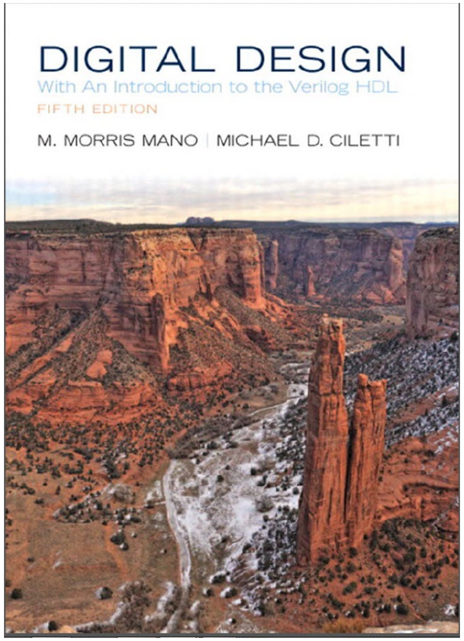
EE 348 Introduction to Logic Design

Course Content

- Textbook
  - Morris Mano and Michael Ciletti, Digital Design, Prentice-Hall Fourth Edition, 2007 or Fifth Edition, 2013



Chapters	Topic
1	Introduction, Number systems, signed arithmetic
2	Boolean Algebra and logic gates
3	Simplification of Boolean functions, Karnaugh map, 2-level implementations
4	Analysis and design of Combinational circuits, MSI circuits (decoders, mux, etc)
5	Basic flipflops, Analysis and design of synchronous sequential logic
6	Registers and serial mode of operation, counters and timing signals.

Grading

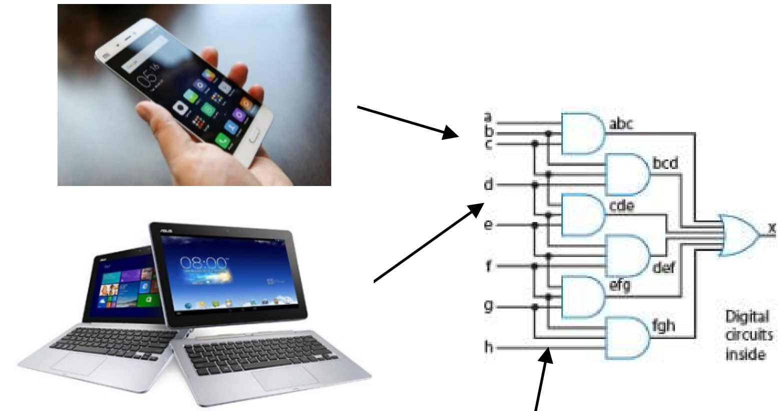
Deliverable	Grade Points
Midterm	25
Final	35
Quizzes (5)	25
Homework assignments (6)	15
Attendance Bonus	10
Total	110

- Policies
  - Any attempt of cheating will be considered as a disciplinary action
  - You can have a makeup if you have a valid health excuse on the exam day
  - Additional info: http://oidb.metu.edu.tr/son-yariyilina-kayitli-olan-onlisans-ve-lisans-ogrencileri-icin-butunleme-sinavlari-yonergesi

Course Objective

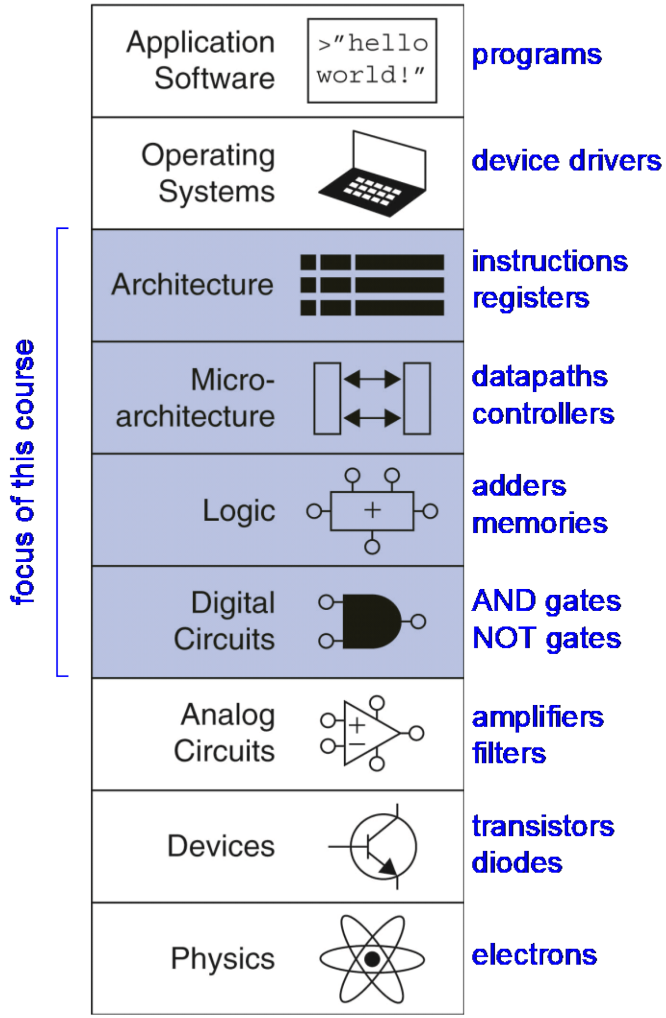
- Why Should You Take This Course?
  - Basics of combinational and sequential logic design
  - → Needed for understanding any computer system
  - Preparation for advanced courses in microprocessors, computer architecture and VLSI
- Motivation
  - Look "under the hood" of computers to get a solid understanding
  - → Gain insight and confidence
  - → Become a better programmer by being aware of hardware issues

Motivation



- ❑ Digital Age
  - Everything is becoming digital
  - Enabled by shrinking and more capable chips
- ❑ Positive Impact
  - Better computing
  - Better devices: Tablets, cameras, cars, smart phones, medical devices,...
  - → "embedded systems"
  - Thousands of new devices every year
  - Designers needed: Potential career direction

Context

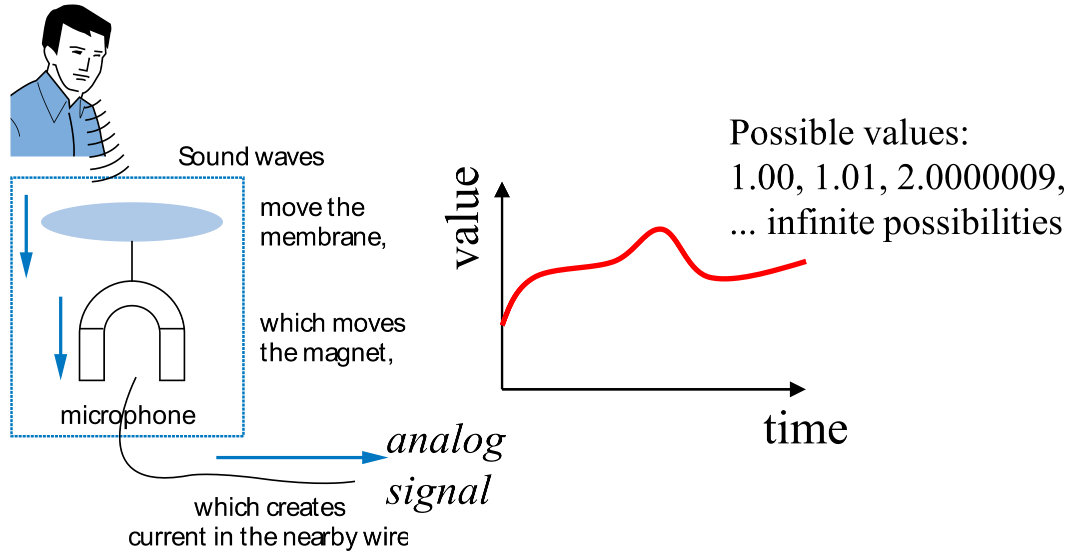


- Software app
- EE 442
- EE 445-446
- EE 348/EE 314
- EE 312
- EE 311
- EE 212
- Phys courses

What Does Digital Mean?

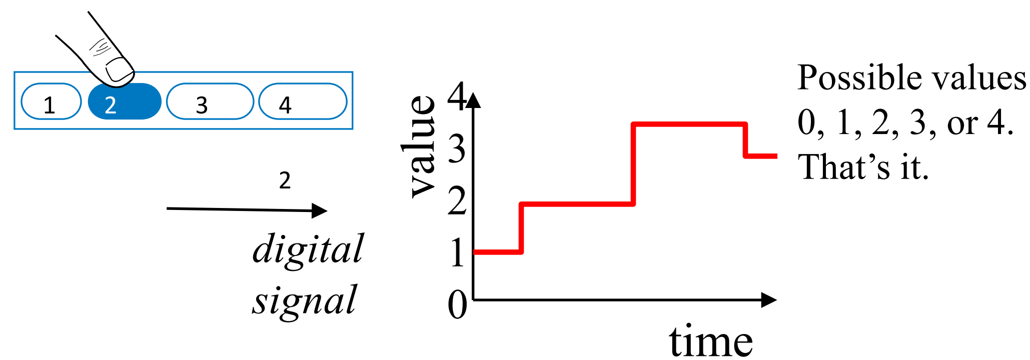
- Analog signal
  - Continuous values

    → infinite number of possible values
  - Example: voltage on a wire of a microphone

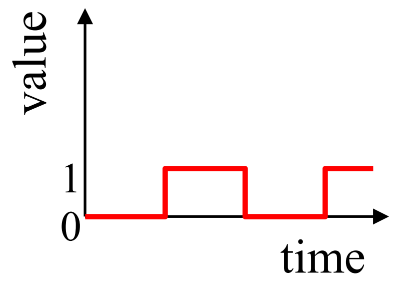


- Digital signal
  - Discrete values

    → Finite number of possible values
  - Example: buttons pressed on a keypad



Digital Signals With Two Values: Binary

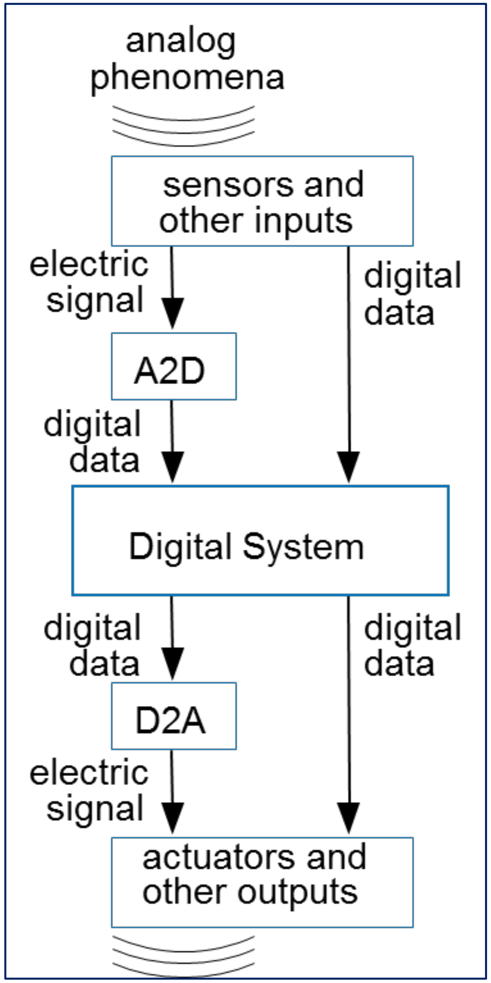


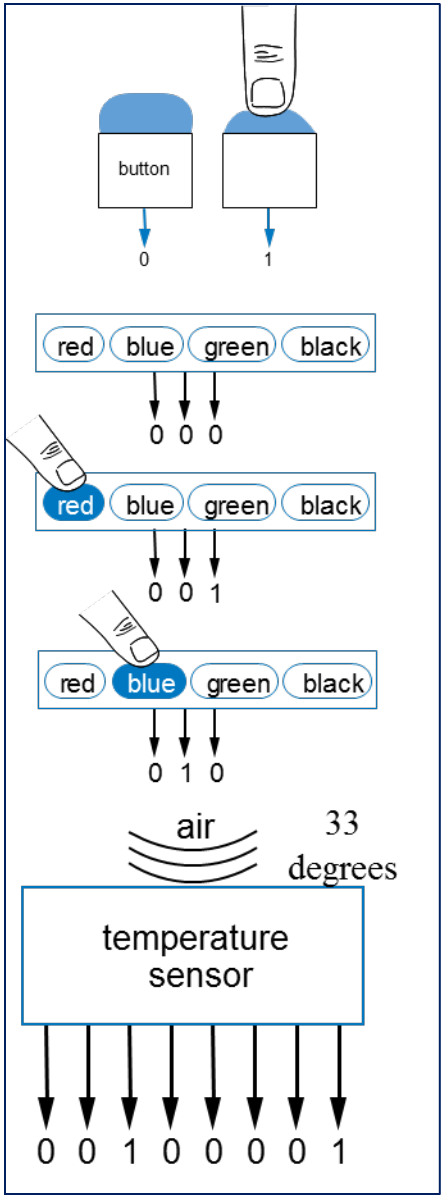
- Binary digital signal
  - Only ***two*** possible values
  - Typically represented as **0** and **1**
- Binary digits
  - One binary digit is a ***bit***

  → We will consider binary digital signals in this lecture
- Popularity of binary digital signals
  - Only need to realize two signal states such as ON and OFF
  - Transistors, the basic digital electric component, operate using *two* voltages (two stable states)
  - Storing/transmitting one of *two* values is easier than three or more

Encoding

- Inherently binary inputs
  - Example: Button
    - Not pressed (0)
    - Pressed (1)
- Inherently digital inputs
  - Need binary encoding
  - Example: encode multi-button
    - red=001; blue=010, ...
- Analog inputs
  - Need analog-to-digital conversion
  - Obtain digital input after sampling
  - Use binary encoding of digital input with bits





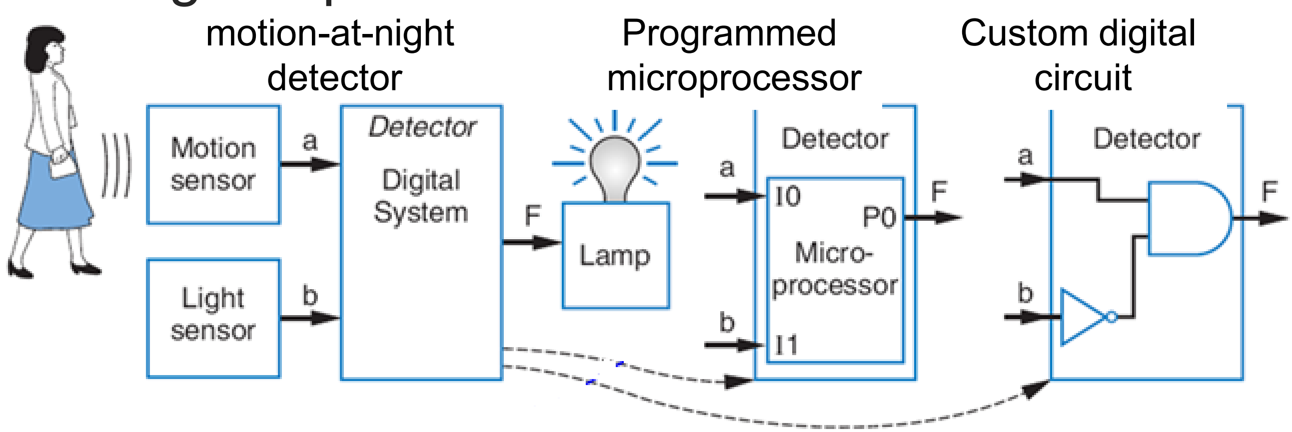
Bits and Bytes

- 1 Byte: 8 bits
- Common metric prefixes
  - kilo (thousand, or 10^3), mega (million, or 10^6), giga (billion, or 10^9), tera (trillion, or 10^{12}), peta (10^{15})
  - e.g., kilobyte, or KByte
- Note: metric prefixes also commonly used inaccurately
  - 2^{16} = 65536 commonly written as "64 Kbyte"
  - Typical when describing memory sizes
  - Just know: 2^{10}=1024=1K
  - 1M=1K*1K=2^{10}*2^{10}=2^{20}
- Also watch out for "KB" for kilobyte vs. "Kb" for kilobit

Implementing Digital Systems

- Designing Digital Circuits
  - Custom design for each task (large design time)
  - (Up to 300 times more) power efficient and fast execution
- Programming Microprocessors Circuits
  - Design flexibility (just change the program for a different task)
  - Possibly more expensive and slower execution

→ Choice based on design requirements

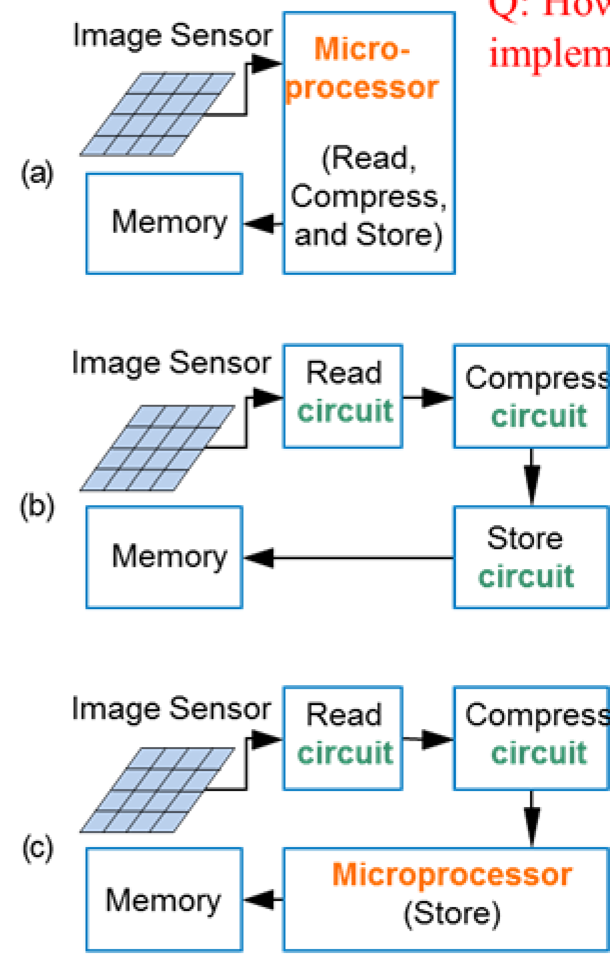


Design Tactics

- Common Procedure
  - Designers partition a system among microprocessor and custom digital circuits
- Example
  - Sample digital camera task execution times on a digital circuit vs. Microprocessor

Task	Microprocessor	Custom Digital Circuit
Read	5	0.1
Compress	8	0.5
Store	1	0.8

Q: How long for each implementation option?



(a) 5+8+1 =14 sec

(b) .1+.5+.8 =1.4 sec

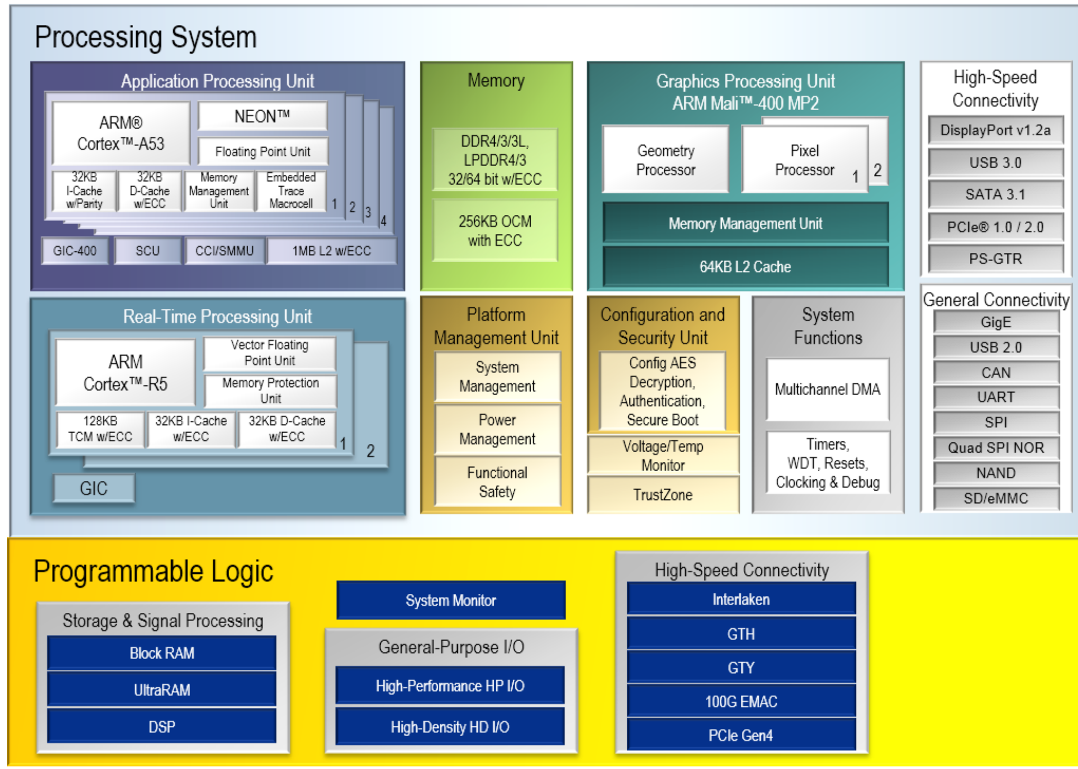
(c) .1+.5+1 =1.6 sec

Good compromise

→ EE348 is on digital circuits. Microprocessors → EE447

Cutting Edge

- Zynq® UltraScale+™ MPSoC devices
  - 64-bit processors
  - Soft and hard engines for graphics, video, signal processing
  - Real-time processor
  - Programmable logic
  - GPU devices
- Applications
  - Real-time control
  - 5G Wireless
  - Advanced driver-assistance systems
  - Internet-of-Things ..... etc.



https://www.xilinx.com/products/silicon-devices/soc/zynq-ultrascale-mpsoc.html

EE 348 Introduction to Logic Design