

# **Project Report and Working**

**Project Name: Remote Control Car**

**Project Prepared By: [Sanif Ali Momin]**

## Table of Contents

Introduction .....	<b>3</b>
Technologies .....	<b>3</b>
Project Working .....	<b>5</b>
Programming Part.....	<b>6</b>
Conclusion .....	<b>9</b>
References .....	<b>10</b>

## **Introduction:**

The aim of this project was to make a remote controlled car that uses Arduino and which can be operated using mobile device using Bluetooth module.

This project helps increases our understanding in using Arduino and how Bluetooth module is used to send signals and how motor shield gets instructions and how they works on motors to move in correct direction.

This Project covers the following implementations:

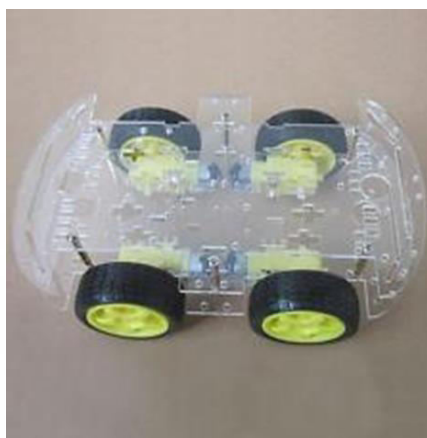
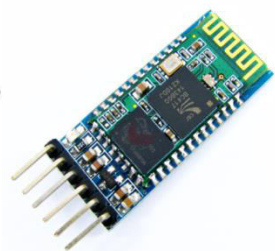
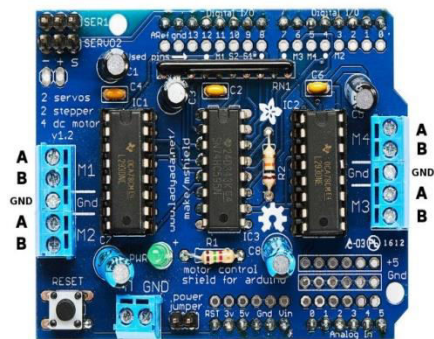
- Moving car in all directions i.e. Forward ,Backward, Right, Left, Forward-Right, Forward-Left, Backward-Right, Backward-Left.
- Varying speed of car from 100 to 255.

## **Technologies:**

### **Component used:**

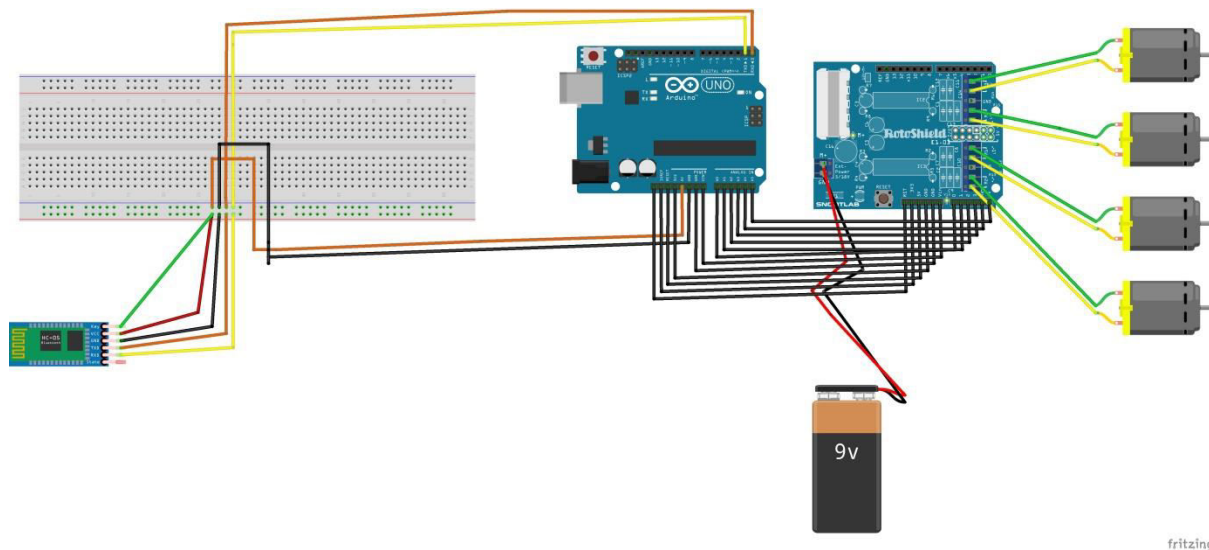
Following Components our included in our RC car.

1. Arduino Uno
2. Motor Sheild L293D
3. Bluetooth Module HC-05
4. Car Kit (4Tyre, Frame)
5. 4 DC Motors
6. Jumper Wires (Male to Female, Male to Male)
7. Battery (made from 3 Li-on 3.7 V cells)
8. Android Device (For operating car)



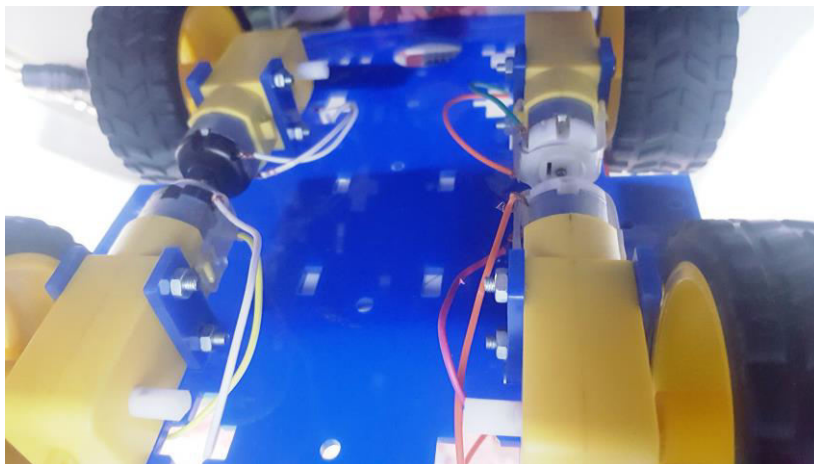
## Project Working:

### SCHEMATIC Diagram:

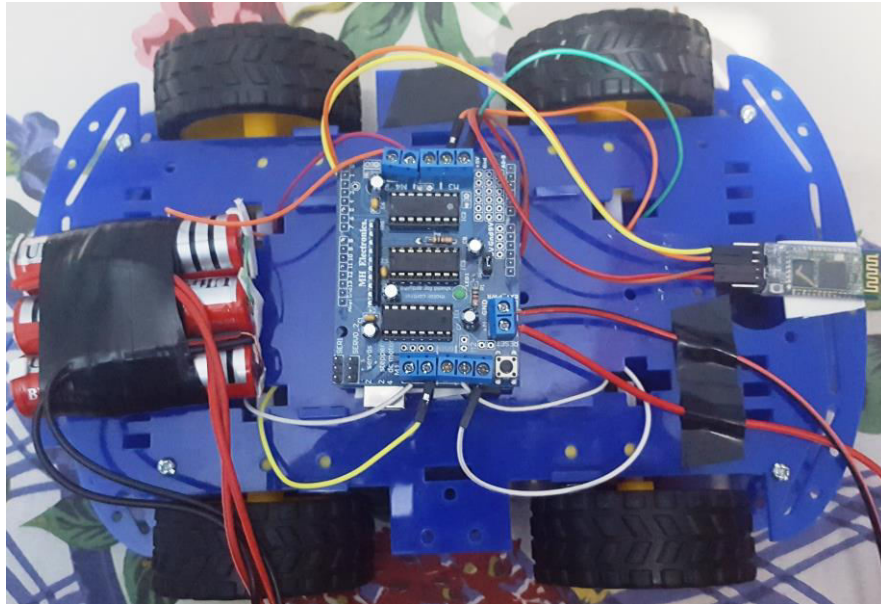


Circuit made according to following schematic diagram.

- Bluetooth is connected to Arduino using 3.3V Vcc pin and GND pin, the Rx(Receiver) of Bluetooth is connected to Tx(Transmitter) of Arduino and Tx to Rx of Arduino so that both devices can communicate easily.
- Motor Shield (L293D) is attached directly through Arduino and all 4 DC motors are connected in the Motor Shield and is giving power by the Battery of 3 li-on 3.7V Cells .



Inner view of car showing connection of Motor Shield with 4 DC- Motors



Outer view of car showing Battery, Motor Shield, Arduino, Bluetooth Module



Side view of car showing connection of motors with tires.

## **Programing Part :**

### **CODE:**

```
#include <AFMotor.h>
```

```
AF_DCMotor motor1(1, MOTOR12_8KHZ);  
AF_DCMotor motor2(2, MOTOR12_8KHZ);  
AF_DCMotor motor3(3, MOTOR12_8KHZ);  
AF_DCMotor motor4(4, MOTOR12_8KHZ);
```

```

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  motor1.setSpeed(150);
  motor2.setSpeed(150);
  motor3.setSpeed(150);
  motor4.setSpeed(150);
}

```

```

int command;
void forward(){
  motor1.run (FORWARD);
  motor2.run (FORWARD);
  motor3.run (FORWARD);
  motor4.run (FORWARD);
  delay(50);
}

```

```

void backward(){
  motor1.run (BACKWARD);
  motor2.run (BACKWARD);
  motor3.run (BACKWARD);
  motor4.run (BACKWARD);
  delay(50);
}

```

```

void right(){
  motor1.run (FORWARD);
  motor2.run (FORWARD);

  motor3.run (RELEASE);
  motor4.run (RELEASE);
  delay(50);
}

```

```

void left(){
  motor1.run (RELEASE);
  motor2.run (RELEASE);

  motor3.run (FORWARD);
  motor4.run (FORWARD);
  delay(50);
}

```

```

void stopcar(){
  motor1.run (RELEASE);
  motor2.run (RELEASE);
  motor3.run (RELEASE);
}

```

```

    motor4.run (RELEASE);
    delay(50);
}
void forwardright(){
    motor1.run (FORWARD);
    motor2.run (FORWARD);

    motor3.run (FORWARD);
    motor4.run (RELEASE);
    delay(50);
}

void forwardleft(){
    motor1.run (RELEASE);
    motor2.run (FORWARD);

    motor3.run (FORWARD);
    motor4.run (FORWARD);
    delay(50);
}

void backwardright(){
    motor1.run (BACKWARD);
    motor2.run (BACKWARD);

    motor3.run (RELEASE);
    motor4.run (BACKWARD);
    delay(50);
}

void backwardleft(){
    motor1.run (BACKWARD);
    motor2.run (RELEASE);

    motor3.run (BACKWARD);
    motor4.run (BACKWARD);
    delay(50);
}
void setspeedofcar(int speedofcar){
    motor1.setSpeed(speedofcar);
    motor2.setSpeed(speedofcar);
    motor3.setSpeed(speedofcar);
    motor4.setSpeed(speedofcar);
    delay(50);
}
void loop() {
    // put your main code here, to run repeatedly:
    if (Serial.available() > 0) {

```



```

    command = Serial.read();
    stopcar();
}
switch(command){
    case 'F':forward(); break;
    case 'B':backward(); break;
    case 'L':left(); break;
    case 'R':right(); break;
    case 'S':stopcar(); break;

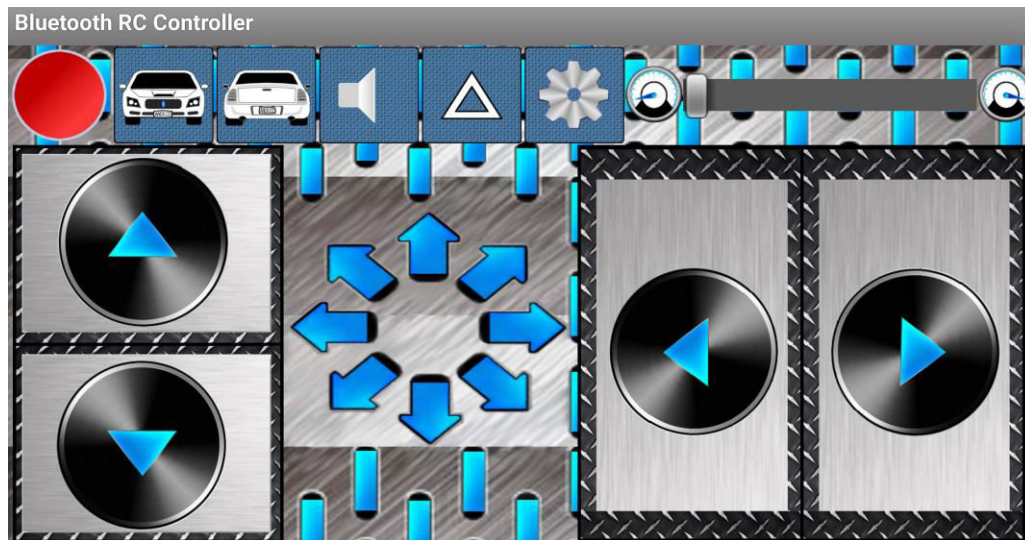
    case 'I':forwardright(); break;
    case 'G':forwardleft(); break;
    case 'J':backwardright(); break;
    case 'H':backwardleft(); break;

    case '0':setspeedofcar(100);break;
    case '1':setspeedofcar(115);break;
    case '2':setspeedofcar(130);break;
    case '3':setspeedofcar(145);break;
    case '4':setspeedofcar(160);break;
    case '5':setspeedofcar(175);break;
    case '6':setspeedofcar(190);break;
    case '7':setspeedofcar(205);break;
    case '8':setspeedofcar(220);break;
    case '9':setspeedofcar(235);break;
    case 'q':setspeedofcar(255);break;
}
}

```

### **Application:**

The application use to operate Remote Control Car is **Bluetooth RC Controller**.



## Conclusion:

The making of this project was learning experience and we learn lot of things in this project and it increases our knowledge regarding Arduino and other components used. The project was up to mark and meets our expectations. It gave us a glimpse of how innovation around us works. As this project was our second, we've got a hand on the working of Arduino and we, therefore, are comfortable in coding the instructions and assembling the components. Now that we have got the idea of how things work, we will surely develop more great useful things in the future.

## Reference:

<https://www.instructables.com/>

<https://www.hackster.io/arduino/projects>

<https://diyhacking.com/diy-projects/arduino-projects>

<https://create.arduino.cc/projecthub>