



INTRODUCTION TO EECS II  
**DIGITAL COMMUNICATION SYSTEMS**

Lecture 26  
 Spring 2009  
 Wrap-up




### Our Goals

- Engineering digital communication systems
  - Understanding key concepts
  - Exposing you to a range of important EECS ideas
- Across the different layers of the “stack”
  - Links, shared channels, multi-hop networks
- Across traditional “EE” and “CS” boundaries




### Three Big Challenges

- Reliability
  - Communication is a notoriously hard problem; many things can go wrong
- Sharing
  - Dedicated links are impossibly expensive
- Scalability
  - Successful networks are large; large networks are successful (utility grows super-linearly with size)

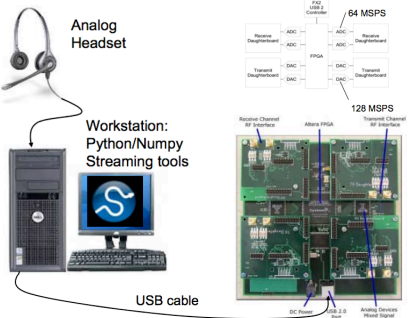


### Approach

- Understand tools and techniques
  - Concepts and principles
  - Labs
  - Small problems
- Begin to understand trade-offs
  - The essence of all engineering systems
  - Science, art, or a mix?
  - Principles and tools matter, as do intuition and experience



### Experimental Apparatus




Analog Headset

Workstation: Python/Numpy Streaming tools

USB cable


64 MSPS

128 MSPS



### Reliability (1): Understanding Problems

- Challenge: Overcome wide range of faults
  - Inter-symbol interference, noise, bit errors, packet loss, buffer overflow, link failures, ...
- Digital abstraction
  - Key to enabling composition
  - Lab: Clock+data recovery, 8b/10b
- Inter-symbol interference
  - LTI, superposition, eye diagrams
  - Lab: unit-sample response, deconvolution
- Noise
  - Understanding Gaussians, PDFs, CDFs
  - Lab: measure, predict bit error rates



## Reliability (2): Overcoming Problems

- Bit error detection + correction
  - Lab: interleaved block coding
  - Lab: Viterbi decoding of convolutional codes
- Packet loss
  - Lab: reliable transport protocols, sliding windows
- Link faults
  - Lab: routing around failures (distance vector and link-state protocols)
- Common theme: apply redundancy in creative ways



## Sharing - 1

- Challenge: Communication resources aren't free or cheap
- Sharing a common medium (channel)
  - Lab: understanding frequency response; designing filters based on zeroes and poles
  - Lab: band-limited multiple sender transmissions modulated at different carrier frequencies
  - Lab: Contention and time division MAC protocols (Aloha, CSMA, TDMA, exponential backoff)
- Reducing amount of data sent: compression



## Sharing - 2

- Switches and link multiplexing
  - Circuit switching → historically came first, but supports only narrow set of apps (telephony)
  - Packet switching → supports wide range of applications
- Best-effort networks
- Queues and Little's law
- Forwarding and routing functions
- Transport protocols
- Layering
- Hierarchical network design



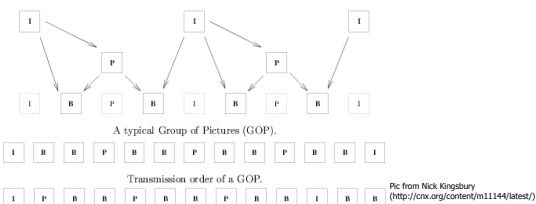
## Trade-Offs

- A number of techniques - how to apply them and make them work together?
- Reliability: apply redundancy in creative ways to build reliable systems out of unreliable components
- Sharing: reduce the amount of resources consumed
- Scalability: hide information, reduce amount of state to be managed



## Example: Delivering Video over the Internet to Mobile Phones

- Compression (e.g., MPEG4/H.264, etc.)



- Reliability: Not all frames equally important
- Rate adaptation / congestion control
- Handling link losses, interference, etc.



## What Should Each Layer Do?

Application	Compression Rate (format) adaptation Reliability? TCP or UDP+selective reliability?
Transport	Forwarding & routing (and addressing)
Network	Framing, retry, channel access
Data Link	Coding, (De)modulation How reliable should these layers be? Trade-off w/ bit rate
Physical	



## EECS Ideas

- Signals and systems
  - LTI, superposition, unit-sample response, frequency response, modulation
- Algorithms, centralized and distributed
  - Trellis decoding (Viterbi), shortest paths (Dijkstra), distance vector routing (Bellman-Ford), compression (LZW, JPEG)
- Computer systems
  - Abstraction and modularity, layering, protocols, hierarchy
- Applied probability
  - Continuous-domain probability (density): reliability analysis
  - Discrete-domain probability: MAC protocol analysis
  - Basic queueing models: packet switch sharing analysis
- Methods: design, simulation, experimentation



## Feedback

- Please give us your feedback
- HKN review form on web site → **please complete this week!**
- Email/talk to us any time
- Thursday: open lab and office hours
  - Finish up remaining labs/checkoffs
  - Come and ask us anything about the material (think of it as small-group review sessions)



## Discussion

- Which activities worked well?
  - Lectures
  - Recitations
  - Labs: did they help understand material? Were they interesting?
  - Online psets: how effective?
  - Review problems
- Did we cover too much? Too little?
- Would LAs help? (Would you like to be one?)

