**ESP32-CAM IoT-Based Smart Car with Voice Control & Live Streaming**

Team Members

1. Guttula Vijay (23911A1225)
2. Juveria Usman (23911A1230)
3. K V Spurthi (23911A1231)
4. Kadimi Jaswanth (23911A1232)

1. Abstract

With advancements in the Internet of Things (IoT) and embedded systems, automation in vehicular technology has become an essential research area. This project aims to develop an IoT-enabled smart car controlled via a mobile application and voice commands while providing live video streaming for real-time monitoring. The car is equipped with four DC motors, controlled using an L298N motor driver and an Arduino, along with an ESP32-CAM for live streaming. It also integrates ultrasonic sensors for automatic obstacle detection and braking, making it highly versatile for surveillance, remote vehicle control, and other real-world applications.

2. Introduction

The rapid adoption of IoT, AI, and robotics in modern applications has paved the way for automated and remotely operated vehicles. Traditional remote-control cars rely on RF modules, which limit their range and capabilities. In contrast, IoT-based vehicles leverage Wi-Fi connectivity, allowing them to be controlled via a mobile app from anywhere in the world.

This project combines:  
✔ ESP32-CAM for live streaming and IoT communication  
✔ Arduino for motor and sensor control  
✔ L298N Motor Driver for driving four DC motors  
✔ Voice commands & app-based control for remote operation  
✔ Ultrasonic sensors for automatic obstacle detection  
✔ Servo motors for braking system

This project can be used in surveillance, search-and-rescue missions, smart transportation, and autonomous vehicles.

3. Objectives

* Live Video Streaming: Enable real-time monitoring via ESP32-CAM.
* App & Voice Control: Control the car wirelessly via a mobile app and voice commands.
* Obstacle Avoidance: Implement ultrasonic sensors for automatic stopping.
* Speed & Direction Control: Use PWM signals for smooth motor control.
* Braking System: Utilize servo motors to engage a braking mechanism.
* LED Headlights & Indicators: Manage LEDs for navigation and signalling.

4. Components Used

Microcontrollers & Modules

* ESP32-CAM (for live video streaming and Wi-Fi-based communication)
* Arduino (Uno/Nano) (for motor and sensor control)

Motor & Driver Circuit

* 4 × DC Motors (two for each side, connected in parallel)
* L298N Motor Driver Module (for controlling motors)

Power Supply

* 12V Battery (for motors and Arduino)
* 5V Regulated Power (from L298N) (for ESP32-CAM)

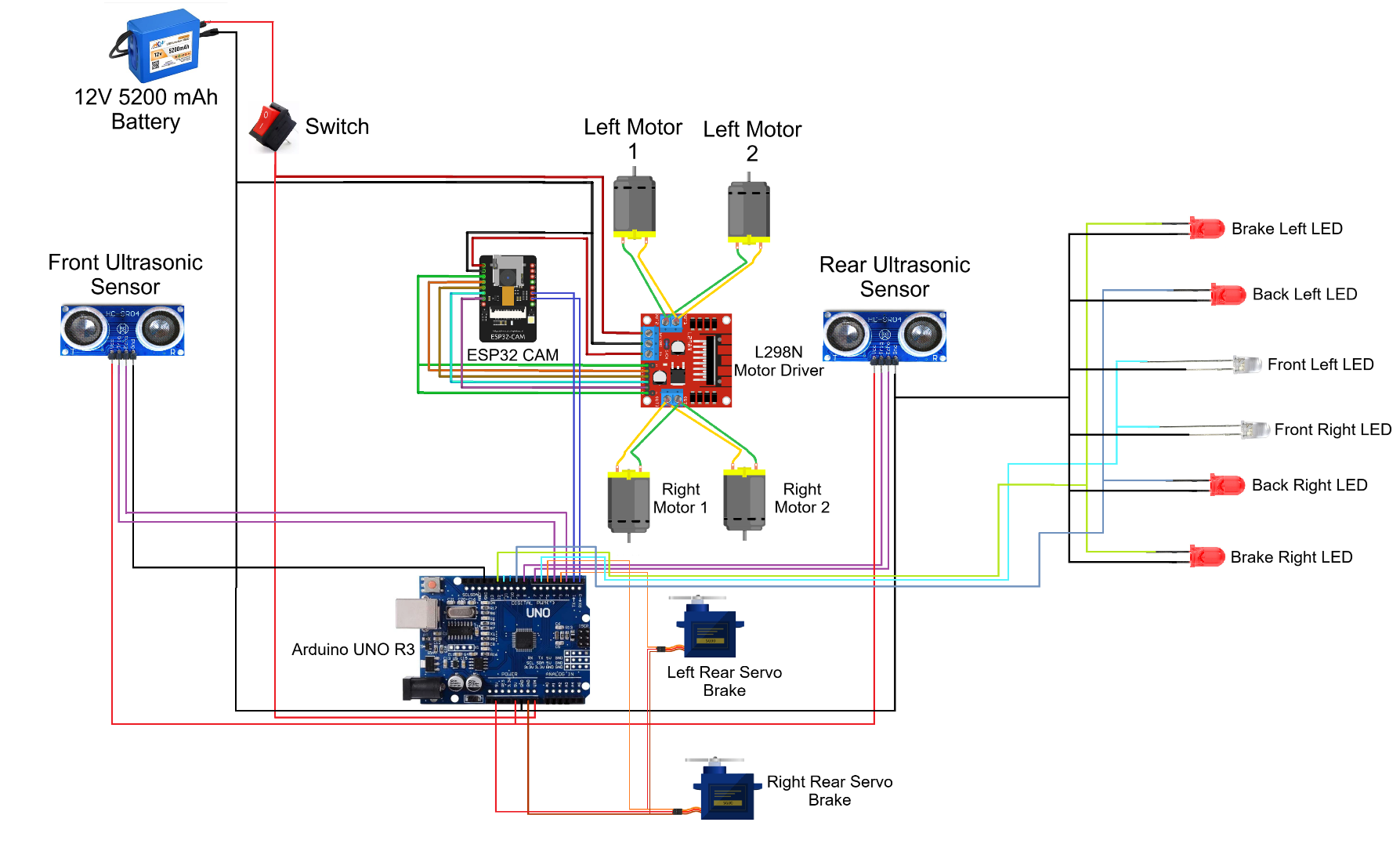
Sensors & Actuators

* 2 × Ultrasonic Sensors (for front and rear obstacle detection)
* 2 × Servo Motors (for braking system)

Others

* LED Headlights & Brake LEDs
* Mobile App (for IoT control and voice commands)

5. Circuit Diagram & Connections



Power Connections:

* 12V Battery +ve → L298N 12V IN, Arduino VIN
* 12V Battery -ve → L298N GND, Arduino GND, ESP32-CAM GND, Servo GND, Sensor GND, LED GND
* L298N 5V OUT → ESP32-CAM 5V IN

Motor Control (L298N to Motors, ESP32 CAM & Arduino):

* OUT1 & OUT2 → Right Front & Right Rear Motor
* OUT3 & OUT4 → Left Front & Left Rear Motor
* ENA (PWM Left and Right Speed Control) → ESP 32 CAM - GPIO 12
* IN1 & IN2 (Left Side Motors Control) → GPIO 13, 15
* IN3 & IN4 (Right Side Motors Control) → GPIO 14, 2

ESP32-CAM Communication (For App & Voice Control):

* TX (ESP32) → Arduino D0 (RX)
* RX (ESP32) → Arduino D1 (TX)
* Power Supply → L298N 5V
* GND → Arduino GND

Ultrasonic Sensors (Obstacle Detection - Auto Stop):

* Power Supply → Arduino 5V
* GND → Arduino GND
* Front TRIG → Arduino D4
* Front ECHO → Arduino D2
* Rear TRIG → Arduino D8
* Rear ECHO → Arduino D7

Servo Motors (Brake System for Rear Tires):

* Power Supply → Arduino 5V
* GND → Arduino GND
* Left Rear Brake Servo → Arduino D3 (PWM)
* Right Rear Brake Servo → Arduino D5 (PWM)

LED Control (Front Adjustable, Others ON/OFF Only):

* Front Headlights → Arduino D6 (PWM)
* Back LEDs → Arduino D9
* Brake LEDs → Arduino D12
* Extra LED Control → Arduino D13

6. Challenges Faced

| Challenges | Solutions Implemented |
| --- | --- |
| Power Management Issues | Used a 12V battery with regulated 5V output for ESP32-CAM. |
| Motor Synchronization | Used PWM control for smooth speed regulation. |
| ESP32-CAM Wi-Fi Range Issues | Optimized Wi-Fi antenna placement and ensured a strong signal. |
| Real-time Streaming Lag | Reduced resolution & frame rate to optimize bandwidth usage. |
| Voice Command Accuracy | Integrated a reliable speech-to-text system with filtering. |
| Obstacle Detection False Positives | Calibrated ultrasonic sensors and added delay filters. |

7. Real-World Applications & Problem-Solving

This project is designed to solve real-world challenges in various domains:

1. Security & Surveillance

* The live video streaming feature allows remote surveillance.
* Can be used in military reconnaissance missions.

2. Search & Rescue Operations

* Can navigate dangerous areas without risking human lives.
* Can be used in disaster-stricken areas for locating survivors.

3. Smart Transportation & Home Automation

* The concept can be extended to autonomous delivery vehicles.
* Can be used as a prototype for self-driving cars.

4. Industrial & Agricultural Use

* Can automate material handling in factories.
* Can be used for monitoring crops in agricultural fields.

8. Feasibility Analysis

Technical Feasibility

* Uses widely available components like ESP32-CAM, Arduino, and L298N.
* Open-source libraries are available for motor control, IoT communication, and voice recognition.

Economic Feasibility

* Low-cost components make it an affordable solution.
* Compared to professional surveillance robots, this is a cost-effective alternative.

Operational Feasibility

* Can be easily controlled via a mobile app.
* Requires minimal training for operation.

Market Feasibility

* High demand in IoT-based automation and security applications.
* Potential for commercialization in security and smart transportation.

9. Future Scope & Enhancements

The project can be further improved with:

1. Autonomous Navigation – Implement AI-based path planning and autonomous driving.
2. Enhanced AI Processing – Use OpenCV for advanced object detection.
3. Long-Range Communication – Integrate LoRa or 4G modules for greater coverage.
4. Solar-Powered Battery System – Improve sustainability with solar panels.
5. Object Tracking & Face Recognition – Upgrade ESP32-CAM for intelligent tracking.
6. Integration with Cloud Storage – Store live footage in the cloud for later analysis.

10. Conclusion

This IoT-enabled smart car provides an efficient solution for remote-controlled vehicles with live streaming, voice control, and obstacle detection. It has applications in security, automation, and autonomous vehicle technology. With further improvements, it can serve as a foundation for fully autonomous AI-powered vehicles.

🚀 This project bridges the gap between IoT, AI, and robotics, making it a stepping stone toward future smart vehicles.