Bluetooth Car Controlled Using Arduino

Dr. Deepak Sonker[1], Dr. Vishal Khatri[2], Dr. Ranjeeta Kaur[1], Ms Ambooj Yadav[1]

Tecnia Institute of Advanced Studies, Delhi, India[1]

Bhagwan Parshuram Institute of Technology, Delhi, India[2]

Guru Gobind Singh Indraprastha University, Delhi, India

Abstract: *It is through efficient electronic programming that a computer can control a robot, hence a robot can be thought of as an Electromechanical machine. Some of the essential characteristics that a robot must have are - sensing, movement, energy, intelligence. It performs a task using control systems, various power supplies and software all working together. We developed an Android application which uses remote buttons to guide an RC car's motion. Hence, the mobile device harboring the Android application acts as the car's remote control. Bluetooth is the basis of communication between the controller and Android, using the USART protocol.*

Keywords: Android phone, Bluetooth module, Robot, Arduino

I. INTRODUCTION

Robots are always a fancy topic for students, hobbyists and DIYers. If you are beginner, then building a robot (like a car or an arm) is probably one of the important projects to do after learning about the basics. If you remember the earlier tutorial, I have discussed about HC-05 Bluetooth Module and how to interface one with Arduino. Also, I have provided a simple Bluetooth Controller App, which can be installed on your Android Phone and start transmitting the data. As a continuation to that project, I will be implementing Bluetooth Controlled Robot using Arduino and a few other components and build a simple robotic car that can be controlled using an Android Phone (through an App) over Bluetooth Communication.

The robotic car can be controlled wirelessly via a Smartphone. The smartphone has an Android app through which the user can send commands directly to Robot. The robot can move forward, backward, left, and right and can also be stopped.

The Arduino's Bluetooth-controlled robot car is interfaced with a Bluetooth module HC-05 or HC-06. We can give specific voice commands to the robot through an Android app installed on the phone. At the receiving side, a Bluetooth transceiver module receives the commands and forwards them to the Arduino, and thus the robotic car is controlled.

II. SYSTEM DESIGN AND IMPLEMENTATION

The following is the circuit diagram of Bluetooth Controlled Robot using Arduino, L298N and HC-05.
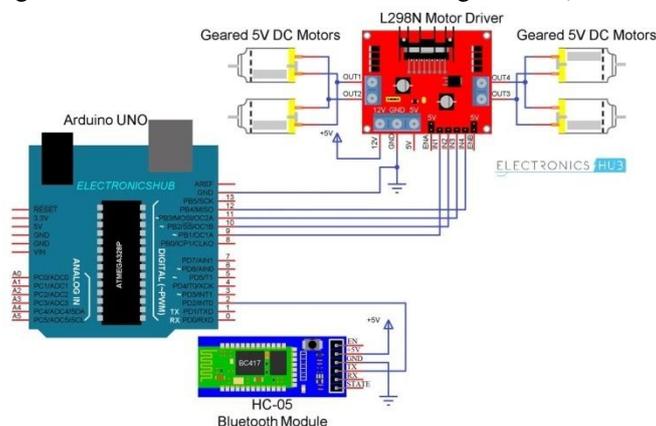

Figure: Circuit Diagram of Bluetooth Controlled Robot

Coming to the design of the circuit, first is the HC-05 Bluetooth Module. The +5V and GND pins of the Bluetooth Module are connected to +5V and GND of Arduino. Since I will be only transmitting data related to the Robot's movement from Android Phone to Bluetooth Module and do not intend to receive any data from Arduino, I will connect only the TX pin of the Bluetooth Module to RX Pin of Arduino. This RX pin of Arduino is based on Software Serial library (Pin 2 and Pin 3 are configured as RX and TX on Arduino). The RX pin of the Bluetooth is left open.

Now, the L298N Motor Driver Module. Digital I/O Pins 9 through 12 of Arduino are configured as Input pins of the Motor Driver and are connected to IN1 through IN4 of the L298N Motor Driver Module. Both the Enable Pins are connected to 5V through provided jumper.The robot chassis which I am using in this Bluetooth Controlled Robot Car project is supplied with 4 geared motors. Since L298N has slots for only two motors, I have joined the left side motors as one set and the right side motors as other set and connected both these sets to the output of L298N Module.

3.1 Code

The Arduino code for Bluetooth Controlled Robot project is given below.

```
#include<SoftwareSerial.h>
#define IN1 12
#define IN2 11
#define IN3 10
#define IN4 9
//#define EN1 6
//#define EN2 5
SoftwareSerial mySerial(2, 3); // RX, TX
String data;
int btVal;
void setup() {
        //Serial.begin(115200);
        pinMode(IN1, OUTPUT);
        pinMode(IN2, OUTPUT);
        pinMode(IN3, OUTPUT);
        pinMode(IN4, OUTPUT);
        //pinMode(EN1, OUTPUT);
        //pinMode(EN2, OUTPUT);
        digitalWrite(IN1, LOW);
        digitalWrite(IN2, LOW);
        digitalWrite(IN3, LOW);
        digitalWrite(IN4, LOW);
        //analogWrite(EN1,63);
        //analogWrite(EN2,63);
        mySerial.begin(9600);
}
void loop(){
        while (mySerial.available()) {
                data = mySerial.readStringUntil('\n');
                //Serial.print(str);
        }
        btVal = (data.toInt());
        //Serial.print("BlueTooth Value ");
        //Serial.println(btVal);
```

```
                switch (btVal) {
                        case 1:
                                //Serial.println("Forward");
                                forward();
                                break;
                        case 2:
                                //Serial.println("Reverse");
                                reverse();
                                break;
                        case 3:
                                //Serial.println("Left");
                                left();
                                break;
                        case 4:
                                //Serial.println("Right");
                                right();
                                break;
                        case 5:
                                //Serial.println("Stop");
                                stoprobot();
                                break;
                        }
                }
                if (mySerial.available() < 0){
                        //Serial.println("No Bluetooth Data ");
                }
        }
void forward(){
         digitalWrite(IN1, HIGH);
         digitalWrite(IN2, LOW);
         digitalWrite(IN3, HIGH);
         digitalWrite(IN4, LOW);
}
void reverse(){
         digitalWrite(IN1, LOW);
         digitalWrite(IN2, HIGH);
         digitalWrite(IN3, LOW);
         digitalWrite(IN4, HIGH);
}
void left(){
         digitalWrite(IN1, LOW);
         digitalWrite(IN2, LOW);
         digitalWrite(IN3, HIGH);
         digitalWrite(IN4, LOW);
}
void right(){
         digitalWrite(IN1, HIGH);
```

```
                digitalWrite(IN2, LOW);
                digitalWrite(IN3, LOW);
                digitalWrite(IN4, LOW);
}
void stoprobot(){
                digitalWrite(IN1, LOW);
                digitalWrite(IN2, LOW);
                digitalWrite(IN3, LOW);
                digitalWrite(IN4, LOW);
}
```

3.2 Android App

If you remember the HC-05 Bluetooth Module tutorial, I have used a simple app called Bluetooth Controller, which is installed on an Android Phone to communicate with the Bluetooth Module. In this project, I have used the same app with modifications in the data to be transmitted.

IV. WORKING

Assemble the robot, make the necessary connections and upload the code to Arduino. If you understood the HC-05 Bluetooth Module tutorial, then understanding the Bluetooth Controlled Robot project is very easy.

First, in the Android App, I have used 5 keys as Forward, Reverse, Left, Right and Stop. The corresponding data associated with each key is as follows:

- Forward – 1
- Reverse – 2
- Left – 3
- Right – 4
- Stop – 5

When a key is pressed, the corresponding data is transmitted to the Bluetooth Module from the Phone over Bluetooth Communication.

In the Arduino code, the Arduino UNO receives any of this data from the Bluetooth Module (as per the key pressed) and performs a simple switch case operation, where each case associated with appropriate instructions to the Motor Driver Input Pins.

For example, if 'Forward' key is pressed in the Android Phone, then '1' is transmitted. Arduino will then make IN1 and IN3 as HIGH and IN2 and IN4 as LOW to achieve a forward motion.
Similarly, other keys correspond to appropriate setting of IN1 – IN4 pins.

V. CONCLUSION AND FUTURE WORK

Robots and smart phone are a perfect match for us to realize smart living not only at work, but in our homes, with the aid of easily available and widely used technology, the Bluetooth.

As the mobile devices are becoming more advanced, using them for controlling Robots and other wireless devices is likely to be a huge trend. It can be concluded that this idea of smart living will let us control our surroundings remotely and wirelessly. 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.

REFERENCES

[1]. https://www.google.co.in/search?q=+bluetooth+car+control+image&tbm=isch&ved=2ahUKEwjGivCh7JnzA hU-nEsFHcmjA14Q2-cCegQIABAA&oq=+bluetooth+car+control+image&gs_lcp=CgNpbWcQAzoFCAAQ gAQ6BggAEAgQHlDnIVjjPmC6Q2gAcAB4AIAB6QGIAaANkgEFMC45LjKYAQCgAQGqAQtnd3Mtd2l6 LWltZ8ABAQ&sclient=img&ei=b_BOYYaXCb64rtoPyceO8AU&bih=578&biw=1366#imgrc=-zU1fr4TfJ MNyM

[2]. http://www.onlinejournal.in/IJIRV2I9/021.pdf

[3]. https://www.researchgate.net/publication/337759619_Arduino_based_Bluetooth_controlled_RC_Car

[4]. https://www.ijarcce.com/upload/2017/march-17/IJARCCE%2011.pdf

[5]. https://www.ijeat.org/wp-content/uploads/papers/v9i2/B3673129219.pdf

[6]. http://www.ijettjournal.org/2016/volume-33/number-8/IJETT-V33P274.pdf

[7]. https://www.slideshare.net/doaamarzook/bluetooth-controlled-android-car