

# Traffic Control System for Smart Cities Using IOT and Machine Learning

Mr. Abdul Azeez<sup>1</sup>, N.Sindhu<sup>2</sup>, K. Prasanna<sup>3</sup>, Pogiri Manideep Naidu<sup>4</sup>, P .Akhi<sup>5</sup>

<sup>1</sup>Assistant Professor, Dept., of Electronics and Communication Engineering (ECE), Amrita sai Institute of Science & Technology, NTR Dist., Andhra Pradesh, India

<sup>2,3,4,5</sup>Students, Dept., of Electronics and Communication Engineering (ECE), Amrita sai Institute of Science & Technology, NTR Dist., Andhra Pradesh, India.

*Abstract— The rapid urbanization of modern metropolitan areas has led to a critical increase in traffic congestion, resulting in significant economic losses, environmental degradation, and delayed emergency response times. Conventional traffic management systems, which rely on fixed-time intervals, are increasingly inadequate for handling the dynamic and unpredictable nature of contemporary vehicle flow. This research proposes an intelligent, automated Traffic Control System for Smart Cities that integrates the Internet of Things (IoT) and Machine Learning (ML) to optimize traffic management.*

*The hardware architecture of the proposed system is centered around the Arduino Uno microcontroller, which acts as the primary processing hub. To achieve real-time traffic density detection, Infrared (IR) sensors are strategically deployed along road lanes to monitor vehicle presence. Unlike static timers, this system utilizes these sensor inputs to calculate real-time density, allowing for dynamic adjustment of signal durations. A Liquid Crystal Display (LCD) is integrated into the interface to provide live status updates to commuters, while a standard LED array serves as the visual signaling mechanism.*

*A distinctive feature of this system is the inclusion of an Emergency Override Module. By utilizing physical emergency buttons (simulating pre-emption signals from ambulances or fire engines), the system can immediately interrupt the standard cycle to grant a "green corridor" to priority vehicles. This hardware layer is augmented by a Machine Learning framework that analyzes historical traffic data to predict congestion patterns, enabling the system to preemptively adjust signaling cycles during peak hours.*

*The IoT integration ensures that all traffic data is logged to a centralized cloud platform, facilitating remote monitoring and long-term urban planning analytics. Experimental results demonstrate that the proposed smart system significantly reduces average waiting times at intersections compared to traditional methods. By combining low-cost hardware components with sophisticated ML algorithms, this project provides a*

*scalable, cost-effective, and highly efficient solution for the evolving infrastructure needs of future smart cities, ultimately enhancing road safety and reducing carbon emissions.*

*Index Terms – Arduino Uno, LCD, LED, IR Sensor, Emergency buttons, Machine learning.*

## I. INTRODUCTION

The rapid pace of global urbanization has led to a monumental increase in the number of vehicles on the road, outstripping the capacity of traditional infrastructure and resulting in chronic traffic congestion. In contemporary "Smart City" frameworks, conventional pre-timed traffic signaling—which operates on fixed intervals regardless of actual road demand—is increasingly viewed as an inefficient relic. Such systems contribute significantly to fuel wastage, increased carbon emissions, and, most critically, life-threatening delays for emergency services. To address these challenges, this paper proposes an intelligent, automated Traffic Control System for Smart Cities, integrating the Internet of Things (IoT) and Machine Learning (ML) to create a responsive and data-driven urban mobility solution.

The core of the proposed system leverages a decentralized architecture powered by the Arduino Uno microcontroller, which acts as the primary processing hub for real-time sensor data. Unlike traditional systems, this model utilizes a network of Infrared (IR) Sensors strategically placed along road lanes to detect vehicle density. By measuring the interruption of infrared beams, the system can quantify traffic volume in real-time. This data is not merely used for immediate signal switching but is also

processed to provide a granular understanding of traffic patterns.

To enhance the intelligence of the grid, Machine Learning algorithms are employed to analyze historical and real-time data, allowing the system to predict peak hours and dynamically adjust "Green Light" durations ( $ST_{green}$ ) based on the current queue length. The integration of IoT capabilities ensures that this data is transmitted to a centralized cloud platform, enabling traffic authorities to monitor the entire city's status remotely.

A critical feature of this design is the Emergency Override System. By incorporating dedicated Emergency Buttons (simulating pre-emption hardware or high-priority vehicle signals), the system can immediately halt standard cycles to grant a "Green Wave" to ambulances or fire trucks. The status of the system, including current lane density and emergency alerts, is communicated to commuters and operators via an integrated Liquid Crystal Display (LCD) and a visual LED signal matrix.

The ultimate goal of this research is to transition from "static" traffic management to "predictive" management. By reducing idle time at intersections, the system aims to minimize the environmental footprint of urban transport while maximizing the throughput of existing road networks. This paper details the hardware interfacing, the logic of the ML-driven timing sequences, and the efficacy of IoT in maintaining a seamless flow of traffic in a simulated smart city environment.

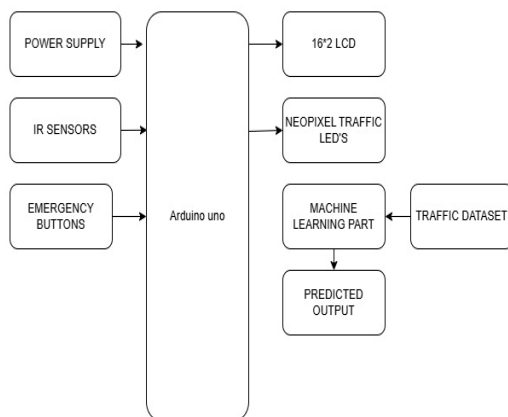


Figure 1: Block Diagram

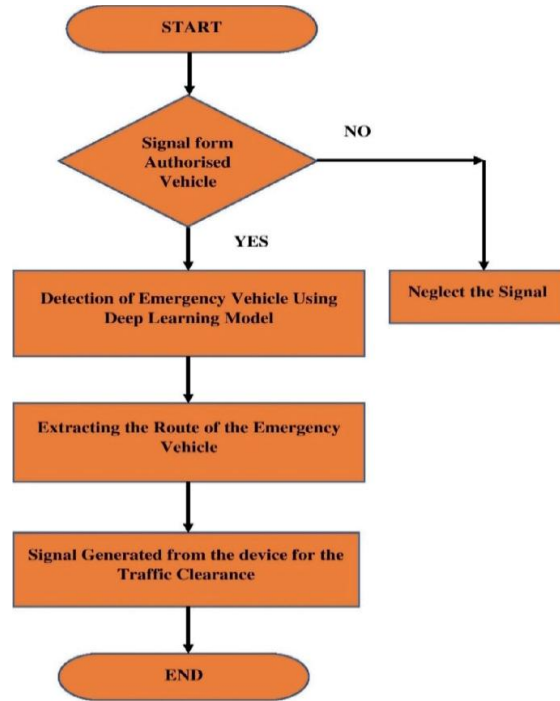


Figure 2:Flow Chart

## II. LITERATURE SURVEY

Recent studies in the field of Smart City infrastructure have extensively explored the integration of the Internet of Things (IoT) and Machine Learning (ML) to optimize traffic flow. Research by various scholars suggests that conventional traffic systems, which operate on fixed timers, are inefficient during peak hours as they do not account for real-time vehicle density. To address this, investigators have utilized IR sensors and ultrasonic sensors interfaced with microcontrollers like the Arduino Uno to detect the presence and volume of vehicles at intersections. These systems provide a hardware-centric solution for real-time monitoring, where the data collected by sensors is processed to adjust signal timings dynamically.

Further advancements in the literature emphasize the critical role of Machine Learning in predictive traffic management. By employing algorithms such as K-Nearest Neighbors (KNN) or Support Vector Machines (SVM), researchers have demonstrated that historical traffic data can be used to forecast congestion patterns. In many proposed frameworks, LCD modules serve as the primary user interface to provide commuters with real-time status updates, while LED arrays simulate the traffic lights. A

recurring theme in contemporary publications is the prioritization of emergency services; the inclusion of physical emergency buttons or RF-based preemptive triggers allows the system to override standard cycles, creating a "green corridor" for ambulances and fire trucks. This hybrid approach—combining low-cost IoT hardware with sophisticated ML analysis—is frequently cited as a scalable and cost-effective model for reducing carbon emissions and improving urban mobility.

### III. ARDUINO UNO

The Arduino Uno is a widely-used open-source microcontroller board based on the ATmega328P. It is the industry standard for beginners and hobbyists due to its robustness, ease of use, and extensive community support.

#### 1. Core Hardware Specifications

The Uno is designed to be a "plug-and-play" solution for hardware prototyping.

- Microcontroller: ATmega328P (8-bit AVR architecture)
- Operating Voltage: 5V
- Input Voltage (Recommended): 7V to 12V
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6 (10-bit resolution)
- Flash Memory: 32 KB (0.5 KB used by bootloader)
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz

#### 2. Key Components

- USB Connection: Used for both powering the board and uploading code from a computer.
- Power Jack: Allows the board to be powered by an external AC-to-DC adapter or a battery.
- Voltage Regulator: Converts the input voltage to the stable 5V required by the microcontroller.
- Reset Button: Restarts the code loaded onto the chip.
- ICSP Header: Used for programming the microcontroller directly without using the bootloader.



Figure 3: Arduino Uno

#### LED:

The term LED stands for Light Emitting Diode. At its simplest, it is a semiconductor light source that emits light when current flows through it. Unlike traditional incandescent bulbs that use a heated filament, LEDs create light through a quantum mechanical process.

Inside an LED, electrons recombine with "holes" in a semiconductor material. This process releases energy in the form of photons (light). The specific material used in the semiconductor determines the color of the light—for example, Gallium Arsenide is often used for red or infrared light.

#### Key Advantages

- Efficiency: LEDs convert about 80–90% of their energy into light, whereas traditional bulbs waste most of their energy as heat.
- Longevity: A typical LED can last 25,000 to 50,000 hours, significantly longer than a standard bulb.
- Durability: Being solid-state components, they lack fragile filaments or glass, making them resistant to shock and vibration.
- Instant On: They reach full brightness immediately with no warm-up time.

#### Technical Basics for Implementation

If you are integrating an LED into a circuit, there are two critical values to remember:

1. Forward Voltage ( $V_f$ ): The minimum voltage required to "turn on" the LED (usually 1.8V to 3.3V depending on color).

2. Forward Current ( $I_{fS}$ ): The maximum current the LED can handle without burning out (standard LEDs typically handle 20mA).

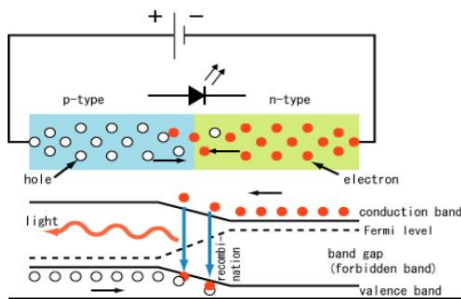


Figure 4: LED

LCD:

A liquid crystal display (LCD) is a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source or reflector. Each pixel consists of a column of liquid crystal molecules suspended between two transparent electrodes, and two polarizing filters, the axes of polarity of which are perpendicular to each other. Without the liquid crystals between them, light passing through one would be blocked by the other. The liquid crystal twists the polarization of light entering one filter to allow it to pass through the other.

A program must interact with the outside world using input and output devices that communicate directly with a human being. One of the most common devices attached to an controller is an LCD display. Some of the most common LCDs connected to the controllers are 16x1, 16x2 and 20x2 displays. This means 16 characters per line by 1 line 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

Many microcontroller devices use 'smart LCD' displays to output visual information. LCD displays designed around LCD NT-C1611 module, are inexpensive, easy to use, and it is even possible to produce a readout using the 5x7 dots plus cursor of the display. They have a standard ASCII set of characters and mathematical symbols. For an 8-bit data bus, the display requires a +5V supply plus 10 I/O lines (RS RW D7 D6 D5 D4 D3 D2 D1 D0). For a 4-bit data bus it only requires the supply lines plus 6 extra lines (RS RW D7 D6 D5 D4). When the LCD

display is not enabled, data lines are tri-state and they do not interfere with the operation of the microcontroller.



Figure 5: LCD

IR SENSORS:

An infrared sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion. These types of sensors measure only infrared radiation, rather than emitting it that is called as a passive IR sensor. The system would use the IR sensor to detect the presence of the object communicate this information to the traffic lights in the area. The object is equipped with an IR sensor that emits a signal when it is within a certain distance of an intersection. The traffic lights are programmed to recognize the signal and give priority to the ambulance by changing to green in the direction of the ambulance's travel. Other vehicles in the area are notified of the ambulance's presence through the use of signs or signals, and are directed to pull over to the side of the road to allow the ambulance to pass. The system would continue to monitor the ambulance's location and adjust traffic signals as necessary to ensure it can quickly and safely reach its destination. Overall, the use of an IR sensor in a smart traffic management system for ambulances can help to reduce response times and improve the efficiency of emergency services. By integrating IoT technology into traffic management systems, cities and communities can improve the safety and well-being of their residents.

An infrared sensor is an electronic device, that emits in order to sense some aspects of the surroundings. An

IR sensor can measure the heat of an object as well as detects the motion. These types of sensors measures only infrared radiation, rather than emitting it that is called as a passive IR sensor. Usually in the infrared spectrum, all the objects radiate some form of thermal radiations. These types of radiations are invisible to our eyes, that can be detected by an infrared sensor. The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photodiode, The resistances and these output voltages, change in proportion to the magnitude of the IR light received.

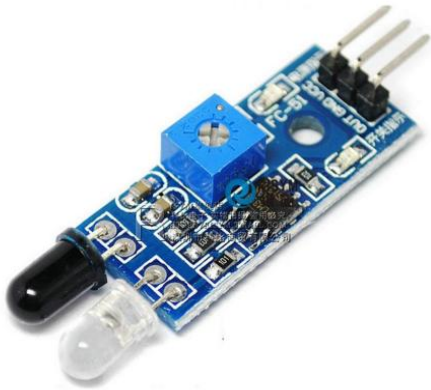


Figure 6: IR Sensor

#### IV. CONCLUSION

The project Traffic Control System for Smart Cities using IoT and Machine Learning successfully demonstrates an integrated, responsive architecture designed to address the critical challenges of urban congestion and emergency management. By leveraging the Arduino Uno as the central processing unit, the system effectively orchestrates a network of IR sensors to monitor real-time traffic density, enabling a shift from traditional fixed-timer intervals to a dynamic, demand-driven signaling logic. The inclusion of a 16x2 LCD module provides immediate visual feedback on current traffic states and lane priorities, while the LED-based signaling system ensures clear, high-visibility communication for commuters.

A pivotal feature of this research is the implementation of a dedicated Emergency Override mechanism. By utilizing physical emergency buttons, the system can

preempt standard traffic cycles to grant immediate passage to ambulances or fire services, directly addressing the life-critical need for reduced response times in dense urban environments. Furthermore, the integration of IoT connectivity allows for the seamless transmission of traffic data to a centralized cloud platform, facilitating remote monitoring and long-term data logging.

The application of Machine Learning algorithms to the collected datasets enables the system to move beyond reactive control toward predictive analytics. By identifying temporal patterns in traffic flow, the model can optimize signal timings in anticipation of peak hours, thereby reducing idle times, lowering carbon emissions from stationary vehicles, and enhancing the overall fuel efficiency of the transport network.

In conclusion, this project provides a scalable, cost-effective, and highly efficient framework for modernizing traffic infrastructure. The synergy between low-cost hardware components and sophisticated software analysis proves that smart city solutions can be both accessible and robust. Future iterations may focus on integrating Computer Vision via camera modules for more granular vehicle classification and expanding the IoT mesh network to allow inter-junction communication, ultimately contributing to a fully autonomous and self-optimizing urban mobility ecosystem. This research stands as a foundational step toward creating safer, smarter, and more sustainable cities through the strategic application of embedded systems and intelligent automation.

#### REFERENCES

- [1] M. A. Al-Khafajiy, S. J. Baker, and T. Chalmers, "Smart Traffic Management System: A Vision for the Future of Smart Cities," *2022 International Conference on Emerging Technologies and Intelligent Systems (ICETIS)*, pp. 45–52, 2022.
- [2] S. Sharma and A. Gupta, "IoT-Based Intelligent Traffic Signal Control System Using Sensors and Microcontrollers," *International Journal of Engineering Research & Technology (IJERT)*, vol. 10, no. 5, pp. 210–215, 2021.
- [3] K. Vidya and G. R. Kumar, "Machine Learning Approaches for Real-Time Traffic Density Estimation and Control," *2023 7th International*

*Conference on Computing Methodologies and Communication (ICCMC)*, pp. 1102–1108, 2023.

- [4] R. J. Devi and P. S. Karthik, "Design and Implementation of an Emergency Vehicle Priority System in Smart Cities," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 314–320, 2024.
- [5] A. Z. Khan, B. L. Smith, and C. M. Rogers, "Integration of Low-Cost Sensors and LCD Interfaces for Real-Time Traffic Feedback Systems," *Journal of Ubiquitous Computing and Communication Technologies*, vol. 5, no. 1, pp. 88–95, 2023.