SWITCHING THEORY PROECT

Bluetooth Controlled Car

Report by: Shivam Tiwari and Vipul Kumar Gond

● Introduction

This is an Arduino based Bluetooth controlled car It is controlled by a smart phone application. Bluetooth controlled car is controlled by using Android mobile phone instead of any other method like buttons, gesture etc. Here only needs to touch button in android phone to control the car in forward, backward, left and right directions. So here android phone is used as transmitting device and Bluetooth module placed in car is used as receiver. Android phone will transmit command using its in-built Bluetooth to car so that it can move in the required direction like moving forward, reverse, turning left, turning right and stop.

● Components Required

(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 to various expansion boards and other circuits.

HC-05 (Bluetooth Module)
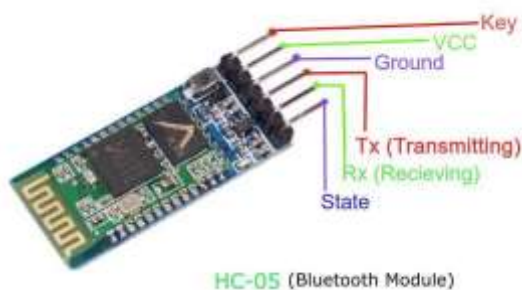
(HC-05 Bluetooth module)   HC-05 Bluetooth Module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Its communication is via serial communication which makes an easy way to interface with controller or PC
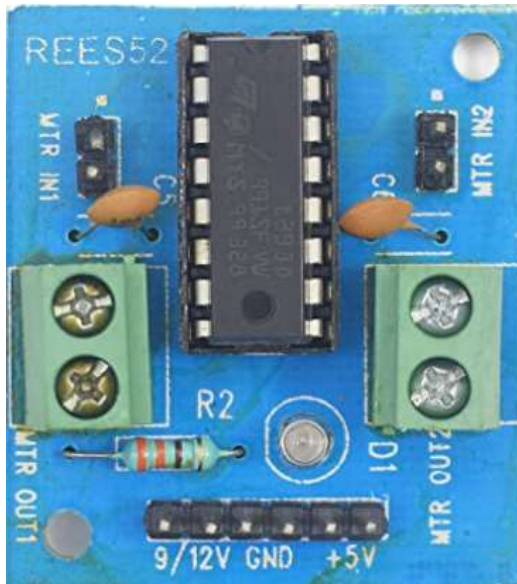
(L293D motor driver module) Motor Driver – L293D Driver Module is a medium power motor driver perfect for driving DC Motors and Stepper Motors. It uses the popular L293 motor driver IC. It can drive 4 DC motors on and off, or drive 2 DC motors with directional and speed control.

Jumper wires (generic)

(**HW Battery 6F22 9V and battery clip connector**)

(**Micromotors and Grippy wheels**)
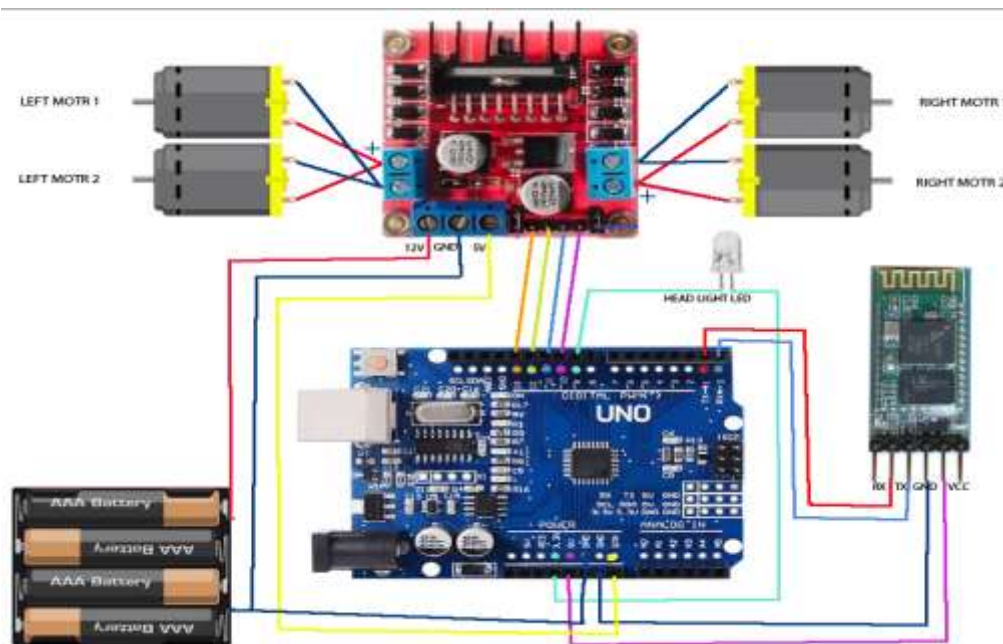

(**Switch**)   (**Wooden chassis**)

● Apps and online services

1. Arduino Bluetooth Controller- All in One. by MyValley Apps

(Google play store)

2. Arduino IDE.

● Circuit Diagram

●Code Analysis

```
char command;
void setup()
{
pinMode(8,OUTPUT);
pinMode(9,OUTPUT);
```

```
  pinMode(10,OUTPUT);

  pinMode(11,OUTPUT);

 Serial.begin(9600);  //Set the baud rate to your Bluetooth module.

 }
```

These lines of code have defined the working of digital pins.

```
void loop() {

 if(Serial.available() > 0) {

  command = Serial.read();

  // Stop(); //initialize with motors stoped

  //Change pin mode only if new command is different from previous.

  //Serial.println(command);

  switch(command) {

  case 'F':

   forward();

   break;

  case 'B':

    back();

   break;

  case 'L':

   left();

   break;

  case 'R':

   right();

   break;

  case 'X':

   Stop();

   break;

  case 'M':

   roundL();

   break;
```

```
  case 'N':
    roundR();
    break;
  }
 }
}
```

Switch function is used for different cases.

```
void forward()

{
  Stop();
  digitalWrite(8,HIGH);
  digitalWrite(10,HIGH);
 }
  void back()
{
  digitalWrite(8,0);
  digitalWrite(10,0);
  delay(100);
  digitalWrite(9,HIGH);
  digitalWrite(11,HIGH);
}
void left()
{
  Stop();
  delay(100);
  digitalWrite(8,HIGH);
  digitalWrite(11,HIGH);
  delay(300);
  Stop();
```
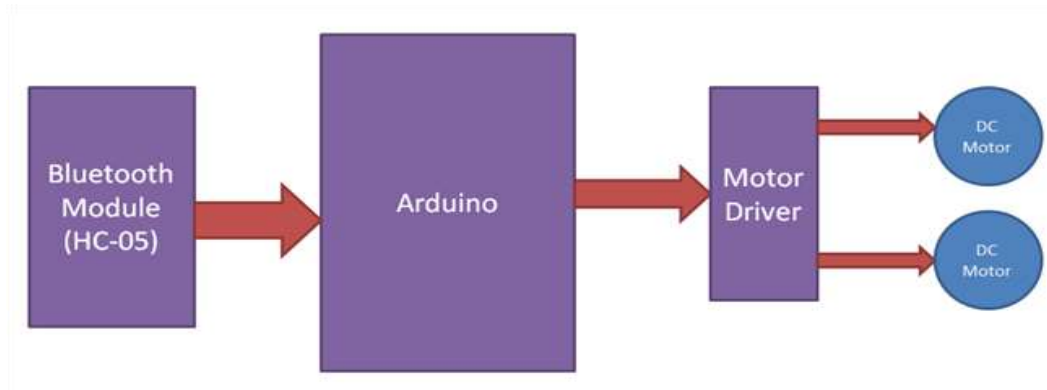
```
}
void right()
{
  Stop();
  delay(100);
  digitalWrite(9,HIGH);
  digitalWrite(10,HIGH);
  delay(300);
  Stop();

}
void Stop()
{
  digitalWrite(8,0);
  digitalWrite(9,0);
  digitalWrite(10,0);
  digitalWrite(11,0);
}
void roundL()
{
  Stop();
  delay(100);
  digitalWrite(8,HIGH);
  digitalWrite(11,HIGH);
}
void roundR()
{
  Stop();
  delay(100);
  digitalWrite(9,HIGH);
  digitalWrite(10,HIGH);
```
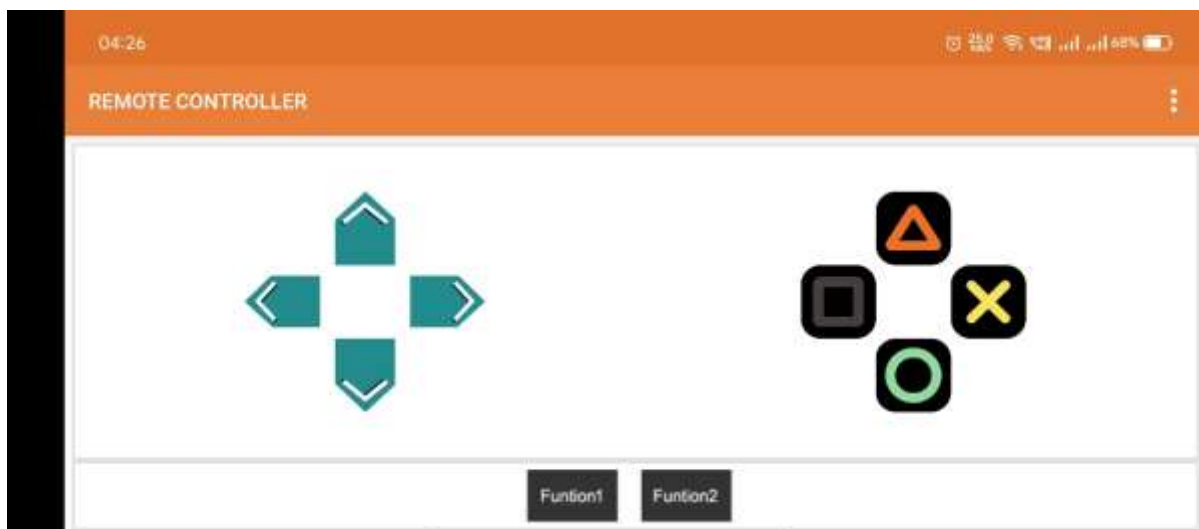
}

All the functions are defined above.

● Working Explanation

Bluetooth controlled car moves according to button touched in the android Bluetooth mobile app. To run this project first we have downloaded Bluetooth app form Google play store.

After installing app, we have selected desired Bluetooth device and then configured the keys. Here in this project, we have used Arduino Bluetooth controller app.

Configure Buttons

Changes saves automatically

L	(Sends '0' on button release)
R	(Sends '0' on button release)
F	(Sends '0' on button release)
B	(Sends '0' on button release)
null	(Sends 'o' on button release)
null	(Sends 'o' on button release)
X	(Sends 'o' on button release)
null	(Sends 'o' on button release)
M	(Sends 'M' on button release)
N	(Sends 'N' on button release)

Now we have configured the keys L, R, F, B, X, M, N ACCORDING their uses, and we have predefined the work of these keys in our code.

● Conclusion

This project can be used for playing purpose for kids and it is affordable too, it may also be improved for racing purpose by adding appropriate battery, motor and tyres.

● **Reference**

https://create.arduino.cc/projecthub/samanfern/bluetooth-controlled-car-d5d9ca