

Autonomous Fire Extingusher

DIY Project

By

Himesh Kundal 23EC10030
Chaitanya Patil 23MT10038

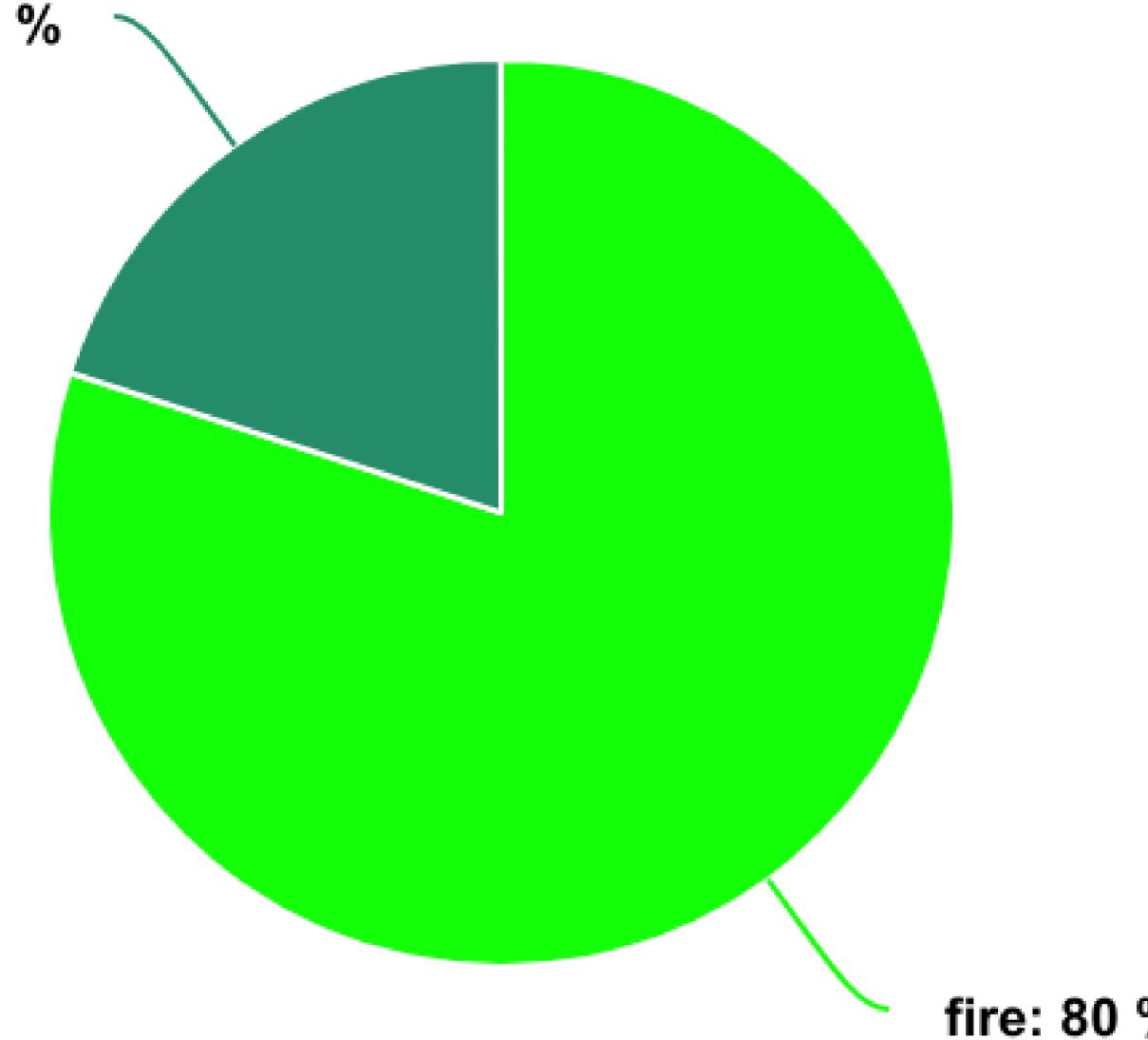
application of the project :

In India, the annual mortality rate due to fire-related incidents amounts to approximately **25,000** individuals

The majority of fire-related fatalities are attributed to the **absence** of autonomous fire prevention systems

property damage

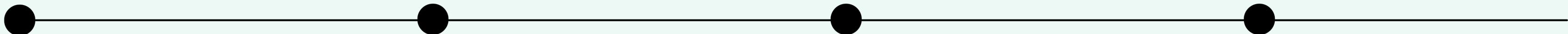
other(water damage,natural disasters,vandalism): 20 %



fire

other(water damage,natural disasters,vandalism)

DIY project



circuit drawing

Basic Circuital Layout and connections.

connecting components

All the sensors and components

writing code

The code written and how it works

Problems Faced

All the problems we faced and how we tackled each of them

working principal

1

**Upon detection of
a flame by the
flame sensors,
both the buzzers
and red lights are
activated
simultaneously**

2

**The servo motor
initiates rotation
in the direction
where the flame
has been
detected**

3

**the pump
mechanism is
engaged to
facilitate the
suppression of
the fire**

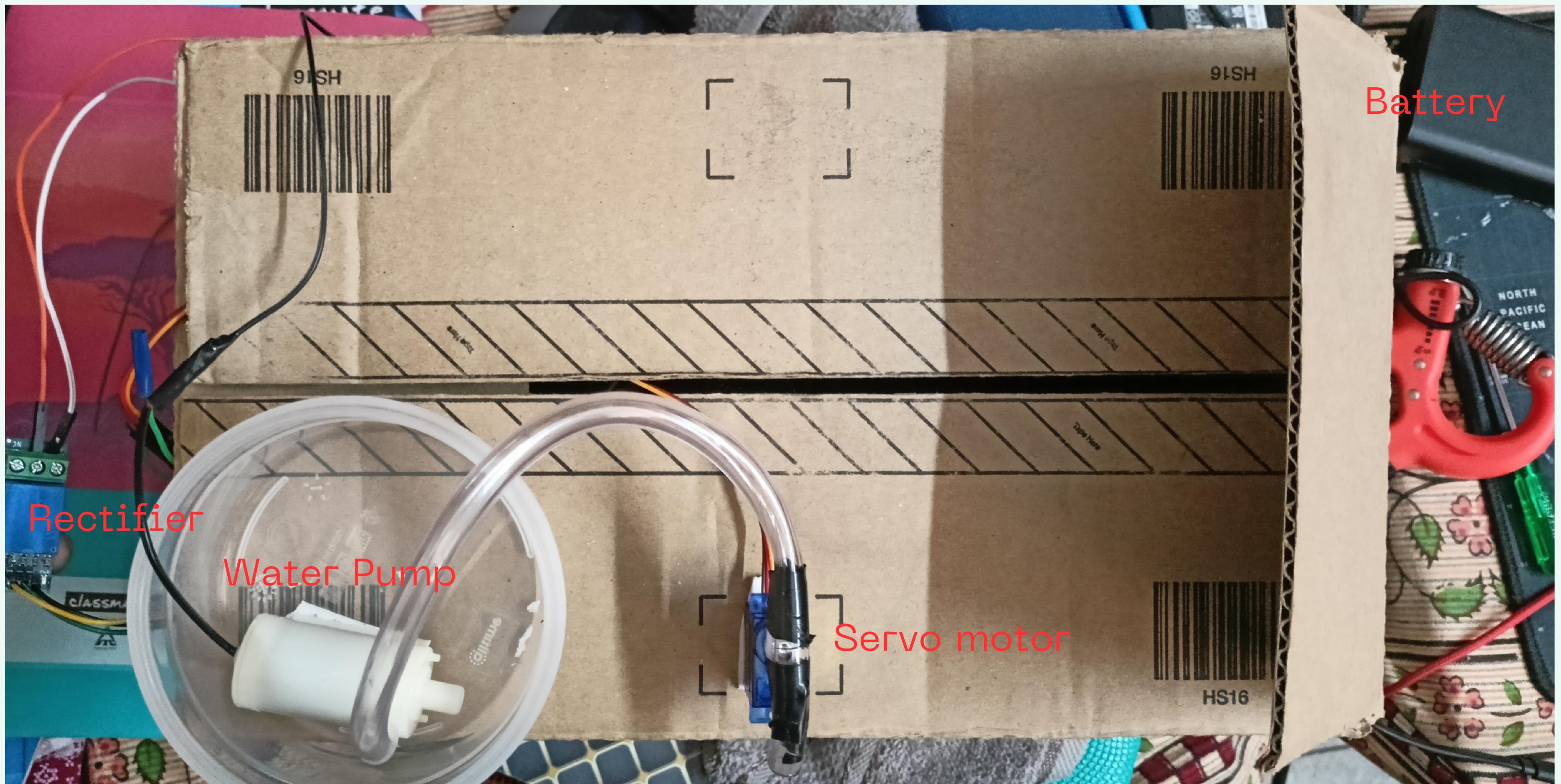
4

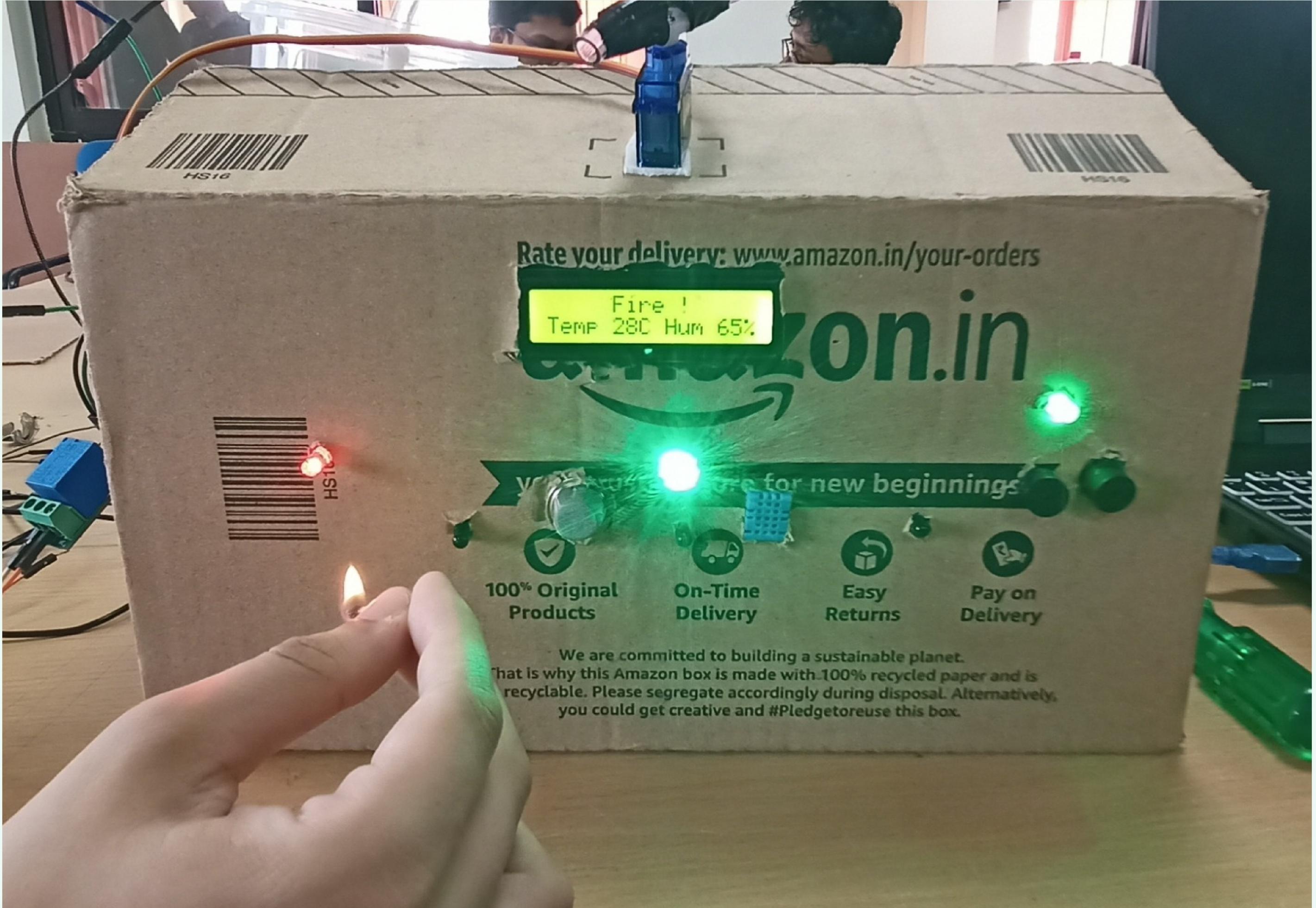
following the successful extinguishing of the fire, the buzzer and light deactivate, signaling the conclusion of the fire extinguishment.

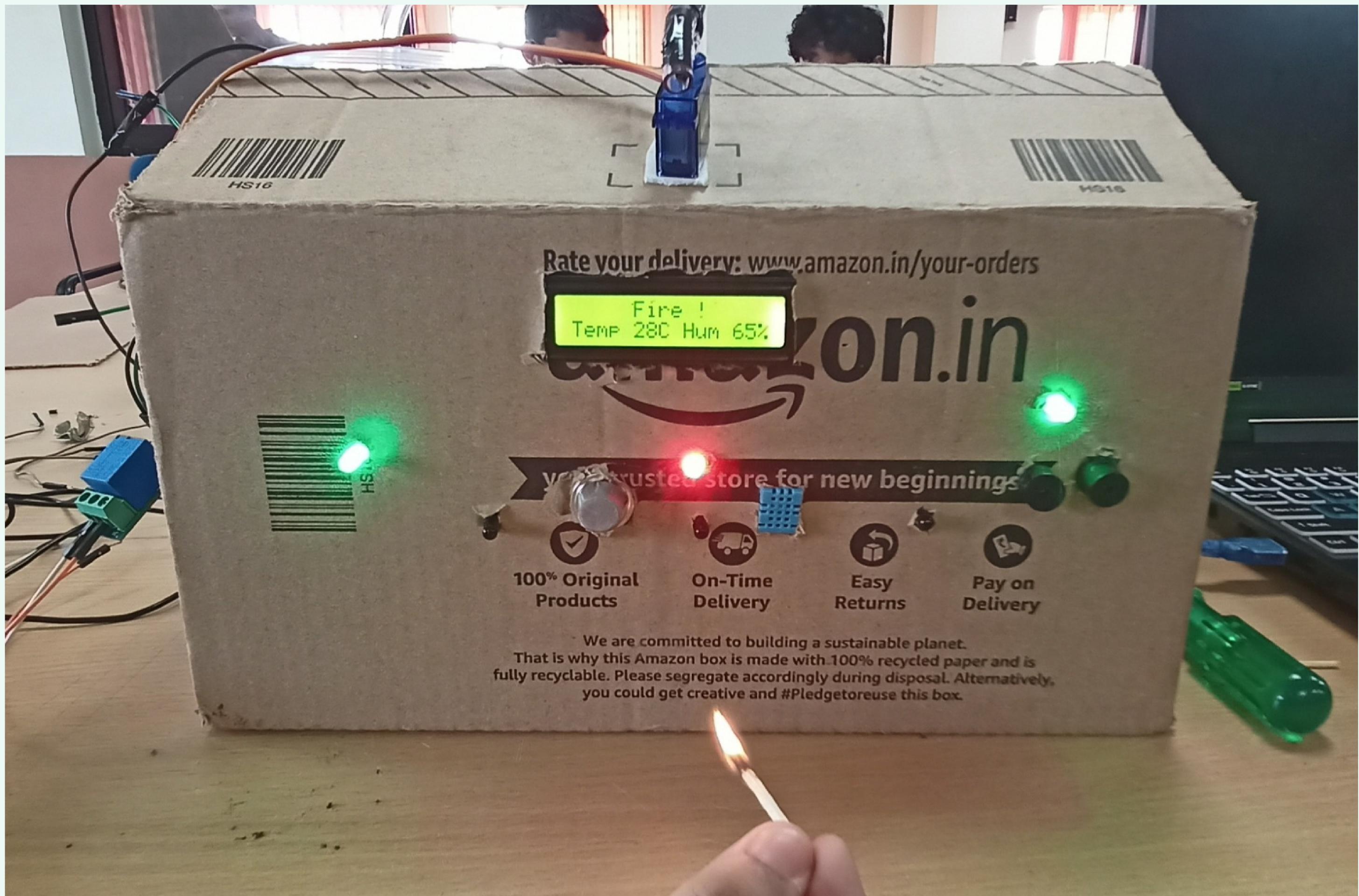
5

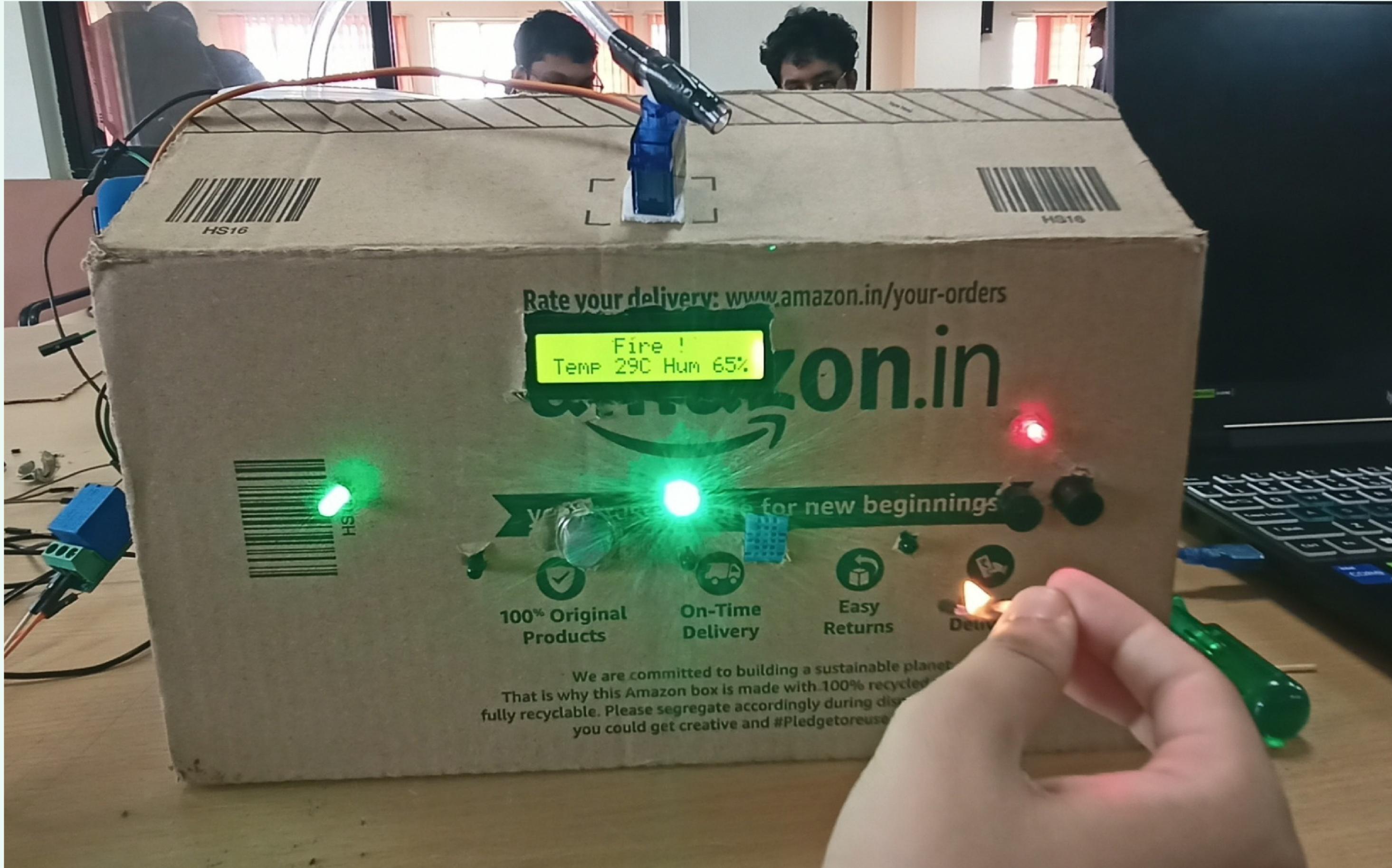
Throughout this sequence, the coordinated activation and deactivation of various components ensure a prompt and efficient response to fire incident

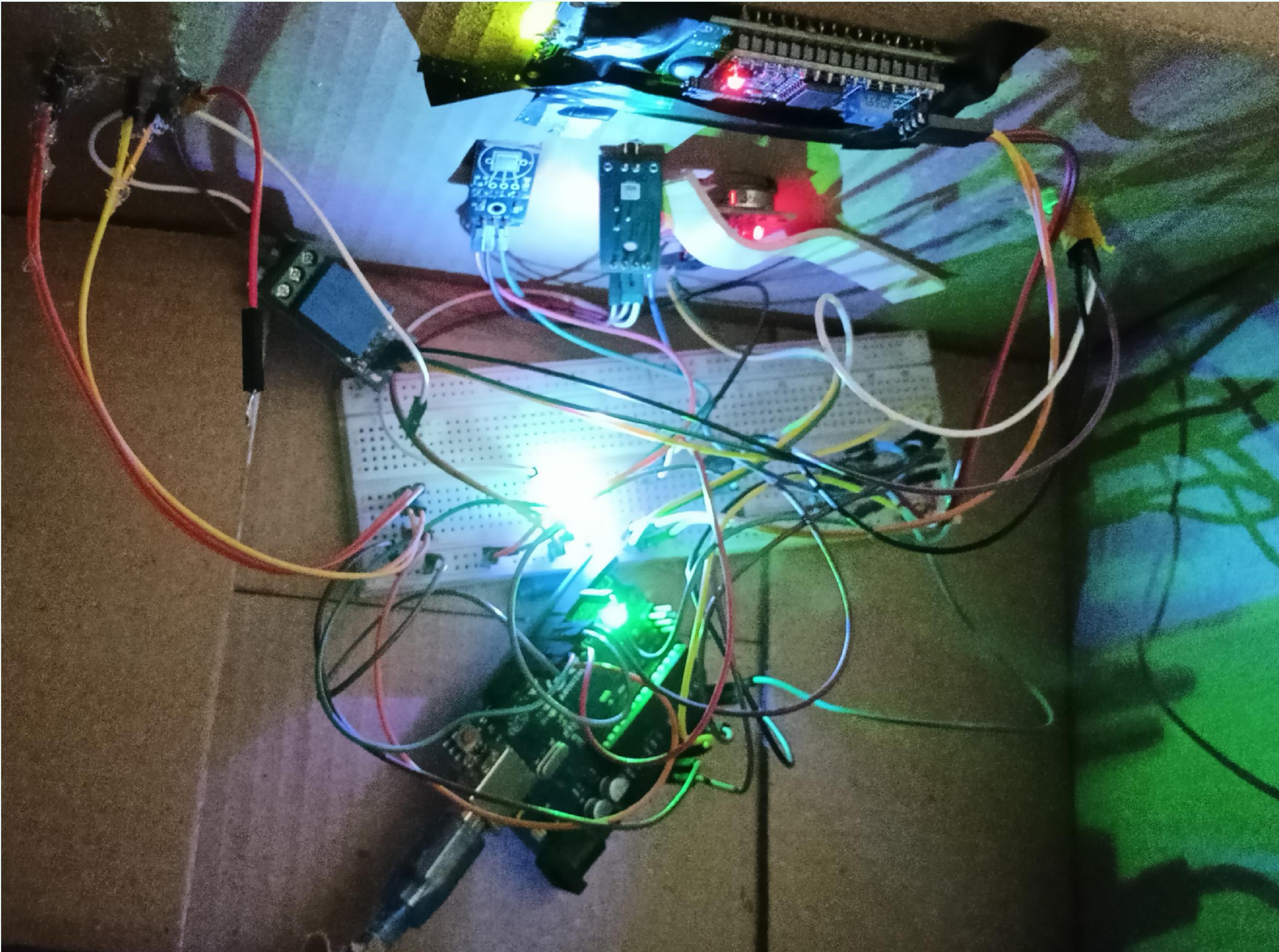












components used :

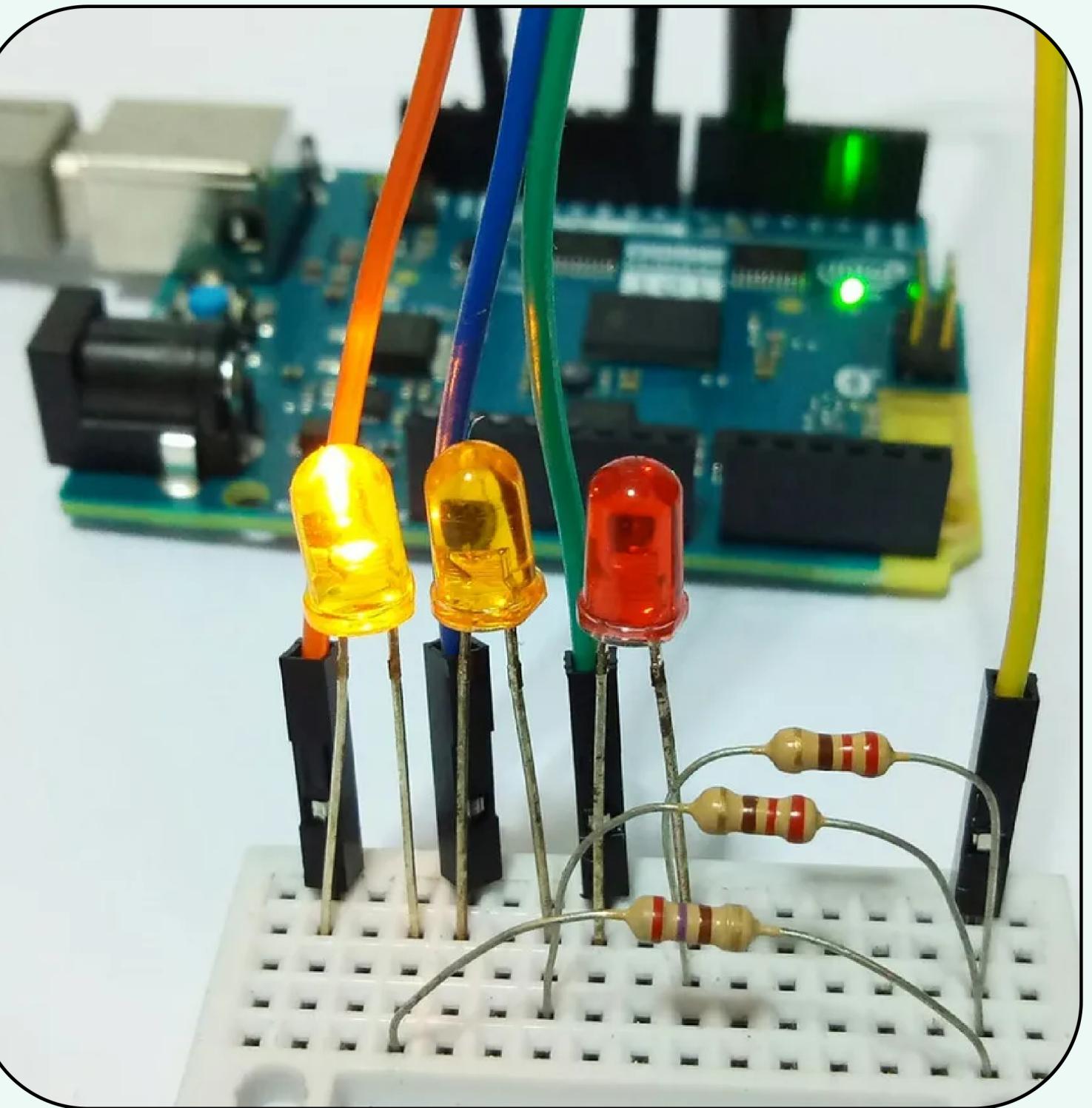
1. buzzer
2. MQ-135 gas sensor
3. DHT11
4. Breadboard
5. IR flame sensor module
6. mini water pump
7. LCD display
8. arduino uno
9. LEDs(RGB)
10. Relays
11. servo motor

The buzzer will sound upon detection of a fire by the system.

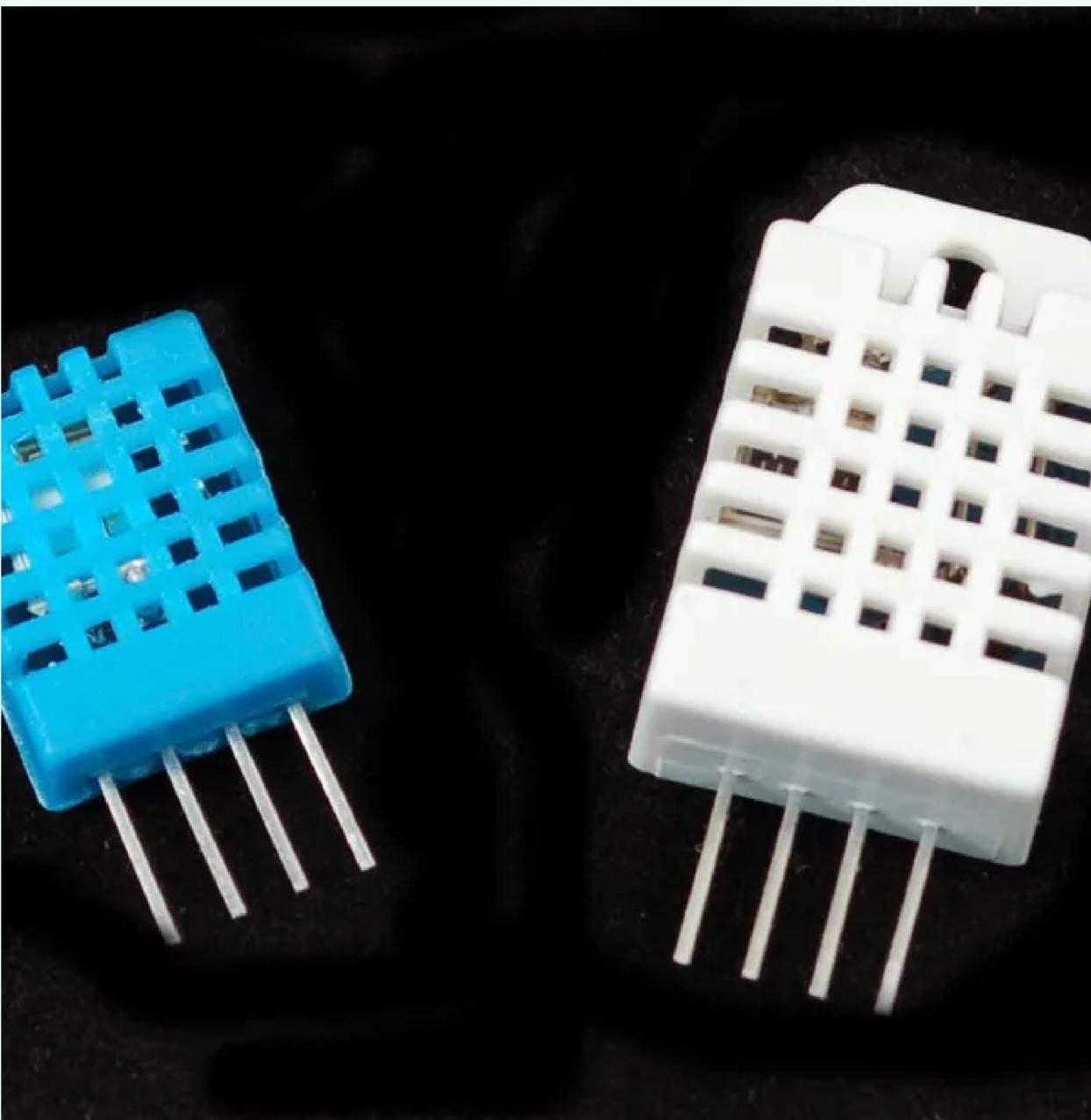
Add more text



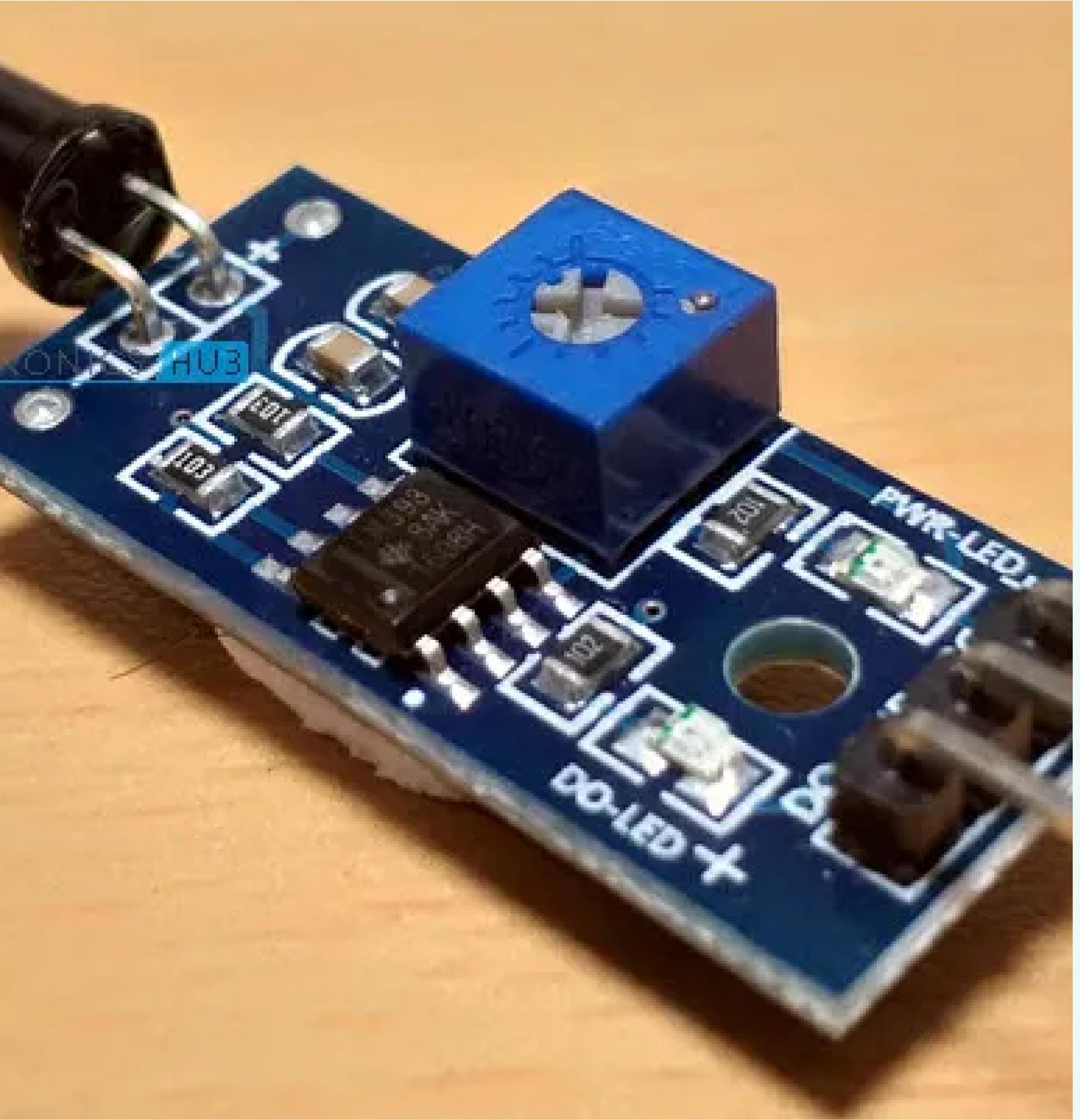
The green light transitions to red upon the detection of fire by the IR flame module.



The DHT11 sensor
measures the
temperature and
humidity level of the
environment.



The infrared
(IR) flame
module detects
the presence of
fire.



The water pump
is activated
upon the
detection of a
fire.



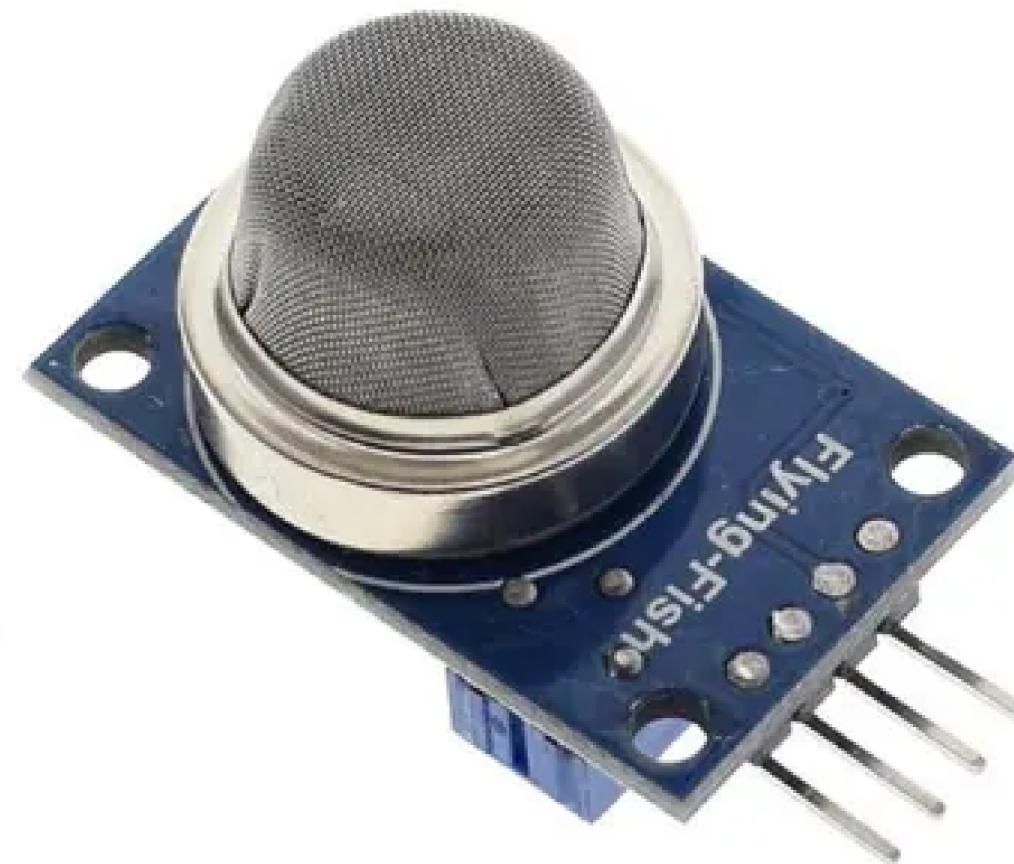
The LCD screen displays the temperature and humidity levels detected by the DHT11 sensor.



Add more text

The MQ135
sensor checks
the Air Quality
of the
surrounding.

Add more text



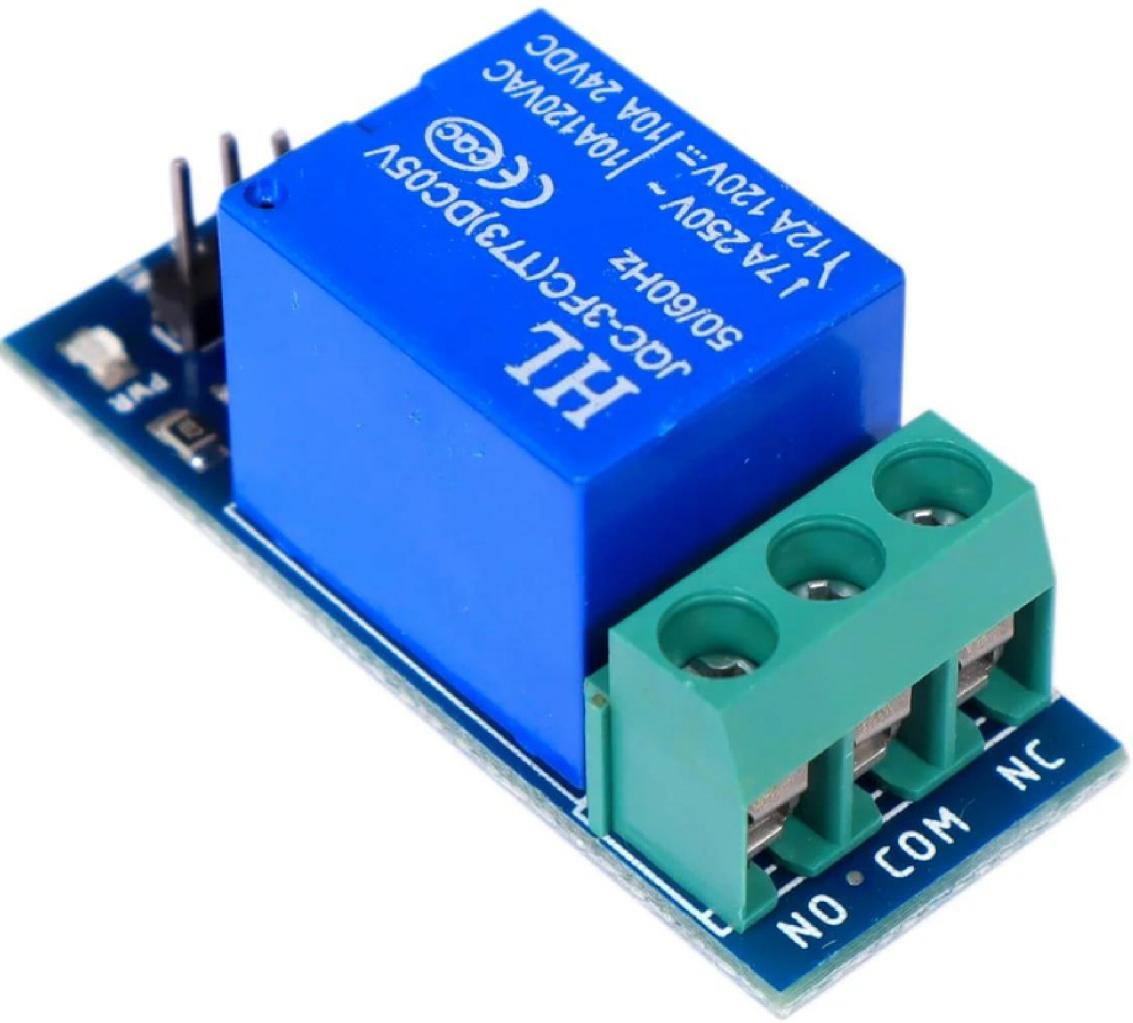
The Arduino
UNO is the Base
components
which controls
all the
components in
the circuit

Add more text



The Relay Helps
us to use
Arduino as a
switch for
higher power
components
like the motor.

Add more text



servo motor rotates in the direction corresponding to the detection of fire as indicated by flame sensors





```
1
2 #include "LiquidCrystal_I2C/LiquidCrystal_I2C.h"
3 #include "LiquidCrystal_I2C/LiquidCrystal_I2C.cpp"
4 #include "MQ135/MQ135.h"
5 #include "MQ135/MQ135.cpp"
6 #include "DHT11/src/DHT11.h"
7 #include "DHT11/src/DHT11.cpp"
8 #include <Servo.h>
9 #include "Pitches/pitches.h"
10
11 Servo myservo;
12
13 int Buzz = 13;
14 int IR_Left = 12;
15 DHT11 dht11(11);
16 int FlameLeftGreen = 10;
17 int Servo = 9;
18 int FlameLeftRed = 3;
19 int IR_Right = 8;
20 int FlameRightRed = 7;
21 int FlameRightGreen = 6;
22 int Pump = 5;
23 int IR_Mid = 4;
24
25 #define FlameMidGreen A0
26 #define FlameMidRed A1
27
28 #define PIN_MQ135 A0
29 MQ135 mq135_sensor(PIN_MQ135);
```

Included Libraries

Initialised the pins

```
31 int FlameLeft=0;
32 int FlameRight=0;
33 int FlameMid=0;
34
35 LiquidCrystal_I2C lcd(0x27, 16, 2);
36
37 void setup() {
38     Serial.begin(9600);
39
40     lcd.init();
41     lcd.backlight();
42
43     pinMode(Buzz,OUTPUT);
44     pinMode(Pump,OUTPUT);
45     pinMode(IR_Left,INPUT);
46     pinMode(IR_Right,INPUT);
47     pinMode(IR_Mid,INPUT);
48     pinMode(FlameRightRed,OUTPUT);
49     pinMode(FlameRightGreen,OUTPUT);
50     pinMode(FlameMidGreen,OUTPUT);
51     pinMode(FlameMidRed,OUTPUT);
52     pinMode(FlameLeftRed,OUTPUT);
53     pinMode(FlameLeftGreen,OUTPUT);
54     myservo.attach(9);
55     dht11.setDelay(500);
56     PEACE();
57
58 }
```

Setup of LED screen

Set the modes of the pin

```
59 void loop() {
60
61     int temperature = 0;
62     int humidity = 0;
63     int result = dht11.readTemperatureHumidity(temperature, humidity);
64
65     float rzero = mq135_sensor.getRZero();                                Temp And Humid
66     float correctedRZero = mq135_sensor.getCorrectedRZero(temperature, humidity);
67     float resistance = mq135_sensor.getResistance();                      Air Quality
68     float ppm = mq135_sensor.getPPM();
69     float correctedPPM = mq135_sensor.getCorrectedPPM(temperature, humidity);
70
71
72     int FlameLeft=(digitalRead(IR_Left));
73     FlameLeft=!FlameLeft; //Flame is inversed.
74
75     int FlameRight=(digitalRead(IR_Right));                                Checking the state of IR Sensors
76     FlameRight=!FlameRight; //Flame is inversed.
77
78     int FlameMid=(digitalRead(IR_Mid));
79     FlameMid=!FlameMid; //Flame is inversed.
80
81     if (result == 0) {
82
83         Serial.print("\tTemperature: ");
84         Serial.print(temperature);
85         Serial.print(" °C\tHumidity: ");
86         Serial.print(humidity);
87         Serial.print(" %");
88
89         Serial.print("\tFlameLeft = ");
90         Serial.println(FlameLeft);
```

```
95     Serial.print("\tCorrected RZero: ");
96     Serial.print(correctedRZero);
97     Serial.print("\tResistance: ");
98     Serial.print(resistance);
99     Serial.print(" PPM: ");
100    Serial.print(ppm);
101    Serial.print("\tCorrected PPM: ");
102    Serial.print(correctedPPM);
103    Serial.println("ppm");
104    Serial.println(".");
105
106    lcd.setCursor(0,1);
107    lcd.print("Temp ");
108    lcd.print(temperature);
109    lcd.print("C Hum ");
110    lcd.print(humidity);
111    lcd.print("%");
112 }
113
114
115 if(FlameLeft || FlameRight|| FlameMid){
116     if(FlameLeft) AAGLeft();
117     if(FlameMid) AAGMid();
118     if(FlameRight) AAGRRight();
119
120     // direction(FlameLeft, FlameMid, FlameRight);
121 }
```

Printing Results on the Serial Console

Respective Function for each sensor

```
126 void direction(int l, int m, int r){  
127     if (l || m || r) myservo.write(90);  
128     else if (l || m) myservo.write(68);  
129     else if (r || m) myservo.write(113);  
130     else if (l) myservo.write(45);  
131     else if (m) myservo.write(90);  
132     else if (r) myservo.write(135);  
133  
134 }  
135  
136 void AAGLeft(){  
137     myservo.write(45);  
138     digitalWrite(FlameLeftGreen,0);  
139     digitalWrite(FlameLeftRed,1);  
140     digitalWrite(Pump,1);  
141     digitalWrite(Buzz,1);  
142     lcd.setCursor(3,0);  
143     lcd.print(" Fire ! ");  
144 }  
145  
146 void AAGMid(){  
147     myservo.write(90);  
148     analogWrite(FlameMidGreen,0);  
149     analogWrite(FlameMidRed,1020);  
150     digitalWrite(Pump,1);  
151     digitalWrite(Buzz,1);  
152 }
```

Direction of Servo motors

All the functions for each
direction for specific
fire fighting response
response

```
156 void HADright()\n157     myservo.write(135);\n158     digitalWrite(FlameRightGreen,0);\n159     digitalWrite(FlameRightRed,1);\n160     digitalWrite(Pump,1);\n161     digitalWrite(Buzz,1);\n162     lcd.setCursor(3,0);\n163     lcd.print("  Fire !      ");\n164 }\n165\n166 void PEACE(){\n167     myservo.write(90);\n168     digitalWrite(FlameLeftGreen,1);\n169     digitalWrite(FlameLeftRed,0);\n170     digitalWrite(FlameRightGreen,1);\n171     digitalWrite(FlameRightRed,0);\n172     analogWrite(FlameMidGreen,1020);\n173     analogWrite(FlameMidRed,0);\n174     digitalWrite(Pump,0);\n175     digitalWrite(Buzz,0);\n176     lcd.setCursor(3,0);\n177     lcd.print("Peace Hai !");\n178 }\n179\n180
```

Normal State

challenges faced :

Faulty Arduino Analog pins

tried to increase Voltage by adding a battery in series and Ended up frying arduino.

after troubleshooting for hours , found out Relay was not working.

IDE was not detecting libraries so had to manually add the header files.

It was a whole mess trying to get the Screen Working and found out screen was faulty.

Had to add more sensors and servo for a better detection and response system.

Thank you!