```
module TrafficController(
    input logic clk,
    input logic reset,
    input logic Sa,
    input logic Sb,
    output logic xLa,
    output logic yLa,
    output logic zLa,
    output logic xLb,
    output logic yLb,
    output logic zLb);

    typedef enum logic[2:0] {S0,S1,S2,S3,S4,S5,S6,S7} statetype;
    statetype state, nextstate;

    //Timer t1(clk, reset, new_clk);
    //state register
    always_ff @(posedge clk, posedge reset)
        if(reset) state <= S0;
        else      state <= nextstate;
    //nextstate comb logic
    always_comb
        case(state)
            S0: if(Sb) nextstate=S1;
                else   nextstate=S0;
            S1: nextstate=S2;
            S2: nextstate=S3;
            S3: nextstate=S4;
            S4: if(!Sa&Sb)  nextstate=S4;
                else        nextstate=S5;
            S5: nextstate=S6;
            S6: nextstate=S7;
            S7: nextstate=S0;
            default: nextstate=S0;
        endcase
    //output logic
    assign zLa = 1;
    assign zLb = 1;

    assign yLa =(!state[0] | !state[2] & state[1] | state[2] & !state[1]);
    assign yLb =(!state[0] | !state[2] & !state[1] | state[2] & state[1]);

    assign xLa =(!state[2] & state[1] | state[2] & !state[1] | state[1] & !state[0]);
    assign xLb =(!state[2] & !state[1] | state[2] & state[1] | state[1] & !state[0]);
endmodule
```

```
module Timer(
    input logic clk,
    input logic reset,
    output logic new_clk
    );
    logic [32:0]count;
    always@(posedge clk, posedge reset)
        begin
            if(reset == 1)
                count <= 32'b0;
            else if(count == 150000000 - 1)
                count <= 32'b0;
            else
                count <= count + 1;
    end

    always @(posedge(clk), posedge(reset))
        begin
            if (reset == 1'b1)
                new_clk <= 1'b0;
            else if (count == 300000000 - 1)
                new_clk <= ~new_clk;
            else
                new_clk <= new_clk;
    end
endmodule

module Traffic_Sim();
    logic clk = 0;
    logic reset = 0;
    logic Sa = 0;
    logic Sb = 0;
    logic xLa = 0;
    logic yLa = 1;
    logic zLa = 1;
    logic xLb = 1;
    logic yLb = 1;
    logic zLb = 1;

    TrafficController dut(clk,reset,Sa,Sb,xLa,yLa,zLa,xLb,yLb,zLb);
    always begin
        clk = 0; #5; clk = 1; #5;
    end
    always begin
        Sa = 0; Sb = 0; #80;
        Sa = 0; Sb = 1; #70; reset = 1; #5; reset = 0; #5;
        Sa = 1; Sb = 0; #80;
        Sa = 1; Sb = 1; #80;
    end
endmodule
```