

# Robot Car

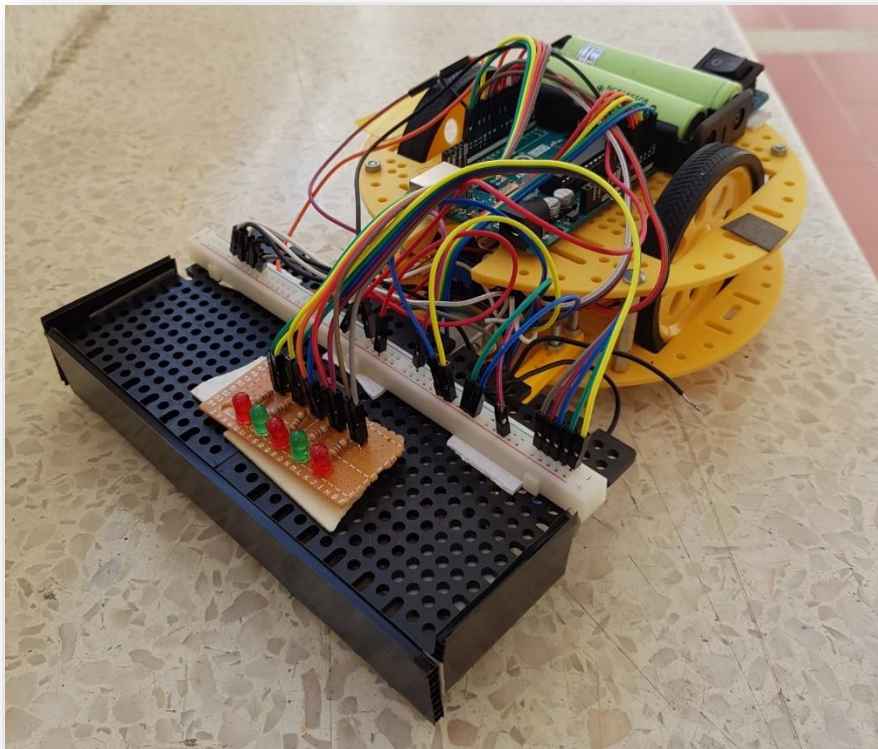
## แนวคิดการออกแบบ

มีการใช้เซนเซอร์ทั้งหมด 5 ตัว โดยวางให้อยู่ระนาบเดียวกันแล้วสูงจากระดับพื้นเท่าๆกันทุกตัว แล้วอ่านค่าจาก Serial Monitor ให้ได้ค่าเท่าๆกัน ถ้าค่าต่างกันมากต้องปรับตำแหน่งของเซนเซอร์ตัวที่ค่าห่างจากตัวอื่นให้อ่านค่าได้ใกล้เคียงกับเซนเซอร์ทุกตัว

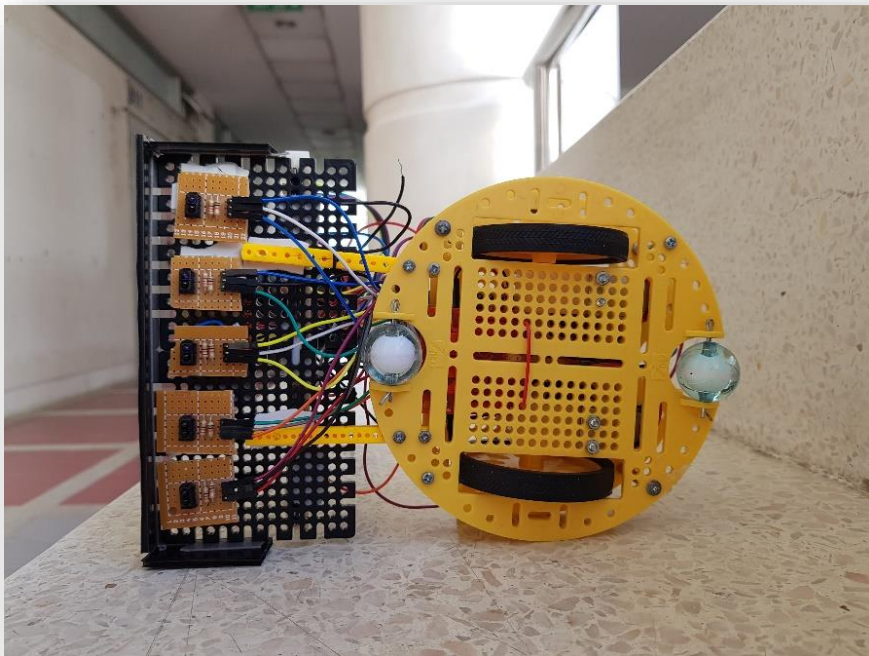
ทุกครั้งที่เราวางรถที่จุดเริ่มต้น จะมีการ calibrate ตัวรถอัตโนมัติ โดยการนำค่าเซ็นเซอร์ซ้ายสุดและขวาสุดมาเฉลี่ยกัน หลังจากนั้นมาเฉลี่ยกับเซ็นเซอร์ตัวกลางที่อยู่บนสีขาว เพื่อให้ได้ค่าที่ใช้แยกระหว่างพื้นขาวและดำ

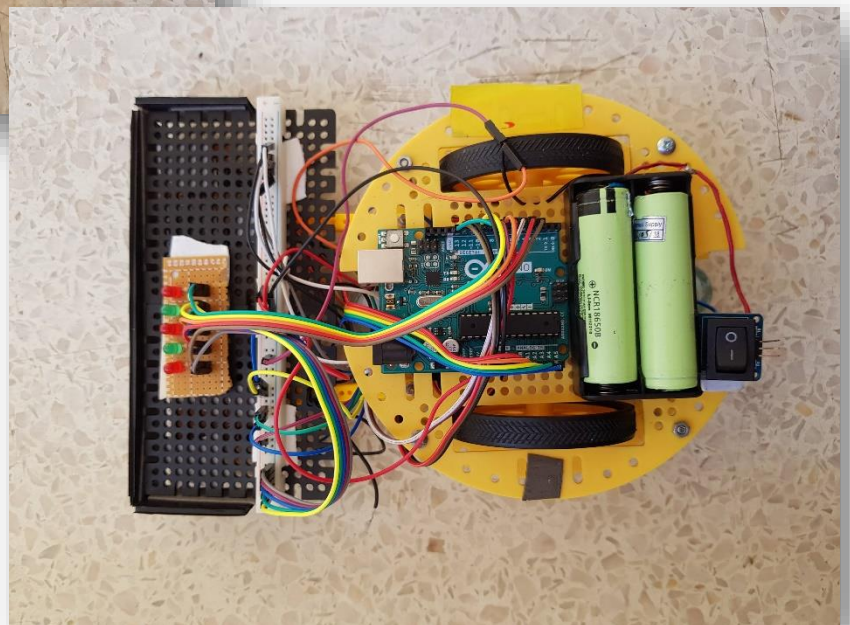
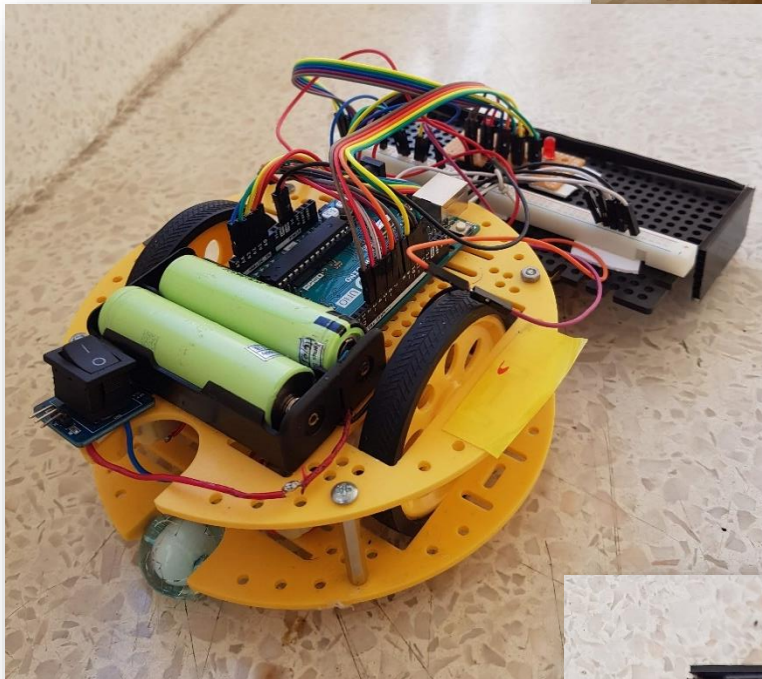
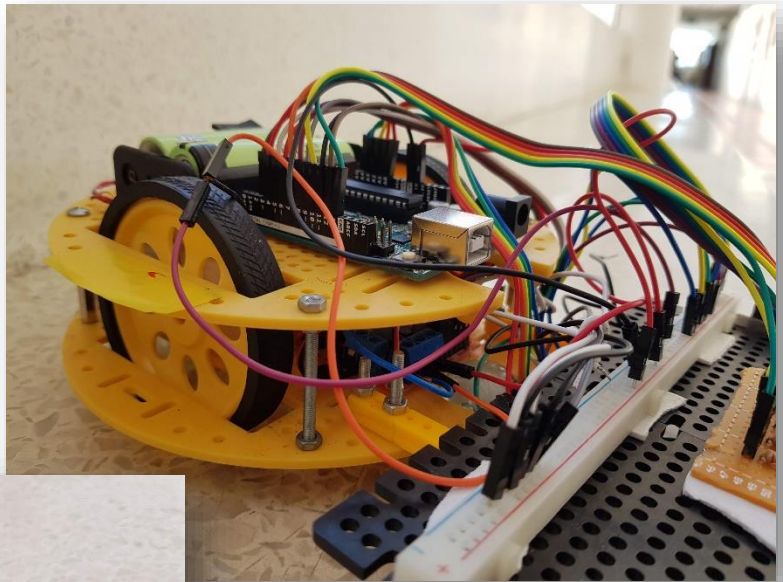
ในส่วนของการขับเคลื่อนรถ จะใช้ PID ในการขับเคลื่อน แต่เนื่องจากสนามมีความยาก ทำให้ PID ไม่สามารถนำรถไปถึงเป้าหมายได้ จึงต้องมีการเขียนโค้ดเพิ่มในส่วนที่จะมีการหลุดเส้นโดยการเช็คว่ามีรถหลุดเส้นไปแล้ว คืออ่านค่าได้สีดำทั้งหมด ก่อนหน้านั้นควรเลี้ยวไปทางไหน แล้วจึงใช้ดีเลย์เลี้ยวตามทิศทางนั้นไปหาเส้น แล้วจึงตามเส้นไปข้างหน้าต่อ ในส่วนของทางแยกจะมีการเขียนโค้ดรองรับในส่วนของการที่รถเจอสีขาว ทั้ง 5 เซนเซอร์ จะให้รถดีเลย์แล้ววิ่งเลยแยกไป

## รูปถ่าย พร้อมแสดงความพิเศษ



ในส่วนความสามารถพิเศษ เราได้มีการสร้างกรอบบังแสงล้อมรอบเซนเซอร์ทั้ง 5 ตัว ทำให้การอ่านค่ามีความนิ่งขึ้น ลดการรบกวนจากแสงภายนอกได้ และอัลกอริทึมที่ใช้ขับเคลื่อนไม่ใช่แบบจำเส้นทาง จึงสามารถนำไปวิ่งในสนามอื่นๆ ได้โดยไม่ต้องมีการแก้ไขใหม่







## โปรแกรม พร้อมคำอธิบายโดยย่อ

1.กำหนดความเร็วให้ตัวรถ แล้วใช้ booline ในการระบุสถานะของแต่ละตัวว่าอ่านได้คืออะไร

```
16 bool sensor[5]; int error = 0, lerror = 0;
17 int motorSpeed, aSpeed, bSpeed, range = 60;
18 int aBaseSpeed = 80, aMaxSpeed = aBaseSpeed + range, aMinSpeed = aBaseSpeed - range;
19 int bBaseSpeed = 80, bMaxSpeed = bBaseSpeed + range, bMinSpeed = bBaseSpeed - range;
20 int Kp = 20, calibrate, state = 0;
```

2.ฟังก์ชันการ Calibrate ค่าให้เซนเซอร์แต่ละตัวแยกสีได้อย่างแม่นยำ

```
22 int Calibrate()
23 {
24     int mid=analogRead(2), edge1=analogRead(0), edge2=analogRead(4);
25     return ((mid+((edge1+edge2)/2))/2);
26 }
```

3.ฟังก์ชัน PID ที่ใช้ในการขับรถไปข้างหน้าและเลี้ยวตามโค้งต่างๆ

```
87 int Forward(int error){
88     digitalWrite(in1, LOW); //Forward
89     digitalWrite(in2, HIGH);
90     digitalWrite(in3, HIGH); //Forward
91     digitalWrite(in4, LOW);
92     motorSpeed = Kp*error;
93     aSpeed = aBaseSpeed + motorSpeed;
94     bSpeed = bBaseSpeed - motorSpeed;
95
96     if(aSpeed > aMaxSpeed) aSpeed = aMaxSpeed;
97     if(bSpeed > bMaxSpeed) bSpeed = bMaxSpeed;
98
99     if(aSpeed < aMinSpeed) aSpeed = aMinSpeed;
100    if(bSpeed < bMinSpeed) bSpeed = bMinSpeed;
101    if(error == 1) aSpeed += 40;
102    if(error == -1) bSpeed += 40;
103    analogWrite(enA, aSpeed); // Speed Control
104    analogWrite(enB, bSpeed); // Speed Control
105    if(error == 4) {
106        analogWrite(enB, 0);
107        delay(400);
108    }
109    if(error == -4)
110    {
111        analogWrite(enA, 0);
112        delay(400);
113    }
114 }
```

4. คำนวณค่า Error จากเซนเซอร์ทั้ง 5 ตัว โดยถ้าเซนเซอร์ตัวไหนเจอสีขาวจะทำการสั่งให้ LED ของเซนเซอร์ตัวนั้นสว่างทันที โดยถ้าเซนเซอร์เจอสีขาวทุกตัวจะส่งค่า 5 กลับไป และจะส่งค่า -4 - 4 กลับไปเมื่อกรณีการเบนออกจากเส้นขาว และจะส่งค่า -5 เมื่อไม่เจอเส้นขาวเลย

```

138 int Error(){
139     for(int i = 0;i<5;i++){
140         if(analogRead(i)>calibrate) //If Detect White floor
141             sensor[i] = true;
142         else sensor[i] = false;
143         digitalWrite(led[i], sensor[i]);
144     }
145
146     if(sensor[0]&&sensor[1]&&sensor[2]&&sensor[3]&&sensor[4]) {return 5;}
147     else if(sensor[4]) {return -4;}
148     else if(sensor[0]) {return 4;}
149     else if(sensor[3]&&sensor[4]) {return -3;}
150     else if(sensor[0]&&sensor[1]) {return 3;}
151     else if(sensor[1]&&sensor[2]) {return 1;}
152     else if(sensor[2]&&sensor[3]) {return -1;}
153     else if(sensor[1]) {return 2;}
154     else if(sensor[3]) {return -2;}
155     else if(sensor[2]) {return 0;}
156     else {return -5;}
157 }

```

5.เมื่อรถอยู่ที่จุดเริ่มต้นแล้วจะมีการเปิดสวิตช์รถจะมีการเรียกใช้ฟังก์ชัน Calibrate ก่อนออกตัว และมีการโชว์ค่าของเซนเซอร์ทุกตัวออกทาง Serial Monitor

```

43 void loop()
44 {
45     if(millis()<1000){
46         calibrate = Calibrate();
47         // turn on motor A
48         digitalWrite(in1, LOW); //Stop
49         digitalWrite(in2, LOW);
50         // turn on motor B
51         digitalWrite(in3, LOW); //Stop
52         digitalWrite(in4, LOW);
53     }
54     else{
55         Serial.print(analogRead(0)); Serial.print(" ");
56         Serial.print(analogRead(1)); Serial.print(" ");
57         Serial.print(analogRead(2)); Serial.print(" ");
58         Serial.print(analogRead(3)); Serial.print(" ");
59         Serial.print(analogRead(4)); Serial.print(" Error : ");

```

6. ส่วนของการทำงานหลักที่อยู่ใน void loop() โดยมีการเรียกฟังก์ชัน Error เพื่อเช็คค่า error ณ ตำแหน่งปัจจุบัน เพื่อตัดสินใจในการเคลื่อนที่ไปข้างหน้า โดยถ้าได้ค่า error มีค่าตั้งแต่ -4 – 4 แสดงว่ายังอยู่บนเส้นสีขาว แล้วทุกครั้งที่ยังวิ่งบนเส้นจะมีการบันทึกค่า Last Error เพื่อนำมาใช้ตัดสินใจทิศทางการเลี้ยวเมื่อรถหลุดออกจากเส้น จะทำให้รถสามารถกลับเข้าหาเส้นได้เสมอ ถ้าค่า error เท่ากับ 5 คือเซนเซอร์เจอสีขาวทุกตัวจะให้รถลองเคลื่อนตัวไปด้านหน้าเพื่อเช็คว่ามีทางข้างหน้าหรือไม่

```

61 error = Error();
62 if(error>=-4 && error<=4)
63 {
64     lerror = error;
65     Forward(error);
66     state = 0;
67 }
68 else if(error == -5) {
69     if(lerror < 0){
70         Forward(-4);
71     }
72     if(lerror > 0){
73         Forward(4);
74     }
75     if(lerror == 0){
76         Forward(4);
77     }
78 }
79 else if(error == 5){
80     Forward(0);
81     delay(300);
82 }
83 Serial.println(error);
84 }
85 }

```