GUJARAT TECHNOLOGICAL UNIVERSITY

Vishwakarma Government Engineering College

Chandkheda, Ahmedabad

Project Report On
Bluetooth controlled car

Under subject of
Physics
Semester – 2nd (EC)

Submitted by:
Team Number: 4

Sr. No.	Student Name	Enrollment Number
1	Dhruvil Shah	220170111113
2	Preet shah	220170111118
3	Manjeet singh	220170111105
4	Mohit topiya	220170111139
5	Pratham Kharaliya	220170111101

Prof. Jigar Chaudhari
(Faculty Guide/Mentor)

Acknowledgement

We would like to express our heartfelt gratitude to our project guide, Prof. Jigar Chaudhari, for his unwavering support, invaluable insights, and constructive feedback throughout the course of this project. His guidance has been instrumental in shaping the quality and direction of our work.

We extend our sincere thanks to the faculty of Physics, whose knowledge and expertise provided us with a strong foundation for completing our domain work. Their guidance in creating various models and project documents was essential to our success.

We are deeply appreciative of the encouragement and assistance from our friends and colleagues, whose inspirational support played a significant role in helping us navigate the challenges we faced during this project.

Thank you all for contributing to the successful completion of our project.

Abstract

The Bluetooth-controlled car project involves designing and implementing a wirelessly operated vehicle using Bluetooth communication technology and a microcontroller platform like Arduino. The main objective is to enable remote control of the car's movements—forward, backward, left, and right—through a mobile application or a Bluetooth-enabled device. Key components include a microcontroller (e.g., Arduino), a Bluetooth module (such as HC-05 or HC-06), motor drivers to control the car's DC motors, and a power supply. The project also integrates simple control logic within the microcontroller to interpret commands received from the mobile device and execute corresponding movement operations. The project demonstrates the practical application of wireless communication in robotics, enabling users to interact with the vehicle in real-time. This project provides hands-on experience in electronics, programming, and communication protocols, making it a valuable learning tool for understanding wireless control systems and mobile-based automation.

Table of content

1.Introduction

1.1 Overview of The Project

The Bluetooth-controlled car project is a practical demonstration of wireless communication and automation using mobile devices. The primary goal is to design and implement a remotely operated vehicle that can be controlled via Bluetooth, using a smartphone application or another Bluetooth-enabled device. The car's movements—forward, backward, left, and right—are governed by the commands sent over Bluetooth to a microcontroller, such as Arduino, which then interprets the signals and drives the motors accordingly. This project highlights the integration of electronics, programming, and communication protocols to build a fully functional system that operates seamlessly without physical interaction.

1.2 Objectives

The main objectives of the project are:

- To design a Bluetooth-controlled car capable of responding to movement commands in real-time.
- To use a Bluetooth module to establish communication between a mobile device and the car’s microcontroller.
- To program the microcontroller to interpret the received commands and control the motors accordingly.
- To implement an easy-to-use mobile application interface for controlling the car.

1.3 Applications of Bluetooth-controlled Vehicles

Bluetooth-controlled vehicles have various applications, including remote inspection in hazardous areas, educational tools for learning automation and robotics, and entertainment purposes such as DIY robotic kits. The project also serves as a stepping stone for more complex projects involving autonomous vehicles and wireless-controlled robotic systems.

2.Literature Review

2.1 Wireless Communication in Robotics

Wireless communication plays a vital role in modern robotics, allowing systems to be remotely controlled without the need for physical wiring. It increases flexibility and extends the range of control. Various technologies, such as RF, Bluetooth, and Wi-Fi, are commonly used for short-range and long-range control of robotic systems.

2.2 Bluetooth Technology in Automation

Bluetooth is a widely used communication protocol for short-range wireless communication. It is often chosen for projects due to its ease of use, low power consumption, and wide availability. It can be implemented in automation systems like smart home devices, medical equipment, and, as in this project, robotics, where devices are wirelessly controlled.

2.3 Related Projects and Work

Numerous Bluetooth-controlled robotic vehicles have been designed in academic and hobbyist environments. Similar projects have used platforms like Arduino and Raspberry Pi to control movements using Bluetooth modules, often focusing on ease of interaction, cost-effectiveness, and educational value.

3.System Design

3.1 Block Diagram of the System

The system consists of the following blocks:

- *Bluetooth Module:* Receives commands from the mobile app.
- *Microcontroller (Arduino):* Interprets the Bluetooth commands and controls the motor driver.
- *Motor Driver (L298N):* Drives the motors based on signals from the microcontroller.
- *DC Motors:* Control the car's movement.

3.2 Hardware Components

3.2.1 *Microcontroller (Arduino):* Arduino serves as the brain of the system. It processes incoming Bluetooth commands and outputs signals to control the motor driver.

3.2.2 *Bluetooth Module (HC-05/HC-06):* This module establishes wireless communication between the car and a mobile device. It operates over a standard 2.4 GHZ frequency and can communicate with most modern smartphones.

3.2.3 *Motor Driver (L298N):* The motor driver provides the necessary voltage and current to drive the motors. It is capable of controlling the direction and speed of two DC motors.

3.2.4 *DC Motors:* Two DC motors are used to move the car in different directions by controlling their speed and rotational direction.

3.2.5 *Power Supply:* The power supply consists of a battery pack that provides the required voltage to the motor and the microcontroller.

3.3 Circuit Design

The circuit connects the Bluetooth module to the Arduino through its serial interface. The motor driver is connected to the Arduino’s output pins, while the motors are connected to the motor driver. Power is

supplied to both the motor driver and Arduino.

4.Software Implementation

4.1 Arduino Programming

The Arduino is programmed using the Arduino IDE. The code listens for Bluetooth commands (like 'F' for forward, 'B' for backward, 'L' for left, 'R' for right) and drives the motors accordingly by sending signals to the motor driver.

4.2 Bluetooth Communication Protocol

Bluetooth modules like HC-05 communicate using standard serial communication. The module is paired with the mobile device, and commands are transmitted via the mobile app. These commands are then processed by the microcontroller.

4.3 Control Logic for Car Movements

The control logic is simple. Based on the command received ('F', 'B', 'L', 'R'), the microcontroller sends signals to the motor driver to rotate the motors in the desired direction, enabling movement of the car.

4.4 Mobile Application Interface

A mobile application, typically built using Bluetooth controller apps available on platforms like Android, sends movement commands. The app can be customized with buttons to control the car’s movement direction.

5.System Operation

5.1 Pairing Bluetooth Device with Mobile Application

The Bluetooth module on the car is paired with the smartphone or Bluetooth-enabled device. Once paired, the car is ready to receive movement commands.

5.2 Sending Commands to Control the Car

Using the mobile application, users can send commands like 'F' for forward, 'B' for backward, 'L' for left, and 'R' for right. These commands are transmitted to the Bluetooth module, which sends them to the Arduino for execution.

5.3 Testing and Troubleshooting

The system is tested for response time, accuracy of movements, and range of Bluetooth control. Troubleshooting involves checking connections, adjusting code for better performance, and ensuring proper motor operation.

6. Results and Discussion

6.1 Functionality of the Bluetooth-controlled Car

The Bluetooth-controlled car was successfully able to respond to movement commands from the mobile application. It operated within the expected Bluetooth range and accurately followed user inputs.

6.2 Performance Analysis

The performance was measured in terms of response time, battery efficiency, and range of Bluetooth communication. The system showed minimal delay in movement commands and had a sufficient control range for indoor use.

6.3 Challenges Faced and Solutions

Some challenges included Bluetooth connection dropouts and motor power fluctuations, which were addressed by improving wiring connections and optimizing power supply.

7.Conclusion

7.1 Summary of Findings

This project demonstrates the practical use of Bluetooth technology to control a car wirelessly. It provides valuable insights into wireless communication, microcontroller programming, and motor control.

7.2 Future Enhancements

Future work could include adding features like obstacle avoidance using sensors, improving range by using Wi-Fi instead of Bluetooth, and building an autonomous navigation system.

8.Appendix

8.1 Code Listings

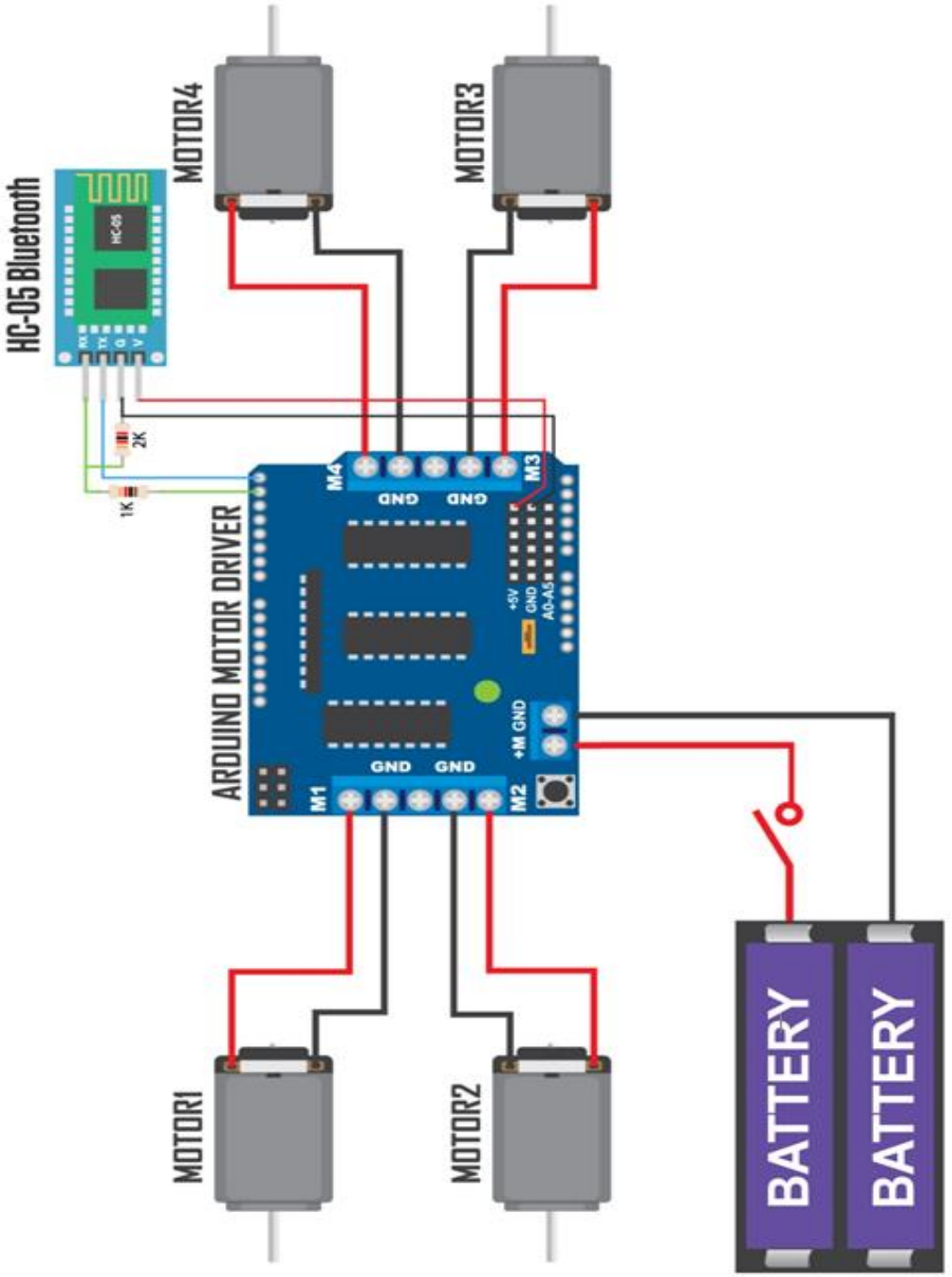
Complete code for the Bluetooth controlled Car.

```
char t;
void setup() {
pinMode(13,OUTPUT);
pinMode(12,OUTPUT);
pinMode(11,OUTPUT);
pinMode(10,OUTPUT);
pinMode(9,OUTPUT);
Serial.begin(9600);
}
void loop() {
if(Serial.available()){
  t = Serial.read();
  Serial.println(t);
}
if(t == 'F')
{
  digitalWrite(13,HIGH);
  digitalWrite(11,HIGH);
}
else if(t == 'B')
{
  digitalWrite(12,HIGH);
  digitalWrite(10,HIGH);
}
else if(t == 'L')
{
  digitalWrite(11,HIGH);
}
else  if(t == 'R')
{
  digitalWrite(13,HIGH);
}
else if(t ==  'W')
{
  digitalWrite(9,HIGH);
}
else if(t == 'w')
{
  digitalWrite(9,LOW);
}
```

```
else if(t == 'S')
{
  digitalWrite(13,LOW);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);
}
delay(100);
}
```

8.2 Circuit Diagrams

Detailed schematics of the circuit connections.



8.3 Prototype

Physical circuit and prototype model of the project.