KALYANI GOVERNMENT ENGINEERING COLLEGE

Kalyani, Nadia -741235

KGEC ROBOTICS SOCIETY

Topic: Bluetooth Controlled Car

Name: Deep Rudra

Department: Electronics and Communication Engineering

Roll No.: 10200323022

Semester: 3rd

Date of submission: 04/10/2024

DECLARATION BY THE CANDIDATE

I hereby declare that the project report entitled **"Bluetooth Controlled Car"** submitted by me to the Kalyani Government Engineering College Robotics Society, Nadia, West Bengal, in partial fulfillment of the requirements for my internship, is a record of bonafide project work carried out by me under the guidance of my senior mentor, *Mr. Somaraho Chaudhary*, and the Joint Secretary *Mr. Aritra Biswas* and additional joint Secretary *Mr. Ajarul Miha*.

I further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other project showcase or similar programs

Kalyani, Nadia

Date: 04/10/2024

Signature of the Candidate
(Deep Rudra)

CERTIFICATE OF APPROVAL

This is to certify that Mr. Deep Rudra, an intern at the KGEC Robotics Society, pursuing a B. Tech in Electronics and Communication Engineering at Kalyani Government Engineering College, has successfully completed the project on **"Bluetooth Controlled Car."**

This project was carried out under my guidance and supervision, and he has provided a satisfactory account of it in his report.

Mr. Somaraho Chaudhary
(Electronics & IoT lead & Mentor)

Mr. Aritra Biswas
(Joint Secretary)
KGEC Robotics Society

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my senior, mentor and Electronics & IoT lead *Mr. Somaraho Chaudhary* and also to my senior and Joint Secretary *Mr. Aritra Biswas* for their able guidance and support in completing my Project. I would also like to extend my gratitude to our Principal, Dr. Sourav Kumar Das Sir for providing me and our Robotics Society with all the facilities

Date: 04/10/2024

(Deep Rudra)

EXECUTIVE SUMMARY

The **Bluetooth Controlled Car** project utilizes an Arduino Uno microcontroller and an L298N motor driver to create a wireless remote-controlled vehicle. This innovative project aims to demonstrate the integration of wireless communication, embedded systems, and motor control technologies.

Key Features:

1. **Wireless Control**:
    - The car is controlled via a Bluetooth module (HC-05), allowing users to operate the vehicle from a smartphone or tablet using a dedicated app.
2. **Arduino Uno Microcontroller**:
    - Serves as the central processing unit, receiving commands from the Bluetooth module and controlling the car's movements.
3. **L298N Motor Driver**:
    - Facilitates the control of the DC motors by providing the necessary power and direction control, enabling smooth movement of the car.
4. **Bidirectional Movement**:
    - The vehicle can move forward, backward, left, and right, offering flexibility in navigation.
5. **User-Friendly Interface**:
    - The mobile application provides an intuitive interface for controlling the car, enhancing user experience.
6. **Compact Design**:
    - The project is designed to be lightweight and portable, making it easy to transport and deploy in various environments.
7. **Customizability**:
    - The system can be easily modified to incorporate additional features such as sensors for obstacle detection, enabling autonomous functionalities.
8. **Educational Value**:
    - This project serves as a practical learning tool for students, providing hands-on experience with programming, electronics, and robotics.

By successfully implementing the Bluetooth Controlled Car, this project showcases the potential of combining various technologies to create an interactive and engaging platform for learning and exploration in the field of robotics and automation.

CONTENTS

List of Figures:

1. INTRODUCTION:

In recent years, the convergence of technology and everyday life has opened up new avenues for innovation and creativity. One such exciting development is the emergence of Bluetooth-controlled devices, which provide users with enhanced convenience and remote operation capabilities. This report focuses on the project of a Bluetooth-controlled car, a captivating blend of engineering, programming, and design. By leveraging Bluetooth technology, this project not only demonstrates the principles of remote control and automation but also serves as a practical application for students and hobbyists interested in robotics and IoT (Internet of Things). Throughout this report, we will explore the design process, technical specifications, challenges faced, and the potential applications of Bluetooth-controlled vehicles, showcasing how such projects can inspire future advancements in smart technology.

2. Components Used

	Name of the Components	Quantities
	300rpm gear motors.	4
	Wheels (6.6cm × 2.6cm, Tyre Centre Hole: 5.3mm Long, 3.5mm wide)	4
	Chassis (22cm × 16.5cm)	1
	Arduino UNO	1
	HC-05 Bluetooth Module	1
	3.7v Rechargeable li-ion battery	3
	Dual H bridge Motor Drivers L298N	1
	Male/Female Jumper wires	19
	Breadboard	400 points
	Arduino IDE	
	Application for control via Bluetooth	
Extra helping components like double tape, Scissor, Screw Driver, wire stripper, Insulation tape		

2.1 300rpm Gear Motors :

The 300 RPM gear motor is an dc motor coupled with a gearbox designed to reduce speed while increasing torque, making it ideal for applications that require controlled movement and power.

- **Speed and Torque**: Operating at 300 RPM, this motor strikes a balance between speed and torque, allowing for effective performance in driving the wheels of the Bluetooth-controlled car.
- **Efficiency**: The gearbox enhances efficiency by optimizing power delivery, ensuring that the motor can handle various loads encountered during operation.

Fig. – 01: Gear Motor

- **Control Mechanism**: The motor's characteristics enable precise control over acceleration and direction, essential for navigating different terrains with a Bluetooth interface.
- **Durability**: Designed for resilience, the 300 RPM gear motor can withstand the demands of continuous use in a project like a Bluetooth-controlled car.

2.2 Wheels:

The wheels are designed to provide optimal traction and stability for the Bluetooth-controlled car.

- **Material**: Made from durable plastic, they ensure longevity and effective grip on various surfaces.
- **Size**: Each wheel measures **6.6 cm in diameter** and 2.6 cm in width, with a tire centre hole of 5.3 mm long and 3.5 mm wide.
- **Attachment to Gear Motor**: The wheels are securely attached to a 300 RPM gear motor, allowing for synchronized rotation and controlled movement.

Fig. – 02: Wheel

2.3 Chassis:

The chassis of the Bluetooth-controlled car is made from lightweight cardboard, offering an economical solution for the project. This material provides structural support for the motors, wheels, and electronics while ensuring optimal balance for smooth operation. Its design allows for easy assembly and modifications, making it ideal for quick adjustments and customization to meet project requirements.

2.4 Arduino UNO:

The Arduino Uno acts as the main microcontroller, coordinating the operations of the Bluetooth module, motors, and other electronic components in the car.

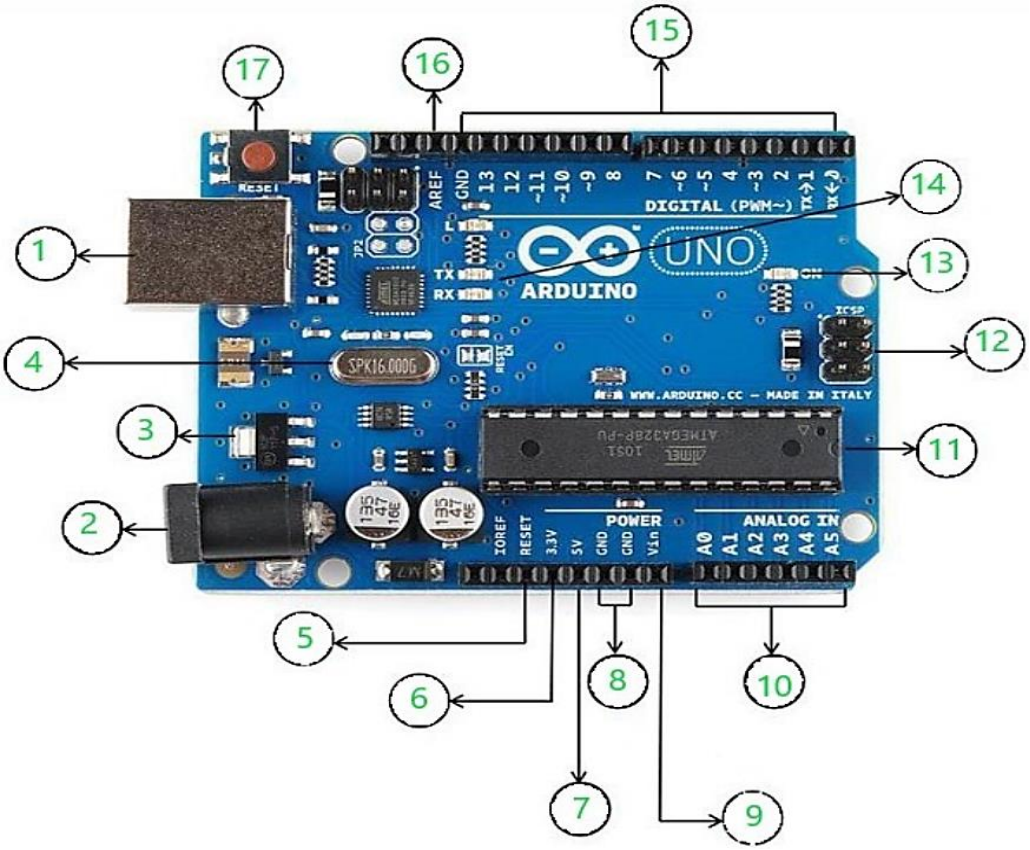

Fig. – 03: Arduino UNO

1. **USB:** can be used for both power and communication with the IDE
2. **Barrel Jack:** used for power supply
3. **Voltage Regulator:** regulates and stabilizes the input and output voltages
4. **Crystal Oscillator:** keeps track of time and regulates processor frequency
5. **Reset Pin:** can be used to reset the Arduino Uno
6. **3.3V pin:** can be used as a 3.3V output
7. **5V pin:** can be used as a 5V output
8. **GND pin:** can be used to ground the circuit
9. **Vin pin:** can be used to supply power to the board
10. **Analog pins(A0-A5):** can be used to read analog signals to the board
11. **Microcontroller (ATMega328):** the processing and logical unit of the board
12. **ICSP pin:** a programming header on the board also called SPI
13. **Power indicator LED:** indicates the power status of the board
14. **RX and TX LEDs:** receive (RX) and transmit (TX) LEDs, blink when sending or receiving serial data respectively

2.5 HC-05 Bluetooth Module:

Wireless communication is swiftly replacing the wired connection when it comes to electronics and communication. Designed to replace cable connections HC-05 uses serial communication to communicate with the electronics. Usually, it is used to connect small devices like mobile phones using a short-range wireless connection to exchange files. It uses the 2.45GHz frequency band. The transfer rate of the data can vary up to 1Mbps and is in range of 10 meters.

The HC-05 module can be operated within 4-6V of power supply. It supports baud rate of 9600, 19200, 38400, 57600, etc. Most importantly it can be operated in Master-Slave mode which means it will neither send or receive data from external sources.

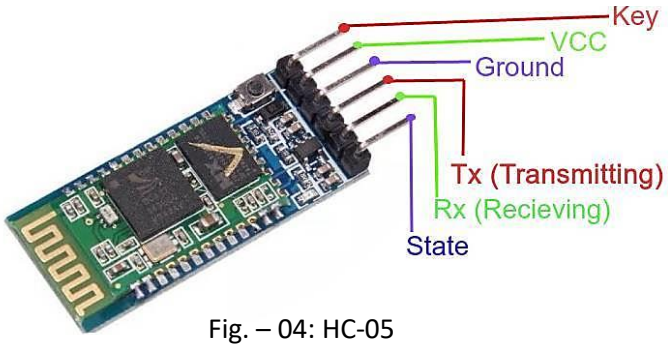

Fig. – 04: HC-05

- Enable - This pin is used to set the Data Mode or and AT command mode (set high).
- VCC - This is connected to +5V power supply.
- Ground - Connected to ground of powering system.
- Tx (Transmitter) - This pin transmits the received data Serially.
- Rx (Receiver) - Used for broadcasting data serially over Bluetooth.
- State -Used to check if the Bluetooth is working properly

2.6 Rechargeable LI-ion battery

The Li-ion battery serves as the primary power source for the Bluetooth-controlled car, offering high energy density in a lightweight design. Rechargeable and cost-effective, it maintains stable voltage output throughout its discharge cycle, ensuring consistent performance for the motors and electronics. With a low self-discharge rate, the battery retains charge over time, making it ideal for infrequent use. Additionally, built-in safety features protect against overcharging and overheating, enhancing operational safety. Here I using 3 batteries in series as a power supply.

Fig. – 05: LI-ion battery

2.7 L298N Dual H bridge Motor Drivers:

The L298N is a dual H-Bridge motor driver IC used for controlling two DC motors simultaneously. It is ideal for applications like Bluetooth-controlled cars.

- Dual H-Bridge: Controls two motors, allowing for forward, reverse, and braking.
- Voltage Range: Operates between 5V and 35V.
- Current Handling: Supplies up to 2A per channel.
- Thermal Protection: Features built-in thermal shutdown to prevent overheating.
- Control Signals: Uses four input pins for direction control and two enable pins for speed control via PWM.

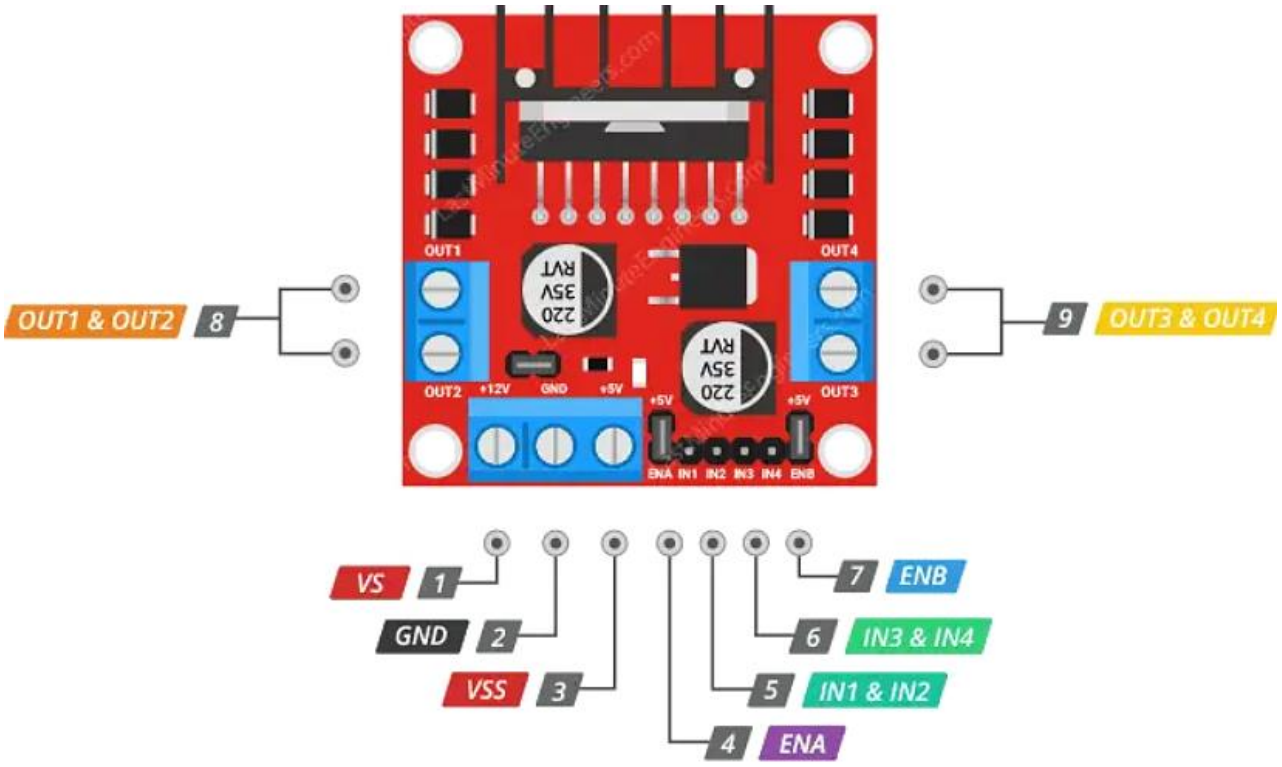

Fig. – 06: L298N

Operations:

Input1	Input2	Spinning Direction
Low(0)	Low(0)	Motor OFF
High(1)	Low(0)	Forward
Low(0)	High(1)	Backward
High(1)	High(1)	Motor OFF

Application: In a Bluetooth-controlled car, the L298N drives the motors based on commands from a Bluetooth module connected to a microcontroller, ensuring responsive movement.

2.8 Male/Female Jumper wires:

Male/female jumper wires are flexible connectors used in electronics to link components like sensors and microcontrollers, facilitating easy prototyping and assembly.

- **Types**: Available in male-to-male, female-to-female, and male-to-female configurations.
- **Length Variety**: Offered in various lengths to suit different project needs.
- **Easy Connectivity**: Male connectors plug into female headers, while female connectors accommodate male pins, facilitating quick connections and disconnections.
- **Flexible and Durable**: Typically made from insulated copper wire, ensuring flexibility and longevity.

Fig. – 07: Jumper Wires

2.9 Breadboard:

A breadboard is a reusable platform used for prototyping electronic circuits without soldering. It consists of a grid of interconnected holes, allowing for easy insertion of various components like resistors, capacitors, and microcontrollers. This modular design enables quick assembly and modifications, making it ideal for testing and experimenting with circuit designs. Breadboards simplify the development process by allowing users to visualize and tweak their circuits before finalizing a permanent setup.

- **Construction**: Composed of a grid of interconnected holes, typically with horizontal and vertical power rails.
- **Easy Connections**: Components and jumper wires can be easily inserted and removed, enabling quick circuit changes.
- **Size Variety**: Available in different sizes to accommodate various project scales.

Fig. – 08: Breadboard

2.10 Arduino IDE:

The Arduino Integrated Development Environment (IDE) is a software platform for writing, compiling, and uploading code to Arduino boards. It is designed to simplify the programming process.

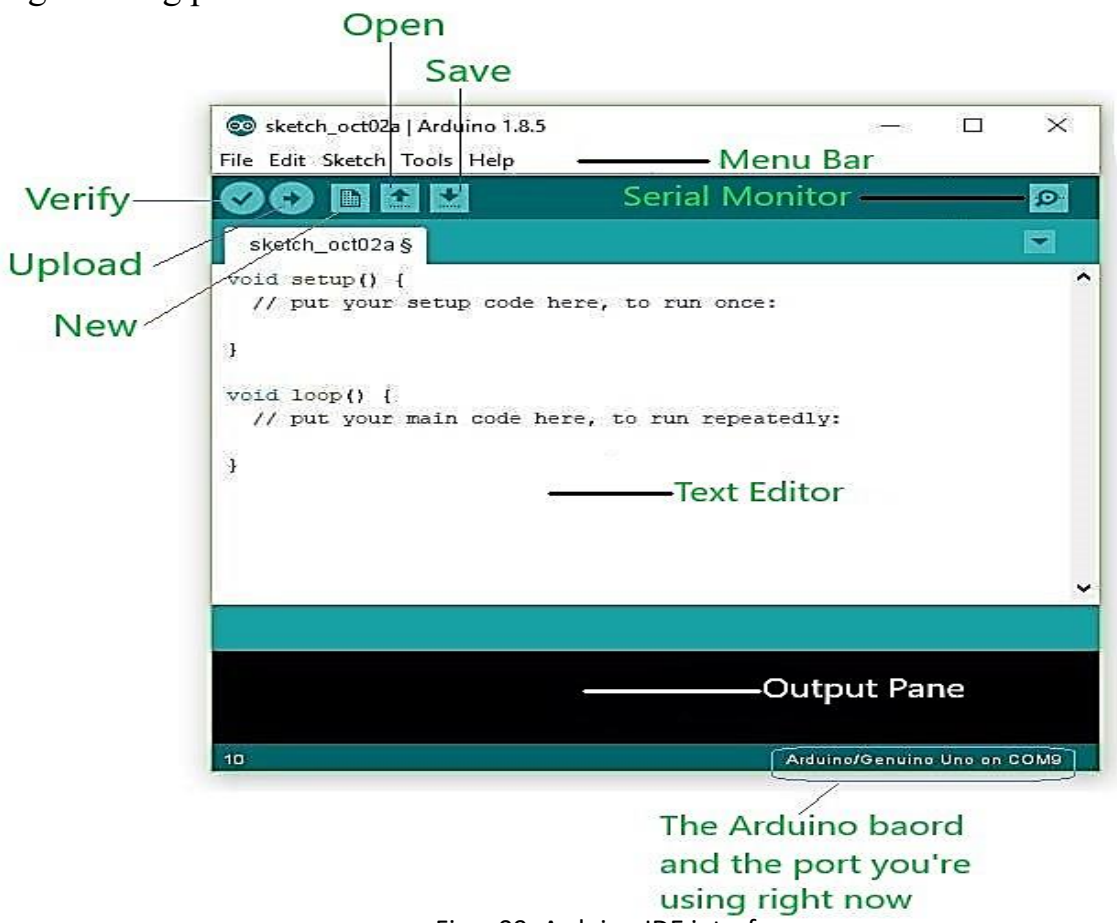

Fig. –09: Arduino IDE interface

- **User-Friendly Interface**: Intuitive layout that makes coding accessible for all skill levels.
- **Programming Language**: Supports C/C++ with easy-to-use syntax and built-in functions.
- **Libraries**: Includes numerous libraries for simplifying tasks like controlling motors, sensors, and communication protocols.
- **Serial Monitor**: Allows real-time monitoring and debugging of data sent between the Arduino and the computer.

2.11 Application for control via Bluetooth:

Here I am using an application which has a good user interface and easy to connect to the Bluetooth module HC-05 and also easy to operate through it.

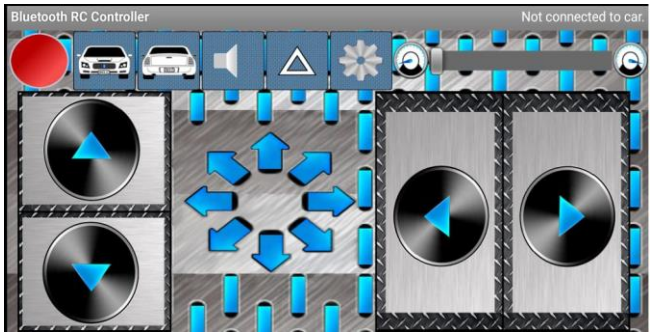

Fig. – 10: App interface

3. Procedure:

Step 1: Assemble the Chassis

Mount the Motors: Attach the 300 RPM gear motors to the chassis using doble side tape.

Attach the Wheels: Fit the wheels onto the motor shafts, ensuring they are securely attached.

Step 2: Connect the Motor Driver

Wire the Motors: Connect the four motors to the L298N motor driver:

Motor A: Connect wires to OUT1 and OUT2

Motor B: Connect wires to OUT3 and OUT4

Power the Motor Driver: Connect the power supply from the 12V rechargeable battery to the L298N:

VCC to positive terminal of the battery

GND to negative terminal of the battery

Connect the Arduino: Use jumper wires to connect the motor driver to the Arduino UNO:

IN1 to pin 5

IN2 to pin 6

IN3 to pin 10

IN4 to pin 11

Step 3: Set Up the Bluetooth Module

Connect the HC-05 Module: Wire the HC-05 Bluetooth module to the Arduino:

VCC to 5V

GND to GND

TX (transmit) to RX (pin 0 on Arduino)

RX (receive) to TX (pin 1 on Arduino)

Step 4: Power the Arduino

Power the Arduino: Connect the Li-ion batteries to power the Arduino via V_{in} pin

Step 5: Code the Arduino

Install Arduino IDE: Open the Arduino IDE on computer.

Write the Code:

Arduino Code:

```
#define in1 5
#define in2 6
#define in3 10
#define in4 11
int command;
int Speed = 204;
int Speedsec;
int buttonState = 0;
int lastButtonState = 0;
int Turnradius = 0;
int brakeTime = 45;
int brkonoff = 1;

void setup() {
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  if (Serial.available() > 0) {
    command = Serial.read();
    Stop();
    switch (command) {
      case 'F':
        forward();
        break;
      case 'B':
        back();
        break;
      case 'L':
        left();
        break;
      case 'R':
        right();
        break;
      case 'G':
        forwardleft();
        break;
      case 'I':
        forwardright();
        break;
      case 'H':
        backleft();
        break;
      case 'J':
        backright();
        break;
      case '0':
        Speed = 100;
        break;
case '1':
      Speed = 140;
      break;
    case '2':
      Speed = 153;
      break;
    case '3':
      Speed = 165;
      break;
    case '4':
      Speed = 178;
      break;
    case '5':
      Speed = 191;
      break;
    case '6':
      Speed = 204;
      break;
    case '7':
      Speed = 216;
      break;
    case '8':
      Speed = 229;
      break;
    case '9':
      Speed = 242;
      break;
    case 'q':
      Speed = 255;
      break;
  }
  Speedsec = Turnradius;
  if (brkonoff == 1) {
   brakeOn();
  } else {
   brakeOff();
  }
 }
}
void forward() {
 analogWrite(in1, Speed);
 analogWrite(in3, Speed);
}
void back() {
 analogWrite(in2, Speed);
 analogWrite(in4, Speed);
}
void left() {
 analogWrite(in3, Speed);
 analogWrite(in2, Speed);
}
void right() {
 analogWrite(in4, Speed);
 analogWrite(in1, Speed);
}
void forwardleft() {
 analogWrite(in1, Speedsec);
 analogWrite(in3, Speed);
}
void forwardright() {
 analogWrite(in1, Speed);
 analogWrite(in3, Speedsec);
}
void backright() {
 analogWrite(in2, Speed);
 analogWrite(in4, Speedsec);
}
void backleft() {
 analogWrite(in2, Speedsec);
 analogWrite(in4, Speed);
}

void Stop() {
 analogWrite(in1, 0);
 analogWrite(in2, 0);
 analogWrite(in3, 0);
 analogWrite(in4, 0);
}

void brakeOn() {

 buttonState = command;

 if (buttonState != lastButtonState) {

  if (buttonState == 'S') {
   if (lastButtonState != buttonState) {
    digitalWrite(in1, HIGH);
    digitalWrite(in2, HIGH);
    digitalWrite(in3, HIGH);
    digitalWrite(in4, HIGH);
    delay(brakeTime);
    Stop();
   }
  }
  lastButtonState = buttonState;
 }
}
void brakeOff() {

}
```

Connect your Arduino to the computer and upload the code.

Step 6: Set Up Bluetooth Control

Pair the HC-05 Module: On the smartphone, enable Bluetooth and pair with the HC-05

Step 7: Test the Car

Power On: Turn on the power for both the Arduino and the motor driver.

Control the Car: Use the Bluetooth app to send commands and test the movement of your car.

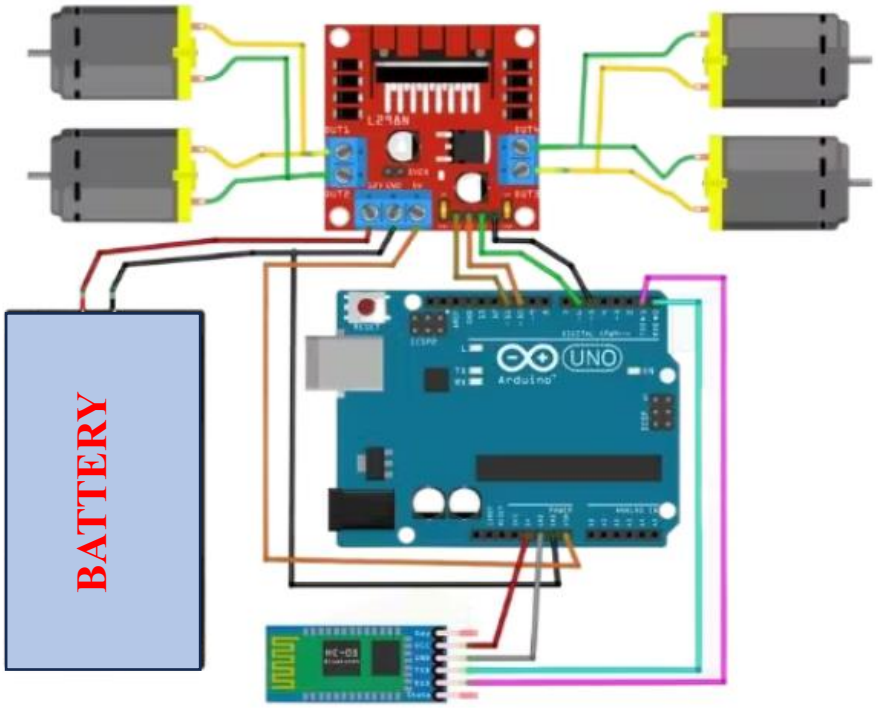

Fig.- 11: Circuit diagram

4. Conclusion:

The Bluetooth-controlled car project successfully demonstrates the integration of various electronic components, including motors, a motor driver, a microcontroller, and a Bluetooth module, to create a functional and interactive vehicle. Through the use of the Arduino IDE and a user-friendly mobile application, users can remotely control the car easily. This project highlights the potential of wireless communication in robotics and offers valuable insights into the design and development process of autonomous systems.

5. Recommendation:

1. Enhanced Features: Future iterations of the project could incorporate additional features such as obstacle detection using ultrasonic sensors or camera integration for video streaming, enhancing the car's functionality and user experience.

2. Improved Power Management: Investigating more efficient power management solutions, such as battery monitoring and optimization, could extend the operational time of the car.

3. Robustness and Durability: Testing and refining the car's physical design to withstand various terrains could improve its durability and performance in real-world applications.

4. Expanded Control Options: Exploring alternative control methods, such as voice commands or gesture-based controls, could further enhance user interaction and accessibility.

5. Documentation and Tutorials: Creating comprehensive documentation and tutorials for building and programming the Bluetooth-controlled car would be beneficial for educational purposes, allowing others to replicate and learn from the project.

6. Future Scopes:

The Bluetooth-controlled car project has many exciting possibilities for the future:

1. **Self-Driving Features**: We can add smart technology that helps the car drive on its own. It could use sensors to avoid obstacles and find its way around.
2. **Better Control Options**: We can create new ways to control the car, like using hand gestures or voice commands, making it easier and more fun to use.
3. **Connecting to the Internet**: Making the car connect to the internet would allow us to control it from anywhere. We could even track its location with GPS.
4. **Building Other Types of Robots**: We could use the same technology to create different kinds of robots, like drones or larger vehicles, for tasks like delivery or monitoring.

REFERENCES :

- https://lastminuteengineers.com/l298n-dc-stepper-driver-arduino-tutorial/
- https://www.geeksforgeeks.org/what-is-arduino/
- https://www.javatpoint.com/arduino
- https://www.geeksforgeeks.org/all-about-hc-05-bluetooth-module-connection-with-android/