module phase\_1(

input clk,

input reset,

input enable,

input code\_in,

output reg phase\_1done

);

parameter s0=3'b000;

parameter s1=3'b001;

parameter s2=3'b010;

parameter s3=3'b100;

parameter s4=3'b101;

reg[2:0] current\_state,next\_state;

always @(posedge clk or negedge reset)

begin

if(!reset)

current\_state <= s0;

else if(enable)

current\_state<= next\_state;

end

always@(\*)begin

next\_state=current\_state;

phase\_1done=1'b0;

case(current\_state)

s0:

begin

if(code\_in)

next\_state = s1;

else

next\_state = s0;

end

s1:

begin

if(~code\_in)

next\_state = s2;

else

next\_state = s1;

end

s2:

begin

if(code\_in)

next\_state = s3;

else

next\_state = s1;

end

s3:

begin

if(code\_in)

next\_state = s4;

else

next\_state = s2;

end

s4:

begin

phase\_1done=1'b1;

end

default:

next\_state=s0;

endcase

end

endmodule