

🡪FPGA Synthesizable code:-

module traffic\_light\_control(hwy,cntry,segment\_1, segment\_2,segment\_3,com\_anode\_1,com\_anode\_2,dp\_1,x,clock,clr);

input clock,clr,x;

output reg [1:0]hwy,cntry;

output wire [6:0]segment\_1;

output wire [2:0]segment\_2,segment\_3;

output reg [3:0]com\_anode\_1;

output reg [3:0]com\_anode\_2;

output wire dp\_1;

assign dp\_1=1;

wire led;

wire led1;

reg[2:0]display;

reg[2:0]enable=0;

parameter RED=2'b00,YELLOW=2'b01,GREEN=2'b10;

parameter sen\_on=1'b1,sen\_off=1'b0;//Sensor will be ON when vehicles are detected in Country Road unless it

parameter Y2Rdelay=4'b1000,R2Gdelay=4'b1000;

//defining states HighWay Countryroad:

parameter s0=3'b000, // GREEN RED

s1=3'b001, // YELLOW RED

s2=3'b010, // RED RED

s3=3'b011, // RED GREEN

s4=3'b100; // RED YELLOW

reg [2:0]y,Y; //y=present-state Y=Next\_state

userclock c1(clock,led);

userclock1 c2(clock,led1);

seven\_segment ss1(display,segment\_1);

//Displaying in Seven Segment

//Triggering of Output

always @ (posedge led or negedge clr)

begin

if (!clr)

y<=s0;

else

y<=Y;

end

//Redefining the States.

always @(y)

begin

case(y)

s0: begin

hwy=GREEN;

cntry=RED;

end

s1: begin

hwy=YELLOW;

cntry=RED;

end

s2:begin

hwy=RED;

cntry=RED;

end

s3:begin

hwy=RED;

cntry=GREEN;

end

s4:begin

hwy=RED;

cntry=YELLOW;

end

endcase

end

//State\_Machine programming

always @(y or x)

begin

case(y)

s0: if (x) //if sensor sends sen\_on,then Highway: Yellow,CountryRoad: Red

Y<=s1;

else

Y<=s0; //if sensor sends sen\_off,then HighWay: Green,countryroad: Red

s1: begin

repeat(Y2Rdelay) //for 8 posedge of led Highway:Yellow,countryRoad:Red

Y<=s1;

Y<=s2; //After 8 posedge of led,Highway:Red,CountryRoad:Red

end

s2: begin

repeat(R2Gdelay) //for 8 posedge of led Highway:Red,CountryRoad: Red

Y<=s2;

if (x)

Y<=s3; //if sensor is on,Highway:Red,Countryroad:Green

else

Y<=s0; //if sensor is off Highway:Green,CountryRoad=Red

end

s3: if (x)

Y<=s3; //until sen\_on comes,Highway:Red, CountryRoad:Green

else

Y<=s4; //at the instance sen\_off comes Highway:Red,countryRoad:Yellow

s4: begin

repeat(Y2Rdelay) //for 8 posedge of led,Highway:Red,Countryroad: Yellow

Y<=s4;

Y<=s2; //Highway:Red,Countryroad:Red

end

default:

Y<=s0; //Default Case: Highway:Green,countryRoad:Red

endcase

end

//Displaying Lights in FPGA Board

led\_display\_hwy fpga3(hwy,segment\_3);

led\_display\_cntry fpga2(cntry,segment\_2);

always @ (posedge led1)

begin

enable<=enable+1;

end

always @ (posedge led1)

begin

if (hwy==0&&cntry==0) //when Highway:RED;"STOP" is showing.

begin

case(enable)

0:begin

display<=0;

com\_anode\_1=4'b0111;

com\_anode\_2=4'b1111;

end

1:begin

display<=1;

com\_anode\_1=4'b1011;

com\_anode\_2=4'b1111;

end

2:begin

display<=2;

com\_anode\_1=4'b1101;

com\_anode\_2=4'b1111;

end

3:begin

display<=3;

com\_anode\_1=4'b1110;

com\_anode\_2=4'b1111;

end

4:begin

display<=0;

com\_anode\_2=4'b0111;

com\_anode\_1=4'b1111;

end

5:begin

display=1;

com\_anode\_2=4'b1011;

com\_anode\_1=4'b1111;

end

6:begin

display=2;

com\_anode\_2=4'b1101;

com\_anode\_1=4'b1111;

end

7:begin

display=3;

com\_anode\_2=4'b1110;

com\_anode\_1=4'b1111;

end

endcase

end

else if(cntry==1 && hwy==0)

begin

case(enable)

0:begin

display=0;

com\_anode\_1=4'b0111;

com\_anode\_2=4'b1111;

end

1:begin

display=1;

com\_anode\_1=4'b1011;

com\_anode\_2=4'b1111;

end

2:begin

display=2;

com\_anode\_1=4'b1101;

com\_anode\_2=4'b1111;

end

3:begin

display=3;

com\_anode\_1=4'b1110;

com\_anode\_2=4'b1111;

end

4:begin

display<=0;

com\_anode\_2=4'b0111;

com\_anode\_1=4'b1111;

end

5:begin

display=4;

com\_anode\_2=4'b1011;

com\_anode\_1=4'b1111;

end

6:begin

display=2;

com\_anode\_2=4'b1101;

com\_anode\_1=4'b1111;

end

7:begin

display=5;

com\_anode\_2=4'b1110;

com\_anode\_1=4'b1111;

end

endcase

end

else if(cntry==2&&hwy==0)

begin

case(enable)

0:begin

display=0;

com\_anode\_1=4'b0111;

com\_anode\_2=4'b1111;

end

1:begin

display<=1;

com\_anode\_1=4'b1011;

com\_anode\_2=4'b1111;

end

2:begin

display<=2;

com\_anode\_1=4'b1101;

com\_anode\_2=4'b1111;

end

3:begin

display=3;

com\_anode\_1=4'b1110;

com\_anode\_2=4'b1111;

end

4:begin

display<=6;

com\_anode\_2=4'b1011;

com\_anode\_1=4'b1111;

end

5:begin

display<=2;

com\_anode\_2=4'b1101;

com\_anode\_1=4'b1111;

end

endcase

end

else if (hwy==1&&cntry==0) //Highway:Yellow;"SLOW" is showing.

begin

case(enable)

0:begin

display=0;

com\_anode\_1=4'b0111;

com\_anode\_2=4'b1111;

end

1:begin

display=4;

com\_anode\_1=4'b1011;

com\_anode\_2=4'b1111;

end

2:begin

display=2;

com\_anode\_1=4'b1101;

com\_anode\_2=4'b1111;

end

3:begin

display=5;

com\_anode\_1=4'b1110;

com\_anode\_2=4'b1111;

end

4:begin

display=0;

com\_anode\_2=4'b0111;

com\_anode\_1=4'b1111;

end

5:begin

display=1;

com\_anode\_2=4'b1011;

com\_anode\_1=4'b1111;

end

6:begin

display=2;

com\_anode\_2=4'b1101;

com\_anode\_1=4'b1111;

end

7:begin

display=3;

com\_anode\_2=4'b1110;

com\_anode\_1=4'b1111;

end

endcase

end

else if (hwy==2&&cntry==0) //Highway:GREEN;"GO" is showing.

begin

case(enable)

0:begin

display=6;

com\_anode\_1=4'b1011;

com\_anode\_2=4'b1111;

end

1:begin

display=2;

com\_anode\_1=4'b1101;

com\_anode\_2=4'b1111;

end

2:begin

display<=0;

com\_anode\_2=4'b0111;

com\_anode\_1=4'b1111;

end

3:begin

display=1;

com\_anode\_2=4'b1011;

com\_anode\_1=4'b1111;

end

4:begin

display=2;

com\_anode\_2=4'b1101;

com\_anode\_1=4'b1111;

end

5:begin

display=3;

com\_anode\_2=4'b1110;

com\_anode\_1=4'b1111;

end

endcase

end

end

endmodule

//Defining output of Seven\_segment\_display

module seven\_segment(input [2:0]display,output reg[6:0]segment\_1);

always @ (display)

begin

case(display)

0:segment\_1=7'b0100100; //"S"

1:segment\_1=7'b1110000; //"T"

2:segment\_1=7'b0000001; //"O"

3:segment\_1=7'b0011000; //"P"

4:segment\_1=7'b1110001; //"L"

5:segment\_1=7'b1000000; //"W"

6:segment\_1=7'b0000100; //"G"

endcase

end

endmodule

module led\_display\_cntry(input wire[1:0]cntry,output reg[2:0]segment\_2);

always@(\*)

begin

case(cntry)

0: segment\_2=3'b100;

1: segment\_2=3'b001;

2: segment\_2=3'b010;

endcase

end

endmodule

module led\_display\_hwy(input wire[1:0]hwy,output reg[2:0]segment\_3);

always@(\*)

begin

case(hwy)

0: segment\_3=3'b100;

1: segment\_3=3'b001;

2: segment\_3=3'b010;

endcase

end

endmodule

module userclock(input clock,output led);

reg clk\_out=0;

reg [29:0]count=0;

always @(posedge clock)

begin

if (count==500000000)

begin

count<=0;

clk\_out<=~clk\_out;

end

else

begin

count<=count+1;

end

end

assign led=clk\_out;

endmodule

module userclock1(input clock,output led1);

reg clk\_out1=0;

reg [29:0]count1=0;

always @(posedge clock)

begin

if (count1==100000)

begin

count1<=0;

clk\_out1<=~clk\_out1;

end

else

begin

count1<=count1+1;

end

end

assign led1=clk\_out1;

endmodule

🡪Verilog code for traffic light (for which test bench was written):-

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 18.04.2023 14:52:51

// Design Name:

// Module Name: tlc

// Project Name:

// Target Devices:

// Tool Versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

// Name:-Akul Verma

//Roll No.:-21PHC1R04

//Traffic Light Control

module tlc(

input clk,rst,x,

output reg [1:0] hwy,cntry

);

//states

parameter count=2'b10;

parameter S0= 4'b000; //hwy= G , cntry=R

parameter S1= 4'b001; //hwy= G , cntry=R

parameter S2= 4'b010; //hwy= G , cntry=R

parameter S3= 4'b011; //hwy= G , cntry=R

parameter S4= 4'b100; //hwy= G , cntry=R

//output parameters

parameter R=2'b00; //Red

parameter Y=2'b01; //Yellow

parameter G=2'b10; //Green

//Internal states

reg [2:0] c\_state; //current state

reg [2:0] n\_state; //next state

initial

begin

n\_state=S0;

c\_state=S0;

hwy=G;

cntry=R;

end

always @(posedge clk)

begin

if(rst==1)

begin

c\_state=S0;

n\_state=S0;

hwy=G;

cntry=R;

end

else

begin

c\_state=n\_state;

case(c\_state)

S0:

if(x==1)

begin

n\_state=S1;

hwy=Y;

cntry=R;

end

else

begin

n\_state=S0;

hwy=G;

cntry=R;

end

S1:

begin

repeat(count)

begin

@(posedge clk)

n\_state=S2;

hwy=R;

cntry=R;

end

end

S2: begin

repeat(count)

begin

@(posedge clk)

n\_state = S3;

hwy=R;

cntry=G;

end

end

S3:

if(x==1)

begin

n\_state=S3;

hwy=R;

cntry=G;

end

else

begin

n\_state=S4;

hwy=R;

cntry=Y;

end

S4:

begin

repeat(count)

begin

@(posedge clk)

n\_state=S0;

hwy=G;

cntry=R;

end

end

default:

begin

repeat(count)

begin

@(posedge clk)

n\_state=S0;

hwy=G;

cntry=R;

end

end

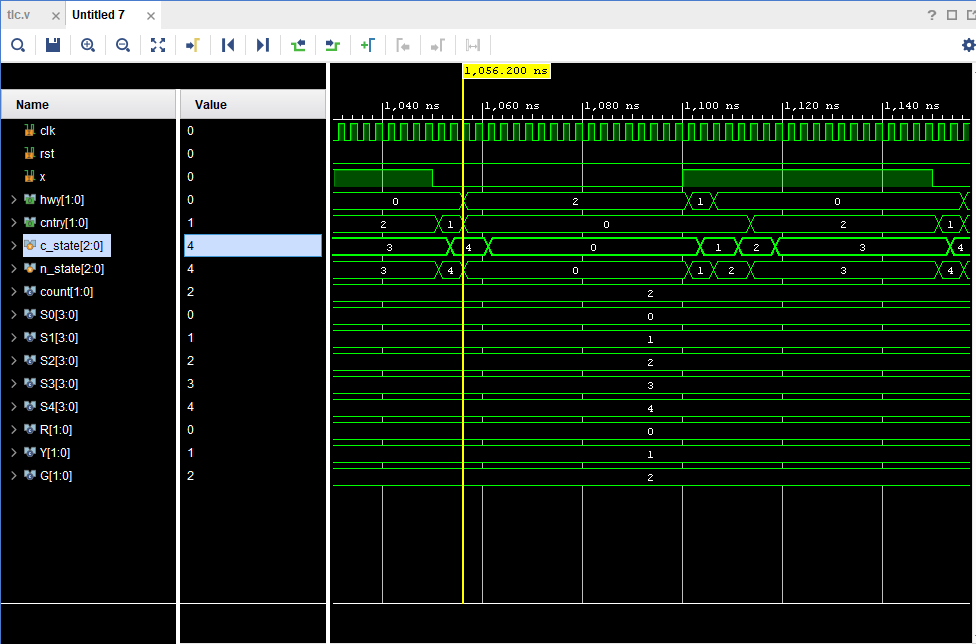
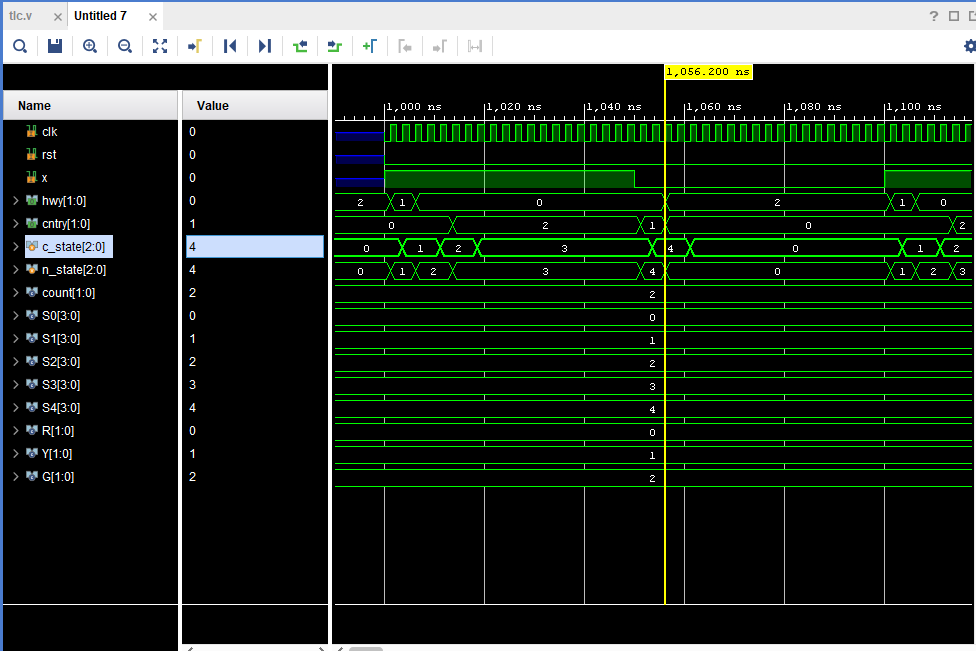
endcase

end

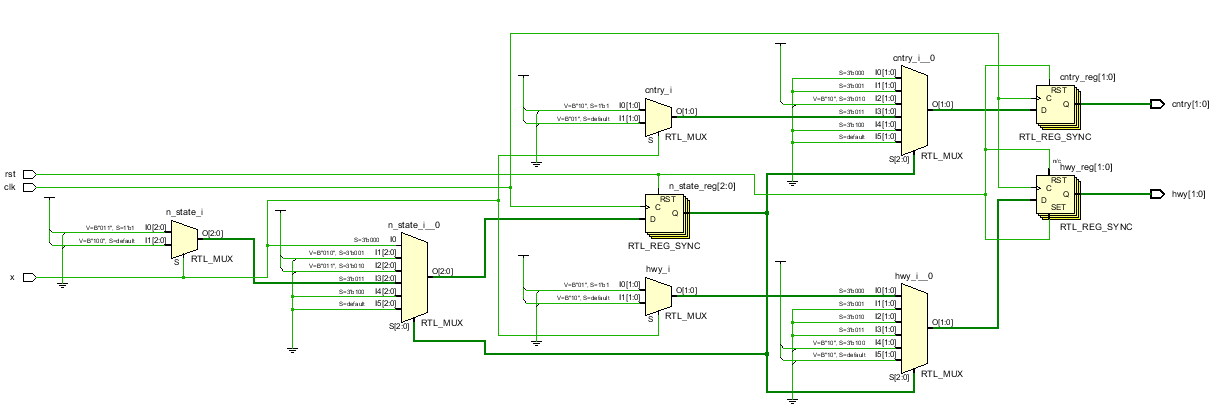
end

endmodule

🡪Waveform output:-



🡪Schematic:-



🡪Testbench:-

`timescale 1ns / 1ps

module tlc\_tb();

reg rst,clk,x;

wire [1:0] hwy ,cntry;

tlc uut(clk,rst,x,hwy,cntry);

initial

begin

rst=0;clk=0;x=1;

$monitor($time," clk=%b,rst=%b,x=%b,hwy=%b,cntry=%b",clk,rst,x,hwy,cntry);

#200

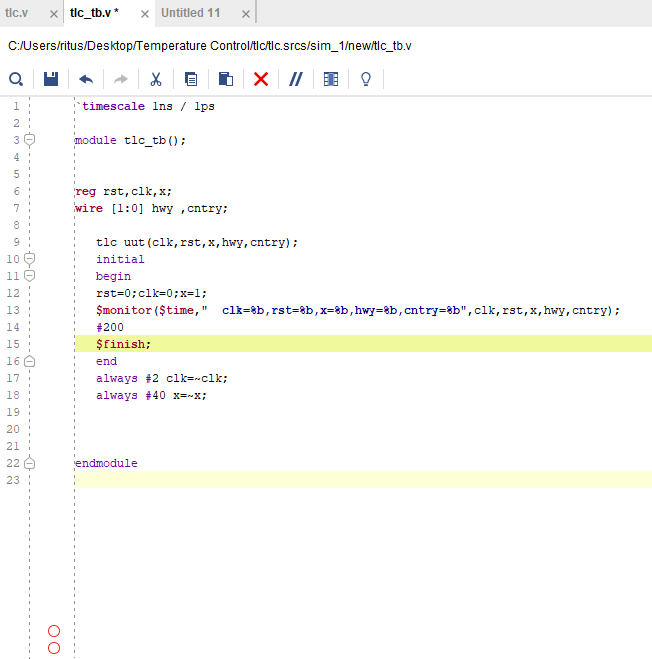
$finish;

end

always #2 clk=~clk;

always #40 x=~x;

endmodule



🡪Testbench Output:-

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generated

run 1000ns

0 clk=0,rst=0,x=1,hwy=10,cntry=00

2 clk=1,rst=0,x=1,hwy=01,cntry=00

4 clk=0,rst=0,x=1,hwy=01,cntry=00

6 clk=1,rst=0,x=1,hwy=00,cntry=00

8 clk=0,rst=0,x=1,hwy=00,cntry=00

10 clk=1,rst=0,x=1,hwy=00,cntry=10

12 clk=0,rst=0,x=1,hwy=00,cntry=10

14 clk=1,rst=0,x=1,hwy=00,cntry=10

16 clk=0,rst=0,x=1,hwy=00,cntry=10

18 clk=1,rst=0,x=1,hwy=00,cntry=10

20 clk=0,rst=0,x=1,hwy=00,cntry=10

22 clk=1,rst=0,x=1,hwy=00,cntry=10

24 clk=0,rst=0,x=1,hwy=00,cntry=10

26 clk=1,rst=0,x=1,hwy=00,cntry=10

28 clk=0,rst=0,x=1,hwy=00,cntry=10

30 clk=1,rst=0,x=1,hwy=00,cntry=10

32 clk=0,rst=0,x=1,hwy=00,cntry=10

34 clk=1,rst=0,x=1,hwy=00,cntry=10

36 clk=0,rst=0,x=1,hwy=00,cntry=10

38 clk=1,rst=0,x=1,hwy=00,cntry=10

40 clk=0,rst=0,x=0,hwy=00,cntry=10

42 clk=1,rst=0,x=0,hwy=00,cntry=01

44 clk=0,rst=0,x=0,hwy=00,cntry=01

46 clk=1,rst=0,x=0,hwy=10,cntry=00

48 clk=0,rst=0,x=0,hwy=10,cntry=00

50 clk=1,rst=0,x=0,hwy=10,cntry=00

52 clk=0,rst=0,x=0,hwy=10,cntry=00

54 clk=1,rst=0,x=0,hwy=10,cntry=00

56 clk=0,rst=0,x=0,hwy=10,cntry=00

58 clk=1,rst=0,x=0,hwy=10,cntry=00

60 clk=0,rst=0,x=0,hwy=10,cntry=00

62 clk=1,rst=0,x=0,hwy=10,cntry=00

64 clk=0,rst=0,x=0,hwy=10,cntry=00

66 clk=1,rst=0,x=0,hwy=10,cntry=00

68 clk=0,rst=0,x=0,hwy=10,cntry=00

70 clk=1,rst=0,x=0,hwy=10,cntry=00

72 clk=0,rst=0,x=0,hwy=10,cntry=00

74 clk=1,rst=0,x=0,hwy=10,cntry=00

76 clk=0,rst=0,x=0,hwy=10,cntry=00

78 clk=1,rst=0,x=0,hwy=10,cntry=00

80 clk=0,rst=0,x=1,hwy=10,cntry=00

82 clk=1,rst=0,x=1,hwy=01,cntry=00

84 clk=0,rst=0,x=1,hwy=01,cntry=00

86 clk=1,rst=0,x=1,hwy=00,cntry=00

88 clk=0,rst=0,x=1,hwy=00,cntry=00

90 clk=1,rst=0,x=1,hwy=00,cntry=10

92 clk=0,rst=0,x=1,hwy=00,cntry=10

94 clk=1,rst=0,x=1,hwy=00,cntry=10

96 clk=0,rst=0,x=1,hwy=00,cntry=10

98 clk=1,rst=0,x=1,hwy=00,cntry=10

100 clk=0,rst=0,x=1,hwy=00,cntry=10

102 clk=1,rst=0,x=1,hwy=00,cntry=10

104 clk=0,rst=0,x=1,hwy=00,cntry=10

106 clk=1,rst=0,x=1,hwy=00,cntry=10

108 clk=0,rst=0,x=1,hwy=00,cntry=10

110 clk=1,rst=0,x=1,hwy=00,cntry=10

112 clk=0,rst=0,x=1,hwy=00,cntry=10

114 clk=1,rst=0,x=1,hwy=00,cntry=10

116 clk=0,rst=0,x=1,hwy=00,cntry=10

118 clk=1,rst=0,x=1,hwy=00,cntry=10

120 clk=0,rst=0,x=0,hwy=00,cntry=10

122 clk=1,rst=0,x=0,hwy=00,cntry=01

124 clk=0,rst=0,x=0,hwy=00,cntry=01

126 clk=1,rst=0,x=0,hwy=10,cntry=00

128 clk=0,rst=0,x=0,hwy=10,cntry=00

130 clk=1,rst=0,x=0,hwy=10,cntry=00

132 clk=0,rst=0,x=0,hwy=10,cntry=00

134 clk=1,rst=0,x=0,hwy=10,cntry=00

136 clk=0,rst=0,x=0,hwy=10,cntry=00

138 clk=1,rst=0,x=0,hwy=10,cntry=00

140 clk=0,rst=0,x=0,hwy=10,cntry=00

142 clk=1,rst=0,x=0,hwy=10,cntry=00

144 clk=0,rst=0,x=0,hwy=10,cntry=00

146 clk=1,rst=0,x=0,hwy=10,cntry=00

148 clk=0,rst=0,x=0,hwy=10,cntry=00

150 clk=1,rst=0,x=0,hwy=10,cntry=00

152 clk=0,rst=0,x=0,hwy=10,cntry=00

154 clk=1,rst=0,x=0,hwy=10,cntry=00

156 clk=0,rst=0,x=0,hwy=10,cntry=00

158 clk=1,rst=0,x=0,hwy=10,cntry=00

160 clk=0,rst=0,x=1,hwy=10,cntry=00

162 clk=1,rst=0,x=1,hwy=01,cntry=00

164 clk=0,rst=0,x=1,hwy=01,cntry=00

166 clk=1,rst=0,x=1,hwy=00,cntry=00

168 clk=0,rst=0,x=1,hwy=00,cntry=00

170 clk=1,rst=0,x=1,hwy=00,cntry=10

172 clk=0,rst=0,x=1,hwy=00,cntry=10

174 clk=1,rst=0,x=1,hwy=00,cntry=10

176 clk=0,rst=0,x=1,hwy=00,cntry=10

178 clk=1,rst=0,x=1,hwy=00,cntry=10

180 clk=0,rst=0,x=1,hwy=00,cntry=10

182 clk=1,rst=0,x=1,hwy=00,cntry=10

184 clk=0,rst=0,x=1,hwy=00,cntry=10

186 clk=1,rst=0,x=1,hwy=00,cntry=10

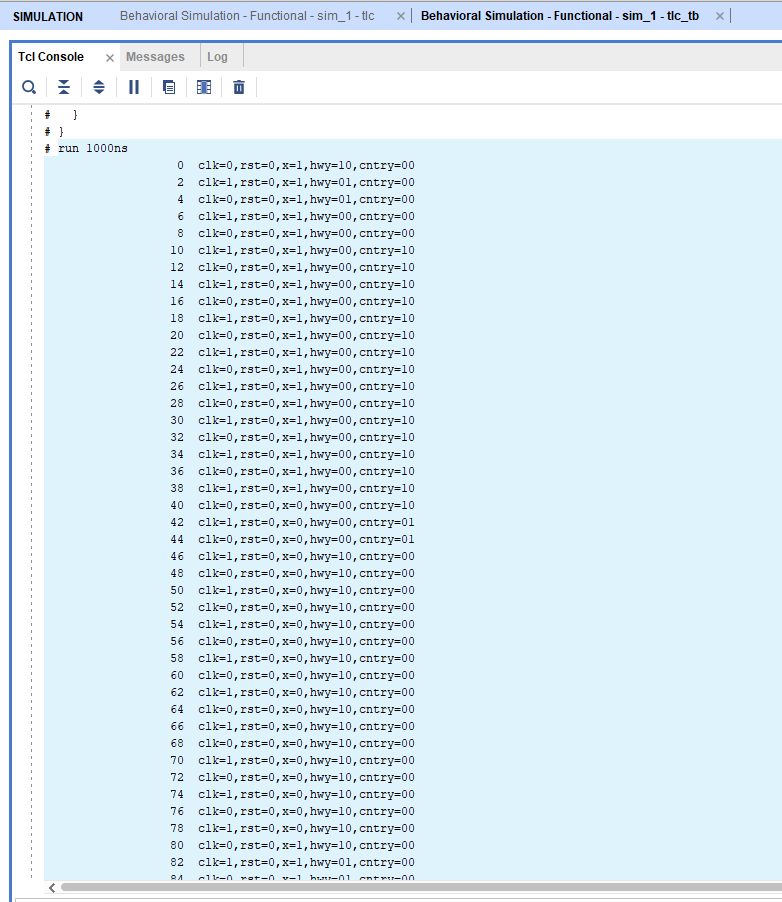
188 clk=0,rst=0,x=1,hwy=00,cntry=10

190 clk=1,rst=0,x=1,hwy=00,cntry=10

192 clk=0,rst=0,x=1,hwy=00,cntry=10

194 clk=1,rst=0,x=1,hwy=00,cntry=10

196 clk=0,rst=0,x=1,hwy=00,cntry=10

198 clk=1,rst=0,x=1,hwy=00,cntry=10Table

Description automatically generatedTable

Description automatically generated