**Laboratory 5 Finite State Machines**

**Prelab**

1. complete all Verilog files
2. compile your designs
3. simulations(optional)

**Objectives**

1. Design basic and timed FSMs
2. Understand the behavior of FSMs

**Part1 a basic Moore FSM**

**Circuit description**

We will design a game on a 7-segment display (SSD). The diagram is shown in Fig.1.

HEX0[0:6]

CLOCK\_50

clk\_1

fsm1

clock

KEY[0]

Fig.1 The diagram of a SSD game.

The first module clock generates a 1 Hz clock signal, *clk\_1*, from a 50MHz clock signal, *CLOCK\_50*. With clk\_1 as clock input and KEY[0] as reset, module fsm1 delivers HEX0[0:6] to drive a 7-segment display on DE2 board.

The LEDs in the 7-seg display are turned on according to the following sequence:

a 🡪ab🡪b🡪bc🡪c🡪cd🡪d🡪de🡪e🡪ef🡪f🡪fa🡪a🡪( repeat…). It looks like the light moves around in a clockwise direction. Sometimes a single LED is turned on, and sometimes two LEDs are turned on. In either case, LED(s) is on for 1 second.

b /HEX0=7’b1011111

ab /HEX0=7’b0011111

a /HEX0=7’b0111111

bc /HEX0=7’b1001111

fa/HEX0=7’b0111101

c /HEX0=7’b1101111

f /HEX0=7’b1111101

ef /HEX0=7’b1111001

cd /HEX0=7’b1100111

d /HEX0=7’b1110111

e /HEX0=7’b1111011

de /HEX0=7’b1110011

Fig.2 State diagram for fsm1

**[prelab: complete the following Verilog files]**

1. Top-level design

module lab5\_part1(CLOCK\_50, KEY, HEX0);

// declare input and output, reg, parameter ..

// instantiate module clock to generate clk\_1

// combinational logic for next state and output

// sequential logic of fsm1

endmodule

**Part2 a timed Moore FSM**

**Circuit description**

The behavior is similar to Part 1, except that single LED-on states remain for 1 second and double LED-on states remain for 0.5 second. Therefore module clock is used to generate 10Hz signal, and single LED-on states last for 10 clock cycle, and double LED-on states last only for 5 clock cycle.

CLOCK\_50

clk\_10

fsm2

clock

HEX0[0:6]

KEY[0]

Fig.3 Diagram of Part 2.

[Prelab]

1. Draw state transition diagram with time parameter for each state transition.
2. Verilog code

module lab5\_part2(CLOCK\_50, KEY, HEX0);

// declare input and output, reg, parameter ..

// instantiate module clock to generate clk\_10

// always block for sequential part of fsm2

// always block for next state logic and output

endmodule