**Edge Detector code**

module edge\_detector(

input wire clk,

input wire reset,

input wire a\_i,

output wire rising\_edge\_o,

output wire falling\_edge\_o

);

logic a\_ff;

always @(posedge clk or posedge reset)

if(reset)

a\_ff <= 1'b0;

else

a\_ff <= a\_i;

assign rising\_edge\_o = ~a\_ff & a\_i;

assign falling\_edge\_o = a\_ff & ~a\_i;

endmodule

**Test Bench Code**

module tb\_edge ();

logic clk;

logic reset;

logic a\_i;

logic rising\_edge\_o;

logic falling\_edge\_o;

edge\_detector day3 (.\*);

// clk

always begin

clk = 1'b1;

#5;

clk = 1'b0;

#5;

end

// Stimulus

initial begin

reset <= 1'b1;

a\_i <= 1'b1;

@(posedge clk);

reset <= 1'b0;

@(posedge clk);

for (int i=0; i<32; i++) begin

a\_i <= $random%2;

@(posedge clk);

end

$finish();

end

initial begin

$dumpfile("edge\_detector.vcd"); //specifies the filename ("mux.vcd") for the VCD file where simulation waveform data will be stored.

$dumpvars(0, tb\_edge);

end

endmodule

