List of Figures

List of Abbreviation

RC: Radio/Remote Control

LCD: Liquid-crystal display

LI-ON Battery: Lithium-Ion Battery

USB: Universal Serial Bus

Chapter-01

Introduction

1.1 Background:

Wireless or Bluetooth control cars are developed to make our day-to-day life easy and simpler. Today for striving in this modern era it is necessary to use modern technology to simplify our day-to-day activities. Thousands of people die because of driving vehicles. If it can be remote-controlled, we will be able to save thousands of lives. The drone started a revolution in this regard. Bluetooth-controlled cars can be a revolutionary project for our upcoming future. It can be used for low-range mobile surveillance devices, (military Applications where no human intervention will be needed), Assistive devices (like wheelchairs), home automation, etc. It is aimed to bring a huge groundbreaking change in society by overcoming the hurdles we are facing today. Driverless cars are providing mobility to many people who cannot drive which is an added advantage to commuters who wish to use cars on daily basis. Bluetooth control cars are the gadget of the future.

1.2 Problem statement:

Bluetooth Control cars can be a helping hand for the cripples, or for the ones who want to drive but are unable due to many physical inabilities, where human life is at risk, we can use it from a very distance as it is fully wireless.

1.3 Project Overview:

The main processing unit consists of Arduino UNO to perform specified operations. Arduino UNO board is used as the main microprocessor of the device. It will run and will control the movement of the vehicle through the use of the motor driver who will control the wheels. The Programs are compiled using Arduino IDE and then the sketch is uploaded to the board using a pc and a USB cable. Speed variations can be achieved by making slight changes to the sketch. It

is based on a 4-wheel drive platform integrated with an ultra-sonic sensor to avoid obstacles and line tracer sensors. It also has an infra-red remote controller for the manual steering of the vehicle. The car is upgradable and can replace, upgrade and expand the input sensors and add other functionality by adding other modules.

Chapter-02

Literature Review

Wireless control is one of the most important basic needs for all people all over the world. Bluetooth is one of the most used wireless technologies. A Bluetooth control car is such a car that can be controlled wirelessly using a Bluetooth control system and Arduino. With the combination of Arduino, and Bluetooth we can control many other things, like home Lighting, air conditioners, and many more through our cell phones. The Arduino can also contribute at large to the Smart Home system. Nowadays it has made us easier to convert digital signals into physical movements with such microcontrollers.

Today people are using autonomation in every sector. There are many benefits, for example, it reduces the risk of injury, it is fast, it can work day long, it is reliable, etc. This Bluetooth control car is just an example of autonomation. This type of car is used in law enforcement and military engagements for some the reasons like Hazard exposure which is controlled from a location of relative safety. Such vehicles are used by many police department bomb squads to defuse or detonate explosives.

These vehicles are also used in space exploration, by using such vehicle NASA. ESA and ROSCOSMOS have explored and collected many data from space, the moon, and mars. In recent days giant companies are using remote control vehicles to deliver their products. And Many of the giant factories have their transportation which is remotely controlled.

Here we have connected this 4-wheeler with Arduino and Bluetooth Module/ There is an Android application that is already installed in the remote device. We send our instruction to the vehicle with that application, that application is connected with the module in the vehicle. Bluetooth module transfer that instruction as a signal to the Arduino, and Arduino works with that signal. In the Arduino, we have installed the code which can work with the signals and convert those instructions to the pre-defined signals. Signals were transferred to the motors, and the car started running.

This car does not have advanced features but we can attach any kind of features like line detecting, or obstacle detecting, even though we can attach the camera to the vehicle and watch it with the remote. This one is just a basic prototype of a remote-control car, we can add a lot of advanced features and get an armed/especially capable RC Car.

This project aimed to design an automated vehicle prototype built with Arduino and controlled with software developed on Android that can perform manual or automatic paths. Until now research and analyzing the simulation of experiments shown, it is believed that it is feasible to use the prototype designed for cognitive development, for future users can learn to insert custom paths that can process logic issues and more complex mathematics allowing the prototype to perform the desired movements. Analyzing the financial costs of design, it is believed that it is feasible to construct this type of prototype because it presents a low cost of the components used, particularly if they are chosen in a largescale production. It is worth noting that both the Arduino programming language and the language for Arduino in development are free, not burdening additional costs for the development of the project, pointing out that this applies also to the tools used for development.

Chapter-03

Methodology

3.1 Tools:

- **Chassis (Including motors and wheels):**

The main body of the car consists of the chassis. Alongside there will be four motors for the four wheels which will lead down the car.

Figure-01: Chassis (Including motors and wheels)

- **Arduino Uno:**

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino. cc. The board is equipped with sets of digital and analog input/output pins that may be interfaced with various expansion boards and other circuits.

Figure-02: Arduino Uno

- **L293D Motor Drive H-Shield:**

The Motor Driver is a module for motors that allows you to control the working speed and direction of two motors simultaneously. This Motor Driver is designed and developed based on L293D IC. L293D is a 16 Pin Motor Driver IC. This is designed to provide bidirectional drive currents at voltages from 5 V to 36 V. Rotation of a motor depends on the enabled pins.

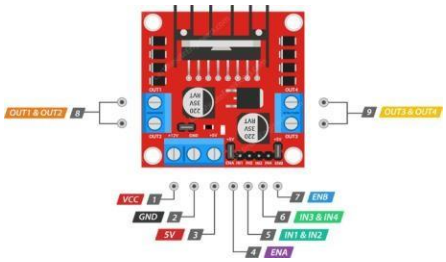


Figure-03: L293D Motor Drive H-Shield

- **Jumper Wires:**

  A jump wire (also known as jumper, jumper wire, DuPont wire) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

Figure-04: Jumper Wires

- **Battery:**

  A battery is an energy source consisting of one or more electrochemical cells and terminals on both ends called an anode (-) and a cathode (+). Here for the project, a 9V battery has been used. (In figure 5)

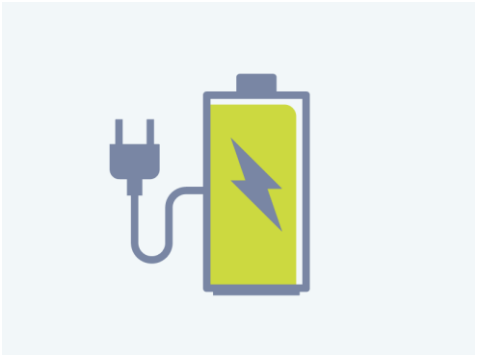


Figure-05: Battery

- **Bluetooth Module:**

  In figure 7 HC-05 6 Pin Wireless Serial Bluetooth Module is a Bluetooth module for use with any microcontroller. It uses the UART protocol to make it easy to send and receive data wirelessly. The HC-06 module is a slave-only device. This means that it can connect to most phones and computers using Bluetooth technology. [8]

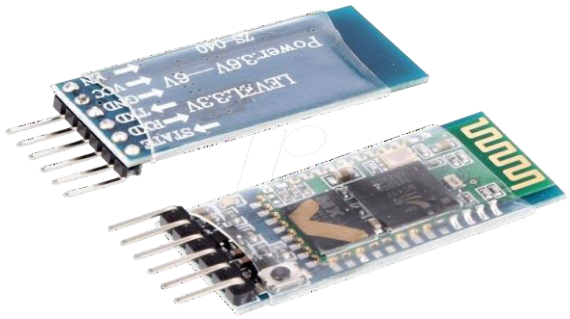


Figure-07: Bluetooth Module

3.2 Pin Diagram & Connection:

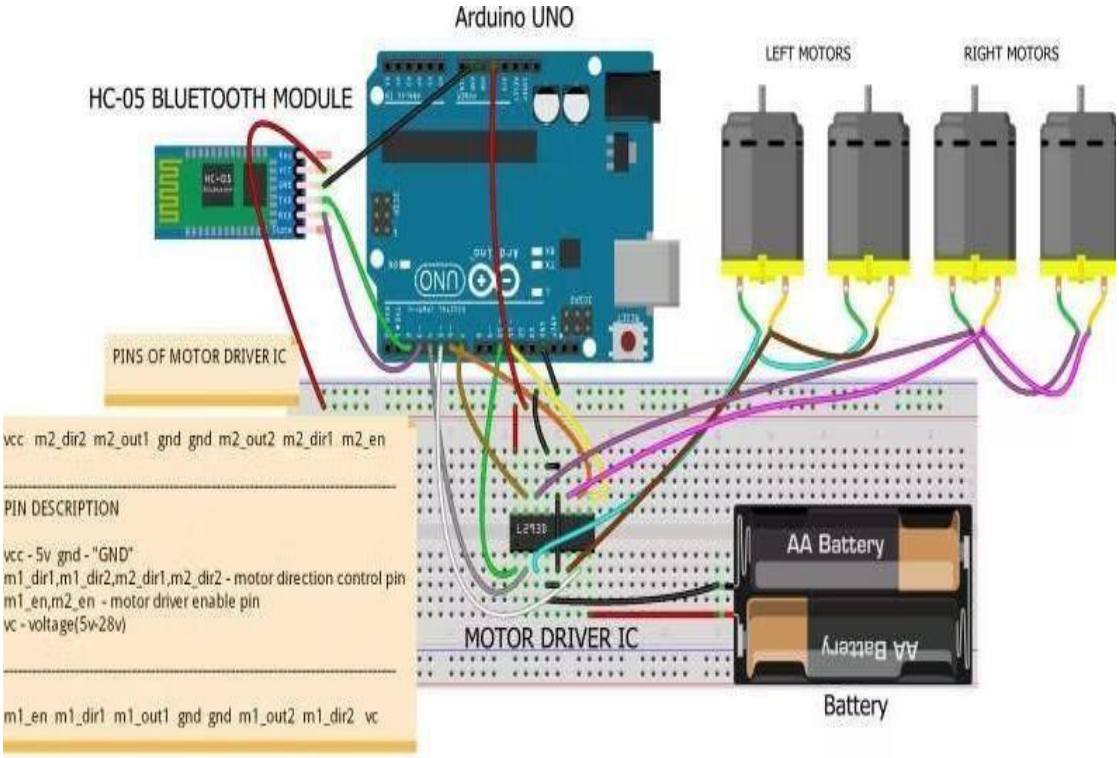


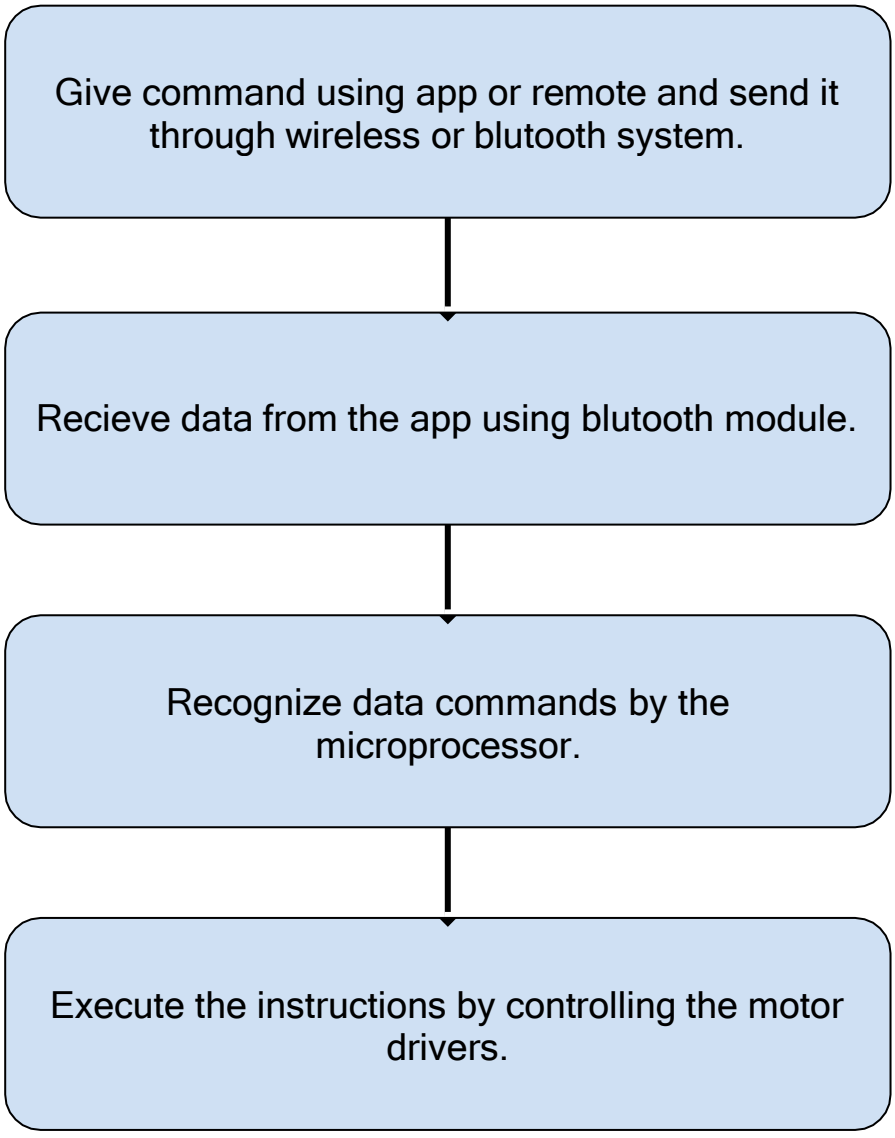
Figure-09: Pin Diagram & Connection

3.3 Source Code:

```
#define IN1 2
#define IN2 3
#define IN3 4
#define IN4 5

#define ENA 9
#define ENB 10

#define Speed 180

void setup() {
  Serial.begin(9600);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
}

void loop() {
  if (Serial.available() > 0) {
    char value = Serial.read();
    Serial.println(value);

    if (value == 'U') {
      Forward();
    } else if (value == 'D') {
      Backward();
    } else if (value == 'S') {
      Stop();
    } else if (value == 'L') {
      Left();
    } else if (value == 'R') {
      Right();
    }
  }
}

void Forward() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
void Backward() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
```

```
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
void Stop() {
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW);
}
void Left() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
void Right() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
```

3.4 Flow chart of Implementation:



Chapter-04

Result and Discussion

The prototype is created using 4 DC motors driven by an H-bridge with dual output, connecting the two left wheels and the two right ones to its outputs. The car is remotely controlled either by Bluetooth an app or by infrared. Some extra functionalities can be added to the software.

Better line tracking mode: When the robot finds an object in front placed on the line, it will try to go around it until it finds the line again, continuing afterward. This function can be added using an ultrasonic sensor.

Custom mode: The ability to program the prototype from the app has not been implemented, as it is relatively easy to use the custom mode by modifying the code.

Chapter-05

Conclusion

This project consists of a basic prototype of a Bluetooth control car above stated. The prototype car can recognize commands from users and can turn the car left, right, and stop with great accuracy. It can be further improved by using different sensors like ultrasonic or infrared and with various levels of coding.

Our Bluetooth control car has a range of 10-20 meter with the mobile Bluetooth controlling system. The range mostly depends on the receivers transmission level.

In the future, we could use rechargeable batteries like Ni-Cd Battery or Li-ion batteries that could avoid the present disadvantage. Also, we could make use of this RC Motor Car as a surveillance system or rovers by adding a few more sensors and updating the code. This would make them into robots. These robots could self-monitor under any human supervision, thereby reducing manpower. These are just the alternatives, on which this project could be improvised and updated.

IoT or the internet of things is a very important part of both computers and our daily lives. The above model describes how the Arduino programs the car motor module and by IoT, we rotate the wheels and give direction to the car. IoT allows us to work with different platforms and it helps us to create various interesting modules to work on.

With the ever-increasing problems, our knowledge has to expand to adapt better to the changes all around us. In the same way, it is hoped that this activity is a small step that would lead us to further enhancements and goals.