UNIVERSITY OF VIRGINIA
DIGITAL LOGIC DESIGN
STUDIO ASSIGNMENT 7

This assignment is to be completed without any aid from anyone other than the teaching staff for this class. You may use only the class text, other materials provided to you by the teaching staff for this class, and your own class notes to complete this assignment. You must not offer or provide aid to others taking this class. Submission of any part of this assignment represents your affirmation that you have complied with these requirements.

OBJECTIVE

This studio assignment will give you the opportunity to design a finite state machine (FSM) to detect a specific input sequence.

PROBLEM DESCRIPTION

Design a finite state machine with one input X and a 2-bit output **Z**=Z_1Z_0. When the state machine starts, the output is **Z**=11 until it receives its first one input. When it receives the its first one input, the output changes to **Z**=00 to signal that the system is ready to receive a secret code. The output stays at **Z**=00 while it looks for one of two valid secret codes (111 and 010) after the initial first one. If it receives 111 then the output changes to **Z**=01 and stays at that value forever (i.e. it stops looking for a secret code). If it receives the other valid secret code (010) first, then the output changes to **Z**=10 and it stays at that value forever. A Moore machine will be easier for this assignment than a Mealy machine.

PRE-WORK

1. Design your finite state machine, using whatever state names make the most sense to you as a human designer.
2. Change your state names to be something innocuous, for example A, B, C, etc.
3. Now assign binary codes to the state names A, B, C, etc. You need to determine how many flip flops are needed to accommodate your design. This state assignment state is arbitrary and a different assignment will result in different equations. That is fine but you need to document your design process very carefully so that the instructors can follow your procedure since it's likely to be different than theirs.
4. Derive the excitation and output functions for your design. Use D flip flops in this step. Implement your design using *Logisim.*
5. Derive the excitation and output functions for your design. Use T flip flops in this step. Implement your design using *Logisim.*
6. For both circuits, use a hex display element to display the current state. Use a pair of LED's to display the output function. Include a "reset" button that asynchronously puts the FSM in its initial state.

IN STUDIO

Bring your completed *Logisim* file and your documented design procedure to the studio to be reviewed by your studio instructor. Be prepared to answer questions about your design and make suggested changes to demonstrate understanding.

GRADING SCALE

- 2 points for correct FSM that is very neatly drawn and clearly described to the instructor.
- 3 points for D FF design and implementation in Logisim
- 3 points for T FF design and implementation in Logisim
- 1 point for reset button
- 1 point for professionalism