Department of Electrical Engineering

UET, Lahore

Digital Systems Laboratory

Lab # 12(b)

Circuit-Based and State Diagram-Based HDL Models

Objective:

In this lab you will implement the Circuit-Based and State Diagram-Based HDL models of clocked sequential circuits.

Theoretical Background:

Read the article 5.6 of the book: **"Digital Design by M. Morris Mano and Michael D. Ciletti 5th Edition"**

An HDL model of the operation of a sequential circuit can be based on the format of the circuit's state diagram. The input, output, clock, and reset are declared in the usual manner. The state of the flip-flops is declared with identifiers state and next_state. These variables hold the values of the present state and the next value of the sequential circuit. The state's binary assignment is done with a parameter statement. (Verilog allows constants to be defined in a module by the keyword parameter).

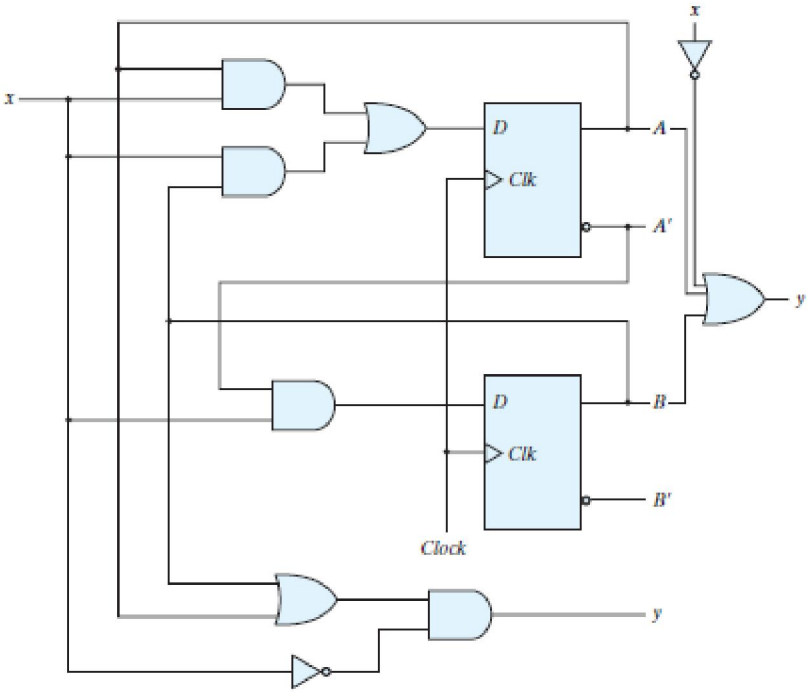

Figure 1: Mealy Zero Detector

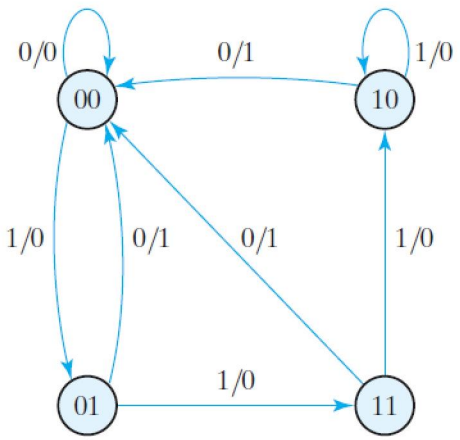

Figure 2: State Diagram of Mealy Zero Detector shown in Figure 1

Lab Task (a):

Simulate behavioral model of the Circuit-Based Verilog HDL Module for the circuit shown in Figure 1 [Mealy Machine: Zero Detector] using HDL modules from the lab manual 11(b).

Lab Task (b):

The state diagram of Mealy zero detector (Figure 1) is shown in Figure 2. Simulate behavioral model of the following State Diagram-Based Verilog HDL module for the circuit of Figure 1.

```
module Mealy_Zero_Detector(output reg y_out,input x_in, clock, reset);

reg [1: 0] state, next_state;
parameter S0 = 2'b00, S1 = 2'b01, S2 = 2'b10, S3 = 2'b11;

always @ ( posedge clock, negedge reset)
   if (reset == 0) state <= S0;
   else state <= next_state;

always @ (state, x_in) // Form the next state
   case (state)
      S0: if (x_in) next_state = S1; else next_state = S0;
      S1: if (x_in) next_state = S3; else next_state = S0;
      S2: if (~x_in) next_state = S0; else next_state = S2;
      S3: if (x_in) next_state = S2; else next_state = S0;
   endcase

always @ (state, x_in) // Form the Mealy output
   case (state)
      S0: y_out = 0;
      S1, S2, S3: y_out = ~x_in;
   endcase

endmodule
```

Note: The notation S2 = 2'b10 is preferable to the alternative S2 = 2. The former uses only two bits to store the constant, whereas the latter results in a binary number with 32 (or 64) bits because an unsized number is interpreted and sized as an integer.

Lab Task (c):

Simulate behavioral model of the Circuit-Based Verilog HDL module for the design problem of lab task (b) of manual 12(a) using HDL modules from lab manual 11(b).

Simulate behavioral model of the State Diagram-Based Verilog HDL module for the design problem of lab task (b) of manual 12(a).

Lab Task (d):

Simulate behavioral model of the Circuit-Based Verilog HDL module for the design problem in the assignment of manual 12(a) using HDL modules from lab manual 11(b).

Simulate behavioral model of the State Diagram-Based Verilog HDL module for the design problem in the assignment of manual 12(a).