware, software, firmware



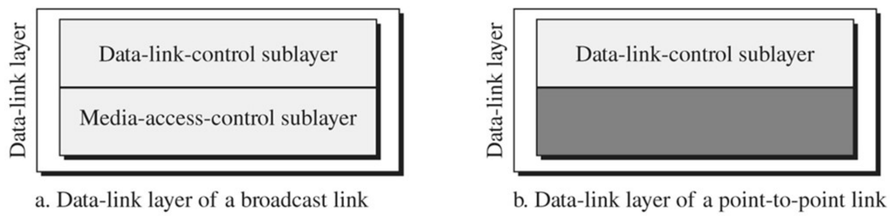
Interfaces Communicating

- Sender:
  - Encapsulates datagram in frame
  - Adds headers, trailers, flow control, etc.
- Receiver:
  - Checks for errors, etc.
  - Extracts datagram and passes to upper layers



Two Sublayers

- Data link layer can be divide into two sublayers:
  - Data Link Control (DLC): procedures for communication between two adjacent nodes whether link is dedicated or broadcast
  - Media Access Control (MAC): channel access control mechanisms to enable nodes to communicate in a network (i.e., broadcast links)



Outline

- Communication at the data link layer
- Data link control
- Media access control
- Addressing and address resolution

Data Link Control (DLC)

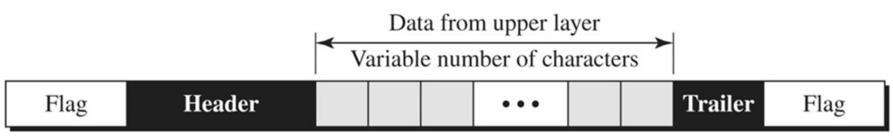
- DLC services:
  - Framing: pack bits into frames
  - Error control: detection and correction of errors and manage retransmissions
  - Flow control: restrict the amount of data the sender can send before waiting for an acknowledgement

Outline

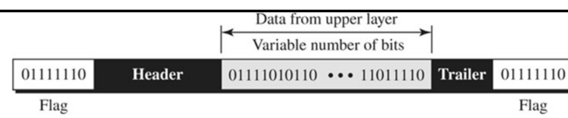
- Communication at the data link layer
- Data link control
  - Framing
- Media access control
- Addressing and address resolution

Framing

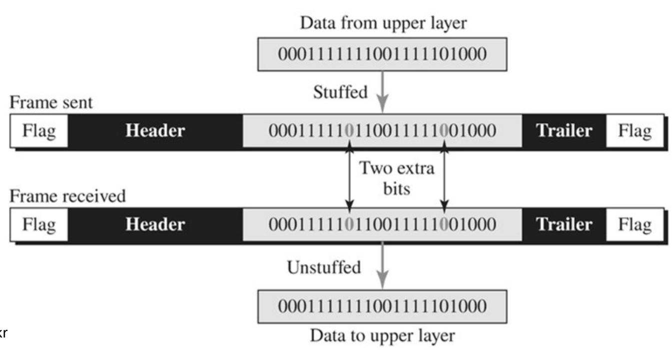
- Encapsulating datagrams into frames and adding header and trailer
- Frames can be fixed or variable size
- For variable-size frames:
  - Need to define the boundary of the frame if size is variable
  - Two approaches are used for variable-size framing: bit-oriented or character-oriented



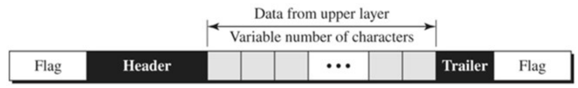
Bit-Oriented Framing



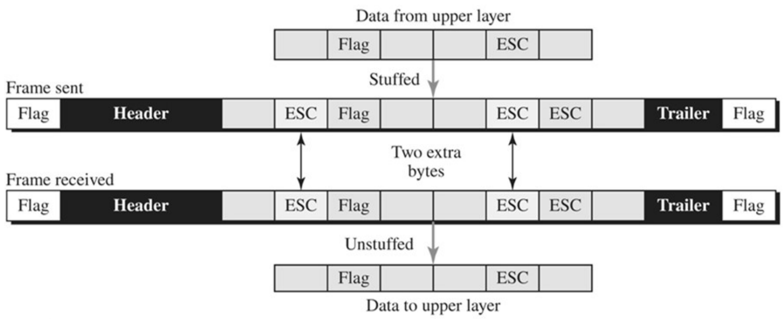
- Data is a sequence of bits + an 8-bit flag (01111110) as a delimiter
- Bit-stuffing to prevent a byte to be interpreted as beginning or end of the message
  - adding one extra 0 whenever five consecutive 1s follow a 0 in the data, so that the receiver does not mistake the pattern for a flag



Character-Oriented Framing



- Data is a sequence of 8-bit characters (e.g., ASCII code) + a special character flag as a delimiter
- Byte-stuffing to prevent a special character to be interpreted as beginning or end of the message
  - adding 1 extra byte whenever there is a flag or escape character in the text

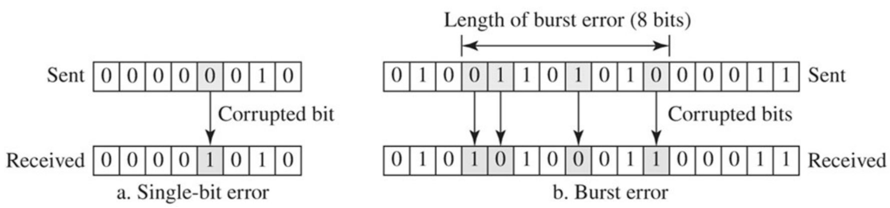


Outline

- Communication at the data link layer
- Data link control
  - Error Control
- Media access control
- Addressing and address resolution

Single-bit vs. Burst Error

- Data unit: a byte, character, or packet
- Single-bit error: only 1 bit of a given data unit is changed from 1 to 0 or from 0 to 1
- Burst error: 2 or more bits in the data unit have changed from 1 to 0 or from 0 to 1
  - does not necessarily mean errors occur in consecutive bits
  - more likely to occur than a single-bit error

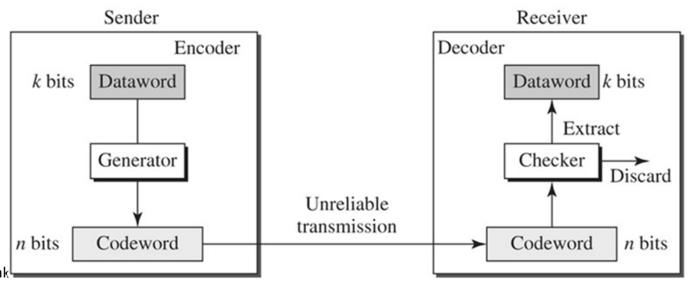


Error Control: Detection vs. Correction

- Errors may be caused by signal attenuation, noise, interference, etc.
  - Unpredictable, change the shape of the signal
  - To detect or correct errors, we need to send extra (redundant) bits with data
- Error detection:
  - Receiver checks if any errors occurred
  - Not interested in the number of corrupted bits
  - Ask for retransmission or drops frame
- Error correction:
  - Receiver identifies and corrects bit error without retransmission
  - Know exact number of corrupted bits and their location
  - More difficult

Block Coding

- Main concept in detecting or correcting errors is redundancy
- Block Coding:
  - Sender divides message into blocks of *k* bits → datawords
  - Sender adds *r* redundant bits that are based on actual data bits → codewords of length *n=k+r*
  - Receiver checks the relationship to detect or correct corrupted bits
  - Higher ratio of redundant to data bits yields better performance



Example: Block Coding

k=2 and *n=3*

Datawords	Codewords
00	000
01	011
10	101
11	110

- Table shows a list of datawords and codewords
- Sender encodes dataword 01, sends codeword 011
- Case I: Receiver gets 011
  - It is a valid codeword, receiver extracts the dataword 01
- Case II: Corrupted during transmission, and 111 is received
  - Not a valid codeword and is discarded
- Case III: Corrupted during transmission, and 000 is received
  - Valid codeword, receiver incorrectly extracts the dataword 00
  - Two corrupted bits have made the error undetectable

How many errors a code can detect or correct?

Hamming Distance

- Very important in coding theory for error control
- Hamming distance between two words (of the same size) is the number of differences between the corresponding bits

d(2173896, 2233796) = 3

d(0001, 0010) = 2

- For binary strings *a* and *b*, it is equal to the number of 1s in *a XOR b*

0001 XOR 0010 = 0011 → **d**(0001, 0010) = 2

10001 XOR 00010 = 10011 → **d**(10001, 00010) = 3

Minimum Hamming Distance

- A code is said to be *k*-error detecting if, and only if, the minimum Hamming distance between any two of its codewords is at least *k*+1

$$d_{min} \geq k+1$$

- A code is said to be *k*-error correcting if, and only if, the minimum Hamming distance between any two of its codewords is at least 2*k*+1

$$d_{min} \geq 2k+1$$

Example 1: Minimum Hamming Distance

What is the minimum Hamming distance of the code in the table below?

Solution:

- Minimum Hamming distance is 2
- This code guarantees detection of only a single error
- If two errors occur, the received codeword may match a valid codeword and the errors are not detected

Datawords	Codewords
00	000
01	011
10	101
11	110

Example 2: Minimum Hamming Distance

A code scheme has a Hamming distance d_{min} = 4. What is the error detection and correction capability of this scheme?

Solution:

- This code guarantees the detection of up to three errors (i.e., 4 ≥ *k*+1)
- It can correct up to one error (i.e., 4 ≥ 2k+1)
- Note: if this code is used for error correction, part of its capability is wasted
  - Error correction codes need to have an odd minimum distance (3, 5, 7, . . .)

Linear Block Codes

- Linear block code is a code in which the XOR of two valid codewords creates another valid codeword
- Minimum Hamming distance is the number of 1s in the nonzero valid codeword with the smallest number of 1s
- Almost all block codes used today are linear block code
  - Parity Check Codes
  - Cyclic Codes

Example: Linear Block Codes

Is the code in the table below a linear block code? If so, find d_{min} using no. of 1s.

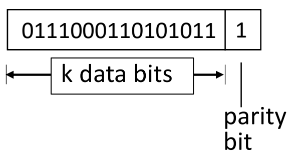
Solution:

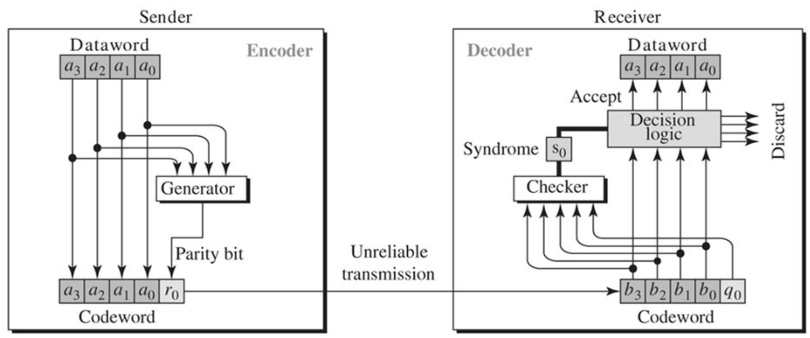
- Code is a linear block code because the result of XORing any codeword with any other codeword is a valid codeword
- The numbers of 1s in the nonzero codewords are 2, 2, and 2, so the minimum Hamming distance is 2

Datawords	Codewords
00	000
01	011
10	101
11	110

Parity Check Code

- The most familiar error-detecting code (linear block code)
- A *k*-bit dataword is changed to an *n*-bit codeword where *n=k+1*
- Extra bit, called the parity bit, is selected to make the total number of 1s in the codeword even





Example: Parity Check Code

Assume sender encodes dataword 1011. Show all possible cases for error detection.

Solution:

Sender encodes dataword 1011 and sends codeword 10111:

- Case I – No error: Receiver gets 10111. Syndrome is 0. Dataword 1011 created.
- Case II – One single-bit error: Receiver gets 10011 is received. Syndrome is 1. No dataword created. Same if parity bit is corrupted.
- Case III – Two single-bit error: Receiver gets 11011 is received. Syndrome is 0. Receiver incorrectly create the dataword 1101.
- Case IV – Three single-bit error: Receiver gets 11010 is received. Syndrome is 1. No dataword created.

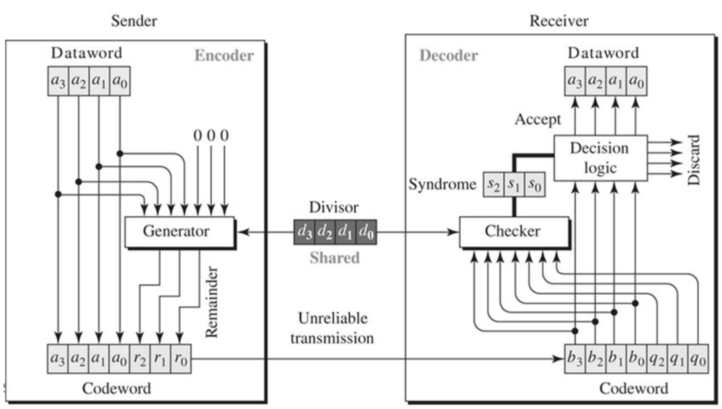
A parity-check code can detect an odd number of errors

Cyclic Codes

- Cyclic codes are special linear block codes with one extra property:
  - If a codeword is cyclically shifted (rotated), the result is another codeword
  - Example: if 1011000 is a codeword and we cyclically left-shift, then 0110001 is also a codeword
- Can detect single-bit errors, double errors, an odd number of errors, and some burst errors
- Can easily be implemented in hardware (faster) and software
- Cyclic Redundancy Check (CRC) codes, a subset of cyclic code, widely used in current networks such as Ethernet and Wi-Fi

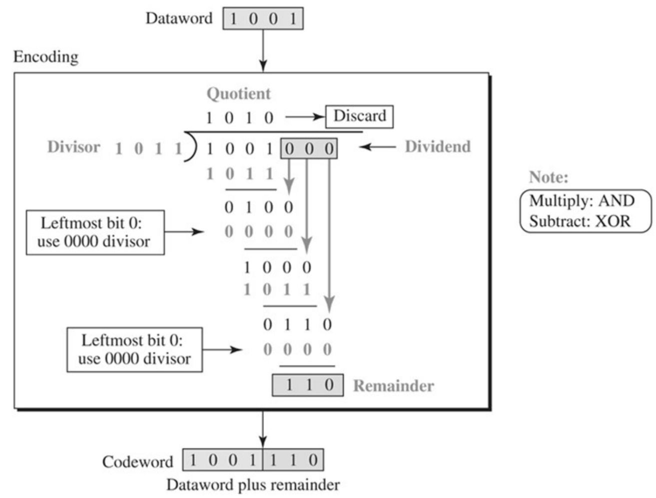
Cyclic Redundancy Check (CRC)

- More powerful error-detecting code (burst errors ≤ *r* bits guaranteed)
- Goal:
  - Given: *k*-bit dataword and *(r+1)*-bit generator (divisor or pattern)
  - Generate *r*-bit CRC bits and create *n*-bit codeword



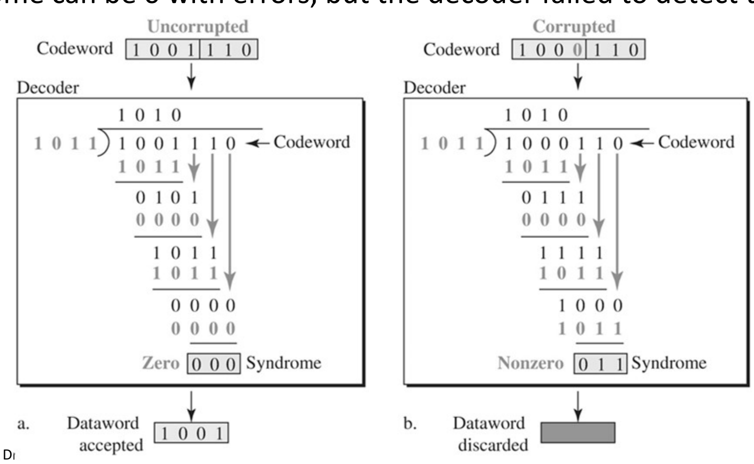
Example: CRC Encoder

- Sender knows generator → divides $2^r \cdot$ dataword by generator → remainder is CRC bits

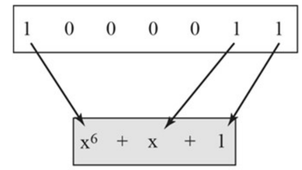


Example: CRC Decoder

- Receiver knows generator → divides codeword by generator → if syndrome/remainder is non-zero → error detected
  - Syndrome can be 0 with errors, but the decoder failed to detect them



Standard Generator Polynomials



- Generators must (for all single errors can be caught):
  - have at least two bits
  - have 1s in the rightmost and leftmost bits
- Common generators:

Name	Binary	Application
CRC-8	100000111	ATM header
CRC-10	11000110101	ATM AAL
CRC-16	10001000000100001	HDLC
CRC-32	100000100110000010001110110110111	LANs

Two DLC Protocols

- The data link layer combines framing, flow control, and error control to achieve the delivery of data from one node to another
- Two DCL protocols actually implement the concepts discussed so far:
  - High-level Data Link Control (HDLC) protocol:
    - Bit-oriented protocol
    - Used for communication over point-to-point and multipoint links
  - Point-to-Point Protocol (PPP):
    - Byte-oriented protocol
    - One of the most common protocols for point-to-point access
    - Used by an Internet service provider (ISP) to provide several services

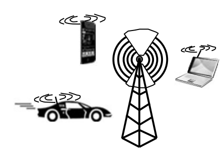
Outline

- Communication at the data link layer
- Data link control
- Media access control
- Addressing and address resolution

Media Access Control (MAC)

- Access control is needed in broadcast (shared) links
  - single shared broadcast channel
  - two or more simultaneous transmissions by nodes → **collision**
- MAC protocol is distributed algorithm that determines how nodes share channel, i.e., determine when node can transmit
- MAC protocols classes:
  - Random Access: random, allow collision
  - Controlled Access: taking turns, more data more wait
  - Channelization: partition channels, exclusive use

shared wire (e.g., cabled Ethernet)



shared radio: 4G/5G

shared radio: Wi-Fi

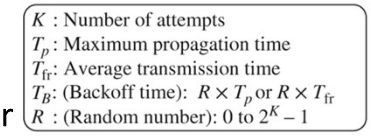
Outline

- Communication at the data link layer
- Data link control
- Media access control
  - Random Access
- Addressing and address resolution

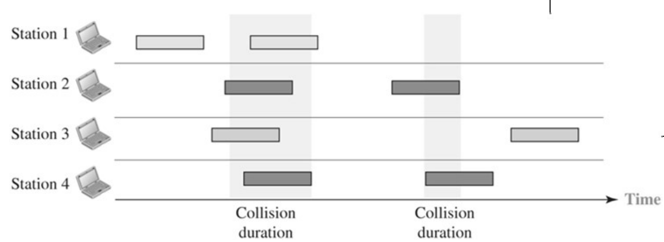
Random Access

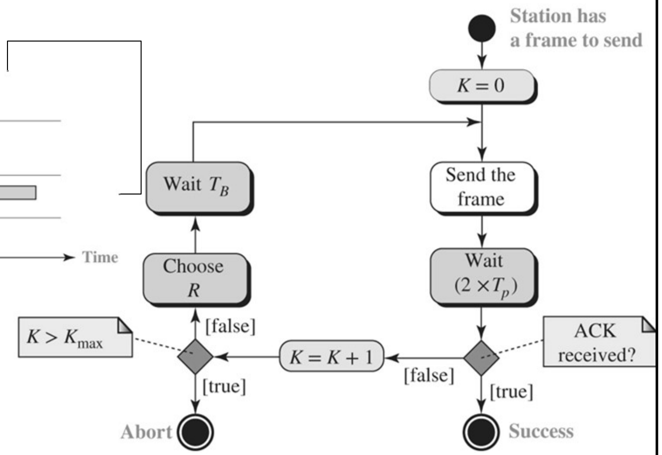
- No node is superior and none is assigned control over another
- When node has packet to send:
  - transmit at full channel data rate
  - no a priori coordination among nodes
- Two or more transmitting nodes: collision
- Random Access MAC protocol specifies:
  - How to detect collisions
  - How to recover from collisions (e.g., via delayed retransmissions)
- Examples:
  - ALOHA, Slotted ALOHA
  - CSMA, CSMA/CD, CSMA/CA

Pure ALOHA



- Node has a frame → send immediately and wait for ACK from the receiver





- Why is R random?

Example: Pure ALOHA

The stations on a wireless ALOHA network are a maximum of 600 km apart. Assume signals propagate at 3×10^8 m/s and second failed attempt to transmit, what are the possible values of the T_B?

Solution:

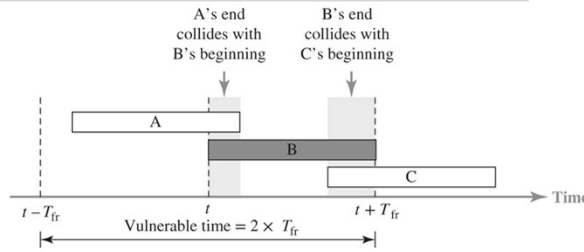
- First, we find $T_p = (600 \times 10^3)/(3 \times 10^8) = 2$ ms
- For K = 2, the range of R is {0, 1, 2, 3}
- This means that T_B can be 0, 2, 4, or 6 ms, based on the outcome of the random variable R

Example: Vulnerable Time for Pure ALOHA

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the requirement to make this frame collision-free?

Solution:

- Average frame transmission time T_{fr} is 200 bits/200 kbps = 1 ms
- The vulnerable time is 2 × 1 ms = 2 ms
- This means no station should send later than 1 ms before this station starts transmission and no station should start sending during the period (1 ms) that this station is sending



Throughput of Pure ALOHA Networks

- Let *G* be the average number of frames generated by the network during one frame transmission time
- Throughput is the average number of successfully transmitted frames

$$S = G \times e^{-2G}$$

- Maximum throughput of pure ALOHA is 18% when *G=0.5 !*

Example: Throughput

A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces: (a) 1000 frames per second, (b) 500 frames per second, and (c) 250 frames per second?

Solution:

- The frame transmission time is 200/200 kbps = 1 ms

a. If the system creates 1000 frames per second, or 1 frame per millisecond, then G = 1
   - In this case $S = G \times e^{-2G}$ = 0.135 (or 13.5%) → Throughput = 1000 × 0.135 = 135 frames per sec
   - Only 135 frames out of 1000 will probably survive

b. If the system creates 500 frames per second, or 0.5 frame per millisecond, then G = 0.5
   - In this case $S = G \times e^{-2G}$ = 0.184 (or 18.4%) → Throughput = 500 × 0.184 = 92 frames per sec
   - Only 92 frames out of 500 will probably survive

Example: Throughput (cont.)

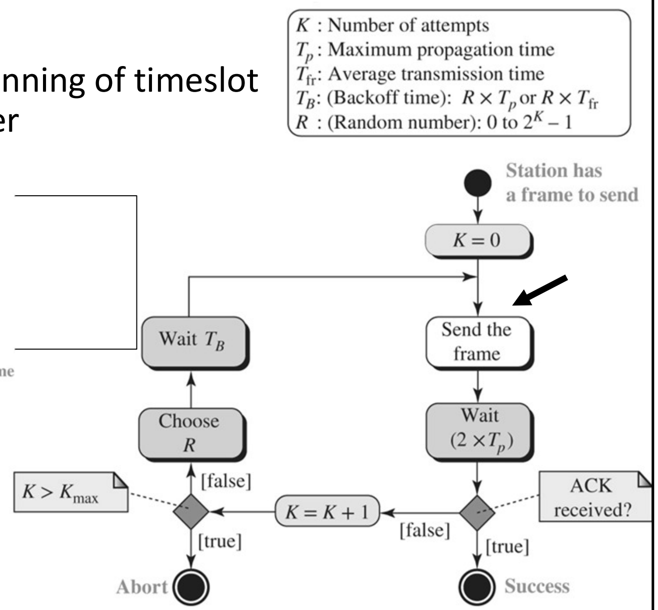
A pure ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces: (a) 1000 frames per second, (b) 500 frames per second, and (c) 250 frames per second?

Solution:

c. If the system creates 250 frames per second, or 0.25 frame per millisecond, then G = 0.25
   - In this case $S = G \times e^{-2G}$ = 0.152 (or 15.2%) → Throughput = 250 × 0.152 = 38 frames per sec
   - Only 38 frames out of 250 will probably survive

Slotted ALOHA

- Node has a frame → send at beginning of timeslot and wait for ACK from the receiver



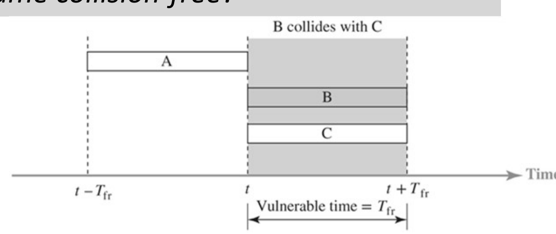
- Synchronization is important

Example: Vulnerable Time for Slotted ALOHA

A slotted ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the requirement to make this frame collision-free?

Solution:

- Average frame transmission time T_{fr} is 200 bits/200 kbps = 1 ms
- The vulnerable time is 1 × 1 ms = 1 ms
- This means no station should start sending at the same timeslot this station is sending



Throughput of Slotted ALOHA Networks

- Let G be the average number of frames generated by the network during one frame transmission time
- Throughput is the average number of successfully transmitted frames

$$S = G \times e^{-G}$$

- Maximum throughput of pure ALOHA is 36% when *G=1 !*

Example: Throughput

A slotted ALOHA network transmits 200-bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces: (a) 1000 frames per second, (b) 500 frames per second, and (c) 250 frames per second?

Solution:

- The frame transmission time is 200/200 kbps = 1 ms

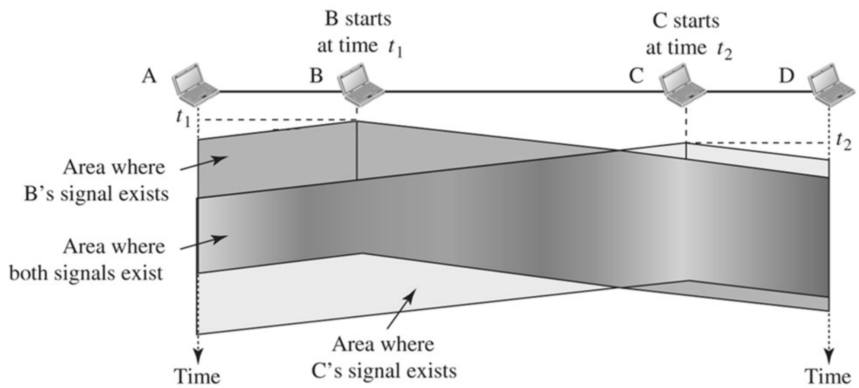
a. If the system creates 1000 frames per second, the G = 1
   - In this case $S = G \times e^{-G} = 0.368$ (or 36.8%) → Throughput = 1000 × 0.368 = 368 frames per sec
   - Only 368 frames out of 1000 will probably survive

b. If the system creates 500 frames per second, then G = 0.5
   - In this case $S = G \times e^{-G} = 0.303$ (or 30.3%) → Throughput = 500 × 0.303 = 151 frames per sec
   - Only 151 frames out of 500 will probably survive

c. If the system creates 250 frames per second, then G = 0.25
   - In this case $S = G \times e^{-G} = 0.195$ (or 19.5%) → Throughput = 250 × 0.195 = 49 frames per sec
   - Only 49 frames out of 250 will probably survive

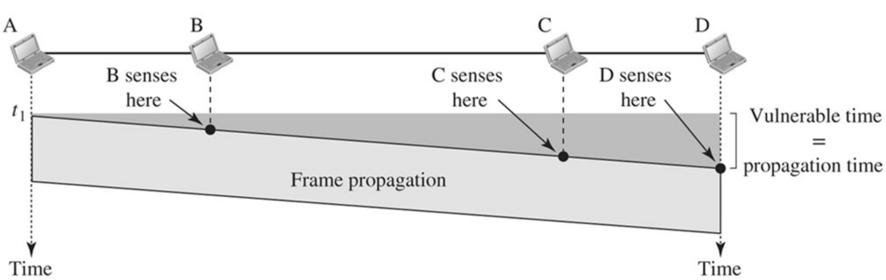
Carrier Sense Multiple Access (CSMA)

- To reduce chances of collision, listen before transmit:
  - If channel sensed idle: transmit entire frame
  - If channel sensed busy: defer transmission
- Space and time model of a collision due to propagation delay



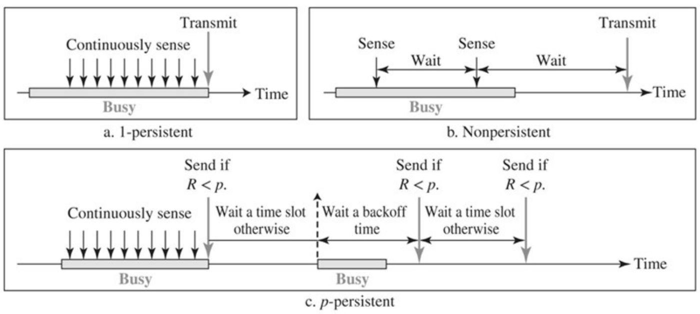
Vulnerable Time in CSMA

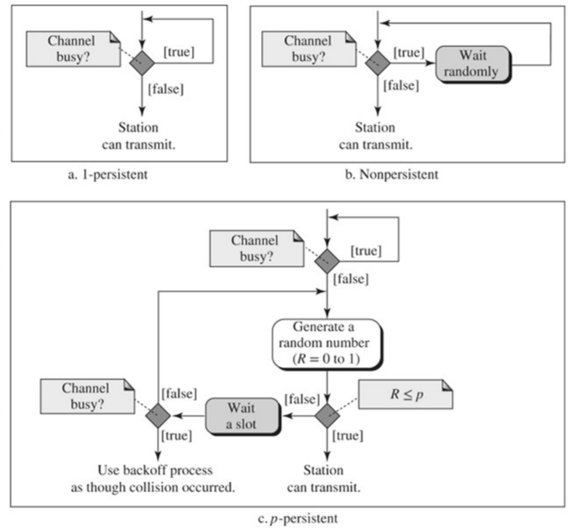
- Distance and propagation delay play role in in determining collision probability



What Happens if Channel is Busy?

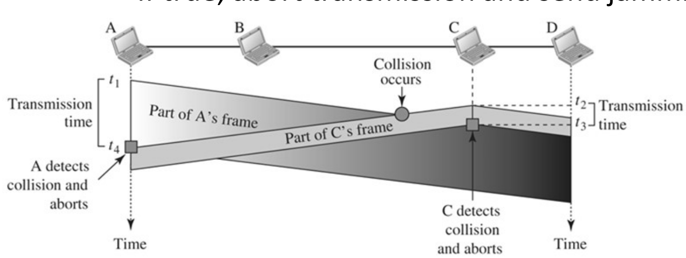
- Persistence methods

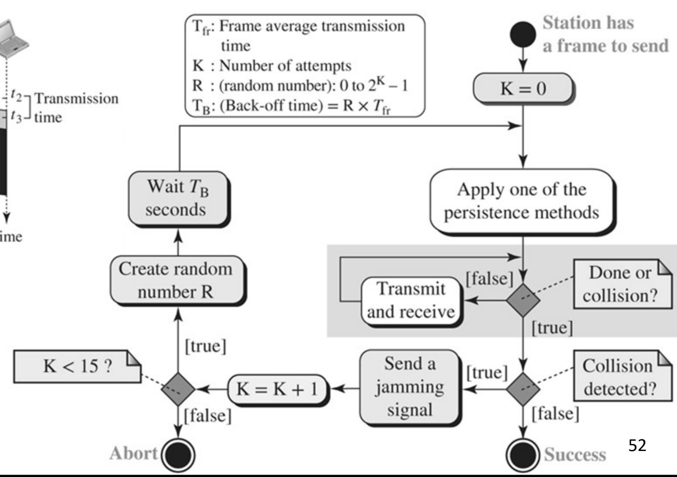




CSMA with Collision Detection (CSMA/CD)

- Keep monitoring medium (energy level) while transmitting to detect collisions
  - If true, abort transmission and send jamming signal on collision detection





- Saves time!

Minimum Frame Size

- Frame size has to be restricted to a minimum size to make sure nodes detect collisions, if any, before sending the last bit of the frame
- Scenario:
  - Nodes A and B are the maximum distance apart
  - Node A transmits at time *0*
  - Node B transmits at time T_p
  - So the effect of a collision may take up to $2T_p$ sec
  - If node A finishes transmission before that, it will stop monitoring the channel and clear the frame
- So, the minimum frame duration T_{fr} is $2T_p$ sec and size is $2T_p \times R$

Example: CSMA/CD

A network using CSMA/CD has a bandwidth of 10 Mbps. If the maximum propagation time is 25.6 μs, what is the minimum size of the frame?

Solution:

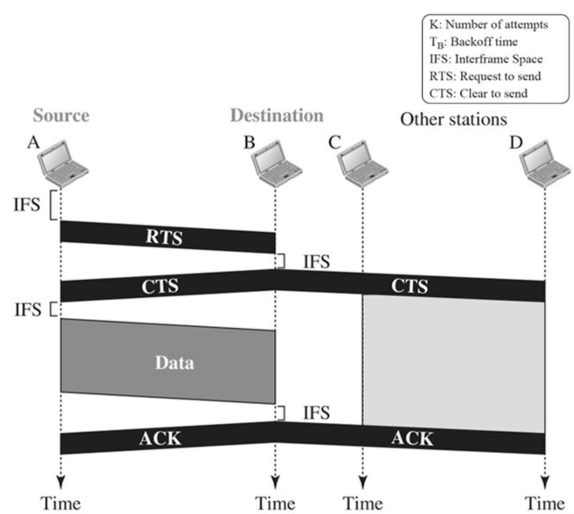
- The minimum frame transmission time is $T_{fr} = 2 \times T_p = 51.2$ μs
- This means, in the worst case, a station needs to transmit for a period of 51.2 μs to detect the collision
- The minimum size of the frame is 10 Mbps × 51.2 μs = 512 bits or 64 bytes
  - This is actually the minimum size of the frame for Standard Ethernet as we will see

CSMA/CD in Standard Ethernet

1. NIC receives datagram from network layer, creates frame
2. NIC senses channel:
   - If idle: start frame transmission
   - If busy: wait until channel idle, then transmit (i.e., 1-persistent)
3. If NIC transmits entire frame without collision, NIC is done
4. If NIC detects another transmission while sending: abort, send jamming signal
5. After aborting, NIC enters binary (exponential) backoff:
   - After K-th collision, NIC chooses *R* at random, NIC waits $R\ T_{fr}$, returns to Step 2
   - More collisions: longer backoff interval

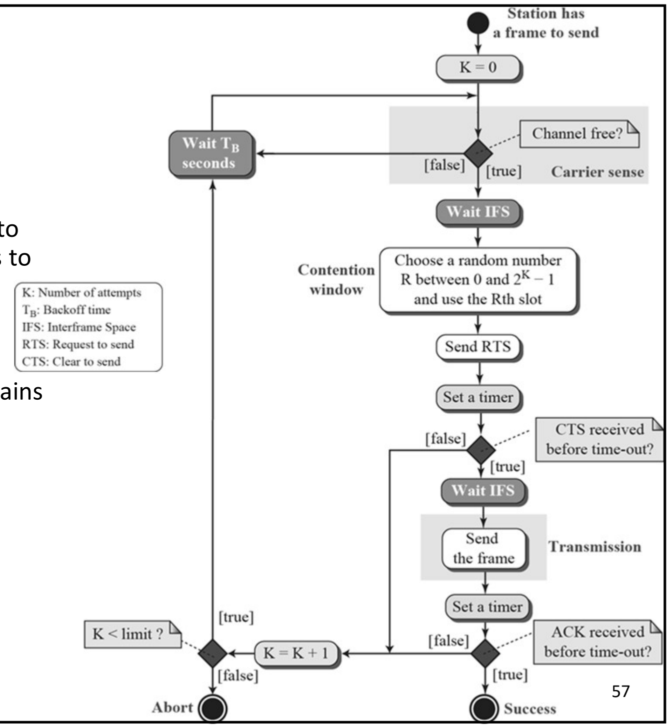
CSMA with Collision Avoidance (CSMA/CA)

- Invented for wireless networks to avoid collisions
- Idea: reserve channel
  - Sender transmits small RTS packet
  - Receiver broadcasts CTS in response
  - All nodes defer transmissions and Sender transmits data frame



CSMA/CA (cont.)

- Uses three strategies:
  - Interframe Space (IFS)
    - Wait sometime before transmission to allow messages from distant stations to arrive, if any
  - Contention window
    - Wait for some random time before transmission to check if channel remains free
  - Acknowledgments



Outline

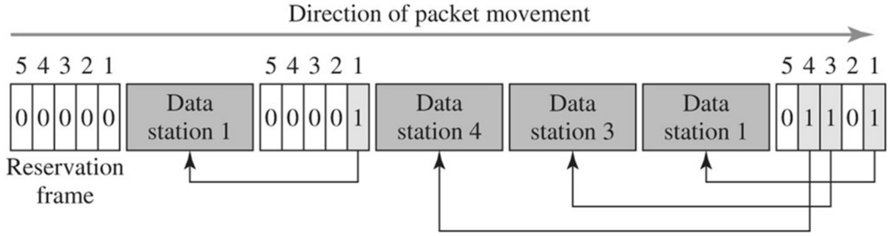
- Communication at the data link layer
- Data link control
- Media access control
  - Controlled Access
- Addressing and address resolution

Controlled Access

- Stations consult one another to find which station has the right to send
- A station cannot send unless it has been authorized by other stations
- Examples:
  - Reservation
  - Polling
  - Token Passing

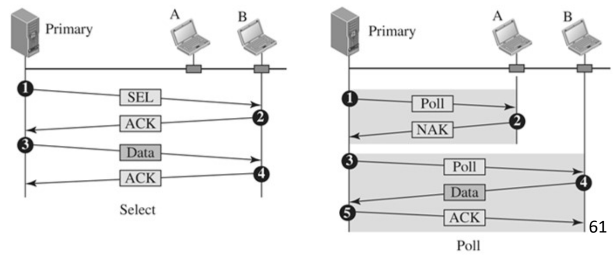
Reservation

- A station needs to make a reservation before sending data
- Time is divided into intervals
- In each interval, a reservation frame precedes the data frames sent in that interval
  - When a station wants to transmit, it makes a reservation in an assigned slot



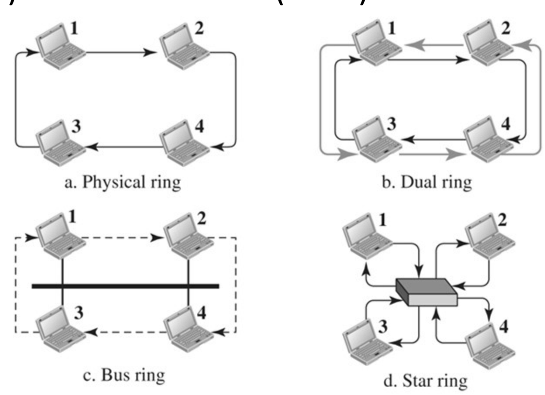
Polling

- One device is designated as a primary station and other devices are secondary stations
- The primary device controls the channel and determine which device is allowed to use the channel at a given time
- Secondary devices must follow its instructions for all data exchanges
  - Select function by primary is used when it needs to send data to some secondary device
  - Poll function is used by the primary device to solicit transmission from the secondary devices



Token Passing

- Stations in a network are organized in a logical ring
- Each station has a predecessor (before) and a successor (after)
- A special packet called a token circulates through the ring
- The possession of the token gives the station the right to access the channel and transmit its data



Outline

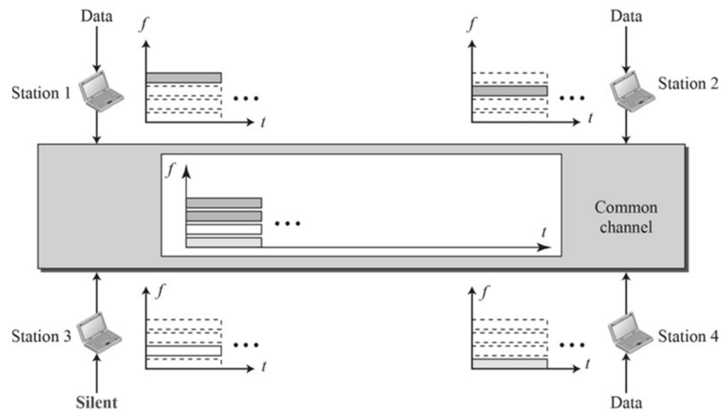
- Communication at the data link layer
- Data link control
- Media access control
  - Channelization
- Addressing and address resolution

Channelization

- Sometimes called channel partitioning
- A multiple-access method in which the available bandwidth of a link is shared in time, frequency, or through code, among different stations
- Used in wireless LAN
- Examples:
  - Frequency Division Multiple Access (FDMA)
  - Time Division Multiple Access (TDMA)
  - Code Division Multiple Access (CDMA)

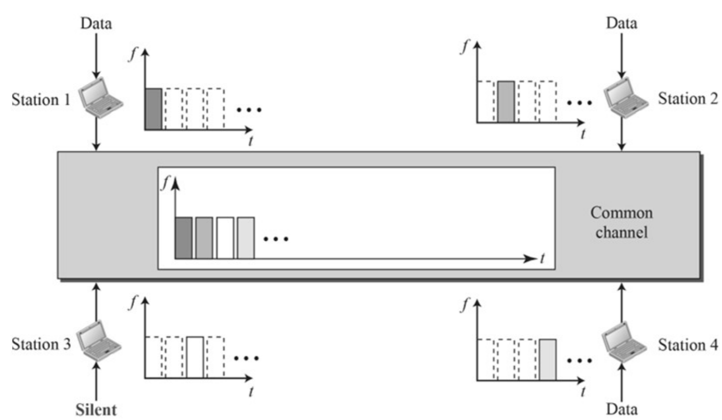
Frequency Division Multiple Access (FDMA)

- Available bandwidth is divided into frequency bands (+ guard bands)
- Each station is allocated a band to send its data simultaneously



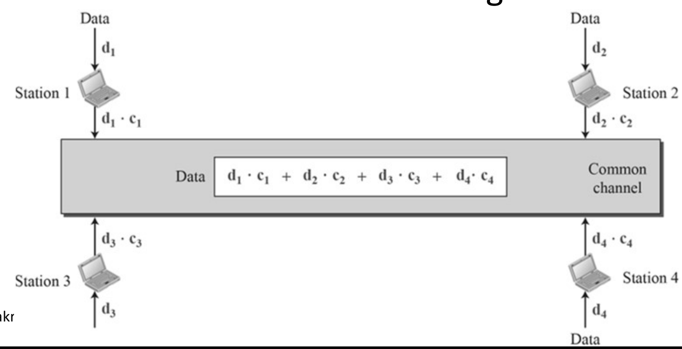
Time Division Multiple Access (TDMA)

- Stations share the entire channel bandwidth in time and take rounds
- Each station is allocated a timeslot during which it can send data

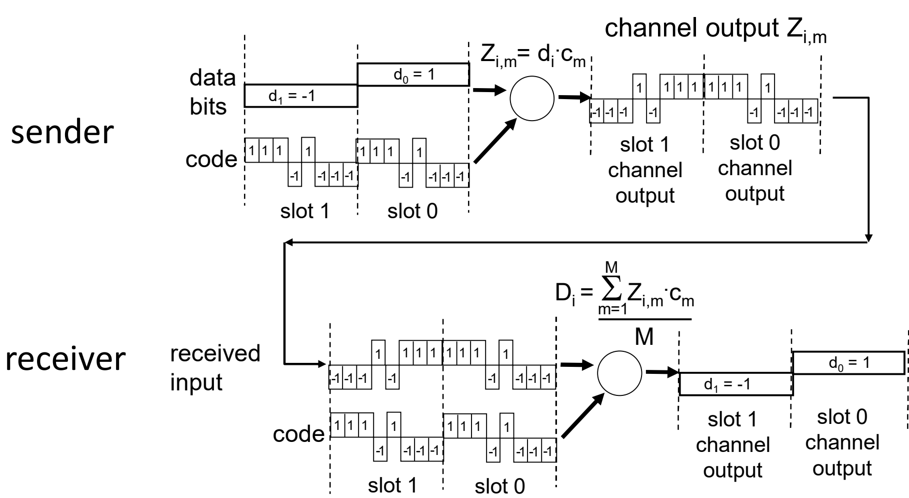


Code Division Multiple Access (CDMA)

- All stations share the entire channel bandwidth and can send data simultaneously as they are separated in code
  - Encoding: each station multiplies its data by its code before transmitting
  - Decoding: a station can detect the data sent by another station using its code
- That is, multiple users coexist and transmit simultaneously with minimal interference when codes are orthogonal

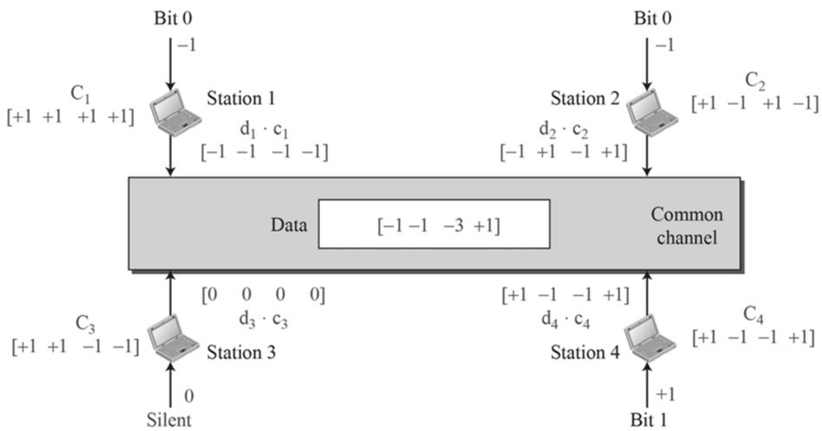


Example: Encoding and Decoding in CDMA



Example: Sharing Channel in CDMA

- Note that the multiplication of two different sequences, element by element and adding the results yield 0

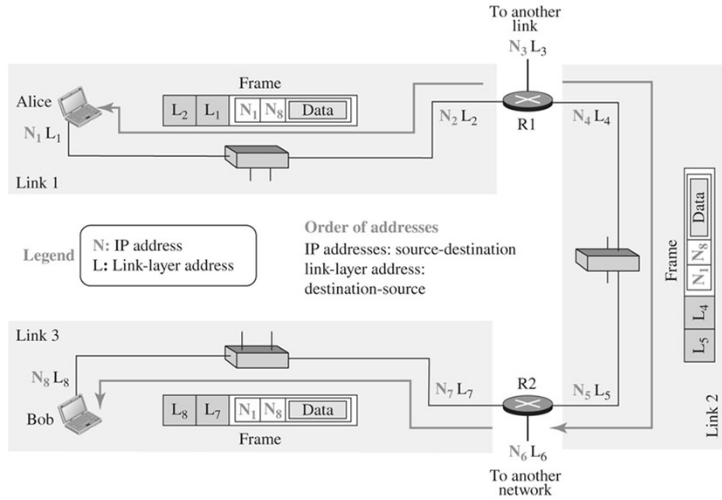


Outline

- Communication at the data link layer
- Data link control
- Media access control
- Addressing and address resolution

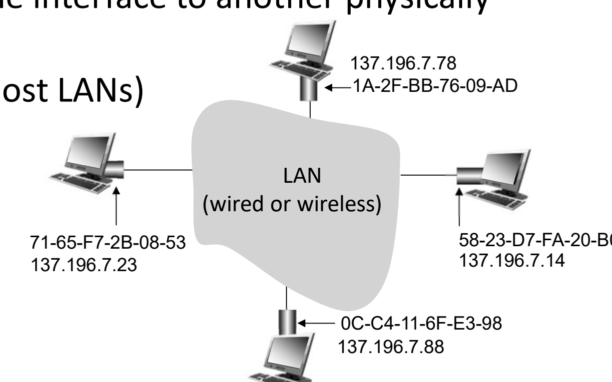
Link Layer Addressing

- In an internetwork such as the Internet we cannot make a datagram reach its destination using only IP addresses
- Source and destination IP addresses define the two ends but cannot define which links the packet should pass through



MAC Addresses

- Also called LAN, Physical, or Ethernet address
- Each interface on LAN has a unique MAC address (paired with a local IP address)
- Used locally to get a frame from one interface to another physically-connected interface
- 48-bit (6-byte) MAC address (for most LANs) burned in NIC ROM

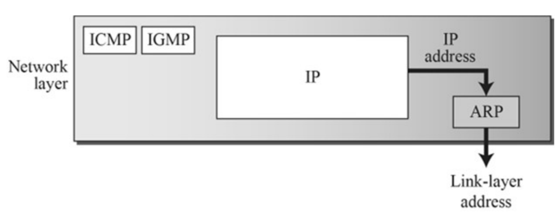


MAC Addresses (cont.)

- Some link-layer protocols define three types of addresses:
  - Unicast Address: Each interface is assigned a unicast address
    - Example: Ethernet **A2-34-45-11-92-F1**
  - Multicast Address: Means one-to-many communication
    - Example: Ethernet **47-20-1B-2E-08-EE**
  - Broadcast Address: Means one-to-all address
    - Example: Ethernet **FF-FF-FF-FF-FF-FF**

Address Resolution Protocol (ARP)

- How to determine interface MAC address, knowing its IP address
- Any time a node has a packet to send to another node, it has its IP address and needs the link-layer address of the next node
- ARP is located in the network layer



ARP Table

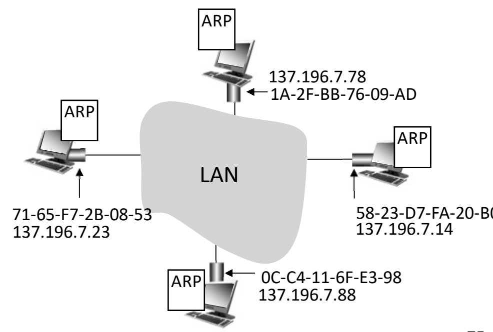
- Each IP node (host, router) on LAN has table contains IP/MAC address mappings for some LAN nodes

  < IP address; MAC address; TTL>

- TTL (Time To Live) is the time after which address mapping will be forgotten (typically 20 min)

```
C:\Users\admin>arp -a

Interface: 192.168.2.10 --- 0xe
  Internet Address      Physical Address      Type
  192.168.2.1           2c-6d-b2-65-18-8c     dynamic
  192.168.2.255         ff-ff-ff-ff-ff-ff     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```



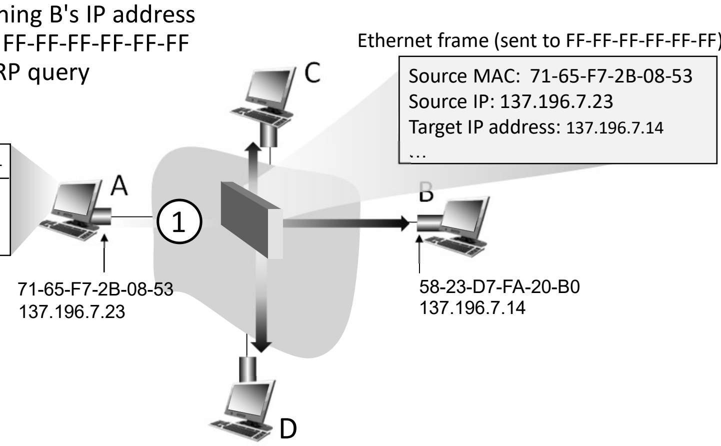
ARP Operation

- Node A wants to send datagram to node B
- Use ARP if MAC address of B is not in ARP table of A

1 A broadcasts ARP query, containing B's IP address
- Destination MAC address = FF-FF-FF-FF-FF-FF
- All nodes on LAN receive ARP query

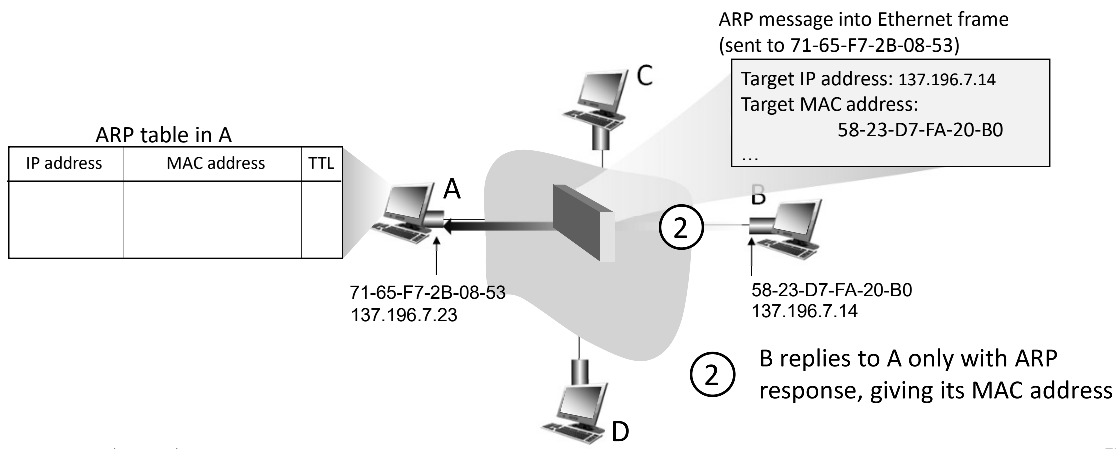
ARP table in A

IP address	MAC address	TTL



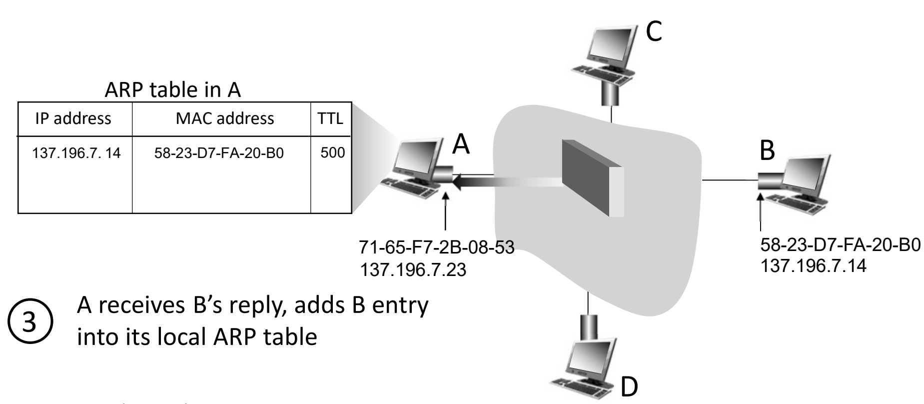
ARP Operation (cont.)

- Node A wants to send datagram to node B
- MAC address of B is not in ARP table of A → use ARP



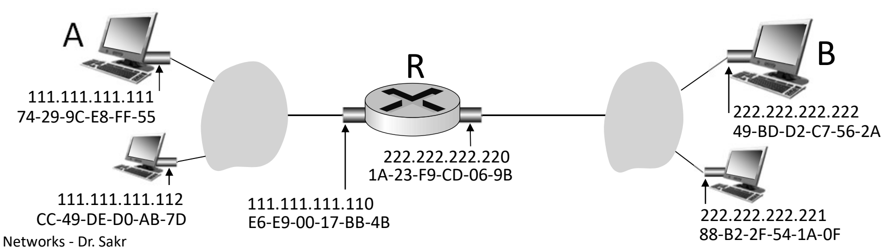
ARP Operation (cont.)

- Node A wants to send datagram to node B
- MAC address of B is not in ARP table of A → use ARP



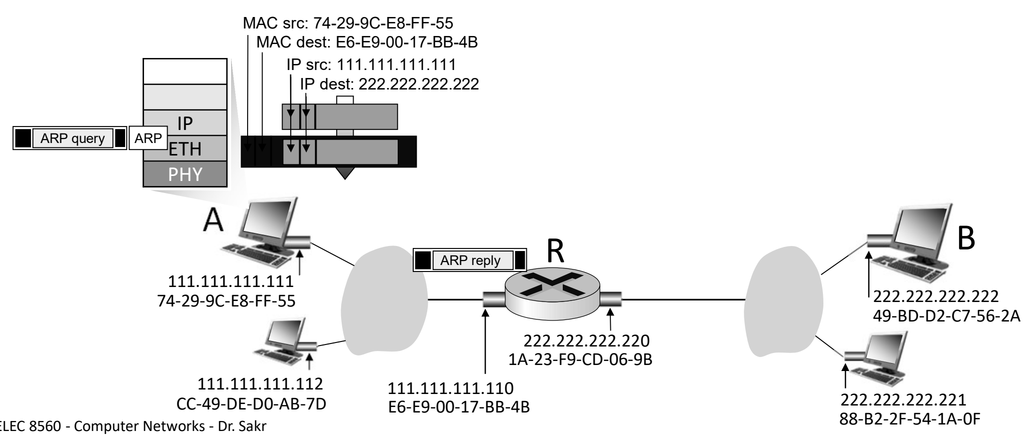
Example: Flow of Packets over Internetworks

- Node A wants to send datagram to node B
- A knows IP address of R and B (more details later)



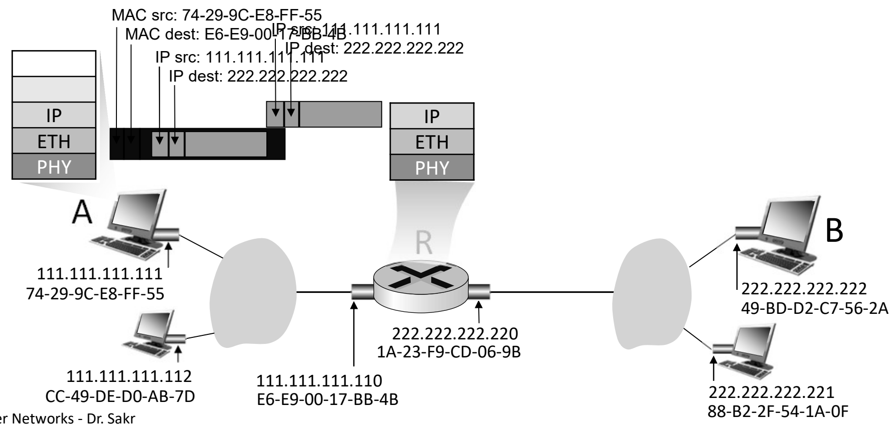
Example: Flow of Packets over Internetworks (cont.)

- A creates IP datagram with IP source A, destination B
- A creates link-layer frame containing A-to-B IP datagram
  - R's MAC address is frame destination, obtained using ARP query



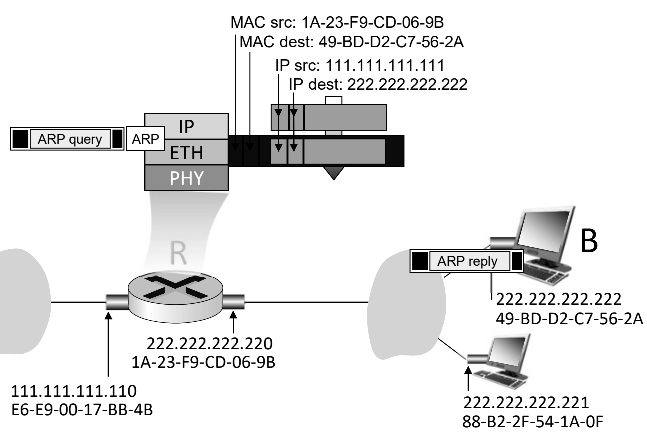
Example: Flow of Packets over Internetworks (cont.)

- Frame sent from A to R
- Frame received at R, datagram removed, passed up to IP



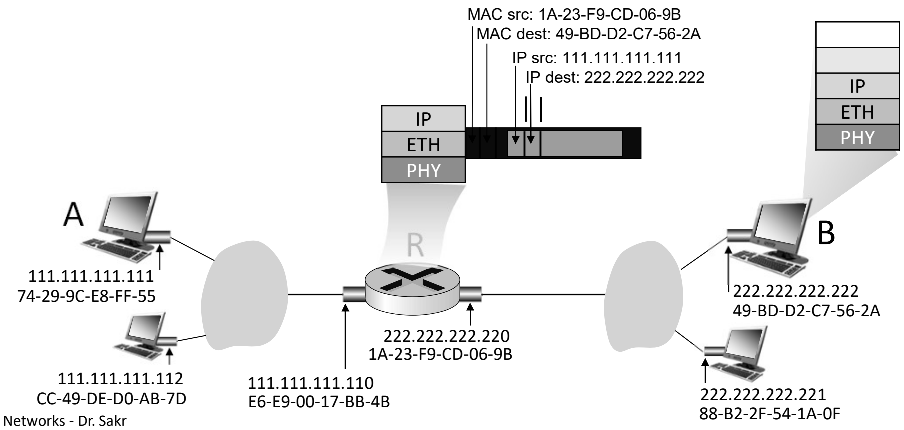
Example: Flow of Packets over Internetworks (cont.)

- R determines outgoing interface, passes datagram to link layer
- R creates link-layer frame containing A-to-B IP datagram
  - B's MAC address is frame destination, obtained using ARP query



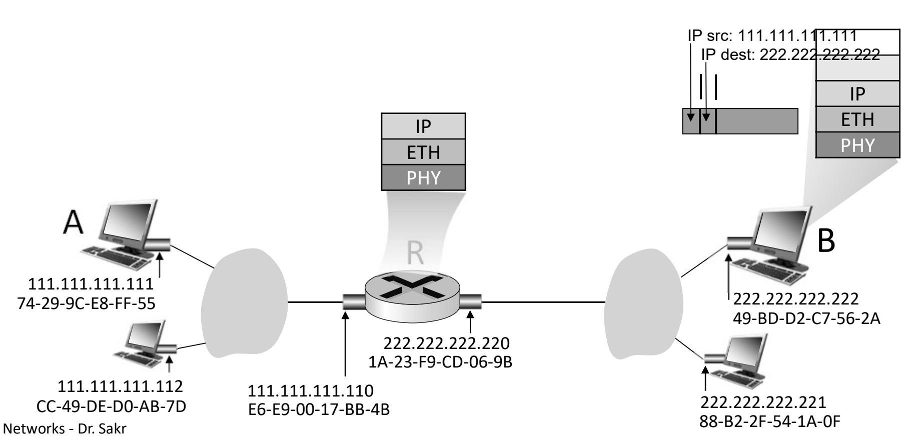
Example: Flow of Packets over Internetworks (cont.)

- Transmits link-layer frame



Example: Flow of Packets over Internetworks (cont.)

- B receives frame, extracts IP datagram destination B
- B passes datagram up protocol stack to IP



Summary

- We covered:
  - Data link control: framing and error control
  - Media access control: random, controlled, and channelization
  - MAC addresses
  - Address resolution protocol (ARP)