STM JOURNALS

Journal of
VLSI Design Tools & Technology

ISSN: 2249-474X (Online)
ISSN: 2321-6492 (Print)
Volume 15, Issue 2, 2025
May–August
DOI (Journal): 10.37591/JoVDTT

https://journals.stmjournals.com/jovdtt

Research JoVDTT

VLSI-Based Traffic Light Controller

Tanmay Dhamne[1,*], Parth Gaikwad[1], Alisha Shaikh[1],
S.H. Barshikar[2]

Abstract

This study presents the development and implementation of an advanced traffic light control system using VHDL (Very High-Speed Integrated Circuit Hardware Description Language) in conjunction with FPGA (Field Programmable Gate Array) technology. The system is designed to regulate a four-way intersection, consisting of one primary high-traffic road intersecting with a less congested side road. To improve traffic efficiency and minimize idle times, the model incorporates intelligent features such as a vehicle detection sensor for the side street and a pedestrian walk-request mechanism. These components enable the controller to adapt to real-time vehicular and pedestrian demands. Additionally, manual switches are integrated to enable direct user control of signal operations when necessary. The controller functions based on four customizable timing parameters, which govern the signal durations for green, yellow, and red lights, as well as the blinking interval. These timing values are user-adjustable, providing flexibility to accommodate varying traffic conditions. The design was initially developed and verified using the Xilinx ISE 14.7i software suite, allowing thorough simulation and debugging. Real-time signal analysis was performed using ChipScope, ensuring accurate system behavior. The final hardware implementation was carried out on the Xilinx Spartan 3E FPGA, which offers an affordable and adaptable solution for embedded system applications. This system demonstrates reliability, ease of use, and cost-effectiveness, making it a viable alternative to traditional traffic control mechanisms. Its modular architecture supports future upgrades, making it suitable for deployment in urban traffic management systems that require intelligent and responsive solutions.

Keywords: VLSI-based traffic controller, FPGA, VHDL, Xilinx ISE

INTRODUCTION

In today's urban environment, traffic congestion is becoming an increasingly serious problem that affects millions of people's daily lives and places a heavy strain on municipal infrastructure. Longer commutes and reduced overall efficiency result from the increased load on road networks caused by growing urban populations. Cities worldwide are plagued by traffic congestion, which reduces productivity for suppliers, employees, and companies, and causes delays. Due to inefficient transportation, congestion not only wastes time but also indirectly increases the cost of products and services. Mismanagement of traffic signal timing and unnecessary delays when highways are comparatively empty are two of the main issues. These inefficiencies are often caused by outdated traffic management systems that lack sufficient flexibility for real-time adjustments. Consequently, traffic police or other human interventions are typically used to address such issues, but they cannot always provide prompt or scalable solutions [1].

The goal of this project was to employ FPGA technology to create an automated traffic light controller that is both economical and efficient.

***Author for Correspondence**
Tanmay Dhamne
E-mail: parthgaikwad149@gmail.com

[1]Student, Department of Electronics and Telecommunication, Sinhgad College of Engineering, Pune, Maharashtra, India
[2]Professor, Department of Electronics and Telecommunication, Sinhgad College of Engineering, Pune, Maharashtra, India

Received Date: March 28, 2025
Accepted Date: May 15, 2025
Published Date: August 21, 2025

Citation: Tanmay Dhamne, Parth Gaikwad, Alisha Shaikh, S.H. Barshikar. VLSI-Based Traffic Light Controller. Journal of VLSI Design Tools & Technology. 2025; 15(2): 40–47p.

Microcontrollers, ASICs, and FPGAs can all be used to operate traffic lights; however, the FPGA is the best option because of its higher performance, faster processing speed, and more input/output ports. Owing to these characteristics, it is ideal for intricate real-time control systems. FPGAs are perfect for quick prototyping and real-time system development because they are reprogrammable and facilitate iterative design modifications, unlike ASICs, which are expensive and fixed in function once created [2–5]. FPGAs' adaptability and reusability of FPGAs improve design flexibility while lowering overall development costs. By using these benefits, the suggested solution seeks to reduce congestion and increase traffic flow efficiency without incurring significant installation costs. This study demonstrates how FPGA-based solutions can be used to build dependable and scalable traffic control systems, making them a viable option for contemporary intelligent transportation infrastructure.

Because VHDL can enable thorough modeling and simulation prior to hardware implementation, it was employed in this project to program the FPGA. With the help of this hardware description language, designers can deconstruct complicated systems into more manageable parts. The independent development and testing of each component enable early confirmation of the overall behavior of the system. Reliability is significantly increased, and debugging time on real hardware is decreased by using simulations to identify and fix any issues before actual deployment. This proactive strategy guarantees that the design satisfies all the functional requirements while increasing the development period. VHDL is especially well suited for FPGA projects because of its robust support for test benches and hierarchical design [6]. As a result, employing VHDL lowers the risk and expense of hardware development while streamlining the design process and improving the overall development efficiency.

A clear and logical path for the study is provided by the structure of the paper. The second section provides background information and context for the investigation by reviewing related publications. The third section describes the system's architecture and development methods in detail. The fourth section highlights the system's performance and validates the design through the presentation and discussion of simulation data. The fifth section discusses the hardware implementation, which converts the theoretical concept into a practical arrangement. The final section concludes the study by summarizing the results and offering recommendations for future research.

RELATED WORKS

Numerous studies have been conducted to address complex traffic conditions using advanced traffic light control systems. However, many of these solutions still struggle to effectively adapt to dynamic and real-time traffic scenarios.

A previous study introduced an FPGA-ChipScope Pro and virtual I/O. This system, implemented in Bahrain, supports four-road intersections with pedestrian sensors and camera assistance. In the same year, a traffic controller was designed using a Mealy-based finite-state machine (FSM), where the output signals depended on the inputs. The design, implemented with VHDL and tested on a Spartan-3 FPGA, used mixed modeling styles and had a memory usage of approximately 105 MB [1].

Another study proposed an adaptive controller that can handle customizable lane configurations. Their FPGA-based design, built on the FSM architecture and structural VHDL coding, provided improved robustness [2].

Furthermore, a researcher developed an intelligent controller focused on reducing vehicle waiting times. Using traffic sensors and an FPGA-based design on an ALTERA Cyclone II chip, they successfully demonstrated improved efficiency compared to traditional systems [4].

One study introduced an advanced controller with multiple functionalities, such as standby control, motion detection, and special requests. Verification and simulation were conducted using ModelSim [5]. In addition, an Intelligent Transportation System (ITS) was created using VHDL and FPGA. They utilized IR sensors to detect vehicle density and dynamically adjusted signal timing, which was verified through a ModelSim simulation [6].

TRAFFIC LIGHT CONTROLLER DESIGN

The proposed system manages a four-way intersection, consisting of one main street and three side streets. Each direction is equipped with a standard traffic light configuration: red for stop, green for go, and yellow as a warning. When both red and yellow are simultaneously on, this indicates that green is imminent. The system also includes sensors on the side streets and a pedestrian walk-request button. The controller supports pedestrian walk signals, which have red and green indicators—green permits crossing and red prohibits it [7–9]. The overall system is based on a FSM and includes essential components such as RAM for data storage, timers, frequency dividers, and synchronizers, such as latches, forming a cost-effective traffic light controller.

Finite-State Machine

The FSM was the central control unit. It loads timing parameters from memory, manages the display of traffic light states, and executes various operational modes. Four adjustable timing parameters are defined as follows:

- *TBASE:* Base green time for side street.
- *TEXT:* Extended green time for the main road and pedestrian walk.
- *TYEL:* Yellow light duration.
- *TBLINK:* Blinking interval time.

Users can select and modify these timings using two control switches (L0 and L1). Additional function switches (F0 and F1) set the system into one of the following four modes:

- Read mode
- Write mode
- Normal operation
- Blinking mode

In the **reset state**, all lights are turned off, and the system remains idle until the GO button is pressed.

RAM (D_RAM)

RAM stores four timing parameters. Depending on the mode, users can either read values (shown on HEX-LEDs) or write new values using the system's control switches [10].

Divider, Sec Pulse, and Timer

The divider reduces the FPGA's 50 MHz clock to 1 MHz. The second pulse module generates a 1-second clock signal used for the light timing. The timer was implemented using a simple counter mechanism.

Latch and Sensors

The walk request signal is latched such that once the pedestrian button is pressed, the request is stored until the FSM is ready to process it. Side-street traffic sensors are synchronized via flip-flops for accurate detection [11].

SIMULATION RESULTS

A key advantage of using VHDL is its ability to model and verify a system's behavior before it is physically implemented in hardware. This allows the early detection of design issues and efficient validation of system functionality [12].

The Register Transfer Level (RTL) and technology schematic views of the traffic light controller (TLC) are shown in Figure 1. These diagrams help to visualize how the HDL code translates into logic elements optimized for the FPGA architecture (Figures 2 and 3). The simulation results were obtained using the Xilinx ISE 14.7i environment.

Figure 1. RTL and technology schematic of the traffic light controller system.

Reading/Writing Mode

The traffic lights are turned off, while the HEX-LEDs show the currently selected timing parameters stored in the memory. Users can navigate through different parameters using control inputs. This mode allows for easy viewing and selection of timing settings without activating the traffic signals. It provides a user-friendly method to read or modify configuration data, ensuring precise control over timing adjustments before deployment. Inactive traffic lights in this mode prevent unintended signal operation, allowing safe and efficient parameter handling during system setup or testing [13].

Normal Operating Mode

This simulation demonstrated the full sequence of traffic light operations controlled by an FSM. This accurately reflects the timing transitions and sensor-based responses outlined in Section 3. The simulation ensured the correct functioning of all traffic lights and pedestrian signals, validating their coordinated behavior under normal conditions. This mode confirms that the system operates as intended, transitioning smoothly between states based on time and sensor input. This effectively verifies the design logic and real-world applicability of traffic control systems [14].

HARDWARE IMPLEMENTATION

The traffic light controller was designed by first synthesizing the VHDL code using Xilinx ISE 14.7, which generated a corresponding bitstream file. This bit stream was then uploaded to the Xilinx Spartan 3E FPGA board, specifically to the xc3s500efg320 model. This process enabled hardware-level implementation and testing of the traffic control logic on the FPGA platform. The FPGA output is shown using LEDs to represent the current state of the system. An expansion chip (FX2 MIB) was also used to enable external pin connections, allowing enhanced display functionality. Moreover, the real-time functioning of the traffic light controller on the FPGA and the corresponding output were monitored using ChipScope [15]. This setup provided a clear visualization of the system behavior during operation, both physically via LEDs and digitally through signal analysis, ensuring accurate verification of the design's performance and functionality in real-time conditions.

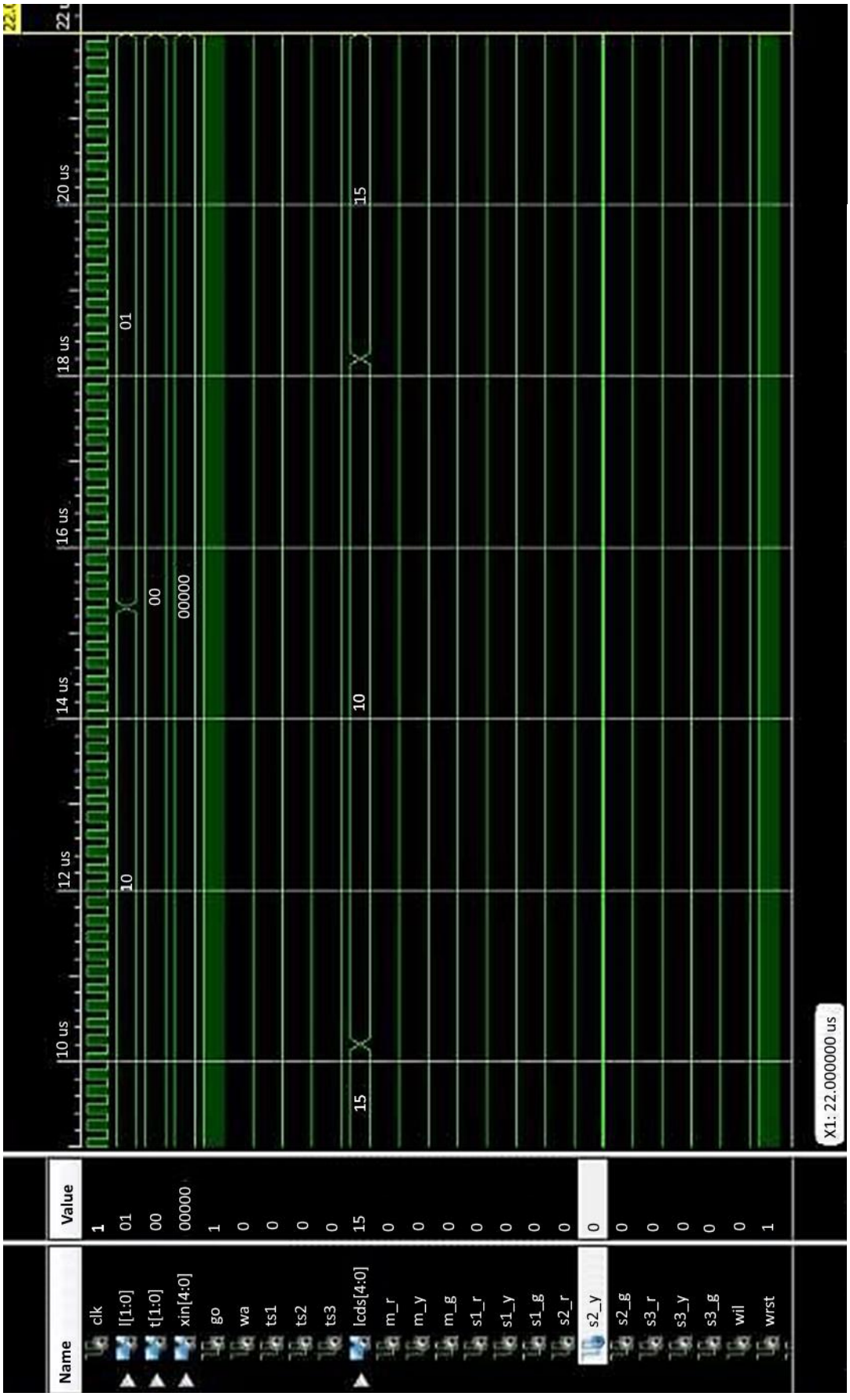

Figure 2. Simulation result of the traffic light controller (TLC) system in reading and writing.

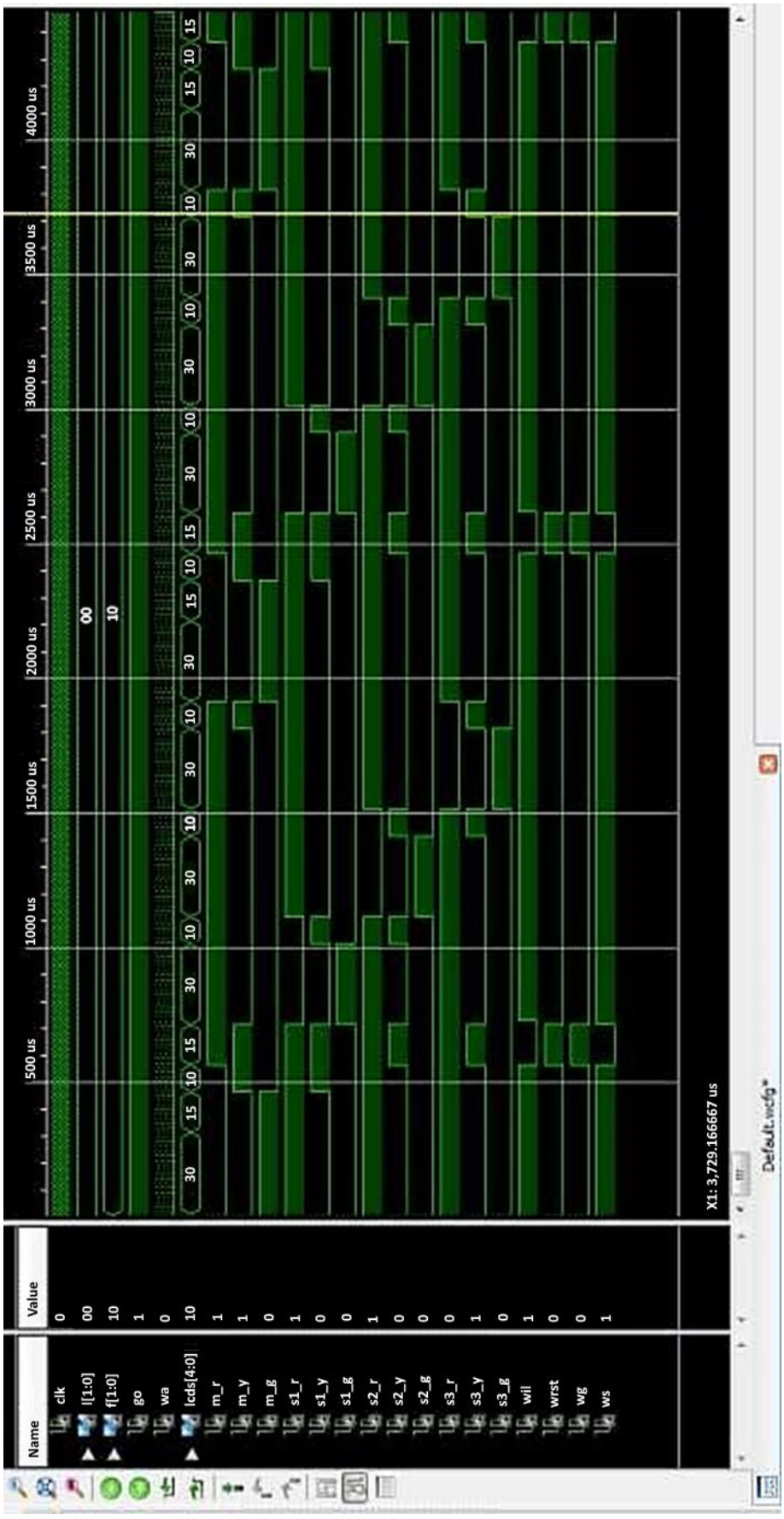

Figure 3. Simulation result of traffic light controller (TLC) in normal mode.

Hardware implementation confirmed the successful execution of both the software and hardware elements of the system. Each subcomponent was thoroughly tested before integration into the complete setup. This ensured that every part functioned correctly and contributed to the overall system performance [16]. The individual verification of modules helped to identify and resolve issues early, leading to a smoother integration process. Consequently, the final system operated as intended, validating the design approach and confirming the effectiveness of the combined hardware and software implementation.

CONCLUSION

A comprehensive smart traffic light controller for a four-way intersection with vehicle sensors and pedestrian requests was successfully designed and realized on an FPGA. The VHDL-based implementation employs a finite-state machine to manage the signal timing, allowing manual adjustment of four key parameters. Each subsystem was rigorously tested in the simulation before integration, and synchronization mechanisms were incorporated to prevent hazards. The final bitstream, loaded onto a Xilinx Spartan 3E xc3s500efg320-4 FPGA, validates the robustness and accuracy of the design. This study demonstrates a scalable framework for developing more sophisticated traffic control systems in the future.

REFERENCES

1. Raza A, Kumar A, Chaudhary E. Traffic light controller using VHDL. Int J Mod Trends Eng Res. 2017;4(4):57-62. doi:10.21884/IJMTER.2017.4118.C529F.
2. Royani T, Haddadnia J, Alipoor M. Control of traffic light in isolated intersections using fuzzy neural network and genetic algorithm. Int J Comput Electr Eng. 2013;5(1):142-6. doi:10.7763/IJCEE.2013.V5.682.
3. Srivastava MD, Prerna SS, Sharma S, Tyagi U. Smart traffic control system using PLC and SCADA. Int J Innov Res Sci Eng Technol. 2012;1(2):169-72.
4. Safi ME. Smart traffic light controller based on microcontroller. Iraqi J Comput Commun Control Syst Eng. 2016;16(1):38-45.
5. Alam J, Pandey PMK. Development of intelligent traffic light system based on congestion estimation using fuzzy logic. IOSR J Comput Eng. 2014;16:36-44. doi:10.9790/0661-16363644.
6. Choudhary AS, Keskar AG, Kothari P, Gajghate A, Pandey A. Adaptive control of traffic grid using fuzzy logic. Int J Electr Electron Data Commun. 2014;2(8):51-7.
7. Shanigarapu L, Reddy KV. Design and implementation of intelligent traffic light system. Int J Comput Sci Mob Comput. 2015;4(7):93-102.
8. Bhaskar L, Sahai A, Sinha D, Varshney G, Jain T. Intelligent traffic light controller using inductive loops for vehicle detection. 2015 1st International Conference on Next Generation Computing Technologies (NGCT), Dehradun, India. 2015. p. 518-22. doi:10.1109/NGCT.2015.7375173.
9. Nath D, Shete VV, Tambe A. Customisation of ARM processor in mixed signal FPGA. Int J Eng Res Technol. 2012;1(6):1-6.
10. Beyrouthy T, Fesquet L. An asynchronous FPGA block with its tech-mapping algorithm dedicated to security applications. Int J Reconfigurable Comput. 2013;2013:1-12. doi:10.1155/2013/517947.
11. Kareem Abdulrazzaq A. Design an intelligent traffic light controller. Univ Thi-Qar J Eng Sci. 2016;7:47-57. doi:10.31663/utjes.v7i1.45.
12. Nath S, Pal C, Sau S, Mukherjee S, Roy A, Guchhait A, et al. Design of an FPGA based intelligence traffic light controller with VHDL. 2012 International Conference on Radar, Communication and Computing (ICRCC), Tiruvannamalai, India. 2012. 2012. p. 92-7. doi:10.1109/ICRCC.2012.6450554.
13. Dilip B, Alekhya Y, Bharathi PD. FPGA implementation of an advanced traffic light controller using Verilog HDL. Int J Adv Res Comput Eng Technol. 2012;1(7):2278-1323.
14. Gadawe NT, Qaddoori SL. Design and implementation of smart traffic light controller using VHDL language. Int J Eng Technol. 2019;8(4):596-602. doi:10.14419/ijet.v8i4.29478.

15. Gaopande SS, Bharade AD. Structural modeling of intelligent traffic light controller on FPGA. Int J Electr Electron Data Commun. 2014;2(9):63-6.
16. Dabahde VV, Kshirsagar RV. FPGA-based intelligent traffic light controller system design. Int J Innov Sci Eng Technol. 2015;2(4):1268-71.

