ESE 531: Digital Signal Processing

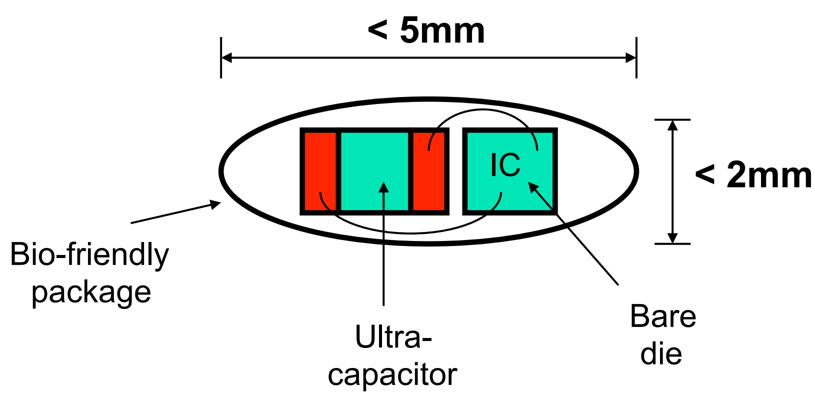
Lec 1: January 12, 2017

Introduction and Overview

Where I come from

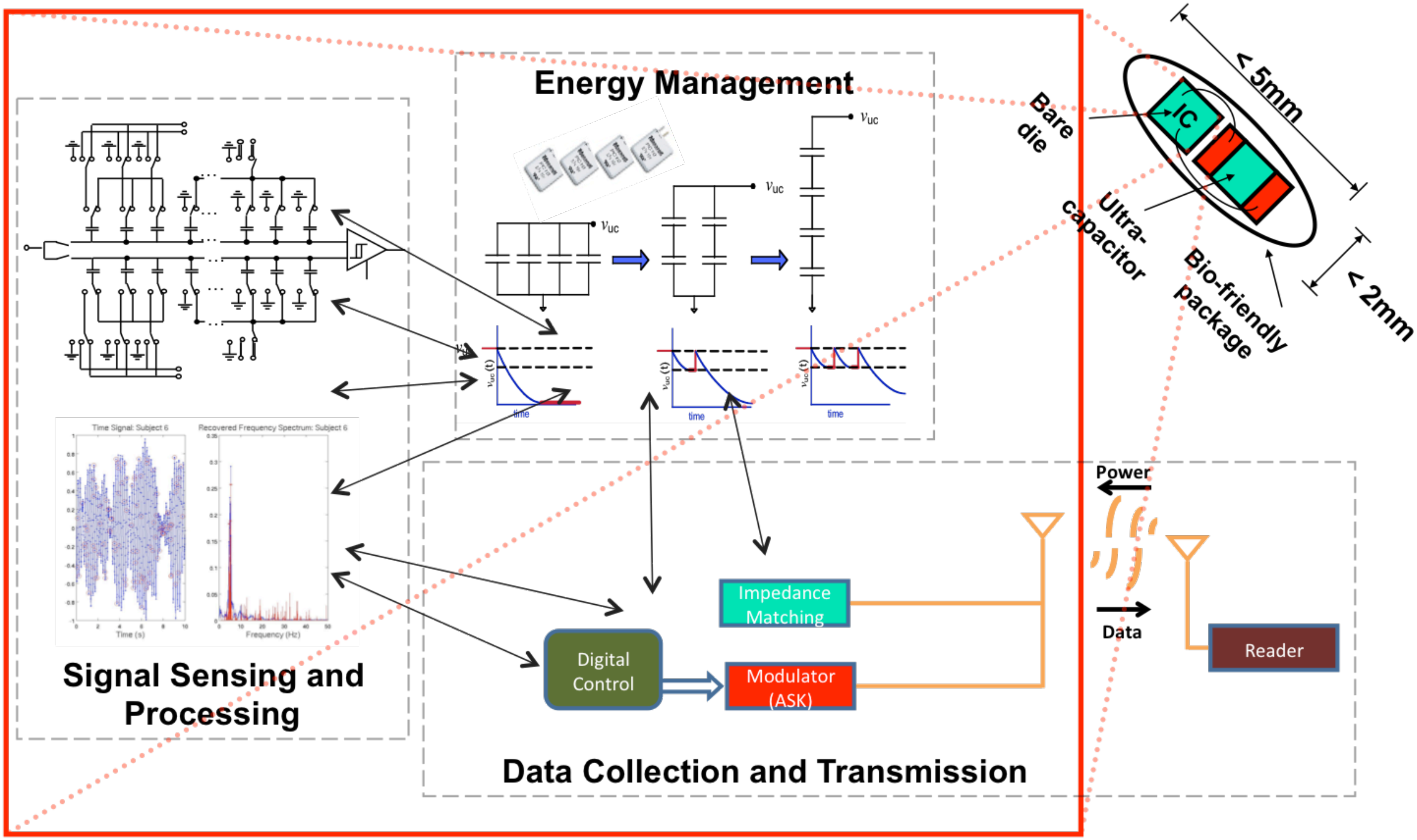
- Analog VLSI Circuit Design
- Convex Optimization
  - System Hierarchical Optimization
- Biomedical Electronics
- Biometric Data Acquisition
  - Compressive Sampling
- ADC Design
  - SAR, Pipeline, Delta-Sigma
- Low Energy Circuits
  - Adiabatic Charging

Minimally Invasive Implant to Combat Healthcare Noncompliance



- Model for implants: reconfigurable RFID tags that continuously record specific biometric
  - During the read operation, energy storage element is recharged
- Size of package small enough to allow injection
- Actigraphy expected to be clinically useful
  - Platform allows for any sensor that gathers information on a slow time scale

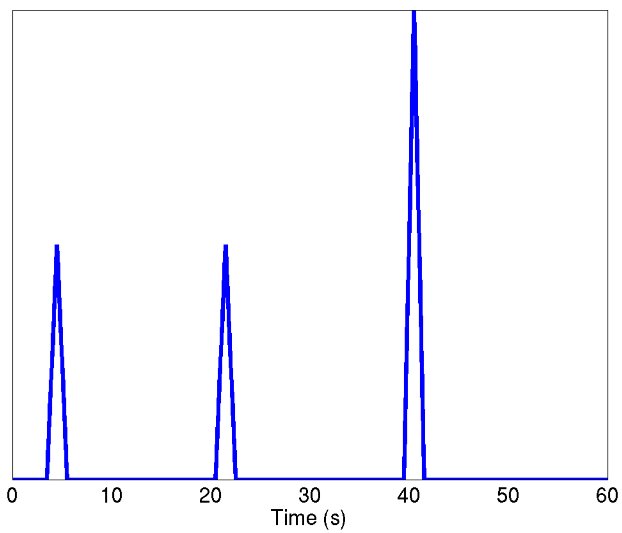
MicroImplant: An Electronic Platform for Minimally Invasive Sensory Monitors



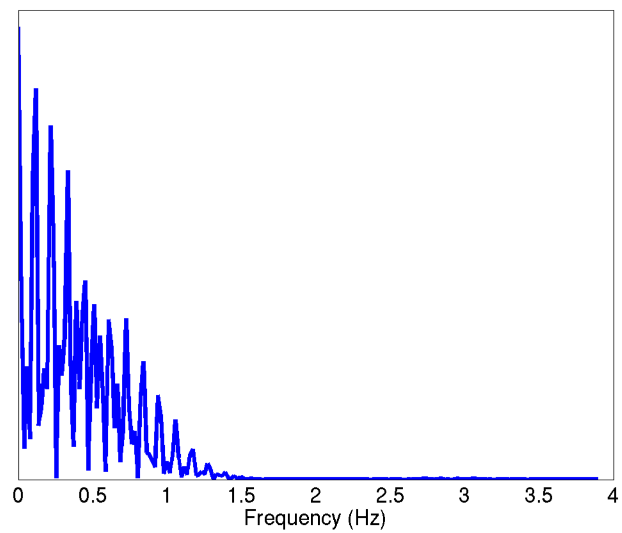
Compressive Sampling

- Sample at lower than the Nyquist rate and still accurately recover the signal, and in some cases exactly recover

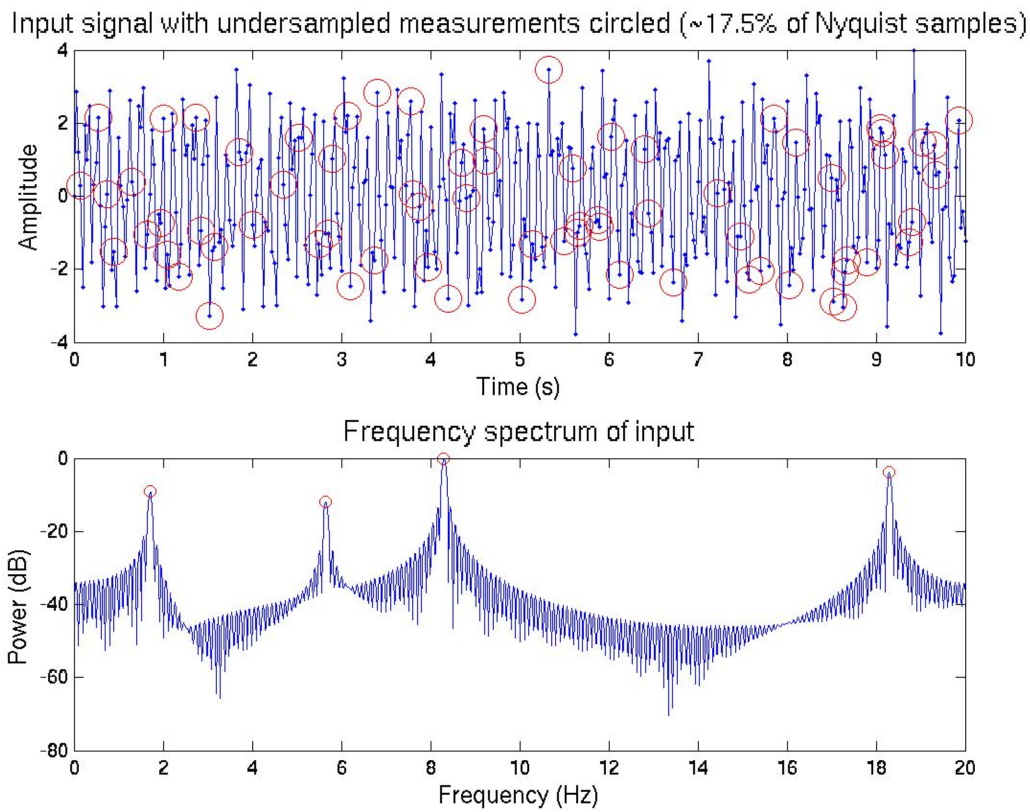
Sparse signal in time



Frequency spectrum

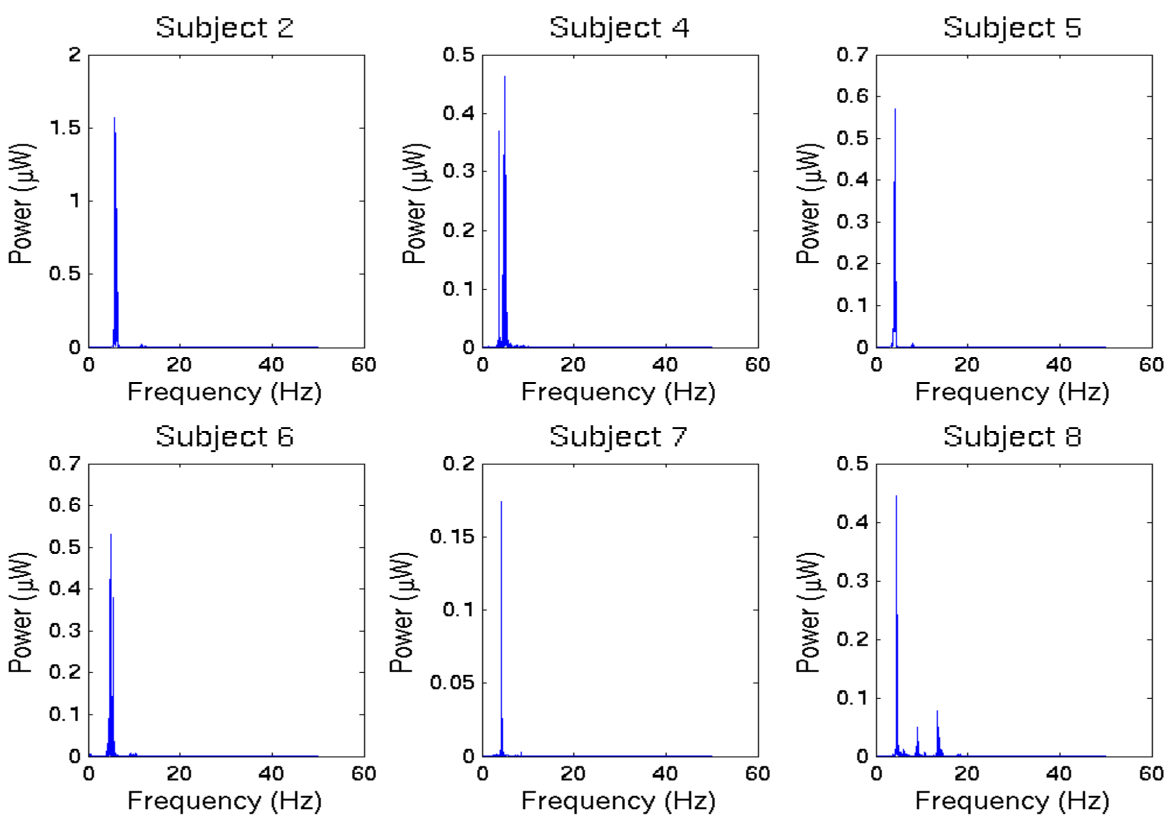


Example: Sum of Sinusoids



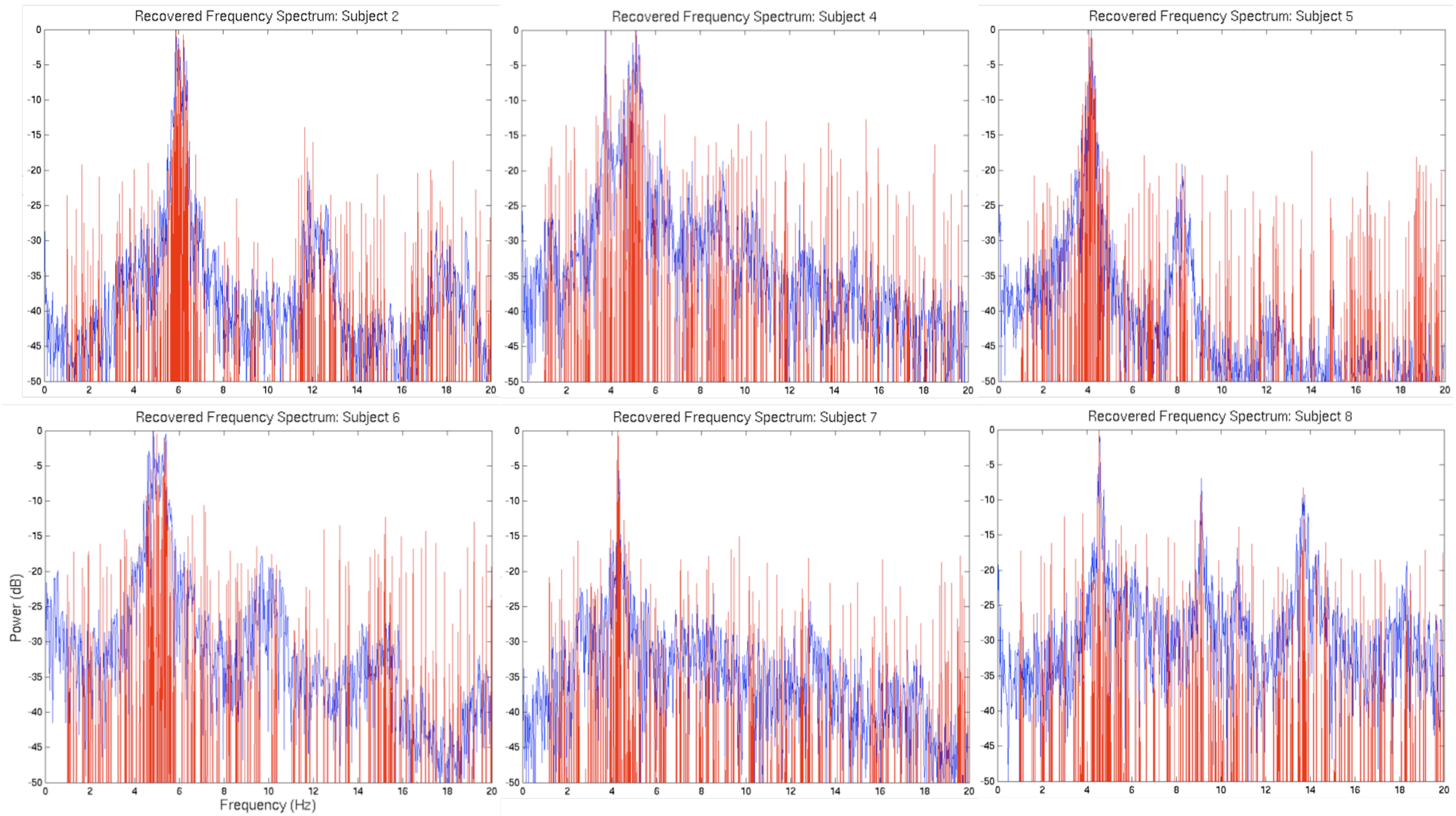
- Sense signal randomly M times
  - $M > C \cdot \mu^2(\Phi, \Psi) \cdot S \cdot \log N$
- Recover with linear program

Biometric Example: Parkinson's Tremors



- 6 Subjects of real tremor data
  - collected using low intensity velocity-transducing laser recording aimed at reflective tape attached to the subjects' finger recording the finger velocity
  - All show Parkinson's tremor in the 4-6 Hz range.
  - Subject 8 shows activity at two higher frequencies
  - Subject 4 appears to have two tremors very close to each other in frequency

Biometric Example: Parkinson's Tremors



- **C=10.5, T=30**
  - 20% Nyquist required samples

Lecture Outline

- Course Topics Overview
- Learning Objectives
- Course Structure
- Course Policies
- Course Content
- Industry Trends
- Design Example

Course Topics Overview

- Discrete-Time (DT) Signals
- Time-Domain Analysis of DT Systems
- Discrete Fourier Transform (DFT)
- Fast Fourier Transform (FFT)
- Discrete-Time Fourier Transform (DTFT)
- z-Transform
- Sampling of Continuous Time Signals
- Data Converters and Modulation
- Upsampling/Downsampling
- Discrete-Time Filter Design

Learning Objectives

- Learn the fundamentals of digital signal processing
- Provide an understanding of discrete-time signals and systems and digital filters
- Enable you to apply DSP concepts to a wide range of fields
- Gain the ability to read the technical literature on DSP
- Apply the techniques learned in a final project encompassing many different application types

Learning Objectives

- In other words…

- Math, Math, Math*

*With MATLAB application for intuition

Course Structure

- TR Lecture, 4:30-6:00pm in Towne 303
  - Start 5 minutes after, end 5 minutes early (~75-80min)
- Website (http://www.seas.upenn.edu/~ese531/)
  - Course calendar is used for all handouts (lectures slides, assignments, and readings)
  - Canvas used for assignment submission and grades
  - Piazza used for announcements and discussions

Course Structure

- Course Staff (complete info on course website)
- Instructor: Tania Khanna
  - Office hours – Wednesday 2-4:30 pm or by appointment
  - Email: taniak@seas.upenn.edu
    - Best way to reach me
- TA: Shlesh Tiwari
  - Office hours – TBD

Course Structure

- Lectures
  - Statistically speaking, you will do better if you come to lecture
  - Better if interactive, **everyone** engaged
    - Asking and answering questions
    - Actively thinking about material
- Textbook
  - A. V. Oppenheim and R. W. Schafer (with J. R. Buck), Discrete-Time Signal Processing. 3rd. Edition, Prentice-Hall, 2010
  - Class will follow text structure… mostly

Course Structure - Assignments/Exams

- Homework – 1-2 week(s) long (8 total) [25%]
  - Due Thursdays at start of class (1:30pm)
  - Combination of book problems and matlab problems
    - MATLAB problems not optional
- Project – two+ weeks long (2 total) [30%]
  - Combination of different DSP applications
- Midterm exam [20%]
- Final exam [25%]

Course Policies

See web page for full details

- Turn homework in Canvas before lecture starts
  - Anything handwritten/drawn must be clearly legible
  - Submit CAD generated figures, graphs, results when specified
  - NO LATE HOMEWORKS!
- Individual work (except project)
  - CAD drawings, simulations, analysis, writeups
  - May discuss strategies, but acknowledge help

Course Content

- Introduction
- Discrete Time Signals & Systems
- Discrete Time Fourier Transform
- Z-Transform
- Inverse Z-Transform
- Sampling of Continuous Time Signals
- Frequency Domain of Discrete Time Series
- Downsampling/Upsampling
- Data Converters, Sigma Delta Modulation
- Frequency Response of LTI Systems
- Signal Flow Representation
- Basic Structures for IIR and FIR Systems
- Design of IIR and FIR Filters
- Butterworth, Chebyshev, and Elliptic Filters
- Filter Banks
- Adaptive Filters
- Computation of the Discrete Fourier Transform
- Fast Fourier Transform

Course Content

ESE531 Spring 2017 Working Schedule

Wk	Lect.	Date		Lecture	Slides	Due	Reading
1	1	1/12	Th	Intro/Overview			review course webpage completely
2	2	1/17	T	Discrete Time Signals & Systems, Part 1			2.1-2.2
	3	1/19	Th	Discrete Time Signals & Systems, Part 2			2.3-2.5
3	4	1/24	T	Discrete Time Fourier Transform			2.5-2.7
	5	1/26	Th	Z-Transform		HW 1	3.0-3.1
4	6	1/31	T	Inverse Z-Transform			3.3
	7	2/2	Th	Z-Transform Properties		HW 2	3.4
5	8	2/7	T	Sampling of Continuous Time Signals			4.0-4.1
	9	2/9	Th	Frequency Domain of Discrete Time Series		HW 3	4.2-4.3
6	10	2/14	T	Downsampling/Upsampling			4.6
	11	2/16	Th	Data Converters, Sigma Delta Modulation		HW 4	4.8-4.9
7	12	2/21	T	Frequency Response of LTI Systems			5.0-5.2
	13	2/23	Th	Signal Flow Representation			6.0-6.2
8	14	2/28	T	Basic Structures for IIR Systems			6.3
	15	3/2	Th	Basic Structures for FIR Systems, Parameter Quantization Effects		HW 5	6.5, 6.8
9		3/7	T	SPRING BREAK -- no class			
		3/9	Th	SPRING BREAK -- no class			
10		3/14	T	Midterm Exam, in class			
	16	3/16	Th	Design of IIR Filters, Part 1			7.0-7.2
11	17	3/21	T	Design of IIR Filters, Part 2			7.0-7.2
	18	3/23	Th	Butterworth, Chebyshev, and Elliptic Filters		HW 6	7.3
12	19	3/28	T	Design of FIR Filters			7.5
	20	3/30	Th	Filter Banks		HW 7	TBD
13	21	4/4	T	Adaptive Filters			TBD
	22	4/6	Th	Discrete Fourier Transform		HW 8	8.0-8.2
14	23	4/11	T	Computation of the DFT, Part 1			9.0-9.1
	24	4/13	Th	Computation of the DFT, Part 2			9.0-9.1
15	25	4/18	T	Fast Fourier Transform			9.2-9.3
	26	4/20	Th	DSP Applications			TBD
16	27	4/25	T	Review		Project	
17		TBD		FINAL EXAM: TBD in TBD			

This working calendar is on the web: http://www.seas.upenn.edu/~ese531/spring2017/syllabus.html. Please, recheck the page on the web as details may be adjusted as the term progresses. Lectures notes, reading, and handouts will be filled as we reach them.

ESE531 Home Page

What is DSP

DSP is Everywhere

- Sound applications
  - Compression, enhancement, special effects, synthesis, recognition, echo cancellation,…
  - Cell Phones, MP3 Players, Movies, Dictation, Text-to-speech,…
- Communication
  - Modulation, coding, detection, equalization, echo cancellation,…
  - Cell Phones, dial-up modem, DSL modem, Satellite Receiver,…
- Automotive
  - ABS, GPS, Active Noise Cancellation, Cruise Control, Parking,…

DSP is Everywhere (con't)

- Medical
  - Magnetic Resonance, Tomography, Electrocardiogram, Biometric Monitoring…
- Military
  - Radar, Sonar, Space photographs, remote sensing,…
- Image and Video Applications
  - DVD, JPEG, Movie special effects, video conferencing…
- Mechanical
  - Motor control, process control, oil and mineral prospecting,…

Signal Processing

- Humans are the most advanced signal processors
  - speech and pattern recognition, speech synthesis,…
- We encounter many types of signals in various applications
  - Electrical signals: voltage, current, magnetic and electric fields,…
  - Mechanical signals: velocity, force, displacement,…
  - Acoustic signals: sound, vibration,…
  - Other signals: pressure, temperature,…
- Most real-world signals are analog
  - They are continuous in time and amplitude
  - Convert to voltage or currents using sensors and transducers

Signal Processing (con't)

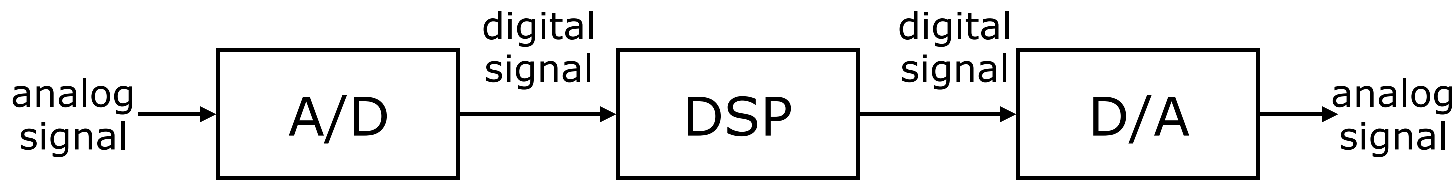
- Analog circuits process these signals using
  - Resistors, Capacitors, Inductors, Amplifiers,…
- Analog signal processing examples
  - Audio processing in FM radios
  - Video processing in traditional TV sets

Limitations of Analog Signal Processing

- Accuracy limitations due to
  - Component tolerances
  - Undesired nonlinearities
- Limited repeatability due to
  - Tolerances
  - Changes in environmental conditions
    - Temperature
    - Vibration
- Sensitivity to electrical noise
- Limited dynamic range for voltage and currents
- Inflexibility to changes
- Difficulty of implementing certain operations
  - Nonlinear operations
  - Time-varying operations
- Difficulty of storing information

Digital Signal Processing

- Represent signals by a sequence of numbers
  - Sampling and quantization (or analog-to-digital conversion)
- Perform processing on these numbers with a digital processor
  - Digital signal processing
- Reconstruct analog signal from processed numbers
  - Reconstruction or digital-to-analog conversion



- Analog input → analog output
  - Eg. Digital recording music
- Analog input → digital output
  - Eg. Touch tone phone dialing, speech to text
- Digital input → analog output
  - Eg. Text to speech
- Digital input → digital output
  - Eg. Compression of a file on computer

Pros and Cons of Digital Signal Processing

- Pros
  - Accuracy can be controlled by choosing word length
  - Repeatable
  - Sensitivity to electrical noise is minimal
  - Dynamic range can be controlled using floating point numbers
  - Flexibility can be achieved with software implementations
  - Non-linear and time-varying operations are easier to implement
  - Digital storage is cheap
  - Digital information can be encrypted for security
  - Price/performance and reduced time-to-market
- Cons
  - Sampling causes loss of information
  - A/D and D/A requires mixed-signal hardware
  - Limited speed of processors
  - Quantization and round-off errors

DSP Examples

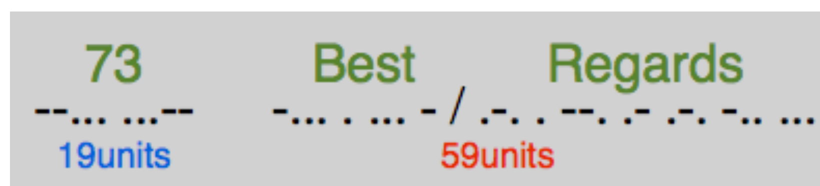
Example I: Audio Compression

- Compress audio by 10x without perceptual loss of quality.
- Sophisticated processing based on models of human perception
- 3MB files instead of 30MB
  - Entire industry changed in less than 10 years!

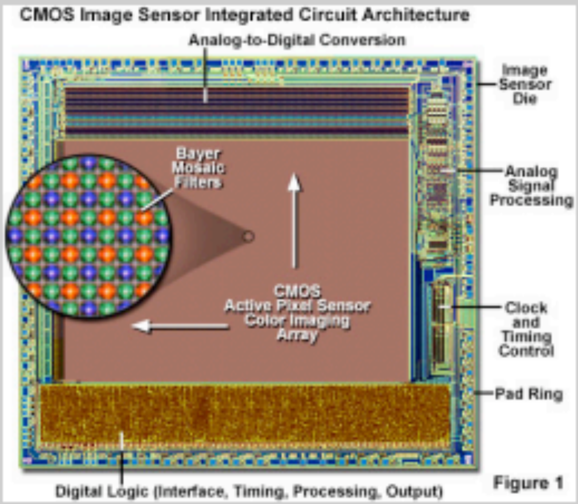
Historical Forms of Compression

- Morse code: dots (1 unit) dashes (3 units)
  - Code Length inversely proportional to frequency of character
    - E (12.7%) = . (1 unit) Q (0.1%) = --.- (10 units)

- “92 Code”
  - Used by Western-Union in 1859 to reduce BW on telegraph lines by numerical codes for frequently used phrases
    - 1 = wait a minute
    - 73 = Best Regards
    - 88 = Loves and Kisses

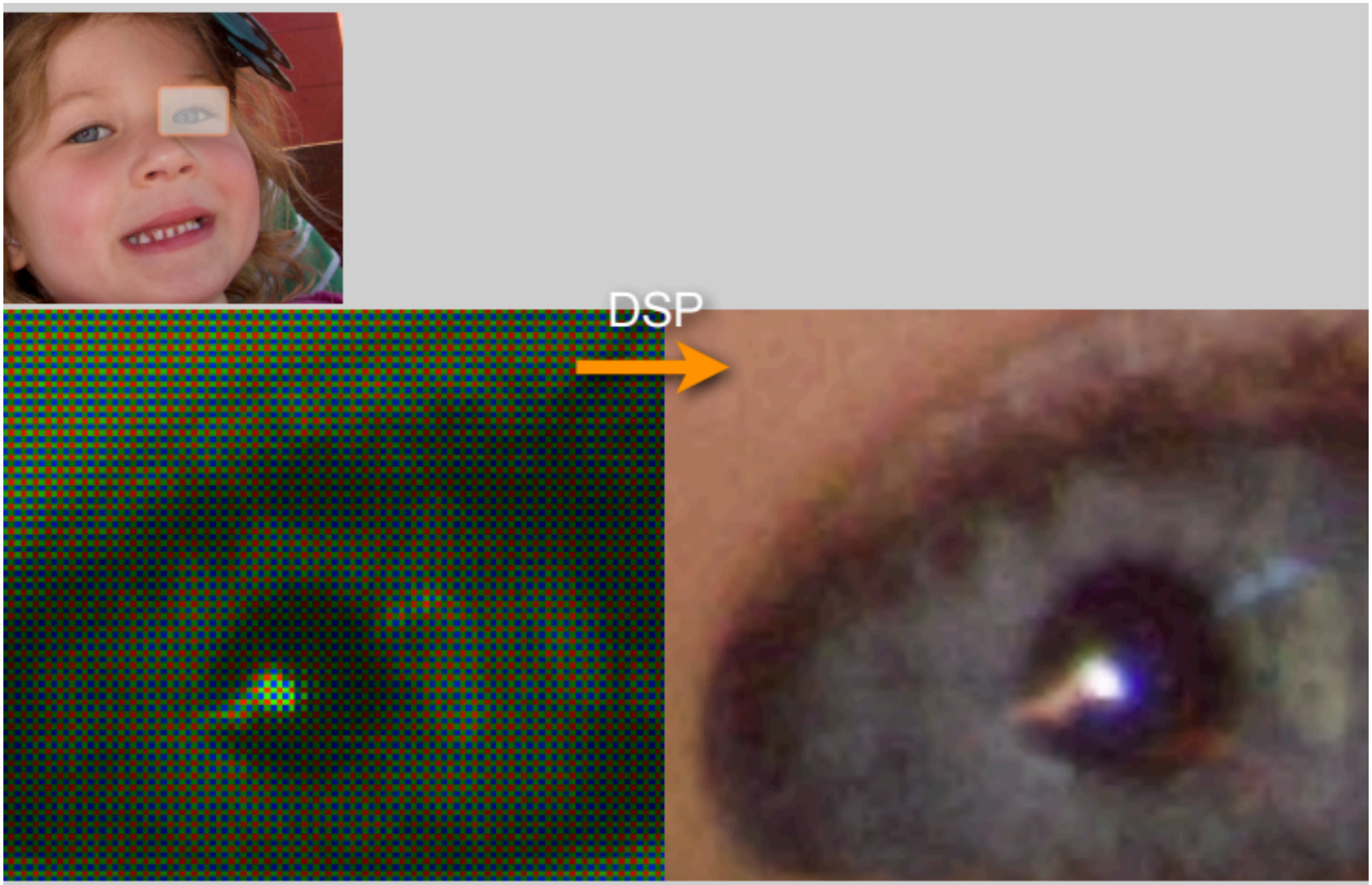


Example II: Digital Imaging Camera

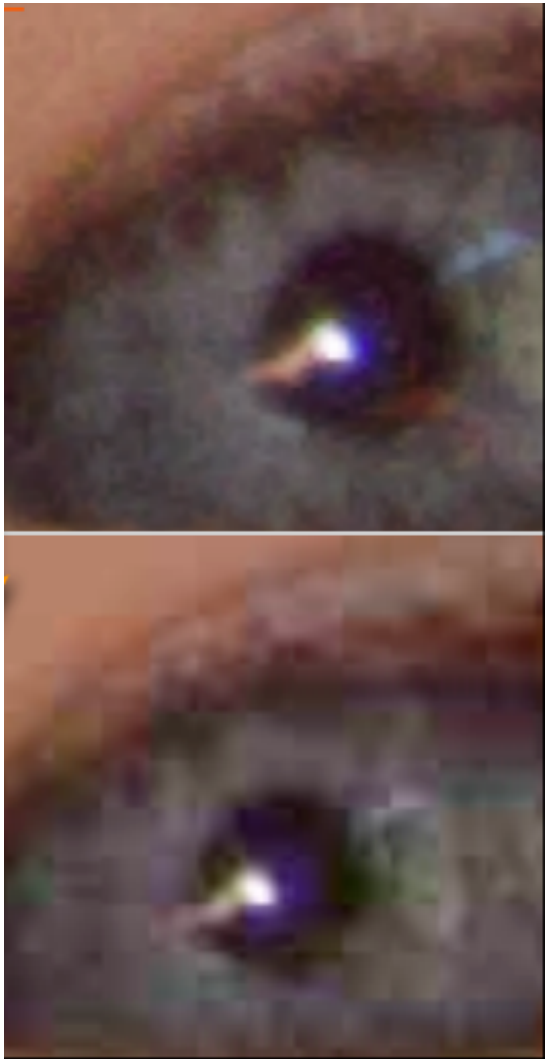


Focus/exposure Control → preprocessing → white-balancing → demosaic → Color transform → Post-processing → Compression

Example II: Digital Imaging Camera



Example II: Digital Imaging Camera



- Compression of 40x without perceptual loss of quality.
- Example of slight overcompression: difference enables x60 compression!

Computational Photography

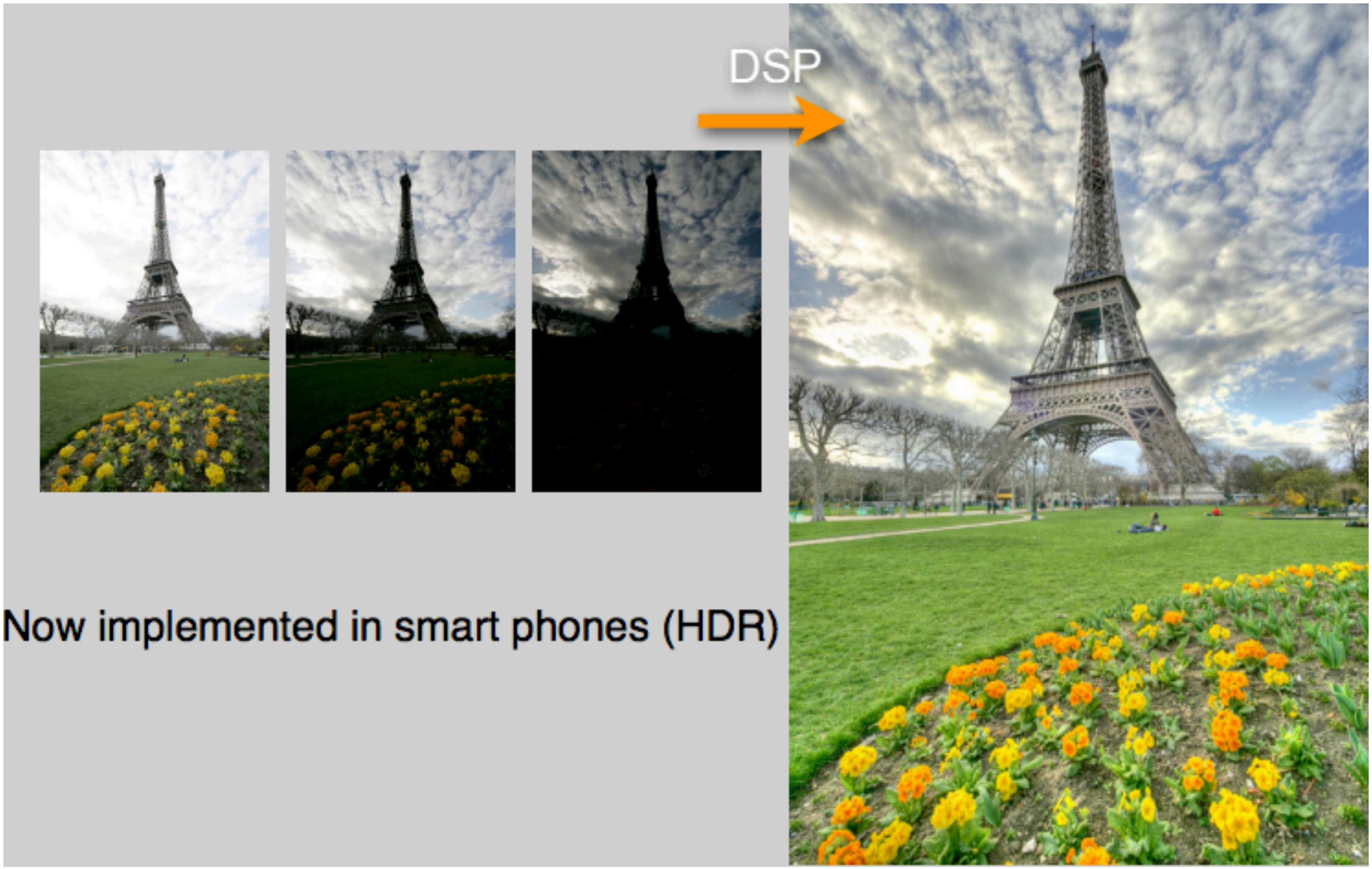


Image Processing - Saves Children

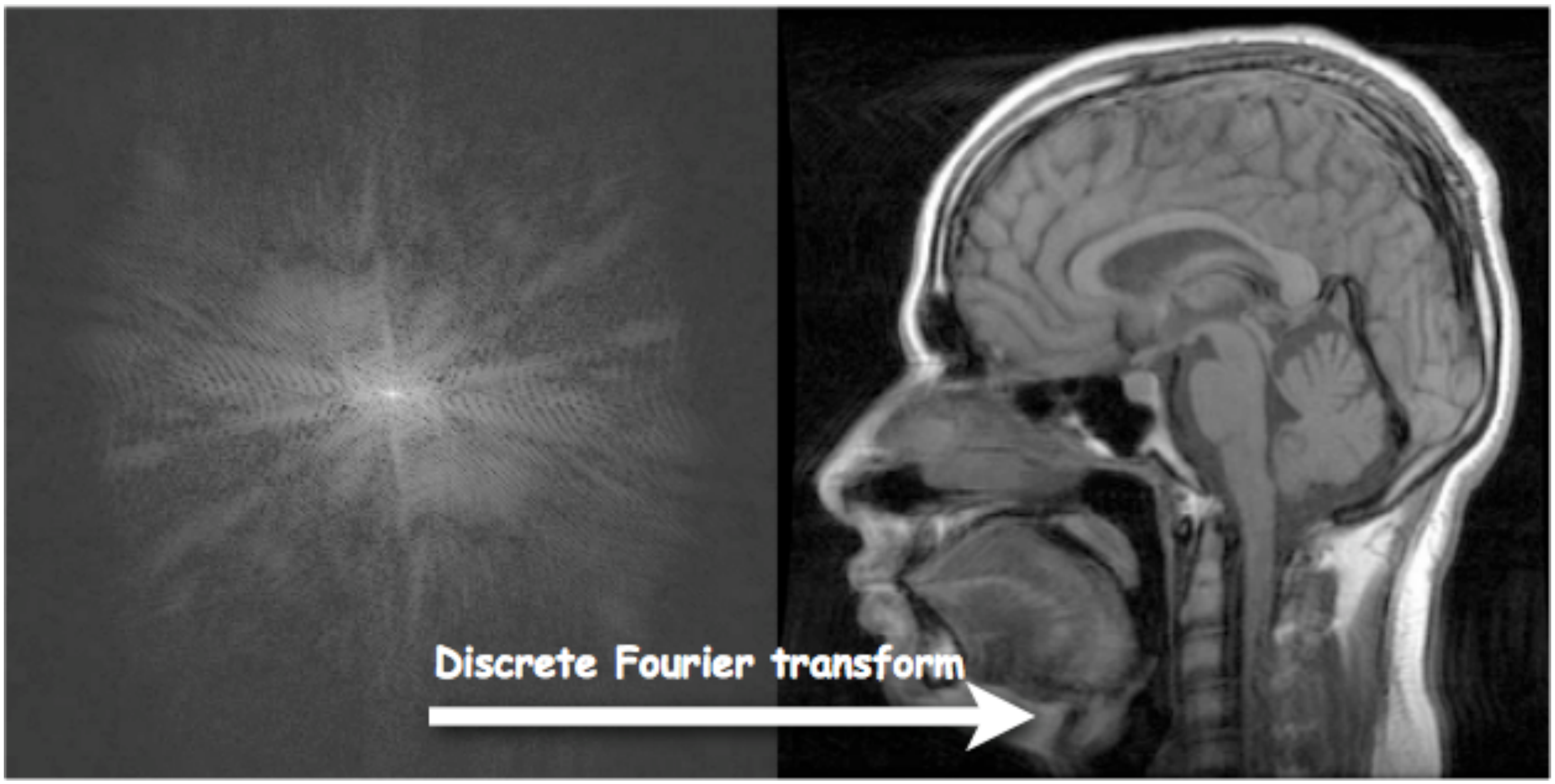
Canadian 'swirl face' pedophile jailed in Thailand

August 15, 2008

Images released by Interpol in 2007 show the 'unswirling' of the internet pictures that led to the capture of Christopher Paul Neil.

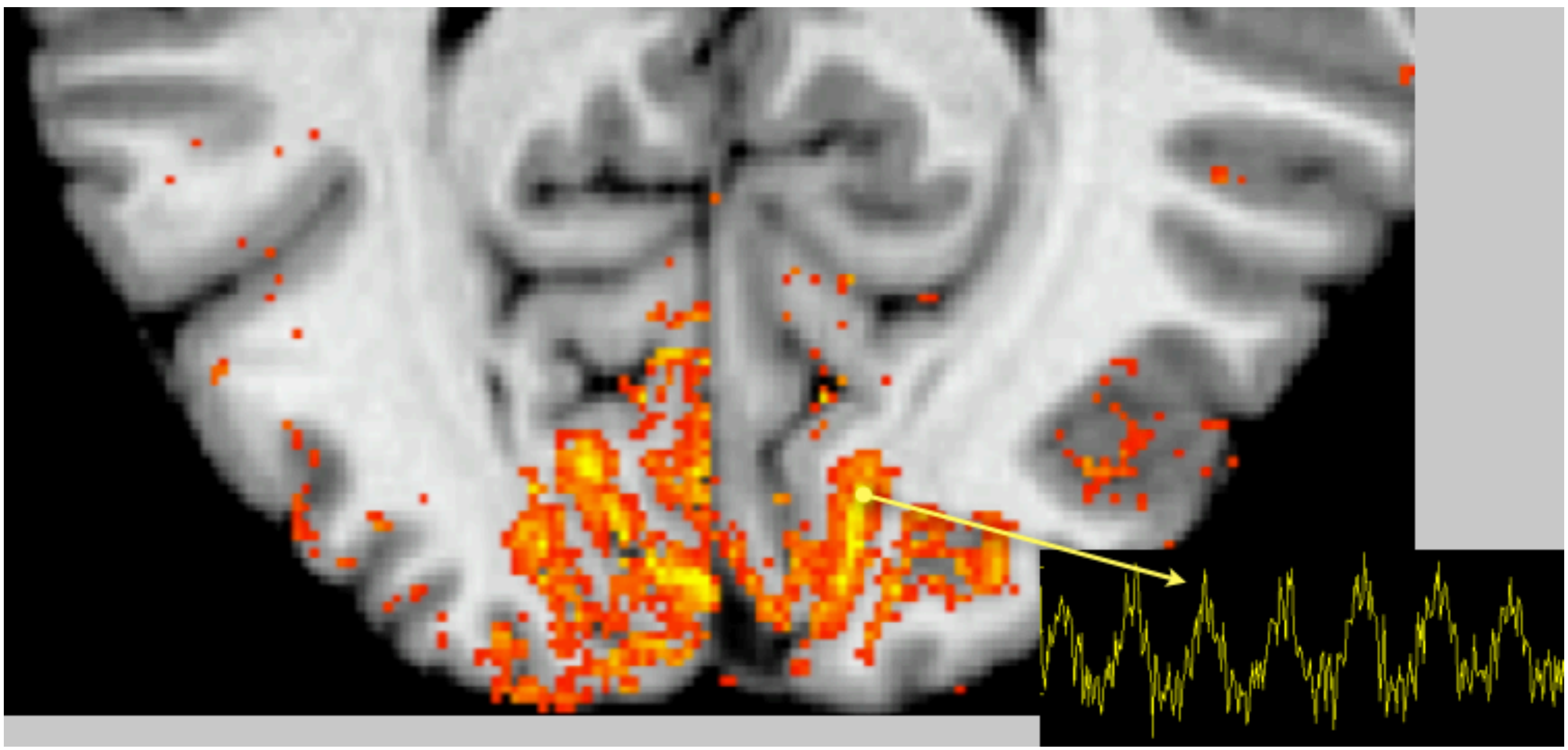
Example III: MRI

k-space (raw data) Image



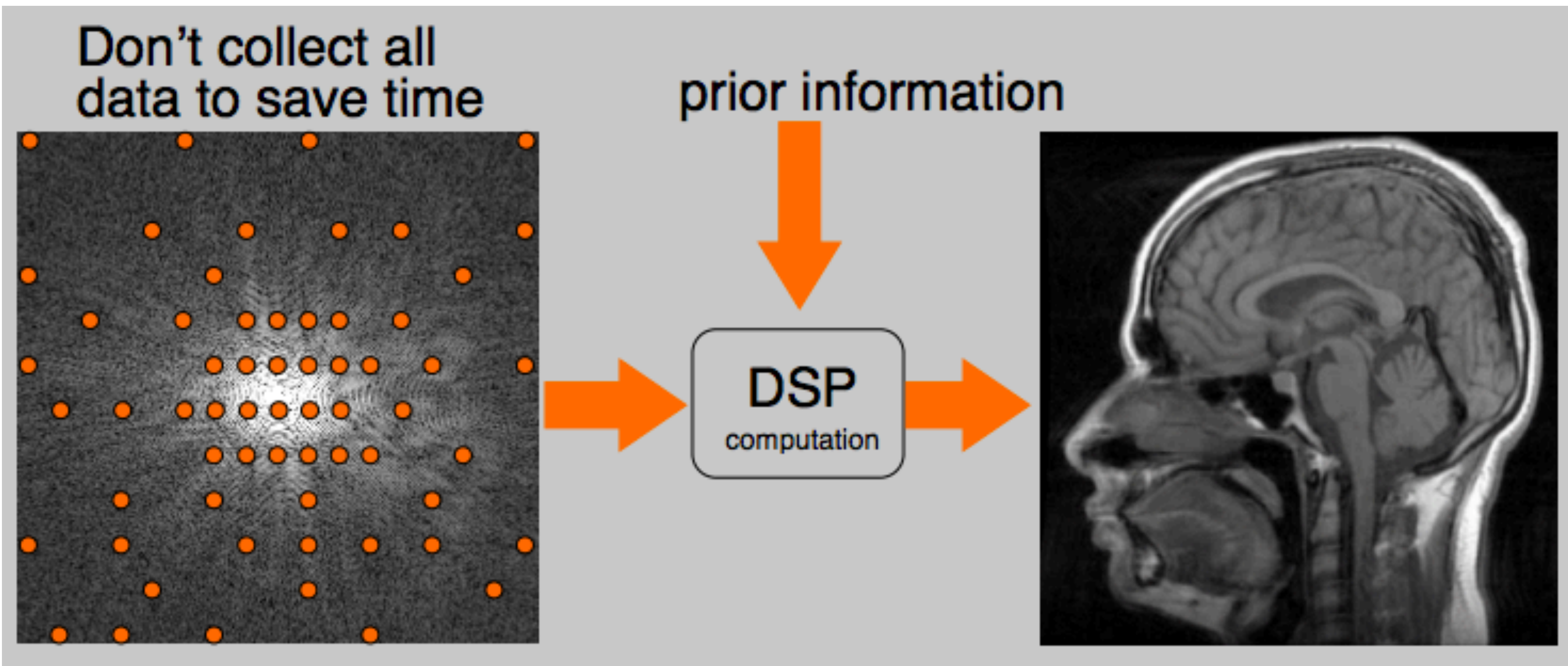
fMRI example

- Sensitivity to blood oxygenation
  - response to brain activity Convert from one signal to another



Compressive Sampling

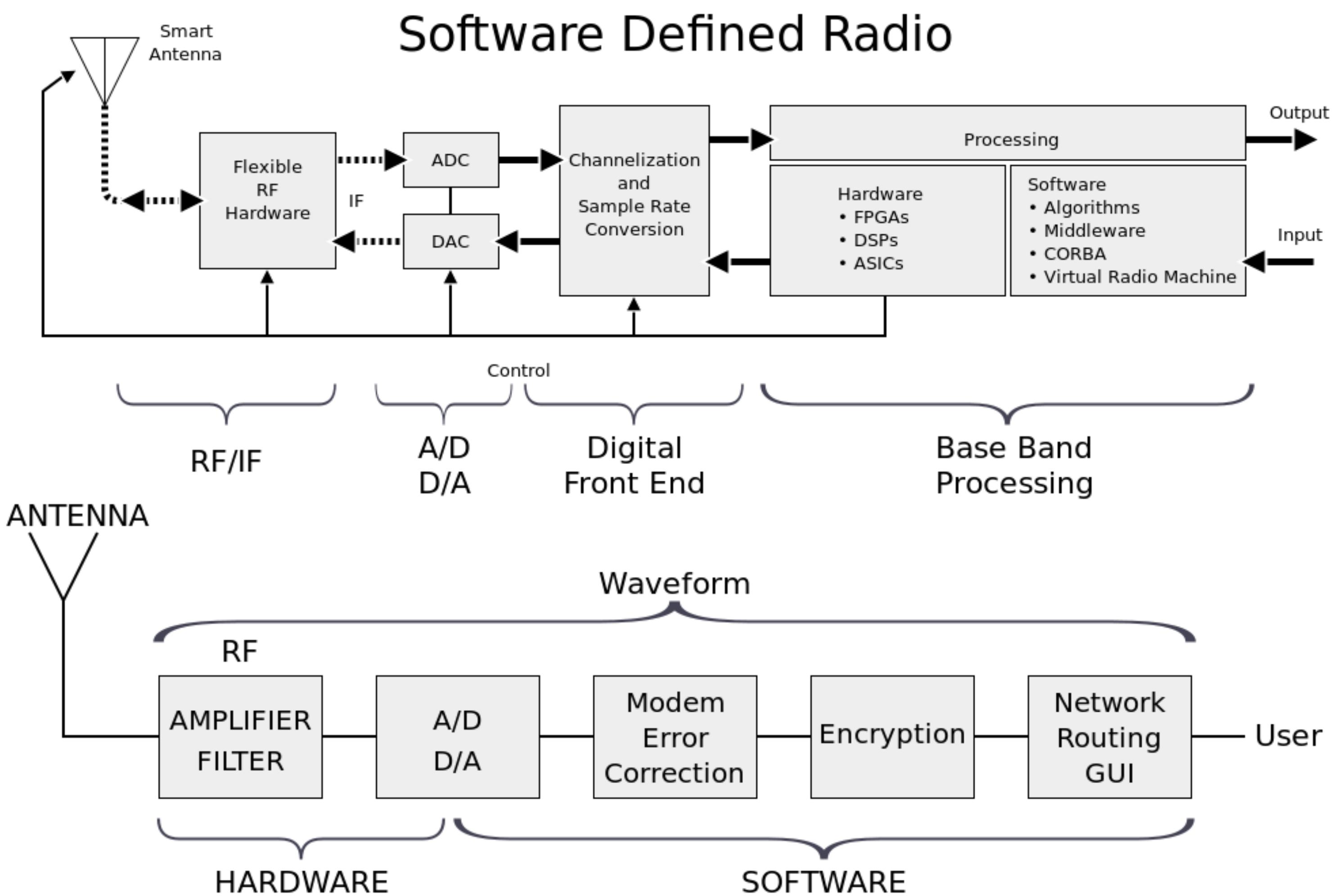
- Compression meets sampling



Example IV: Software Defined Radio

- Traditional radio:
  - Hardware receiver/mixers/demodulators/filtering
  - Outputs analog signals or digital bits
- Software Defined Radio:
  - Uses RF font end for baseband signal
  - High speed ADC digitizes samples
  - All processing chain done in software

Software Defined Radio



Software Defined Radio

- Advantages:
  - Flexibility
  - Upgradable
  - Sophisticated processing
  - Ideal Processing chain
  - not approximate like in analog hardware
- Already used in consumer electronics
  - Cellphone baseband processors
  - Wifi, GPS, etc....

Shameless Plug

- If you are interested in how Analog to digital converters, amplifiers etc…work and how to make them
- Take ESE 568!
- Good to know both sides of the system

Future of ADC design

- Today's ADCs are extremely well optimized
- For non-incremental improvements, we must explore new ideas in signal processing that tackle ADC inefficiency at the system level
  - Compressed sensing
  - Finite innovation rate sampling
  - Other ideas?

From Before: Implementing Compressive Sampling

- Devised a way to randomly sample 20% of the Nyquist required samples and still detect the tremor frequencies within 100mHz
  - Requires post processing to randomly sample!

- Implement hardware on chip to "choose" samples in real time
  - Only write to memory the "chosen" samples
    - Design random-like sequence generator
  - Only convert the "chosen" samples
    - Design low energy ADC

Discrete Time Signals

Signals

DEFINITION

Signal (n): A detectable physical quantity . . . by which messages or information can be transmitted (Merriam-Webster)

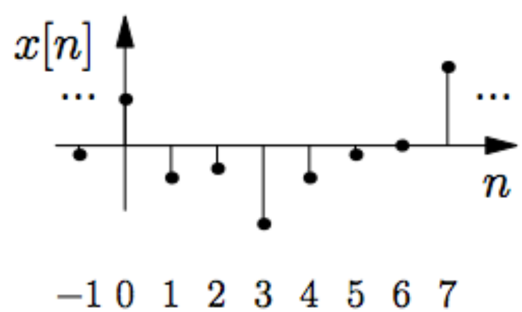
- Signals carry information
- Examples:
  - Speech signals transmit language via acoustic waves
  - Radar signals transmit the position and velocity of targets via electromagnetic waves
  - Electrophysiology signals transmit information about processes inside the body
  - Financial signals transmit information about events in the economy
- Signal processing systems manipulate the information carried by signals

Signals are Functions

DEFINITION

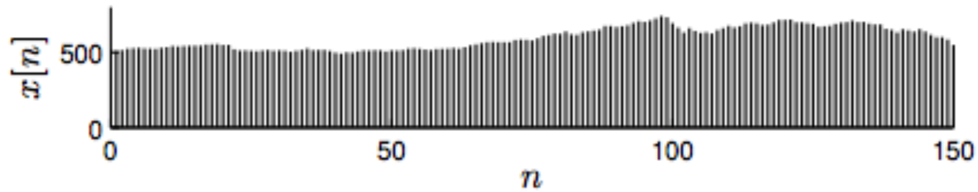
A **signal** is a function that maps an independent variable to a dependent variable.

- Signal $x[n]$: each value of n produces the value $x[n]$
- In this course, we will focus on **discrete-time** signals:
  - Independent variable is an **integer**: $n \in \mathbb{Z}$ (will refer to as time)
  - Dependent variable is a real or complex number: $x[n] \in \mathbb{R}$ or $\mathbb{C}$

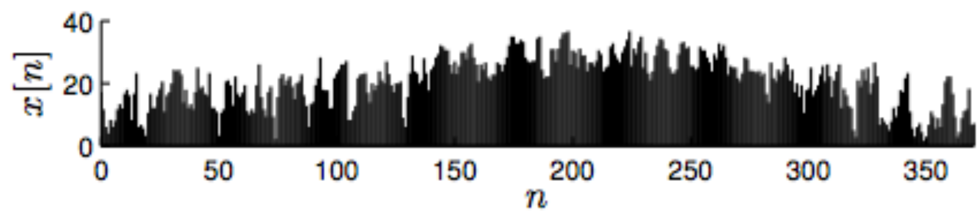


A Menagerie of Signals

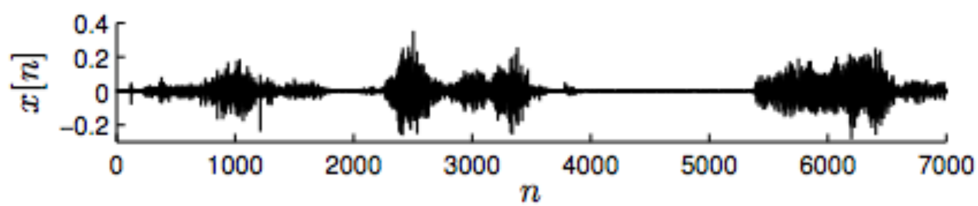
- Google Share daily share price for 5 months



- Temperature at Houston Intercontinental Airport in 2013 (Celcius)

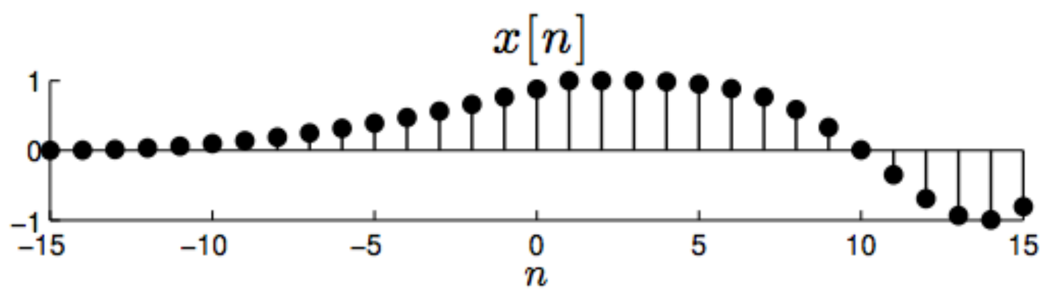


- Excerpt from Shakespeare's *Hamlet*

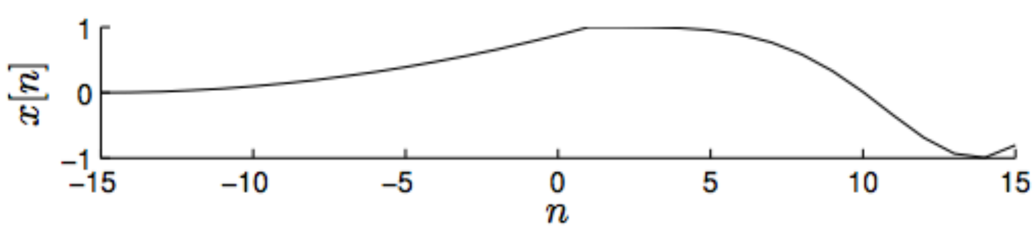


Plotting Signals Correctly

- In a discrete-time signal $x[n]$, the independent variable n is discrete (integer)
- To plot a discrete-time signal in a program like Matlab, you should use the `stem` or similar command and not the `plot` command
- Correct:



- Incorrect:



Wrap up

- Admin
  - Find web, get text, assigned reading…
  - http://www.seas.upenn.edu/~ese531
  - https://piazza.com/upenn/spring2017/ese531/
  - https://canvas.upenn.edu/
- Big Ideas/takeaway
  - Analysis of sampled and quantized signals
- Remaining Questions?