**논리설계 과제2**

2017320122 김정규

1. synccounter.v – Top module

module counter\_top (

input clk,

input rst,

output [3:0] out0, out1, out2, out3

);

reg [3:0] cnt0, cnt1, cnt2, cnt3;

always @ (posedge clk, negedge rst)

begin

if (!rst)

cnt0 <= 4'b0;

else

begin

if (cnt0 == 4'd15)

cnt0 <= 4'b0;

else

cnt0 <= cnt0 + 1'b1;

end

end

always @ (posedge clk, negedge rst)

begin

if (!rst)

cnt1 <= 4'b0;

else

begin

if (cnt1 == 4'd15)

cnt1 <= 4'b0;

else

cnt1 <= cnt1 + 1'b1;

end

end

always @ (posedge clk, negedge rst)

begin

if (!rst)

cnt2 <= 4'b0;

else

begin

if (cnt2 == 4'd15)

cnt2 <= 4'b0;

else

cnt2 <= cnt2 + 1'b1;

end

end

always @ (posedge clk, negedge rst)

begin

if (!rst)

cnt3 <= 4'b0;

else

begin

if (cnt3 == 4'd15)

cnt3 <= 4'b0;

else

cnt3 <= cnt3 + 1'b1;

end

end

assign out0 = cnt0;

assign out1 = cnt1;

assign out2 = cnt2;

assign out3 = cnt3;

endmodule

1. synctb.v – Testbench

module counter\_tb;

reg clk , rst;

wire [3:0] out0, out1, out2, out3;

counter\_top uut (.clk(clk), .rst(rst), .out0(out0), .out1(out1), .out2(out2), .out3(out3));

always

#5 clk = ~clk;

initial

begin

clk = 1'b0;

rst = 1'b0;

repeat(2)

@(negedge clk);

rst = 1'b1;

repeat(20)

@(negedge clk);

$finish;

end

endmodule

1. Wave

