Birla Institute of Technology and Science Pilani
CS F342 Computer Architecture
Second Semester 2017-18
Lab Sheet– 4 (06th February 2018)

Finite State Machine (FSM) Modeling

In today's lab we shall implement a Moore FSM (behavioral modeling) and verify their functionality using a suitable test bench.

Introduction to FSM modeling in HDL

- The HDL description of any FSM is accomplished using a standard template.
- The generalized way is to use three different procedural blocks one each for "next state calculation", "state update" and "output" (one can club next state calculation and output in case of Moore FSM).
- The next state calculation and output are pure combinational circuit. The state updated is sequential in nature and done on the suitable clock edge.
- Since named states provide a flexibility while writing the codes they can be declared as parameters. As, the scope of the states is limited to the given module alone we use the keyword "***localparam***"
- It is convenient to define to registers with the name next_state and current_state to hold the values of present state and the next state of the FSM. The length of these registers in a n-state FSM is typically $\log_2(n)$ rounded to nearest integer.

Example:

Develop a Moore type FSM to detect the occurrences of a sequence (non-overlapping) "111" in a stream of input bits.

Procedure:

- Draw the Moore FSM state machine.
- Draw the black box model of the same and identify the inputs and outputs.
- Identify the no. of states (n) and give them a suitable name.
- Identify the minimum no. of bits needed to encode the states.
- Develop the Verilog model. Use the template given below.
- Develop a proper test bench for the model.
- Simulate and cross check the output obtained

****** The End ******