A RECONFIGURABLE CONTROL MECHANISM FOR SMART TRAFFIC MANAGEMENT FOR HIGHLY CONGESTED ROUTE

Varun Singh[1], J.Selvakumar[2]

[1,2]ECE DEPARTMENT, SRM UNIVERSITY, CHENNAI, INDIA

ABSTRACT

Traffic lights are the signaling devices used to manage traffic on multi-way road. These are positioned to control the competing flow of the traffic at the road intersections to avoid collisions. By displaying lights (Red, Yellow and Green), they alternate the way of multi-road users. The implementation of Traffic Light Controller can be through a Microcontroller, Field Programmable Gate Array or Application Specific Integrated Circuit. FPGA implementation is advantageous over ASIC and microcontroller; number of In-Out (I/O) ports and performance compared to microcontroller and implementation with FPGA is less expensive compared to ASIC design. This paper consists of an efficient design of a reconfigurable Traffic Control System which apart from the mainstream traffic controllers, is designed in VLSI using Finite State Machines for programming in FPGA and also it can be reconfigured from a remote place according to the need for emergencies such as ambulances and also can provide a proper adaptable logic for changing signals according to the density of vehicles on the road. The coding of the traffic controller model is done in Verilog HDL and the code and the simulation results are also provided in the paper. The design is simulated and tested on Xilinx Vivado 2013.4 using Model Sim and the configurable logic is also designed and simulated on LabVIEW software of NI to provide a scenario of what is actually happening in the logic and how from one place a whole traffic system can be monitored and controlled without using heinous control systems but simply using a single system such as a computer machine. This system can be implemented in hardware using Spartan-3E FPGA. The system interfacing of lights is simulated on LabVIEW with logic for stopwatch also inserted in it which can be traffic userfriendly.

Index Terms—FPGA, FSM, ASIC, LUT, XILINX, LabVIEW, TLC.

INTRODUCTION

Conventional traffic control systems has two major drawbacks: First, [1] due to lack of adjustments in timings of traffic signals, the traffic has to wait a long on the lane with few vehicles while on same lane, the traffic cannot pass through in short time due to rush on lane. Second, [2] there is no provision of movement of emergency vehicles like ambulance and fire brigades etc. In rush hours these emergency vehicles have to wait a long and results in human and financial loss. So, there is a need to develop a secure, fast, configurable and reliable traffic control system capable to control the vehicular traffic in rush hours without a need of traffic policemen but rather a simple computerhumaninterfaceremotely.Asimplesensorsystemtodetectcarswaitingonroadandtoget the information about the density of traffic or a camera with GSM module can be used to provide necessary information to the single person controlling the traffic and he/she can configure the adjustments in no time. Here we will not go into video streaming or sensor module concepts which can be easily and readily implemented but rather discuss about developmental concepts of the TLC working on a sensor response for detecting traffic density. The basis of the traffic light controller is the FSM which stands for Finite State Machines. For designing a TLC we should know the concept of FSM or rather mealy machines. In this paper, we have developed a real traffic control system using Mealy state machines. The design is implemented in Verilog HDL Hardware Description language.Weusebehaviouralmodelingstyletoimplementstatemachinestoimprovethereadability of code and to increase the speed. The implemented architecture can then be tested for thevalidation of the design on Spartan-3E FPGA DevelopmentKit.

A. Mealymachines

Finite State Machines are used to generate sequence of control signals. There are two types of state machines: Mealy machines and Moore machines. The difference between Mealy and Moore machines relies in the methods of output generation. Mealy machine is a Finite State Machine (FSM) in which the outputs of the machine are dependent not only on the state of the machine but also on the input to the machine.

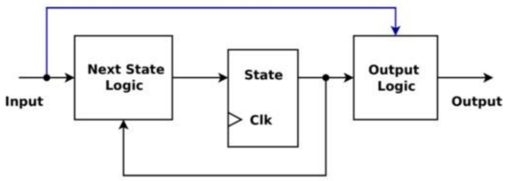


Fig. 1. The Mealy Machine.

B. Field Programmable Gate Array(FPGA).

Reconfigurable hardware platform is useful for the implementation of high digital functions such as this TLC. Using fixed point, parallel computational structures, FPGA provides computational speeds as much as 100 times greater than those possible with Digital Signal Processors (DSP). The extremely fast computational capability of FPGAs allows a few microseconds for real-time computation of algorithms in spite of their complexities. Furthermore, as DSPs, FPGAs are very low cost components [3]-[4]. Xilinx Spartan-3 FPGAs are ideal for low-cost, high-volume applications and are targeted as replacements for fixed-logic gate arrays. The Spartan-3 FPGA is not only available for a very low cost, but it integrates many architectural features associated with high-end programmable logic. This combination of low cost and integrated features has made it an ideal replacement for ASICs (Application Specific Integrated Circuits)[6].

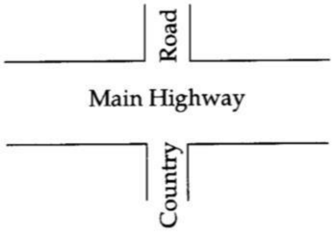
C. Encoding

State encoding scheme have a great influence on circuit size as well as performance. Different types of state encoding schemes can be used like Sequential State Encoding, Gray State Encoding, and One-Hot State Encoding. Out of these encoding schemes, One-Hot encoding is more efficient in terms of speed and size. The choice of state encoding scheme depends on the design application [7]. State machines are widely used in applications that require prescribed sequential activity e.g. sequence detector, digital combinational lock, elevator control, traffic light controller.

TRAFFIC SIGNAL CONTROLLER

2.1 Design

We have considered a controller for traffic at the intersection of a main highway and a country road or a farm road.



Following specifications are considered.

The traffic signal for the main highway/line gets highest priority because cars are continuously present on the main highway. Thus, the main highway signal remains green by default.

Occasionally or in emergency, cars from the country road arrive at the traffic signal. The traffic signalforthecountryroadmustturngreenonlylongenoughtoletthecarsonthecountryroad go.

As soon as there are no cars on the country road, the country road traffic signal turns yellow and then red and simultaneously highway traffic signal glows green.

There is a sensor to detect cars waiting on the country road. The sensor sends a signal X as input to the controller. X=1, then there are cars on the country road; otherwise X=0.

There are delays on transition from S1 to S2, S2 to S3, S3 to S4 and S4 to S0 and these delays must be controllable according to the needs or standards.

2.2 Finite State Machine (FSM) for TLCmodule

The state machine diagram and the state definitions for TLC are shown.

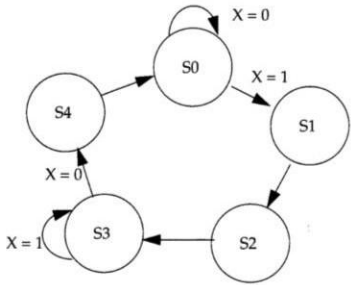


Fig. 2. FSM for Traffic Light Controller.

The state signal values are as follows:-

State Signal Table

TRAFFIC CONTROLLER FSM		
State	***highway***	***country***
S0	G	R
S1	Y	R
S3	R	R
S4	R	G
S5	R	Y

2.2 Structure of TLC.

Structure of the traffic signal control is very basic and can be easily integrated as:

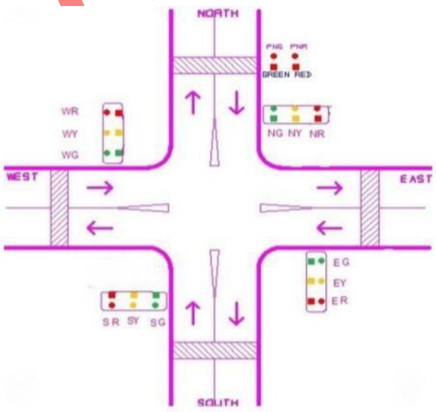


Fig. 3. Basic Traffic Light Controller Structure.

HDL DESCRIPTION FOR TLC

3.1 Program for Traffic Light/SignalControl

```
`define TRUE 1'b1
`define FALSE 1'b0
`define RED2'd0
`define YELLOW 2'd1
`define GREEN 2'd2

//statedefinition  HWY      CNTR
`defineS03'd0      //GREENRED
`defineS13'd1      //YELLOWRED
`defineS23'd2      //REDRED
`defineS33'd3      //REDGREEN
`defineS43'd4      //REDYELLOW

//DELAYS
`define Y2RDELAY 3 //YELLOW TO RED
`define R2GDELAY 2 // RED TO GREEN

module sig_control (hwy,cntry,X,clock,clear);
   output [1:0] hwy,cntry;
   reg [1:0] hwy,cntry;
   input X; //IF TRUE THEN CAR IS ON ELSE OFF
   input clock,clear;

   reg[2:0] state;
   reg[2:0] next_state;

   initial
   begin
      state= `S0;
      next_state=`S0;
      hwy = `GREEN;
      cntry = `RED;
   end
   always@ (posedge clock)
   state = next_state;
```

```
  always@ (state)
  begin
    case(state)
      `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
  always@(state or clear or X)
  begin
  if(clear)
  next_state=`S0;
  else
  case(state)
    `S0: if( X)
    next_state = `S1;
    else
    next_state=`S0;
`S1 :begin
      repeat(`Y2RDELAY) @ (posedge clock);
      next_state =`S2;
    end
    `S2: begin
      repeat(`R2GDELAY) @ (posedge
      clock); next_state=`S3;
    end
    `S3: if( X)
    next_state=`S3;
    else
    next_state=`S4;
    `S4: begin
      repeat(`Y2RDELAY) @(posedge clock);
      next_state = `S0;
    end
    default : next_state = `S0;
  endcase
  end
  endmodule
```

3.2 TEST BENCH for TLC

STIMULUS can be applied to check if the traffic signal transitions correctly when cars arrive on the country road. The following program instantiates the TLC and check all possible states of the controller.

```
`define TRUE 1'b1
`define FALSE 1'b0

module stimulus;

 wire[1:0] MAIN_SIG,CNTRY_SIG;

reg CAR_ON_CNTRY_RD;
reg CLOCK,CLEAR;

sig_control SC (MAIN_SIG,CNTRY_SIG,CAR_ON_CNTRY_RD,CLOCK,CLEAR);

initial
$monitor ($time, "Main Sig = %b Country Sig = %b Car_on_cntry= %b",
MAIN_SIG,CNTRY_SIG,CAR_ON_CNTRY_RD);
```

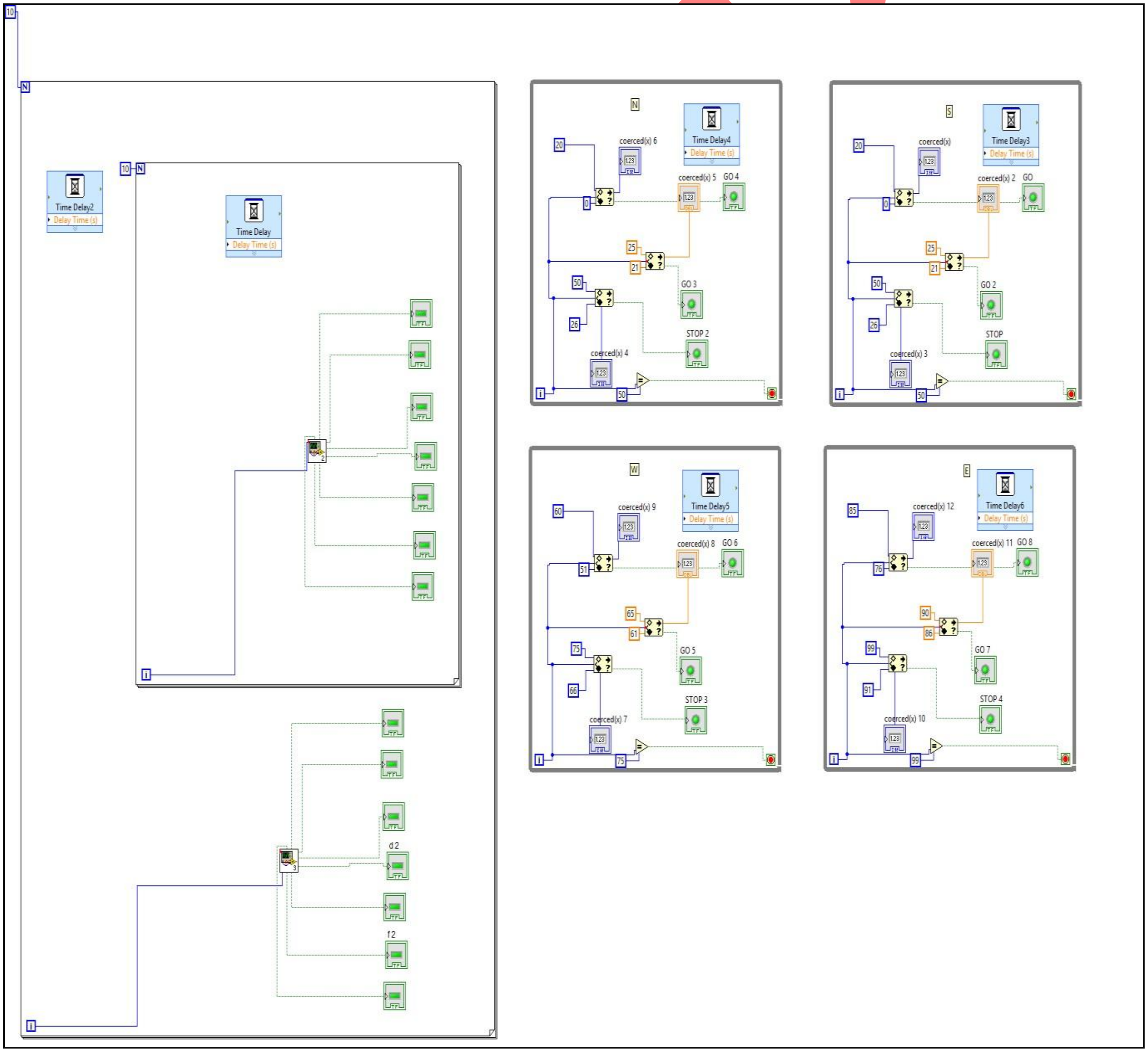
```
initial
begin

   CLOCK = `FALSE;
   forever #5 CLOCK= ~CLOCK;
end

initial
begin

   CLEAR= `TRUE;
   repeat (5) @(negedge CLOCK);
   CLEAR = `FALSE;
end

initial
begin
   CAR_ON_CNTRY_RD= `FALSE;

   #200 CAR_ON_CNTRY_RD = `TRUE;
   #100 CAR_ON_CNTRY_RD = `FALSE;

   #200 CAR_ON_CNTRY_RD =`TRUE;
   #100 CAR_ON_CNTRY_RD=`FALSE;
     #200 CAR_ON_CNTRY_RD=`TRUE;
   #100 CAR_ON_CNTRY_RD=`FALSE;

   #100 $stop;

end
endmodule
```

An important thing to note here is that “sig_control” is the command used to instantiate signal controller.Andthensetupmonitor.Alsoaftermonitoringclockissetupwithclearcontrolandthe stimulus isapplied.

LABVIEW IMPLEMENTATION

AftertheverilogcodingoftheTLC, theLabVIEWphysicalrepresentationofthecontrollerusing basic elements of design were made and also they were made configurable so that the delays and selection for the lights can be remotely controlled by the monitor so that changes can be made accordingly.

Fig.4 .Block diagram for making front panel of the circuit for TLC with stopwatch synchronisation



LabVIEW is short for Laboratory Virtual Instrument Engineering Workbench and it is a system design platform development environment. It is a very helpful tool for circuit and device designers whoarenotthatmuchgoodinprogramminganditalsohasauserfriendlyinterface. Itisextensively used for interfacing and controlling externalperipherals.

TheideaofusingLabviewhereistoprovide,aproperpractical LEDinterfacetothecontrollerso that it becomes more realistic graphically so that any mainstream controller or car driver can see and analysethesituationsofthetrafficscenario,secondlythisLabviewprogram canmaketheTLC fully configurable remotely so that any standard data, of time required for a specific light to glow, can be easily installed in the Mealy FSM of traffic signal controller. Additionally this gives an userfriendly exposure and notation that is useful for the drivers and also because of a stopwatch installed, it becomes more user friendly so that the car drivers can have a pre-requisite knowledge about the situation of traffic and also they get an idea of waiting time needed for them to wait. The implementation in Labview describes the movement and switching of LED or GREEN, RED and YELLOW lights to be specific. The switching of the lights depends on a basic comparator circuit coupled with a timing circuit which is also connected logically to the designed stopwatch. The comparator circuit consists of a timing circuit and a case structure sub VI file together in a loop circuit. The case structure sub VI file is basically a logic designed for a seven segment display. This seven segment display timing algorithm is synchronised with four different timing circuits represented as N, S, E, W which signifies north, south, east, west respectively. These directions timing circuits are the ones used for displaying and controlling the traffic lights. The input of the timing is made user defined or configurable according to the response of the sensors placed at the four way N, S, E, W and is compared with a suitable standard value (threshold) which is basically the country or state standard and hence taking the threshold decisions into consideration the specific lights glows ON, making other two lightsOFF.

4.1 Design specifications ofcircuit

Thecontrollerisdesignedsothataffiliationsarenotrepeatedeachtimeforsamesignal,butrather it depends upon the density or the need of the emergency situations, and accordingly they all aremet withspecificationsprovidedbytheusereasily.Thefollowingstepsaretakentodevelopthisdesign.

4.2 For STOPWATCH : To change the numerical timing values according to the standard second rate digitally which is visible with the traffic lights red, green and yellow which make user updated about the timing when the lights are ON or OFF. It can be count-up or countdown depending upon thedesigner.

Selection : Firstly case-structures are defined for various values of a 7 segment display.

Assignment: Then this 7-Segment display is made to move in a loop of timing and then a sub VI file of this design is created. And this sub VI is taken and another clone is made and the second clone contains the original VI inside it with twice the time period of the original VI.

For LED control(Red, Green and Yellow): To change and control the LED user controller can define the time period for movements by getting the response from the sensor.

Selection: The selection depends upon the desired value of delay and also the necessity of traffic movement for the cars either in the country road or highway.

Copy and change of Assignment: Select the defined comparator circuit and make them synchronised in order to work mutually.

Run: Run the program by assigning proper threshold values and connecting the sensor

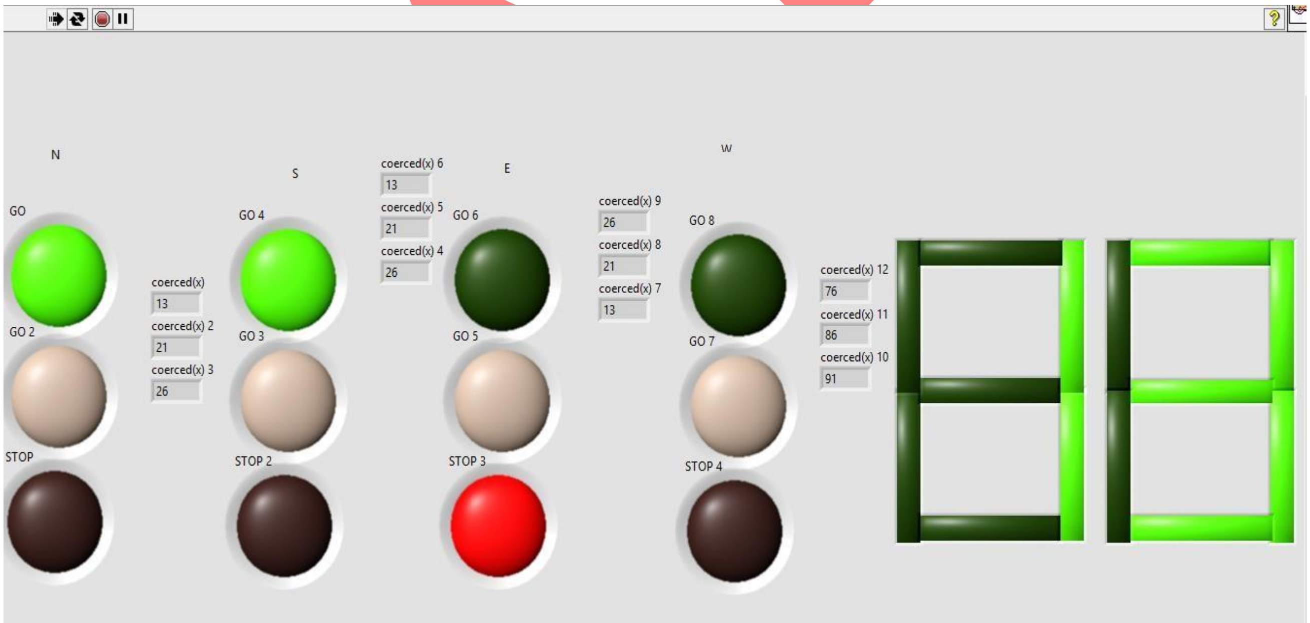
SIMULATION RESULTS

The design of the verilog coding was simulated in XILINX VIVADO 2013.4 ISE with running sim generators in MODEL SIM and the specified and expected output were verified.

This was then simulated under different conditions as shown in fig.6. The simulation results of instantiation of TLC or STIMULUS is displayed in fig.8.

The simulation of the Labview program is also mentioned with switching of the lights of traffic signalinrealtimewiththestopwatchvalueandhereN,E,W,Srepresentslightspresentinnorth,east, west and south respectively. This orientation can be of any form and is upto the standard feature or governmentalspecifications.

Fig. 5. Labview Front Panel of the TLC module



coerced(x) 13
coerced(x) 2 21
coerced(x) 3 26
coerced(x) 6 13
coerced(x) 5 21
coerced(x) 4 26
coerced(x) 9 26
coerced(x) 8 21
coerced(x) 7 13
coerced(x) 12 76
coerced(x) 11 86
coerced(x) 10 91

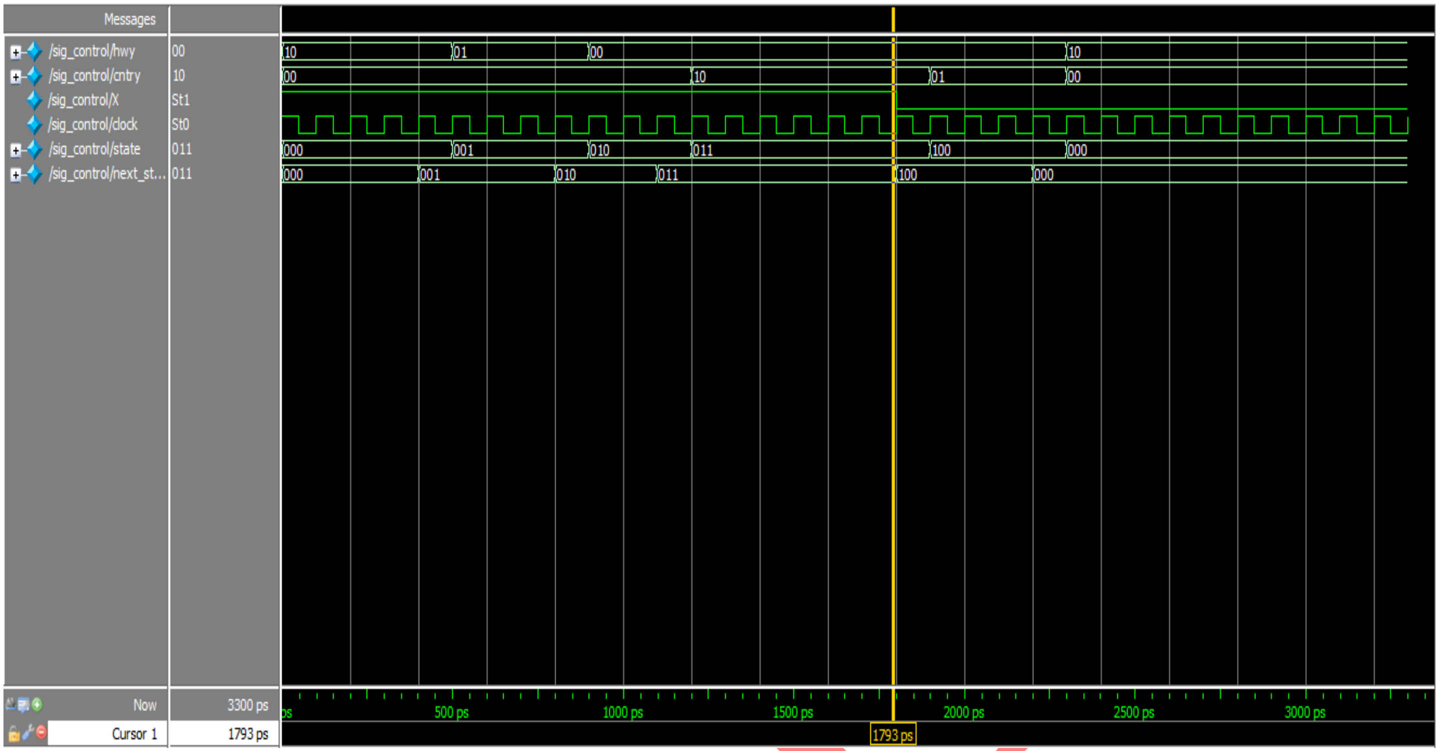


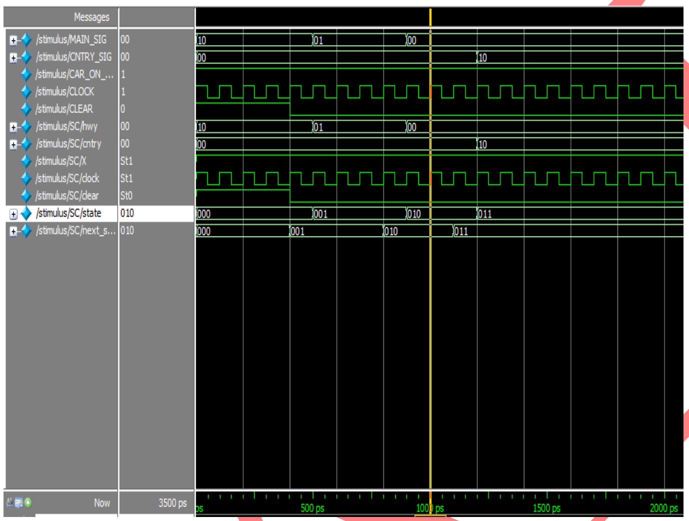
Fig. 6. Simulation result of Verilog program for TLC.

The simulation result of Labview is displayed in fig. 7.This simulation result in Labview shows four traffic lights in an array and a double seven segment display which signifies the stopwatch programmed in the block diagram panel. The three configurable boxes which are present adjacent to each traffic light is the place where the user can input the threshold values according to thestandard. Theprocesscanbeputuseinrealtimeapplicationsandcanbereconfiguredatanyinstance.

RESULTS

Simulation results are shown in fig.7 for HDL based Traffic Light Controller program, we can see that the states getting changed at each change in signal control of highway and country defined as ***sig_control/hwy*** and ***sig_control/cntry*** respectively. The state and is defined according to the traffic at these two places and also the clock pulse given to it, and with these results the next state is also displayed.

Fig. 7. Stimulus for the Verilog code.



The stimulus of the TLC is applied to check if the traffic signal transitions correctly when car arrive on the country road and it also instantiates the TLC and checks all possible states of the controller. Similarly we can see the transitions in Labview program's front panel switching according to the time intervals defined by the user. Note that the change in all the light bulbs takes place simultaneously.

FUTUREWORK

This Verilog program and also the block diagram in Labview can be used for the FPGA implementation of this system which can be put into practice in place of the mainstream traffic control systems. In Labview this can be done by selecting FPGA project on the Getting Started page and calling the VI sequence of this design. Verilog program through Xilinx ISE can be implemented in the hardware Spartan 3E trainer kit. These sequences can solve the real time delay problem seen in many TLCmodels.

CONCLUSION

The modern ways of multi-way smart Traffic management through this TLC improves the traffic condition up to a large extent. Advanced signaling controllers contribute to the improvement of the urban traffic; which is proportional to the complexity of the controller. These more complex controllers can be well handled using States Machines which is implemented in this paper. Methods to reduce the states in the state machine also help in reducing the required hardware thus leading to low power and area efficient design. In addition to that a reconfigurable Traffic Light Control designed on LabVIEW can make it even more specific and the stopwatch associated with it makes it even more modern and helpful. This technology can be made even more efficient by installing real time high grade sensors which can give data input to the TLC at a faster and precise rate so that traffic mechanism could no longer be considered as a problem of this era. Moreover in addition to the general procedure, Xilinx tool gives the flexibility in verification for the design with large number of inputs and outputs, also used for easy implementation of the design into the FPGA Spartan-3E.

REFERENCES

[1] Z. Yuye and Y. Weisheng, “Research of Traffic Signal Light Intelligent Control System Based On Microcontroller,” 2009 First International Workshop on Education Technology and ComputerScience.

[2] W. M. E. Medany and Mr Hussain, “FPGA-based Advanced Real Traffic Light Controller System Desigh,” IEEE International workshop on Intelligent Data Acquisition Computing Systems: Technology and Applications, 6-8 September, Dortmund, Germany,2007.

[3] FPGA Design Tutorial, Copyright © 1-CORE Technologies, 2004-2009. [Online]. Available:http://www.1-core.com/library/digital/fpga-design-tutorial/quick-start-guide.shtml.

[4] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April1955.

[5] Y.S. Huang, T.H. Chung, and T. Lin, "Design and Analysis Urban Traffic Lights Using Timed Colour Petri Nets," in Proc. of IEEE International Conf. on Networking, Sensing and Control, pp. 248-253,2006.

[6] J. Niittymaki And M. Pursula, "Signal control using fuzzy logic," Fuzzy Sets and Systems, vol. 116, no. 1, pp. 11-22,2000.

[7] C.P. Pappis, and E.H. Mamdani, "A fuzzy logic controller for a traffic junction," IEEE Transations on Systems, Man, and Cybernetics, vol. 7, no.10, pp. 707-717,1977.