CSCE 312 Lab 3

By Dr. Eun Jung Kim

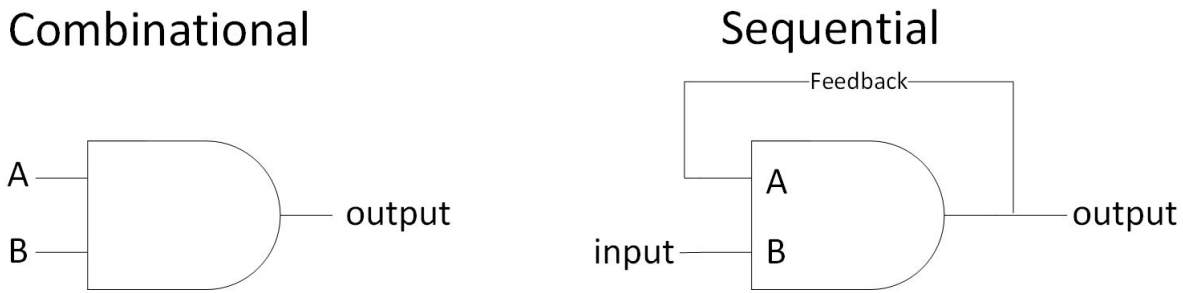
Announcements

1. Lab 4 is a group Project. Fill out this google sheet with details about your group members.
   a. Group Size: 3
2. Lab 3 deadlines:
   a. Demo: February 22, 2024
   b. Report: February 25, 2024

No late demo will be considered

Combinational Logic vs Sequential logic

- Combinational logic: Output depends on only current inputs
- Sequential logic: output depends on both current input & previous output



Why we need sequential circuits

To implement,

- State Machines,
- Counters,
- Memory, etc

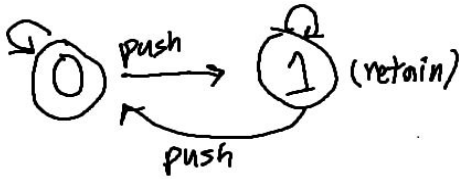
Problem 1

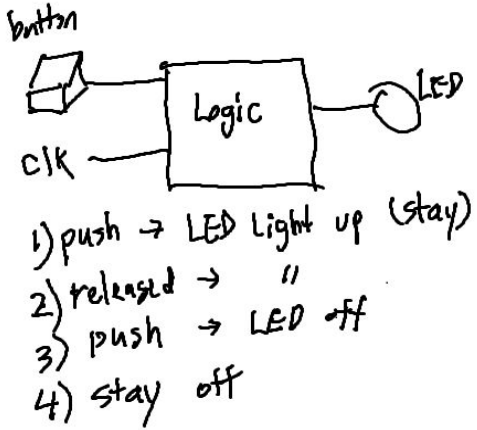
Create a D flip flop

- Using 2 D-Latches
- The latches should be connected using master-servant design
- Build the full circuit using **only NAND** gates.

Problem 2

Design a toggle Button

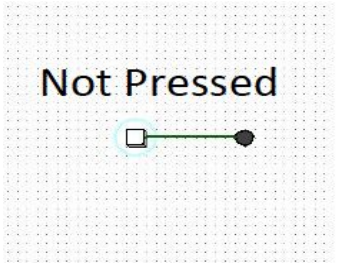


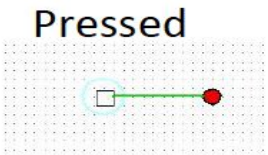


Default Button Behavior

This is not what we want
We want Toggle Push Button

- Pressed(& Released): LED on and stay
- Again Pressed(& Released): LED off and stay





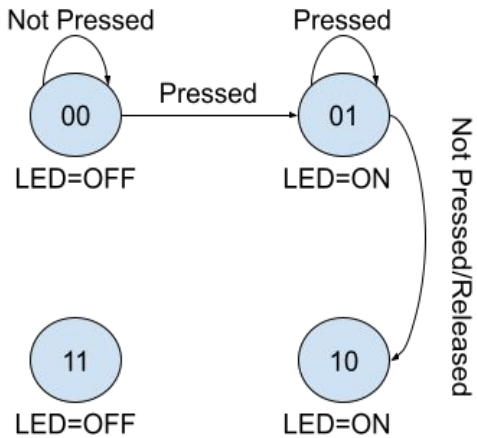
How can we achieve that!!

States:

1. State 00 - Initial State, Button not Pressed, LED OFF
2. State 01 - Button Pressed Once, but not Released, LED ON
3. State 10 - Button Pressed & Released Once, LED ON
4. State 11 - Button Pressed Again, but not Released, LED OFF

When Button is Released, it will move to State 00 and LED OFF

Figure out all the transitions



Convert State Diagram to a Truth Table

S1, S0: Previous State

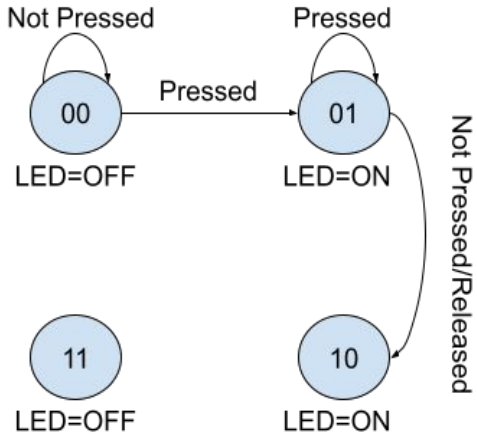
B: Button Pressed or Not

N1, N0: New State

LED: LED ON or OFF

Figure out all the transitions and form a truth table

S1	S0	B	N1	N0	LED
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	1	0	1
0	1	1	0	1	1
…	…	…	…	…	…

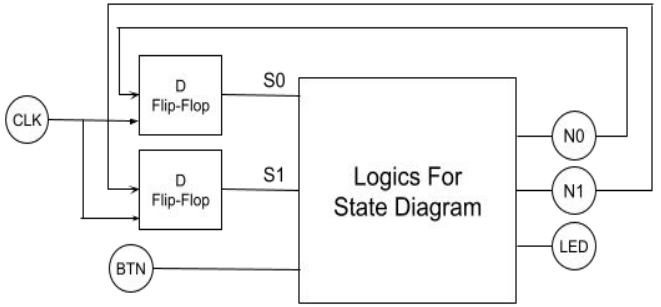


Suggested Logisim Implementation

Figure out Boolean Expressions for N1, N0 and LED

You can use any built-in circuits from Logisim

S1	S0	B	N1	N0	LED
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	1	0	1
0	1	1	0	1	1
…	…	…	…	…	…



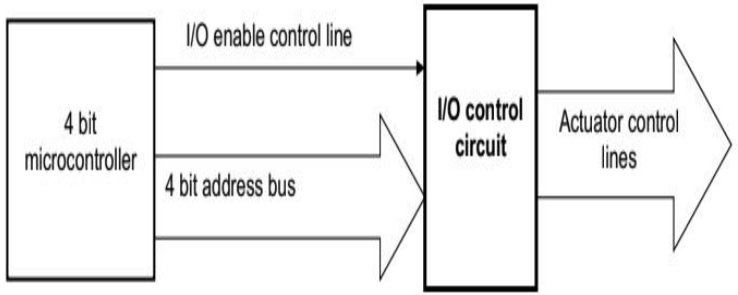
Problem 3 Description

4-bit address(B3 B2 B1 B0)

B3 = 1: Enable

B3 = 0: Disable

Actuators	Binary(B2 B1 B0)
Headlights	000
Door 1	001
Door 2	010
Door 3	011
Door 4	100
Left Indicator	101
Right Indicator	110
Wipers	111

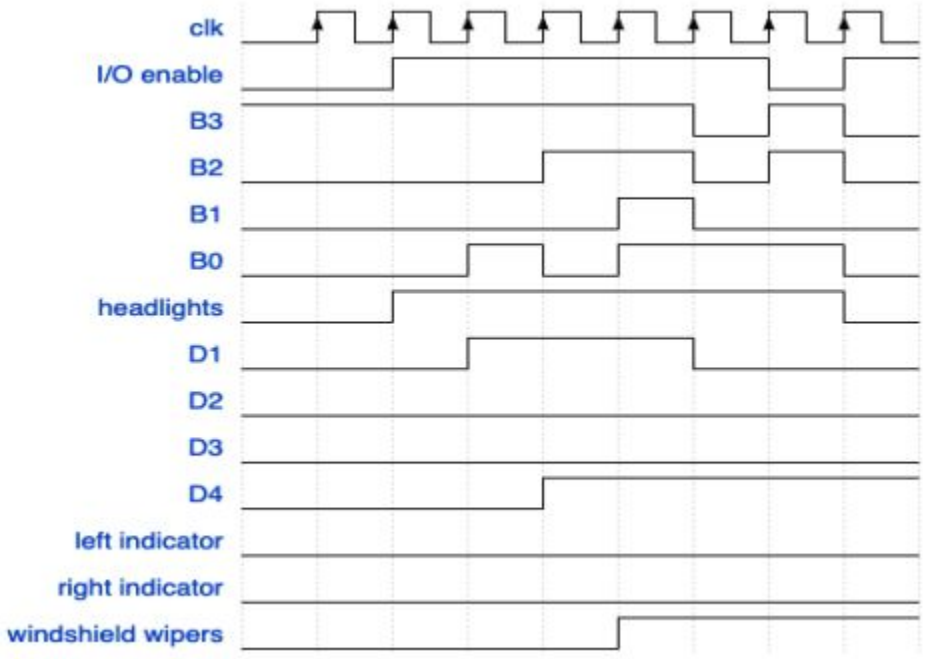


Problem 3

Assume, initially all actuator outputs are 0

Address bus lines				I/O Enable	Output actuator control lines								Comments
B3	B2	B1	B0		Headlights	D1	D2	D3	D4	Left Indicator	Right Indicator	Windshield wipers	
X	X	X	X	0	0	0	0	0	0	0	0	0	Nothing happens as I/O disabled
1	0	0	0	1	1	0	0	0	0	0	0	0	Headlights are ON
1	0	0	1	1	1	1	0	0	0	0	0	0	Door 1 locks
1	1	0	0	1	1	1	0	0	1	0	0	0	Door 4 locks
1	1	1	1	1	1	1	0	0	1	0	0	1	Wipers are ON
0	0	0	1	1	1	0	0	0	1	0	0	1	Door 1 unlocks
1	1	0	1	0	1	0	0	0	1	0	0	1	Nothing happens as I/O disabled
0	0	0	0	1	0	0	0	0	1	0	0	1	Headlights are OFF

Problem 3: Timing Diagram Example



Main TODOs for Problem 3

1. Design the circuit.
2. Draw timing diagram for the 8 clock cycles given in the manual.

Helpful tips for Problem 3

- Store the state of the actuators.
- Use A2,A1,A0 to figure out which actuator to change. Use A3 to figure out whether turn on or off.
- If I/O is off, all actuators should maintain the previous states.
- For timing diagram, assume clocks are positive edge triggered.
- Suggested tool to create timing diagram
  - Wavedrom

Thank You