**Sequence detector code**

module seq\_detector(

input wire clk,

input wire reset,

input wire x\_i,

output wire det\_o

);

logic [11:0] shift\_ff;

logic [11:0] nxt\_shift;

always\_ff @(posedge clk or posedge reset)

if(reset)

shift\_ff <= 12'h0;

else

shift\_ff <= nxt\_shift;

assign nxt\_shift = {x\_i,shift\_ff[11:1]};

assign det\_o = (shift\_ff[11:0] == 12'b1110\_1101\_1011);

endmodule

**Test Bench Code**

module tb\_seq\_detector();

logic clk,reset,x\_i,det\_o;

seq\_detector Day12(.\*);

always begin

clk = 1'b0;

#5;

clk = 1'b1;

#5;

end

logic [11:0]seq = 12'b1110\_1101\_1011;

logic [2:0]i;

logic [11:0]j;

initial begin

reset = 1'b1;

@(posedge clk);

reset = 1'b0;

for ( i = 0; i<4; i++)begin

for( j=0; j<12; j++)begin

if(i==1 || i==2)begin

x\_i = seq[j];

end

else begin

x\_i = 1'b0;

end

@(posedge clk);

end

end

end

endmodule

**Simulation**

