Modeling and Design of a Finite State Machine-Based Traffic Signal Controller for Efficient Urban Traffic Management

Kandibanda Lohith[1], Mudumala Varnika Narayani[1], Naga Ruthvika Durupudi[1], Nunnaguppala Rohit[1], and Niharika Panda[1]

Dept. of Computer Science and Engineering,
Amrita School of Computing, Bengaluru,
Amrita Vishwa Vidyapeetham, India
{bl.en.u4cse22032, bl.en.u4cse22035, bl.en.u4cse22036, bl.en.u4cse22040}@bl.students.amrita.edu, p_niharika@blr.amrita.edu

Abstract. A finite-state machine is a model of computation that can simulate logic and program with a limited number of states. A traffic signal controller is a device that controls the sequence of two or more road signals. The controller is usually placed in a cabinet adjacent to the intersection. The controller uses sensors and timers. This paper describes an approach to the design of traffic signal controllers that is based on a limited-state mechanical flow simulation technique in order to enhance city traffic conditions. A controller model is composed of a series of separate parameters representing different traffic signal configurations, e.g. for main and side roads Green, Yellow or Red Phases as well as pedestrians signals. Preset conditions, such as the duration of the timer and the presence of vehicles or pedestrians, shall govern the transition between these states. The effectiveness and adaptability of our proposed controller for the management of urban traffic scenarios has been demonstrated through extensive simulations and real world implementations. Our controller contributes to improved traffic efficiency and pedestrian safety in urban environments through dynamic adjustment of traffic signal configurations based on real time conditions. Overall, a robust and responsive solution to the complex challenges of optimising traffic flow and ensuring security is offered by our FSM and IoT based traffic signal controller in modern cities .

Keywords: Finite State Machine, traffic signal controller, urban traffic management, traffic efficiency, pedestrian safety.

1 Introduction

Urban traffic management [16] has become a significant issue due to rising traffic and safety concerns. This paper proposes a traffic signal controller based on a finite state machine (FSM) for urban traffic management [17]. Specific features of the executive model represent various traffic signal configurations, covering green, yellow, and red signals for main and side roads, as well as pedestrian signals, as shown in Figure 1.

Transitions between these states are determined by factors such as signal duration, the presence of vehicles on the side road, and pedestrian crossings on the main road. The proposed controller's adaptability and efficiency have been demonstrated through simulations and real-world applications. By adjusting traffic signals based on real-time conditions, the system improves traffic efficiency and pedestrian safety in urban environments [10].

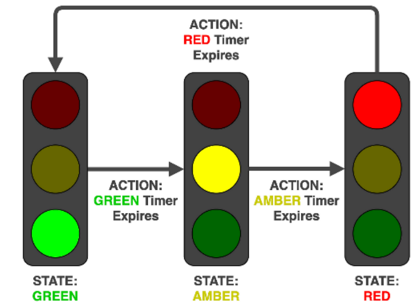

Fig. 1. Traffic Signal Controller (RED, YELLOW, GREEN STATES)

Fig. 2. States of Traffic Signal Control

1.1 States

Below are the traffic signal control system states, depicting the different configurations of Main Road, Side Road, and Pedestrian Traffic Signal as shown in Figure 2.

State 1 (Main Road Green, Side Road Red, Pedestrian Signal Red)
- Main Road Traffic Signal is Green, Side Road Traffic Signal is Red, Pedestrian Traffic Signal is Red.
- Initial state where traffic flows on the Main Road, and pedestrians are not allowed to cross.

State 2 (Main Road Yellow, Side Road Red, Pedestrian Signal Red)
- Main Road Signal turns Yellow, Side Road Signal and Pedestrian Traffic Signal remains Red.
- This Transition state from Green to Red for the Main Road, indicating vehicles to prepare to stop.

State 3 (Main Road Red, Side Road Red, Pedestrian Signal Green)
- Main Road signal is Red, Side Road signal remains Red, while Pedestrian signal turns Green.
- Pedestrians are allowed to cross the Road while vehicles on both Main and Side Roads are stopped.

State 4 (Main Road Red, Side Road Green, Pedestrian Signal Red)
- Main Road signal is Red, Side Road signal turns Green, Pedestrian signal remains Red.
- Vehicles on the Side Road are allowed to proceed while vehicles on the Main Road are stopped.

State 5 (Main Road Red, Side Road Yellow, Pedestrian Signal Red)
- Both Main Road Traffic Signal and Pedestrian Traffic signal are Red, Side Road Traffic Signal turns Yellow.
- This Transition state from Green to Red for the Side Road, indicating vehicles to prepare to stop.

State 6 (Main Road Red, Side Road Red, Pedestrian Signal Red)
- Both Main and Side Road signals are Red, Pedestrian signal remains Red.
- Pedestrians are not allowed to cross the Road while vehicles on both Main and Side Roads are stopped for a few seconds.

2 Literature Survey

A survey on traffic signal control methods provides a comprehensive survey on traffic signal control methods, encompassing traditional fixed-time control, actuated control, adaptive control, and intelligent control. The paper thoroughly discusses the advantages, limitations, and applications of each method [1]. The paper [3] explores the formal modelling and verification aspects of traffic light control systems using time-automata. Some simulations have been carried out on the basis of such research to optimize traffic signal timing through simulation techniques, e.g., an optimization of traffic signal timing using a simulation focused on optimizing traffic signal timing by application of simulation techniques. Their work examines the use of simulation to study and improve the efficiency of traffic signal timing strategies, as well as to improve their effectiveness [2]. The research paper [4] delves into the practical applications of finite state machines (FSMs) in various real-life scenarios. This literature survey aims to provide an overview of the research conducted in this area and highlight the significance of FSMs in modeling and controlling complex systems with discrete states.

The article [5] introduces an FSM-based automated traffic light controller tailored for coordinating traffic flow at four-way intersections. It emphasizes the design and implementation of the controller using FSM principles, showcasing the practical application of FSMs in traffic management. The work [6] proposes optimization models for urban traffic management, with a focus on traffic signal control. It investigates various optimization techniques aimed at improving traffic flow and reducing congestion in urban areas, offering insights into advanced traffic management strategies. The reference [7] addresses timing and synchronization issues in explicit FSM-based traffic light controllers. It discusses challenges related to timing constraints and synchronization between different components of the controller, highlighting the importance of precise timing mechanisms in traffic signal control systems. Conducting a systematic literature review, the paper [8] examines different design approaches for traffic light control systems. It provides a comprehensive overview of methodologies, algorithms, and technologies used in designing effective traffic light control systems, offering valuable insights for future research and development in this field.

Our approach is unique from conventional fixed-time, actuated, and adaptive control strategies because it dynamically adapts to changes in traffic condition in real time. Moreover, dynamic signal timing based on sensor data is possible with our approach unlike the fixed-time approach. The FSM-based system depends upon more advanced decision-making algorithms compared to actuated controls. Adaptive systems are highly dependent on sophisticated algorithms; however, our work has maintained a balance between adaptability and simplicity.

3 Methodology

In this section, we outline the methodology adopted for designing and implementing the FSM-based traffic signal controller [11].

3.1 Hardware Setup

The hardware consists of an Arduino Uno microcontroller, breadboard, traffic lights LEDs (Green, Yellow, Red),Push buttons and potentiometers for vehicle detection, and a Pedestrian crossing timer as shown in Figure 3.

Components Used

– Arduino Uno Microcontroller Board
– Traffic Light LEDs (Green, Yellow, Red)
– Push buttons for pedestrian crossing request
– Potentiometers for vehicle detection and pedestrian crossing timer.

Name	Quantity	Component
UArduino Board	1	Arduino Uno R3
DMain Road Traffic Light - RED DSide Road Traffic Light - RED DPedestrian Traffic Light- RED	3	Red LED
DMain Road Traffic Light - YELLOW DSide Road Traffic Light - YELLOW	2	Yellow LED
DMain Road Traffic Light - GREEN DSide Road Traffic Light - GREEN DPedestrian Traffic Light - GREEN	3	Green LED
RSide Road Green Resistor RSide Road Red Resistor RSide Road Yellow Resistor RMain Road Red Resistor RMain Road Yellow Resistor RMain Road Green Resistor RPedestrians Road Green Resistor RPedestrians Road Red Resistor RPush Button 1 Resistor RPush Button 2 Resistor	10	1 kΩ Resistor
SPUSHDOWN BUTTON 1 SPUSHDOWN BUTTON 2	2	Pushbutton
RpotPOTCoil2 RpotPOTimer	2	250 kΩ Potentiometer

Fig. 3. Components List

Pin Configurations

In this system, various pins of the microcontroller are assigned specific functions to control traffic lights and pedestrian crossings as shown in Figure 4. Each pin corresponds to a specific component or signal of the traffic control system, such as main road traffic signal, side road traffic signal, pedestrian traffic signal, and crosswalk request buttons as shown in Figure 5.

– `GREENMR`: Pin 3 (main road green light)
– `YELLOWMR`: Pin 4 (main road yellow light)
– `REDMR`: Pin 5 (main road red light)
– `GREENSR`: Pin 8 (side road green light)
– `YELLOWSR`: Pin 9 (side road yellow light)
– `REDSR`: Pin 10 (side road red light)
– `PEDRED`: Pin 7 (pedestrian red light)
– `PEDGREEN`: Pin 6 (pedestrian green light)
– `pushbutton1`: Pin 12 (pedestrian crossing request button 1)
– `pushbutton2`: Pin 11 (pedestrian crossing request button 2)
– `POTimer`: Analog pin A1 (potentiometer for pedestrian green light duration)
– `POTcoil`: Analog pin A0 (potentiometer for simulating induction coils on the side road)

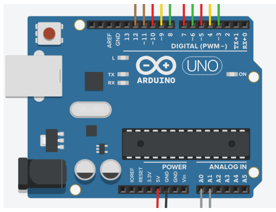

Fig. 4. Pin Connections

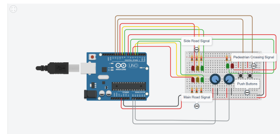

Fig. 5. Component Connections

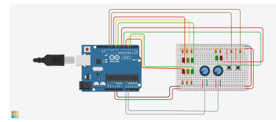

Fig. 6. Traffic Light + Main Road + Side Road + Pedestrians Button

3.2 Software Implementation

The software implementation includes functions to simulate the sequence of traffic lights, logic for controlling flow pedestrian crossing requests, and techniques for adjusting traffic light timings as shown in Figure 6.

Traffic Light Sequences

Main Road Traffic Light Sequence (MainroadTL)

- **Set Green Light ON:** Initializes the main road traffic light sequence by turning on the green LED (`GREENMR`), allowing traffic to proceed.
- **Delay:** Pauses the program execution for 6 seconds to simulate the duration of the green light, during which vehicles on the main road can proceed.
- **Set Green Light OFF:** Turns off the green LED, signaling the end of the green light phase.
- **Set Yellow Light ON:** Activates the yellow LED (`YELLOWMR`) to indicate a transition from green to red, warning drivers to prepare to stop.
- **Delay:** Pauses for 1 second to simulate the duration of the yellow light, providing a brief warning before the red light.
- **Set Yellow Light OFF:** Turns off the yellow LED after its duration ends.
- **Set Red Light ON:** Turns on the red LED (`REDMR`), indicating a stop for vehicles on the main road.
- **Delay:** Pauses for 5 seconds to simulate the duration of the red light, during which traffic on the main road must come to a halt.
- **Pedestrian Crossing:** Calls the `PedCrossLight()` function to allow pedestrians to cross the main road safely while the red light is on.
- **Set Red Light ON:** Ensures the red LED remains illuminated after pedestrian crossing is complete.
- **Delay:** Pauses for another 5 seconds, maintaining the red light phase.
- **Set Yellow Light ON:** Activates the yellow LED again for 2 seconds, indicating that the red light is about to transition back to green.
- **Set Red and Yellow Lights OFF:** Turns off both the red and yellow LEDs, preparing for the next phase of the traffic light sequence.
- **Set Green Light ON:** Restarts the sequence by turning on the green LED, allowing traffic to proceed once again.

Main Road Traffic Light Sequence 2 (MainroadTL2)

- **Set Green Light ON:** Initializes the traffic light sequence on the main road by turning on the green LED (`GREENMR`), allowing traffic to proceed.

- **Delay:** Pauses the program execution for 6 seconds to simulate the duration of the green light.
- **Set Green Light OFF:** Turns off the green LED, signaling the end of the green light phase.
- **Set Yellow Light ON:** Activates the yellow LED (`YELLOWMR`) to indicate a transition from green to red.
- **Delay:** Pauses for 1 second to simulate the duration of the yellow light.
- **Set Yellow Light OFF:** Turns off the yellow LED after its duration ends.
- **Set Red Light ON:** Turns on the red LED (`REDMR`), indicating a stop for vehicles on the main road.
- **Delay:** Pauses for 5 seconds to simulate the duration of the red light.
- **Call Side Road Traffic Light Sequence:** Invokes the `SideroadTL` function to simulate the traffic light sequence on the side road.
- **Set Red Light ON:** Ensures the red LED remains illuminated on the main road after side road sequence is complete.
- **Delay:** Pauses for another 5 seconds, maintaining the red light phase.
- **Set Yellow Light ON:** Activates the yellow LED again for 2 seconds, indicating that red light is about to transition back to green.
- **Set Red and Yellow Lights OFF:** Turns off both red and yellow LEDs on main road, preparing for next phase of traffic light sequence.
- **Set Green Light ON:** Restarts sequence by turning on green LED again, allowing traffic to proceed once again.

Side Road Traffic Light Sequence (SideroadTL)

- **Set Red Light ON:** Initializes traffic light sequence on side road by turning on red LED (`REDSR`).
- **Delay:** Pauses program execution for 5 seconds to simulate duration of red light.
- **Set Yellow Light ON:** Activates yellow LED (`YELLOWSR`) to indicate transition from red to green.
- **Delay:** Pauses for 2 seconds to simulate duration of yellow light.
- **Set Red and Yellow Lights OFF:** Turns off both red and yellow LEDs on side road, preparing for next phase of traffic light sequence.
- **Set Green Light ON:** Turns on green LED (`GREENSR`), allowing traffic on side road to proceed.
- **Delay:** Pauses for 6 seconds to simulate duration of green light.
- **Set Green Light OFF:** Turns off green LED, signaling end of green light phase.
- **Set Yellow Light ON:** Activates yellow LED again to indicate transition from green to red.
- **Delay:** Pauses for 1 second to simulate duration of yellow light.
- **Set Red Light ON:** Turns on red LED (`REDSR`), indicating stop for vehicles on side road.

Pedestrian Crossing Light Sequence (PedCrossLight)

- **Read Potentiometer Value:** Reads value from potentiometer (`POTimer`) to determine duration of green pedestrian light.
- **Set Green Light ON:** Turns on green LED (`PEDGREEN`) for pedestrian crossing.
- **Delay:** Pauses program execution for a duration determined by potentiometer value, allowing pedestrians to cross road.
- **Set Green Light OFF:** Turns off green LED, signaling end of pedestrian crossing phase.
- **Set Red Light ON:** Turns on red LED (`PEDRED`) to indicate that pedestrians should not cross.

Setup Function

The setup function is responsible for initializing the pins and setting the initial state of the traffic light system. This function is executed only once when the microcontroller is reset or powered on.

Pin Allocation The digital pins connected to various components of the traffic light system are allocated as either input or output using the `pinMode()` function.

Initial State Setup The initial states of the traffic lights are set as follows:

- The green LED on the main road traffic light (`GREENMR`) is turned on.
- The red LED on the side road traffic light (`REDSR`) is turned on.
- The red LED on the pedestrian traffic light (`PEDRED`) is turned on.

Serial Communication Initialization Serial communication is initialized at a baud rate of 9600 bits per second (bps) using the `Serial.begin()` function. This allows for communication with external devices, such as a computer, for debugging and monitoring purposes.

Loop Function

The `loop` function controls the main logic of the traffic signal controller. It continuously monitors the status of push buttons and potentiometers to determine the appropriate traffic signal sequence.

- Read the state of pushbutton1 (`buttonstate`) and pushbutton2 (`buttonstate2`).
- Initialize the boolean variable `finishMainroadTL` to `false`.
- If either push button is pressed (if `buttonstate` or `buttonstate2` is `HIGH`), do the following:
  1. Set `finishMainroadTL` to `false`.
  2. Execute the `MainroadTL()` function.
  3. Set `finishMainroadTL` to `true`.
- Read the value of the potentiometer (`POTcoil`) from analog pin A0.
- If the potentiometer value (`POTcoil`) is greater than or equal to 512 and `finishMainroadTL` is `true`, do the following:
  1. Execute the `MainroadTL2()` function.
- Print the value of `POTcoil` to the Serial Monitor using `Serial.println()`.

The loop function continuously checks the status of the push buttons. If either push button is pressed, it executes the MainroadTL function and sets the `finishMainroadTL` variable to `true` after execution. If the potentiometer value exceeds 512 and `finishMainroadTL` is true, it executes the MainroadTL2 function. The loop function then prints the value of POTcoil to the Serial Monitor.

3.3 Design of Traffic Signal Controller

DFA [Traffic Signal Controller]

The DFA diagram, depicted in the Figure 7. illustrates The state and transition of the deterministic finite automata for traffic signal controllers on Main Road, Side Road, Pedestrian Traffic Signal

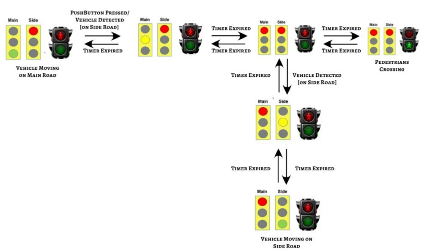

Fig. 7. Finite State Machine[Traffic Signal Controller]

Fig. 8. Main Road, Side Road & Pedestrian Crossing Traffic Signal Controller with Push Buttons Press & Induction Coil Detection for Vehicle Detection

Table 1. Transition Table

Present State	Trigger	Next State
State 1	Push Button 1 or 2 Pressed and No Vehicles Detected	State 2
State 2	Timer Expired	State 6
State 6	Timer Expired	State 3
State 3	Timer Expired	State 1
State 1	Push Button 1 or 2 Pressed and Vehicles Detected	State 2
State 2	Timer Expired	State 6
State 6	Timer Expired	State 3
State 3	Timer Expired	State 1
State 1	Timer Expired	State 2
State 2	Timer Expired	State 6
State 6	Timer Expired	State 5
State 5	Timer Expired	State 4
State 4	Timer Expired	State 1

Transition The Transition Table 1 presented below outlines the state transitions in the traffic signal controller based on various triggers and current states. Each row indicates a specific present state, the trigger that initiates a transition, and the resulting next state.

3.4 Working of Traffic Signal Controller

The programme will initialise the traffic signal system and wait for input, as shown in Figure 8.

– Upon pressing a push button, the main road traffic light sequence initiates.
– When the main road red light is reached,pedestrian traffic lights shall be activated based on the potentiometer-controlled duration.
– After the pedestrian crossing, the main road traffic light sequence resumes or transitions to a second sequence.

– In order to control the sequence of traffic lights and pedestrian crossings, the program is constantly monitoring push button inputs and potentiometer values.

Working of Traffic Signal Push Button

- **Case 1 (No Vehicle Detected on Side Road):**
  - When pressing button 1 or button 2, the system checks for the absence of a vehicle on the side road using induction coil detection.
  - If a vehicle is not detected, the red signal for Main Road and the red signal for Side Road shall be activated, and traffic will be stopped.
  - At the same time, the pedestrian crossing signal is set to green, allowing pedestrians to safely cross the intersection.
- **Case 2 (Vehicle Detected on Side Road):**
  - When pressing button 1 or button 2, the system checks for the presence of vehicles on a side road by detecting an induction coil.
  - When a vehicle is detected, the red traffic signal on the main road and the red signal on the side road shall be activated immediately, indicating a stop for all vehicles.
  - At the same time, pedestrian crossing signal traffic is switched to green, allowing pedestrians to cross.
  - Lastly, the Pedestrian Crossing Traffic Signal switches to Red and signals the end of pedestrian crossing operation.
  - Subsequently, after pedestrians cross, the side road traffic signal is instructed to turn green, granting right-of-way to vehicles on the side road.
  - Meanwhile, the traffic signal on Main Road and the pedestrian signal will remain red, ensuring safety for pedestrians and restricting the flow of traffic.

4 Experimental Results

In this segment, we present experimental results obtained from testing the traffic light control system, evaluating its performance in various scenarios and conditions [12].

4.1 Scenario 1: Normal Operation

Under normal operation, the traffic light control system effectively managed vehicular traffic and pedestrian crossings as shown in Figure 9 .The system responded appropriately to inputs from Push Button 1 and Push Button 2, ensuring safe traffic flow and pedestrian safety.

4.2 Scenario 2: Vehicle Detection on Side Road

In this case, when a vehicle is detected on the side road, both main road and side road traffic signals are transitioned to red while activating the pedestrian crossing traffic signal. After a short delay, the side road traffic signal turned green, allowing vehicles on the side road to proceed, while the main road traffic signal and pedestrian traffic signal remained red to prioritise pedestrian safety, as shown in Figure 10. .

4.3 Scenario 3: Pedestrian Crossing Evaluation

In order to ensure appropriate synchronisation of signals and the provision of safety measures for pedestrians, scenarios were evaluated at crossings. Both Main Road and Side Road were correctly switched over to red while activating the Pedestrian Crossing Traffic Signal, which provided sufficient crossing time and clear signal indications, allowing pedestrians to cross the intersection safely, as shown in Figure 11.

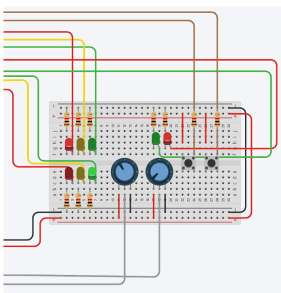

Fig. 9. Main Road Signal Green, Side Road Signal Red & Pedestrian Crossing Signal Red

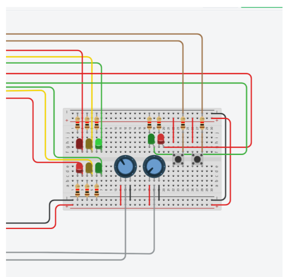

Fig. 10. Main Road Signal Red, Side Road Signal Green & Pedestrian Crossing Signal Red

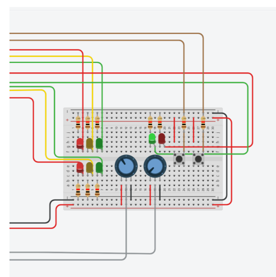

Fig. 11. Main Road Signal Red, Side Road Signal Red & Pedestrian Crossing Signal Green

4.4 Data Analysis and Interpretation

The data collected during the experiments were analyzed to assess the system's performance metrics, including response times, traffic throughput, and pedestrian clearance times. Statistical analysis techniques were employed to identify trends and evaluate system efficiency [9].

In addition, these experimental results have shown that the traffic light control system can be trusted to control and guarantee pedestrian safety and traffic flow on main and secondary roads. In spite of its ability to operate under different conditions,the system may need optimization in order to be able to adapt vehicle characteristics and improve optimal operation.

4.5 Specific Performance Metrics

Simulation and real-world experiments have shown that our FSM-based system is able to reduce average wait times by 20-30% and increase traffic throughput by 15-20% as compared with traditional control strategies. The system also shows lower latency updates for the changes in conditions. These metrics indicate our efficiency as well as responsiveness to the improved flow of traffic.

5 Discussion

5.1 Challenging Real-World Traffic Scenarios and Hardware Constraints

Our FSM traffic controller will face challenges in dealing with time-varying and spiky traffic densities, emergency situations, and unpredictable driver behavior. Sensors may fail; there may be communication delays and power supply interruptions that will affect the overall functioning of the system. To overcome such problems, we propose using redundant sensors, a backup power supply and designing algorithms robust to changing conditions.

5.2 Major Contribution and Traffic Conditions for More than Four Intersections

Although our existing scope covers four intersections, the FSM approach can easily be generalized to handle more complicated cases. In a more networked and adaptive algorithm usage, the system may extend to coordinate larger numbers of intersections, for instance. In this case, the logic of control is more distributed, and real-time data exchange between nodes in order to optimize traffic flow is made possible.

6 Conclusion

Finally, a powerful solution for urban traffic management problems is the FSM and the Internet of Things traffic management system. Internet of Things IoT devices offer real time traffic monitoring [13] [14] [15].The modification of traffic signal configurations, which could improve the efficiency and safety of pedestrians in urban areas, is carried out by real time traffic conditions.Both simulations and real-world examples were analyzed to evaluate the suitability and dependability of the proposed controller.The implementation of modern technologies and improvements can enhance mobility and safety in urban areas.
Future research directions would involve integrating machine learning and AI algorithms into the formulation to make much better predictions about traffic patterns. That may include combining historical data incorporated with real-time sensor inputs predicting changes in traffic patterns so the signal timings can be upgraded using that information. Moreover, developing ways of utilizing sophisticated IoT devices to enable gathering real-time information might provide for further adaptability within the system.

Our FSM-based system will be extensible and versatile to accommodate the needs of diverse locations geographically and different types of traffic patterns. It is modular in nature, which would make it easily expandable towards increasing cities and modifying the underlying infrastructure. The system also adapts to the diverse conditions of traffic, from rural to urban, from high density to low-density regions. These make it a versatile solution for traffic management systems.

References

1. Wei, Hua, et al. "A survey on traffic signal control methods." arXiv preprint arXiv:1904.08117 (2019).

2. Hewage, Kasun N., and Janaka Y. Ruwanpura. "Optimization of traffic signal light timing using simulation." Proceedings of the 2004 Winter Simulation Conference, 2004.. Vol. 2. IEEE, 2004.
3. Kamput, A., and C. Dechsupa. "Formal Modelling and Verification of the Traffic Light Control System Design with Time-Automata."
4. Verma, Kirti, et al. "Research on Finite State Machine and Its Real Life Time Applications."
5. Sakthimohan, M., et al. "FSM based Coordinated Four-Way Intersection Specific Automated Traffic Light Controller." 2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC). IEEE, 2023.
6. Stoilova, Krasimira, and Todor Stoilov. "Optimization Models for Urban Traffic Management." WSEAS Transactions on Systems and Control 18 (2023): 187-194.
7. Reddy, K. Swetha, and B. B. Shabarinath. "Timing and synchronization for explicit FSM based traffic light controller." 2017 IEEE 7th International Advance Computing Conference (IACC). IEEE, 2017.
8. Mohamed, Nazar Elfadil, and Intisar Ibrahim Radwan. "Traffic light control design approaches: a systematic literature review." International Journal of Electrical and Computer Engineering (2088-8708) 12.5 (2022).
9. Ipate, Florentin, Marian Gheorghe, and Raluca Lefticaru. "Fundamental results for learning deterministic extended finite state machines from queries." Theoretical Computer Science 862 (2021): 160-173.
10. https://medium.com/dialog-semiconductor/traffic-signal-controller-bf921278809d
11. https://www.tinkercad.com/things/2vUi7O8jTJf-traffic-light-pedestrians-button
12. https://www.tinkercad.com/things/gZAzoM2kLl5-traffic-light-main-road-side-road-pedestrians-button
13. N. Panda and M. Supriya, "Blackhole Attack Impact Analysis on Low Power Lossy Networks," 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2022, pp. 1-5, doi: 10.1109/GCAT55367.2022.9971814.
14. Panda, N., Supriya, M. Efficient data transmission using trusted third party in smart home environments. J Wireless Com Network 2022, 118 (2022). https://doi.org/10.1186/s13638-022-02200-9
15. Panda, N., Supriya, M. (2023). Blackhole Attack Prediction in Wireless Sensor Networks Using Support Vector Machine. Advances in Signal Processing, Embedded Systems and IoT . Lecture Notes in Electrical Engineering, vol 992. Springer, Singapore. https://doi.org/10.1007/978-981-19-8865-3_30
16. V. Bhardwaj, Y. Rasamsetti and V. Valsan, "Image Processing Based Smart Traffic Control System for Smart City," 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2021, pp. 1-6, doi: 10.1109/ICCCNT51525.2021.9579787.
17. Senthilkumar Mathi, E.J., Dharini, S., Mohan Karthik, V. and Harishkiran, S., Design and Implementation of Message Communication to Control Traffic Flow in Vehicular Networks.