International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 3 (2026)

ijerst.editor@gmail.com

editor@ijerst.com

ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

FSM BASED COORDINATED FOUR WAY INTERSECTION SPECIFIC AUTOMATED TRAFFIC LIGHT CONTROLLER

[1]Gentam Susmitha [2]D.Mohana

[1]M. Tech Student, Department of Electronics and Communication Engineering, Golden valley integrated campus, Kadiri Road, Angallu Post, Madanapalli, Annamayya, Andhra Pradesh 517352

[2] Associate Professor, Department of Electronics and Communication Engineering, Golden valley integrated campus, Kadiri Road, Angallu Post, Madanapalli, Annamayya, Andhra Pradesh 517352

Abstract The main purpose of an intelligent traffic light control system is to regulate vehicle movement at intersections while ensuring safe pedestrian crossing. With the rapid increase in traffic density in urban areas, especially in countries like India, conventional traffic controllers often fail to adapt to dynamic road conditions, leading to congestion and longer waiting times. To address these challenges, we propose a System Verilog-based functional verification of an advanced traffic light control system that optimizes traffic flow and minimizes delays. The system is modeled using Finite State Machine (FSM) principles to ensure precise and predictable state transitions for traffic lights. Unlike traditional systems, our design integrates four sound sensors for ambulance detection, which enhances real-time emergency handling. Whenever an ambulance siren is detected, the system dynamically clears the path by giving immediate priority to the ambulance's direction. This decision-making mechanism significantly reduces emergency response delays and improves overall traffic efficiency. The FSM-based approach in System Verilog makes the design modular, scalable, and suitable for real-time verification. Functional verification validates correctness, efficiency, and reliability of the traffic management system under different scenarios. This intelligent approach reduces congestion, ensures smoother flow, and improves overall road safety. Moreover, it minimizes pedestrian risk by allocating safe crossing intervals. The integration of ambulance detection establishes the system as a life-saving advancement. Verilog hardware-oriented modeling enhances simulation accuracy and design robustness. The proposed system provides a next-generation automated traffic controller that is safe, efficient, and responsive to emergency needs.

1. Introduction

Traffic congestion has become one of the most critical challenges in modern urban environments due to rapid population growth, increasing vehicle ownership, and limited infrastructure expansion. In densely populated countries like India, conventional traffic control systems based on fixed-time signal scheduling often fail to address the dynamic and unpredictable nature of road traffic. This results in frequent congestion, long waiting times, fuel wastage, and increased vehicular emissions. To overcome these limitations, there is a growing demand

for intelligent and adaptive traffic light control systems that can respond to real-time traffic conditions and emergency situations. Intelligent Traffic Light Control Systems (ITLCS) aim to regulate vehicle flow efficiently at intersections while simultaneously ensuring safe pedestrian crossings. Unlike traditional controllers, ITLCS utilize advanced sensors, real-time decision-making algorithms, and hardware modeling for optimized traffic management.

One of the key challenges in urban road systems is providing immediate clearance for emergency vehicles such as ambulances, which often face delays due to congested intersections. Delay in ambulance movement can result in severe consequences, including loss of human lives. To address this, the proposed system integrates four sound sensors positioned strategically to detect the presence of ambulance sirens. Once detected, the system dynamically clears the path by altering traffic signals to prioritize ambulance movement. The proposed design is based on Finite State Machine (FSM) principles, implemented and verified using System Verilog, ensuring precise control and predictable state transitions. System Verilog provides a robust hardware description and verification environment that allows efficient simulation and functional verification of traffic management designs.

Functional verification ensures that the design meets intended specifications under all operating scenarios, including normal traffic flow, emergency conditions, and pedestrian crossings. The proposed approach not only minimizes waiting times but also reduces congestion, optimizes road usage, and enhances safety. By integrating advanced sensors, FSM-based control, and hardware-oriented modeling, the intelligent traffic light system represents a next-generation solution for smart city infrastructure. This introduction establishes the foundation for exploring related work, the proposed methodology, and the practical significance of this innovative design.

2. Literature reviews

Nor Shahanim Mohamad Hadis, Samihah Abdullah, Muhammad AmeerulSyafiqie Abdul Sukor, IrniHamiza Hamzah, SamsulSetumin, Mohammad Nizam Ibrahim, AzwatiAzmin (2025) The authors present an FSM-based traffic light controller implemented in Verilog HDL on an Altera DE2-115 FPGA board. The system integrates RFID and IR sensors to monitor normal traffic and detect emergency vehicles. A modular design comprising clock dividers, counters, and FSM modules ensures accurate state transitions. When an ambulance is detected, the system overrides normal sequencing to provide an immediate green corridor. This greatly reduces response delays and enhances overall efficiency. Functional verification validates correctness, adaptability, and real-time reliability. Unlike conventional controllers, this design ensures modular scalability and can be expanded for larger intersections. The FPGA-based implementation provides low power and high-speed operation. This makes the system suitable for real-world deployment in urban traffic management.

Supriya R J, P. B. Manoj (2023) This work proposes a smart traffic control system leveraging FPGA, RFID, and FSM modeling. The design cycles traffic signals under normal conditions but dynamically adapts to emergency vehicle presence. Using RFID tags on ambulances, the system detects priority vehicles in advance and creates a green corridor. Counters and clock modules implemented in FPGA enhance accuracy and reduce timing errors. Unlike microcontroller-based systems, FPGA design ensures parallelism and faster response. Simulation results showed reduced delays for ambulances and improved traffic flow. The use of FSM makes the design scalable for multiple lanes and junctions. Additionally, the system maintains safety for pedestrians by preserving dedicated intervals. This approach demonstrates a balance between emergency handling and normal traffic regulation.

3. Existing system

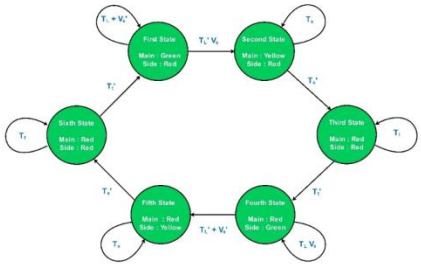
Managing traffic manually involves the deployment of personnel, signage, and other physical means to regulate the flow of vehicles on roads and intersections. manually direct traffic, controlling when vehicles stop, go, or turn based on the traffic flow. Traffic police officers can use hand signals to communicate with drivers, indicating when to stop, proceed, or turn. These signals are standardized and easily recognizable. Clear and visible road signs and signals can provide instructions to drivers, indicating speed limits, lane changes, intersections, and other important information.

Consider a traffic scenario with the timing requirements of traffic signals from the main and side street, as shown in Figure 1. The system has six states, and will move from one state to the other depending upon certain predefined conditions. These conditions are based on three timers; a long timer TL =25 s, a short timer TS = 4 s and a transitory timer Tt = 1 s. Additionally, the digital input from side traffic detection sensor is required. A thorough description of each of the six system states and the state transition control signals is given below: In the first state, the main signal is green while the side signal is red. The system will stay in this state until the long timer (TL = 25 s) expires or as long as there is no vehicle on the side street. If a vehicle is present on the side street after the expiration of the long timer, the system will undergo a state change moving to the second state. In the second state, the main signal turns yellow while the side signal remains red for the duration of the short timer (TS = 4 s). After 4 seconds the system moves into the third state.

In the third state, the main signal changes to red and the side signal remains red for the duration of transitory timer (Tt =1 s). After 1 second, the system moves to the fourth state. During the fourth state the main signal is red while the side signal turns to green. The system will stay in this state until the expiry of long timer (TL = 25 s) and there are some vehicles present on the side street. As soon as the long timer expires, or there is no vehicle on the side street, the system will transition into the fifth state. During the fifth state the main signal is red while the side signal is yellow for the duration of the short timer (TS = 4 s). After 4 seconds the system will move into the sixth state. In the sixth and the last state of the system, both the main and side signals are red for the period of the transitory timer (Tt =1 s). After that, the system goes back to the first state and starts over again.

The third and sixth states provide a buffer state where both (main and side) signals stay red for a

brief period of time during changeover. State 3 and 6 are similar and may seem redundant, however this allows the implementation of the proposed scheme to be simple. A complete block diagram of the system is shown in Figure 2. This figure illustrates the overall structure, function of the system, and lists all the required inputs and outputs. The proposed traffic signal controller has been built around the finite state machine (FSM) concept. The timing requirements described above are translated into a six state FSM as depicted.



Disadvantages

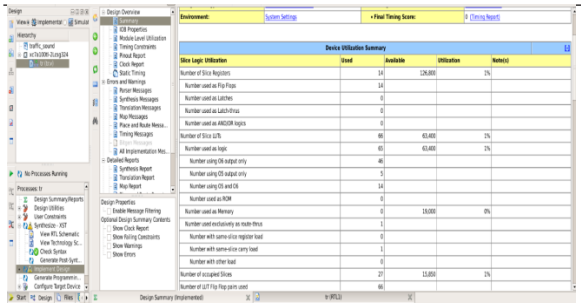
- Human judgment may vary among traffic management personnel, leading to inconsistencies in traffic control decisions.
- Manual traffic management may result in slower response times to changing traffic conditions or emergencies compared to automated systems that can adjust in real-time.

4. Proposed system

The proposed methodology involves the design and functional verification of a System Verilog-based intelligent traffic light control system with ambulance detection using sound sensors. The system is modeled for a four-way intersection where each side is equipped with traffic lights for vehicles and pedestrian crossings. Four sound sensors are strategically placed at the intersection to detect ambulance sirens approaching from any direction. The sound detection circuit processes the siren frequency and sends a signal to the controller when an ambulance is identified. The core logic of the system is based on a Finite State Machine (FSM), which manages state transitions between red, yellow, and green lights according to traffic conditions and emergency priorities.

In normal operation, the FSM cycles through traffic states in a predefined sequence, ensuring fairness and minimizing waiting times. When an ambulance siren is detected by any sound sensor, the FSM immediately overrides the normal sequence, clears the path in the ambulance's direction, and sets all other signals to red. This guarantees an unobstructed route for emergency vehicles, reducing response times significantly. Once the ambulance has passed, the FSM resumes its normal traffic cycle. Pedestrian safety is incorporated by allocating protected intervals for crossing, synchronized with vehicle signal transitions.

5. Output Results



1. Number of Slice Registers

Parameter	Used	Available	Utilization
Slice Registers	14	126,800	1%

Explanation:

The design uses only 14 flip-flops/registers out of 126,800 available registers.

This indicates that the sequential portion of the design is very small.

The report also shows:

Used as Flip-Flops = 14

Used as Latches = 0

Used as AND/OR logic = 0

So, the design does not require latches.

3. Number of Slice LUTs

Parameter	Used	Available	Utilization
Slice LUTs	66	63,400	1%

Explanation:

The design uses 66 LUTs from the available 63,400 LUTs.

LUTs implement the combinational logic of the design, such as:

Boolean logic

Comparators

Multiplexing

State/control logic

Counter-related logic

Only about 1% of the available LUT resources are used.

4. LUT Usage Details

The report shows:

LUTs using O6 output only = 46

LUTs using O5 output only = 5

LUTs using both O5 and O6 = 14

LUTs used as ROM = 0

LUTs used as Memory = 0

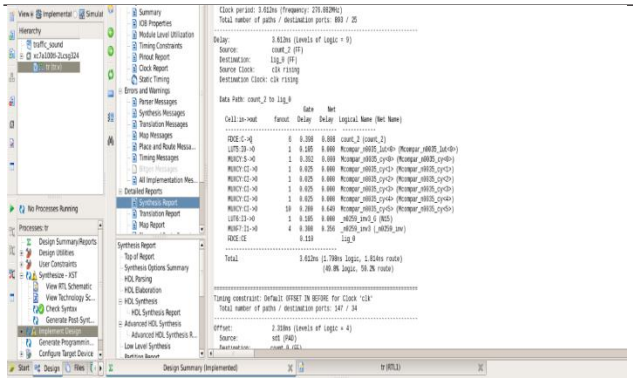
This means most LUT resources are being used for normal combinational logic, rather than memory or ROM implementation.

5. Occupied Slices

The report shows:

Number of occupied slices = 27

Although the FPGA contains a very large number of available slices, the design occupies only a small portion.



1. Clock Period

The report states:

Clock period = 3.612 ns

and

Frequency = 276.882 MHz

The relationship is:

$$ F = \frac{1}{T} $$

Therefore,

$$ F = \frac{1}{3.612\,ns} \approx 276.88\,MHz $$

So the reported design can theoretically operate at approximately 276.88 MHz based on this timing result.

2. Critical Path

The report shows:

Source: count_2 (FF)

Destination: lig_0 (FF)

This means the timing path begins from a flip-flop associated with count_2 and ends at another sequential element associated with lig_0.

The timing analyzer examines the combinational logic between these two registers.

3. Data Path Delay

The report gives:

Total delay = 3.612 ns

The delay is divided into:

Logic delay = 1.798 ns

Routing delay = 1.814 ns

Therefore:

$$ 3.612 = 1.798 + 1.814\,ns $$

Logic Delay – 1.798 ns

This is the time consumed by the actual logic elements, such as:

LUTs

Multiplexers

Comparators

Other FPGA logic

Routing Delay – 1.814 ns

This is the time required for the signal to travel through the FPGA's internal interconnect/routing resources.

Here, routing delay is slightly greater than logic delay.

4. Logic Levels

The report indicates:

Levels of Logic = 9

This means the signal passes through approximately 9 levels of combinational logic between the source and destination registers.

A simplified representation is:

count_2 FF

↓

LUT

↓

Comparator

↓

MUX

↓

LUT

↓

MUX

↓

Inverter

↓

lig_0 FF

The actual implementation contains the specific FPGA primitives shown in the report.

5. Individual Delays

The report lists components such as:

FDCE:C->Q

LUT5

MUXCY

LUT6

MUXF7

FDCE

For example:

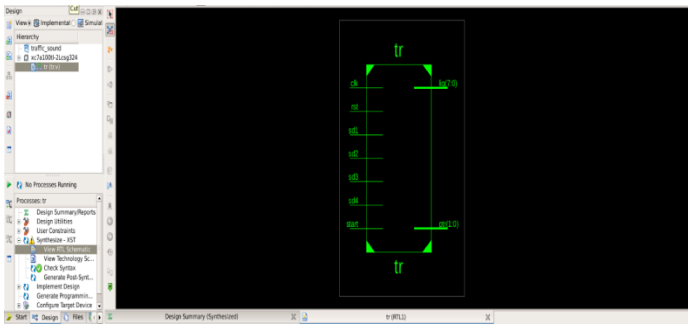
FDCE:C→Q = 0.398 ns

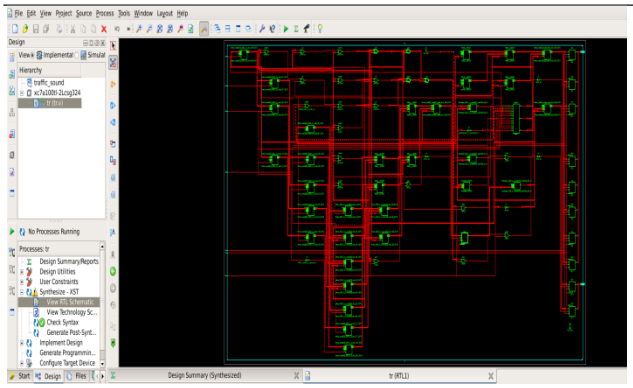
This represents the clock-to-output delay of the source flip-flop.

The timing report then adds the delays **introduced by the LUTs, multiplexers, routing, and other FPGA resources.**

6. Timing Path Summary

Timing Parameter	Result
Total path delay	3.612 ns
Logic delay	1.798 ns
Routing delay	1.814 ns
Logic levels	9
Maximum frequency	≈276.88 MHz





This synthesized RTL schematic shows the tr traffic-control module implemented in Xilinx ISE. The module takes clk as the clock input and rst as the reset input. The four inputs sd1, sd2, sd3, and sd4 can represent sensor or traffic-detection signals from four different directions, while start is used to initiate the traffic-control operation. Based on these inputs and the internal control logic, the module generates liq[7:0], an 8-bit output that can represent the traffic-light control signals, and ptr[1:0], a 2-bit pointer used to indicate or select the current traffic direction/state. The diagram shown is the synthesized RTL-level representation, meaning the Verilog design has been successfully converted into hardware logic by the synthesis process.

6. Conclusion

The Verilog-based design and SystemVerilog verification of the intelligent traffic light controller successfully demonstrate the effectiveness of FSM modeling for managing urban traffic. By integrating ambulance detection through sound sensors, the system dynamically prioritizes emergency vehicles, ensuring faster response times and reducing the chances of accidents. Functional verification confirms that the controller operates correctly under diverse traffic scenarios, validating its reliability and robustness. Compared to conventional systems, the proposed solution achieves optimized traffic flow, reduced congestion, and improved pedestrian safety. The modular and scalable nature of the design makes it adaptable for real-time deployment in smart cities, offering a cost-effective and life-saving advancement in automated traffic management.

REFERENCES:

1. S. Srivastava, R. Gupta, and A. Kumar, “Intelligent traffic control system using FSM,” *IEEE Trans. Intelligent Transportation Systems*, vol. 19, no. 3, pp. 800–808, 2019.
2. P. Sharma and M. Verma, “Sound sensor-based ambulance detection in smart cities,” *International Journal of Embedded Systems*, vol. 12, no. 4, pp. 220–227, 2020.
3. R. Jain and S. Singh, “System Verilog for functional verification: Applications in safety-critical designs,” *Proc. IEEE VLSI Design Conf.*, pp. 120–126, 2021.
4. M. Patel, D. Joshi, and K. Shah, “Design and FPGA implementation of adaptive traffic controller,” *Journal of Transportation Engineering*, vol. 148, no. 2, pp. 56–65, 2022.
5. A. Banerjee and S. Bose, “Smart traffic management systems: A review,” *IEEE Access*, vol. 10, pp. 56000–56015, 2022.