CSE 461: Midterm Review

Winter 2021

Administrivia

- **Midterm!**
  - February 8th, **Monday**, 12:30pm - 1:30pm PST
  - Different time zone?

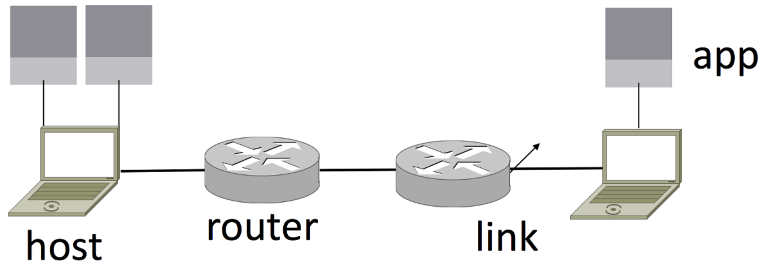
- Assignment 3 due Feb 8th, **Monday**, 11pm

Network Components

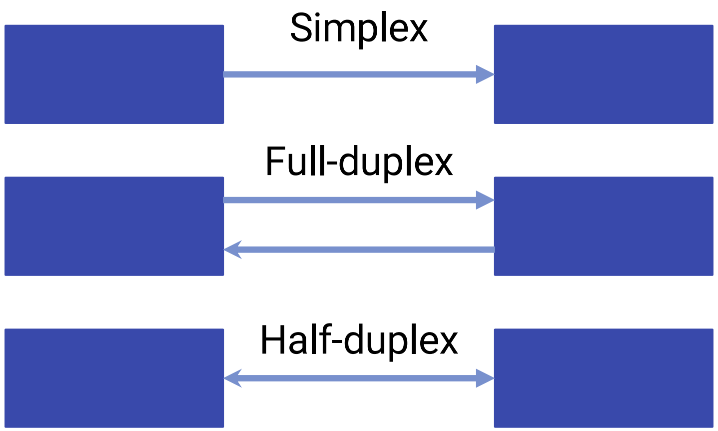
- Parts of a Network
- Types of Links
- Protocols and Layers
- Encapsulation

Parts of a Network

- Parts of a Network



- Types of Links



Protocols and Layers

	Purpose	Protocols	Unit of Data
Application	Programs that use network service	HTTP, DNS	Message
Transport	Provides end-to-end data delivery	TCP, UDP	Segment
Network	Sends packets across multiple networks	IP	Packet
Link	Sends frames across a link	Ethernet, Cable	Frame
Physical	Transmit bits	__	Bit

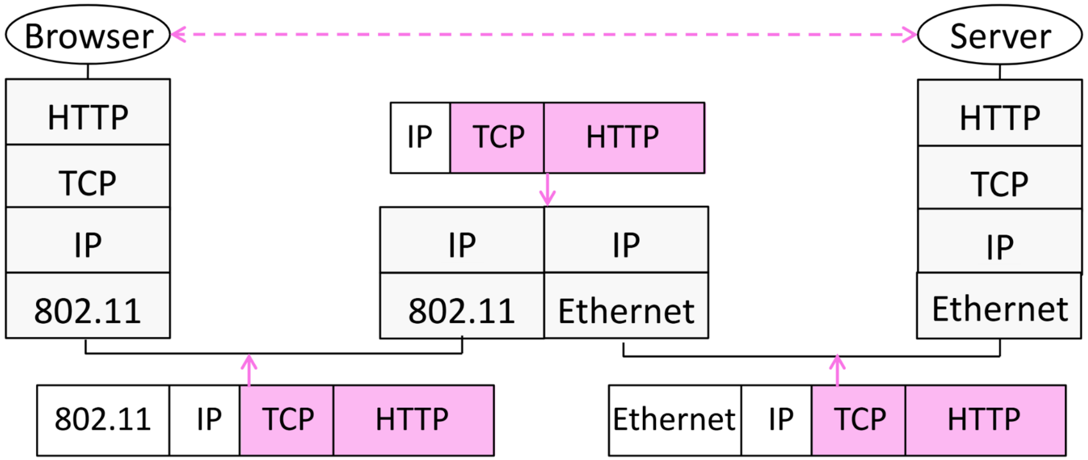
Protocols and Layers

ADVANTAGES

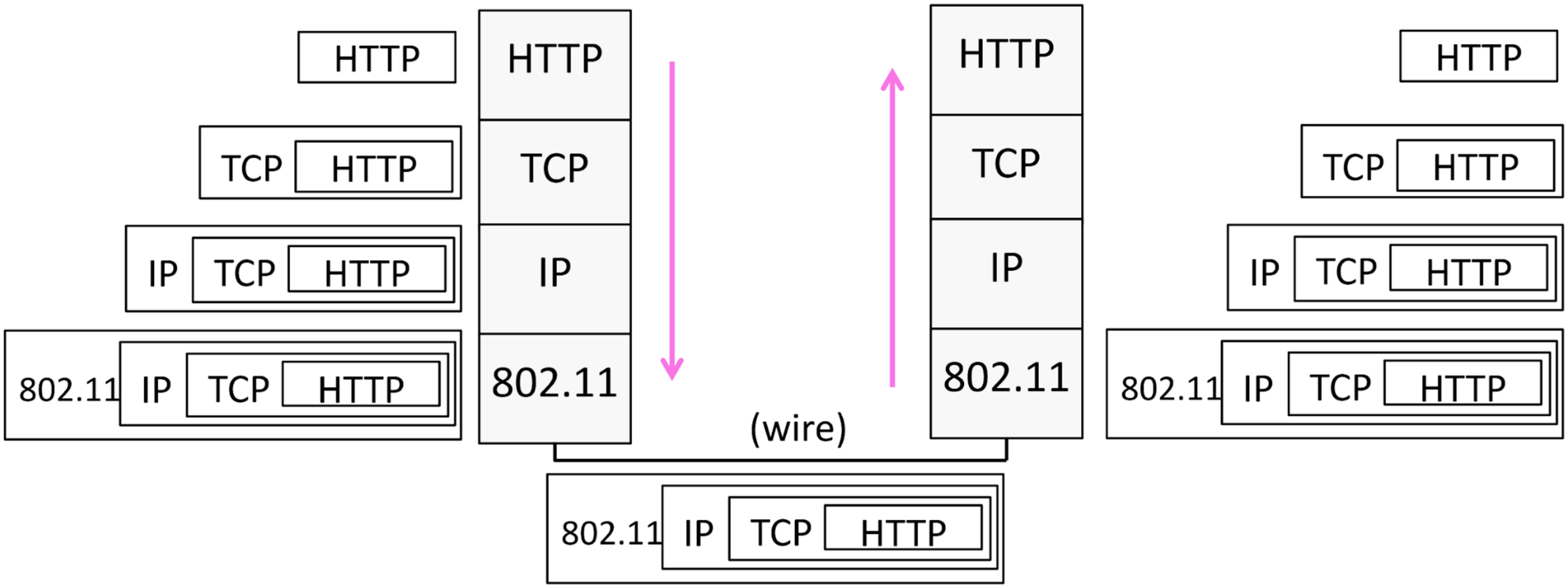
- Use information hiding to connect different systems
- Information reuse to build new protocols

DISADVANTAGES

- Adds overhead
- Hides information



Encapsulation



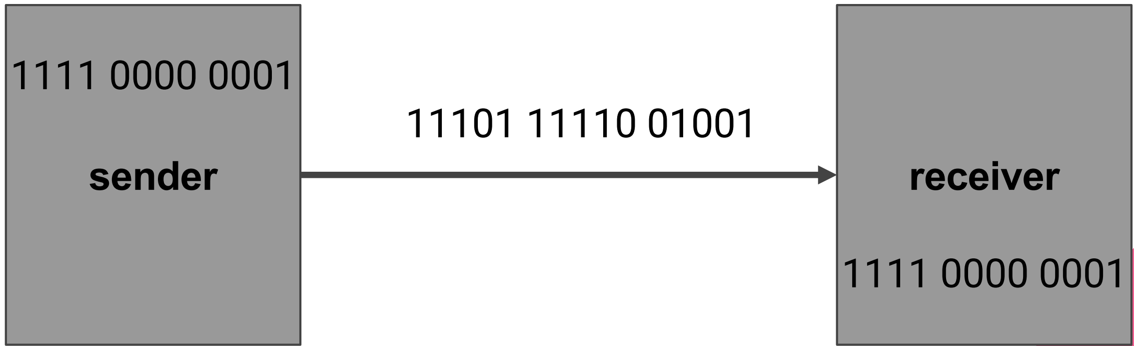
Physical Layer

- Coding: Clock Recovery
- Modulation
- Latency
- Media and Theoretical Limits

Coding: Clock Recovery

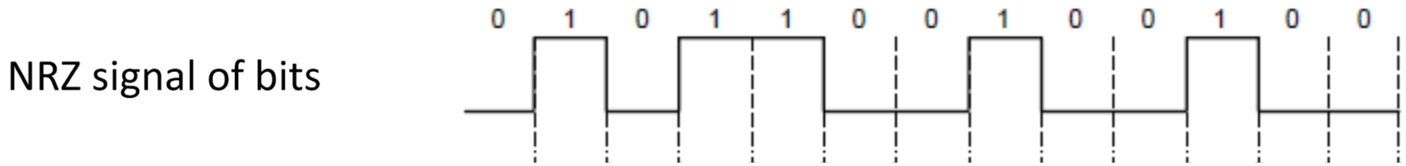
1000000000000000000000000000000000000000000001

- One answer - 4B/5B
  - map every 4 data bits to 5 data bits
  - such that there are no more than 3 zeros in a row
  - invert signal level on a 1 to break up long runs of 1s

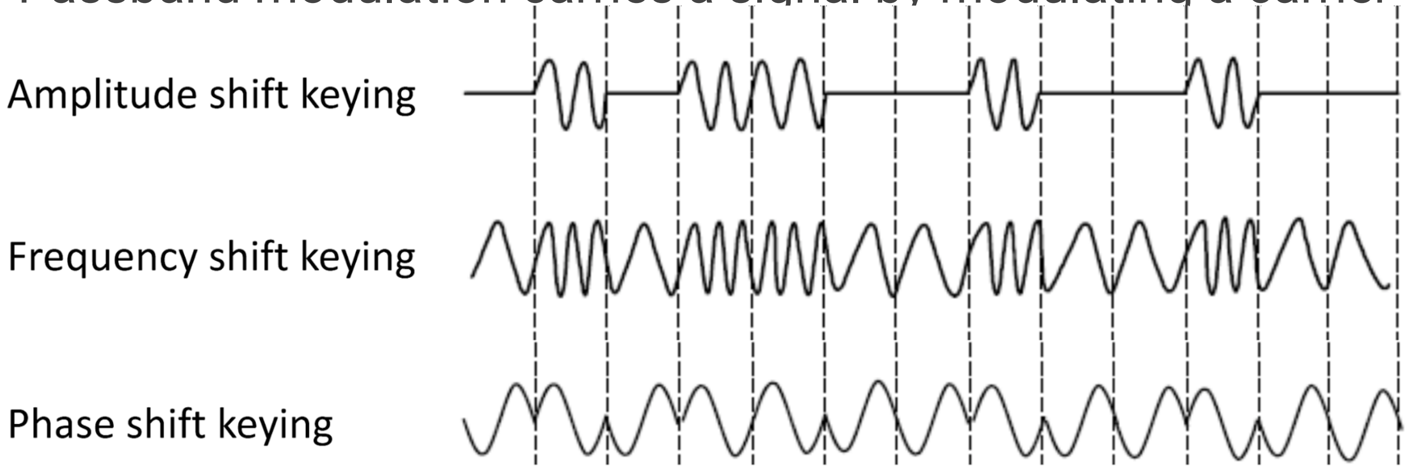


Modulation

- Baseband modulation allows signal to be sent directly on wire

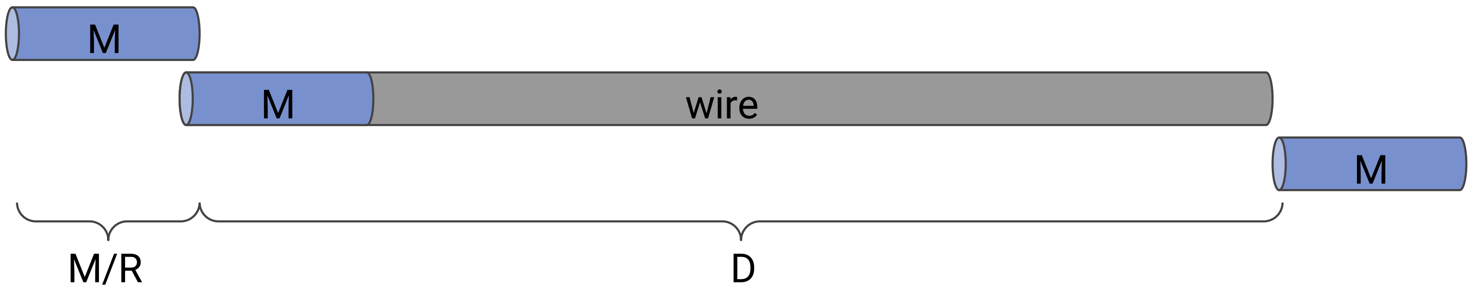


- Passband modulation carries a signal by modulating a carrier



Latency

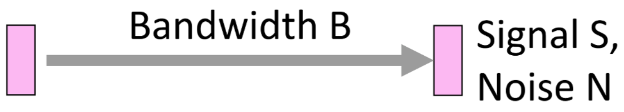
- Latency = Transmission Delay + Propagation Delay
- Transmission Delay = M (bits) / R (bits/sec) = M/R (sec)
- Propagation Delay = Length / Speed of Signals = Length / ⅔c = D (sec)



- Bandwidth-Delay Product = R (bits/sec) x D (sec) = BD (bits)
- RTT = round-trip time

Media and Theoretical Limits

- Media
  - Wire, Fiber
  - Wireless: radiates signal over a region
- Channel Limits: how rapidly can we send information over a link?
  - Bandwidth (**B)**, Signal Power (**S**), Noise Power (**N**)
  - Shannon Capacity - maximum lossless info carrying rate



$$C = B \log_2(1 + S/N) \text{ bits/sec}$$

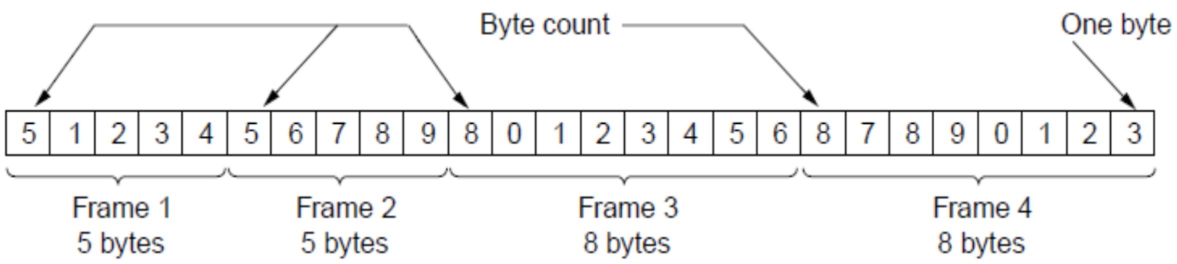
Link Layer

- Framing
- Error detection and correction
- Retransmissions
- Multiple Access
- Switching

Framing Methods

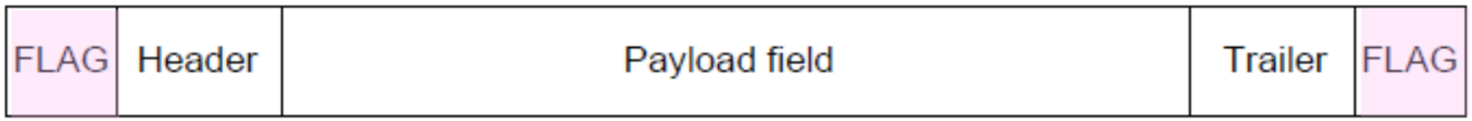
- How do we know where a bit sequence (frame) begins and ends?
  - Byte count
  - Byte stuffing
  - Bit stuffing

- Byte Count

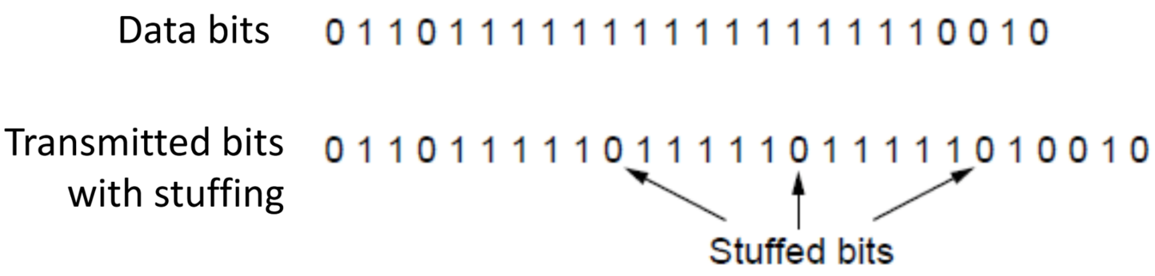


Framing Methods

- Byte Stuffing
  - Replace ESC in data with ESC ESC, and replace FLAG in data with ESC FLAG

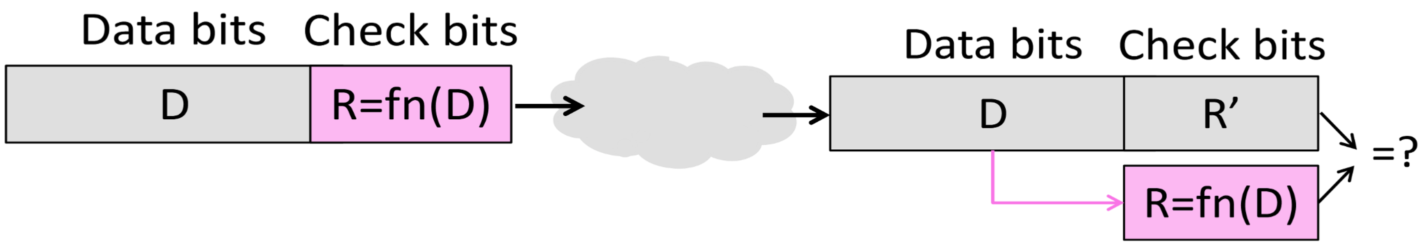


- Bit Stuffing
  - Sequences of 1s as flag, and then add 0 after each flag within data



Error Detection and Correction

- Add check bits to the message bits to let some errors be detected
- Add more check bits to let some errors be corrected



Hamming Distance

- HD between two codes (D1, D2)
  - the number of bit flips needed to change D1 to D2
  - D1 = 0110110101001
  - D2 = 0100000100001
- HD of a coding
  - the minimum error distance between any pair of codewords that cannot be detected
- For a Hamming distance of *d + 1*, up to *d* errors will be **detected**
- For a Hamming distance of *2d + 1*, up to *d* errors can be **corrected**

Error Detection Methods

	Description	Hamming Distance
Parity Bit	Add 1 check bit that is sum/XOR of d data bits	2
Internet Checksum	1s complement sum of 16 bit word	2
Cyclic Redundancy Check (CRC)	For n data bits, generate n+k bits that are evenly divisible by C	4

HD of Internet Checksum

```
   0001
   f204
   f4f5
   f6f7
+  220c
------
  2fffd

   fffd
+     2
------
   ffff

   0000
```

Error Correction - Hamming Code

Hamming Distance = 3

Suppose we want to send a message M of 4 bits: **0101**
We add k=3 check bits, because ($n = 2^k - k - 1 = 2^3 - 3 - 1 = 4$)

So, we will have a n+k = 7 bit code, with check bits in positions 1, 2, 4
Each check bit is an XOR of certain positions.

Error Correction - Hamming Code

```
      421          421          421
1 = 0b001    1 = 0b001    1 = 0b001
2 = 0b010    2 = 0b010    2 = 0b010
3 = 0b011    3 = 0b011    3 = 0b011
4 = 0b100    4 = 0b100    4 = 0b100
5 = 0b101    5 = 0b101    5 = 0b101
6 = 0b110    6 = 0b110    6 = 0b110
7 = 0b111    7 = 0b111    7 = 0b111
```

0 1 0 0 1 0 1

1 2 3 4 5 6 7

```
p1 = b3+b5+b7 = 0+1+1 = 0
p2 = b3+b6+b7 = 0+0+1 = 1
p4 = b5+b6+b7 = 1+0+1 = 0
```

- Example, continued

→ $\underline{0}$ $\underline{1}$ 0 $\underline{0}$ 1 1 1

1 2 3 4 5 6 7

p_1= 0+0+1+1 = 0, p_2= 1+0+1+1 = 1,

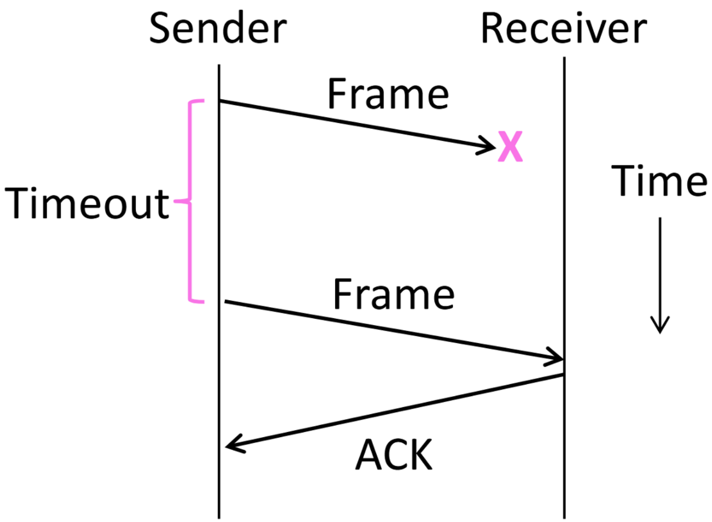
p_4= 0+1+1+1 = 1

Syndrome = 1 1 0, flip position 6

Data = 0 1 0 1 (correct after flip!)

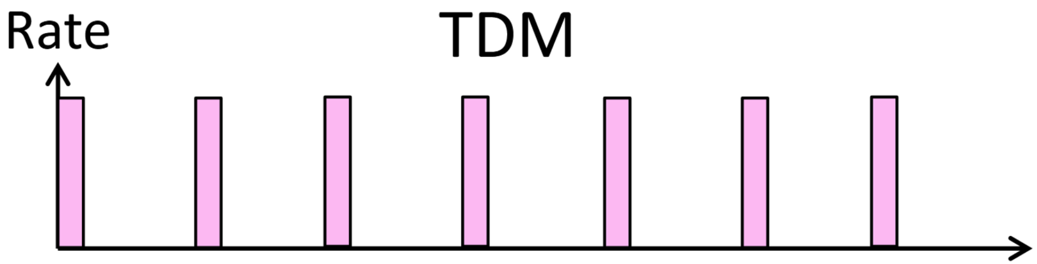
ARQ - Automatic Repeat Request

- ARQ
- Stop-and-wait
- Sliding window

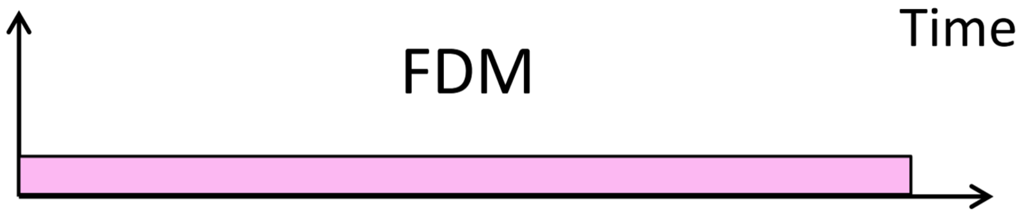


Multiplexing

- Time Division Multiplexing - high rate at some times



- FDM - low rate all the time



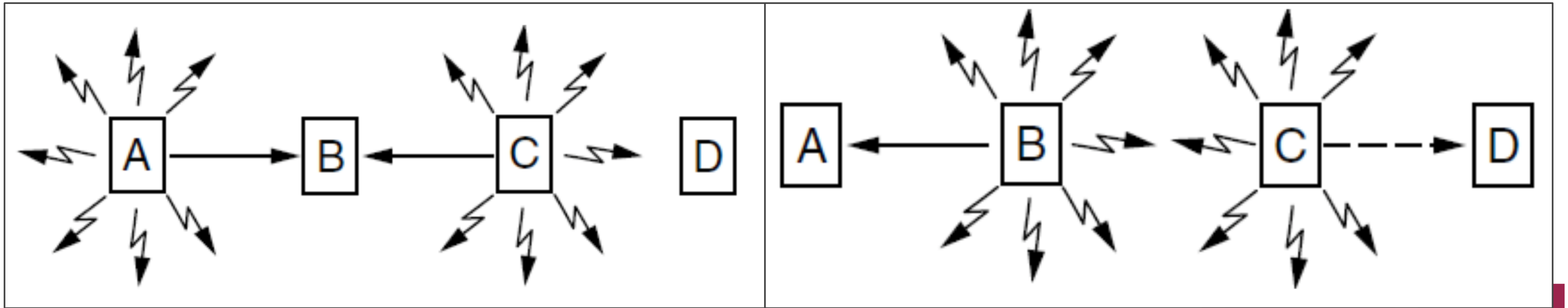
Multiple Access

- ALOHA: Node just sends when it has traffic; if collision happens, wait for a random amount of time and try again.
  - Huge amount of loss under high load
- CSMA (Carrier Sense Multiple Access): Listen before send.
  - Collision is still possible because of delay; good only when BD is small
- CSMA/CD (Carrier Sense Multiple Access with Collision Detection): CSMA + Aborting JAM for the rest of the frame time
  - Minimum frame length of 2D seconds
- CSMA “Persistence”: CSMA + P(send) = 1 / N
  - Reduce the chance of collision
- Binary Exponential Backoff (BEB): Doubles interval for each successive collision
  - Very efficient in practice

Issues with Wireless

Hidden Terminal Problem: nodes A and C are hidden terminals when sending to B

Exposed Terminal Problem: nodes B and C are exposed terminals when sending to A and D



MACA is a potential solution: Sender sends Request to Send(RTS) and receiver replies Clear To Send(CTS).

Switches

- Backward Learning
  - Learn the sender’s port by looking at the packets
- Spanning Tree
  - Elect the root node of the tree (Usually the switch with the lowest address)
  - Grow tree based on the shortest distance from the root
  - Ports not on the spanning tree are turned off

Network Layer

- Core protocols
  - IP
  - ICMP
  - ARP
  - DHCP
- Routing / Forwarding
  - Routing - decide where to send
  - Forward - send