Lecture 28

- Logistics
  - HW8 and Ant extra credit problem are due now
  - All lab must be done today.
  - Review session Sunday 3pm EEB 045
  - Final Exam 12/8 Monday here 8:30am
- Today
  - Review materials from the entire class
  - Evaluation: leave last 10-15 min for this

"Why" take CSE 370

- Required (okay, but let's talk about why it is required and will be useful for your future)
- Most basic building blocks of computer science (0's and 1's)
- It is important to understand how they are used as baseline for more complex operations (adding, storing, other logic like if/while)
- It is like studying anatomy of neurons and neuronal connections if you want to become a neuroscientist (even if you become a computational neuroscientist).
- It is good to understand what can be implemented in hardware, and why understanding hardware is useful even if you are going to be a software person
- Understand how some of the technology you interact with on daily basis (memory stick, vending machine, etc) at the hardware logic level.
- Knowledge gained in this course is used directly in industry/research

What you should know

- Combinational logic basics
  - Binary/hex/decimal numbers
  - Ones and twos complement arithmetic
  - Truth tables
  - Boolean algebra
  - Basic logic gates
  - Schematic diagrams
  - Timing diagrams
  - de Morgan's theorem
  - AND/OR to NAND/NOR logic conversion
  - K-maps (up to 4 variables), logic minimization, don't cares
  - SOP, POS
  - Minterm and maxterm expansions (canonical, minimized)

I like Pink and Blue but not Yellow...

What you should know

- Combinational logic applications
  - Combinational design
    - Input/output encoding
    - Truth table
    - K-map
    - Boolean equation
    - Schematics
  - Multiplexers/demultiplexers
  - PLAs/PALs
  - ROMs
  - Adders

4529
+ 34532

????

What you should know

- Sequential logic building blocks
  - Latches (R-S and D)
  - Flip-flops (D and T)
  - Latch and flip-flop timing (setup/hold time, prop delay)
  - Timing diagrams
  - Asynchronous inputs and metastability
  - Registers

Remember that the last number was 1

What you should know

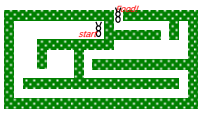
- Counters
  - Timing diagrams
  - Shift registers
  - Ring counters
  - State diagrams and state-transition tables
  - Counter design procedure
    1. Draw a state diagram
    2. Draw a state-transition table
    3. Encode the next-state functions
    4. Implement the design
  - Self-starting counters

1, 2, 3, 4, ...

What you should know (Final exam focus is here though exam is cumulative)

- Finite state machines
  - Timing diagrams (synchronous FSMs)
  - Moore versus Mealy versus synchronized Mealy
  - FSM design procedure
    1. State diagram
    2. state-transition table
    3. State minimization
    4. State encoding
    5. Next-state logic minimization
    6. Implement the design
  - State minimization
  - One-hot / output-oriented encoding
  - FSM design guidelines
    - Separate datapath and control
  - Pipelining, retiming partitioning basics

The last coin was 25cents and already had 50cents deposited so let's pop out a soda



Final exam logistics

- 8:35 – 10:20 (1 hour and 45 minutes long)
- Materials: cumulative but focus on Lectures 20 – 27 and HW 7 and 8
- Closed book/notes, no calculator
- Scratch papers provided
- Just have your pencil/pen and eraser
- Raise hand for questions (don't walk to get help)

Thank you

Thank you for the great quarter together

I had fun (could you tell?) and hope you did too

Send me an email or drop in for questions about CSE, research, grad school, jobs, life, etc.

Good luck on final exams and have a great holiday season!