# Recitation 11

Sequential Logic

# Today's agenda

- We will discuss in recitation
  - Sequential Logic
  - Finite State Machines
- For homework tonight
  - R11
    - You will create a truth table for a given finite state machine

# Building Blocks

For sequential logic

# Sequential Logic

- There is memory
  - Outputs depend on prior state as well as the current inputs
  - State can be stored and used later
- We rely on clock signals
  - Clock signals tell us when things should happen
  - We should only write to state when the clock is set a certain way

# SR Latch

- Constructed from two NOR gates
- You can either **S**et the latch (make it remember 1), or **R**eset it (make it remember 0)
- Two inputs: S and R
- Two outputs: Q and NOT Q
  - Q is what it remembers, NOT Q is the opposite
- Both S and R cannot be 1 at the same time, or sadness occurs

# D Latch

- Constructed from some additional logic and an SR Latch
- Two inputs: C and D
- You can have the latch remember D as long as C is true
- Two outputs: Q and NOT Q
  - Q is what the latch remembers, NOT Q is the inverse
- Ensures that S and R inputs to the SR Latch aren't both true

# D Flip Flop

- Constructed from some additional logic and two D latches
- Same inputs and outputs as D latches
- But, the output is only stored on a chosen clock edge

# Finite State Machines

# Finite State Machines

- There are a number of states, inputs, and outputs
- To the beat of the clock, we read in inputs and go to new states, and set the outputs
- Both the output and the next state are defined by the current state and the inputs
- You can think of it as following a flow chart “when I’m on this step, and this is true, I go here”

# An FSM Example

- The NYC Subway Turnstile
- There is a lock controlled by the FSM
- If the user didn't pay yet then the lock is active and the user can't push through
- If the user pays the lock unlocks until they push through
- Draw an FSM for this
- Write out a truth table
- Create the circuit

# Another FSM Example

- Say we have an alarm system with a motion sensor
- The owner of the alarm system can press a button to arm the alarm
- The owner of the alarm system can press a different button to reset the alarm
- We want the alarm to go off if the motion sensor sees something, but only if the alarm has been armed
- If the owner resets the alarm, it both stops going off and is no longer armed
- The alarm goes off until the owner resets it
- Draw an FSM for this
- Write out a truth table
- Create the circuit