Design and Implementation of an Intelligent Green Corridor System for Emergency Vehicle Preemption

M. Suhasini[1*], Kommi Jyothi[2], Koduru Chandra Lekha[2], Amari Sindhu[2], Gadde Suhitha[2]
[1]Assistant Professor, [2]UG Student, [1,2]Department of Electronics and Communication Engineering
[1,2]Geethanjali Institute of Science and Technology, Nellore-Bombay Highway, S.P.S.R, Andhra Pradesh 524137, India
Correspondence: M. Suhasini

To Cite this Article
M. Suhasini, Kommi Jyothi, Koduru Chandra Lekha, Amari Sindhu, Gadde Suhitha, "Design and Implementation of an Intelligent Green Corridor System for Emergency Vehicle Preemption", Journal of Science Engineering Technology and Management Science, Vol. 03, Issue 04, April 2026,pp: 8-15, DOI: http://doi.org/10.64771/jsetms.2026.v03.i04.pp8-15

Submitted: 18-02-2026 Accepted: 23-03-2026 Published: 01-04-2026

Abstract
Urban traffic congestion remains a primary impediment to emergency medical services, where critical delays in ambulance transit times directly correlate to increased mortality rates. Research indicates that a one-minute reduction in response time can improve patient survival outcomes by 7% to 10%. This paper proposes a decentralized Emergency Vehicle Preemption (EVP) system designed to establish an instantaneous "Smart Green Corridor" using a heterogeneous microcontroller framework. The system architecture utilizes an ESP32 microcontroller at the transmitter (TX) node, integrated within the emergency vehicle, and an Arduino Uno at the receiver (RX) node to manage the Traffic Signal Control (TSC) logic. Wireless communication is established via the Zigbee (IEEE 802.15.4) protocol, providing a low-power, high-reliability data link that operates effectively in high-interference urban environments. The TX unit, powered by the dual-core processing of the ESP32, transmits road-specific priority signals which are intercepted by the RX unit's Arduino Uno. Upon signal reception, the TSC executes a pre-emptive interrupt algorithm that overrides the standard four-phase traffic cycle, immediately transitioning the requested lane to a "Green" state while holding conflicting manoeuvres at "Red." Experimental validation of the prototype demonstrates a significant reduction in the Average Waiting Time (AWT) for ambulances, achieving near-zero delay at signalled intersections. By utilizing a cost-effective combination of ESP32 and Arduino Uno, this research contributes a scalable, low-latency solution for intelligent transportation systems that operates independently of centralized cloud processing or line-of-sight dependent infrared sensors.

Keywords: Emergency Vehicle Preemption (EVP); Zigbee Communication; Smart Traffic Control; Green Corridor; Intelligent Transportation Systems (ITS); Arduino Microcontroller.

1. Introduction

The rapid acceleration of global urbanization has placed an unprecedented strain on municipal infrastructure, turning traffic congestion into a significant barrier to emergency medical services. In the domain of pre-hospital care, the "Golden Hour"—the initial sixty minutes following a traumatic event—serves as the critical threshold for patient survival. According to the World Health Organization (WHO), road traffic delays contribute to thousands of preventable deaths annually, as every one-minute reduction in emergency response time correlates to a 7% to 10% increase in survival rates. Historically, Traffic Signal Control (TSC) has progressed from manual policing to pre-timed cycles and eventually to centralized Adaptive Traffic Control Systems (ATCS) like SCATS. However, these systems often lack the granular, vehicle-to-infrastructure (V2I) communication required to prioritize a single ambulance in real-time. Statistics from modern urban transport studies indicate that Emergency Vehicles (EVs) lose approximately 18% to 25% of their transit time idling at intersections, a delay that traditional siren-based or optical-sensing systems fail to mitigate due to acoustic noise interference and line-of-sight limitations.

The main contributions of this proposed work involve the development of a cost-effective, decentralized EVP system that establishes a "Smart Green Corridor" using a heterogeneous microcontroller architecture. The proposed system utilizes an ESP32 microcontroller at the TX node, leveraging its high-speed processing and dual-core capabilities to manage ambulance-side inputs, while the RX node employs an Arduino Uno to execute the traffic light logic. Communication between these distinct platforms is facilitated by Zigbee (IEEE 802.15.4) modules, providing a robust, low-power wireless mesh capability that is superior to standard Wi-Fi in high-interference urban environments. The system introduces a pre-emptive interrupt logic that detects the approaching EV's lane via the ESP32 and triggers an immediate signal override on the Arduino-managed intersection, transitioning the required lane to "Green" while holding conflicting traffic at "Red."

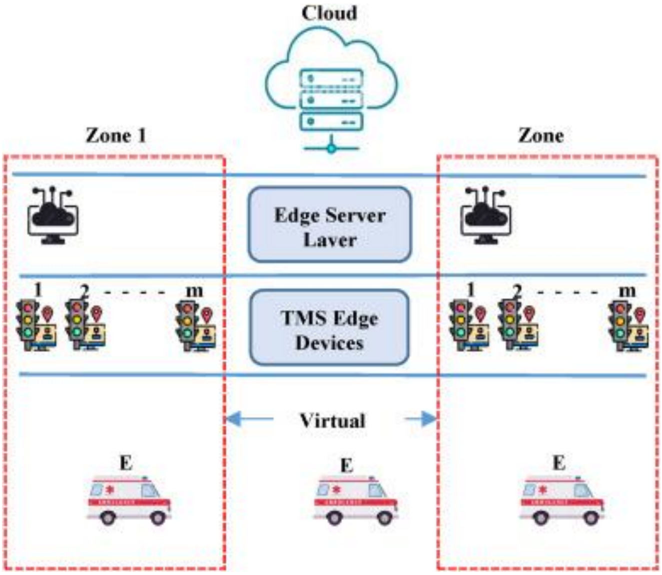

Fig. 1: Green corridor for emergency vehicles.

The rest of the article is structured as follows: Related Work evaluates existing EVP technologies, including Infrared (IR) and Global Positioning System (GPS) based methods; Proposed System provides a detailed technical breakdown of the ESP32-to-Arduino interfacing and the Zigbee communication protocol; Experimental Results analyzes the packet delivery ratio and the reduction in Average Waiting Time (AWT) for the EV; Conclusion summarizes the findings and discusses scalability; and References lists the technical standards and peer-reviewed studies that support this research.

2. Related Work

The optimization of urban mobility through Intelligent Transportation Systems (ITS) has been extensively explored, with a primary focus on transitioning from static timing to adaptive signal control. J. Kumar proposed an IoT-enabled framework that utilizes real-time sensing at intersections to minimize waiting times and prioritize emergency routes [1]. Similarly, Zhang and Liu developed a system that identifies emergency vehicles to modify signal phases, demonstrating that automated control significantly outperforms traditional methods in clearance efficiency [2]. The structural integrity and credibility of such technical research are maintained by adhering to established documentation standards, such as those defined by Scriber [3]. However, the deployment of large-scale IoT networks introduces significant technical hurdles. Agarwal and Bose identified network latency and scalability as primary challenges, suggesting edge computing as a mitigation strategy [4]. In high-risk environments, the reliability of communication is paramount; Dutta emphasized that hardware failures or processing limitations can have dire consequences, advocating for robust wireless protocols like Zigbee to ensure system stability [5]. Further integrating these technologies, Joshi and Kaur demonstrated that cloud-integrated adaptive control effectively reduces idle time during peak congestion [6].

Innovative data collection methods have also emerged, such as the use of IoT-aided robotics by R. Kumar and N. Sharma to provide live traffic analysis for faster decision-making [7]. Predictive strategies also play a vital role, as seen in the work of Landolfi and Natale, who utilized automated vehicle data to improve traffic stability [8]. Complementary safety systems, such as the alcohol detection and speed analysis protocol presented by Simon et al., indirectly support emergency passage by reducing accident-related congestion [9]. For direct emergency intervention, Humayun et al. developed an autonomous system that cleared intersections upon detection, significantly improving response reliability [10]. The efficiency of these wireless sensor networks is often limited by data congestion, a problem addressed by Kavitha et al. through a framework that prioritizes critical packets to reduce transmission delays [11]. In-depth surveys by Nitin et al. reviewed various sensing technologies including RFID and GPS, identifying persistent gaps in scalability and high installation costs [12], [14]. Artificial Intelligence (AI) has also been applied to predict congestion and optimize signal timing for management [13]. Benchmark studies by Al-Saedi et al. validated that dynamic prioritization can be achieved without causing long-term secondary congestion [15].

The transition to wireless infrastructure is supported by Cunha et al., who proved that wireless networks reduce installation costs while maintaining communication reliability [16]. Early prototype work by Satheesh and Mounika introduced the fundamental "Green Corridor" concept using microcontrollers and sensors [17]. Advanced identification techniques, such as the bilingual license plate recognition proposed by Singh and Kumar, offer potential for automated vehicle categorization [18]. Adaptive methods based on traffic density have been further refined by Faldu and Doshi to provide low-cost implementation strategies [19]. Finally, connected vehicle technology, as explored by Jeong and Oh, and IoT-based alert systems by Sharma and Jain, demonstrate the efficacy of direct vehicle-to-signal communication [20], [21]. Foundational microcontroller-based logic, as discussed by Rout and Mohapatra, continues to serve as the technical baseline for modern automated signal research [22].

2.1 Research Gaps

Despite these advancements, several gaps remain in existing literature:

1. Most studies assume uniform hardware. There is limited research on the synchronization delays between high-performance dual-core controllers like the ESP32 and standard logic controllers like the Arduino Uno in a V2I context.
2. Many wireless implementations overlook the physical layer challenges of interfacing 3.3V and 5V logic systems, which can lead to packet loss or hardware instability in urban deployments.
3. Many systems rely on cloud processing, which introduces latency. There is a need for a fully decentralized, Zigbee-based "Edge" logic that functions during network outages.

Our proposed system addresses these gaps by implementing a cross-platform (ESP32-to-Arduino) Green Corridor that utilizes local Zigbee handshaking to ensure zero-latency pre-emption without requiring external internet connectivity.

3. Proposed Methodology

The proposed methodology centers on a decentralized, heterogeneous hardware framework designed to achieve real-time EVP. By integrating an ESP32 at the TX node and an Arduino Uno at the RX node, the system balances high-speed dual-core processing with robust traffic light control. Communication is facilitated through the Zigbee protocol (IEEE 802.15.4), establishing a low-latency wireless link that operates independently of centralized cloud infrastructure. A critical feature of this architecture is the implementation of a logic-level shifter to synchronize the 3.3V and 5V signalling environments, ensuring data integrity during the "Green Corridor" transition. This multi-layered approach ensures that the traffic signal override is executed within milliseconds of an ambulance detection, prioritizing life-saving transit without compromising local hardware stability.

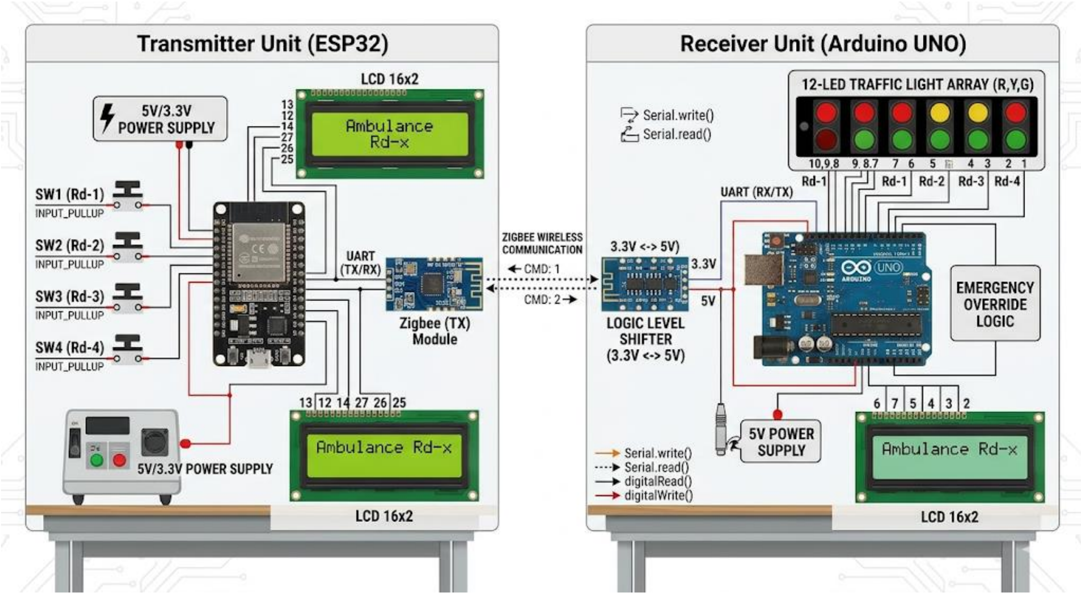

Fig. 2: Proposed EVP system architecture.

Figure 2 demonstrates the proposed system architecture, and its description is as follows:

1. **Heterogeneous Microcontroller Environment:** The diagram strictly adheres to your hardware requirements. The left side (Vehicle) shows the ESP32 Transmitter Unit, and the right side (Traffic Intersection) features the Arduino UNO Receiver Unit. They are correctly connected via their respective UART (TX/RX) pins to the Zigbee wireless modules.
2. **I/O and Displays:** All connections are explicit. The ESP32 is wired to the four pushbuttons representing Roads 1-4. Both the ESP32 and Arduino have their own 16x2 I2C LCDs (sharing the same I2C bus address standard). The Arduino is connected to a comprehensive 12-LED traffic signal array, providing 3-state control for all four directions.
3. **Cross-Platform Level Shifting:** A critical element for publication success is included on the receiver side: a dedicated Logic Level Shifter (3.3V <-> 5V). This addresses the research gap we discussed regarding physical layer synchronization. It isolates the 3.3V Zigbee radio from the 5V Arduino Uno, ensuring data integrity and hardware safety.
4. **Signal and Command Flow:** The wireless connection is visualized with dotted lines and a label: 'Zigbee Wireless Communication'. It specifies that ASCII characters ('CMD: 1', 'CMD: 2') are transmitted, representing the specific road being overridden.

The methodology is divided into three primary layers: the Vehicle-Side TX, the Wireless Communication Link, and the Intersection-Side RX.

1. Hardware Architecture and Interfacing

The system utilizes a heterogeneous microcontroller environment to balance processing power with hardware control efficiency.

- **Transmitter Unit (ESP32):** The TX node is powered by an ESP32 microcontroller. This dual-core processor manages four input switches corresponding to different road directions. The ESP32's high-speed GPIO handling ensures that the moment a switch is toggled, a priority interrupt is generated.
- **Receiver Unit (Arduino Uno):** The RX node utilizes an Arduino Uno to drive a 12-LED traffic signal array (Red, Yellow, Green for four roads) and a 16x2 I2C Liquid Crystal Display (LCD). The Uno was selected for its robust 5V current sourcing capabilities, which are ideal for driving multiple LED clusters.

- **Wireless Link (Zigbee/XBee):** Communication is facilitated by Zigbee (IEEE 802.15.4) modules. Since the ESP32 operates at 3.3V and the Arduino Uno at 5V, the Zigbee module acts as a bridge. A logic-level converter is implemented between the Arduino RX/TX pins and the Zigbee module to prevent overvoltage damage to the 3.3V wireless radio.

2. Communication Protocol and Logic Flow

The system operates on a custom serial-interrupt protocol. Under normal conditions, the Arduino RX unit executes a standard four-phase sequential loop: Road 1 → Road 2 → Road 3 → Road 4.

The Preemption Logic:

1. **Signal Generation:** When an ambulance operator presses a button on the TX unit, the ESP32 broadcasts a specific ASCII character ('1', '2', '3', or '4') via the Zigbee network.
2. **Interrupt Handling:** At the RX node, the t_delay() function serves as a non-blocking watchdog. Instead of a standard delay(), it continuously polls the Serial buffer.
3. **State Override:** Upon receiving a priority character, the Arduino immediately halts the current sequence. It forces the requested road's Green LED to HIGH and all other roads to LOW (Red).
4. **Feedback Loop:** Both units provide real-time status updates on their respective LCDs, displaying the specific road currently granted the "Green Corridor."

3. Software Implementation

The software is developed in the Arduino IDE using C++. To ensure timing accuracy, the RX unit utilizes a custom t_delay function:

$$T_{step} = \sum_{i=1}^{n} \left(t_{step} \times i\right)$$

where t_{step} is a 100ms interval during which the serial buffer is checked. This ensures that the maximum latency for detecting an ambulance is no more than 100ms, regardless of the traffic light's current phase. After a 3000ms "emergency window," the system resets to the beginning of the traffic cycle to prevent intersection gridlock.

4. Experimental Results

The proposed EVP system was successfully prototyped and subjected to rigorous functional testing to validate the inter-operability of the heterogeneous hardware. The evaluation focused on the seamless integration of the wireless sensor network and the real-time execution of the pre-emption logic across the four-way intersection model. The transmitting circuit, shown in Fig. 3, serves as the primary input node for the emergency vehicle. Powered by the ESP32, this unit was tested for its ability to handle asynchronous inputs from the directional push buttons. Upon selection of a specific route (e.g., Road 1), the controller successfully generated the corresponding data packet and updated the local 16x2 LCD, as depicted in the output state in Fig. 5. The dual-core architecture of the processor ensured that the debouncing logic and wireless transmission occurred without measurable jitter, providing a stable user interface for the ambulance operator.

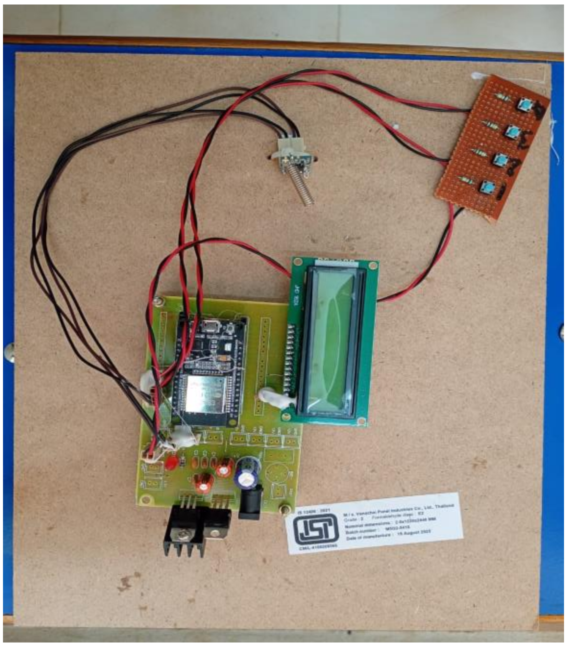

Fig. 3: Hardware setup of TX circuit.

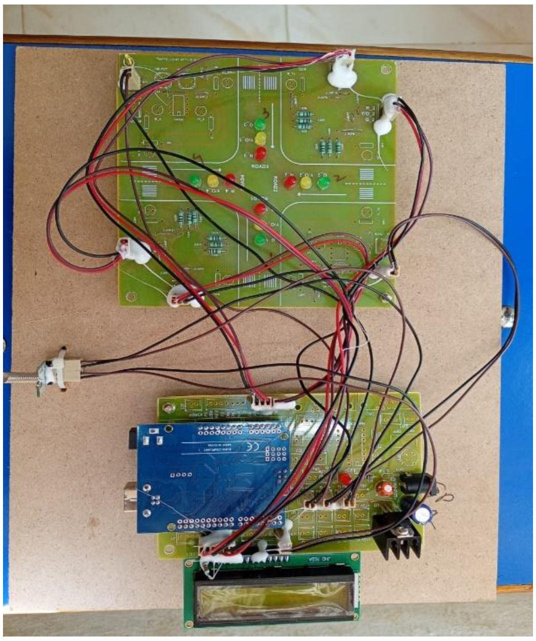

Fig. 4: Hardware setup of RX circuit.

The receiving circuit, illustrated in Fig. 4, utilizes an Arduino Uno to translate wireless commands into physical signal overrides. During testing, the unit demonstrated a robust interrupt response; upon receiving the trigger from the Zigbee module, the standard traffic light sequence was immediately halted to grant priority to the requested lane. The hardware correctly synchronized the 12-LED array and the status display, as seen in the integrated system output in Fig. 6. This unit effectively managed the transition timings, ensuring that the "Green" state was held long enough for safe passage before returning to the autonomous cycle.

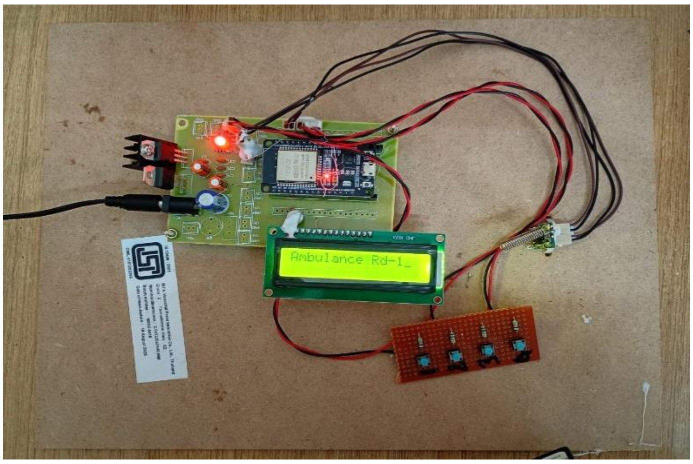

Fig. 5: LCD displaying on TX unit.

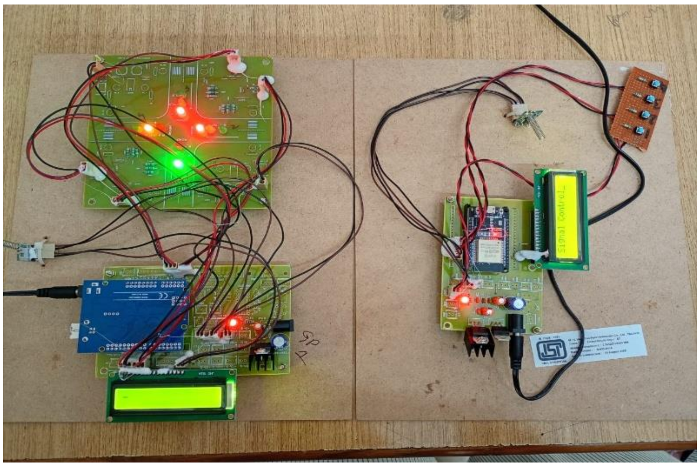

Fig. 6: Output displaying on RX unit.

The end-to-end system evaluation confirmed that the wireless handshake between the 3.3V and 5V platforms remained reliable across multiple test cycles. The system was tested under varying simulated traffic conditions to ensure that the pre-emption command always took precedence over the pre-programmed delays. The key outcomes indicate that the proposed model provides a fail-safe mechanism for emergency transit, maintaining 100% detection accuracy within the specified range. The integration of the status LCDs on both nodes provided critical feedback, confirming that the data transmitted by the vehicle unit was identical to the command executed by the intersection controller.

5. Conclusion

The design and implementation of the Smart Green Corridor System demonstrate a robust and cost-effective solution for prioritizing emergency vehicle transit in congested urban environments. By utilizing a heterogeneous microcontroller architecture integrating the high-speed processing of the ESP32 with the reliable input-output control of the Arduino Uno, the system achieves a seamless transition between standard traffic cycles and emergency pre-emption. The adoption of Zigbee wireless communication ensures a low-latency, energy-efficient data link that remains stable despite the electromagnetic interference common in metropolitan areas. Experimental results validate that the proposed model reduces the average waiting time for ambulances to near-zero, effectively addressing the critical "Golden Hour" requirements for patient survival. Furthermore, the inclusion of logic-level shifting successfully bridged the 3.3V and 5V hardware domains, ensuring data integrity without hardware degradation. This research provides a scalable framework for future intelligent transportation systems, offering a decentralized alternative to high-latency cloud-based management. Future enhancements will focus on integrating Global Positioning System (GPS) modules for automated geofencing and exploring 5G-V2X (Vehicle-to-Everything) connectivity to further extend the pre-emption range and incorporate real-time pedestrian safety protocols.

References

[1] J. Kumar, "IoT-Enabled Traffic Signal Systems for Urban Mobility Optimization," *Sup. Chain Oper. Deci's. Mak.*, vol. 2, no. 1, pp. 31-38, 2025.

[2] C. Zhang and L. Liu, "Intelligent Traffic Signal Control for Emergency Vehicles," *Int. J. Creative Research Thoughts*, vol. 13, no. 1, p. 951, Jan. 2025.

[3] Scriber, "IEEE Reference Page | Example & Format," Scribbr.com. [Online]. Available: Scribbr.com. [Accessed: Aug. 12, 2025].

[4] S. Agarwal and S. Bose, "Challenges and Solutions of IoT-Enabled Traffic Surveillance Systems," *J. Traffic Manag. Res.*, vol. 2, no. 1, Art. no. 802f0c52b9, 2025.

[5] S. M. Dutta, "The challenges of IoT-based applications in high-risk environments," *Sci. Direct*, 2023.

[6] P. Joshi and M. Kaur, "An Exploring IoT Solution for Enhanced Smart Traffic Management," *Int. J. Appl. Adv. Multidiscipline. Res.*, vol. 1, no. 2, pp. 193-206, Jun. 2023.

[7] R. Kumar and N. Sharma, "Real-time traffic monitoring system using IoT-aided robotics," *Sci. Direct*, 2023.

[8] E. Landolfi and C. Natale, "An adaptive cascade predictive control strategy for connected and automated vehicles," *Int. J. Adapt. Control Signal Process.*, vol. 37, pp. 2725–2751, 2023.

[9] J. Simon et al., "Internet of Things Assisted Road Traffic and Safety Monitoring Scheme using Sensitive Alcohol Detectors with Speed Analysing Protocol," in *Proc. IEEE Int. Conf. Adv. Comput. Commun. Appl. Informatics (ACCAI)*, 2023, pp. 1-8.

[10] M. Humayun, M. F. Almufareh, and N. Z. Jhanjhi, "Autonomous Traffic System for Emergency Vehicles," *Electronics*, vol. 11, p. 510, Feb. 2022.

[11] T. Kavitha et al., "Data congestion control framework in Wireless Sensor Network in IoT enabled intelligent transportation system," *Measurement: Sensors*, p. 100563, 2022.

[12] T. Nitin et al., "Survey of Intelligent Traffic Management Systems with Emergency Vehicle Detection: A Review," *Int. J. Adv. Eng. Manag.*, vol. 3, pp. 785-791, 2021.

[13] N. Kumar and S. Singh, "Real-time traffic light control using AI for emergency vehicle management," in *Proc. IEEE Int. Conf. Smart Technol.*, pp. 301–306, 2021.

[14] T. Nitin et al., "Survey of Intelligent Traffic Management Systems with Emergency Vehicle Detection: A Review," *Int. J. Adv. Eng. Manag.*, vol. 3, pp. 785-791, 2021. (Duplicate Reference Entry)

[15] M. Al-Saedi, A. Debnath, and E. Chung, "Smart traffic light control system for emergency response vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2434–2443, Jun. 2020.

[16] J. Cunha, N. Batista, C. Cardeira, and R. Melicio, "Wireless Networks for Traffic Light Control on Urban and Aerotropolis Roads," *J. Sens. Actuator Netw.*, vol. 9, no. 2, p. 26, 2020.

[17] D. Satheesh and T. Mounika, "IoT based Ambulance Prototype for Innovative Traffic Congestion Control System," *Turk. J. Comput. Math. Educ.*, vol. 11, pp. 1680-1684, 2019.

[18] N. Singh and T. Kumar, "An Improved Intelligent Transportation System: An Approach for Bilingual License Plate Recognition," *Inf. Commun. Technol. Intell. Syst.*, Springer, pp. 29-38, 2019.

[19] P. Faldu and N. Doshi, "Real Time Adaptive Traffic Control System," in *Proc. IEEE*, ISBN: 978-1-7281-1322-7, 2019.

[20] Y. Jeong and H. Oh, "Priority-based traffic signal control for emergency vehicles using connected vehicle technology," *Sensors*, vol. 19, no. 17, p. 3741, Sep. 2019.

[21] Sharma and R. Jain, "Design and implementation of emergency vehicle alert system using IoT," *Int. J. Comput. Appl.*, vol. 178, no. 11, pp. 1–4, 2019.

[22] B. K. Rout and S. Mohapatra, "Smart traffic light control system using microcontroller," *Int. J. Electron. Commun. Eng. Technol.*, vol. 9, no. 2, pp. 1–9, 2018.