

```
1 #include <Servo.h>
2
3 // البينات الأصلية
4 #define ENA 5
5 #define ENB 6
6 #define IN1 7
7 #define IN2 8
8 #define IN3 9
9 #define IN4 10
10
11 #define FLAME_LEFT A0
12 #define FLAME_FRONT A1
13 #define FLAME_RIGHT A2
14 #define FLAME_PUMP A3 // الحساس الرابع لتشغيل المضخة فقط
15
16 #define PUMP 2
17 #define SERVO_ARM 11 // السيرفو اللي ينزل ويرفع الذراع
18 #define SERVO_BASE 12 // السيرفو اللي يحرك القاعدة يمين ويسار
19
20 Servo servoArm;
21 Servo servoBase;
22
23 // العتبات
24 #define FIRE_NEAR_THRESHOLD 80
25 #define FIRE_DETECT_THRESHOLD 400
26 #define FIRE_PUMP_THRESHOLD 300 // عتبة تشغيل المضخة من الحساس الرابع
27
28 void setup() {
29     Serial.begin(9600);
30
31     // المحركات
32     pinMode(ENA, OUTPUT);
33     pinMode(ENB, OUTPUT);
34     pinMode(IN1, OUTPUT);
35     pinMode(IN2, OUTPUT);
36     pinMode(IN3, OUTPUT);
37     pinMode(IN4, OUTPUT);
38
39     // الحساسات
40     pinMode(FLAME_LEFT, INPUT);
41     pinMode(FLAME_FRONT, INPUT);
42     pinMode(FLAME_RIGHT, INPUT);
43     pinMode(FLAME_PUMP, INPUT);
44
45     // مؤشرات ومضخة
46     pinMode(PUMP, OUTPUT);
47     digitalWrite(PUMP, LOW);
48     pinMode(LED_RED, OUTPUT);
49     pinMode(LED_BLUE, OUTPUT);
```

```
50
51 // السيرفو هات
52 servoArm.attach(SERVO_ARM);
53 servoBase.attach(SERVO_BASE);
54 servoArm.write(90);
55 servoBase.write(90);
56
57 Serial.println(" Fire Fighting Robot Ready ");
58 }
59
60 void loop() {
61   int leftVal = analogRead(FLAME_LEFT);
62   int frontVal = analogRead(FLAME_FRONT);
63   int rightVal = analogRead(FLAME_RIGHT);
64   int pumpVal = analogRead(FLAME_PUMP);
65
66
67   if (leftVal < FIRE_DETECT_THRESHOLD || frontVal < FIRE_DETECT_THRESHOLD || rightVal < FIRE_DETECT_THRESHOLD) {
68     moveTowardFlame(leftVal, frontVal, rightVal);
69     digitalWrite(LED_BLUE, HIGH);
70     digitalWrite(LED_RED, LOW);
71   }
72
73
74   if (frontVal < FIRE_NEAR_THRESHOLD) {
75     stopMoving();
76     digitalWrite(LED_BLUE, LOW);
77     digitalWrite(LED_RED, HIGH);
78
79     // ينزل الذراع
80     servoArm.write(130);
81     delay(300);
82
83     // القاعدة تتحرك يمين ويسار
84     for (int pos = 60; pos <= 120; pos += 5) {
85       servoBase.write(pos);
86       delay(60);
87     }
88     for (int pos = 120; pos >= 60; pos -= 5) {
89       servoBase.write(pos);
90       delay(60);
91     }
92
93     // يرجع الذراع للوضع الطبيعي
94     servoArm.write(90);
95     servoBase.write(90);
96   }
97 }
```

```
98 // فقط مسؤول عن تشغيل وإيقاف المضخة A3 الحساس الرابع
99 if (pumpVal < FIRE_PUMP_THRESHOLD) {
100     digitalWrite(PUMP, HIGH); // شغل المضخة
101 }
102 else {
103     digitalWrite(PUMP, LOW); // طفي المضخة
104 }
105
106 delay(200);
107 }
108
109 // دوال الحركة
110 void moveForward() {
111     digitalWrite(IN1, LOW);
112     digitalWrite(IN2, HIGH);
113     digitalWrite(IN3, HIGH);
114     digitalWrite(IN4, LOW);
115     analogWrite(ENA, 150);
116     analogWrite(ENB, 150);
117 }
118
119 void turnRight() {
120     digitalWrite(IN1, LOW);
121     digitalWrite(IN2, HIGH);
122     digitalWrite(IN3, LOW);
123     digitalWrite(IN4, LOW);
124     analogWrite(ENA, 150);
125     analogWrite(ENB, 150);
126 }
127
128 void turnLeft() {
129     digitalWrite(IN1, LOW);
130     digitalWrite(IN2, LOW);
131     digitalWrite(IN3, HIGH);
132     digitalWrite(IN4, LOW);
133     analogWrite(ENA, 150);
134     analogWrite(ENB, 150);
135 }
136
137 void stopMoving() {
138     digitalWrite(IN1, LOW);
139     digitalWrite(IN2, LOW);
140     digitalWrite(IN3, LOW);
141     digitalWrite(IN4, LOW);
142 }
143
144 void moveTowardFlame(int leftVal, int frontVal, int rightVal) {
145     if (frontVal <= leftVal && frontVal <= rightVal) {
146         moveForward();
```

```
147     }  
148     else if (leftVal < rightVal) {  
149         turnLeft();  
150     }  
151     else {  
152         turnRight();  
153     }  
154 }  
155
```