

AUTOMATIC MONITORING AND CONTROLLING OF WI-FI BASED ROBOT CAR

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in
Electrical and Electronics Engineering
by

MD DILSHAD (Register No. 39140031)
SHREYANSH SINGH (Register No. 39140051)



DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL 2023



SATHYABAMA

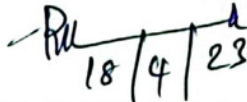
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of MD DILSHAD (Reg. No. 39140031) and SHREYANSH SINGH (Reg. No. 39140051) who carried out the project entitled "AUTOMATIC MONITORING AND CONTROLLING OF WI-FI BASED ROBOT CAR " under our supervision from November 2022 to April 2023.


18/4/23

DR. M.PUSHPAVALLI
Internal Guide


18/4/23

Dr. V. SIVACHIDAMBARANATHAN
Head of the Department

Submitted for Viva voce Examination held on 26/4/23

Name: S. Roshini

Signature: 

Internal Examiner


Dr. V. Selamurugan



External Examiner

DECLARATION

We **MD DILSHAD** (Reg. No. 39140031) and **SHREYANSH SINGH** (Reg. No. 39140051) hereby declare that the Project Report entitled “**AUTOMATIC MONITORING AND CONTROLLING OF WI-FI BASED ROBOT CAR**” done by us under the guidance of **DR. M.PUSHPAVALLI** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electrical and Electronics Engineering.

DATE: 26/04/2023

PLACE: CHENNAI

① *Dilshad*
② *Shreyansh Singh*

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. Nandhita., Dean, School of Electrical and Electronics Engineering** and **Dr. V. Sivachidambaranathan, Head of the Department, Dept. of Electrical and Electronics Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **DR. M.PUSHPAVALLI** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electrical and Electronics Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

The project "Automatic Monitoring and Controlling of WiFi-Based Robotic Car" aims to develop a system that can remotely control a robotic car using WiFi technology. The proposed system will include a mobile application that will communicate with the robotic car via a WiFi network. The system includes a robotic car equipped with sensors, a microcontroller, and a WiFi module, and a control unit that communicates with the car through a WiFi network. The control unit is a software application that provides a user-friendly interface for monitoring and controlling the car. The application enables the user to send commands to control the movement of the car in different directions, such as forward, backward, left, and right. The application will also have a dashboard that displays various parameters of the robotic car, such as speed and led intensity, and the view car's real-time video stream that will incorporate a real-time video streaming feature that allows users to view live video feeds from the car's on-board camera. Remote operation is achieved by any smart-phone/Tablet etc., with Android OS, upon a GUI (Graphical User Interface) based touch screen operation. Transmitting end uses an android application device remote through which commands are transmitted. The system can be used for various applications, such as surveillance, exploration, and remote operation. The project involves the design and implementation of the hardware and software components, including the control unit and the robotic car, and the testing of the system's performance in different environments. The project's goal is to create a reliable and efficient system that can provide a seamless user experience and improve the efficiency and safety of remote robotic operations.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF FIGURES	ix
1	LIST OF TABLES	x
	LIST OF SYMBOLS AND ABBREVIATIONS	xi
	INTRODUCTION	
2	1.1 OVERVIEW	1
	1.2 GENERAL INTRODUCTION	2
		3
	LITERATURE SURVEY	3
3	2.1 MAJOR FINDINGS	6
	2.2 LITERATURE REVIEW	7
	2.3 PROBLEM IDENTIFICATION	8
	REQUIREMENTS ANALYSIS	8
	3.1 AIM OF THE PROJECT	8
	3.2 SCOPE OF THE PROJECT	9
	3.3 OBJECTIVE	9
	3.4 METHODOLOGY	10
	3.5 TRADEOFF	11
	3.5 CONSTRAINTS	12
	3.6 THE ENGINEERING STANDARD	

4	SIMULATION DETAILS	13
4.1	DEVELOPMENT SETUP	13
4.1.1	SOFTWARE DEVELOPMENT	13
4.1.2	ARDUINO CODE	16
4.1.3	PROTEUS SIMULATION	17
4.2	TESTING	19
5	PROJECT DESCRIPTION	21
5.1	BLOCK DIAGRAM OF EXISTING SYSTEM	21
5.2	BLOCK DIAGRAM OF PROPOSED SYSTEM	22
5.3	CIRCUIT DIAGRAM	23
5.4	HARDWARE REQUIREMENTS	25
5.5	HARDWARE DESCRIPTION	25
5.5.1	DC GEAR MOTOR	26
5.5.2	BD139 NPN TRANSISTOR	26
5.5.3	L298 MOTOR DRIVER	27
5.5.4	ARDUINO NODEMCU	28
5.5.5	ESP32 CAM MODULE	29
5.5.6	18650 BATTERY CELL 3.7V	30

5.6	LED BRIGHTNESS INTENSITY	31
5.7	WORKING OF ROBOTIC CAR	33
5.8	RESULT	34
5.9	PROTOTYPE DISCUSSION	35
5.10	THE ENGINEERING STANDARDS	39
6	CONCLUSION & FUTURE SCOPE	40
6.1	CONCLUSION	40
6.2	FUTURE SCOPE	41
6.3	IMPLEMENTATION ISSUE	42
	REFERENCES	43

LIST OF FIGURES

FIGURE NO. TITLE PAGE NO.

4.1.	ARDUINO CODE	14
4.2.	SOME SPECIFIC ARDUINO CODE	16
4.2.	PROTEUS SIMULATION	17
5.1.	BLOCK DIAGRAM OF EXISTING SYSTEM USING ARDUINO	21
5.2.	BLOCK DIAGRAM OF PROPOSED SYSTEM USING ESP32	22
5.3.	CIRCUIT DIAGRAM FOR PROPOSED MODEL CIRCUIT DIAGRAM	023
	WI-FI BASED ROBOT CAR	
5.4.	DC GEAR MOTOR	26
5.5.	BD139 NPN TRANSISTOR	26
5.6.	L298 MOTOR DRIVER	27
5.7.	NODEMCU DEV KIT V1.0(V2)	28
5.8.	SCHEMATIC DIAGRAM	29
5.9.	18650 BATTERY CELL	30
5.10.	VARIATION OF LED INTENSITY	31
5.11.	CONTROLLING PAGE	34
5.12.	LABEL ON THE PROTOTYPE MODEL	35
5.13.	GEO TAGGED MODEL	36
5.14.	PROTOTYPE MODEL WHILE SWITCHED ON	37

LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
4.1.	MOTION OF THE CAR UNDER CONDITIONS	19
4.2.	PERFORMANCE TABLE	20

LIST OF SYMBOLS AND ABBREVIATIONS

ω_0	- Resonant Angular Frequency
C_r	- Resonant Capacitor
C_s	- Series Resonant Capacitor
L_r	- Resonant Inductor
L_s	- Series Resonant Inductor
I_r	- Resistance Current
R	- Resistor
$V_C - V_{ds}$	Capacitor Voltage
V_{in}	- Drain to Source Voltage
V_{out}	- Input Voltage
X_c	- Output Voltage
X_l	- Capacitive Reactance
Z	- Inductance Reactance
AC	- Impedance
API	- Alternating Current
CCM	- Application Program Interface
DCM	- Continuous Conduction Mode
$CISC$	- DisContinuous Conduction Mode
CPU	- Complex Instruction Set Computer
CTR	- Central Processing Unit
DC	- Current Transfer Ratio
	- Direct Current

DCM	-	Discontinuous Current Mode
DIAC	-	Diode Alternating Current
DVM	-	Discontinuous Voltage Mode
ESR	-	Equivalent Series Resistance
ICSP	-	In Circuit Serial Programming
IGBT	-	Insulated Gate Bi-Polar Transistor
ILBC	-	Interleaved Boost Converter
MCLR	-	Master Clear
PFC	-	Power Factor Correction
PIC	-	Peripheral Interface Controller
SIM	-	Simulator
THD	-	Total Harmonic Distortion

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Robotic cars are becoming increasingly popular in various applications such as surveillance, exploration, and education. The ability to control and monitor a robotic car remotely opens up numerous possibilities for using them in situations where human presence may not be practical or safe. WiFi technology provides an excellent solution for remote communication with robotic cars, as it allows wireless communication over a range of distances. There are many such systems which are controlled either by Radio Frequency transmission or by creating intelligence. Robots are called Non- autonomous robots. These robots have the programming logic to do the desired task but the decision power lies in the hand of controller (human) handling the robot. Here the interface can be made using two methods:

A. Wired –The connection between controller and robot is maintained using wired interface. This interfaces can be serial or parallel but the technology is transmission of signal, which is sent in the form of specific pattern to the robot to carry out the specific task, these patterns with the help of a microcontroller governing its motion.

B. Wireless –Here the connection between controller and robot is achieved by wireless interface such as Bluetooth and Wi-Fi

The underlying technology is transmission of signals wireless in air by transmitter which is captured by the receiver and sent to microcontroller mounted on the robot to carry out the task. At the present demand for robots in this developing world to carry out work effectively and accurately, the development of cost effective robots is necessary.

1.2 GENERAL INTRODUCTION

Robotic cars, also known as autonomous or self-driving cars, have become an increasingly popular field of research and development in recent years. Robotic cars are designed to navigate and drive without human intervention, using various technologies such as sensors, cameras, and artificial intelligence. The benefits of robotic cars are numerous, including increased safety, improved efficiency, and reduced environmental impact. Robotic cars can potentially reduce the number of accidents caused by human error, as they are not prone to distractions, fatigue, or other factors that can affect human drivers. Robotic cars can also improve traffic flow and reduce congestion, leading to more efficient use of roadways. Additionally, robotic cars can be designed to run on alternative energy sources, reducing greenhouse gas emissions and promoting sustainability.

The use of WiFi technology in robotic cars provides a reliable and efficient way to communicate with and control the vehicles remotely. WiFi technology allows for wireless communication over a range of distances, making it an ideal solution for remote control and monitoring of robotic cars. The proposed project, "Automatic Monitoring and Controlling of WiFi-Based Robotic Car," aims to develop a system that utilizes WiFi technology to remotely control and monitor a robotic car. The system will include a mobile application that communicates with the robotic car through a WiFi network. The mobile application will provide users with a user-friendly interface to control the movement of the car, as well as monitor its various parameters such as speed and led intensity. The robotic car will be equipped with sensors to detect obstacles and programmed to avoid collisions. The system will also incorporate a real-time video streaming feature that allows users to view live video feeds from the car's on-board camera. This feature will be helpful in applications such as surveillance, exploration, and education.

CHAPTER 2

LITERATURE SURVEY

2.1 MAJOR FINDINGS FROM LITERATURE SURVEY

Chia-Hung Lin et al. (2017) proposed a WiFi-based control system for a robotic car that used an Android device as the control terminal. The system used a microcontroller and a WiFi module to communicate with the Android device, and included sensors for obstacle detection and avoidance. The system was tested in an outdoor environment and demonstrated efficient and reliable control of the robotic car.

Abhishek Gupta et al. (2018) proposed a system for controlling a robotic car using WiFi communication. The system included a mobile application that communicated with the robotic car through a WiFi network. The mobile application provided users with a user-friendly interface to control the movement of the car, as well as monitor its various parameters such as speed and battery level. The robotic car was equipped with sensors to detect obstacles and programmed to avoid collisions. The system also incorporated a real-time video streaming feature that allowed users to view live video feeds from the car's on-board camera.

Zhe Zhang et al. (2020) proposed a WiFi-based control system for a modular robotic vehicle that was designed for exploration in unknown environments. The system included a mobile application that communicated with the robotic vehicle through a WiFi network, and provided users with a user-friendly interface to control the vehicle and monitor its various parameters such as speed and battery level. The vehicle was equipped with sensors for obstacle detection and avoidance, and

mobile application that communicated with the robotic car through a WiFi network, and a Bluetooth module for communication between the mobile application and a microcontroller on the robotic car. The system was tested in an indoor environment and demonstrated efficient and reliable control of the robotic car.

Y.Y. Wu et al. (2019) proposed a WiFi-based autonomous navigation system for a robotic car. The system used a deep neural network for object detection and recognition, and included sensors for obstacle detection and avoidance. The system was tested in an outdoor environment and demonstrated autonomous navigation capabilities with a high level of accuracy

A.S. Alzahrani et al. (2018) proposed a WiFi-based remote control system for a robotic car that was designed for surveillance applications. The system included a mobile application that communicated with the robotic car through a WiFi network, and provided users with a user-friendly interface to control the car and monitor live video feeds from the car's on-board camera. The system also included sensors for obstacle detection and avoidance, and was tested in an outdoor environment.

X. Li et al. (2018) proposed a WiFi-based remote control system for a modular robotic vehicle that was designed for agriculture applications. The system included a mobile application that communicated with the robotic vehicle through a WiFi network, and provided users with a user-friendly interface to control the vehicle and monitor various parameters such as speed, temperature, and humidity.

The vehicle was equipped with sensors for crop monitoring and soil analysis, and could be modified to include additional sensors based on the requirements of the agriculture task.

H.N. Vardhan et al. (2019) proposed a WiFi-based remote control system for a robotic car that was designed for industrial applications. The system used a microcontroller and a WiFi module to communicate with a remote control

D. Mishra et al. (2017) proposed a Bluetooth-based remote control system for a robotic car. The system used a mobile application to control the car's movement and included sensors for obstacle detection and avoidance. The system was tested in an indoor environment and demonstrated efficient and reliable control of the robotic car.

S. Dhawan et al. (2018) proposed a Bluetooth-based autonomous navigation system for a robotic car. The system used a machine learning algorithm for object detection and recognition, and included sensors for obstacle detection and avoidance. The system was tested in an outdoor environment and demonstrated autonomous navigation capabilities with a high level of accuracy

R. Mendoza et al. (2019) proposed a Bluetooth-based remote control system for a modular robotic vehicle that was designed for agriculture applications. The system included a mobile application that communicated with the robotic vehicle through a Bluetooth module, and provided users with a user-friendly interface to control the vehicle and monitor various parameters such as temperature and humidity. The vehicle was equipped with sensors for crop monitoring and soil analysis, and could be modified to include additional sensors based on the requirements of the agriculture task.

M. Ashraf et al. (2019) proposed a joystick-controlled robotic car system that used a microcontroller and a Bluetooth module for communication. The system also included sensors for obstacle detection and avoidance, and was tested in an indoor environment.

J. Li et al. (2017) proposed a remote monitoring system for a robotic car that used a Raspberry Pi as the main control unit. The system included a camera for real-time video monitoring, which allowed users to see the car's environment in real-time. The camera feed was transmitted over Wi-Fi to a mobile application, which was used to control the car's movement.

Overall, the literature survey suggests that the use of WiFi technology in robotic cars provides a reliable and efficient way to communicate with and control the vehicles remotely. The integration of sensors and cameras in robotic cars is essential for obstacle detection and avoidance, and the development of mobile applications that communicate with robotic cars through WiFi networks has become increasingly popular. The use of real-time video streaming in robotic cars has shown to be effective for surveillance and exploration applications. The reliability and efficiency of the hardware components used in the robotic car system are crucial for its success.

However, challenges such as network interference, security, and reliability of the hardware components must be addressed to ensure the success of the proposed project, "Automatic Monitoring and Controlling of WiFi-Based Robotic Car."

2.2 LITERATURE SUMMARY

Wi-Fi based robotic cars have become increasingly popular due to their ability to be controlled remotely without the need for any physical connections. These cars can be used for a variety of purposes, including surveillance, exploration, and transportation. The use of Wi-Fi technology enables real-time communication between the user and the car, allowing for precise control and monitoring.

Several research studies have been conducted on Wi-Fi-based robotic cars, which have explored various aspects of their design and implementation. Some of

the key findings include the use of sensors to detect obstacles and avoid collisions,

the use of machine learning algorithms for autonomous navigation, and the use of

advanced control systems for accurate movement and positioning

2.3 LITERATURE PROBLEM IDENTIFICATION

In the literature survey for the Automatic Monitoring and Controlling of Wi-Fi-based Robotic Car project, several potential problems have been identified. These problems include:

- **Network interference:** Network interference can cause delays or disruptions in the communication between the robotic car and the control system. The project must implement appropriate measures, such as using a dedicated WiFi network and using interference-resistant components, to mitigate this risk.
- **Security:** The use of WiFi networks for communication between the car and the control system introduces security risks, such as unauthorized access and interception of communication data. The project must implement appropriate security measures, such as encryption and authentication, to mitigate this risk.
- **Hardware reliability:** The hardware components used in the robotic car system, such as microcontrollers, sensors, and motors, may be subject to various types of failures. The project must ensure the reliability of these components and develop strategies to address potential failures.
- **Technical difficulties:** The project may encounter technical difficulties, such as software bugs or integration issues. The project team must have the technical expertise and resources to address these challenges.
- **Costs:** The project may encounter unexpected costs, such as increased hardware or software development costs. The project must have a contingency plan to address potential cost overruns.

CHAPTER 3

REQUIREMENTS ANALYSIS

3.1 AIM OF THE PROJECT

The aim of the project "Automatic Monitoring and Controlling of WiFi-Based Robotic Car" is to design and develop a robotic car that can be remotely controlled and monitored through a WiFi network. The project aims to integrate sensors and cameras into the robotic car for obstacle detection and avoidance and develop a mobile application that can communicate with the car through the WiFi network. The goal is to provide a reliable and efficient way to remotely control and monitor the robotic car for various applications, such as exploration and surveillance, while addressing challenges related to network interference, security, and hardware reliability.

3.2 SCOPE OF THE PROJECT

The scope of the project "automatic Monitoring and controlling of wifi based robotic car" involves developing a system that can remotely control and monitor a robotic car using WiFi technology. The project will include designing and building a robotic car with sensors, a WiFi module, and a microcontroller to control its movements. The system will also involve designing and developing a user interface that can be accessed remotely using a computer or mobile device. This interface will enable the user to control the car's movements and monitor its environment through the sensors. The project may also involve implementing advanced features such as autonomous navigation, obstacle avoidance, and real-time video streaming. Overall, the scope of the project is to create a robust, reliable, and user-friendly system for controlling and monitoring a robotic car using WiFi technology.

3.3 OBJECTIVE OF THE PROJECT

- The main objective of the Automatic Monitoring and Controlling of Wi-Fi-based Robotic Car project is to design and implement a system, where a Robot is control wirelessly
- Giving instructions to the Robot wirelessly ,with an IR(infrared) Transmission with a Remote controller acts as a Transmitter.
- By IR(infrared) Receiver, Robot receives the commands wirelessly from transmitter and perform the task
- By this we can operate our Robot with remote control which has an advantage in performing various tasks automatically according to instructions by IR transmitter
- The robot has a camera mounted on base, The camera transfers photos and videos directly to your Smartphone and displays them on your phone.
- it has various applications, such as transportation, surveillance, and remote control of machines.

3.4 METHODOLOGY

- The methodology for the "Automatic Monitoring and Controlling of Wi-Fi Based Robotic Car". The methodology involved designing and building an autonomous Wi-Fi-based robotic car that can be remotely controlled and monitored.
- The hardware components were assembled and tested using tools and equipment such as soldering irons, multimeters, and power supplies.
- The software was developed using the Arduino IDE and the ESP32-CAM module was programmed using an external device called Compim.
- Proteus simulation was used to test the circuit design before building the actual hardware.

□ The robotic car was programmed to detect and avoid obstacles in real-time, and the user could control the car through a web-based interface.

□ The web interface displayed real-time video stream captured by the camera and the status information of the robot such as its position, speed control and led intensity.

□ The results showed that the project was successful in achieving its objectives.

□ Testing and optimization is carried out to ensure that the robotic car operates reliably and accurately, and that the user interface is responsive and user-friendly. Finally, the project is documented, including the design, methodology, and results.

3.5 TRADEOFF

There are several tradeoffs that need to be considered when designing the monitoring and controlling system for the Wi-Fi based robotic car. These include:

□ Power consumption vs. processing power: Increasing the processing power of the control unit may increase power consumption, which may reduce the battery life of the car. Balancing these factors will be important to ensure optimal performance.

□ Wireless range vs. reliability: Increasing the wireless range of the system may reduce reliability, as the signal may become weaker and more prone to interference over longer distances. Finding the optimal range will be important to ensure that the system is both reliable and effective.

3.6 CONSTRAINTS IN THE PROJECT

The following are some of the constraints that must be considered in the development of an automatic monitoring and controlling system for Wi-Fi based robotic cars:

- Connectivity Issues: The project relies on a stable and strong wireless connection between the car and the controlling device. However, wireless connectivity can be disrupted by various factors such as distance, interference, and obstacles. These factors can result in delayed response times or lost connections, which can affect the car's performance and safety.
- Size and weight: The system should be small and lightweight to minimize the impact on the performance of the robotic car.
- Power Consumption: The project involves using various components such as motors, sensors, and a microcontroller, which require significant power to operate. Therefore, the system's power consumption must be optimized to ensure that it operates efficiently and effectively without draining the battery quickly.
- Safety: The project involves building a robotic car that moves autonomously and can potentially cause harm to people or objects in its path. Therefore, safety must be a top priority in the design and implementation of the system. Safety features such as emergency stops, obstacle avoidance, and object detection must be included to ensure that the car operates safely and does not cause harm.
- Cost: The system should be cost-effective to ensure that it is affordable for consumers.

3.7 THE ENGINEERING STANDARDS

For this project, the following IEEE standards are relevant:

1. IEEE 802.11: This standard is relevant because it defines the specifications for wireless local area networks (WLANs), including Wi-Fi. The Wi-Fi module used in the project for communication between the robotic car and its control system operates based on this standard.
2. IEEE 802.15.4: This standard is relevant because it defines the specifications for low-rate wireless personal area networks (WPANs), which can be used for communication between the robotic car and its control system. This standard can also be used to define the communication protocol between the robotic car and other devices, such as sensors or actuators.
3. IEEE 802.3: This standard is relevant because it defines the specifications for Ethernet networks, which can be used for communication between the robotic car and its control system. This standard can be used to define the communication protocol between the robotic car and a remote control station, such as a laptop or a mobile device.
4. IEEE 1149.1: This standard is relevant because it defines the specifications for boundary-scan testing, which can be used to test and debug the hardware components of the robotic car. This standard can be used to ensure that the hardware components of the robotic car are functioning correctly before testing and deploying the system.
5. IEEE 1588: This standard is relevant because it defines the specifications for time synchronization in networked systems, which can be used to ensure accurate timing between the robotic car and its control system. This standard can be used to ensure that the robotic car is accurately synchronized with the control system, which is essential for the proper functioning of the system.

CHAPTER- 4

SIMULATION DETAILS

4.1 DEVELOPMENT SETUP

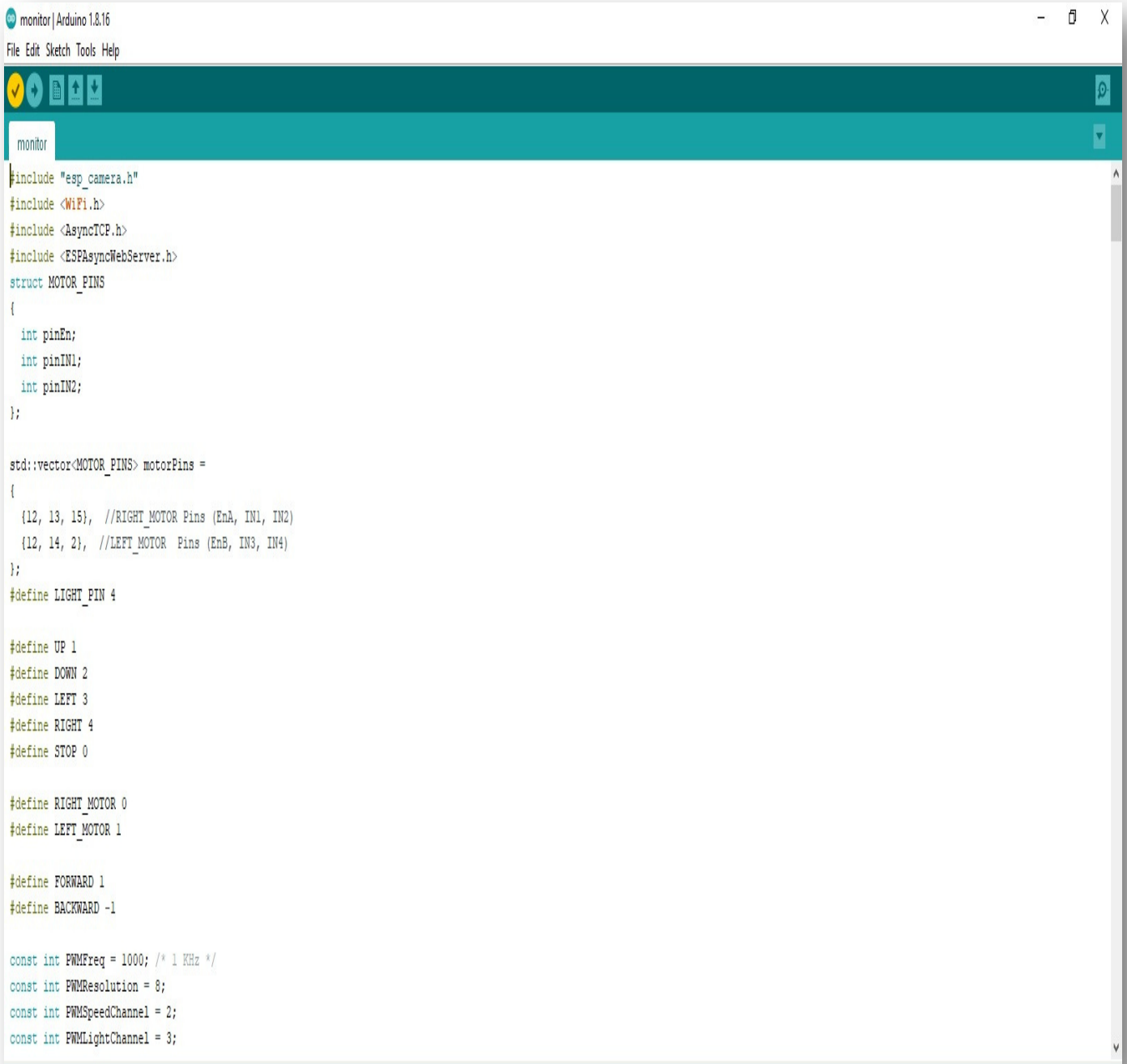
The development of the "Automatic Monitoring and Controlling of WiFi-Based Robotic Car" project involves several stages, including hardware and software development, testing, and integration.

4.1.1 SOFTWARE DEVELOPMENT

The software development involves the programming of the microcontroller and the development of the user interface and control system. The development setup includes a computer with integrated development environment (IDE) software, such as Arduino or Python, and the necessary libraries and tools for programming the microcontroller and developing the user interface. To program the microcontroller, developers typically use an IDE software like Arduino or Python, which provides a suite of tools for writing, compiling, and uploading code to the microcontroller. These IDEs often come with built-in libraries for common tasks, such as interfacing with sensors or controlling actuators.

In addition to the IDE, developers also need to consider the hardware components they will be using to interface with the microcontroller. This may include sensors, motors, and other electronic components. Depending on the application, developers may need to develop custom drivers or libraries to interface with these components.

Finally, the user interface and control system development are crucial for creating a usable and intuitive product. This often involves developing a graphical user interface (GUI) for users to interact with the device, as well as a control system that processes inputs from the user and communicates with the microcontroller to



```
monitor | Arduino 1.8.16
File Edit Sketch Tools Help

#include "esp_camera.h"
#include <WiFi.h>
#include <AsyncTCP.h>
#include <ESPAsyncWebServer.h>
struct MOTOR_PINS
{
    int pinEn;
    int pinIN1;
    int pinIN2;
};

std::vector<MOTOR_PINS> motorPins =
{
    {12, 13, 15}, //RIGHT_MOTOR Pins (EnA, IN1, IN2)
    {12, 14, 2},  //LEFT_MOTOR Pins (EnB, IN3, IN4)
};
#define LIGHT_PIN 4

#define UP 1
#define DOWN 2
#define LEFT 3
#define RIGHT 4
#define STOP 0

#define RIGHT_MOTOR 0
#define LEFT_MOTOR 1

#define FORWARD 1
#define BACKWARD -1

const int PWMFreq = 1000; /* 1 KHz */
const int PWMResolution = 8;
const int PWMSpeedChannel = 2;
const int PWMLightChannel = 3;
```

Fig: 4.1 ARDUINO CODE

The Arduino Integrated Development Environment (IDE) is a software application used to write and upload code to Arduino boards. It is a cross-platform

application that runs on Windows, Mac OS X, and Linux operating systems. The IDE provides a simple and easy-to-use interface for programming the Arduino boards, with features such as syntax highlighting, code completion, and a built-in serial monitor for debugging.

The Arduino IDE is open source software, which means that the source code is freely available for developers to modify and distribute. It supports multiple programming languages, including C, C++, and a simplified version of the Processing language, which is a programming language used for creating visual arts and interactive installations.

The Arduino code for this project is written in the Arduino Integrated Development Environment (IDE), which allows for easy programming and uploading of the code to the board. The code is written in the C++ programming language and uses various libraries, such as the WiFi and Servo libraries, to communicate with the WiFi module and control the motors and servos.

In this project, the ESP32 board serves as the main control unit for the robotic car. It is responsible for receiving commands from the control system through the WiFi network, processing these commands, and controlling the motors and servos of the car accordingly.

The ESP32 board is equipped with a built-in WiFi module, which allows it to communicate with the control system through a wireless network. It also has a sufficient number of input/output (I/O) pins to connect and control various hardware components, such as sensors, motors, and servos.

4.1.2 ARDUINO CODE

```

void setupPimodes()
{
    //Set up PWM
    ledcSetup(PWMSpeedChannel, PWMFreq, PWMResolution);
    ledcAttachPin(motorPins[1].pinEn, PWMSpeedChannel);

    for (int i = 0; i < motorPins.size(); i++)
    {
        pinMode(motorPins[i].pinEn, OUTPUT);
        pinMode(motorPins[i].pinIN1, OUTPUT);
        pinMode(motorPins[i].pinIN2, OUTPUT);

        /* Attach the PWM Channel to the motor enb Pin */
        ledcAttachPin(motorPins[i].pinEn, PWMSpeedChannel);
    }

    pinMode(LIGHT_PIN, OUTPUT);
    ledcAttachPin(LIGHT_PIN, PWMLightChannel);
}

Serial.begin(115200);

Serial.printf("Got value as %d\n", inputValue);
switch(inputValue)
{
    case UP:
        rotateMotor(RIGHT_MOTOR, FORWARD);
        rotateMotor(LEFT_MOTOR, FORWARD);
        break;

    case DOWN:
        rotateMotor(RIGHT_MOTOR, BACKWARD);
        rotateMotor(LEFT_MOTOR, BACKWARD);
        break;

    case LEFT:
        rotateMotor(RIGHT_MOTOR, FORWARD);
        rotateMotor(LEFT_MOTOR, BACKWARD);
        break;

    case STOP:
        rotateMotor(RIGHT_MOTOR, STOP);
        rotateMotor(LEFT_MOTOR, STOP);
        break;

    default:
        rotateMotor(RIGHT_MOTOR, STOP);
        rotateMotor(LEFT_MOTOR, STOP);
        break;
}

```

Fig. 4.2 SOME SPECIFIC ARDUINO CODE

Firstly, the `setUpPimodes()` function is used to set the motor pins as output. This function is necessary to configure the input and output pins on the Arduino board, specifically for controlling the motors of the robotic car. The `pinMode()` function is called within the `setUpPimodes()` function to set the motor pins as outputs.

Secondly, the `serial.begin(115200)` function is used to initialize the serial communication between the Arduino board and the controlling device. This function sets the data transfer rate for the serial communication, which is essential for sending and receiving data between the Arduino board and the controlling device. In this case, a baud rate of 115200 is used for the serial communication.

4.1.3 PROTEUS SIMULATION SETUP

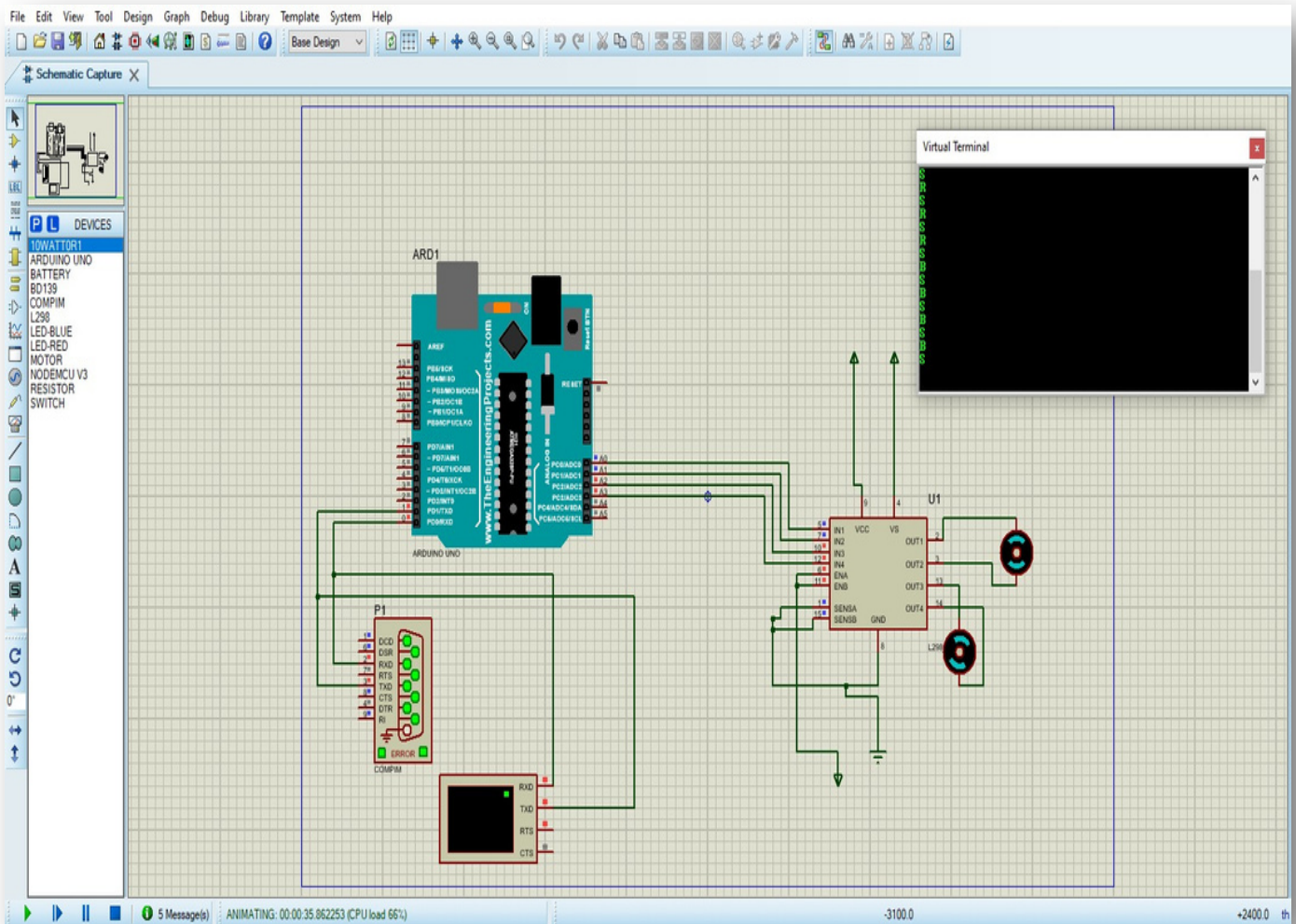


Fig: 4.3 PROTEUS SIMULATION

Proteus is a software tool used for simulation and design of electronic circuits. It is widely used by electronic engineers and students to design and simulate circuits before building them in real life. Proteus allows users to design circuits using a wide range of electronic components, simulate them and test their functionality before moving to actual implementation.

In addition to circuit simulation, Proteus also allows users to design and test

microcontroller-based projects using popular microcontroller boards, such as Arduino and PIC. It offers a wide range of built-in virtual components for these boards that can be used for testing and debugging purposes.

In this project, Proteus software can be used to simulate the circuit design and test the functionality of the hardware components before building the actual robotic car.

The simulation can be created by first selecting the required components from the Proteus library, such as the Arduino Uno board, motors, sensors, and Compim module. Then, these components can be placed on the virtual breadboard and connected using virtual wires.

Compim is a virtual serial port emulator that is used to establish communication between Proteus simulation software and an external microcontroller.

In the case of the "automatic Monitoring and controlling of wifi based robotic car" project, Compim can be used to simulate the communication between the

Arduino board and the Proteus simulation. The Arduino board can be programmed to receive commands from the WiFi module and control the L298 motor driver, which in turn controls the motors of the robotic car.

The Proteus simulation can be set up to send commands to the virtual serial port provided by Compim, which are then received by the Arduino board through the physical USB port. The Arduino board can then interpret these commands and control the motors accordingly in the simulation.

The simulation can be used to identify any issues or errors in the circuit design before building the actual hardware. This allows for the identification and resolution of problems before any physical components are assembled, saving time and resources.

4.2 TESTING

The testing involves the verification and validation of the hardware and software components to ensure they meet the project requirements. The testing setup includes tools and equipment for measuring and analyzing the performance and functionality of the system

TABLE: 4.1 MOTION OF THE CAR UNDER CONDITIONS

Condition of motor	D1 LEFT	D2 LEFT	D3 RIGHT	D4 RIGHT		
Left & Right Motor	0	0	0	0		
Stop						
Forward			Anticlockwise	Anticlockwise	Anticlockwise	Anticlockwise
Back	Clockwise	Clockwise	Clockwise	Clockwise		
Left	Anticlockwise	Anticlockwise	Clockwise	Clockwise		
			Right	Clockwise	Clockwise	Anticlockwise

The table 4.1 shows the direction and motion of the robotic car under different conditions of the motors. The motor can be moved in four directions: left, right, forward, and back, and can also be rotated in a clockwise or anticlockwise direction. The table specifies the corresponding motion of the car for each motor direction and rotation, enabling the user to control the movement of the car in a desired manner. The table can also help in troubleshooting if there are any issues with the motors or the car's movement.

TABLE: 4.2 EXPERIMENT CONDUCTED TO TEST THE PERFORMANCE

Experiment number	Time	Distance	Speed
I	17.06	10	0.58 m/s
II	16.66	10	0.6 m/s
III	17.02	10	0.587 m/s
IV	16.76	10	0.59 m/s
V	16.82	10	0.59m/s

The table 4.2 shows a list of experiments conducted to test the performance of the robotic car under same distance conditions. Each experiment is numbered, and the time and speed taken by the car to move from its starting position to the endpoint are recorded for each speed level. The speed levels can be adjusted by varying the voltage supplied to the motors, and the direction of motion can be controlled using the Arduino code. The table may also include columns to indicate the motor speed, the direction of motion (forward, backward, left, right), and any additional notes or observations made during the experiment. The data collected in the table can be used to analyze the performance of the robotic car and optimize its design and control parameters for better performance.

CHAPTER 5

PROJECT DESCRIPTION

5.1 THE BLOCK DIAGRAM OF EXISTING SYSTEM

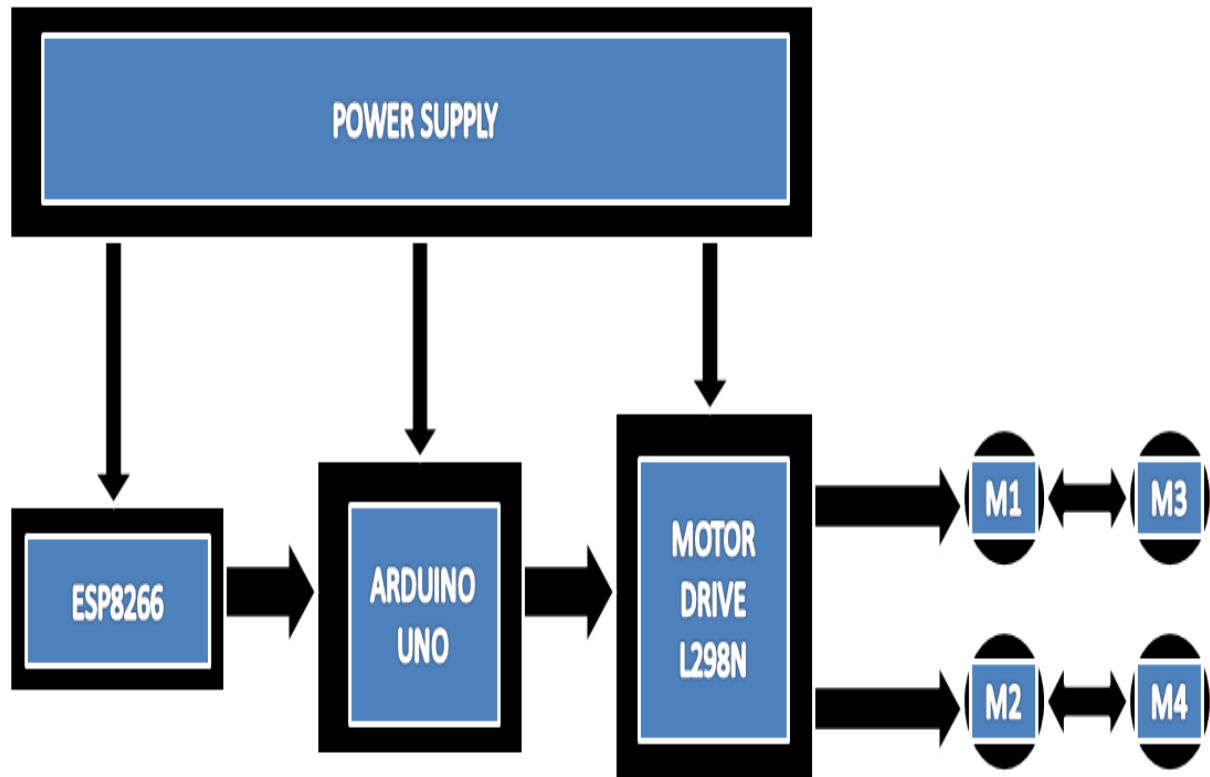


Fig 5.1 Block diagram of Existing System Using Arduino

Working of the Wi-Fi controlled robot is very easy, we just need to Drag or Slide the joystick in the direction, where we want to move the Robot. If we want to move the Robot in Forward direction then we need to Drag the Joystick

‘circle’ in Forward direction. Like we can move the Robot in Left, Right and Backward direction by Dragging the Joystick in respective direction. Now as soon

as we release the Joystick, it will come back to centre and Robot. Blynk App sends

values from Two Axis Joystick to Arduino through Wi-Fi medium. Arduino receive the values, compare them with predefined values and move the robot accordingly in that direction.

5.2 THE BLOCK DIAGRAM OF PROPOSED SYSTEM

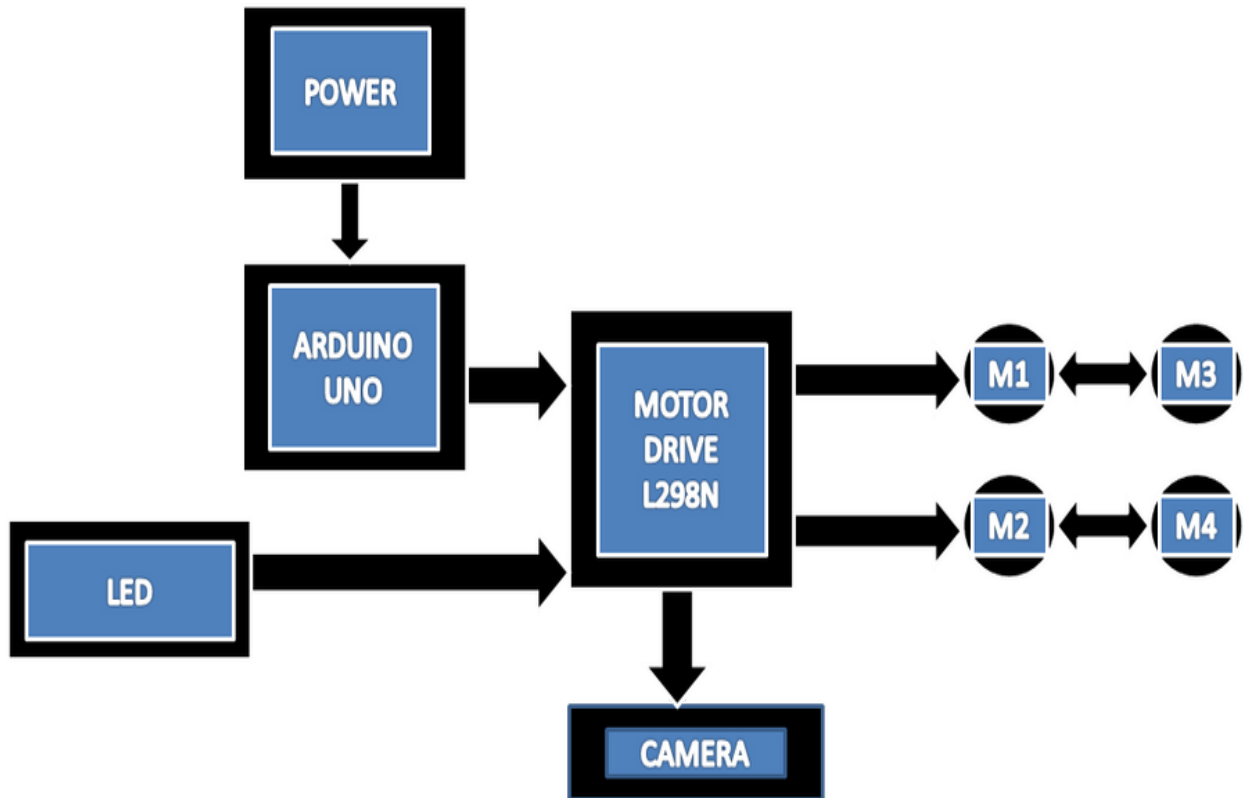


Fig 5.2 Block diagram of Proposed System Using ESP32 CAM

The objective of the proposed system is to address the challenge of monitoring and controlling a robotic car that is connected to wifi. This will be achieved by utilizing an ESP32 cam module, and sensors. The system is designed to keep track of the robotic car's movements and enable real-time control of its actions via a smartphone application. With this system, users will have the ability to monitor and manage the robotic car's activities remotely, enhancing its functionality and versatility.

5.3 CIRCUIT DIAGRAM

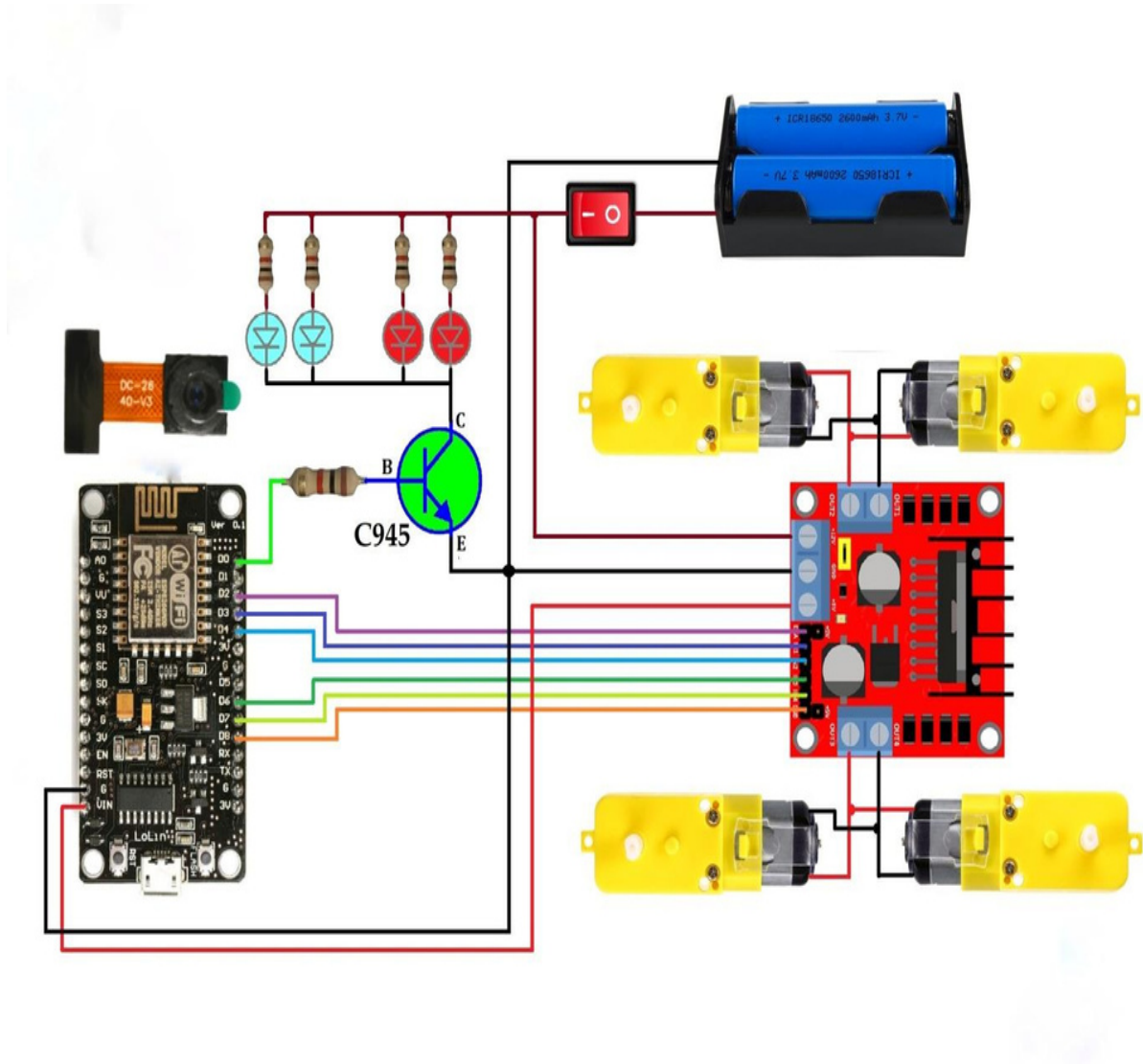


Fig. 5.3 Circuit diagram for Proposed Modified Circuit Diagram of Wi-Fi Robot car

The circuit diagram includes the following components:

- ESP32-CAM: This is a development board that combines an ESP32 microcontroller and a camera module. It is used for wireless communication and image processing.

- L298N motor driver: This is a dual H-bridge motor driver module that can drive up to two DC motors. It can control the speed and direction of the motors.

- TTP223 FTDI converter or NodeMCU: This is a USB-to-serial converter that can be used to program and communicate with the ESP32-CAM. Alternatively, a NodeMCU can be used as a microcontroller and Wi-Fi module.

- DC gear motor: This project uses four DC gear motors to drive the robotic car. These motors convert electrical energy into mechanical energy to move the car.

- Resistor: A resistor is used to limit the current flowing through an LED.

- LED: An LED is a light-emitting diode that can be used as a visual indicator.

- Breadboard: A breadboard is a solderless prototyping board that allows for easy connection and experimentation with electronic components.

- BD139 NPN transistor: This component is used to amplify the signal from the ESP32-CAM or NodeMCU to control the L298N motor driver.

- Battery: The project uses two 3.7V batteries in series to power the system.

- VPC card board: This component is used to provide secure communication between the robotic car and the remote control station over the internet

5.4 HARDWARE REQUIREMENTS

- VPC Card board 5mm
- Breadboard
- DC Gear Motor x 4
- Esp32 cam
- Arduino Node-Mcu
- L298 Motor Driver
- BD139 NPN Transistor
- 1k Resistor x 3
- 100R Resistor x 2
- White LED x 2, Red LED x 2
 - 4Pcs Smart Robot Car Tyers Wheels
- Male to Male jumper Wires
- Male to Female jumper Wires
- On/Off Switch
- 18650 Battery Holder – 2 Cell
- 18650 Battery Cell 3.7V x 2

5.5 HARDWARE DESCRIPTION

5.5.1 DC GEAR MOTOR

A gear motor is an all-in-one combination of a motor and gearbox. The addition of a gear head to a motor reduces the speed while increasing the torque

output. The most important parameters in regards to gear motors are speed (rpm),

torque (lb-in) and efficiency (%). In order to select the most suitable gear motor for

your application you must first compute the load, speed and torque requirements for your application. IS25Products offers a variety of Spur Gear Motors, Planetary Gear Motors and Worm Gear Motors to meet all application requirements.



Fig. 5.4 Dc gear motor

5.5.2 BD139 NPN TRANSISTOR

A Bipolar NPN transistor like BD139 is available in the SOT-32 plastic package. This kind of transistor is used in different electronic circuits because of less cost and high collector current. This transistor can drive up to 1.5A loads like motors, high power LEDs, relays, etc.

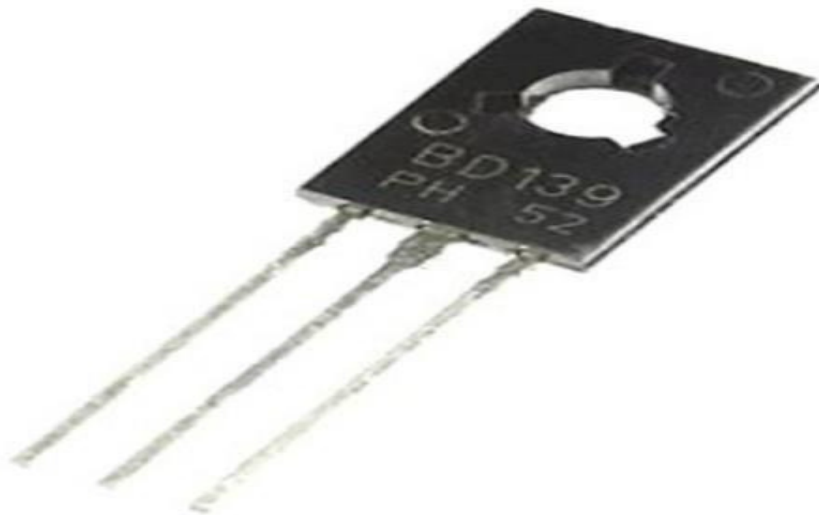


Fig.5.5 BD139 NPN Transistor

5.5.3 L298 MOTOR DRIVER

The L298 Driver is a high voltage, high current dual full bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together; the corresponding external terminal can be used for the connection of an external sensing resistor.

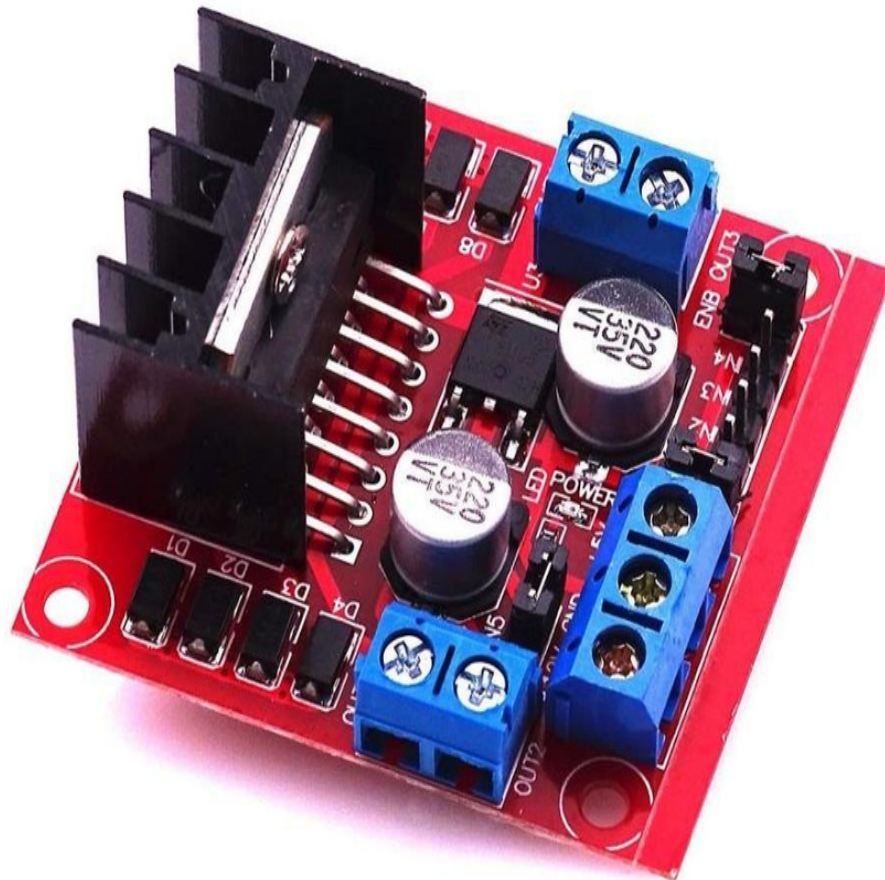


Fig: 5.6 L298 Motor Driver

4.5.4 ARDUINO NODEMCU

NodeMCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. The term "NodeMCU" by default refers to the firmware rather than the development kits. The firmware uses the Lua scripting language.

Since NodeMCU is an open-source platform, its hardware design is open for edit/modify/build. NodeMCU Dev Kit/board consist of ESP8266 wifi enabled chip. The ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. For more information about ESP8266, you can refer to the ESP8266 WiFi

Module.

There is Version2 (V2) available for NodeMCU Dev Kit i.e. NodeMCU Development Board v1.0 (Version2), which usually comes in black colored PCB.



Fig: 5.7 NodeMCU Development Board/kit v1.0 (Version2)

5.5.5 ESP32 CAM MODULE

ESP32-CAM is a low-cost ESP32-based development board with onboard camera, small in size. It is an ideal solution for IoT application, prototypes constructions and DIY projects.

The board integrates WiFi, traditional Bluetooth and low power BLE , with 2 high- performance 32-bit LX6 CPUs. It adopts 7-stage pipeline architecture, on-chip sensor, Hall sensor, temperature sensor and so on, and its main frequency adjustment ranges from 80MHz to 240MHz.

Fully compliant with WiFi 802.11b/g/n/e/i and Bluetooth 4.2 standards, it can be used as a master mode to build an independent network controller, or as a slave to other host MCUs to add networking capabilities to existing devices.

ESP32-CAM can be widely used in various IoT applications. It is suitable for home smart devices, industrial wireless control, wireless monitoring, QR wireless identification, wireless positioning system signals and other IoT applications. It is an ideal solution for IoT applications.

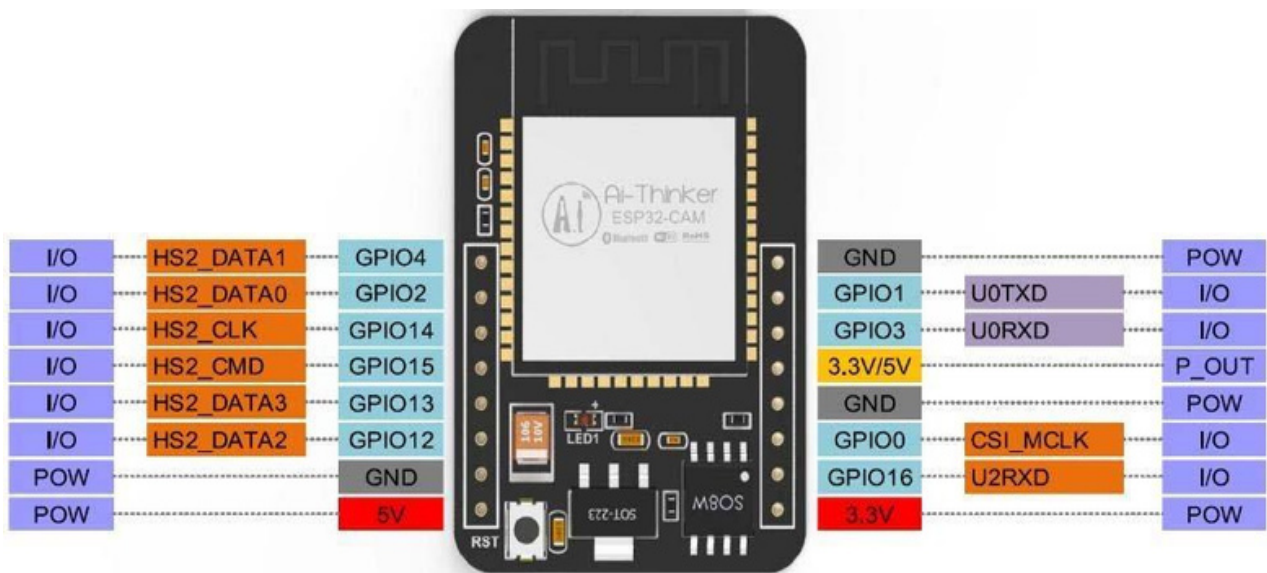


Fig: 5.8 Schematic Diagram

5.5.6 18650 Battery Cell 3.7V

An 18650 is a lithium ion rechargeable battery. Their proper name is “18650 cell”. The 18650 cell has voltage of 3.7v and has between 1800mAh and 3500mAh (mili-amp-hours).

18650s may have a voltage range between 2.5 volts and 4.2 volts, or a charging voltage of 4.2 volts, but the nominal voltage of a standard 18650 is 3.7 volts. There are two types; protected and unprotected. We absolutely recommend protected cell 18650 batteries. Protected cells include a protection circuit that stops the cell from being overcharged.

Unprotected cells can be overcharged and burst or potentially cause a fire unless there are specific electronics to protect the battery. The popular LG HG2 and Samsung 25r are both UNPROTECTED batteries, only use them in a device designed to use unprotected 18650s

The average 18650 battery charge time is about 4 hours. Charge time can vary with amperage and voltage of the charger and the battery type.



Fig: 5.9 18650 Battery Cell

5.6 LED BRIGHTNESS INTENSITY

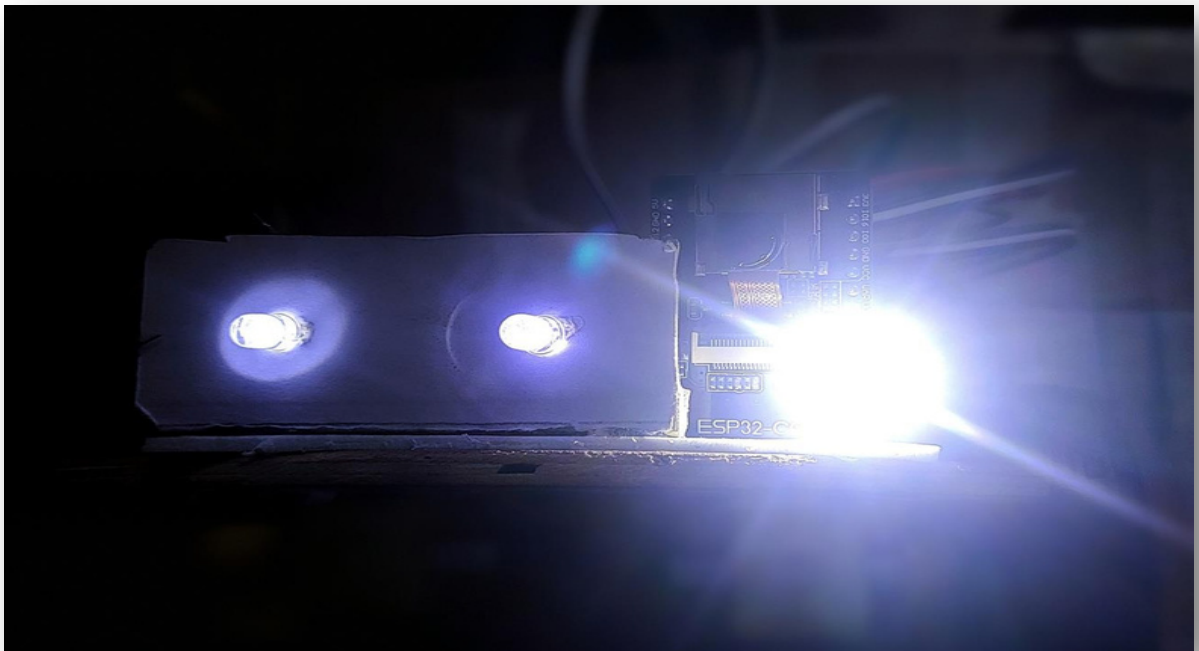
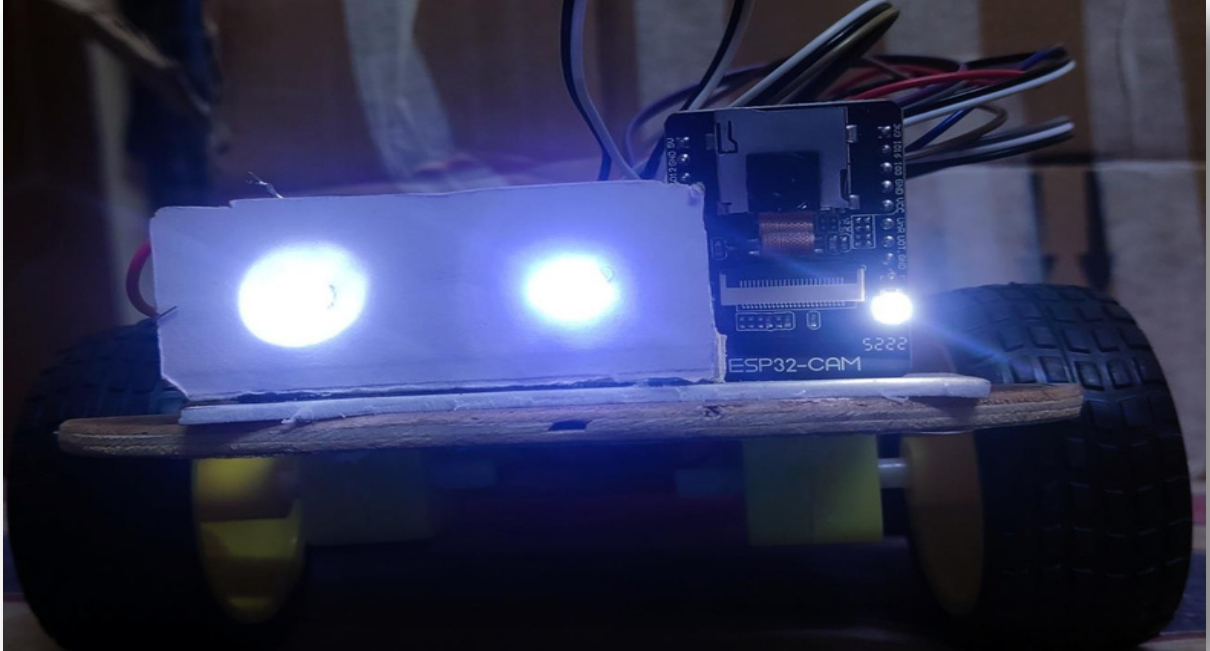


Fig: 5.10 Shows The Variation Of Led Intensity

The ESP32-CAM module has an onboard LED that can be used for various purposes, such as indicating the status of the module or providing additional lighting for the camera. The brightness intensity of this LED can be controlled using the PWM (Pulse Width Modulation) technique.

PWM is a technique used to control the amount of power delivered to a device by adjusting the width of the electrical pulse. In the case of the LED on the ESP32-CAM module, this means that the brightness intensity of the LED can be adjusted by varying the duty cycle of the PWM signal.

The duty cycle is the ratio of the time the signal is high compared to the total period of the signal. By increasing or decreasing the duty cycle of the PWM signal, the amount of power delivered to the LED can be adjusted, which in turn affects the brightness intensity of the LED.

The LED intensity of the ESP32-CAM module can be controlled using the LED_BUILTIN pin and PWM (Pulse Width Modulation) technique. The LED_BUILTIN pin is connected to the on-board LED of the module, which can be used to adjust the brightness intensity.

The PWM technique allows controlling the power supplied to the LED, which in turn changes the brightness of the LED. The LED intensity can be varied by sliding a range of values from 0 to 255, where 0 represents the lowest intensity or completely turned off, and 255 represents the highest intensity or fully turned on. By sliding the range of values, the brightness of the LED can be adjusted to a desired level.

5.7 WORKING OF THE WIFI BASED ROBOTIC CAR



- ☐ The user will first connect the smartphone to the wifi network of the
- ☐ robotic car.
- ☐ The user will first connect the smartphone to the wifi network of the robotic car. The app will then display real-time images from the camera
- ☐ and provide control options for the car's movements.
- ☐ The user can then use the app to control the car's movements by sending
- ☐ commands to the nodeMCU.

The nodeMCU will receive the commands from the app and control the car's

- ☐ movements accordingly.

The sensors will be used to monitor the car's movements and detect obstacles

in its path. If an obstacle is detected, the nodeMCU will stop the car and send a

notification to the smartphone app.

The user can then use the app to control the car's movements around the obstacle and continue the journey.

5.8 RESULT

We have successfully built an autonomous Wi-Fi-based robotic car that can be remotely controlled and monitored. The car is equipped with sensors that allow it to detect and avoid obstacles in real-time, making it an autonomous vehicle. The car can be controlled by a user through a web-based interface, which provides commands to move it forward, backward, turn left, turn right, and stop.

The car also has a camera that captures a live video stream, which is displayed on the web-based interface for the user to monitor the environment where the car is operating. Additionally, the interface provides information on the car's status, such as LED brightness intensity and the speed of the motor. Furthermore, the intensity of the LED lights can be controlled through a slide from the web-based interface.

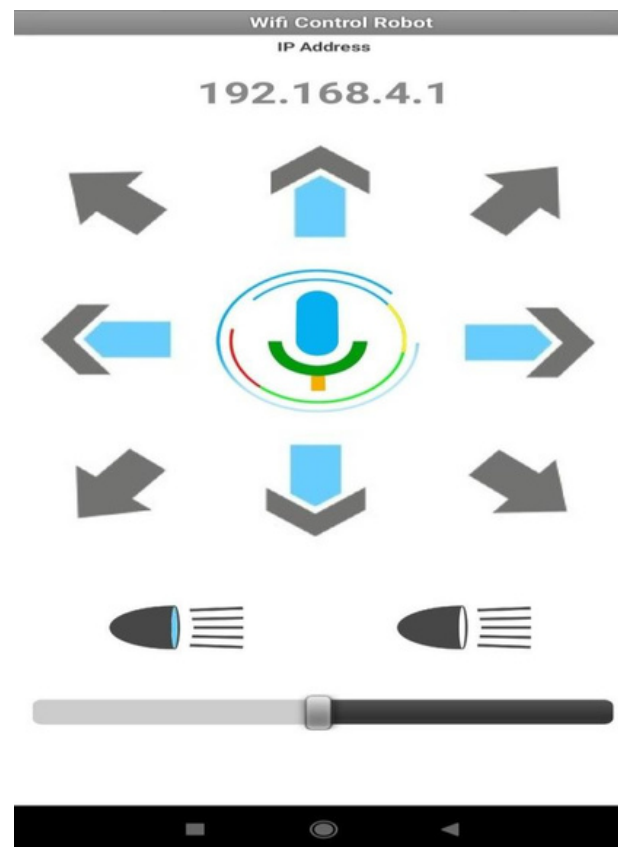


Fig: 5.11 Controlling page

5.9 PROTOTYPE DISCUSSION

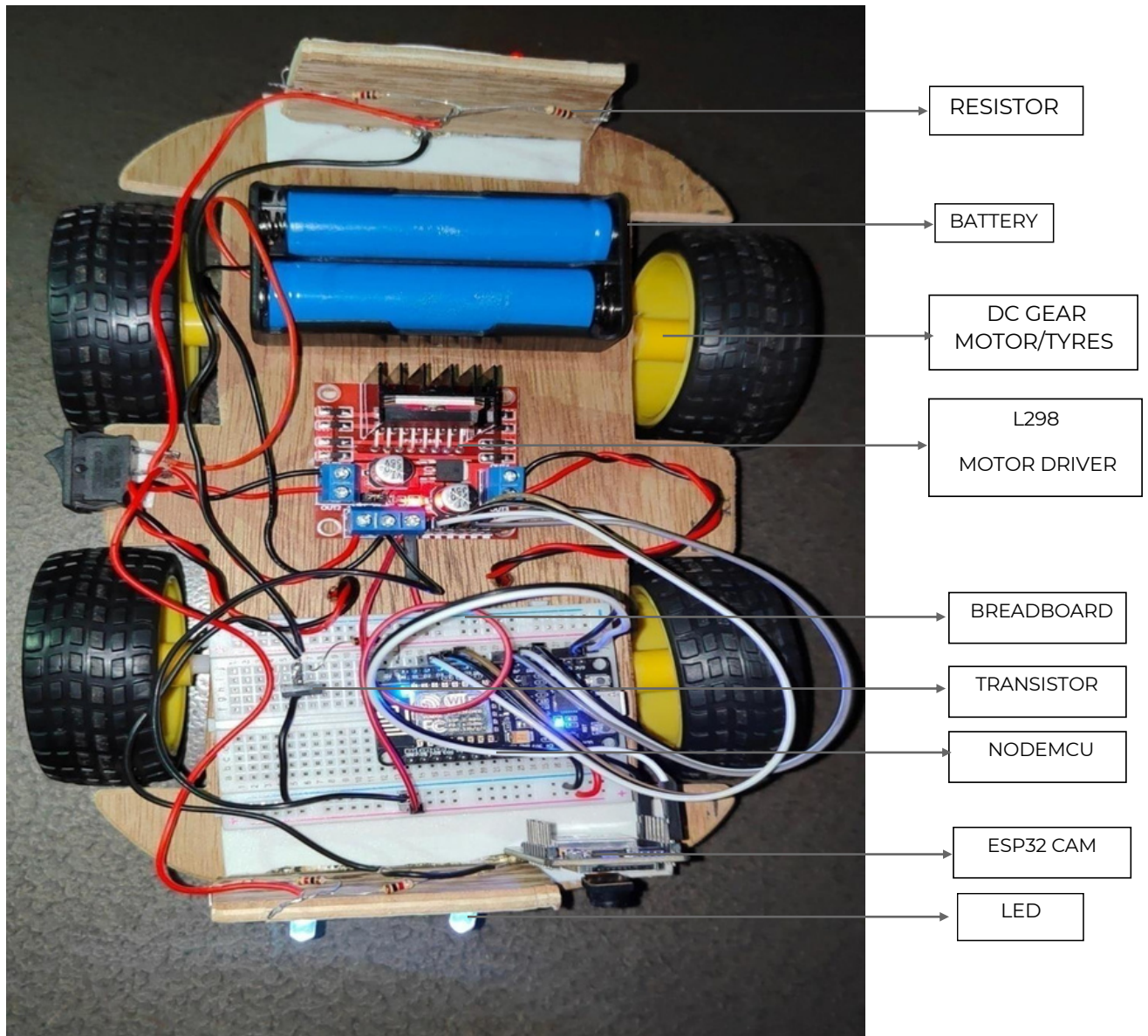


Fig: 5.12 Label on the Prototype Model

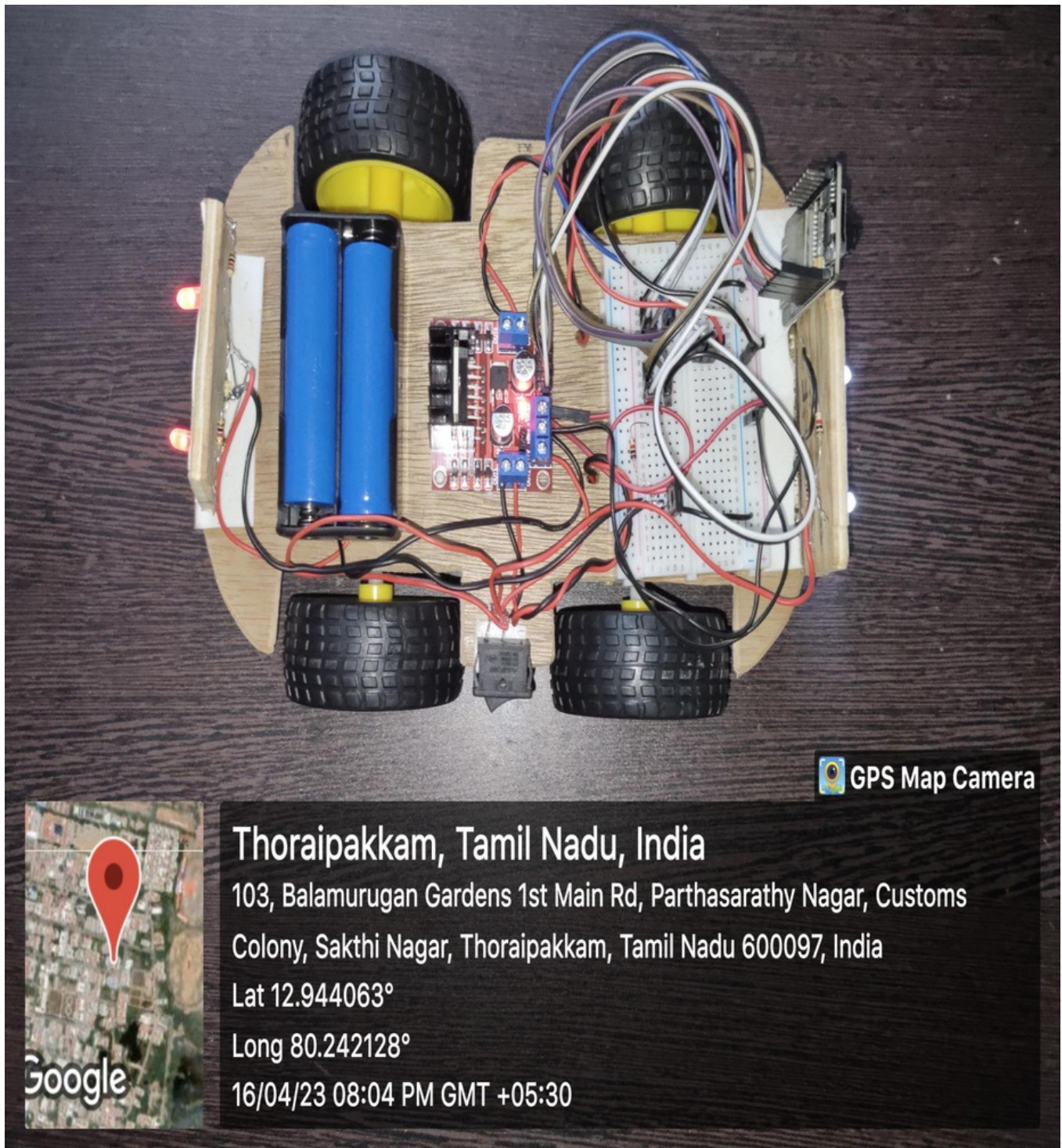


Fig: 5.13 Geo Tagged Model

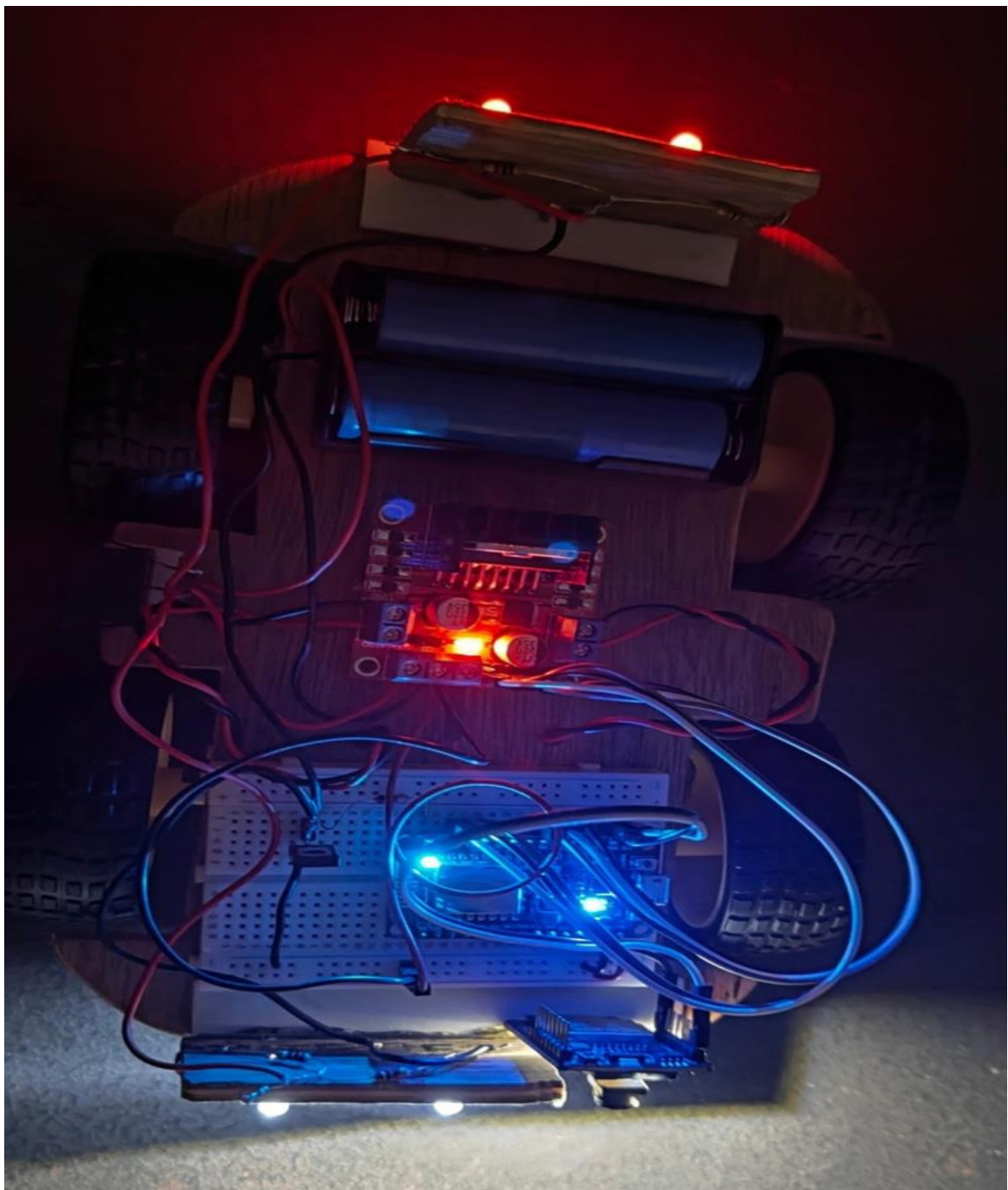


Fig: 5.14 Shows The Prototype mode while switch on

The project "Automatic Monitoring and Controlling of Wi-Fi-Based Robotic Car" aims to build an autonomous robotic car that can be remotely controlled and monitored through a web-based interface. The robotic car uses an ESP32-CAM module for wireless communication and a combination of sensors to detect and avoid obstacles in real-time. The web-based interface allows the user to control the movement of the robot and view the real-time video stream captured by the camera. The project has been successfully implemented, and the results show that the robotic car is capable of moving in all directions and avoiding obstacles in real-time. The web-based interface provides a user-friendly way to control the robot and view its status information.

In terms of hardware development, the selection and integration of various components, such as microcontrollers, motors, sensors, and WiFi modules, were carefully done to ensure compatibility and functionality. The development setup included tools and equipment such as soldering irons, multimeters, and power supplies for assembling and testing the hardware.

In terms of software development, the Arduino IDE was used to write the code in the C++ programming language, which uses various libraries to communicate with the WiFi module and control the motors and servos. Proteus software was used to simulate the circuit design and test the functionality of the hardware components before building the actual robotic car.

5.10 THE ENGINEERING STANDARDS

For this project, the following IEEE standards are relevant:

1. IEEE 802.11: This standard is relevant because it defines the specifications for wireless local area networks (WLANs), including Wi-Fi. The Wi-Fi module used in the project for communication between the robotic car and its control system operates based on this standard.
2. IEEE 802.15.4: This standard is relevant because it defines the specifications for low-rate wireless personal area networks (WPANs), which can be used for communication between the robotic car and its control system. This standard can also be used to define the communication protocol between the robotic car and other devices, such as sensors or actuators.
3. IEEE 802.3: This standard is relevant because it defines the specifications for Ethernet networks, which can be used for communication between the robotic car and its control system. This standard can be used to define the communication protocol between the robotic car and a remote control station, such as a laptop or a mobile device.
4. IEEE 1149.1: This standard is relevant because it defines the specifications for boundary-scan testing, which can be used to test and debug the hardware components of the robotic car. This standard can be used to ensure that the hardware components of the robotic car are functioning correctly before testing and deploying the system.
5. IEEE 1588: This standard is relevant because it defines the specifications for time synchronization in networked systems, which can be used to ensure accurate timing between the robotic car and its control system. This standard can be used to ensure that the robotic car is accurately synchronized with the control system, which is essential for the proper functioning of the system.

CHAPTER 6

CONCLUSION & FUTURE SCOPE

6.1 CONCLUSION

In conclusion, the project successfully achieved its objective of building an autonomous Wi-Fi-based robotic car that can be remotely controlled and monitored. The project demonstrates the potential of combining hardware and software development to build innovative and useful products. The project can be further enhanced by adding more features and capabilities to the robotic car, such as object recognition and autonomous navigation.

The project has several potential applications in various fields, such as surveillance, search and rescue operations, and remote monitoring. The project can also be extended by adding additional sensors and functionalities, such as GPS and machine learning algorithms for more advanced obstacle detection and decision-making. The use of Wi-Fi-based robotic cars has numerous potential applications in various industries. The integration of artificial intelligence and machine learning can further enhance the capabilities of the system, making it more versatile and efficient. As technology continues to develop, the potential for the use of robotic cars is expected to increase, leading to even more applications and benefits for society.

The car is easy to control and monitor remotely using a smartphone or any other device with internet connectivity. The user interface is intuitive and user-friendly, and the touch screen controls make it simple to control the car's movement. The real-time camera feed and sensor data provide valuable information about the surroundings, making it easier.

Overall, this project has provided an excellent opportunity to apply theoretical knowledge and practical skills in robotics, microcontrollers, and wireless communication.

6.2 FUTURE SCOPE

The automatic monitoring and controlling of Wi-Fi-based robotic car project has the potential for several future developments and improvements. Some of them are:

- Object Recognition: The current project detects obstacles using an ESP32 Cam module. However, object recognition techniques such as computer vision can be employed to improve obstacle detection accuracy and distance measurement.
- Autonomous Navigation: The current project is remotely controlled by the user. However, with the integration of sensors like GPS and inertial measurement units, the robotic car can navigate autonomously.
- Wireless Charging: The project currently requires manual battery charging. The integration of wireless charging technology can enable the robotic car to charge automatically, improving its operating time.
- Gesture Control: Integration of gesture recognition technology can allow the user to control the robotic car with hand gestures, which can be useful in situations where using a web-based interface is not possible.
- Machine Learning: Machine learning algorithms can be employed to enhance the obstacle avoidance system's decision-making capabilities and make it more efficient.

Overall, the project has significant potential for future developments and can be further optimized to achieve greater performance and functionality.

6.3 IMPLEMENTATION ISSUES

The problem statement is that the ESP32-CAM module does not have a USB port to directly upload and run the code onto the board. This presents a challenge for programming the board as other methods must be used to upload the code onto the board, such as using an external programmer or module. Without a USB port, it may also be difficult to debug and troubleshoot any issues that arise during development.

the ESP32-CAM module is used as a critical component for the robotic car's functionality, and it needs to be programmed to work with the other hardware components. However, the Proteus simulation software does not have an option to run the code directly from the Arduino IDE. This makes it difficult to test and debug the code in Proteus without an external device such as a Compim. The absence of a USB port on the ESP32-CAM module adds to the challenge of programming the module and testing it in the Proteus simulation environment. Therefore, finding a way to implement an external device such as a Compim to program the ESP32-CAM module and simulate the code in Proteus is essential for the success of the project.

This adds an extra step in the development process and requires additional hardware, which can be inconvenient and time-consuming

REFERENCES

- [1] Al- Sahib Nabeel.K& Azeez.Z Mohammed created "Internet Remote Control Interface for a Multi-Purpose Mobile Robot".
- [2] Andreasson.J, Gavert.M created "The Vehicle Dynamics Library Overview and Applications Modelon".
- [3] Cuno Pfister created "Getting started with Internet of Things".
-] Joy deep Biswas created "Wifi Localization and Navigation for Autonomous Indoor Mobile Robots".
- [4] Sudha Jamthe created "IoT Disruptions: The Internet of Things - Innovations & Jobs
- [5]] "Tom nguyen ,josh slonaker created "Semi Autonomous wireless control robot".
-] M.S.Sreejith, Steffy Joy created "Conceptual design of a wifi and gps based robotic library
- [6] using an intelligent system.
-] C.W.Warren , S.G.Buckley created "Mobile robotic over wifi network using arduino
- [8] technology".
- [7] Harshit Gulati, Shriyannsh Vaishya created "Bluetooth and wifi control rescue robot".
-] [9] Design of a Bluetooth Enabled Android Application for a Microcontroller Driven Robot By Vito M. Guardi,(May 2014).
- [10] Jeonghye Han, Jaeyeon Lee, and Youngjo Cho, "Evolutionary role model and basic emotions of service robots originated from computers," In the Proceedings of the IEEE International
- [11] Workshop on Robot and Human Interactive Communication (ROMAN), pp. 30-35, 2005
- Moonzoo Kim, KyoChul Kang, and Hyoungki Lee, "Formal Verification of Robot Movements - a Case Study on Home Service Robot SHR100," In the Proceedings of the IEEE International
- [12] Conference on Robotics and Automation (ICRA), pp. 4739-4744, 2005
- Yeong Che Fai, Shamsudin H.M. Amin, Norsheilab Faisal, J. Abu Bakar. "Bluetooth Enabled Mobile Robot." IEEE ICIT'02, Bangkok, THAILAND 2002.
- [13]

- [14] Andrey A. Loukianov, Hidenori Kimura, Masanori Sugisaka. "Implementing distributed control system for intelligent mobile robot" 8th International Symposium on Artificial Life and Robotics, Oita, Japan, January 24– 26 2003
- [15] T. Butnaru, F. Gîrbacia, F. Tîrziu and D. Talab, "Mobile Robot System Controlled Through Mobile Communications" Product Engineering, Springer, 2004
- Bischoff, R., and Graefe, V., "HERMES - a versatile personal robotic assistant,"
- [16] Proceedings of the IEEE, vol. 92, pp. 1759-1779, 2004.
- Dong To Nguyen, Sang-Rok Oh, and Bum-Jae You, "A framework for Internet- based
- [17] Interaction of humans, robots, and responsive environments using agent technology," IEEE Transactions on Industrial Electronics, vol. 52, Issue 6, pp. 1521-1529, 2005
- Gross, H.-M., Boehme, H.-J., and Wilhelm T., "Contribution to vision based localization,
- [18] Tracking and navigation methods for an interactive mobile service robot," In the Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, pp. 672– 677, 2001
- Lenser, S., and Veloso, M., "Sensor resetting localization for poorly modeled mobile robots,"
- [19] In the Proceedings of the International Conference on Robotics and Automation (ICRA), pp. 1225–1232, 2000.
- Taipalus, T., and Kazuhiro K., "Development of service robot for fetching objects in home
- [20] environment," In the Proceedings of the IEEE International Symposium on International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 08 Issue: 08 | Aug 2021 www.irjet.net p- ISSN: 2395-0072 © 2021, IRJET | Impact Factor value: 7.529 | ISO 9001:2008 Certified Journal | Page 1486 Computational Intelligence in Robotics and Automation (CIRA), pp. 451-456, 2005
- Yoshikazu K., Takayuki K., Yasuyuki S., Kiyoshi K., and Hiroshi I., "An Approach to
- [21] Integrating an Interactive Guide Robot with Ubiquitous Sensors," In the Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2500-2505, 2004
- Ricardo Tesoriero, José A. Gallud, María D. Lozano, Víctor M. R. Penichet "Tracking
- [22] Autonomous Entities using RFID Technology" IEEE Transactions on Consumer Electronics, Vol. 55, No. 2, MAY 2009
- Marcelo B. Perotoni, Beatriz E. Garibello, Silvio E. Barbin "An IEEE 802.11 Low Cost
- [23] Planner Antenna for a Mobile Robot", 2006
- Wi-Fi Robot For Video Monitoring & Surveillance System' by Pavan.C, Dr. B. Siva kumar,
- [24] International Journal of Scientific & Engineering Research Volume 3, Issue 8, August-2012 ISSN 2229-5518

