CS 150
Lab Lecture 2

Schematic Entry & Simulation, Part 010_2
Finite State Machines
Laura Todd, 1-26-01

I hope not…

Overview

- Objectives
- Prelab
- Brief Description
- To-do
- Hints, tips, and tricks

Objectives

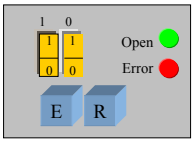
- Practice using Xilinx software
- Enter a graphical gate schematic from logic equations
- Simulate the schematic
- Practice high-level description-to-gates design flow
- Work in a group

Prelab

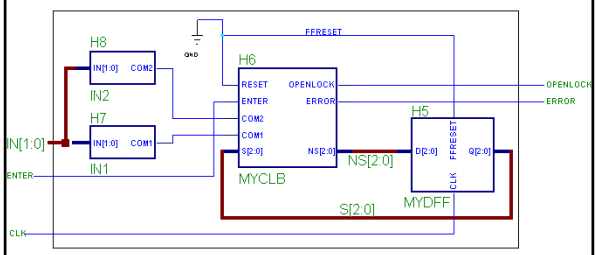
- **Get a partner in YOUR lab section**
- Complete the **IN1** and **IN2** blocks
- Write a script file to test your design

Brief Description – "High Level Spec"

- The high-level spec is a description of the problem in "plain English"
  - This is a very general description of the system you're trying to build, in our case, a combination lock.



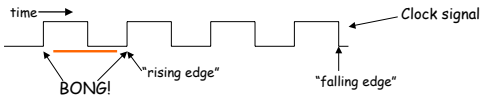
Brief Description – Block Diagram



System-level view: these are the building blocks we need

Brief Description – the Clock

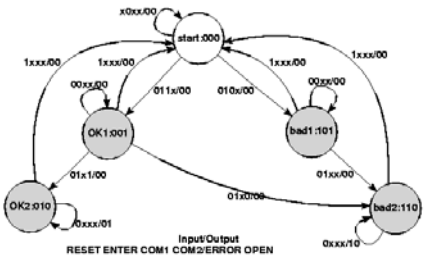
- The "clock" keeps the FSM synchronized.



Orange: values of combinational logic can change

- When the clock goes "BONG!" the FSM will change state
- The time from one rising edge to another is called the "clock cycle" time.

Brief Description – FSM



Logical view: the finite state machine

Brief Description – the IN* blocks

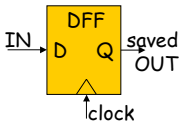
- The IN1 and IN2 blocks compare the code the user inputs to the "secret" code that will open the lock.
- Choose your own combination. Write it down.
- The two combinations must be different:
  - Good: 01 11
  - Bad: 10 10
- IN1 and IN2 are very simple blocks. Just some AND gates and inverters.

Brief Description – MyCLB

- CLB = Combinational Logic Block => AND, OR, INV…
- The MyCLB block acts as the controller for our lock, it's the brains
- Truth table and logic equations are in the printout
  - Implement this block during lab time.
- KISS-Keep It Simple & Stupid
  - Use the "multi-level" equations, they're much easier to debug

Brief Description – MyDFF

- What the heck are D flip-flops????
  - D flip-flops are memory elements
  - Put what you want to save on D
  - When the clock has a rising edge the DFF will look at the value on D and move it to Q
  - Now until the next rising edge on the clock, Q will remain the same, giving us a saved output



- This is the simple version: lots more later in class!
- Lab uses the "FDR" model.
  - R stands for "Reset to Zero"

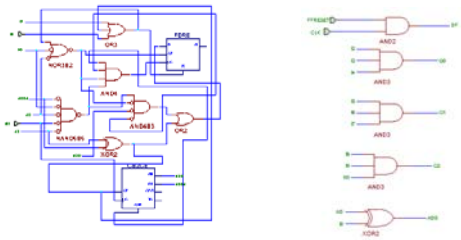
ToDo's – What you need for credit

- Enter the IN1, IN2, MyCLB schematics
- Make a test script for your CLB. Run it. Make sure it's correct.
- Make the MyDFF block.
- Wire up all the individual pieces you just made.
- Write a script to simulate scenarios on the full lock
  - Successful/Unsuccessful entries of the combination
  - Use of the RESET and ENTER buttons
  - Do all the paths on the state diagram work correctly?
  - Create a log file to show your TA

Hints, Tips, and Tricks

- Buses are your friends.
  - CS150 buses: collections of wires that have a similar purpose
  - Usually same wire name + a number, as in DATA7, DATA6, … DATA1, DATA0
  - Simplify, simplify, simplify…
- Scripts are your friends.
  - Note: the pipe character " | " will comment out a line.
- Don't print things!
- Xilinx software is buggy. When in doubt, restart it.
- Don't draw spaghetti wires =>
  - Too easy to mislabel wires
  - Harder to debug

Hints, Tips, and Tricks



Spaghetti vs. Not Spaghetti

The End… for now.

Ya ain't seen nuthin' yet…