

## ECE 1315 - Lab #8: FSM II

Design and build a synchronous finite state machine with one input variable,  $x$ , and one output variable  $z$ , that behaves as follows:

$z = 1$  if  $x$  was 1 for *exactly* two of the preceding three clock cycles (including the current cycle). Otherwise  $z = 0$ .

For example,

```

Time→
x:  001101001000111100010101101100...
z:  000111000000010010000101111110...

```

In generating your state diagram, assume that the input variable,  $x$ , initially has been 0 for many clock cycles. Use JK Flip Flops (jkff in QuartusII) to construct your circuit in addition to whatever combinational logic you need. Minimize the total cost of your implementation with respect to the associated costs in Table 1.

**Table 1: Available Logic Elements and Associated Cost**

Logic Element(s)	Associated Cost
NOT Gate	2
2-input NAND, NOR, XOR, XNOR	4
2-input AND, OR                  3-input NAND, NOR	6
3-input AND, OR	8
JK Flip Flop	12

Test your state machine by defining an input sequence in QuartusII that exercises all the possible cases for your state machine. The first 16 or so steps in the sequence shown above should test every case, so that is a good sequence to use, or try other sequences of your own choosing.

For your report, show your circuit, the state diagram you used for your state machine, the steps you took to generate your design (tables and Karnaugh maps) and the pattern you observed on the output to record your circuit's operation. Be sure to have your lab instructor observe your operating circuit and answer the questions below.

Q#1: Is this a Moore or Mealy FSM? What part of the problem description defines this?

Q#2: What event(s) can cause a change on the output?

Q#3: For a given arc in your state diagram, show your lab instructor where this corresponds to in your simulation file.