# 4WD Robot Car Guide

V1.0 – The components listed in this document may become superseded over time with newier model. if this occurs and are unable to adapt the component to this guide, lodge a support request at: https://www.phippselectronics.com/contact-us

## Introduction:

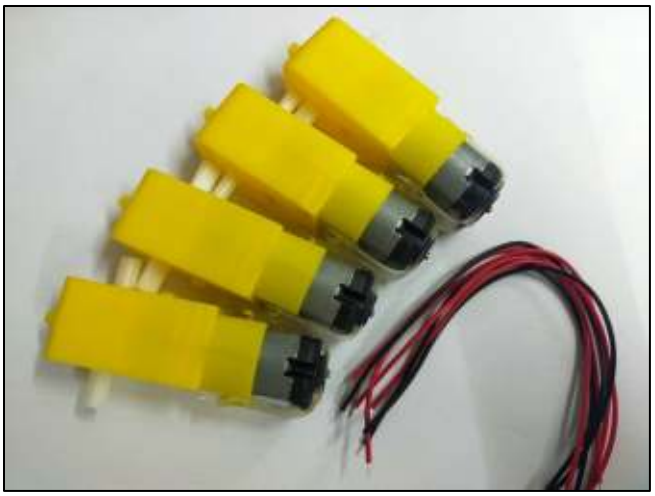
This kit includes multiple configuration options where you can assemble the robot car to act as any of the following:

1. Bluetooth Controlled Robot Car
2. Obstacle Avoiding Robot Car
3. Line Following Robot Car
4. IR Remote Controlled Robot Car

The configurations and codes are given in this document.

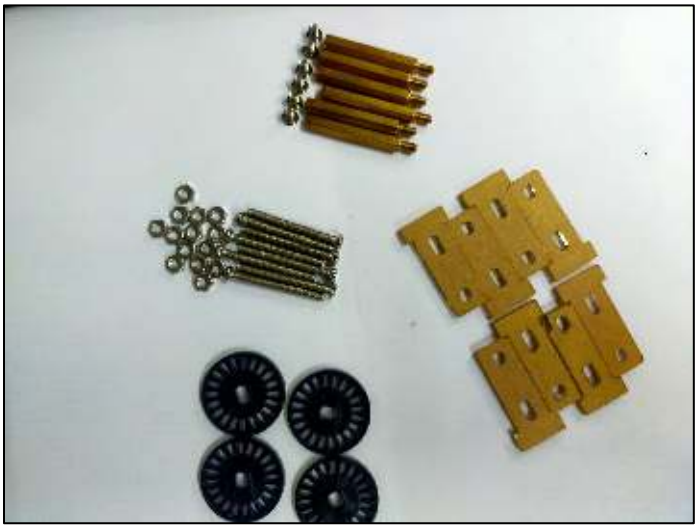
## Components:

1. 4 x Geared motor – the geared motors make the car move forward, backward, or sideward.

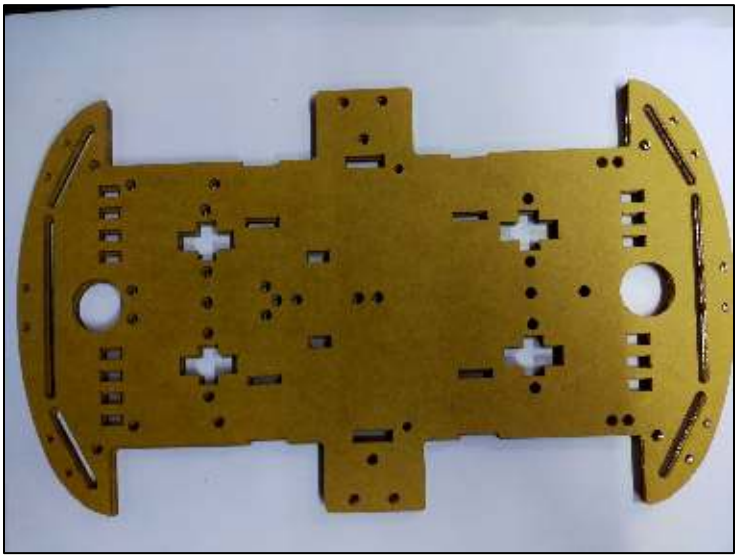


2. 4 x Tire – the tires add grip on the track while the car is moving

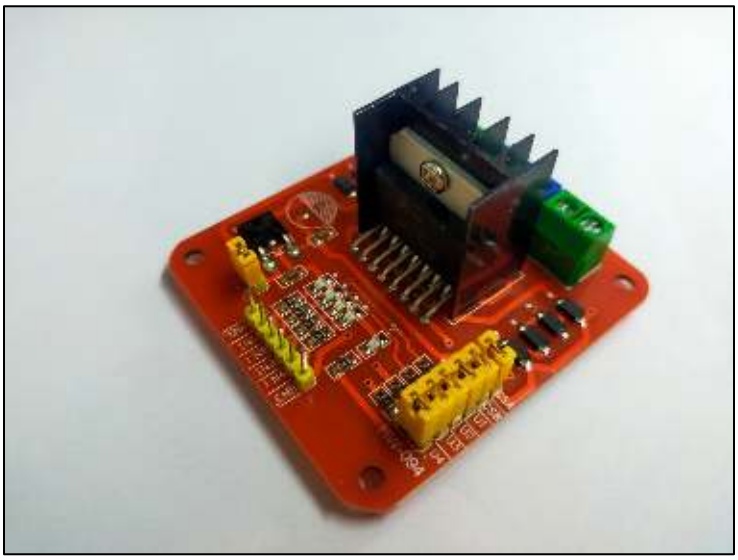
3. 4 x Motor fixing – the motor fixings make sure that the motor is installed correctly
4. 13 x Copper pillar (6 x 35mm / 7 x 6mm) – copper pillars are used to give the distance between the top and bottom chassis
5. 1 x 3mm screws kit – 3mm screws kit are used to make sure that the chassis is tight and robust but still available for disassembly.



6. 2 x (255 x 160 x 3mm) transparent acrylic glass plate – the transparent acrylic glass plate is stylish and light but strong making the car more durable



7. 1 x L298N motor driver – the L298N motor driver receives the signal from the microcontroller to move each of the motors in the car.

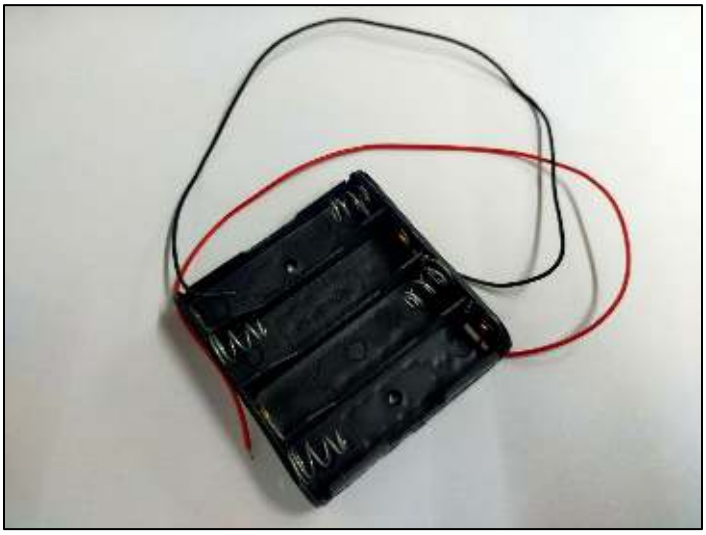


8. 1 x USB cable (100cm) – A USB cable is used to connect the Arduino to the pc when uploading the program

9. 1 x UNO328 controller board – the UNO328 controller board is the programmable microcontroller that can be programmed based on the user's wants


10. 1 x sensor board – the sensor board is where all the sensors are connected to make the car chassis compact and easy to assemble

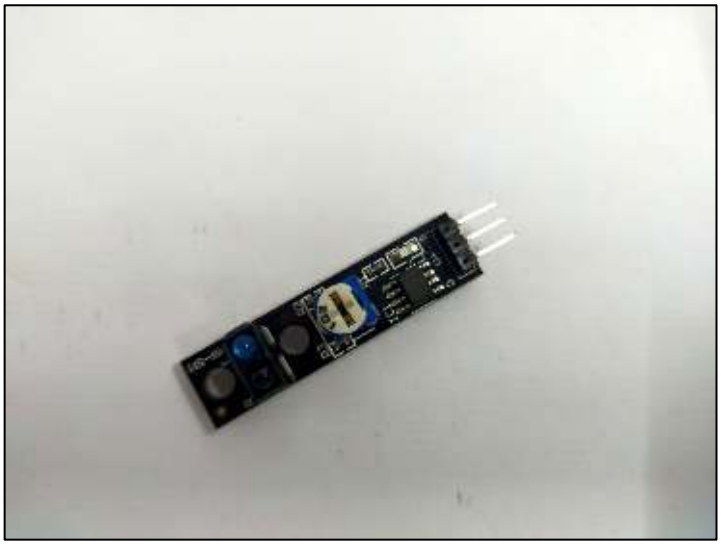
11. 1 x 4 AA Battery holder – the Battery holder is where the battery is safely installed, making it easy to replace when the battery is drained



12. 1 x Holder kit – the holder kit holds the additional steering gear and ultrasonic sensor

13. 1 x Steering gear – the steering gear is a servo motor that moves the ultrasonic sensor from left to right

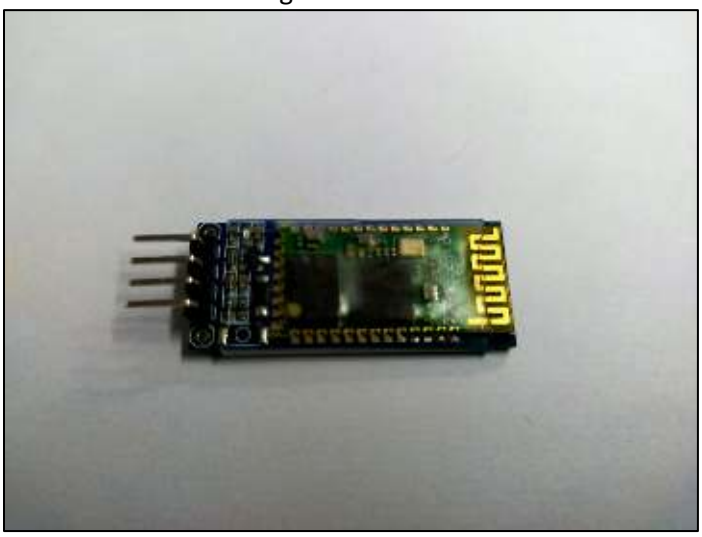
14. 2 x Line inductive module – the Line inductive modules are used to make the car that follows a dark line



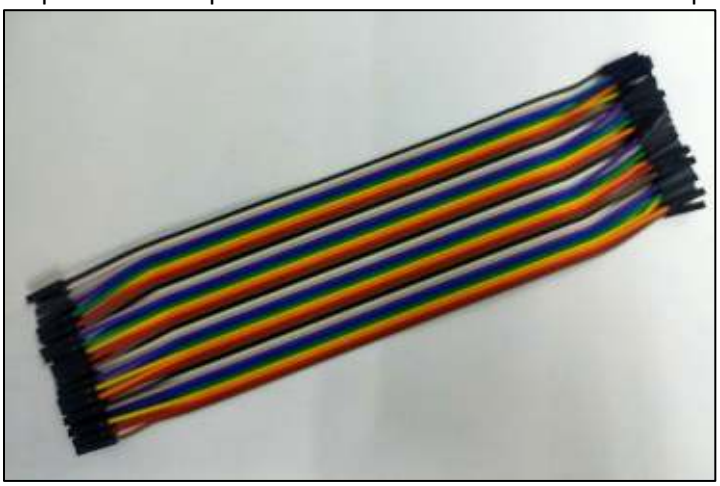
15. 1 x Infrared receiver module – the Infrared receiver module is used to receive commands from the MCU remote control

16. 1 x MCU remote control – the MCU remote control sends signal commands to the car to control the movements


17. 1 x Bluetooth adapter – the Bluetooth adapter receives the signal from your Bluetooth device that can send control signals to the car

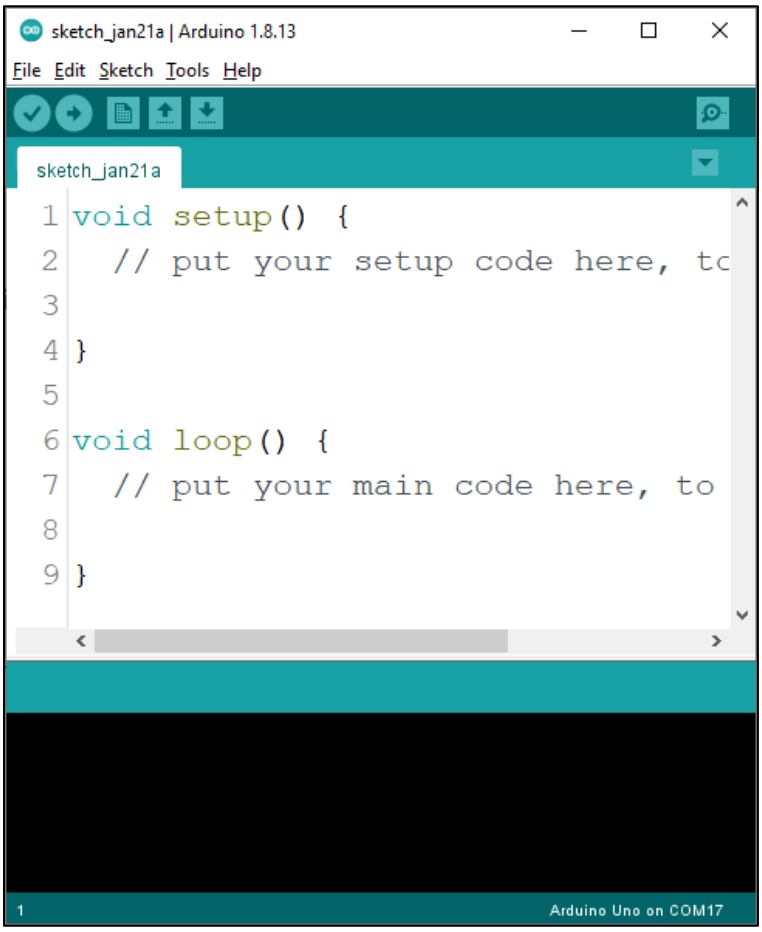


18. Dupont line – Dupont lines are used to connect each component to one another.

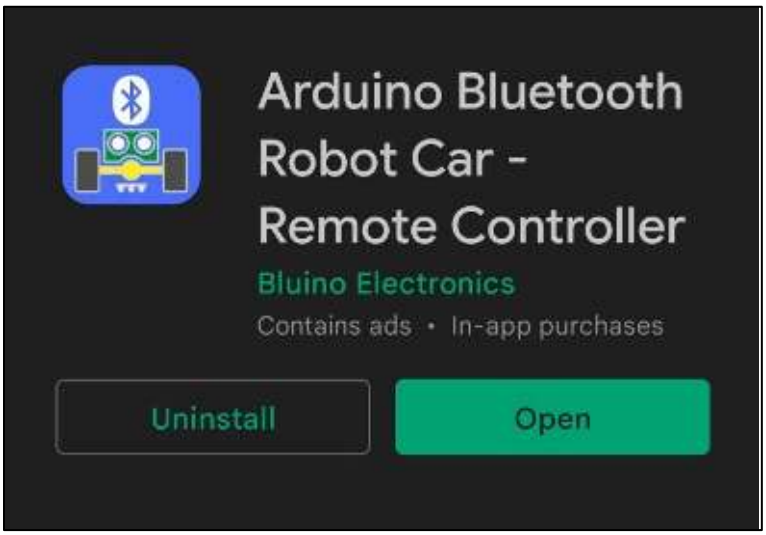


Software:

1. Arduino IDE




2. Arduino Bluetooth Car – Remote Control



Tools:

1. Soldering Iron and Lead – this will be used to solder the wires to the geared motor

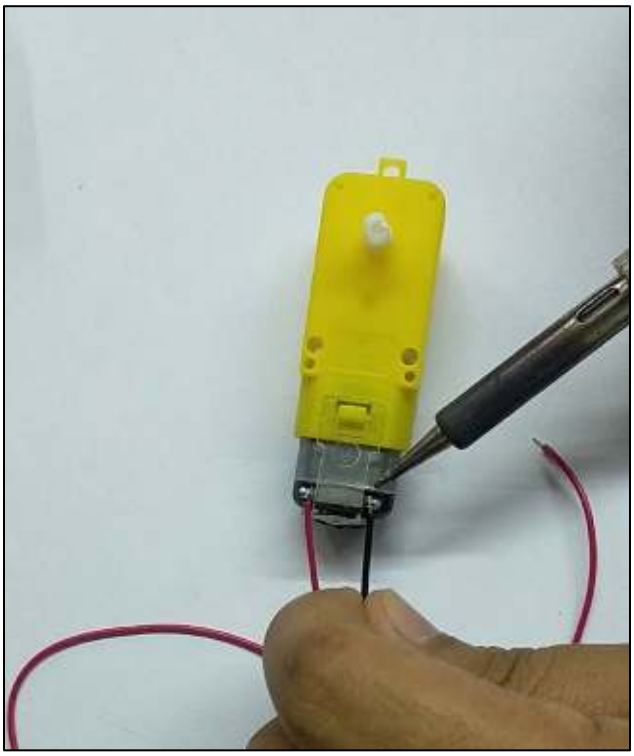

2. Wire Cutter – this will be used to cut some wires for the connection

3. Screw Driver – this will be used to tighten or losen the screws in the assembly

4. Glue gun and glue stick – this will be used to glue in some of the components

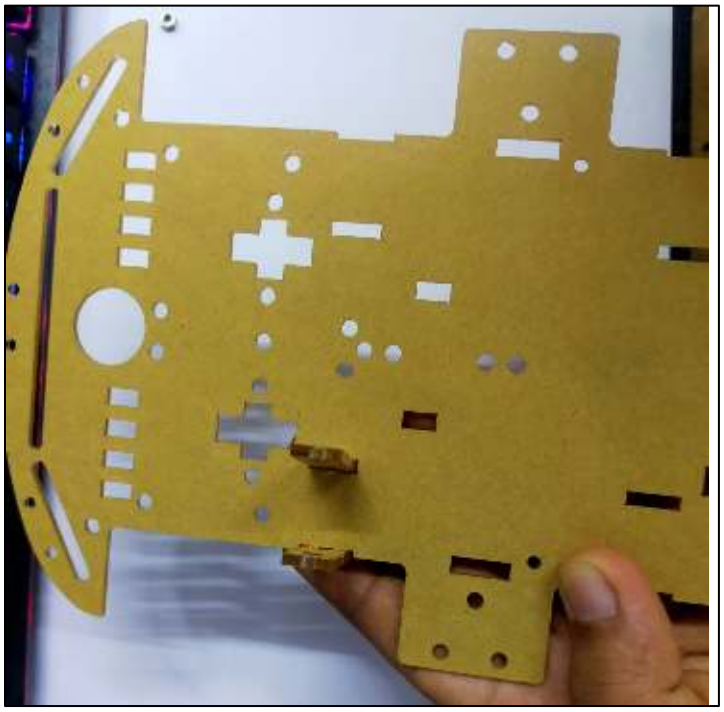
## Assembly:

1. Unbox the OBSTACLE AVOIDING BLUETOOTH CONTROLLED DIY 4WD ROBOT KIT and take out all the listed materials above.
2. Solder the wires on the Geared motor. Make sure all wires are connected in the same way for the 4 gears (Red wire on left and Black Wire on right).

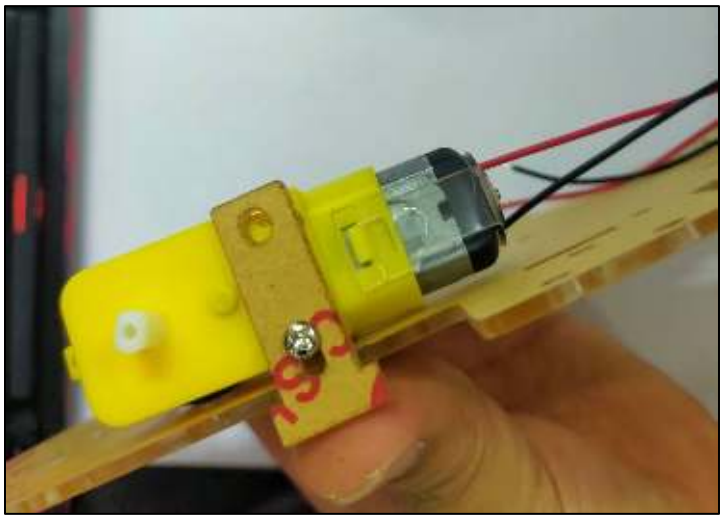


3. Install the geared motor on one of the transparent acrylic glass plates with screws.

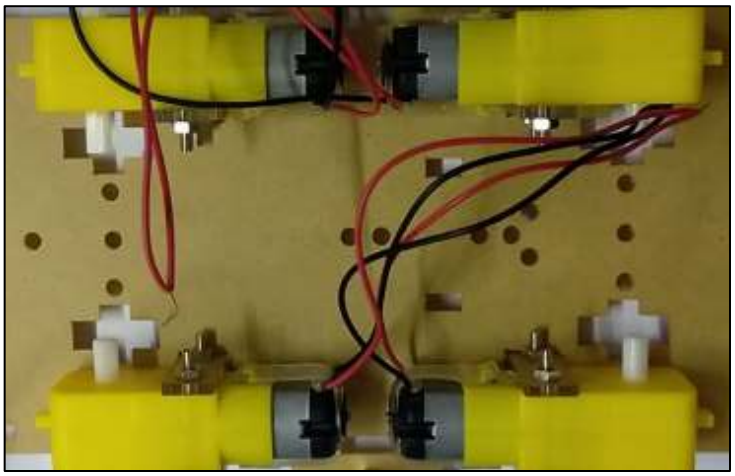
    a. Install a pair of the acrylic geared motor holder ong the transparent acrylic glass pate. Insert one on the hole from below and put the one beside it. These will hold the geared motors in position once screwed.



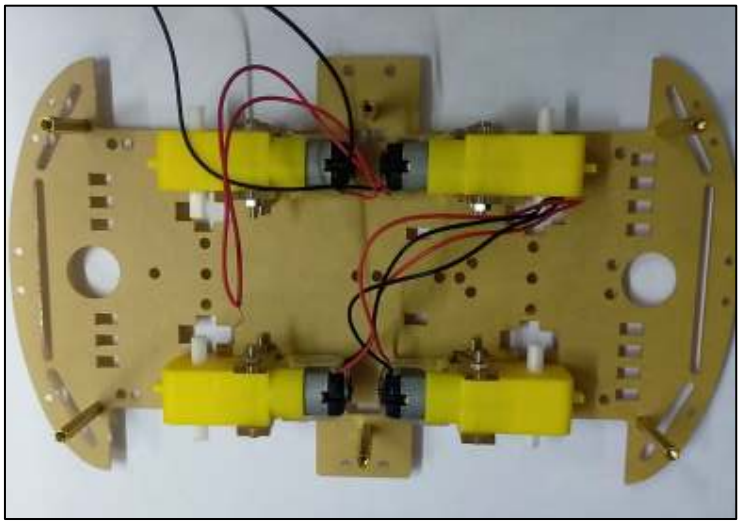
b. Insert the geared motor between the two holders and screw it through the holes.



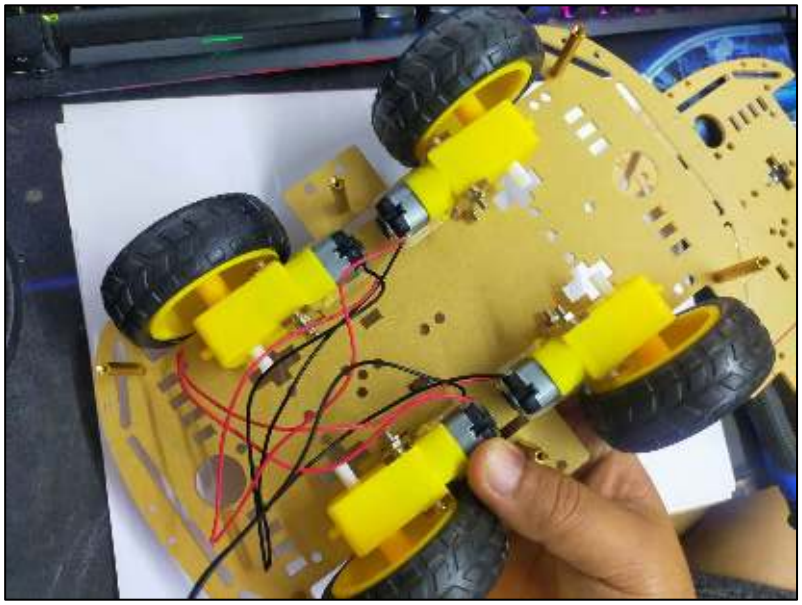
c. Repeat the process until all 4 geared motors are installed.
d. Using the wire cutter strip the tip of the wires and connect the two geared motors located on the same side in parallel (Red wire to red wire and Black wire to Black Wire). This will ensure the proper rotation of the motors for the robot car to move or turn.



4. Install the copper pillars on 4 corners and 2 sides of the transparent acrylic glass plate.

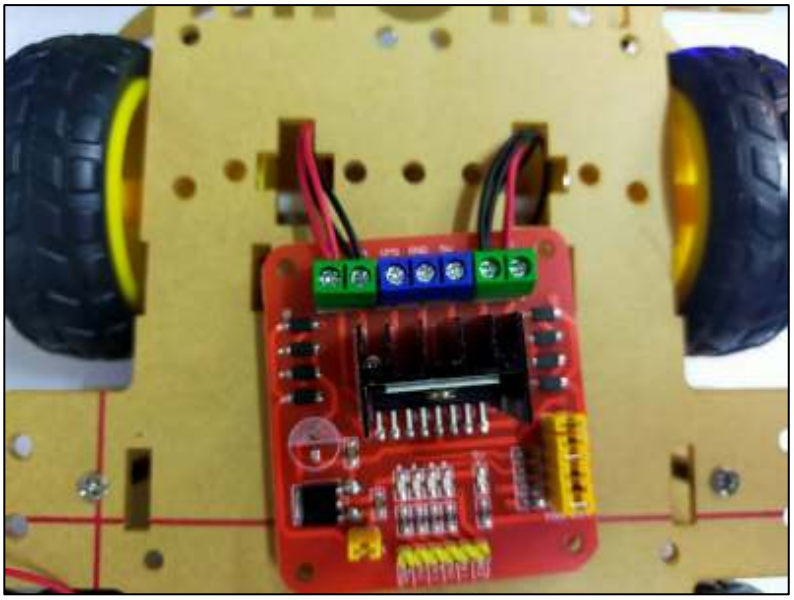


5. Install the wheels.

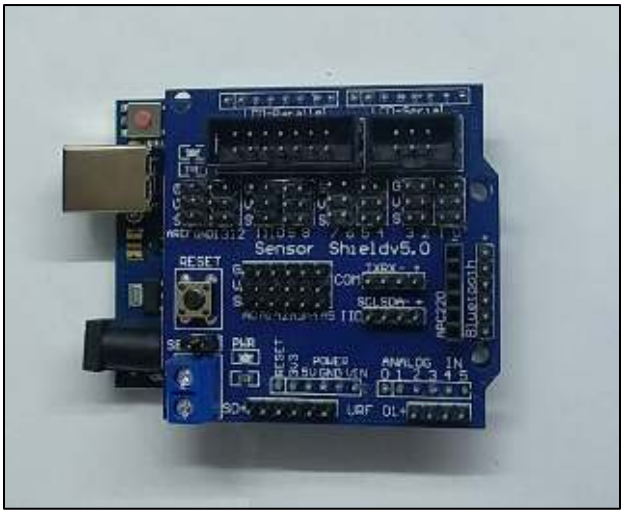


6. Screw the second transparent acrylic glass plate on top of the first one separated by the copper pillars. Make sure to separate the wires of the motors on the left from the motors from the right.

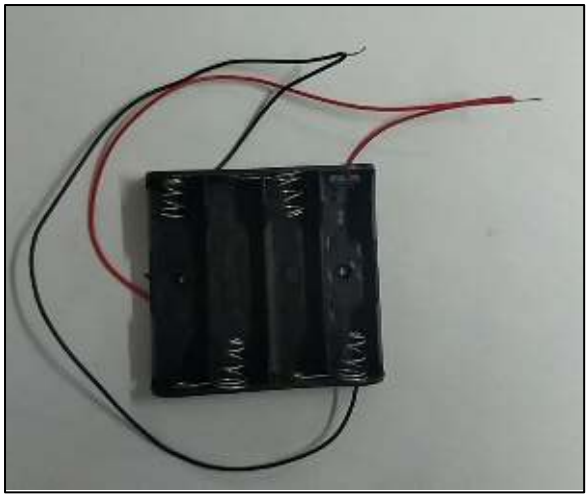
7. Connect the motors on the L298N motor driver using the following connection based on the picture.
   Left motors (Red - Black) to MotorA
   Right motors (Black - Red) to MotorB



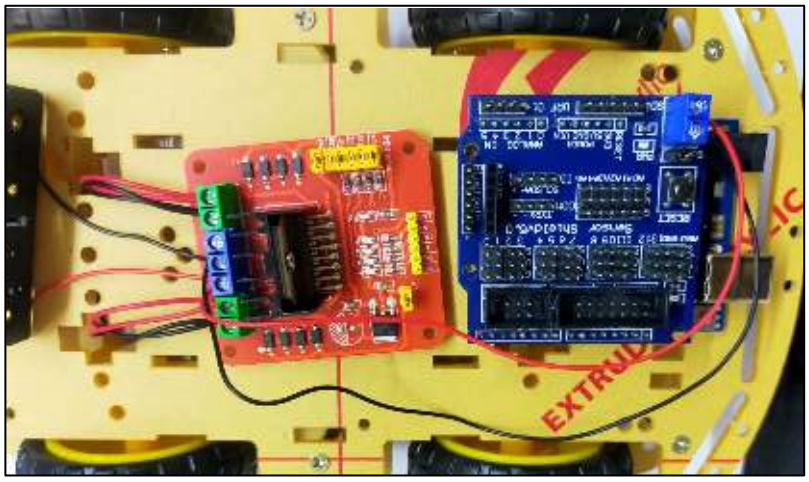
8. Install the sensor board on top of the Arduino board. Just align the pins and press it.



9. Connect the wires of the battery holder to the L298N motor driver and the Arduino sensor shield power pins.
    a. Using wire cutter, cut the excess wire on the battery holder and create a wire extension. This will be used as a connection from the battery to the Arduino sensor shield and the L298N motor driver.

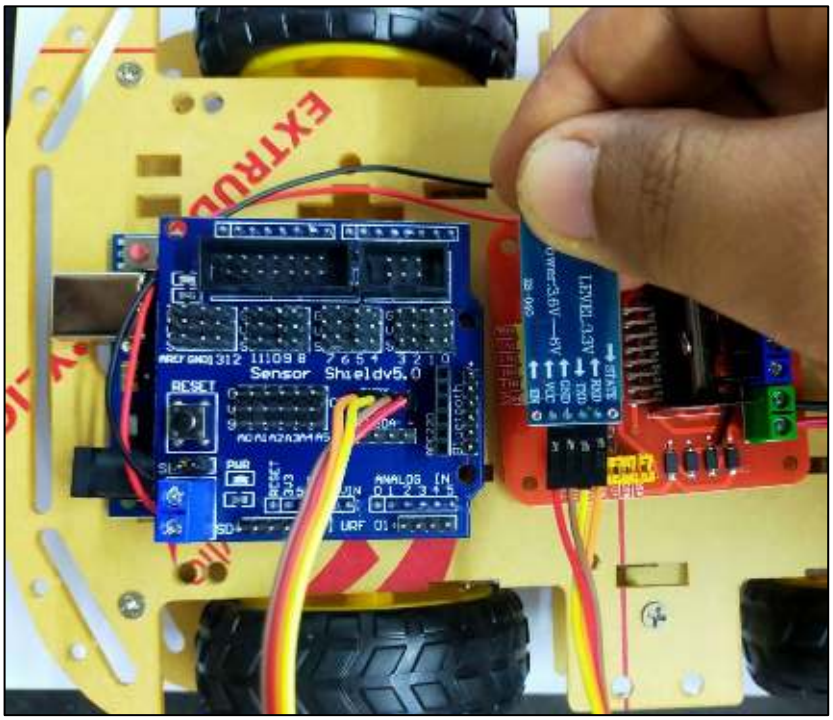


    b. Connect the short end to the L298N motor driver (Red to VMS and Black to GND).
    c. Connect the long end to the Arduino sensor shield (Red to VCC and Black to GND).



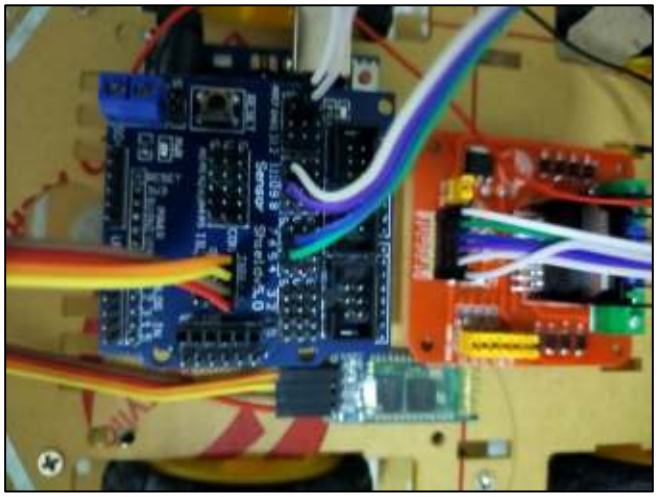
10. Using the 40cm Female to Female Dupont line connect the bluetooth module to the arduino sensor shield using the following connection. (Look for the COM pins)

| **Bluetooth** | **ArduinoSensor Shield (COM)** |
|---|---|
| RX | TX |
| TX | RX |
| GND | - |
| VCC | + |




11. Connect the L298N to the Arduino Sensor Shield using the following connection.

| **L298N Motor Driver** | **Arduino Sensor Shied** |
|---|---|
| ENA | V (Any 5V Pin) |
| IN1 | 5 |
| IN2 | 6 |
| IN3 | 10 |
| IN4 | 11 |
| ENB | V (Any 5V Pin) |



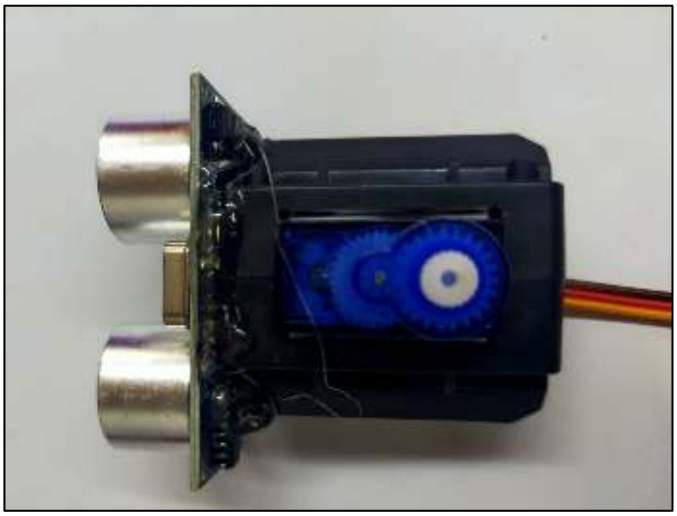
At this poing you can use the robot car as a 4WD bluetooth controlled car. The following steps are the installation of additional components to give you an option to use different functionality for the robot car.

12. Assemble the servo motor into the holder kit to create the stering of the ultrasonic sensor. Clup servo motor between the holders.

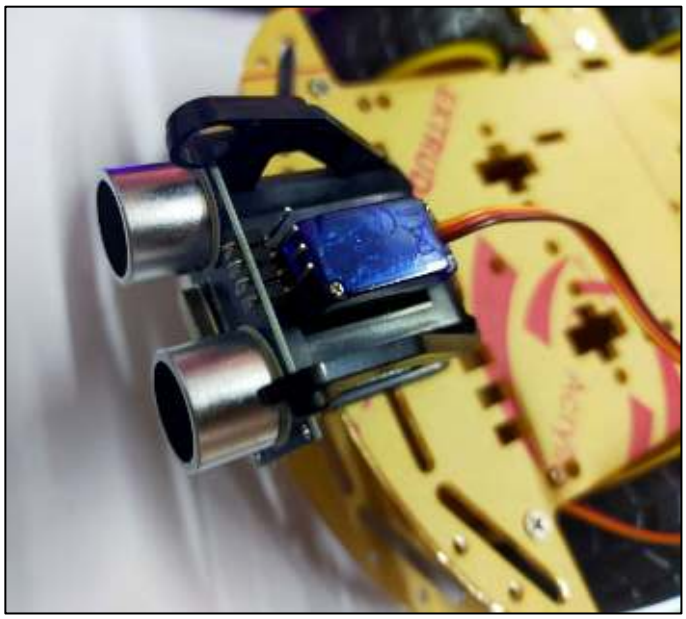
13. Using the wire cutter trim the excess gear for the servo and screw it to the bottom part of the holder.

14. Screw the bottom holder to the front part of the chasis. If the holes are too small for the bolts and knots, use the screw driver to expand the hole.

15. Using the glue gun, put some glue on the back of the ultrasonic sensor to fix it into the holder.



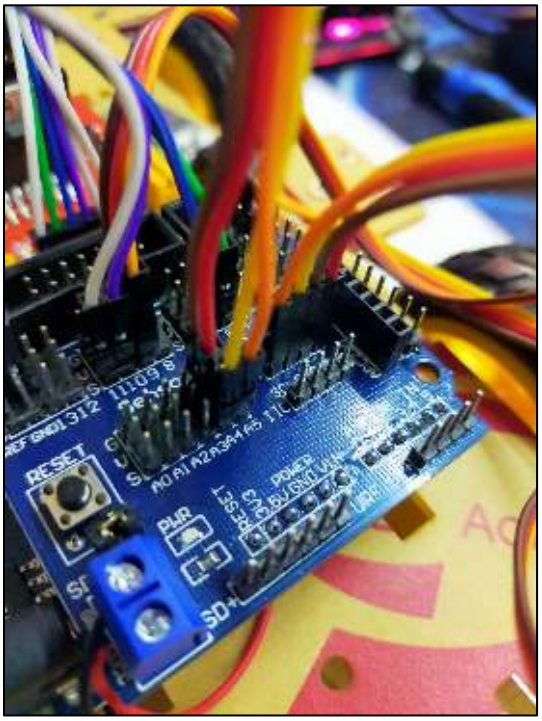
16. Connec the servo motor to the installed bottom holder. No additional screws needed here, the gear are tight enough to hold the servo. Also make sure that the servo can move 180 degrees facing the front.

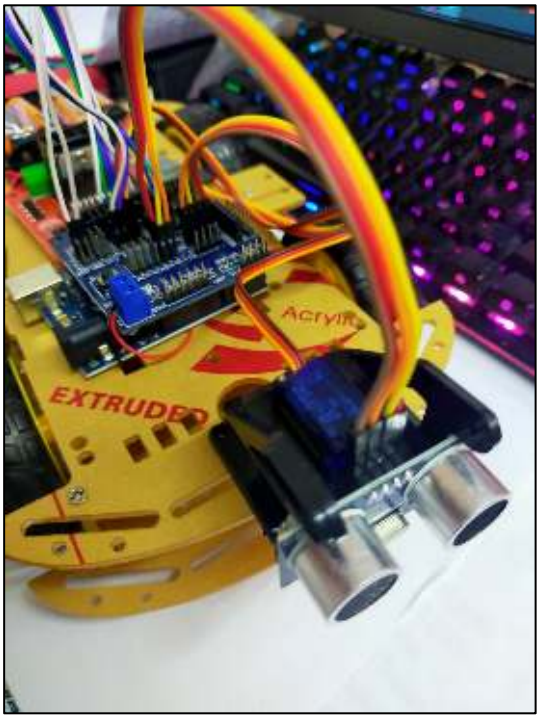


17. Connect the Servo motor and the Ultrasonic Sensor the Arduino Sensor shield using the following connections:

| **Servo Motor** | **Arduino Sensor Shield** |
|---|---|
| Brown Wire | G (Any Ground Pin) |
| Red Wire | V (Any 5V Pin) |
| Orange Wire | 9 |

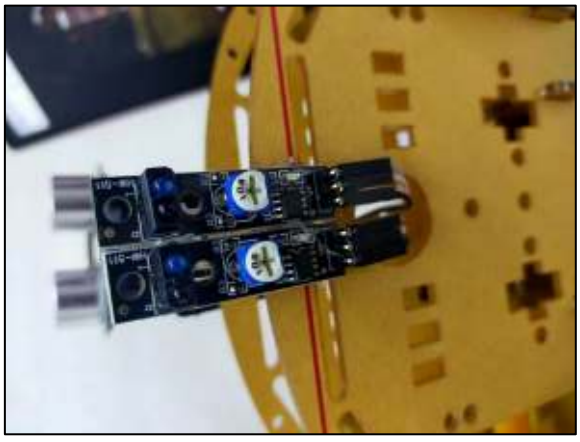
| Ultrasonic Sensor | Arduino Sensor Shied |
|---|---|
| Vcc | V (Any 5V Pin) |
| Trig | A4 |
| Echo | A5 |
| GND | G (Any Ground Pin) |




**At this point, you have assembled the 4WD Robot car as an obstacle-avoiding robot car.**

18. Using the glue gun, glue the 2 ir sensors at the bottom part of the chassis.



19. Using female to female dupont wires, connect the 2 ir sensors to the arduino sensor shield using the following connections

| IR Sensor (Right) | Arduino Sensor Shied |
|---|---|
| G | G (Any Ground Pin) |
| V+ | V (Any 5V Pin) |
| S | 12 |

| IR Sensor (Left) | Arduino Sensor Shield |
|---|---|
| G | G (Any Ground Pin) |
| V+ | V (Any 5V Pin) |
| S | 13 |

**At this point, you have assembled the 4WD Robot car as a line following robot car.**

20. Using female to female dupont wires, connect the ir remote sensor to the arduino sensor shield using the following connections.

| IR Sensor (Right) | Arduino Sensor Shield |
|---|---|
| - | G (Any Ground Pin) |
| + | V (Any 5V Pin) |
| S | 12 |

**At this point, you have assembled the 4WD Robot car controlled by an IR Remote.**

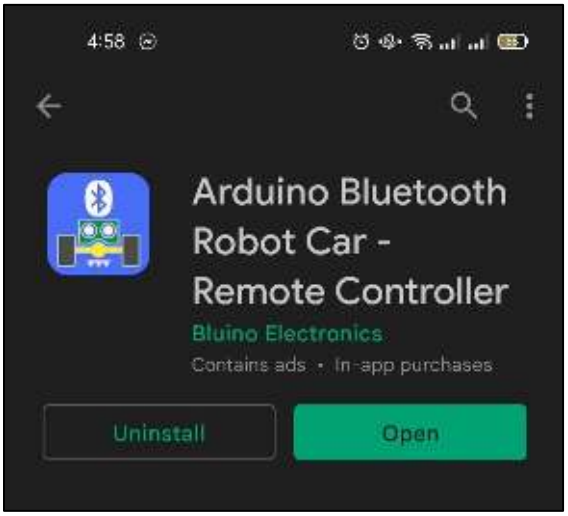
## Programming / Code:

1. Using the arduino IDE. Upload the code in the arduino.
    a. Connect the Arduino to your pc using the USB cable.
    b. Open the Arduino IDE and copy the following code.

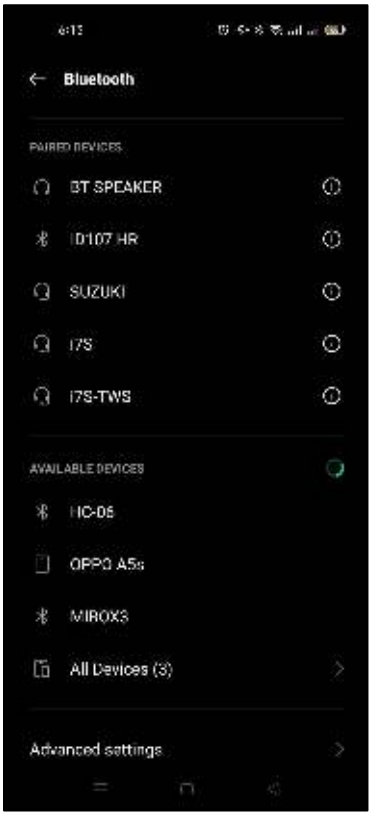
```
1.  int in1 = 5;
2.  int in2 = 6;
3.  int in3 = 9;
4.  int in4 = 10;
5.  int state; //
6.
7.  void setup() {
8.    Serial.begin(9600); // bluetooth communication
9.    pinMode(in3, OUTPUT);
10.   pinMode(in4, OUTPUT);
11.   pinMode(in1, OUTPUT);
12.   pinMode(in2, OUTPUT);
13. }
14.
15. void loop() {
16.   if (Serial.available() > 0) { //when bluetooth command is available
17.     state = Serial.read();
18.   }
19.   if (state == 'F') { // Forward
20.     Serial.println(state);
21.     digitalWrite(in4, 0);
22.     digitalWrite(in2, 0);
23.     digitalWrite(in3, 1);
24.     digitalWrite(in1, 1);
25.   }
26.   if (state == 'R') { // right
27.     Serial.println(state);
28.     digitalWrite(in4, 1);
29.     digitalWrite(in2, 0);
30.     digitalWrite(in3, 0);
31.     digitalWrite(in1, 1);
32.   }
33.   if (state == 'S') { // Stop
34.     Serial.println(state);
35.     digitalWrite(in4, 0);
36.     digitalWrite(in2, 0);
37.     digitalWrite(in3, 0);
38.     digitalWrite(in1, 0);
39.   }
40.   if (state == 'L') { // left
41.     Serial.println(state);
42.     digitalWrite(in4, 0);
43.     digitalWrite(in2, 1);
44.     digitalWrite(in1, 0);
45.     digitalWrite(in3, 1);
46.   }
47.
48.   if (state == 'B') { // Reverse
49.     Serial.println(state);
50.     digitalWrite(in3, 0);
51.     digitalWrite(in1, 0);
52.     digitalWrite(in4, 1);
53.     digitalWrite(in2, 1);
54.   }
55. }
56.
```

c. Press upload and let it finish uploading. When you encounter an error saying the com ports failed to sync, remove the connection of the Bluetooth module to the arduino and upload the code again. Once done reconnect the bluetooth module to the arduino.

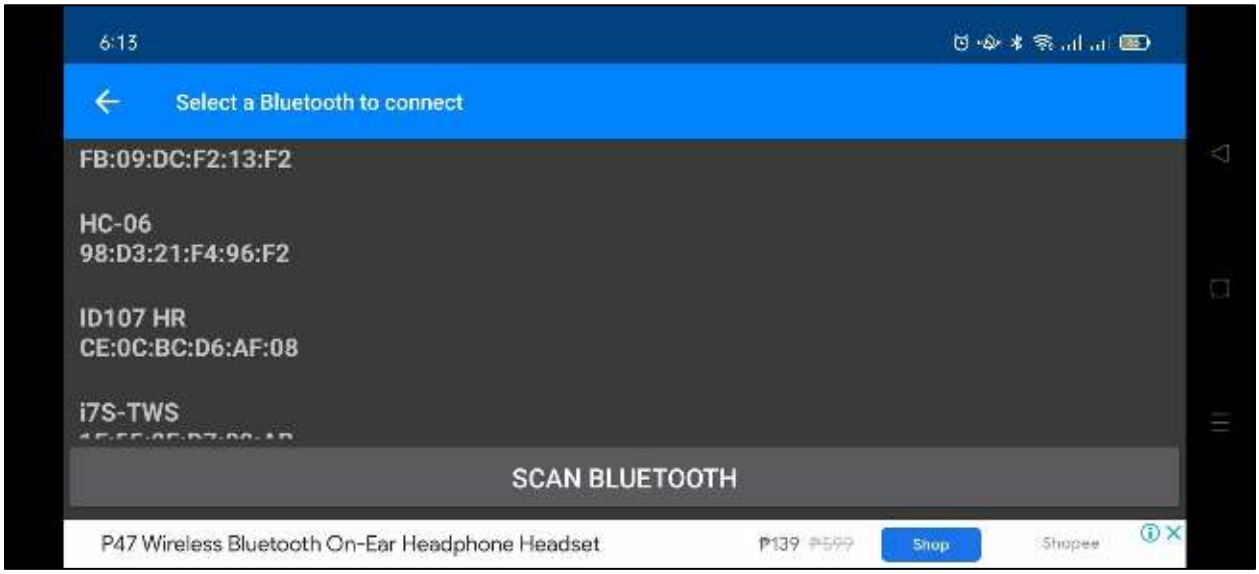
2. Go to Playstore and search for Arduino Bluetooth Car – Remote Control.
3. Install the referenced application (Arduino) // (iPhone instructions coming soon).

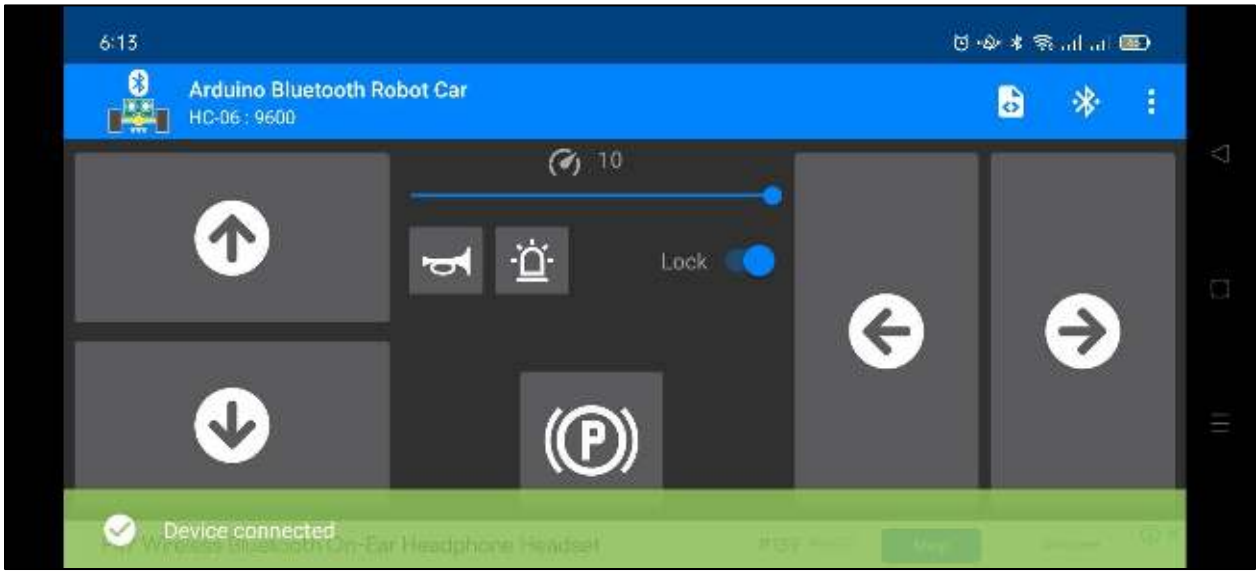



4. On your phone. Turn on your bluetooth and scan for bluetooth until you find the HC-06. (Make sure that the batteries are installed on the robot car already.) It will ask for Pin, input 1234 and then pair.




5. Once pairing is succesful. Open the Application. Once open press the bluetooth icon on the upper right part of the application. It will show the list of bluetooth devices available. Select the HC-06. Once connected you can now control your 4WD robot car.



## Programming / Code (Bluetooth Control):

1. Using the arduino IDE. Upload the code in the arduino.
    a. Download the NewPing Library in this link (https://bitbucket.org/teckel12/arduino-new-ping/downloads/) and extract the folder to C:\Program Files (x86)\Arduino\libraries
    b. Connect the Arduino to your pc using the USB cable.
    c. Open the Arduino IDE and copy the following code.

```
#include <NewPing.h>
#include <Servo.h>

#define TRIG_PIN A4
#define ECHO_PIN A5
#define MAX_DISTANCE 200
NewPing sonar(TRIG_PIN, ECHO_PIN, MAX_DISTANCE);
Servo myservo;

boolean goesForward = false;
int distance = 100;

int in1  = 5;
int in2  = 6;
int in3  = 10;
int in4  = 11;


void setup() {
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  myservo.attach(9);
  myservo.write(115);
  delay(2000);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
}

void loop() {
  int distanceR = 0;
  int distanceL =  0;
  delay(40);

  if (distance <= 20)
  {
    moveStop();
    delay(100);
    moveBackward();
    delay(300);
    moveStop();
```

```
    delay(200);
    distanceR = lookRight();
    delay(200);
    distanceL = lookLeft();
    delay(200);

    if (distanceR >= distanceL)
    {
      turnRight();
      moveStop();
    } else
    {
      turnLeft();
      moveStop();
    }
  } else
  {
    moveForward();
  }
  distance = readPing();
}

int lookRight()
{
  myservo.write(50);
  delay(500);
  int distance = readPing();
  delay(100);
  myservo.write(115);
  return distance;
}

int lookLeft()
{
  myservo.write(170);
  delay(500);
  int distance = readPing();
  delay(100);
  myservo.write(115);
  return distance;
  delay(100);
}

int readPing() {
  delay(70);
  int cm = sonar.ping_cm();
  if (cm == 0)
  {
    cm = 250;
  }
    return cm;
  }

  void moveStop() {
    digitalWrite(in4, 0);
    digitalWrite(in3, 0);
    digitalWrite(in2, 0);
    digitalWrite(in1, 0);
  }

  void moveForward() {
    digitalWrite(in4, 0);
    digitalWrite(in2, 0);
    digitalWrite(in3, 1);
    digitalWrite(in1, 1);
```

```
115.
116.  }
117.
118.  void moveBackward() {
119.    digitalWrite(in3, 0);
120.    digitalWrite(in1, 0);
121.    digitalWrite(in4, 1);
122.    digitalWrite(in2, 1);
123.
124.  }
125.
126.  void turnRight() {
127.    digitalWrite(in4, 1);
128.    digitalWrite(in2, 0);
129.    digitalWrite(in3, 0);
130.    digitalWrite(in1, 1);
131.    delay(300);
132.    moveForward();
133.
134.  }
135.
136.  void turnLeft() {
137.    digitalWrite(in4, 0);
138.    digitalWrite(in2, 1);
139.    digitalWrite(in1, 0);
140.    digitalWrite(in3, 1);
141.    delay(300);
142.    moveForward();
143.  }
144.
```

d. Press upload and let it finish uploading. When you encounter an error saying the com ports failed to sync, remove the connection of the Bluetooth module to the arduino and upload the code again. Once done reconnect the bluetooth module to the arduino.

## Programming / Code (Line Following):

1. Using the arduino IDE. Upload the code in the arduino.
    a. Connect the Arduino to your pc using the USB cable.
    b. Open the Arduino IDE and copy the following code.

```
int in1  = 5;
int in2  = 6;
int in3  = 10;
int in4  = 11;

int left = 13;
int right = 12;

int Left = 0;
int Right = 0;

void LEFT (void);
void RIGHT (void);
void STOP (void);

void setup()
{
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);

  pinMode(left, INPUT);
  pinMode(right, INPUT);

  digitalWrite(left, HIGH);
  digitalWrite(right, HIGH);

  analogWrite(in1, 255);
  analogWrite(in2, 0);
  analogWrite(in3, 255);
  analogWrite(in4, 0);

}

void loop()
{

  Left = digitalRead(left);
  Right = digitalRead(right);

  if (Left == 0 && Right == 1)
  {
    LEFT();
  }
  else if (Right == 0 && Left == 1)
    RIGHT();
}

void LEFT (void)
{
  analogWrite(in3, 0);
  analogWrite(in4, 30);
```

```
54.
55.   while (Left == 0)
56.   {
57.     Left = digitalRead(left);
58.     Right = digitalRead(right);
59.     if (Right == 0)
60.     {
61.       int lprev = Left;
62.       int rprev = Right;
63.       STOP();
64.       while (((lprev == Left) && (rprev == Right)) == 1)
65.       {
66.         Left = digitalRead(left);
67.         Right = digitalRead(right);
68.       }
69.     }
70.     analogWrite(in1, 255);
71.     analogWrite(in2, 0);
72.   }
73.   analogWrite(in3, 255);
74.   analogWrite(in4, 0);
75. }
76.
77. void RIGHT (void)
78. {
79.   analogWrite(in1, 0);
80.   analogWrite(in2, 30);
81.   while (Right == 0)
82.   {
83.     Left = digitalRead(left);
84.     Right = digitalRead(right);
85.     if (Left == 0)
86.     {
87.       int lprev = Left;
88.       int rprev = Right;
89.       STOP();
90.       while (((lprev == Left) && (rprev == Right)) == 1)
91.       {
92.         Left = digitalRead(left);
93.         Right = digitalRead(right);
94.       }
95.     }
96.     analogWrite(in3, 255);
97.     analogWrite(in4, 0);
98.   }
99.   analogWrite(in1, 255);
100.    analogWrite(in2, 0);
101.  }
102.  void STOP (void)
103.  {
104.    analogWrite(in1, 0);
105.    analogWrite(in2, 0);
106.    analogWrite(in3, 0);
107.    analogWrite(in4, 0);
108.
109.  }
110.
```

c. Press upload and let it finish uploading. When you encounter an error saying the com ports failed to sync, remove the connection of the Bluetooth module to the arduino and upload the code again. Once done reconnect the bluetooth module to the arduino.

## Programming / Code (IR Remote Control):

1. Using the arduino IDE. Upload the code in the arduino.
    a. Download the NewPing Library in this link (https://codeload.github.com/Arduino-IRremote/Arduino-IRremote/zip/refs/heads/master) and extract the folder to C:\Program Files (x86)\Arduino\libraries
    b. Connect the Arduino to your pc using the USB cable.
    c. Open the Arduino IDE and copy the following code.

```
#include <IRremote.h>

char command;
int receiver_pin = 4;   //Connect the output pin of IR receiver at pin 4
IRrecv irrecv(receiver_pin);
decode_results results;

// connect motor controller pins to Arduino digital pins
int in1 = 5;
int in2 = 6;
int in3 = 10;
int in4 = 11;


void setup()
{
  Serial.begin(9600);
  irrecv.enableIRIn();
  // set all the motor control pins to outputs
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);

}

void loop() {
  if (irrecv.decode(&results)) {
    irrecv.resume();
    if (results.value == 0xFF18E7) { // press button 2 forward robot control
      digitalWrite(in1, 1);
      digitalWrite(in2, 0);
      digitalWrite(in3, 1);
      digitalWrite(in4, 0);

    } else if (results.value == 0xFF10EF) { // press button 4 turn left robot control
      digitalWrite(in1, 1);
      digitalWrite(in2, 0);
      digitalWrite(in3, 0);
      digitalWrite(in4, 0);


    } else if (results.value == 0xFF30CF) { // press button 1 rotate left robot control
      digitalWrite(in1, 1);
```

```
45.       digitalWrite(in2, 0);
46.       digitalWrite(in3, 0);
47.       digitalWrite(in4, 1);
48.
49.
50.     } else if (results.value == 0xFF5AA5) { // press button 6 turn right robot control
51.       digitalWrite(in1, 0);
52.       digitalWrite(in2, 0);
53.       digitalWrite(in3, 1);
54.       digitalWrite(in4, 0);
55.
56.     } else if (results.value == 0xFF7A85) { // press button 3 rotate right robot control
57.       digitalWrite(in1, 0);
58.       digitalWrite(in2, 1);
59.       digitalWrite(in3, 1);
60.       digitalWrite(in4, 0);
61.
62.     } else if (results.value == 0xFF4AB5) { // press button 8 backward robot control
63.
64.       digitalWrite(in1, 0);
65.       digitalWrite(in2, 1);
66.       digitalWrite(in3, 0);
67.       digitalWrite(in4, 1);
68.
69.     } else if (results.value == 0xFF38C7) { // press button 5 stop robot control
70.       digitalWrite(in1, 0);
71.       digitalWrite(in2, 0);
72.       digitalWrite(in3, 0);
73.       digitalWrite(in4, 0);
74.     }
75.
76.   }
77.
78. }
79.
```

d. Press upload and let it finish uploading. When you encounter an error saying the com ports failed to sync, remove the connection of the Bluetooth module to the arduino and upload the code again. Once done reconnect the bluetooth module to the arduino.

---