FPGA based Traffic Light Controller

S. Venkata Kishore[1], Vasavi Sreeja[2], Vibhuti Gupta[3], V.Videesha[4], I. B. K. Raju[5] , K. Madhava Rao[6]
[1][2][3][4] Undergraduate Students, [5] Professor, [6] Assistant Professor
Department of Electronics and Communication Engineering, B V Raju Institute of Technology, Telangana, India
kishore1771996@gmail.com, sreejamuppidi@gmail.com, vibhutigupta.19@gmail.com and videesha.charya2595@gmail.com.

Abstract— The main purpose of the traffic light control system is to control the congestion of vehicles at the junctions and also for safer pedestrian crossing. There have been many technologies used for implementing a traffic light controller all over the world. India being one of the densely populated countries, upgrading to a new control system and imposing it all over is a tedious process. This paper proposes the reconfigurable Traffic Light controller which can display the time of waiting in all the directions. It has been observed that the designed traffic light controller is working up to a maximum operating frequency of about 300 MHz. The coding has been done using the Verilog Hardware Descriptive Language.

Keywords— FPGA (Field Programmable Gate Array), Seven segment displays, LEDs (Light Emitting Diodes), traffic control

I. INTRODUCTION

Traffic jamming is a critical crisis in many of the cities and towns all over the world [1]. The congestion of the traffic has become the cause for many difficulties and challenges in the major and most occupied cities all over the globe. Travel within the cities to the place of work or recreation has become a huge issue. This has led to the people losing their time, money and most essentially the energy resources getting wearied due to the constant use in the automobiles and other applications.

Previously, traffic light controller (TLC) has been implemented using microcontroller. FPGA has many advantages over the microcontroller that is, it is highly configurable, performs high speed computations and has more input output ports necessary for a traffic light controller design [4].

The Traffic Light System proposed in this project aims at minimizing the waiting times of vehicles at the traffic signals [2]. Field programmable gate arrays (FPGAs) are extensively used in electronic systems, for rapid prototyping and verification of a conceptual design, especially when the mass-production of a conventional integrated circuit becomes prohibitively expensive due to the small quantity [6]. Many electronic system designs that used to be built in conventional silicon VLSI are now implemented in Field Programmable Gate Arrays.

The paper mainly works on the design of FPGA traffic light controller for a four-junction road, with two seven segment displays at the directions and LED's for the red and green lights.

The brief review of the remaining paper is- the section II discusses the developed VLSI (Very large scale integration) architecture of the traffic light controller, section III discusses about the hardware setup, section IV gives the obtained simulation and synthesis results followed by the conclusion in the section V.

II. VLSI ARCHITECTURE FOR THE TRAFFIC LIGHT CONTROLLER

The paper mainly works on the design of FPGA traffic light controller for a four-junction road, with two seven segment displays each at the four directions and LED's for the red and green lights. A traffic light controller system consists of three lights (red, green and yellow) in each direction generally [11]. The red light indicates stop, green light indicates the flow of traffic and yellow light gives caution that the traffic is going to be stopped in few seconds. In this the yellow light is excluded and instead a timer circuit is developed to specify the duration for which the traffic is needed to wait [8]. The traffic light controller is a sequential circuit and is modeled as a finite state machine. This contrast is with respect to four junction road, shown in figure 1.

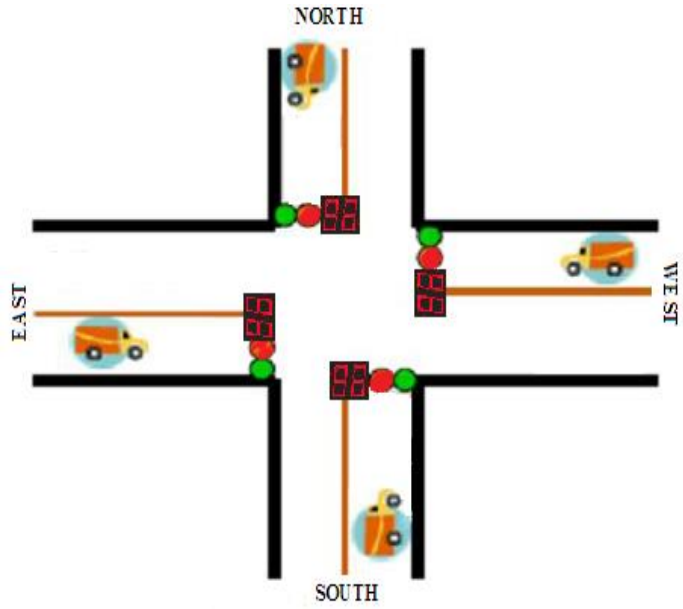

Fig. 1. Signals at the Junction.

A. Proposed Traffic Light Controller

We have considered four junctions i.e. North, West, South and East and five states. Initially the TLC is of reset state i.e. all the junctions displays red light. When the enable switch which is turned on, the TLC will work by transition of states

from S0 to S1for 30 second duration and this cyclic process continues until enable switch is turned low (which is of active low pin of FPGA).

When TLC is in S0 state then green light will be turned on at North junction and simultaneously the red lights, along with their corresponding timers get turned on, at the remaining directions. It can be seen that the down count of timer in West direction will be 30 seconds and that of South and East directions will be 60 and 90 seconds respectively and this continues when there is a state transition. The switching of states and reset button is controlled by the user which is one of the important advantages of this project. Table 1 shows the activity of traffic lights and timer across the each direction.

Table 1–Signals at the junction

STATE TRANSITION	DIRECTIONS			
	NORTH	WEST	SOUTH	EAST
RESET(0000)	RED	RED	RED	RED
S0(1000)	GREEN (30)	RED (30)	RED (60)	RED (90)
S1(0100)	RED (90)	GREEN (30)	RED (30)	RED (60)
S2(0010)	RED (60)	RED (90)	GREEN (30)	RED (30)
S3(0001)	RED (30)	RED (60)	RED (90)	GREEN (30)

B. Internal Architecture of the Traffic Light Controller

The architecture incorporated in the development of the Traffic Light Controller is shown in Figure 2. The different internal components employed in the working of it are as follows:

The system clock of the FPGA is 100MHz and traffic light controller incorporates the clock of time period of 30 second duration. This down conversion is taking place by: 100MHz to1 second duration and from that to 30 second duration.

The 30 second duration signal is given to the Finite State Machine (FSM) which controls the switching mechanism across the different junctions.

Two standard signals of a traffic light controller are used that is RED and GREEN, which carry their usual meanings that of stop and go respectively. The time duration of the traffic that needs to be waiting under RED Light is controlled by the Timer.

A four junction road is being considered here– which could be of highway or country road. Equal duration of 30 seconds is given to all the roads corresponding to the junction (North, West, South and East) under GREEN light, in order to clear the corresponding junction's traffic. This is done by switching of the control from junction to another by Finite State Machine (FSM).

In this process when one of the junction is turned on say North junction for 30 seconds GREEN controlled by FSM, then West, South and East are turned RED. The duration of the traffic that needs to wait in those RED areas is controlled by decay of Timer for 90, 60 and 30 seconds down count is seen across the junction West, South and East. This is repeated and is a cyclic process

In order to view the outputs on the FPGA board the ring counter is being used to enable the tri-state buffers so that when one of the 8 bits of the ring counter value is turned on, the corresponding buffer is turned on and the output of the buffer drives the display of the FPGA.

Since the switching is done at 200 MHz frequency, which is the input to ring counter, the entire display is seen to be turned on of the corresponding junctions. The frequency of 200MHz clock is incorporated to control the ring counter in order to attain the persistence of vision (Which is used for multiplexing).

Once the desired results on FPGA are seen we are realizing the outputs to TLC model (shown in figure 3) by incorporating the external hardware (shown in figure 5) to interface the signals coming out from FPGA board.

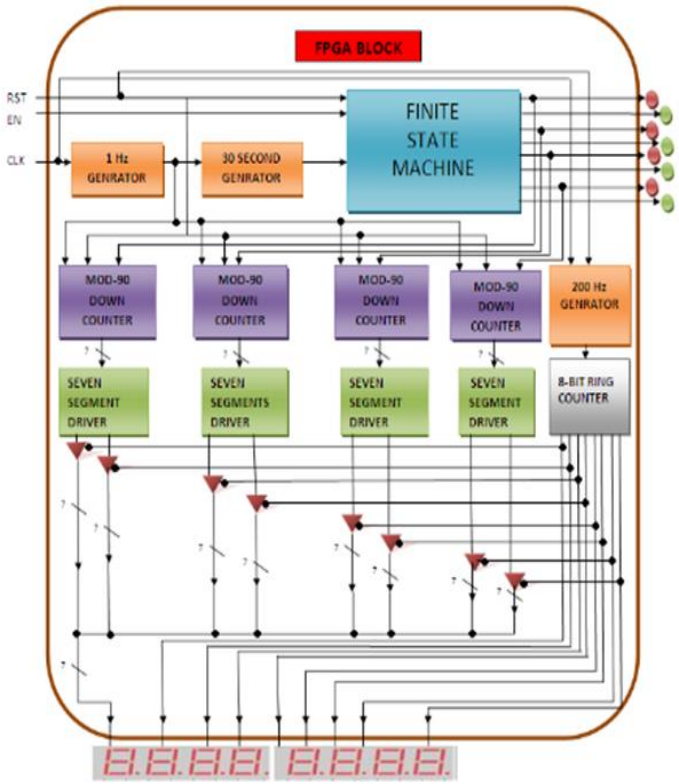

Fig. 2. Internal Architecture of Traffic Light Controller.

Traffic Light Controller is designed on two assumptions. Initially Red signal is in all the directions. Now when the Reset is made high the North side traffic is allowed to move and traffic in all the other directions are stopped. Later the traffic in all the other directions is allowed to move in a cyclic manner [3]. The advantage of this particular Traffic Light Controller, is that the modifications can be done easily as per the requirements i.e., suppose there is traffic on main road and the side road, it can be controlled by changing the states accordingly, i.e., when the main road traffic is heavy when compared to the traffic on the side road. Five states have been chosen for the finite state machine and the state diagram in figure 3 depicts the transition from S1 to S0.

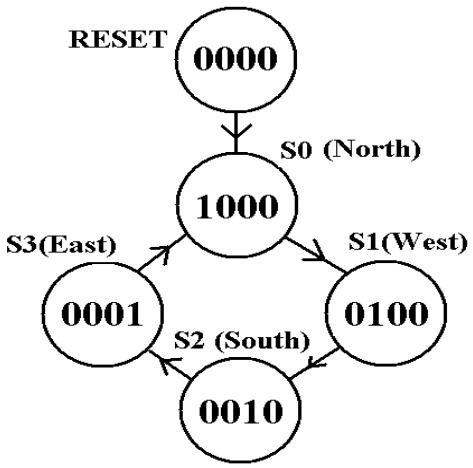

Fig. 3. State Diagram for traffic light controller.

III. HARDWARE IMPLEMENTATION

Nexys4 FPGA Develpomental Kit

This paper works on the design of traffic light controller which is simulated and synthesized in the Xilinx 14.4 version and executed on the Artix7 NEXYS4™ FPGA developmental kit.

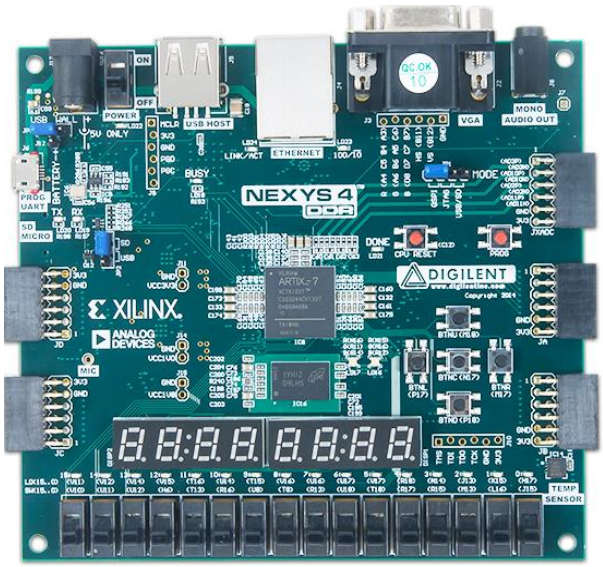

Fig. 4. Artix-7 Nexys4™ FPGA.

Field Programmable Gate Array (FPGA) from Xilinx. With its large, high-capacity FPGA (Xilinx part number XC7A100T-1CSG324C), generous external memories, and collection of USB, Ethernet, and other ports, the Nexys4 can host designs ranging from introductory combinational circuits to powerful embedded processors. Several built-in peripherals, including an accelerometer, temperature sensor, MEMs digital microphone, speaker amplifier and lots of I/O devices allow the Nexys4 to be used for a wide range of designs without needing any other components.

The Artix-7 FPGA (shown in figure 4) is optimized for high performance logic, and offers more capacity, higher performance, and more resources than earlier designs. Artix-7 100T features include:

- 15,850 logic slices, each with four 6-input LUTs and 8 flip-flops
- 4,860 Kbits of fast block RAM
- Six clock management tiles, each with phase-locked loop (PLL)
- 240 DSP slices
- Internal clock speeds exceeding 450MHz
- On-chip analog-to-digital converter (XADC)

The Nexys4 also offers an improved collection of ports and peripherals, including:

- 16 user switches
- USB-UART Bridge
- 12-bit VGA output
- 3-axis accelerometer
- 16Mbyte Cellular RAM
- Pmod for XADC signals
- 16 user LEDs
- Two tri-color LEDs
- PWM audio output
- Temperature sensor
- Serial Flash
- Digilent Adept USB port for programming and data
- Two 4-digit 7-segment displays
- Micro SD card connector
- PDM microphone
- 10/100 Ethernet PHY
- Four Pmod ports
- USB HID Host for mice, keyboards and memory sticks

Initially only 2 user switches are utilized to see the results on FPGA i.e. one is Enable pin (right most switch), to active the TLC and the other is Reset pin (left most switch).

Here Light Emitting Diodes (LEDs) and 7-segment displays are used to represent the outputs. The number of LEDs used is eight which represents four Green and Red lights across four directions. Similarly, the displays are eight in number and each direction corresponds to having two displays. On FPGA the first pair of displays from left corresponds to North, West, South and East directions. This is done by referring to UCF (user constraint file) file of Nexys4™ FPGA board.

Once the results are satisfied on the FPGA we are realizing the outputs onto TLC model by changing the UCF file by utilizing additional ports of FPGA i.e. P-mod ports of JA, JB and JC. Figure 6 shows the interconnection of FPGA with external hardware. Figure 7 shows the working model of TLC.

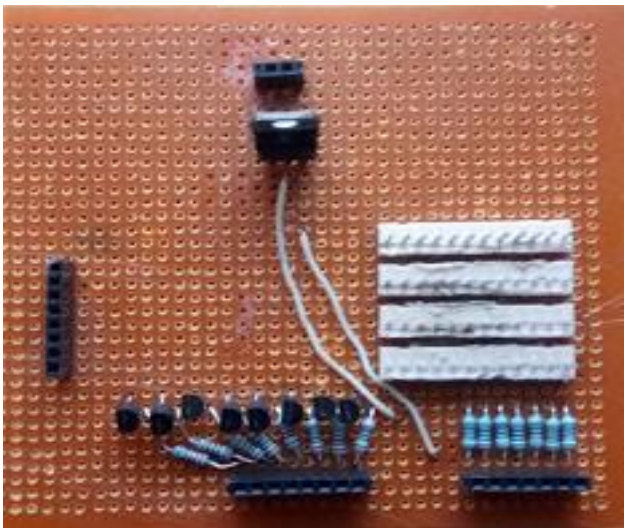

Fig. 5. External Hardware.

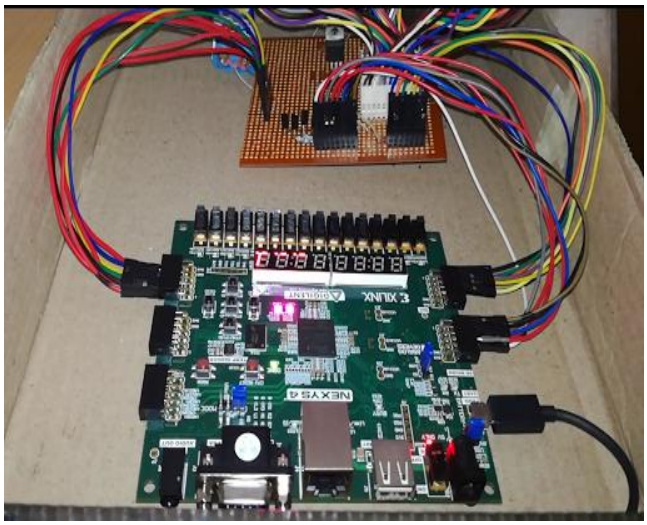

Fig. 6. Hardware connections to the FPGA from the PCB.

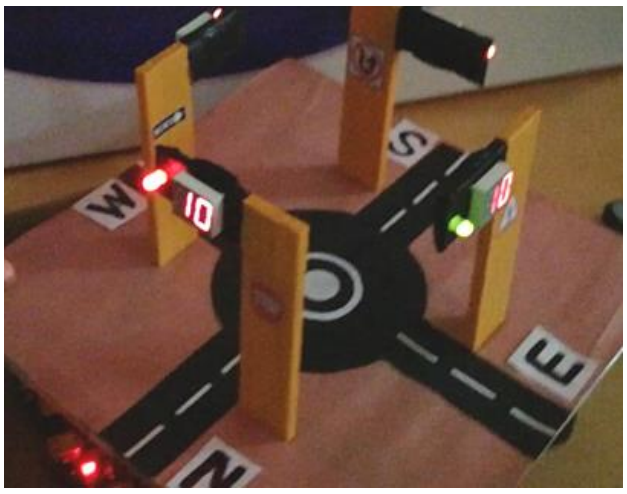

Fig. 7. Working model of TLC

IV. RESULTS

A. Simulation Results

The below figure shows the Wave form of the Traffic Light Controller when the test bench is applied to the source code.

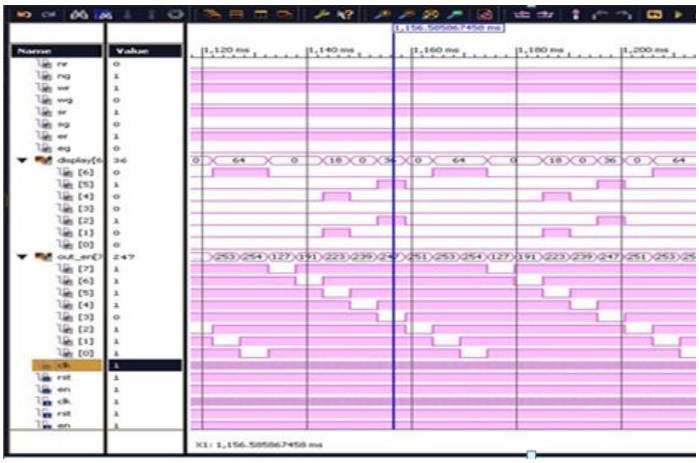

Fig. 8. Simulation Results.

In this waveform the top part of the section consists of the led control signals across all the four junctions. The middle part of the waveform contains the control signals of the seven segment display i.e., “a, b, c, d, e, f and g.” The bottom part contains the enable and clock signals in order to control the displays enable pin and the FSM.

B. Synthesis Results

The figure below shows the RTL Schematic diagram of the Traffic Light Controller of Top Module. The schematic allows one to view, a technology level representation of one’s HDL, optimized for specific device architecture, which may be aiding to discover the design issues early in the design process.

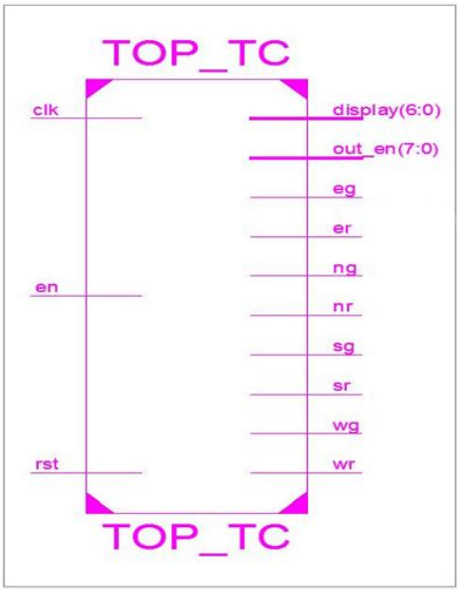

Fig. 9. Top Module Schematic.

The following figure shows the logic that has been realized in the RTL schematic. This is normally done by the tool itself in order to show the input and output pin activity for each and every blocks of the Top Module of TLC.

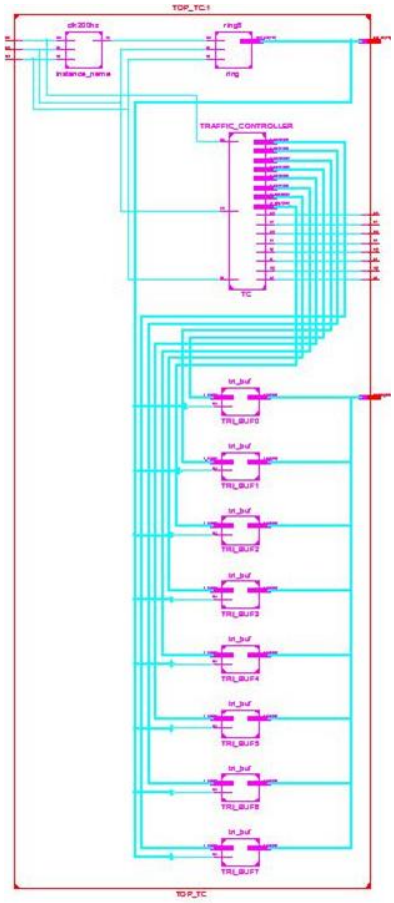

Fig. 10. Inter Logic Blocks of the Schematic.

C. Device utilization summary

This specifies the area that has been utilized, in order to realize the logic in the Artix-7 Nexys4™ FPGA Board. The following figure shows the corresponding results.

```
Device utilization summary:
---------------------------
Selected Device: 7a100tcsg324-3

Slice Logic Utilization:
 Number of Slice Registers:              73 out of 126800    0%
 Number of Slice LUTs:                  266 out of 63400     0%
    Number used as Logic:               266 out of 63400     0%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:    271
   Number with an unused Flip Flop:     198 out of    271   73%
   Number with an unused LUT:             5 out of    271    1%
   Number of fully used LUT-FF pairs:    68 out of    271   25%
   Number of unique control sets:         5

IO Utilization:
 Number of IOs:                         26
 Number of bonded IOBs:                 26 out of    210   12%

Specific Feature Utilization:
 Number of BUFG/BUFGCTRLs:               2 out of     32    6%
---------------------------
```

Fig. 11. Screenshot of device utilization summary.

D. Timing Report

This specifies the timing information related to clock for the operation of the Traffic Light Controller [9]. The following figure gives the timing that has been realized during synthesis.

```
Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 3.428ns (Maximum Frequency: 291.723MHz)
   Minimum input arrival time before clock: 1.135ns
   Maximum output required time after clock: 3.564ns
   Maximum combinational path delay: No path found
---------------
```

Fig. 12. Screenshot of Timing summary.

E. Place & Route and Implementation Results

The Figure 13 shows the place and route of the logic blocks of TLC inside the FPGA.

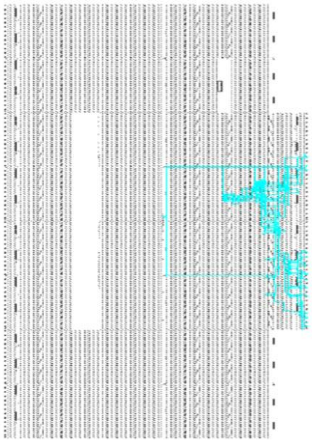

Fig. 13. Screenshot of Place and Route in the FPGA.

Figure 14 shows the successful bit map file generation during dumping of RTL code into FPGA.

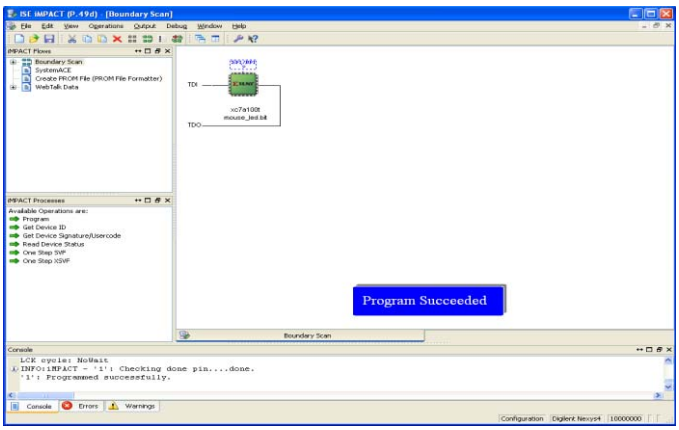

Fig. 14. Screenshot showing the successful dumping of code onto the FPGA.

V. CONCLUSION

The implemented traffic light control system controls complex traffic in modern cities. This system uses FPGA which can be configured by designer or user. The very useful application of FPGA is that designer can change the program at any instant which is easy to reprogram as per the requirement [7]. Verilog HDL is used for writing the code which is dumped on to the FPGA board using Xilinx [9]. Nexys4™ FPGA Board is used as development kit.

In this paper a novel VLSI architecture for the traffic light controller has been developed and implemented on the Artix7 Nexys4™ FPGA developmental kit which is working at a maximum operating frequency of 290 MHz This work can be further extended by reconfiguring the waiting time in all the four directions at a junction based on the traffic density. The FPGA based on which the prototype is designed, has a space utilization of about 1% approximately and on this basis a single FPGA can be employed to integrate a large number of traffic light controllers at every junction which is efficient and cost effective.

Future Scope

For future use, the TLC design with FPGA can include pedestrian crossing lights and also sensors for calculating the density of the traffic thus allowing the free flow of vehicles. The reliability of the design can be much enhanced. Ultimately, a comprehensive and an exceptional TLC design can be implemented into an embedded circuit board to control the traffic flow in the city's traffic intersections.

The future scope of this project is that it can be directly used in real time applications by employing more number of such circuits.

ACKNOWLEDGEMENT

We are thankful to our college management for providing Xilinx tools under university program. This work is carried out at the Center for VLSI design at BVRIT.

REFERENCES

[1] *"Design of FPGA Based Intelligence Traffic light Controller"*, Sourav Nath, Suman Sau, Abhishek Roy, 2012 Conference on Radar, Communication & Computing, SKP College , Tiruvannamalai, TN, December .

[2] Y. Liu, X. Chen, "Design of Traffic Lights Controlling System Based on PLC and Configuration Technology", *Proceeding of International Conference on Multimedia Information Networking and Security 2009*, pp. 561-563, 2009, ISBN 978-0-7695-3843-3.

[3] "*FPGA Implementation of an Advanced Traffic Light Controller using Verilog HDL*", Dilip, Y. Alekhya, P. Divya Bharathi ,Advanced Research in Computer Engineering & Technology; Volume 1, Issue 7,pp: 2278 – 1323,2012.

[4] Wayne Wolf, *"FPGA-Based System Design"* in Prentice Hall, 2005.

[5] Jose E.Ortiz and Robert H. Klenke. "Simple Traffic Light Controller: A Digital Systems Design Project," IEEE SoutheastCon 2010(SoutheastCon), Concord, NC, March 2010, pp. 85-88

[6] W.M. El-Medany and M.R. Hussain. "FPGA-Based Advanced Real Traffic Light Controller System Design," Technology and Applications, 2007, IDAACS 2007. 4th IEEE Workshop, pp. 100-105

[7] M.F.M. Sabri, M.H. Husin, W.A.W.Z Abidin, K.M. Tay and H.M. Basri "Design of FPGA-based Traffic Light Controller System," Computer

Science and Automation Engineering (CSAE), 2011 IEEE International Conference, Shanghai, June 2011, pp.114-118

[8] L. Zhenggang, X. jiaolong, Z. Mingyun and D. Hongwei "FPGA-Based Dual-Mode Traffic Lights System Design," Information Science and Engineering (ICISE), 2009 1st International Conference, Dec. 2009, pp.558-561

[9] https://www.xilinx.com/products/design-tools/ise-design-suite.html, Software manual

[10] https://reference.digilentinc.com/_media/nexys4-ddr:nexys4ddr_rm.pdf

[11] Parasmani, Shri Gopal Modani, " FPGA-Based Advanced Traffic Light Controller Simulation", International Journal of Scientific & Engineering Research, Volume 4, Issue 9, September-2013.

[12] "Structural modeling of intelligent traffic light controller on fpga" , surabhi s. gaopande, ashwini d. Bharade.Department of Electronics Engineering, Visvesvaraya National Institute of Technology, India.