Lab 7

Finite State Machines (FSMs)

Part 1

Write Verilog modules to describe the FSMs explained in the following videos:

- https://youtu.be/oSP5Gl1Z8DI (`dual_sequence_detector.v`)
- https://youtu.be/doItlPBETEM (`even_odd_conseq00.v`)
- Write appropriate testbenches for both FSMs

There is no FPGA implementation for this part.

Part 2

Design a finite state machine that has an input `x` and output `y`. The output should be asserted whenever `x = 1` or `x = 0` for three consecutive clock pulses. In other words, the FSM should detect the sequences 111 or 000. Overlapping sequences are allowed, so a sequence of four or five 0s or 1s should also output 1.

- Draw a state diagram and include it with your submission
- Write a Verilog module to describe the FSM (conseq_sequence.v)
- Write a testbench to verify your design

There is no FPGA implementation for this part.

Submission check list:

[ ] All Verilog code you generated or modified
[ ] All testbenches written
[ ] Embed all screenshot of your testbench output in your README.md
[ ] Embed all block diagram or state diagrams generated in your README.md