1. **ไฟรถยนต์** 
   1. **Logic Gate Diagram**

รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างขึ้นโดยมีความน่าเชื่อถือสูงมาก

รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างขึ้นโดยมีความน่าเชื่อถือสูงมาก**b. Verilog Code**

**----- car.v -----**

module T\_FF(q, t, clk, reset);

output q;

input t,clk, reset;

reg q;

initial

begin

q=1'b0;

end

always @ (posedge clk or posedge reset)

if(reset)

q <= 0;

else

begin

q <= (t == 1) ? ~q : q;

end

endmodule

module car\_light(q, t, clk, reset);

output [2:0] q;

input t, clk, reset;

wire q0, q1;

T\_FF t\_ff1(q0, t, clk, reset);

T\_FF t\_ff2(q1, q0, clk, reset);

assign q[0] = q0 | q1 ;

assign q[1] = q1;

assign q[2] = q0 & q1;

endmodule

module car(light\_left, light\_right, left , right, E, clk);

output [2:0] light\_left, light\_right;

input left, right, E, clk;

wire or\_clk\_out\_1, re1, clk\_left;

wire or\_clk\_out\_2, re2, clk\_right;

nor or\_re\_1(re1, left, E);

or or\_clk\_1(or\_clk\_out\_1, left, E);

and and\_clk\_1(clk\_left, or\_clk\_out\_1, clk);

car\_light car\_light\_left(light\_left, 1'b1, clk\_left, re1);

nor or\_re\_2(re2, right, E);

or or\_clk\_2(or\_clk\_out\_2, right, E);

and and\_clk\_2(clk\_right, or\_clk\_out\_2, clk);

car\_light car\_light\_right(light\_right, 1'b1, clk\_right, re2);

endmodule

**----- car\_stimulus.v -----**

module car\_light\_stimulus;

reg left, right, E, clk;

reg din\_0, din\_1;

wire [2:0] light\_left, light\_right;

car\_light car1(light\_left, light\_right, left , right, E, clk);

initial

begin

$dumpfile("TimeDiagram.vcd");

$dumpvars(0,car\_light\_stimulus);

left = 1'b0;

right = 1'b0;

E = 1'b0;

clk = 1'b0;

end

always

#5 clk = !clk;

initial

begin

#10 left = !left;

#40 left = !left;

right = !right;

#40 left = !left;

#40 left = !left;

right = !right;

#10 E = !E;

end

initial

begin

#250 $finish;

end

initial

$monitor ($time, clk, light\_left, light\_right, E);

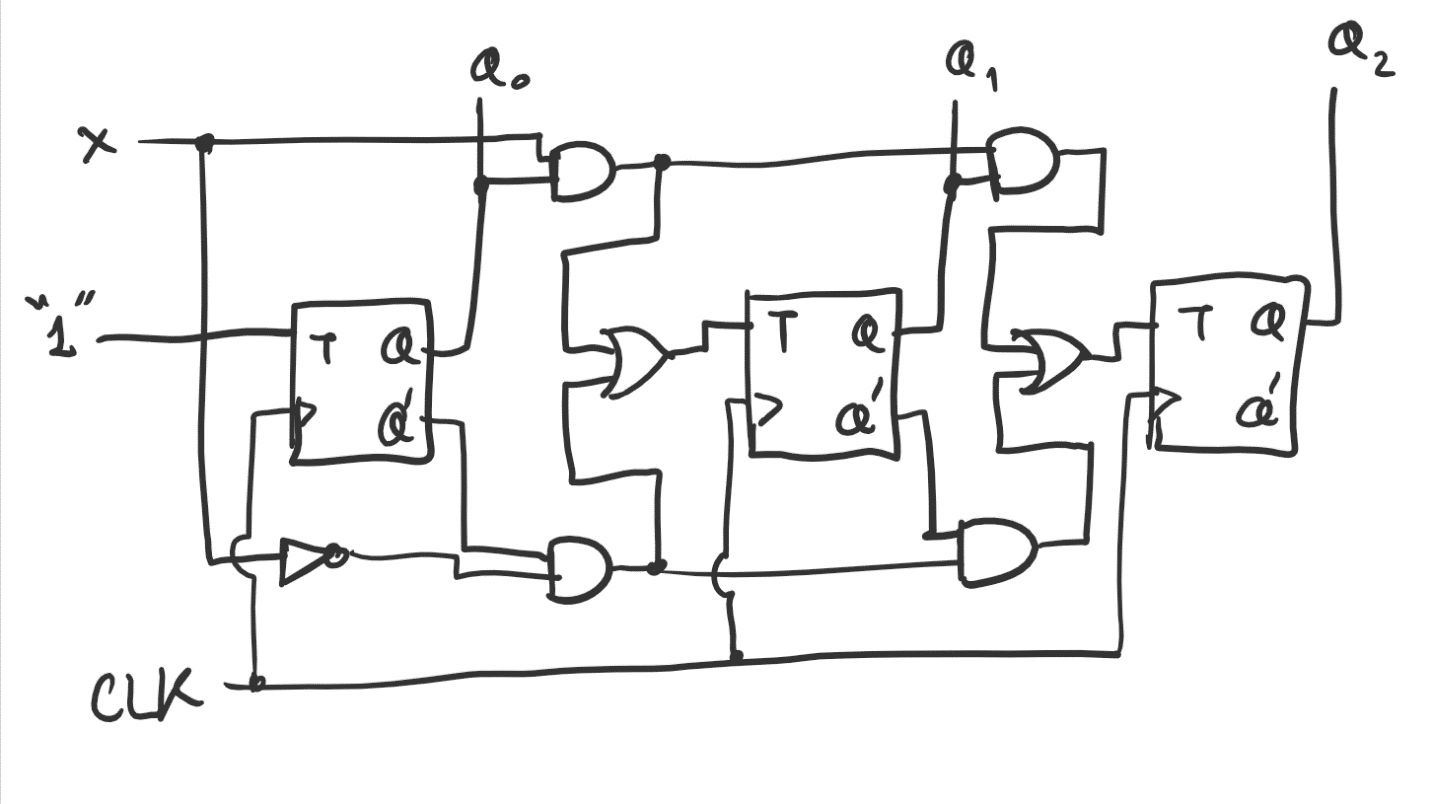
endmodule

**รูปภาพประกอบด้วย ภาพหน้าจอ, จอภาพ, คอมพิวเตอร์

คำอธิบายที่สร้างขึ้นโดยมีความน่าเชื่อถือสูงมากC. Time diagram**

1. **Count Up and Down**

**Logic Gate Diagram**

****

**Verilog Code**

module T\_FF(q, t, clk, reset);

output q;

input t,clk, reset;

reg q;

initial

begin

q=1'b0;

end

always @ (posedge clk)

if(reset)

q <= 0;

else

begin

q <= (t == 1) ? ~q : q;

end

endmodule

module count\_up\_down(q, x, clk, reset);

output [2:0] q;

input x, clk, reset;

wire and\_out1,and\_out2,or\_out1;

wire and2\_out1,and2\_out2,or\_out2;

wire q1, q2, q3;

T\_FF t\_ff1(q1, 1'b1, clk, reset);

and and1\_1(and\_out1, !x, !q1);

and and1\_2(and\_out2, x, q1);

or or1\_1(or\_out1, and\_out1, and\_out2);

T\_FF t\_ff2(q2, or\_out1, clk, reset);

and and2\_1(and2\_out1, and\_out1, !q2);

and and2\_2(and2\_out2, and\_out2, q2);

or or2\_1(or\_out2, and2\_out1, and2\_out2);

T\_FF t\_ff3(q3, or\_out2, clk, reset);

assign q[0] = q1;

assign q[1] = q2;

assign q[2] = q3;

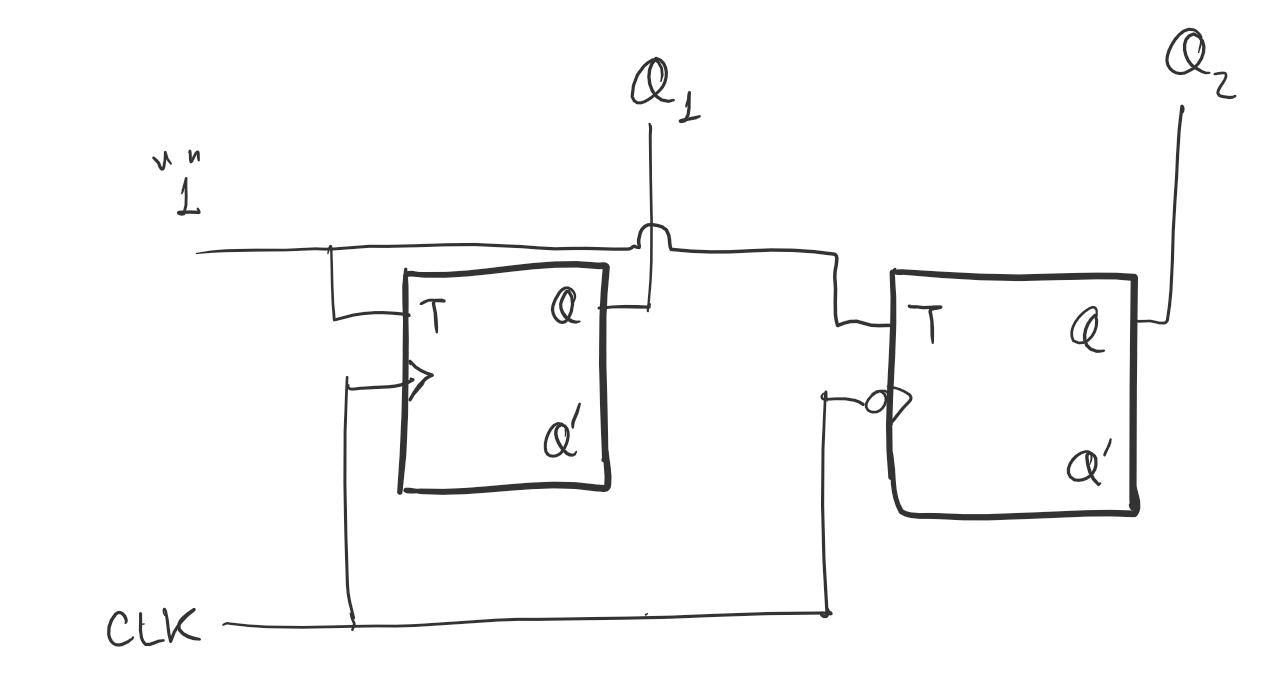
endmodule

**Time Diagram**

**รูปภาพประกอบด้วย ภาพหน้าจอ

คำอธิบายที่สร้างขึ้นโดยมีความน่าเชื่อถือสูงมาก**

1. **Gray Code**

**Logic Gate Diagram**