**Lab No.6.**

**1)**

module lab6\_1(T,Clock,Reset, Q);

input T,Clock, Reset; output Q;

reg Q;

always @ (negedge Clock)

begin

if (T == 1)

Q <= ~Q;

if (Reset == 0)

Q <= 0;

end

endmodule

**2)**

module lab6\_2(J, K, Clock, Reset, Q);

input J, K, Clock, Reset;

output Q;

reg Q;

always @ (posedge Clock or posedge Reset) begin

case({J, K}) 0: Q <= Q;

1: Q <= 0;

2: Q <= 1; 3: Q <= ~Q;

endcase

if(Reset == 1)

Q <= 0;

end

endmodule

**3)** **a)**

module DFlipFlop(D, clock,reset, q);

input D, clock, reset; output q;

reg q;

always @ (posedge clock)

begin

if(reset)

q <= 0;

else if(D)

q <= 1; else q <= 0;

end

endmodule

module j2bitc(clock, reset, q);

input clock, reset;

output [0:1]q;

wire [0:1]q;

DFlipFlop d1(~q[1], clock, reset, q[0]);

DFlipFlop d2(q[0], clock, reset, q[1]);

endmodule

module dec2to4(W,En,Y);

input[1:0]W; input En; output [0:3]Y; reg [0:3]Y; always@(W or En) begin

if(En==1)

case(W)

0: Y=4'b1000;

1: Y=4'b0100;

2:Y=4'b0010; 3:Y=4'b0001; endcase else

Y=4'b0000; end

endmodule

module lab6\_3\_a(Clock, Reset, Q); input Clock, Reset; output [3:0] Q;

wire [3:0] Q;

wire [1:0] temp;

j2bitc c1(Clock, Reset, temp);

dec2to4 d1(temp, 1'b1, Q);

endmodule

**b)**

module DFlipFlop(D, clock, reset, q);

input D, clock,reset; output q;

reg q;

always @ (posedge clock)

begin

if(reset)

q <= 0; else if(D)

q <= 1;

else

q <= 0;

end

endmodule

module lab6\_3\_b(clock, reset, q); input clock, reset;

output [0:4]q;

wire [0:4]q;

DFlipFlop d1(~q[4], clock, reset, q[0]);

DFlipFlop d2(q[0], clock, reset, q[1]);

DFlipFlop d3(q[1], clock, reset, q[2]);

DFlipFlop d4(q[2], clock, reset, q[3]);

DFlipFlop d5(q[3], clock, reset, q[4]);

endmodule