PROJECT

TRAFFIC LIGHT SIGNAL CONTROL

PROCEDURE:

Step1: Create a Vivado Project using IDE sourcing verilog HDL model and targeting a specific FPGA device located on the Basys3 (Xilinx Artix-7 FPGA: XC7A35T1CPG236C)

Step2: Simulate the design using Vivado Simulator (Test Bench Module).

Step3: Synthesize the design and observe the Schematic.

Step4: Implement the design.

Step5: Write Design Constraint (XDC) file to constrain the pin locations.

Step6: Generate the bitstream.

Step7: Configure the FPGA using the generated bitstream and verify the functionality in hardware.

DESIGN SOURCE:

```
module trafficlightsignal_3(
input logic clk,reset,
input logic x,
output logic [2:0]hwy,[2:0]cntry
);
typedef enum
logic[2:0]{S0=3'd0,S1=3'd1,S2=3'd2,S3=3'd3,S4=3'd4}statetype;
statetype [2:0] state,nextstate;

parameter[2:0]red=3'b001;
parameter [2:0]yellow=3'b010;
parameter [2:0]green=3'b100;

// state register
always_ff @(posedge clk, posedge reset)
if (reset) state <= S0;
else
state <= nextstate;

always_comb
case (state)
S0: if(x)nextstate=S1;
   else nextstate=S0;
```

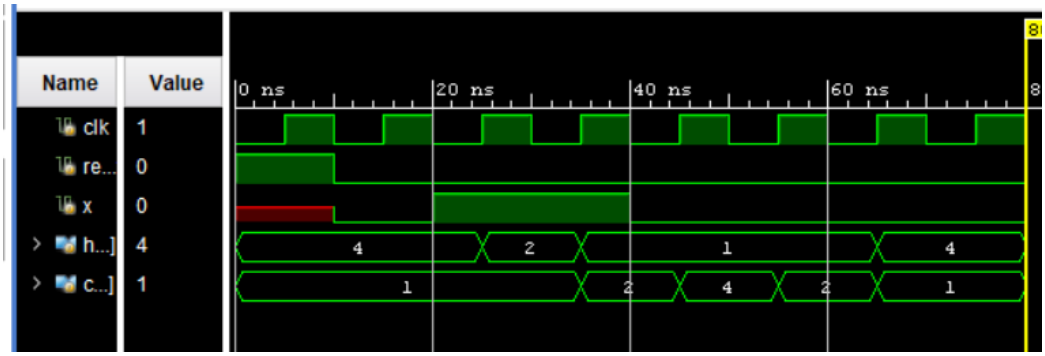
```
S1:nextstate = S2;
S2: nextstate = S3;
S3:if(x) nextstate=S3;
   else nextstate = S4;
S4:nextstate=S0;
default:nextstate = S0;
endcase

always_comb
case(state)
S0:{hwy,cntry}={green,red};
S1:{hwy,cntry}={yellow,red};
S2:{hwy,cntry}={red,yellow};
S3:{hwy,cntry}={red,green};
S4:{hwy,cntry}={red,yellow};
endcase
endmodule
```

SIMULATION SOURCE (TEST BENCH):

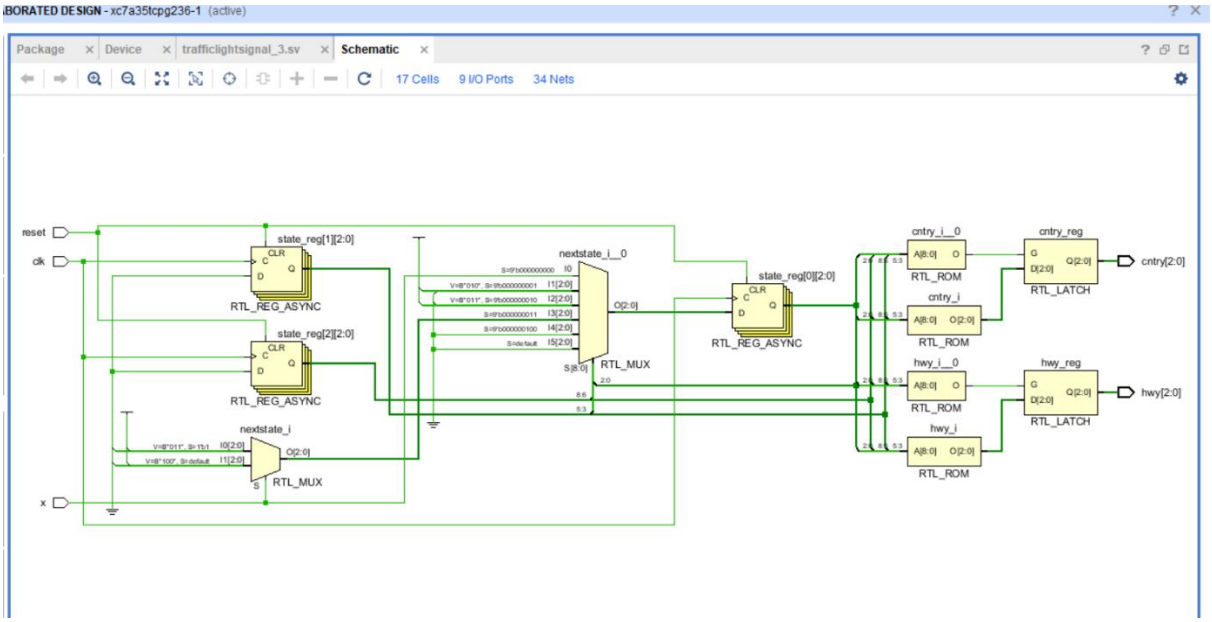
```
module trafficlightsignal_3tb();
logic clk,reset,x;
logic [2:0]hwy;
logic [2:0]cntry;
```

```
trafficlightsignal_3 dut(clk,reset,x,hwy,cntry);
initial begin
clk=0;
forever #5 clk=~clk;
end
initial begin
reset=1;
#10
reset=0;
x=0 ;
#10
 x=1;
#20
x=0;
#10
#30 $finish;
end
```

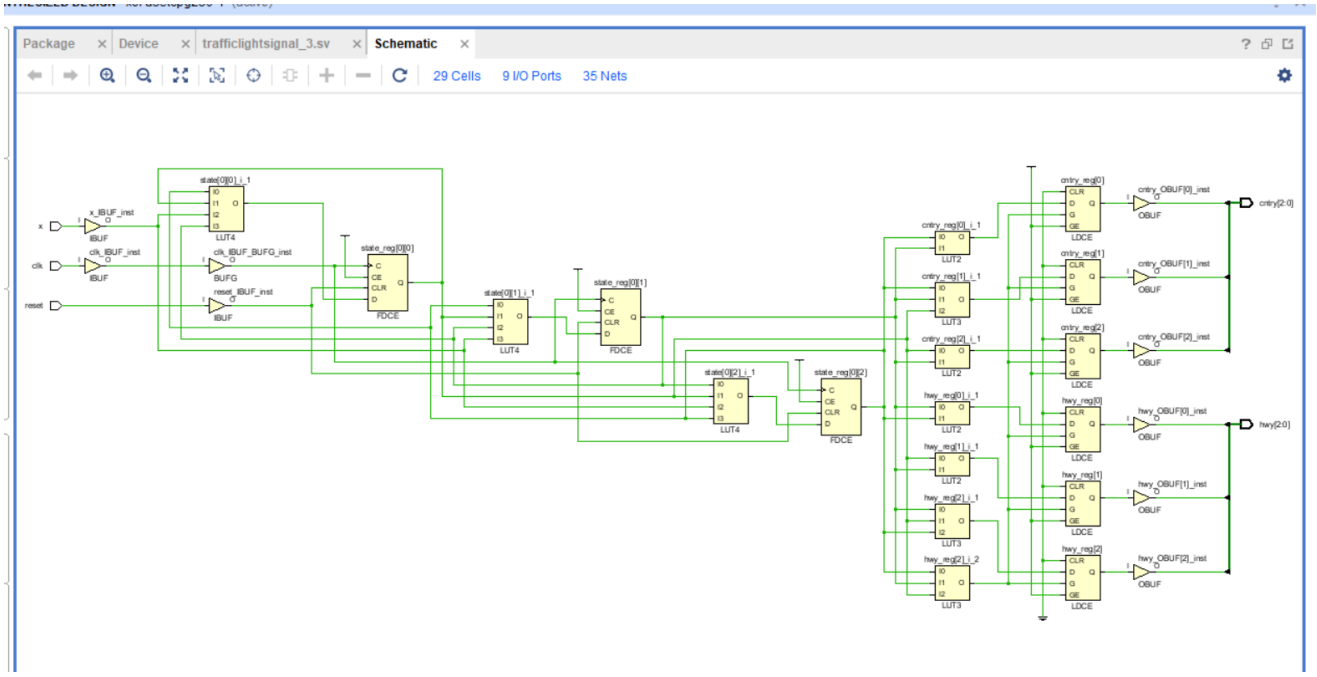
OUTPUT WAVEFORMS:



RTL SCHEMATIC:



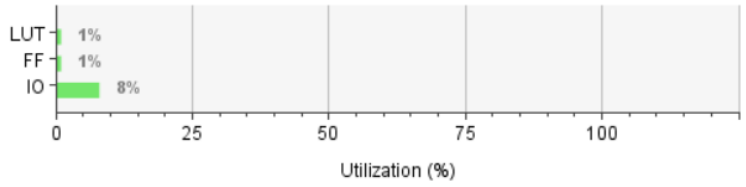
IMPLEMENTATION SCHEMATIC:



REPORT UTILISATION:

Summary

Resource	Utilization	Available	Utilization %
LUT	5	20800	0.02
FF	9	41600	0.02
IO	9	106	8.49



Primitives

Ref Name	Used	Functional Category
OBUF	6	IO
LDCE	6	Flop & Latch
LUT2	4	LUT
LUT4	3	LUT
LUT3	3	LUT
IBUF	3	IO
FDCE	3	Flop & Latch
BUFG	1	Clock

OUTPUT: