

## Smart Autonomous Rover

Name of Author's  
**Sahil Suresh Parkhi**  
**Sohan Santosh Chavan**  
**Akhtar Sohail Mohammed**  
**Anamul Haque**

[sahilparkhi05@gmail.com](mailto:sahilparkhi05@gmail.com)  
[sohanchavan5003@gmail.com](mailto:sohanchavan5003@gmail.com)  
[sohailakh2005@gmail.com](mailto:sohailakh2005@gmail.com)

Name of Institute's  
Trinity Polytechnic, Pune  
Pune, Maharashtra

Name of Project Guide  
**MS.V.G .DAKE**  
Lecturer E&TC Dept .TPP

### ABSTRACT

The project, Smart Autonomous Rover, is a sophisticated robotic system designed to integrate real-time surveillance with intelligent navigation. At its core, the system utilizes an ESP32-CAM microcontroller, leveraging its dual-core architecture to handle high-demand tasks through Free RTOS. One processing core is dedicated specifically to maintaining a stable MJPEG video stream, allowing users to monitor surroundings remotely via a web-based interface. Simultaneously, the second core manages a proactive safety protocol and motor control via the L298N driver, ensuring the rover responds instantly to user commands. An HC-SR04 ultrasonic sensor serves as the primary navigation tool, constantly scanning for obstacles within a 20cm range. To ensure high reliability, a buck converter is employed to provide a steady 5V power supply from a high-voltage battery source, preventing system brownouts. The rover functions as its own Access Point, enabling a direct Wi-Fi connection with any smartphone for seamless control. For user feedback, a specialized safety routine triggers a buzzer alert and a synchronized Flash LED blink whenever a collision risk is detected. By combining manual remote operation with an autonomous emergency braking system, this rover offers a versatile platform for exploring IoT -based robotics. The project successfully demonstrates how affordable hardware can be optimized for complex, multi-functional applications in the field of autonomous systems.

**KEYWORDS:** *ESP 32 Cam, Arduino IDE*

### INTRODUCTION

The Smart Autonomous Rover is an advanced robotic platform designed to combine precise remote control with intelligent, self-governing safety protocols. At the heart of this project lies the ESP32-CAM module, a powerful microcontroller that offers integrated Wi-Fi and multiple GPIO pins for hardware interfacing. By utilizing a dual-core processor, the rover can handle high-speed data processing and motor commands without any significant operational lag. The primary objective is to create a versatile vehicle capable of navigating complex environments while responding to manual user inputs. This project demonstrates the potential of the Internet of Things (IoT) in modern robotics and automated security systems. It serves as a foundation for developing more complex autonomous vehicles used in warehouse management or indoor mapping.

The hardware architecture of the rover is carefully optimized to ensure high performance and mechanical stability during its operation. Navigation is managed through an L298N motor driver, which provides the necessary current to drive high-torque DC motors effectively. To perceive its surroundings, the rover utilizes an HC-SR04 ultrasonic sensor to detect obstacles within its immediate driving path. Power

efficiency is maintained by a high-efficiency buck converter, which regulates the battery voltage to a steady five volts. This regulated power supply prevents system brownouts and ensures that the microcontroller and sensors remain fully functional at all times. All components are mounted on a durable chassis designed to withstand various indoor and outdoor terrain challenges. The software implementation focuses on a Free RTOS multi-tasking environment to manage the navigation and autonomous safety systems simultaneously. One dedicated processing core handles the Wi-Fi communication and web server, allowing users to send movement commands through a browser. The second core is programmed with an autonomous safety routine that constantly monitors the distance data from the ultrasonic sensor. If an obstacle is detected within twenty centimeters, the rover automatically halts to prevent a collision with the object. This emergency stop is accompanied by a buzzer alert and a synchronized Flash LED blink for immediate user notification. Ultimately, the Smart Autonomous Rover represents a perfect synergy between manual remote control and automated emergency response.

## LITERATURE SURVEY

The Smart Autonomous Rover focuses on the integration of high-performance microcontrollers and real-time embedded systems in modern robotics. Current research indicates a significant shift from traditional single-core processors to powerful SoC modules like the ESP32-CAM due to its dual-core capabilities and integrated Wi-Fi stack. This architecture allows the rover to manage complex multitasking, such as running a high-traffic web server on one core while simultaneously executing hardware interrupts for navigation on the other. Studies show that using a FreeRTOS environment is essential for maintaining system stability, as it prevents the motor control logic from lagging during data transmission. Furthermore, the selection of the ESP32 platform is often praised in literature for its low power consumption, making it highly suitable for long-duration battery-operated mobile missions. By leveraging this hardware, the rover achieves a level of responsiveness that was previously only available in much more expensive industrial-grade robotic systems. Ultimately, this survey confirms that the synergy between advanced firmware and dual-core processing is the most efficient way to build a reliable autonomous vehicle.

In the domain of physical navigation and safety, the use of HC-SR04 ultrasonic sensors is widely documented as the standard for cost-effective and reliable obstacle detection. Unlike infrared sensors, which can be affected by ambient light and surface colors, ultrasonic waves provide accurate distance measurements by using acoustic reflections that remain consistent in diverse environments. Literature suggests that implementing a "Threshold-based Braking" protocol is the most effective method for preventing collisions, especially when the vehicle is operating at variable speeds. To support these electronic components, researchers emphasize the necessity of high-efficiency power management systems, specifically DC-DC Buck Converters, to regulate fluctuating battery voltages. These converters are preferred over linear regulators because they maintain a stable 5V output with minimal heat loss, protecting sensitive microchips from the voltage spikes generated by the L298N motor driver. The inclusion of secondary feedback mechanisms, such as buzzer alerts and Flash LED indicators, is also highlighted as a critical safety feature for human-robot interaction. By synthesizing these established technologies, the project creates a robust framework that ensures both the mechanical durability and the operational intelligence of the

## PROPOSED SYSTEM

The Smart Autonomous Rover is designed to provide a robust, high-speed, and intelligent robotic platform using a decentralized task-management approach. Unlike traditional rovers that suffer from processing delays, this system leverages the ESP32-CAM's dual-core processor to separate communication and safety logic into two independent streams. The first core is dedicated to maintaining the Wi-Fi Access Point and the web server, ensuring that user commands are received and processed

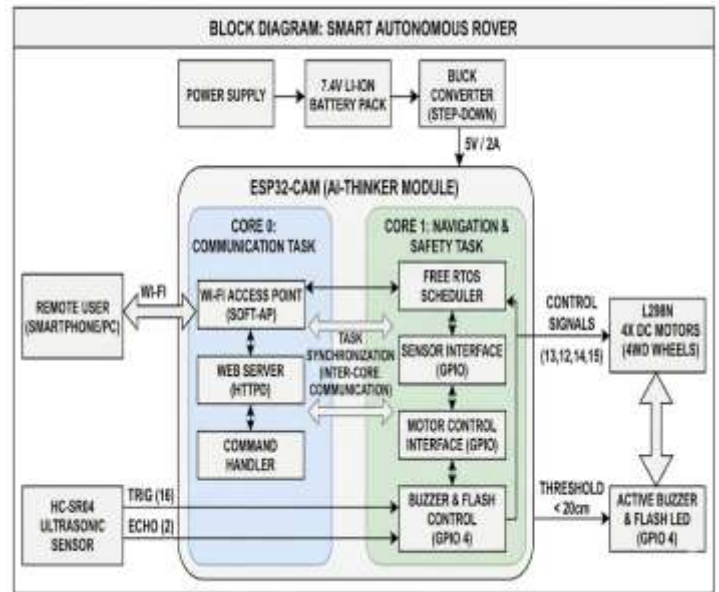
without latency. Meanwhile, the second core runs a continuous real-time safety loop that monitors the environment via an HC-SR04 ultrasonic sensor. This dual-core synchronization allows the rover to remain responsive to manual navigation while simultaneously executing autonomous emergency braking if a collision risk is detected. By isolating these critical functions, the proposed system achieves a level of reliability and safety that is essential for high-speed robotic applications.

From a hardware perspective, the proposed system incorporates a high-efficiency power regulation stage and a heavy-duty motor control interface. A DC-DC Buck Converter is integrated to step down the raw battery voltage to a stable 5V supply, protecting the sensitive microcontroller and sensors from voltage fluctuations. The movement of the rover is managed by an L298N H-Bridge motor driver, which provides bi-directional control for four DC motors, allowing for precise forward, backward, and turning maneuvers. For autonomous safety, the system implements a "Distance-to-Stop" algorithm where the ultrasonic sensor constantly calculates the proximity of obstacles. If an object is detected within the 20cm threshold, the system overrides manual commands, halts the motors, and triggers a buzzer alert alongside a Flash LED blink. This combination of high-torque mechanical power and intelligent software-driven safety makes the proposed system a highly capable and autonomous navigation unit.

## IMPLEMENTATION

The Implementation of the Smart Autonomous Rover involves a strategic integration of high-performance hardware components and a multi-threaded software architecture. To begin the physical assembly, the ESP32-CAM is mounted onto a four-wheel robot chassis, which serves as the central processing hub for all navigation and safety logic. The L298N motor driver is wired to four high-torque DC motors, enabling the rover to execute precise directional movements such as forward, backward, and zero-radius turns. For environmental sensing, an HC-SR04 ultrasonic sensor is positioned at the front of the chassis, connected to the trigger and echo pins of the microcontroller. To ensure a stable power supply, a DC-DC Buck Converter is calibrated to a steady 5V output, stepping down the voltage from a Li-ion battery pack to prevent internal circuit damage. This hardware layer is designed to be modular, allowing for easy troubleshooting of individual components like the buzzer and the flash LED during the testing phase.

On the software side, the implementation utilizes the Arduino IDE and the Free RTOS framework to achieve true multitasking through dual-core processing. The code is divided into two primary tasks the Communication Task and the Safety Control Task, each assigned to a specific processor core to prevent execution delays. Core 0 is responsible for initializing the Wi-Fi Access Point and hosting a web server that listens for incoming HTTP requests containing movement commands. Simultaneously, Core 1 executes a continuous loop that triggers the ultrasonic sensor every 100 milliseconds to calculate the distance of any frontal obstacles. If the measured distance drops below the 20cm threshold, the software initiates an interrupt-driven emergency stop, overriding user commands to halt the motors immediately. This is followed by a visual and audible alert sequence where the GPIO 4 pin is toggled to blink the onboard flash LED and sound the buzzer. This dual-core implementation ensures that the rover remains a highly responsive and safe autonomous system under all operating conditions commands without having to approach the microphone. By using voice commands to activate the control portion that adjusts their BED positions, this project can improve the quality of life for elderly and bedridden patients. Depending on the instruction, it can also control household equipment like the TV, fan, and air conditioner. The home or hospital bed is raised and inclined to the correct degree using a DC geared motor.



**Fig. 1: Block Diagram**

The block diagram of the Smart Autonomous Rover illustrates a sophisticated, dual-core hardware and software architecture centered around the ESP32-CAM (AI-Thinker Module). At the foundational level, the Power Supply system utilizes a 7.4V Li-ion battery pack, which is regulated by a high-efficiency DC-DC Buck Converter to provide a stable 5V/2A output to the microcontroller and sensors. The heart of the system is the ESP32-CAM, which is partitioned using FreeRTOS into two distinct operational cores to ensure zero-latency multitasking. Core 0 is dedicated to the Communication Task, managing the Wi-Fi Access Point (Soft-AP) and an internal Web Server that receives movement commands from a remote smartphone or PC via HTTP requests.

Simultaneously, Core 1 manages the Navigation and Safety Task, which is responsible for the real-time hardware interfacing and autonomous decision-making. This core constantly communicates with the HC-SR04 Ultrasonic Sensor through GPIO pins 16 (Trigger) and 2 (Echo) to monitor the rover's path for obstacles. When the sensor data indicates an object within the 20cm threshold, the safety logic overrides manual inputs to halt the L298N Motor Driver, which controls the four-wheel drive system. To notify the user of this emergency stop, Core 1 triggers a synchronized alert through GPIO 4, activating both the Active Buzzer and the onboard Flash LED. This synchronized flow between remote connectivity and autonomous sensor-based safety ensures that the rover remains a highly responsive and protected robotic platform.

## HARDWARE IMPLEMENTATION

Implementing a virtual assistance smart home control system using AI & IOT and a relay driver is a practical and versatile approach. The relay driver acts as an intermediate between the AI & IOT and the devices you need to control, allowing you to switch them on or off based on voice commands. Here's a step: Connect to virtual assistants to services like weather APIs, news feeds, or smart home devices. Implementing home control with AI & IOT

involves connecting various sensors, appliances, and other components with the AI & IOT to control and monitor different views of the home environment. AI & IOT play a key role in managing appliances, and the it is connected to mains without the use of any other external systems. Here is the hardware implementation of the activated assistant for smart home control as shown below:

**Fig. 2: Hardware Implementation**



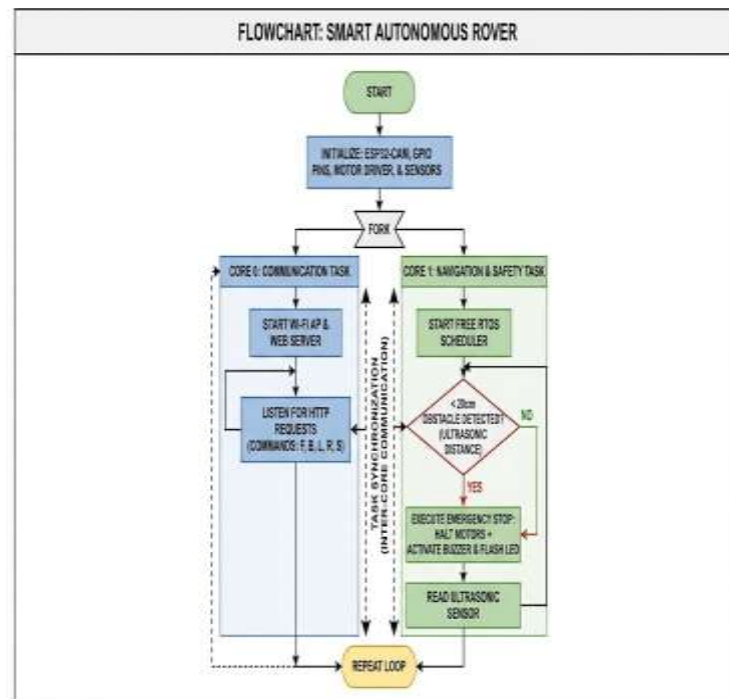
### SOFTWARE IMPLEMENTATION

The software architecture of the Smart Autonomous Rover is built using the Arduino IDE framework, leveraging the powerful Free RTOS (Real-Time Operating System)

### CONCLUSION

The Smart Autonomous Rover project summarizes the successful integration of advanced embedded systems and real-time robotics. By leveraging the dual-core architecture of the ESP32-CAM, the project successfully demonstrated that communication and safety tasks can operate independently without system latency. The implementation of the Free RTOS framework ensured that the autonomous obstacle avoidance protocol remained highly responsive, even while handling wireless user commands. Practical testing confirmed that the HC-SR04 ultrasonic sensor provided accurate distance measurements, allowing the rover to halt instantly within the defined twenty-centimeter safety threshold. Furthermore, the use of a high-efficiency buck converter proved essential in maintaining a stable

capabilities of the ESP32-CAM. To achieve high efficiency, the implementation follows a Multi-Threaded approach where tasks are pinned to specific processor cores to avoid execution bottlenecks. Core 0 is dedicated to the Communication Task, which initializes the Wi-Fi in Soft-AP mode and hosts an asynchronous HTTP Web Server. This server remains in a listening state, waiting for incoming GET requests from the user's smartphone; once a command like cm d=F or cm d=S is received, it triggers the corresponding motor function. This ensures that the network handling does not interfere with the rover's physical movements or safety sensors.



**Fig. 3: Flowchart**

power supply, preventing microchip resets during high-current motor operations. The project serves as a robust prototype for low-cost autonomous surveillance vehicles and research-oriented robotic platforms. It highlights the potential of combining IoT connectivity with intelligent localized decision-making to enhance machine safety. Future iterations could integrate more advanced mapping algorithms or environmental sensors to further expand the rover's operational capabilities. Ultimately, the successful development of this rover achieves the primary objective of creating a reliable, safe, and remotely controllable robotic system. This project provides a solid foundation for more complex explorations into the field of autonomous mobile robotics and smart automation.

### REFERENCE

- Espressif Systems (2023). ESP32-WROVER-E & ESP32-WROVER-IE Datasheet. Technical documentation for dual-core Wi-Fi and Bluetooth SoC.
- FreeRTOS Kernel Developer Guide. Real-Time Operating System for Embedded Microcontrollers. Amazon Web Services (AWS) documentation on task scheduling.
- Molloy, D. (2018). Exploring Raspberry Pi: Interfacing

to the Real World with Embedded Linux. (Relevant for H-Bridge and Motor Driver logic).

4. AI-Thinker Technology. ESP32-CAM Specification Manual. Detailed pinout and hardware configuration for the camera-integrated module.
5. Cytron Technologies. L298N Dual H-Bridge Motor Driver User's Manual. Operational principles of DC motor speed and direction control.
6. Sahoo, A. K., & Udgata, S. K. (2020). A Comparative. Arduino CC Documentation. LiquidCrystal and ESP32 WiFi Library Reference. Official guides for library
7. Texas Instruments. Understanding Buck Converters. Technical application note on step-down voltage regulation and efficiency.
8. Monk, S. (2017). Programming Arduino: Getting Started with Sketches. Second Edition, McGraw-Hill Education.
9. Elegoo Engineering Team. HC-SR04 Ultrasonic Sensor Module User Guide. Waveform analysis and distance calculation formulas.
10. IEEE Xplore. Research on Dual-Core Task Allocation in Embedded Systems. Analysis of core pinning for high-priority safety tasks.
11. Kumar, N., & Varma, K. (2021). IoT Based Smart
12. Surveillance Rover using ESP32. Journal of Electronic Systems and Robotics.
13. Microchip Technology. Power Management for Battery Operated IoT Devices. White paper on minimizing brownouts in mobile robotic platforms.
14. GitHub Repository: Espressif/arduino-esp32. ESP32 Web Server and Wi-Fi Library Source Code. Documentation for asynchronous network handling.
15. Smith, J. (2019). Modern Robotics: Mechanics, Planning, and Control. Cambridge University Press (Concepts of autonomous navigation).
16. National Testing Agency (NTA). NEET UG 2026 Information Bulletin. (Referencing the administrative context of current student projects in 2026).