Overview

- Methods to share physical media: **multiple access**
  - Fixed partitioning
  - Random access

- Channelizing mechanisms

- Contention-based mechanisms
  - Aloha
  - Ethernet
Fixed Partitioning

- Need to share media with multiple nodes \((n)\)
  - Multiple *simultaneous* conversations

- A simple solution
  - Divide the channel into multiple, separate channels

- Channels are physically separate
  - Bitrate of the link is split across channels
  - Nodes can only send/receive on their assigned channel

- Several different ways to do it
  - _____ Multiple Access madlibs…
Frequency Division (FDMA)

- Divide bandwidth of $f$ Hz into $n$ channels each with bandwidth $f/n$ Hz
  - Easy to implement, but unused subchannels go idle
  - Used by traditional analog cell phone service, radio, TV

CSE 123 – Lecture 8: Media Access Control
Time Division (TDMA)

- Divide channel into rounds of \( n \) time slots each
  - Assign different hosts to different time slots within a round
  - Unused time slots are idle
  - Used in GSM cell phones & digital cordless phones

- Example with 1-second rounds
  - \( n=4 \) timeslots (250ms each) per round

CSE 123 – Lecture 8: Media Access Control
Code Division (CDMA)

- Do nothing to physically separate the channels
  - All stations transmit at same time in same frequency bands
  - One of so-called spread-spectrum techniques

- Sender modulates their signal on top of unique code
  - Sort of like the way Manchester modulates on top of clock
  - The bit rate of resulting signal much lower than entire channel

- Receiver applies code filter to extract desired sender
  - All other senders seem like noise with respect to signal

- Used in newer digital cellular technologies

CSE 123 – Lecture 8: Media Access Control
Partitioning Visualization

FDMA

TDMA

CDMA

Courtesy Takashi Inoue

CSE 123 – Lecture 8: Media Access Control
Problem w/Channel partitioning

- Not terribly well suited for random access usage
  - Why?

- Instead, design schemes for more common situations
  - Not all nodes want to send all the time
  - Don’t have a fixed number of nodes

- Potentially higher throughput for transmissions
  - Active nodes get full channel bandwidth
Aloha

- Designed in 1970 to support wireless data connectivity
  - Between Hawaiian Islands—rough!

- Goal: distributed access control (no central arbitrator)
  - Over a shared broadcast channel

- Aloha protocol in a nutshell:
  - When you have data send it
  - If data doesn’t get through (receiver sends acknowledgement) then retransmit after a random delay
  - Why not a fixed delay?
Collisions

- Frame sent at $t_0$ collides with frames sent in $[t_0-1, t_0+1]$
  - Assuming unit-length frames
  - Ignores propagation delay
Slotted Aloha

- Time is divided into equal size slots (frame size)
- Host wanting to transmit starts at start of next slot
  - Retransmit like w/Aloha, but quantize to nearest next slot
- Requires **time synchronization** between hosts

Success (S), Collision (C), Empty (E) slots
Q: What is max fraction slots successful?
A: Suppose $n$ stations have packets to send
  - Each transmits in slot with probability $p$
  - Prob[successful transmission], $S$, is:

$$S = p \cdot (1-p)^{(n-1)}$$

- any of $n$ nodes:

$$S = \text{Prob[one transmits]} = np(1-p)^{(n-1)}$$

(optimal $p$ as $n \to \infty$ = $1/n$)

$$= 1/e = .37$$

At best: channel used for useful transmissions 37% of time!
Carrier Sense (CSMA)

- Aloha transmits even if another host is transmitting
  - Thus guaranteeing a *collision*

- Instead, listen *first* to make sure channel is idle
  - Useful only if channel is frequently idle
  - Why?

- How long to be confident channel is idle?
  - Depends on maximum propagation delay
  - Small (<<1 frame length) for LANs
  - Large (>>1 frame length) for satellites
non-persistent CSMA

- Give up, or send after some random delay
- Problem: may incur larger delay when channel is idle

1-persistent CSMA

- Send as soon as channel is idle
- Problem: blocked senders all try to send at once

P-persistent CSMA

- If idle, send packet with probability $p$; repeat
- Make sure $(p \times n) < 1$
For Next Time

- Read 3-3.2 in P&D
- Get started on HW2
- Keep going on the project…