ARDUINO PROJECT

Introduction

In this project we are going to prepare a Robo using Arduino. Its a car that will be operated with your android phone as a remote. In this project several instruments are used but the most specific is Arduino which works as the brain of the whole system.

Requirements

Hardware Components

1. Smart car set - 1
2. Arduino Board - 1
3. Bluetooth Module (HC-05) - 1
4. Motor driver (L293D dual motor driver) - 1
5. Battery (1.5 V) - 4
6. Bread Board - 2
7. Connecting Wires

- Male to Female - 12
- Male to Male - 6

Software Components

1. Arduino Programming Software (Arduino IDE)
2. Arduino Car Bluetooth Mobile Controller (Android RC Bluetooth Controller)

Connections

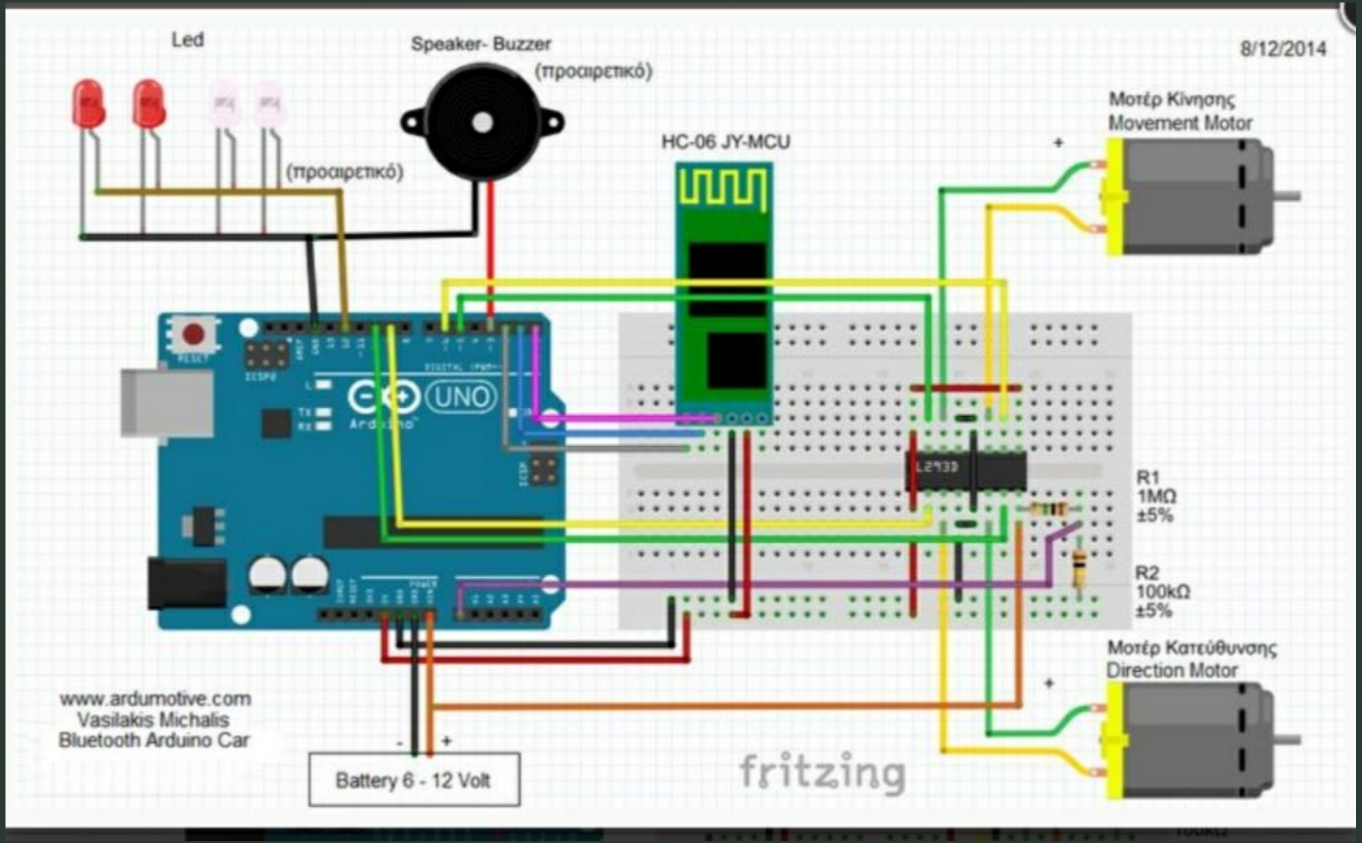
Bluetooth Module (HC-05)

1. Vcc → 5V
2. Gnd → Gnd
3. Tx →Rx
4. Rx → Tx

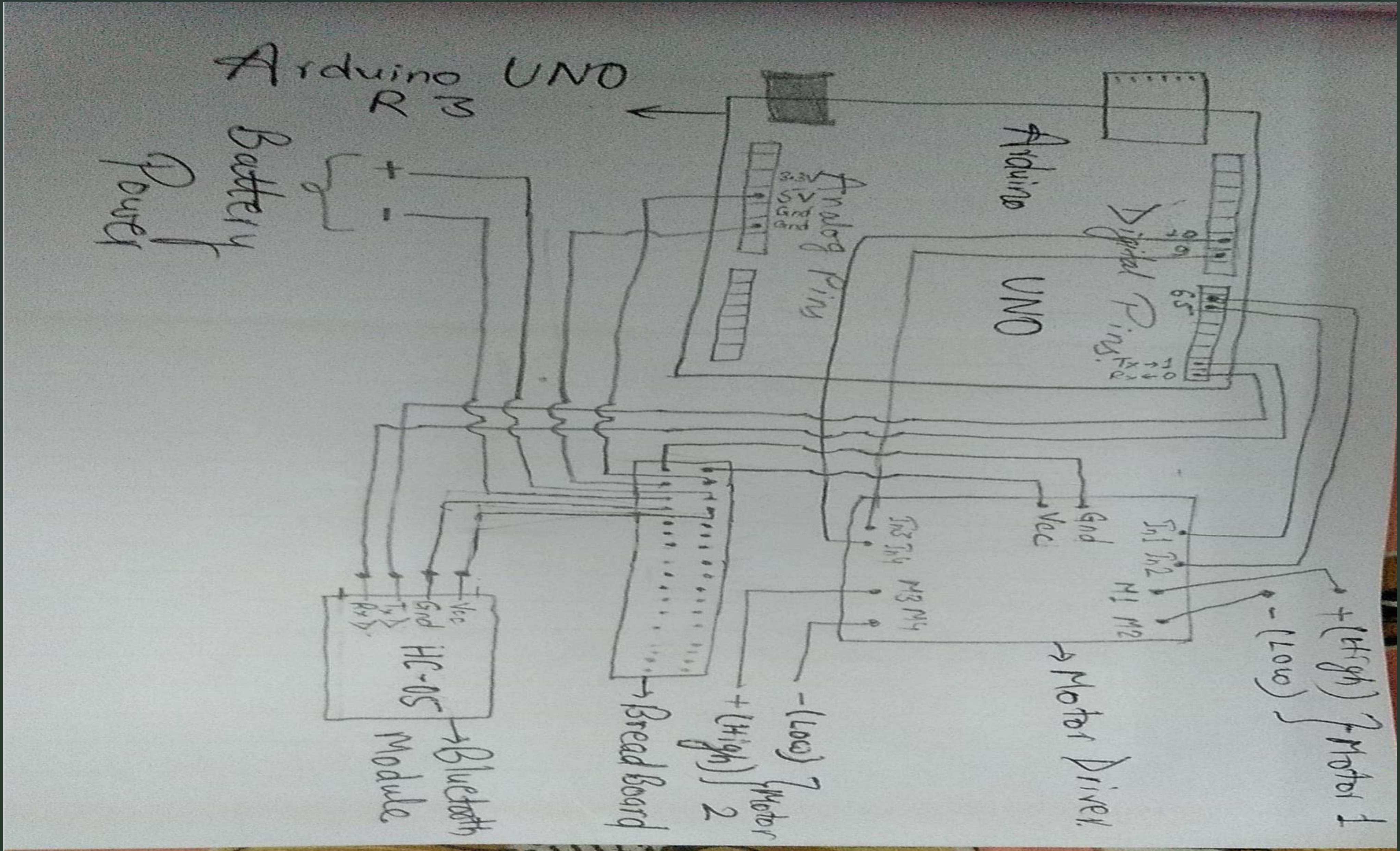
Motor Driver (L293D Dual Motor Controller)

1. Vcc → 5V
2. Gnd → Gnd
3. In1 → Pin 5
4. In2 → Pin 6
5. In3 → Pin 10
6. In4 → Pin 9

Complete Connection Diagram



Schematic Diagram



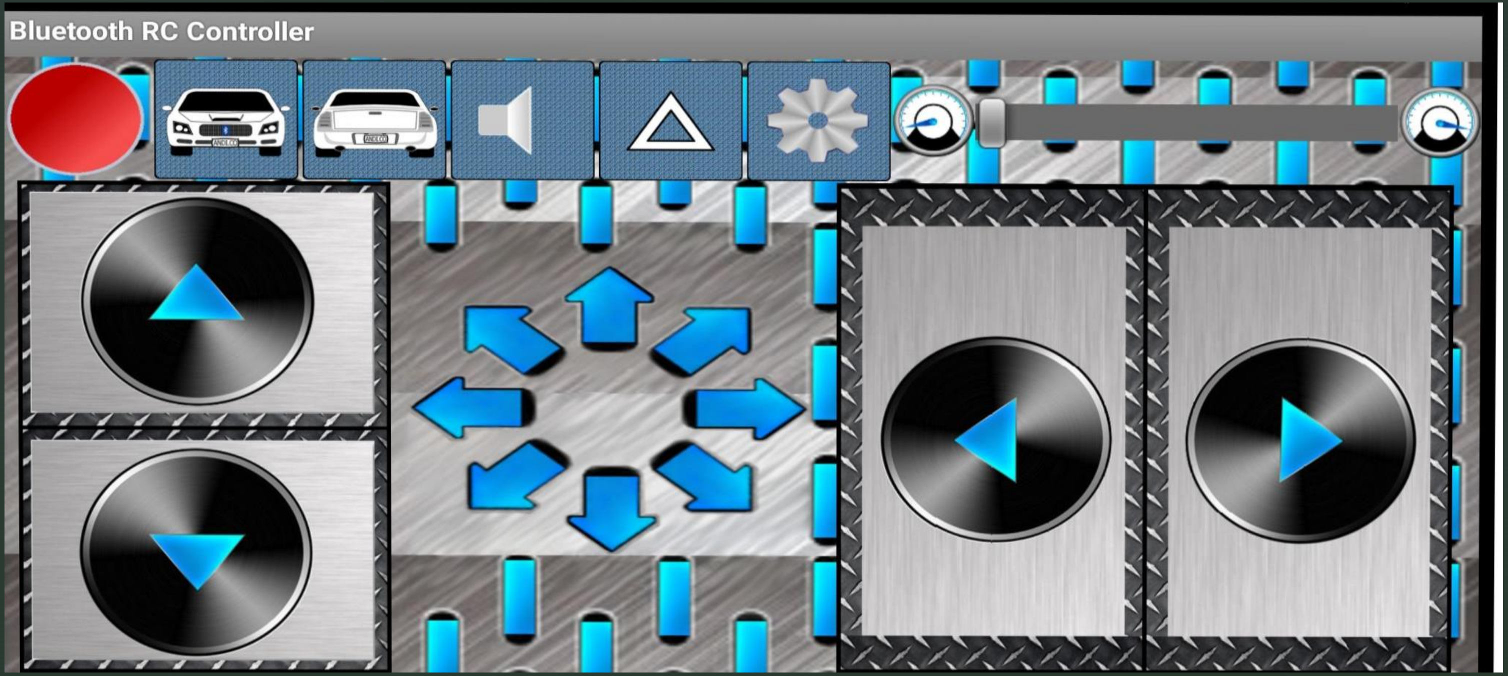
Arduino Code Programming

```
//L293 Connection      const int motorA1  = 5;  // Pin  2 of L293   const int motorA2  = 6;  // Pin  7 of L293   const int motorB1  = 10; // Pin 10 of L293  const int motorB2  = 9;  // Pin 14 of L293//Leds
connected to Arduino UNO Pin 12  const int lights  = 12;//Buzzer / Speaker to Arduino UNO Pin 3  const int buzzer = 3 ;   //Bluetooth (HC-06  JY-MCU) State pin on pin 2 of Arduino   const int
BTState = 2;//Calculate Battery Level   const float maxBattery = 8.0;// Change value to your max battery voltage level!   int perVolt;                 // Percentage variable    float voltage  = 0.0;         //
Read battery voltage   int level;// Use it to make a delay... without delay() function!  long previousMillis  = -1000*10;// -1000*10=-10sec.  to read the first value.  If you use 0 then you will take the
first value after 10sec.    long interval  = 1000*10;        // interval at which to read battery voltage, change it if you want! (10*1000=10sec)   unsigned long currentMillis;   //unsigned long
currentMillis;//Useful Variables   int i=0;  int j=0;  int state;  int vSpeed=200;      // Default speed, from 0 to 255void setup() {    // Set pins as outputs:    pinMode(motorA1,  OUTPUT);
pinMode(motorA2,  OUTPUT);     pinMode(motorB1,  OUTPUT);     pinMode(motorB2,  OUTPUT);     pinMode(lights,  OUTPUT);      pinMode(BTState,  INPUT);         // Initialize serial communication
at 9600 bits per second:    Serial.begin(9600);}  void loop() {  //Stop car when connection lost or bluetooth disconnected//      if(digitalRead(BTState)==LOW)  { state='S'; } //Save income data to
variable 'state'    if(Serial.available()  > 0){            state = Serial.read();        }    //Change speed if state is equal from 0 to 4. Values must be from 0 to 255 (PWM)     if (state == '0'){       vSpeed=0;}
else if (state == '1'){       vSpeed=100;}     else if (state == '2'){       vSpeed=180;}     else if (state == '3'){       vSpeed=200;}     else if (state == '4'){       vSpeed=255;}
/***********************Forward****************************/   //If state is equal with letter 'F', car will go forward!     if (state == 'F') {            analogWrite(motorA1,  vSpeed);  analogWrite(motorA2,  0);
analogWrite(motorB1,  0);      analogWrite(motorB2,  0);     }  /**********************Forward  Left************************/   //If state is equal with letter 'G', car will go forward left    else if (state == 'G') {
                         analogWrite(motorA1,  vSpeed);  analogWrite(motorA2,  0);          analogWrite(motorB1,  200);    analogWrite(motorB2,  0);     }  /**********************Forward
Right************************/    //If state is equal with letter 'I', car will go forward right    else if (state == 'I') {                  analogWrite(motorA1,  vSpeed);  analogWrite(motorA2,  0);
analogWrite(motorB1,  0);      analogWrite(motorB2,  200);     }  /***********************Backward****************************/    //If state is equal with letter 'B', car will go backward     else if (state == 'B')
{                        analogWrite(motorA1,  0);   analogWrite(motorA2,  vSpeed);          analogWrite(motorB1,  0);   analogWrite(motorB2,  0);     }  /**********************Backward
Left************************/    //If state is equal with letter 'H', car will go backward left    else if (state == 'H') {                  analogWrite(motorA1,  0);   analogWrite(motorA2,  vSpeed);
analogWrite(motorB1,  200); analogWrite(motorB2,  0);     }  /**********************Backward  Right************************/   //If state is equal with letter 'J', car will go backward right     else if (state ==
'J') {                   analogWrite(motorA1,  0);   analogWrite(motorA2,  vSpeed);          analogWrite(motorB1,  0);   analogWrite(motorB2,  200);     }
/***************************Left*****************************/   //If state is equal with letter 'L', wheels will turn left    else if (state == 'L') {     analogWrite(motorA1,  0);   analogWrite(motorA2,  0);
analogWrite(motorB1,  200); analogWrite(motorB2,  0);     }  /***************************Right*****************************/   //If state is equal with letter 'R', wheels will turn right    else if (state == 'R') {
                         analogWrite(motorA1,  0);   analogWrite(motorA2,  0);         analogWrite(motorB1,  0);   analogWrite(motorB2,  200);                                    }
/************************Lights*****************************/   //If state is equal with letter 'W', turn leds on or of off     else if (state == 'W') {      if (i==0){            digitalWrite(lights, HIGH);           i=1;      }
else if (i==1){          digitalWrite(lights, LOW);           i=0;      }      state='n';    }  /**********************Horn  sound**************************/   //If state is equal with letter 'V', play (or stop) horn sound
else if (state == 'V'){       if (j==0){           tone(buzzer,  1000);//Speaker on          j=1;      }      else if (j==1){          noTone(buzzer);     //Speaker off          j=0;      }      state='n';      }
/************************Stop*****************************/   //If state is equal with letter 'S', stop the car    else if (state == 'S'){         analogWrite(motorA1,  0);  analogWrite(motorA2,  0);
analogWrite(motorB1,  0);  analogWrite(motorB2,  0);    }  /***********************Battery*****************************/   //Read battery voltage every 10sec.    currentMillis = millis();    if(currentMillis  -
(previousMillis)  > (interval))  {       previousMillis  = currentMillis;         //Read voltage from analog pin A0 and make calibration:       voltage  = (analogRead(A0)*5.015  / 1024.0)*11.132;
//Calculate percentage...       perVolt  = (voltage*100)/  maxBattery;       if      (perVolt<=75)              { level=0; }       else if (perVolt>75  && perVolt<=80)  { level=1; }    //       Battery level       else if
(perVolt>80  && perVolt<=85)  { level=2; }    //Min -----------------------    Max       else if (perVolt>85  && perVolt<=90)  { level=3; }    //    | 0 | 1 | 2 | 3 | 4 | 5 | >       else if (perVolt>90  && perVolt<=95)
{ level=4; }    //    -----------------------         else if (perVolt>95)                { level=5; }          Serial.println(level);         }    }
```

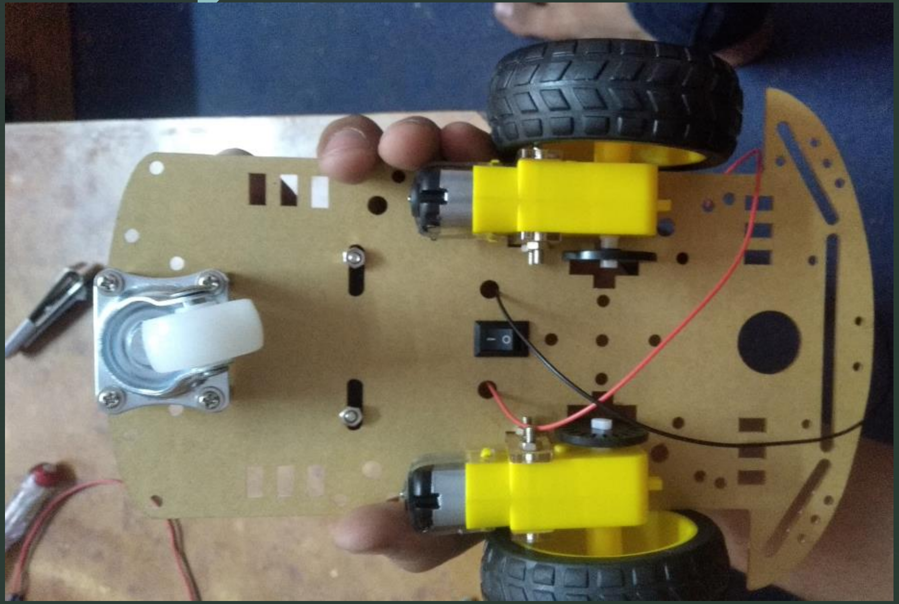
SOURCE: www.ardumotive.com/Bluetooth-RC-Car/

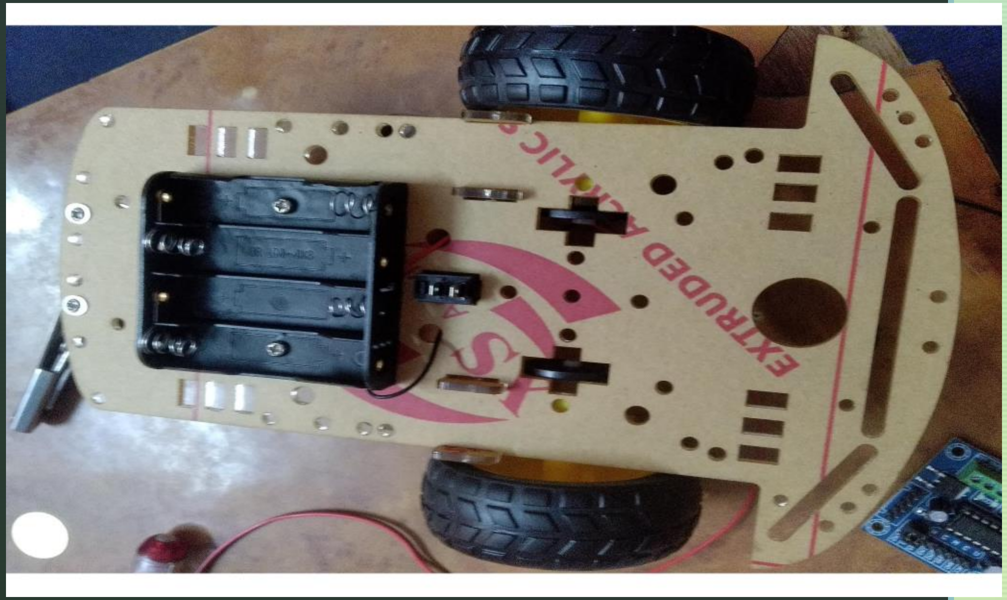
Bluetooth Controller App

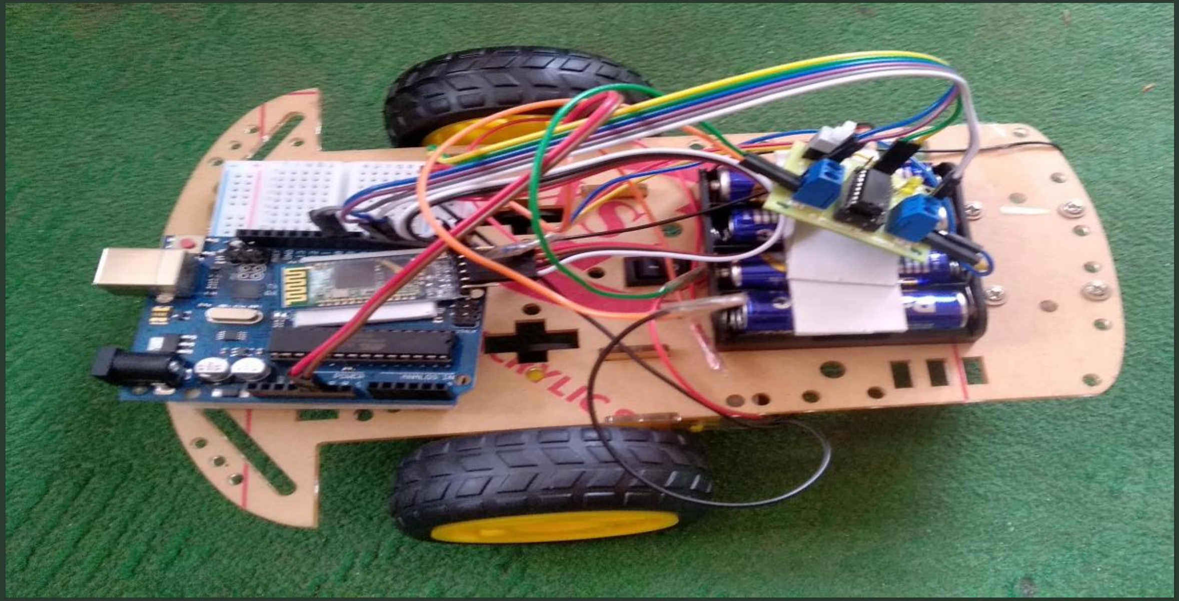
Go to Google and download and download Arduino Bluetooth RC Car



Completed Car View







Application

1. Can be used as a remote security device.
2. Can be used in examining mini – tunnels.
3. Can be used in search and find.
4. Can be used in home security or as a alert at a time of any environmental changes.
5. Can be used for search and rescue and to explore unreachable places.
6. Can be used by millitary for finding bombs or as a remote surveillance.

Future Scope Of The Project

1. Can be controlled by phone’s accelerometer sensor to control the car via movement of the phone.
2. Automation of household tasks.
3. Can be used as better security and surveillance by implementing more functioning systems such as alarm trigger system.
4. Use camera to transmit frames back to android application.
5. Can be implemented using WiFi connectivity as Bluetooth is of low bandwidth

Conclusion

The Arduino is an open source device that has been the brain for the numerous projects. As the mobile devices are becoming more advanced, using them for controlling Robots and other wireless devices is likely to be a large trend.It can be concluded that this idea of smart living will let us control our sorroudings remotely and wirelessly.The Arduino has everything that is required by the user which includes it’s inbuilt converters, I/O Pins, etc. With the combination of Arduino and Bluetooth we can control over many other things like home lightings, air conditioner and many more through our cell phones.