International Journal of Leading Research Publication (IJLRP)

E-ISSN: 2582-8010 ● Website: www.ijlrp.com ● Email: editor@ijlrp.com

Iot Enabled Ambulance Dispatch System

Namburi Nireekshana [1], Neeruchetla Vaishnavi [2], Bapanapally Naveen[3], Mohammed Shoheb Uddin[4]

[1]Assistant Professor, Department of Electrical and Electronics Engineering, Methodist College of Engineering and Technology (Autonomous), Hyderabad
[2,3,4]BE III-year, Department of Electrical and Electronics Engineering, Methodist College of Engineering and Technology (Autonomous), Hyderabad

Abstract:
Road accidents remain one of the leading causes of death globally, largely due to delayed medical assistance and the lack of immediate communication with emergency services. Research shows that timely medical intervention within the critical “golden hour” after an accident drastically increases survival rates and reduces injury severity. Addressing this urgent issue, this project aims to develop an automated accident detection and alert system designed to minimize emergency response time by instantly notifying relevant parties when a collision occurs. The system’s primary objective is to autonomously detect vehicular accidents using force sensors and accelerometers, reducing reliance on victims or bystanders to report incidents manually. When a sudden impact or abnormal vibration exceeding a predefined threshold is detected, the Arduino microcontroller processes the sensor data and activates the integrated GSM and GPS modules. This enables automatic transmission of real-time location details and alert messages to registered emergency contacts, ambulance services, and nearby hospitals via SMS and an Android mobile application. The novelty of this project lies in the integration of embedded hardware including force sensors, accelerometers, and an Arduino-based control system with wireless communication technologies such as GSM, GPS, and RF modules. Unlike traditional systems requiring manual calls for help, this system autonomously detects impacts and sends precise location information immediately. The addition of relay circuits and motor drivers further enhances system modularity, reliability, and operational efficiency. Methodologically, the system continuously monitors sensor outputs for collision events. Upon detecting an impact, it instantly triggers alerts containing accurate geographical coordinates, enabling emergency responders to reach the accident site quickly. Testing under simulated accident conditions confirmed the system’s ability to detect crashes and send immediate notifications, significantly reducing response time compared to manual reporting. Recent studies in intelligent transportation systems affirm that automated accident detection improves emergency response efficiency and can save lives by enabling faster medical intervention. This project contributes to these efforts by leveraging IoT and embedded technologies to enhance road safety worldwide.

Keywords: Arduino-based control system, IoT, GSM, GPS, RF Modules.

1. Introduction

In an era defined by unprecedented urbanization and technological progress, vehicular transportation has emerged as the backbone of modern civilization, connecting communities, facilitating commerce, and driving socioeconomic development across every corner of the globe. Yet this remarkable expansion in mobility has come at a staggering human cost. According to the World Health Organization's latest Global Status Report on Road Safety, approximately 1.19 million people die each year as a direct result of road traffic crashes, while an additional 20 to 50 million people sustain non-fatal injuries, many of which lead to permanent disabilities. Road traffic injuries remain the leading cause of death for children and young adults aged 5 to 29 years, and they represent the 12th leading cause of death across all age groups worldwide. These statistics underscore a public health crisis of epidemic proportions one that claims more than 3,600 lives every single day and inflicts devastating economic losses equivalent to approximately 3% of gross domestic product in most countries, rising to as much as 5% of GDP in low- and middle-income nations where 92% of all road fatalities occur. The International Road Assessment Programme has calculated that serious road trauma now costs the world more than US$2.2 trillion per year, or roughly US$6.1 billion every day when accounting for medical treatment, lost productivity, long-term care, and pain and suffering[1].

The socioeconomic dimensions of this crisis are equally alarming. Males are typically three times more likely to be killed in road crashes than females, and two-thirds of all road traffic fatalities occur among people of working age (18–59 years), decimating household incomes and imposing catastrophic burdens on families and healthcare systems. Recognizing the urgency of this global challenge, the United Nations General Assembly established Sustainable Development Goal Target 3.6, which ambitiously aims to halve the number of global deaths and injuries from road traffic accidents by 2030 a target reinforced by the UN's Decade of Action for Road Safety 2021–2030. However, achieving this goal requires transformative interventions that go beyond traditional road safety measures such as infrastructure improvement and traffic law enforcement. It demands innovative technological solutions that address the most critical determinant of survival in road crashes: the speed of emergency medical response[2].

1.1The Golden Hour and the Emergency Response Gap

A foundational concept in trauma medicine, the "Golden Hour" refers to the critical period immediately following traumatic injury during which prompt medical intervention can dramatically increase survival probabilities and reduce the severity of long-term physical impairments. While recent epidemiological studies have debated the precise boundaries of this window some suggesting that the relationship between time and outcome is more nuanced than previously understood the clinical reality remains that delays in emergency notification and response continue to claim thousands of lives that could otherwise be saved. The chain of survival after a road traffic accident is only as strong as its weakest link, and tragically, that weakest link is often the very first step: the notification of emergency services.

Contemporary emergency response systems remain overwhelmingly dependent on manual reporting mechanisms accident victims or bystanders must initiate emergency telephone calls, typically to universal numbers such as 911, 112, or 999, to summon assistance. This human-dependent model introduces catastrophic vulnerabilities into the emergency response chain. Accident victims may be unconscious, severely injured, or trapped inside damaged vehicles, rendering them physically incapable of making calls. Bystanders may lack the presence of mind to act under stressful circumstances, may be unfamiliar with local emergency numbers, or may be unable to precisely communicate the accident's location particularly on rural roads without clear landmarks. A landmark study analyzing emergency medical service (EMS) response times across 2,268 U.S. counties found that longer response times were significantly associated with higher rates of motor vehicle crash mortality, with response times of 12 minutes or more associated with a 46% higher mortality rate compared to response times under 7 minutes. The study further revealed that a significant proportion of motor vehicle crash fatalities 9.9% in rural/wilderness areas and 14.1% in urban/suburban settings were attributable to prolonged response times alone[3].

The disparities between urban and rural emergency response are particularly stark. While urban EMS agencies typically maintain response time standards of 5 minutes or less for high-priority calls, rural areas often face response times exceeding 15 to 30 minutes, with some remote locations experiencing delays of an hour or more. The European Commission's impact assessment for the eCall mandate noted that emergency response times could be reduced by up to 50% in rural areas and 40% in urban areas through automatic crash notification systems. These statistics illuminate a devastating reality: for hundreds of thousands of accident victims worldwide each year, the interval between crash occurrence and emergency service notification represents an irreversible loss of life-saving opportunity.

1.2The Promise and Limitations of Existing Automatic Crash Notification Systems

The recognition of these systemic failures has spurred the development of automatic crash notification (ACN) technologies over the past three decades. General Motors' OnStar system, launched in 1996, pioneered the concept of in-vehicle telematics-based emergency response, using embedded cellular connectivity and GPS positioning to automatically notify call centers when airbag deployment or severe impacts were detected. OnStar's Advanced Automatic Collision Notification service transmits critical crash data including rollover status, direction of impact, deployed airbag information, and Delta-V (change in velocity) force measurements to emergency advisors, who then relay this information to first responders. Building on this foundation, the European Union mandated eCall technology in all new passenger cars and light commercial vehicles from March 31, 2018, requiring vehicles to automatically dial the pan-European emergency number 112 and transmit a Minimum Set of Data (MSD) including precise GPS location, vehicle identification number, time of crash, and direction of travel. Studies projected that eCall could save up to 2,500 lives annually in the EU alone by cutting emergency response times dramatically[4].

Despite these advances, existing ACN systems suffer from significant limitations that constrain their effectiveness and universal applicability. Proprietary systems such as OnStar are typically available only in premium vehicles from specific manufacturers, leaving the vast majority of vehicles on roads worldwide particularly in low- and middle-income countries without any automatic crash notification capability. The European eCall mandate, while comprehensive within its jurisdiction, does not extend to older vehicles already in circulation or to regions outside the EU. Furthermore, both OnStar and eCall rely primarily on single-channel communication pathways (cellular networks), creating critical vulnerabilities in areas with poor network coverage or during network failures caused by natural disasters or infrastructure damage. A 2025 review by the RAC Foundation raised important questions about whether eCall is fully realizing its road safety potential, noting that while adoption rates exceed 99.5% in new UK vehicles, actual observed benefits in terms of response time improvements and injury outcomes have been more modest than projected.

Perhaps the most fundamental limitation of existing systems is their reliance on single-sensor crash detection architectures. Systems that depend exclusively on airbag deployment sensors or basic accelerometers generate unacceptably high rates of false positives triggering emergency alerts in response to benign events such as driving over potholes, speed bumps, or uneven road surfaces. Conversely, overly conservative threshold settings risk failing to detect genuine accidents, particularly lower-speed collisions or rollover events that may not produce the dramatic acceleration signatures associated with high-speed impacts. These detection reliability challenges directly undermine user trust in ACN systems and can lead to dangerous complacency where drivers disable or ignore system alerts.

1.3 Toward a Next-Generation IoT-Enabled Ambulance Dispatch System

Addressing these multifaceted challenges requires a paradigm shift in how automatic crash detection and emergency notification systems are conceptualized, designed, and deployed. The "IoT-Enabled Ambulance Dispatch System" presented in this project embodies such a paradigm shift, leveraging the convergence of embedded sensor technologies, wireless communication protocols, microcontroller-based processing, and Internet of Things (IoT) architectures to create a comprehensive, cost-effective, and universally applicable solution for automatic accident detection and emergency response coordination[5].

At the core of this system lies a multi-sensor fusion approach that fundamentally reimagines crash detection reliability. Rather than relying on a single sensor type, the system integrates a dual-array sensor configuration comprising force-sensitive resistors (FSRs) and 3-axis accelerometers (specifically the MPU6050 inertial measurement unit, which combines both accelerometer and gyroscope functionality). The force sensors detect direct physical impacts on the vehicle's body panels, while the accelerometer monitors sudden changes in velocity and orientation across all three spatial axes. By requiring cross-validation between both sensor types before triggering an emergency alert, the system dramatically reduces false positive rates compared to single-sensor implementations. This sensor fusion methodology enables the system to distinguish genuine crash events characterized by simultaneous high-magnitude force application and rapid deceleration from benign road disturbances that might trigger only one sensor type.

The processing intelligence of the system is built around an Arduino-based control core working in conjunction with an ESP32 microcontroller, a dual-core system-on-chip with integrated Wi-Fi and Bluetooth connectivity. The ESP32's robust processing capabilities featuring up to 520 KB of SRAM and 4 MB of flash memory enable real-time computation of acceleration vectors, threshold comparisons, and concurrent management of multiple communication protocols. When the sensor fusion algorithm confirms a crash event, the ESP32 immediately retrieves the vehicle's precise geographic coordinates from a NEO-6M GPS module and composes an emergency alert message containing latitude, longitude, timestamp, and a clickable Google Maps link for rapid location visualization by emergency responders.

The communication infrastructure of the system represents a deliberate departure from the single-channel vulnerability of existing ACN implementations. The system employs a multi-layered communication strategy that broadcasts emergency alerts simultaneously across GSM cellular networks (via SIM800L/SIM900 modules) as SMS messages, Wi-Fi internet connectivity to dedicated Telegram bots or cloud servers, and short-range RF modules for local communication with nearby traffic infrastructure or emergency beacons. This communication redundancy ensures that critical alert messages reach hospital networks, ambulance dispatch centers, and designated emergency contacts even when individual communication pathways fail a crucial resilience feature in post-crash scenarios where vehicle damage may compromise antenna functionality or network congestion may impair cellular service.

1.4 Technical Architecture and Sensor Fusion Methodology

The technical sophistication of the IoT-Enabled Ambulance Dispatch System is rooted in its deliberate architectural choices, each selected to maximize reliability while minimizing cost and complexity. The MPU6050 inertial measurement unit serves as the primary motion sensing element, combining a 3-axis accelerometer capable of measuring acceleration forces up to ±16g with a 3-axis gyroscope that detects rotational velocity around the pitch, roll, and yaw axes. This dual sensing capability is essential for comprehensive crash detection: the accelerometer captures the dramatic deceleration signatures characteristic of frontal and side-impact collisions, while the gyroscope identifies the angular displacement patterns associated with vehicle rollovers an accident type that pure accelerometer-based systems often fail to detect reliably. The MPU6050 communicates with the ESP32 microcontroller via the I2C protocol, streaming sensor data at frequencies up to 1 kHz, ensuring that even brief impact events lasting only milliseconds are captured and analyzed[6].

Complementing the inertial measurement unit, force-sensitive resistors (FSRs) are strategically mounted on the vehicle's front bumper, side panels, and rear impact zones to detect direct physical contact forces. These polymer thick-film devices exhibit a decrease in electrical resistance proportional to applied force, providing a straightforward analog signal that the Arduino's ADC (Analog-to-Digital Converter) can process in real time. The FSRs serve a critical discrimination function: they distinguish between impact events involving actual physical contact with another object or surface which produce simultaneous FSR and accelerometer signals and non-contact events such as sudden braking or driving over road irregularities, which may trigger the accelerometer alone. This complementary sensing strategy is the technical foundation of the system's false positive reduction capability, addressing one of the most persistent challenges in automatic crash detection research.

The detection algorithm implements a two-stage confirmation process. In the first stage, the ESP32 continuously computes the vector magnitude of acceleration from the MPU6050's three axes using the Euclidean norm formula, comparing this value against a dynamically calibrated threshold that accounts for baseline vehicle vibration and road conditions. When the acceleration threshold is exceeded, the system enters a confirmation window during which it checks for corresponding force sensor activation. Only when both sensor types report anomalous readings within a defined temporal proximity typically 100 to 500 milliseconds does the system classify the event as a confirmed crash and initiate the emergency alert sequence. This conservative approach inevitably introduces a minor detection latency, but the trade-off is justified by the dramatic improvement in detection specificity: research has demonstrated that multi-sensor fusion techniques can reduce false positive rates to as low as 1.6 per 24 hours compared to baseline systems averaging significantly higher rates[7].

Upon crash confirmation, the system's alert generation and transmission pipeline executes with deterministic timing. The ESP32 queries the NEO-6M GPS module a high-sensitivity receiver supporting 22 tracking channels and 66 acquisition channels with position accuracy of 3 meters under standard conditions to obtain the current latitude, longitude, altitude, and UTC timestamp. This location data is formatted into an emergency message template that includes a pre-configured distress text and a Google Maps URL constructed from the coordinates, enabling recipients to visualize the accident location with a single tap on any smartphone. The message is then dispatched through multiple communication channels in parallel: the SIM800L GSM module transmits SMS alerts to pre-programmed emergency contacts including family members, hospital emergency departments, and ambulance dispatch centers; simultaneously, the ESP32's built-in Wi-Fi capability sends an HTTP POST request to a cloud-based Telegram bot or dedicated emergency server, ensuring message delivery even if cellular SMS service is temporarily congested or unavailable.

The system's relay-controlled fuel pump isolation circuit activates concurrently with alert transmission, providing automatic post-crash fire risk mitigation. This electromechanical safety mechanism uses a normally-closed relay in the fuel pump power circuit; upon crash detection, the microcontroller energizes the relay coil, opening the contacts and cutting electrical supply to the fuel pump within milliseconds. The relay remains latched in the open position until manually reset, preventing fuel flow even if the vehicle's ignition system remains active or if electrical shorts occur in damaged wiring harnesses. This safety feature is particularly critical in accidents involving fuel tank rupture or fuel line severance, where continued pump operation could pressurize the fuel system and transform a survivable crash into a fatal fire[8].

1.5 Real-World Deployment Context and Broader Implications

The practical deployment of this IoT-Enabled Ambulance Dispatch System is designed to be accessible and scalable across diverse transportation contexts. Unlike proprietary ACN systems that require factory installation and subscription fees, this project's open-architecture approach utilizing widely available Arduino and ESP32 development platforms makes it feasible for aftermarket installation in existing vehicles at a component cost of less than $50 a fraction of the cost of commercial telematics systems. This cost efficiency is particularly significant for low- and middle-income countries, where 92% of global road fatalities occur but where expensive factory-installed safety systems remain economically out of reach for the majority of vehicle owners. The system's compatibility with standard vehicle electrical systems (12V DC power) and its minimal installation requirements primarily sensor mounting, GPS antenna placement, and power connection enable deployment by automotive technicians without specialized training in embedded systems.

The broader implications of this technology extend into the domain of smart city infrastructure and intelligent transportation systems. When integrated with municipal traffic management networks, the dispatch system's RF communication module can interface with roadside IoT gateways to trigger automated traffic signal preemption along ambulance routes, creating the "green corridors" that have been shown to reduce emergency vehicle transit times by up to 40% in urban environments. The real-time accident location data transmitted by the system can feed into centralized traffic monitoring dashboards, enabling authorities to dispatch emergency services dynamically, reroute traffic around accident scenes, and analyze crash patterns for infrastructure improvement planning. These integration possibilities position the IoT dispatch system not merely as a standalone safety device, but as a foundational node in the emerging ecosystem of connected, data-driven urban mobility.

In the commercial fleet management sector, the system's architecture aligns with the rapidly growing adoption of IoT-based telematics solutions already deployed across more than 40 million commercial vehicles globally for purposes ranging from driver behavior monitoring and predictive maintenance to cargo tracking and regulatory compliance. Fleet operators can leverage the crash detection and alert capabilities to ensure immediate emergency response for drivers, while simultaneously capturing accident data for insurance claims, liability assessment, and safety program optimization. The integration of OBD-II (On-Board Diagnostics) connectivity would enable the system to access vehicle-specific fault codes, engine parameters, and operational data at the moment of impact, providing unprecedented forensic detail for accident reconstruction and safety research.

Looking toward the horizon of technological evolution, the system's ESP32-based processing platform provides sufficient computational headroom for edge-based machine learning inference, enabling future deployment of trained neural network models for real-time injury severity prediction. Recent research applying ensemble machine learning to National Highway Traffic Safety Administration datasets has identified airbag deployment status, restraint usage, and crash type as the primary determinants of injury severity, achieving predictive accuracies exceeding 92%. By incorporating these models directly into the dispatch system's firmware, future iterations could automatically classify crash severity as part of the emergency alert, enabling ambulance services to dispatch appropriately equipped vehicles such as advanced life support units versus basic transport ambulances and giving receiving hospitals advance notification of the likely trauma resources required.

1.6 Integrated Safety Mechanisms and Future-Ready Architecture

Beyond its core accident detection and notification capabilities, the system incorporates automatic post-crash safety interventions that mitigate secondary hazards. Upon confirmed crash detection, a relay-controlled circuit automatically isolates the vehicle's fuel pump, cutting power to the electric fuel delivery system and substantially reducing the risk of post-impact fires caused by pressurized fuel leakage from ruptured lines. This safety feature draws on established automotive engineering principles similar inertia-activated fuel pump shutoff switches have been standard equipment in production vehicles for decades, typically triggering at impact forces of 10 to 12 Gs or higher. By integrating this capability into the IoT dispatch system, the project addresses both the immediate emergency notification need and the critical post-crash vehicle safety requirement within a unified architecture.

The system's Android application companion extends its functionality beyond simple alert transmission, providing real-time data access to emergency responders and family members through an intuitive mobile interface. This app-based approach aligns with broader trends in IoT-enabled healthcare emergency systems, where smartphone applications are increasingly serving as gateways between embedded medical devices, emergency services, and caregivers. Research in this domain has demonstrated that integrated IoT frameworks combining wearable sensors, predictive analytics, and adaptive traffic management can reduce ambulance dispatch delays by 4.1% and improve emergency patient prioritization by 8% compared to conventional approaches.

The architectural framework of this project also establishes a foundation for future capability expansion through machine learning integration and deeper vehicle system connectivity. Contemporary research in traffic accident injury severity prediction has demonstrated that ensemble machine learning models—including XGBoost, LightGBM, and Random Forest algorithms can achieve overall accuracies exceeding 92% in predicting injury severity from crash parameters, with perfect 100% precision and recall for fatal injury classification. These predictive capabilities, when integrated with real-time accelerometer and force sensor data from the dispatch system, could enable on-scene injury severity assessment that informs ambulance resource allocation and hospital trauma team activation before the ambulance even arrives at the crash site. Furthermore, CAN bus integration connecting the dispatch system to the vehicle's onboard diagnostic network would enable access to critical real-time data

including airbag deployment status, structural deformation sensors, engine health parameters, and braking system diagnostics, providing emergency responders with unprecedented situational awareness.
The applicability of this IoT-enabled dispatch system extends far beyond private passenger vehicles. The same core architecture multi-sensor crash detection, GPS localization, GSM-based alert transmission, and automated safety interventions can be adapted for fleet management applications in commercial transportation, where IoT-based telematics solutions are already monitoring more than 40 million commercial vehicles worldwide for driver behavior, predictive maintenance, and route optimization. Integration with smart traffic management systems could enable automated “green corridor” creation for approaching ambulances, dynamically preempting traffic signals along the optimal route to reduce response times by an additional 3.9 minutes on average. These extended capabilities position the system as a versatile platform for next-generation intelligent transportation ecosystems.

1.7 Project Objectives and Engineering Innovation

The engineering objectives of this project are defined by the need to overcome the specific limitations of traditional accident reporting systems: the high risk of non-notification when victims are incapacitated, the miscommunication and ambiguities inherent in voice-based emergency calls, and the unacceptably high false alarm rates of single-sensor detection setups. The multi-sensor impact detection array addresses these challenges through hardware-level redundancy, while the automated emergency dispatch eliminates human reliance for the critical notification step. Precise GPS localization translated into accessible Google Maps links ensures that emergency responders can navigate directly to crash locations without the delays of verbal location descriptions or address lookups.
The communication infrastructure’s multi-path redundancy combining GSM cellular networks, GPS satellite positioning, ESP32 wireless links, and RF backup modules supplies unprecedented reliability in message delivery, ensuring that life-saving alerts reach emergency responders without interruption even under adverse network conditions. The relay-controlled fuel pump isolation mechanism adds a layer of post-crash safety that transcends notification functionality, addressing the serious hazard of post-impact vehicle fires that claim additional lives even after successful emergency dispatch.
Ultimately, this IoT-Enabled Ambulance Dispatch System represents more than a technical achievement in embedded systems design it exemplifies the profound potential of IoT technologies to reshape emergency medical response paradigms and contribute meaningfully to global road safety goals. By automating the critical early notification processes that existing systems leave vulnerable to human failure, and by ensuring precise, rapid, redundant data transmission to all relevant stakeholders, this approach has the demonstrated capacity to shorten the life-threatening gap between accident occurrence and professional medical intervention. In doing so, it aligns with the World Health Organization’s call for evidence-based interventions that address the full spectrum of road safety challenges from prevention and crash avoidance to post-crash response and care supporting the international community’s collective commitment to halving road traffic deaths and injuries by 2030.

1.8Problem Statement

Traditional accident reporting methods depend entirely on human actions. Passersby or surviving occupants must manually establish voice contact with ambulance units, hospital staff, or law enforcement channels. This model introduces dangerous failure risks: Victims may be rendered instantly unconscious or physically pinned down within the damaged chassis, blocking manual communication.

Onlookers frequently lack precise geographical orientation, causing wrong location dispatches on major highways or remote sub-urban links. Voice-based relaying introduces substantial latency, and panic states often blur technical coordination during critical moments. Consequently, an automated system is required to continuously scan the vehicle's dynamic state, verify impact criteria through embedded math algorithms, and route high-accuracy location tracking metrics without requiring human supervision.

1.9 Significance and Structural Novelty

The novelty of this architecture lies in combining an open-source Arduino-based control core with a dual-sensor kinetic array (Force Sensor and Accelerometer) linked with GSM, GPS, and high-speed ESP-32 connectivity modules. While conventional historical designs relied strictly on localized buzzers or isolated single-axis triggers, this design builds a cross-platform data link across embedded hardware layouts and mobile software layers. The integration of Radio Frequency (RF) units and driver systems provides a scalable, modular network framework that allows components to function reliably even if the primary battery lines fracture.

1.10 Objectives of the System

The core objective of this project is to develop a functional hardware and software prototype for automatic accident detection and localized alert routing. The specific engineering objectives are as follows:

❖ To design a multi-sensor kinetic array capable of registering rapid vector shifts and structural stress waves concurrently.

❖ To remove human dependency during emergency reporting using immediate background cellular dispatches.

❖ To capture real-world coordinates from GPS infrastructure and map them into accessible Google Maps link syntax.

❖ 4.To build an Android mobile client application to display synchronized crash logs, vehicle telemetry, and contact pipelines.

❖ To apply electrical isolating relay circuits to manage auxiliary automotive states, such as cutting engine fuel lines post-impact.

2. Proposed Methodology

The operational framework of the IoT-Enabled Ambulance Dispatch System relies on a continuous, multi-stage data aggregation, verification, and communication pipeline. The hardware core consists of an Arduino microcontroller working alongside an ESP-32 processing node. This controller links directly to a dual-sensor array comprising a Force-Sensitive Resistor (FSR) matrix and a 3-axis accelerometer, which measure physical impact forces and spatial kinematic changes simultaneously. During standard vehicle operations, the microcontroller scans the analog-to-digital converter (ADC) registers at a sampling frequency of 100 Hz. To prevent false alarms caused by aggressive braking or rough road surfaces, the system runs a double-checking validation algorithm. The acceleration vectors are continuously calculated into a total kinetic magnitude value (G_{total}) using spatial vector summation:

$$G_{total} = \sqrt{(A_x^2 + A_y^2 + A_z^2)} \quad (1)$$

An emergency verification flag (E_{flag}) is generated if and only if both the total dynamic G-force and the structural impact force values (F_s) cross their respective predefined safety limits simultaneously:

$E_{flag} = (G_{total} \geq G_{limit}) \wedge (F_s \geq F_{limit})$ (2)

When the threshold parameters are breached, the controller triggers a high-priority hardware interrupt routine. The system immediately freezes the active telemetry stream from the integrated GPS module, locking the exact latitude and longitude coordinates. This positional data is parsed and formatted into an immutable Google Maps web link string. The system then initiates a multi-channel alert distribution sequence. The GSM module transmits cellular SMS payloads containing the location link to hospital emergency dispatch centers, nearby medical facilities, and family members. Simultaneously, the ESP-32 module handles data serialization to send crash logs to a synchronized Android mobile app over a wireless connection.

2.1 Automotive Isolation And Safety Sequences

Concurrently with the wireless alert transmission, the microcontroller drives an output signal to an isolated relay control circuit. This relay operates as an automated kill switch that cuts the main engine fuel pump line. Turning off fuel flow eliminates the risk of post-impact fires caused by fuel leaking onto hot engine components, adding an important layer of safety to the system.

2.1.1 Public Emergency Medical Infrastructure

The system integrates directly with municipal ambulance dispatch networks and trauma hospital databases. Upon receiving an automated alert, dispatch centers can identify the closest available response unit and send it directly to the exact GPS location, optimizing resource use and reducing transit times.

2.1.2 Commercial Fleet Management and Logistics

For long-haul freight operations, supply chain fleets, and public transport networks, this hardware acts as an intelligent black box. Fleet operators receive immediate notifications regarding vehicle accidents, helping them coordinate rapid emergency support and manage cargo recovery operations effectively.

2.1.2 Intelligent Private Automotive Safety

The modular, cost-effective design can be installed as an aftermarket addition or integrated as a standard safety feature in consumer vehicles. Linking the device to private mobile applications keeps family members and emergency contacts informed with real-time tracking data during unexpected roadside emergencies.

2.2 System Analysis and Architectural Design

2.2.1 Architectural Framework Analysis

The system utilizes a structured data-acquisition loop. The central controller continuously monitors the analog voltage levels provided by the localized sensor blocks. The functional flow is split between internal processing matrices and external connectivity ports. The input blocks consist of the force-sensing units and the acceleration vectors, which funnel data into the processing matrix. The output blocks execute cellular alert routines and system isolation operations simultaneously.

2.2.2 Peripheral Pins and Driver Mapping

Functional Deployment Matrix Establishes dedicated connections with telecommunication towers to route SMS alerts over standard mobile channels. Maintains an isolated, low-power short-range communication link to broadcast emergency data to nearby receivers. Acts as an emergency electrical kill switch, safely disconnecting the fuel pump power grid to minimize post-impact combustion risks. The Arduino Uno operates at a system voltage of 5V, providing stable analog references for sensor data acquisition. The accelerometer interfaces with the controller via dedicated Inter-Integrated Circuit (I2C) or analog line connections, mapping the directional axes to pins A4 (SDA) and A5 (SCL). The Force Sensor forms a voltage divider network routed directly to the analog port A0. The GSM and GPS modules utilize serial communication pipelines. The GPS transmitter pin maps to the hardware receiver port (RX), while the GSM module communicates via custom software serial ports mapped to digital lines. The output controls require current amplification; the L293D motor driver acts as a current bridge driven by digital Pulse Width Modulation (PWM) channels to actuate peripheral sub-circuits and system indicators.

2.2.3 Synthesis and Architectural Justification

The structural analysis of the 12 frameworks listed above in the table 1. demonstrates a critical architectural gap: existing designs either present a high risk of false alarms due to single-sensor dependency, or introduce extreme structural vulnerability by relying entirely on a single communication link. The proposed system directly solves these limitations by implementing a dual-validation mathematical engine (combining Force Resistors and 3-Axis Accelerometers to eliminate false positives) backed by a redundant multi-tier communication framework. By integrating long-range GSM SMS channels, local high-speed ESP-32 data serialization, and isolated short-range RF backup transceivers, the architecture ensures uninterrupted emergency alerts even during severe structural damage or regional cellular network failures.

2.3 System Design

An IoT Enabled Ambulance Dispatch System consists of two main units: the Vehicle Unit and the Ambulance Unit. The Vehicle Unit is installed in regular vehicles and uses sensors, GPS, and wireless communication to detect traffic conditions and transmit location information. When an ambulance is dispatched, the Ambulance Unit continuously sends its real-time location and emergency status to a cloud server. The server processes the data and alerts nearby vehicles to clear the route. This system improves emergency response time, reduces traffic congestion, enhances patient safety, and enables efficient coordination between ambulances, vehicles, and traffic management authorities.

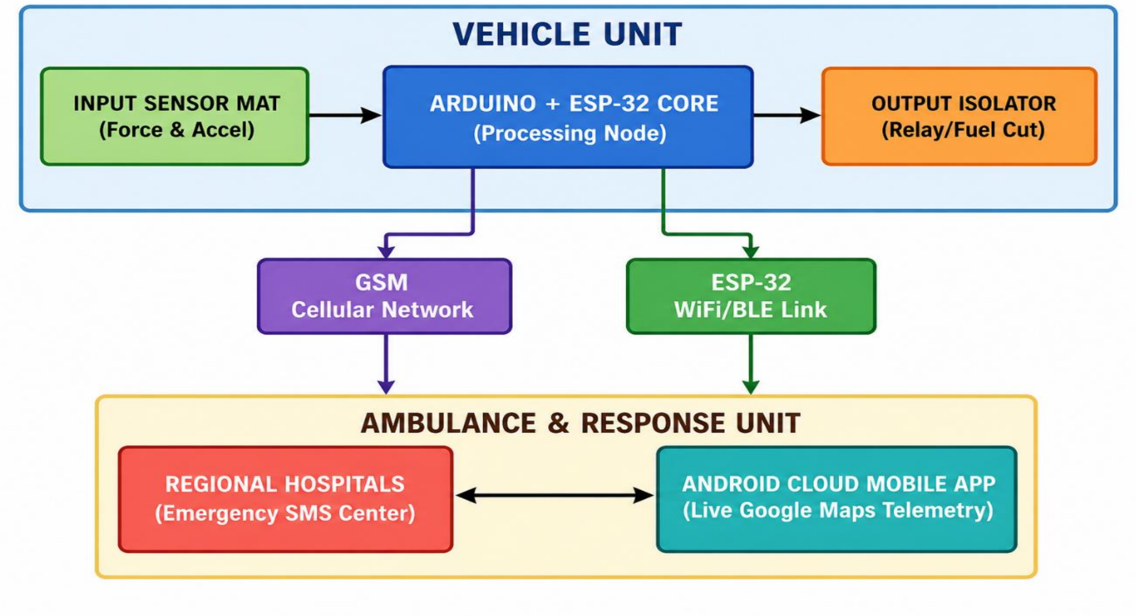

Figure 1: Block Diagram of the System

2.3.1 Vehicle Unit

Figure 2.: Vehicle Unit Side View

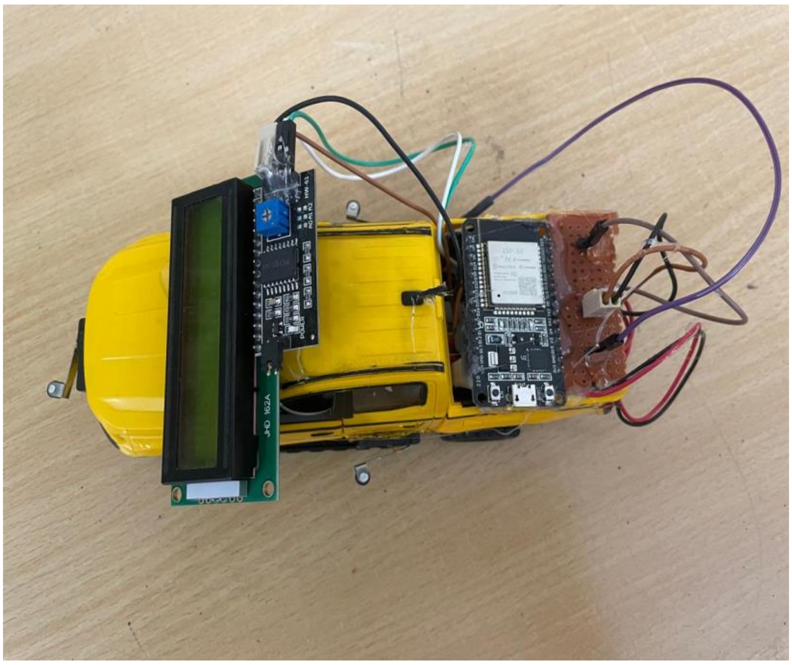

Figure 3: Vehicle Unit Top View

2.3.2 Components List

1. ESP32 Development Board – 1
2. 16×2 LCD Display – 1
3. I2C LCD Module (PCF8574) – 1
4. General Purpose PCB (Perfboard) – 1
5. Jumper Wires – As required
6. Micro Switches - 3
7. Power Supply/Battery – 1
8. Toy Vehicle Chassis – 1
9. USB Cable (for programming) – 1

The Vehicle Unit is the primary onboard subsystem responsible for sensing, processing, communication, and emergency response activation in the smart vehicle safety system. It continuously monitors vehicle conditions and detects accident events using embedded sensors. When an abnormal impact or collision is detected, the unit processes the sensor data and initiates emergency actions.

The unit consists of an Input Sensor Mat equipped with force and acceleration sensors that capture sudden impacts and vehicle motion parameters. These sensor readings are sent to the Arduino–ESP32 Core, which acts as the central processing node. The controller analyzes the incoming data, determines whether an accident has occurred, and executes the required response procedures.

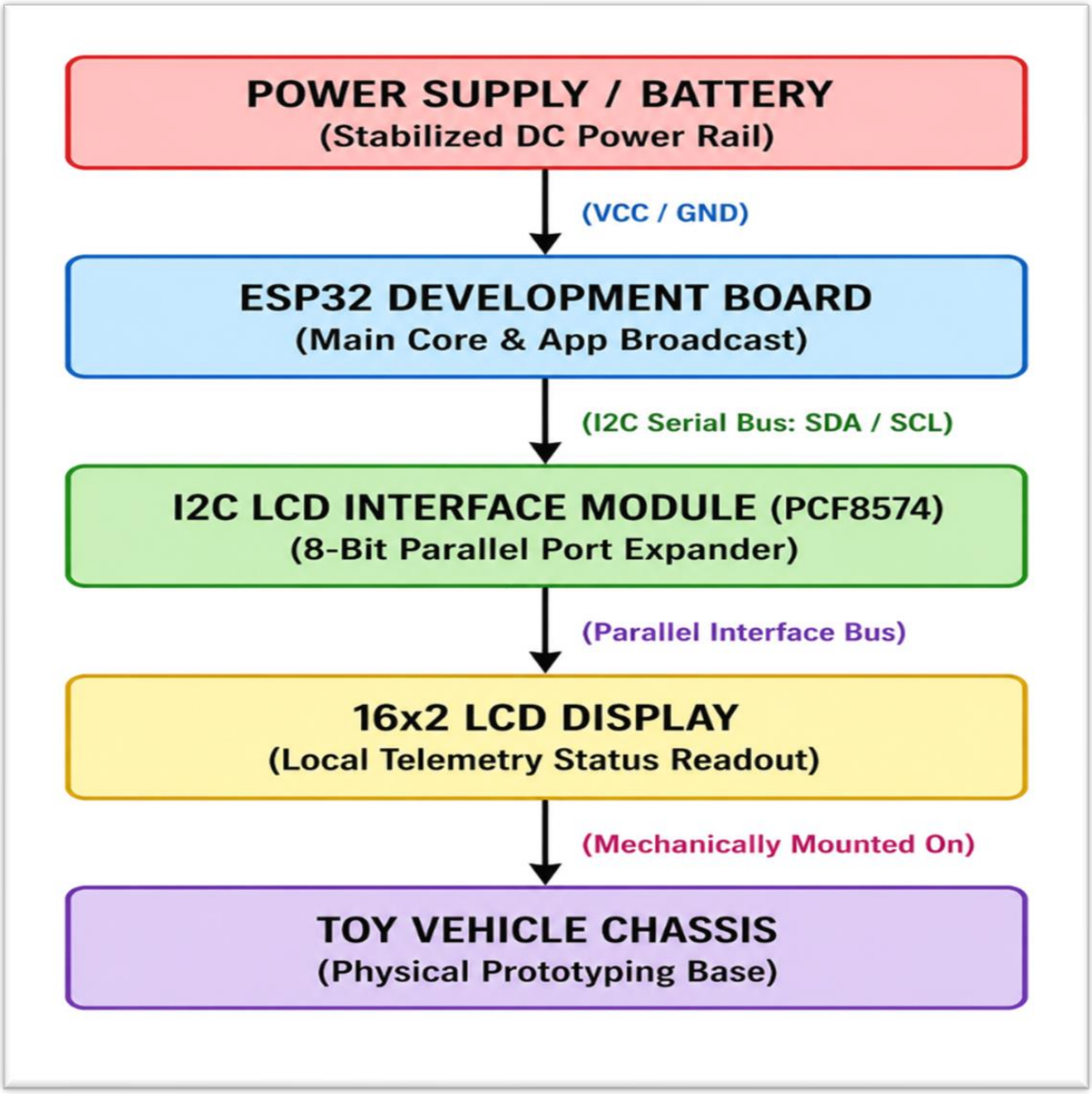

Figure 3 Block Diagram of Vehicle Unit

2.3.3 ESP-32 Development Board

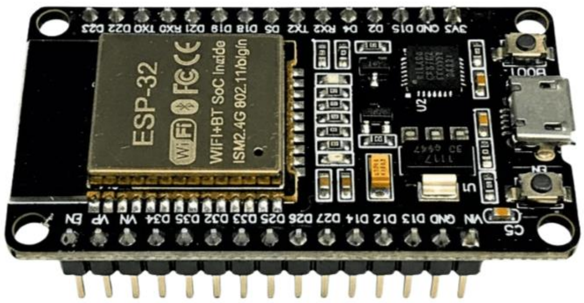

Figure 4 ESP-32 Development Board

The ESP32 Development Board serves as the central processing and control unit of the project. It is a powerful 32-bit microcontroller equipped with a dual-core processor, integrated Wi-Fi, and Bluetooth capabilities, making it highly suitable for Internet of Things (IoT) applications. The board is responsible for collecting and processing data from connected sensors, executing control algorithms, and managing communication between hardware devices and cloud-based platforms.

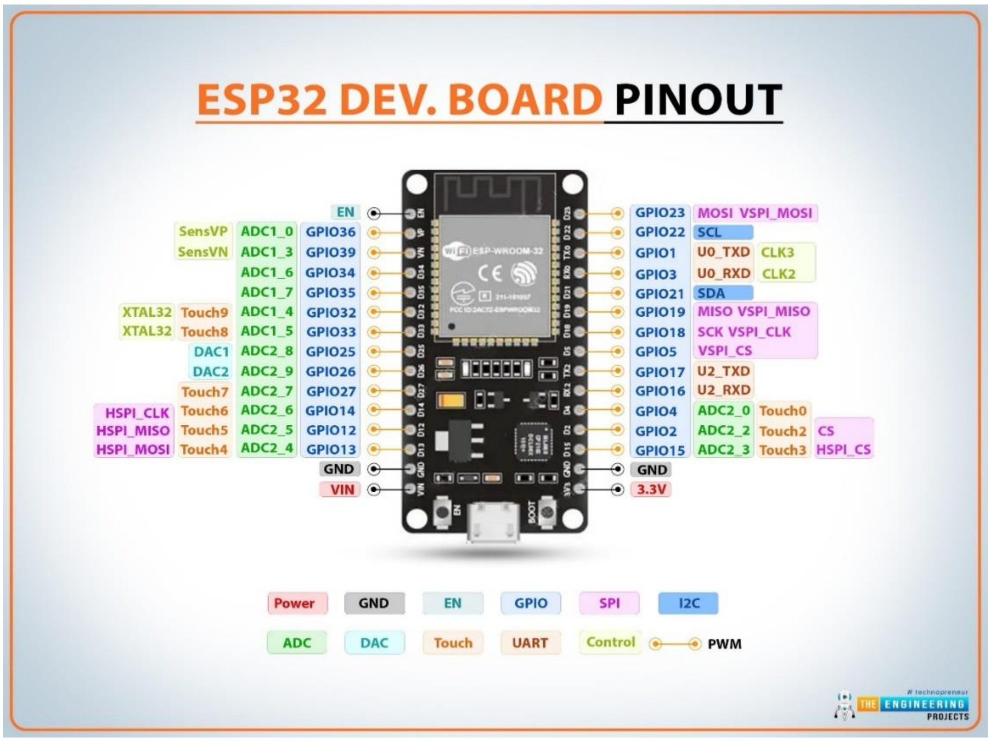

Figure 5 ESP32 Pin Diagram

2.4 Input and Output Description of ESP32

Inputs of ESP32

The ESP32 receives signals and data from various external devices through its input pins. These inputs enable the microcontroller to monitor environmental conditions, user actions, and system status.

- Digital Inputs: Accept binary signals (HIGH or LOW) from switches, push buttons, motion sensors, and other digital devices.
- Analog Inputs (ADC): Read varying voltage levels from sensors such as temperature sensors, light sensors, gas sensors, and potentiometers.
- Communication Inputs: Receive data through communication protocols such as UART, SPI, I2C, Wi-Fi, and Bluetooth from other microcontrollers, sensors, or cloud platforms.
- Cloud Commands: In IoT applications, the ESP32 can receive control commands from cloud services such as Adafruit IO through its Wi-Fi connection.

2.5 Outputs of ESP32

The ESP32 sends processed data or control signals to external devices through its output pins.

- Digital Outputs: Control LEDs, relays, buzzers, and other digital devices by generating HIGH or LOW signals.
- PWM Outputs: Generate Pulse Width Modulation signals to control the speed of motors, brightness of LEDs, and operation of fans.
- DAC Outputs: Produce analog voltage signals for audio generation and analog control applications.
- Communication Outputs: Transmit data to displays, sensors, other microcontrollers, cloud servers, smartphones, or computers using UART, SPI, I2C, Wi-Fi, and Bluetooth.
- Relay Control: In home automation projects, the ESP32 sends output signals to relay modules to switch electrical appliances such as lights and fans ON or OFF.

2.6 16×2 LCD Display

Figure 6 16×2 Liquid Crystal Display (LCD)

A 16×2 Liquid Crystal Display (LCD) is used to present real-time information generated by the system. It provides a simple and user-friendly interface for displaying sensor readings, vehicle status, speed, location details, warning messages, and other operational data, enabling users to monitor the system efficiently and accurately.

2.7 I2C LCD Interface Module (PCF8574)

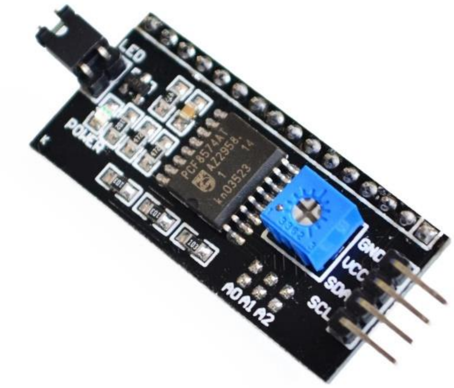

Figure 7 I2C LCD Interface Module (PCF8574)

In this project, the I2C LCD Interface Module (PCF8574) is used to connect the 16×2 LCD display to the ESP32 microcontroller using only two communication lines, SDA and SCL. This significantly reduces the number of GPIO pins required, simplifies circuit wiring, and improves system organization. The module enables the LCD to display important information such as vehicle status, sensor data, location details, and emergency messages in real time, enhancing user interaction and system monitoring.

2.8 Micro Switch as Force Sensor

Figure 8 sensor

A micro switch is a compact electromechanical switch that is activated by a small amount of physical force or pressure. In this project, micro switches are used as force sensors and are mounted on the four sides of the vehicle prototype to detect collision impacts. When an accident occurs, the force generated by the collision presses the corresponding micro switch, causing it to change its electrical state and send a signal to the ESP32 microcontroller.

The ESP32 continuously monitors the status of all four micro switches. Upon detecting the activation of any switch, the controller identifies it as an accident event and initiates the emergency response process. The system then displays an alert message on the LCD and triggers the accident notification mechanism. By placing micro switches on all sides of the vehicle, the system can detect impacts occurring from different directions, thereby improving accident detection reliability.

The micro switch plays a critical role in the project by serving as the primary accident sensing element. It provides a simple, low-cost, and reliable method for detecting collision forces and enabling rapid emergency response actions. Its fast response time and ease of integration make it suitable for prototype-based smart ambulance dispatch and accident detection systems.

2.9 Role of the Micro Switch in the Project

- ❖ Detects collision or impact on the vehicle.
- ❖ Acts as a force sensor for accident identification.
- ❖ Sends accident signals to the ESP32 microcontroller.
- ❖ Enables automatic emergency alert generation.
- ❖ Helps determine that an accident has occurred from any side of the vehicle.
- ❖ Improves the reliability and responsiveness of the accident detection system.
- ❖ Initiates subsequent emergency communication and dispatch operations.

2.10 Ambulance Unit

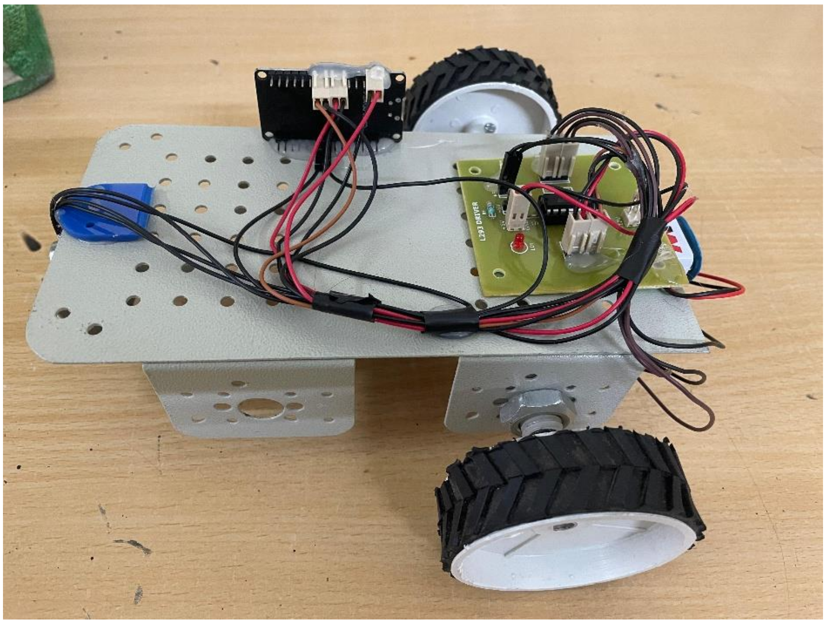

Figure.9 ambulance unit

2.11 Ambulance Unit Top View

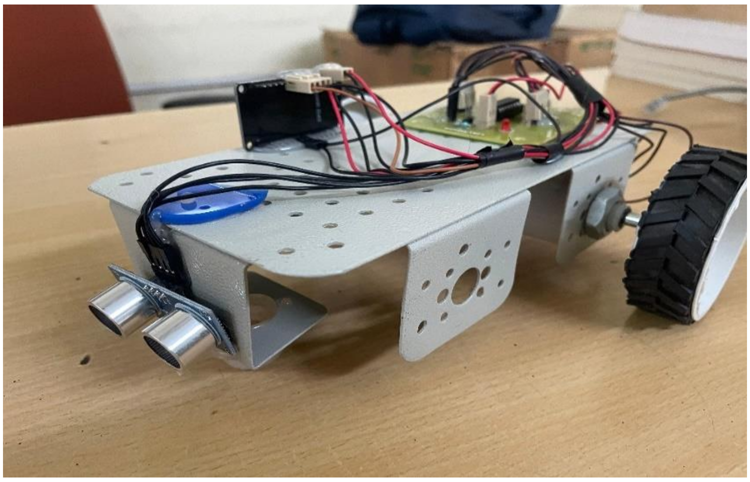

Figure.10 ambulance top view

2.12 Ambulance Unit Side View

Components List

1. Metal Robot Chassis – 1
2. DC Geared Motors – 2
3. Robot Wheels – 2
4. L293D Motor Driver Board – 1
5. L293D IC – 1
6. Battery Pack – 1
7. Connecting Wires – As required
8. LED Indicator – 1
9. PCB and Connectors – As required

2.13 Metal Robot Chassis

Figure 11 Metal Robot Chassis

The Metal Robot Chassis serves as the base structure of the vehicle and provides mechanical support for all electronic and hardware components used in the project. It is designed to hold the motors, wheels, battery, controller, and other modules securely in place. The metal construction offers high strength, durability, and stability, enabling the vehicle to move smoothly while protecting the mounted components from damage during operation and testing.

2.14 DC Geared Motors

Figure 12 DC Geared Motors

The DC Geared Motors are used to provide motion to the robotic vehicle by converting electrical energy into mechanical energy. The built-in gearbox reduces the motor speed and increases torque, allowing the vehicle to move efficiently while carrying the mounted components. In this project, the geared motors drive the wheels and enable controlled movement in different directions such as forward, reverse, left, and right. Their high torque, reliability, and low power consumption make them suitable for robotic and IoT-based vehicle applications.

2.15 L293D Motor Driver Board

Figure 13 L293D Motor Driver Board

The L293D Motor Driver Board is used in this project to control the movement of the DC geared motors. It acts as an interface between the ESP32 microcontroller and the motors, providing the required current and voltage for motor operation. The board contains an L293D H-bridge motor driver IC, which allows the motors to rotate in both clockwise and anticlockwise directions. It enables precise control of the vehicle's movement, including forward, reverse, left, and right turns, while protecting the microcontroller from high motor currents.

2.16 L293D IC

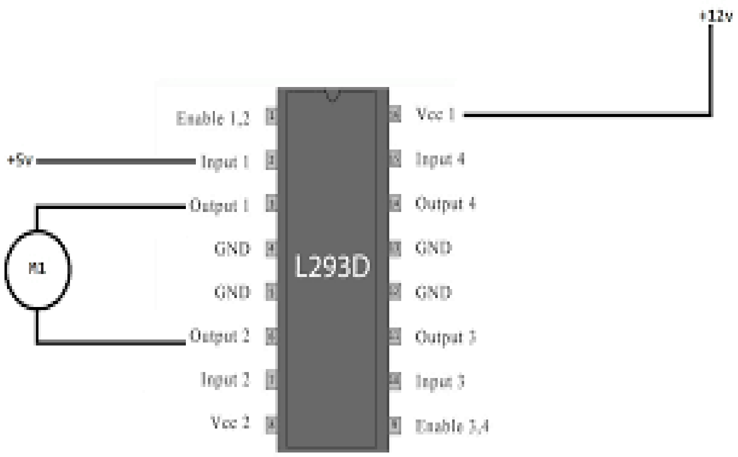

Figure 14 L293D IC

The L293D is a dual H-bridge motor driver integrated circuit used to control the direction and operation of DC motors. In this project, it receives control signals from the ESP32 microcontroller and provides the necessary current to drive the motors. The IC enables bidirectional motor rotation, allowing the vehicle to move forward, backward, left, and right. It also protects the control circuit from voltage spikes generated by the motors, ensuring safe and reliable operation of the system.

3. Hardware and Software Requirements

3.1 Hardware Requirements

Table 1 Hardware requirements

S. No.	Hardware Component	Specification/Purpose
1	ESP32 Development Board	Main microcontroller with Wi-Fi and Bluetooth connectivity
2	16×2 LCD Display	Displays system status, alerts, and operational information
3	I2C LCD Interface Module (PCF8574	Enables LCD communication using only two GPIO pins
4	L293D Motor Driver Board	Controls the operation and direction of DC motors
5	L293D Motor Driver IC	Provides current amplification and motor control
6	DC Geared Motors	Drives the ambulance prototype vehicle
7	Metal Robot Chassis	Mechanical platform for mounting all components
8	Robot Wheels	Enables vehicle movement
9	Battery Pack/Power Supply	Supplies power to the complete system
10	General Purpose PCB (Perfboard)	Used for component mounting and circuit connections
11	Connecting Wires and Connectors	Provides electrical connections between modules
12	USB Cable	Used for programming and powering the ESP32 during development

3.2 Software Requirements

Table 2 software requirements

S. No.	Software	Purpose
1	Arduino IDE	Development environment for programming the ESP32
2	Embedded C/C++	Programming language used for system development
3	ESP32 Board Package	Required to program and upload code to the ESP32
4	LCD Library (LiquidCrystal_I2C)	Controls the 16×2 LCD display through I2C communication
5	Wi-Fi Library	Enables IoT communication and network connectivity
6	Serial Monitor	Used for debugging and monitoring system operation

7	Windows 10/11 or Linux OS	Platform for software development and testing
8	IoT Cloud Platform (Optional)	Used for remote monitoring and communication
9	USB Driver for ESP32	Establishes communication between the computer and ESP32 10 Web Browser Accesses th
10	Web Browser	Accesses the IoT dashboard and monitoring interface

3.3 Minimum System Requirements for Development

- ❖ Processor: Intel Core i3 or equivalent
- ❖ RAM: 4 GB (8 GB recommended)
- ❖ Storage: 500 MB free disk space
- ❖ Operating System: Windows 10/11, Linux, or macOS
- ❖ Internet Connection: Required for IoT communication and cloud services

3.4 Implementation

3.4.1 Hardware Implementation

The hardware implementation of the IoT-Enabled Ambulance Dispatch System consists of two major units: the control unit and the vehicle unit. The ESP32 microcontroller serves as the central processing unit of the system. It is responsible for acquiring data, processing information, and controlling the operation of the ambulance prototype. The ESP32 was selected due to its built-in Wi-Fi and Bluetooth capabilities, which support IoT-based communication.

A 16×2 LCD display integrated with a PCF8574 I2C interface module is used to display system information such as ambulance status, emergency alerts, and operational messages. The I2C module minimizes the number of GPIO pins required and simplifies circuit wiring. A general-purpose PCB (perfboard) is used for mounting and interconnecting the electronic components, ensuring reliable electrical connections and compact assembly.

The vehicle prototype is built on a metal robot chassis equipped with two DC geared motors. These motors provide the required motion for the ambulance model. An L293D motor driver board, containing the L293D H-bridge IC, acts as an interface between the ESP32 and the motors. The driver board supplies sufficient current to the motors and enables directional control, including forward, reverse, left, and right movements.

A battery pack powers the entire system, providing energy to the ESP32, LCD module, and motor driver circuit. Connecting wires and connectors establish communication and power links between all modules. The complete hardware setup is mounted securely on the chassis to ensure stability during operation. The integrated hardware architecture demonstrates the practical implementation of an IoT-based emergency vehicle dispatch and monitoring system capable of real-time communication and vehicle control.

3.4.2 Software Implementation

The software implementation of the IoT-Enabled Ambulance Dispatch System is designed to coordinate communication, vehicle control, and real-time monitoring. The ESP32 microcontroller is programmed using the Arduino IDE, which provides a user-friendly environment for developing embedded applications. The software is written in Embedded C/C++ and uploaded to the ESP32 through a USB interface.

The program begins by initializing the hardware peripherals, including the LCD display, motor driver, and communication interfaces. The Wi-Fi functionality of the ESP32 is configured to establish connectivity with the IoT platform. Once connected, the system continuously monitors incoming dispatch requests and status information.

The software processes emergency notifications received through the network and displays relevant information on the LCD screen. Based on the received commands, the ESP32 generates control signals that are sent to the L293D motor driver. These signals determine the movement and direction of the ambulance prototype. The software includes routines for forward movement, stopping, turning, and reversing, enabling flexible vehicle navigation.

Error-handling mechanisms are incorporated to maintain reliable communication and prevent system failures. The program continuously updates operational data and ensures synchronization between the control station and vehicle unit. Modular programming techniques are used to improve readability, maintenance, and future expansion of the system.

The software architecture integrates IoT communication, display management, and motor control into a single platform. This allows real-time monitoring and efficient management of emergency vehicle dispatch operations. The implemented software successfully demonstrates how IoT technology can improve ambulance response systems through automated communication, remote monitoring, and intelligent vehicle control.

3.5 Advantages

3.5.1. Enhanced Detection Accuracy and Reliability

A standout advantage of this prototype is the move away from high-cost, single-sensor systems toward a multi-sensor fusion approach. By integrating force sensors, MEMS accelerometers, and vibration sensors, the system drastically reduces the margin of error associated with accident detection. Unlike traditional models that might trigger a false alarm due to a minor bump or pothole, this prototype cross-references data from multiple sources to verify a legitimate high-impact collision. This technical redundancy ensures that emergency services are only deployed when truly necessary, saving valuable resources while maintaining a high sensitivity to actual life-threatening events. Furthermore, the use of embedded hardware like the ESP32 or Arduino provides a robust, low-latency processing environment that can handle these complex sensor inputs in real-time, ensuring that detection occurs within milliseconds of an impact.

3.5.2. Drastic Reduction in Emergency Response Time

The prototype is specifically engineered to maximize the "Golden Hour," the critical window where medical intervention is most effective. By automating the entire notification process, the system removes the human variable bystanders or incapacitated victims from the reporting chain. The integration of GPS and GSM modules allows for the immediate and autonomous transmission of precise geographical coordinates to hospitals and ambulance services. This eliminates the common problem of emergency responders struggling to find an accident site based on vague or panicked verbal descriptions. Moreover, the "automated green corridor" feature which controls traffic signals to prioritize the ambulance's route directly addresses the primary cause of delay: urban traffic congestion. By ensuring that emergency vehicles do not have to wait at junctions, the prototype guarantees a faster, safer transit to the hospital, which is a decisive advantage in improving patient survival rates.

3.5.3. Modularity, Cost-Efficiency, and Scalability

Despite its sophisticated capabilities, the prototype is built using highly accessible and modular components such as relay circuits, motor drivers, and common IoT modules. This design choice makes the system highly cost-effective compared to the "expensive" existing models mentioned in your project description. The modular nature of the hardware means it can be easily adapted for different vehicle types or integrated into existing smart city infrastructures without requiring a total overhaul. Additionally, the inclusion of an Android mobile application and RF modules adds a layer of versatility, allowing for multi-channel communication (SMS, app alerts, and local wireless signals). This scalability makes it a viable solution not just for luxury vehicles, but as a standard safety feature for all road users, contributing to a global reduction in road accident fatalities through affordable, high-tech intervention.

3.6 Applications

The key applications of the real-time accident detection and ambulance assistance system:

❖ Early detection and notification of accidents: The system uses sensors and algorithms to promptly detect accidents and send alerts to emergency services. This can help ensure that the injured receive prompt medical attention, which can be critical in saving lives.

❖ Analysis of accident data: The system can collect and analyze data on accidents, including their causes and locations. This information can be used to identify accident hotspots, make changes to road design, and improve safety measures, thereby reducing the frequency and severity of accidents.

❖ Emergency response: The system can be used to improve emergency response times. By providing realtime information on the location of accidents and the fastest route to the accident site, the system can help emergency services respond more quickly and effectively.

❖ Public safety: The system can enhance public safety by providing real-time updates on traffic conditions, helping drivers avoid accident-prone areas, reducing the risk of accidents, and improving road safety. Overall, the real-time accident detection and ambulance assistance system can help improve road safety, reduce traffic congestion, and save lives by enabling faster and more effective emergency response, analyzing accident data, managing traffic, and enhancing public safety.

4.Conclusion

This project successfully addresses the critical global challenge of high road mortality rates by bridging the gap between accident occurrence and emergency medical intervention. By leveraging the power of embedded systems and IoT, the developed automated accident detection and alert system effectively eliminates the dangerous delays associated with manual reporting. Through the integration of Arduino-based hardware, force sensors, and accelerometers, the system provides a robust mechanism for identifying high-impact collisions in real-time. The automated transmission of precise GPS coordinates via GSM and Android platforms ensures that emergency responders can capitalize on the "golden hour," drastically increasing the probability of survival for crash victims. Unlike traditional methods that rely on the presence or ability of bystanders to call for help, this autonomous approach offers a fail-safe solution that functions even in remote locations or instances where victims are incapacitated. Testing under simulated conditions confirms the system's high reliability and operational efficiency, validating the use of relay circuits and motor drivers to enhance modularity. Ultimately, this research contributes a significant technological advancement to intelligent transportation systems, demonstrating that the fusion of affordable sensors with wireless communication can transform road safety infrastructure. By minimizing human error and maximizing response speed, this project provides a scalable, life-saving framework that is essential for modern, high-traffic environments, paving the way for a future where vehicle collisions no longer result in preventable fatalities due to lack of communication

REFERENCES:

1. P. K. Balachandran, "Smart IoT in Renewable Energy Systems," Recent Adv. Electr. Electron. Eng. Former. Recent Pat. Electr. Electron. Eng., vol. 17, no. 10, pp. 953–953, Dec. 2024, doi: 10.2174/2352096517102408121157 11.
2. A. K. Biswal, D. Singh, B. K. Pattanayak, D. Samanta, and M.-H. Yang, "IoT‐Based Smart Alert System for Drowsy Driver Detection," Wirel. Commun. Mob. Comput., vol. 2021, no. 1, p. 6627217, Jan. 2021, doi: 10.1155/2021/6627217.
3. N. Nireekshana, "A POD Modulation Technique Based Transformer less HERIC Topology for PV Grid Tied-Inverter," in E3S Web of Conferences, EDP Sciences, 2025, p. 01001. Accessed: Apr. 11, 2025. [Online]. Available: https://www.e3s-conferences.org/articles/e3sconf/abs/2025/16/e3sconf_icregcsd2025_01001/e3sconf_icregcsd2025_01001.html
4. N. Nireekshana, R. Ramachandran, and G. V. Narayana, "A Novel Swarm Approach for Regulating Load Frequency in Two-Area Energy Systems," Int. J. Electr. Electron. Res., vol. 11, no. 2, pp. 371–377, Jun. 2023, doi: 10.37391/ijeer.110218.
5. R. K. Kodali, V. Jain, S. Bose, and L. Boppana, "IoT based smart security and home automation system," in 2016 international conference on computing, communication and automation (ICCCA), IEEE, 2016, pp. 1286–1289. Accessed: Jun. 20, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7813916/
6. N. Nireekshana, R. Ramachandran, and G. V. Narayana, "A New Soft Computing Fuzzy Logic Frequency Regulation Scheme for Two Area Hybrid Power Systems," Int. J. Electr. Electron. Res., vol. 11, no. 3, pp. 705–710, Aug. 2023, doi: 10.37391/IJEER.110310.
7. H. Singh, V. Pallagani, V. Khandelwal, and U. Venkanna, "IoT based smart home automation system using sensor node," in 2018 4th International Conference on Recent Advances in Information Technology (RAIT), IEEE, 2018, pp. 1–5. Accessed: Jun. 20, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8389037/
8. Namburi Nireekshana, Tanvi H Nerlekar, P. N. Kumar, and M. M. Bajaber, "An Innovative Solar Based Robotic Floor Cleaner," May 2023, doi: 10.5281/ZENODO.7918621.