

- **Parts used :**

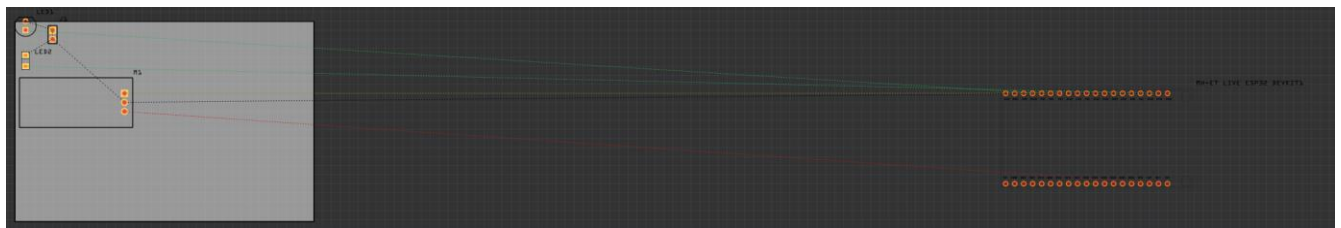
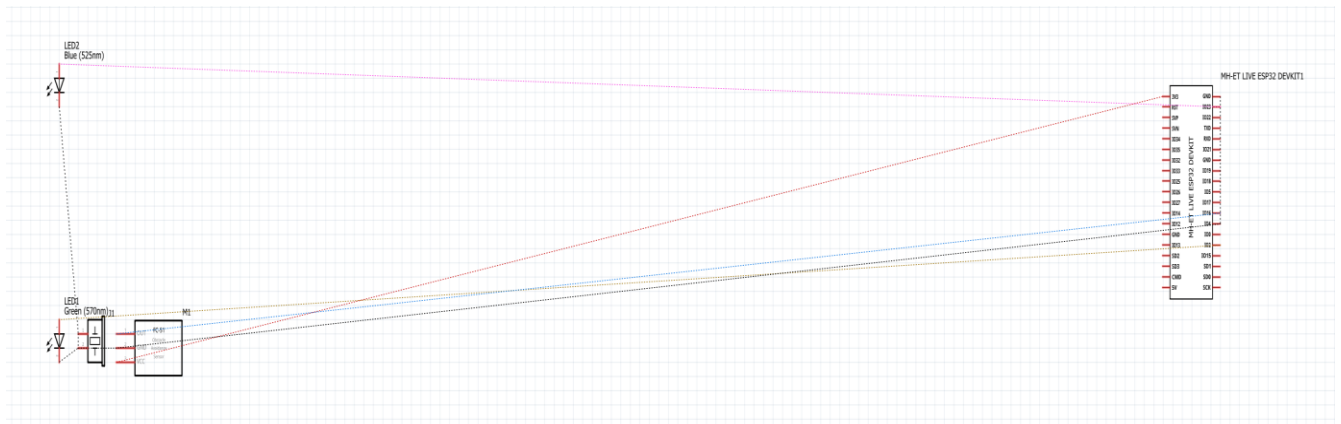
- (2x) Breadboard
- (1x) ESP32 Dev Module
- (1x) Buzzer
- (1x) FC-51
- (1x) Green LED
- (1x) Blue LED
- Jumper Wires

- **Software used :**

- Arduino – IDE
- Arduino Cloud Website

Project Description :

The IoT Door Monitoring System is a sophisticated and versatile solution designed to provide advanced security and real-time door status notifications in various settings. Leveraging the power of Internet of Things (IoT) technology, this project offers comprehensive monitoring and remote-control capabilities, ensuring the safety and convenience of users.



Algorithm and Program source code:

A) Algorithm :

1. Import Necessary Libraries:

- Include the "thingProperties.h" library to define properties for the Arduino IoT Cloud.

2. Define Pin Constants:

- Declare constants for the IR sensor input pin (IRPin), the buzzer output pin (buzzerPin), and an additional pin (I).

3. Setup Function:

- Initialize the serial communication at a baud rate of 9600 for debugging purposes.
- Configure the pin modes for the IR sensor, buzzer, and an additional pin.
- Initialize the Arduino IoT Cloud properties and establish a connection to the cloud service.
- Set the debug message level and print debug information.

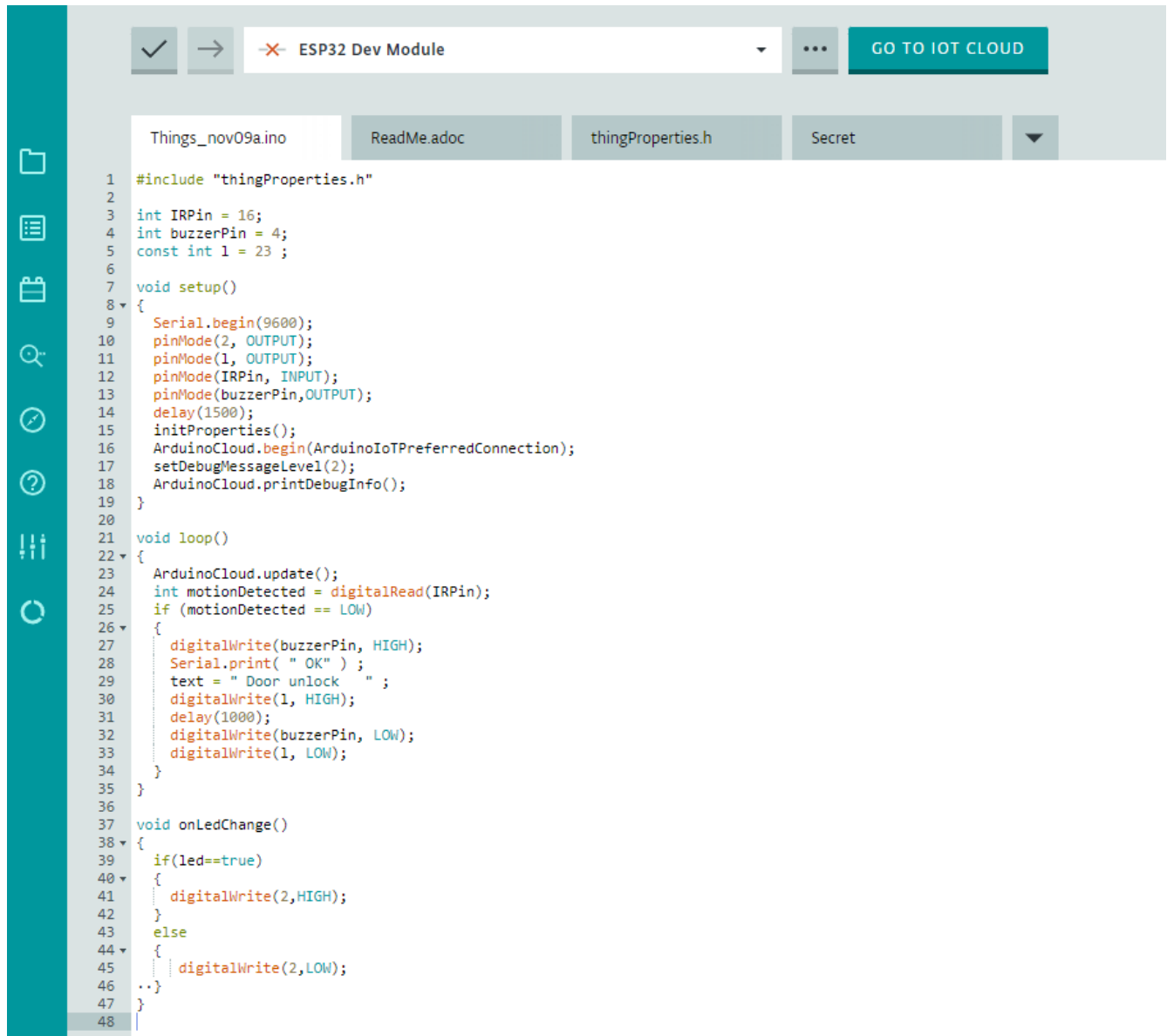
4. Loop Function:

- **Continuously update the Arduino IoT Cloud to handle property changes and cloud communication.**
- **Read the IR sensor (motion detector) input to detect motion.**
- **If no motion is detected (motionDetected is LOW), execute the following steps:**
 - **Turn on the buzzer to produce an audible alert.**
 - **Print "OK" to the serial monitor for debugging.**
 - **Set the text variable to "Door unlock."**
 - **Turn on an additional pin (I) to indicate door unlock status.**
 - **Delay for 1 second.**
 - **Turn off the buzzer and the additional pin (I).**

5. onLedChange Function:

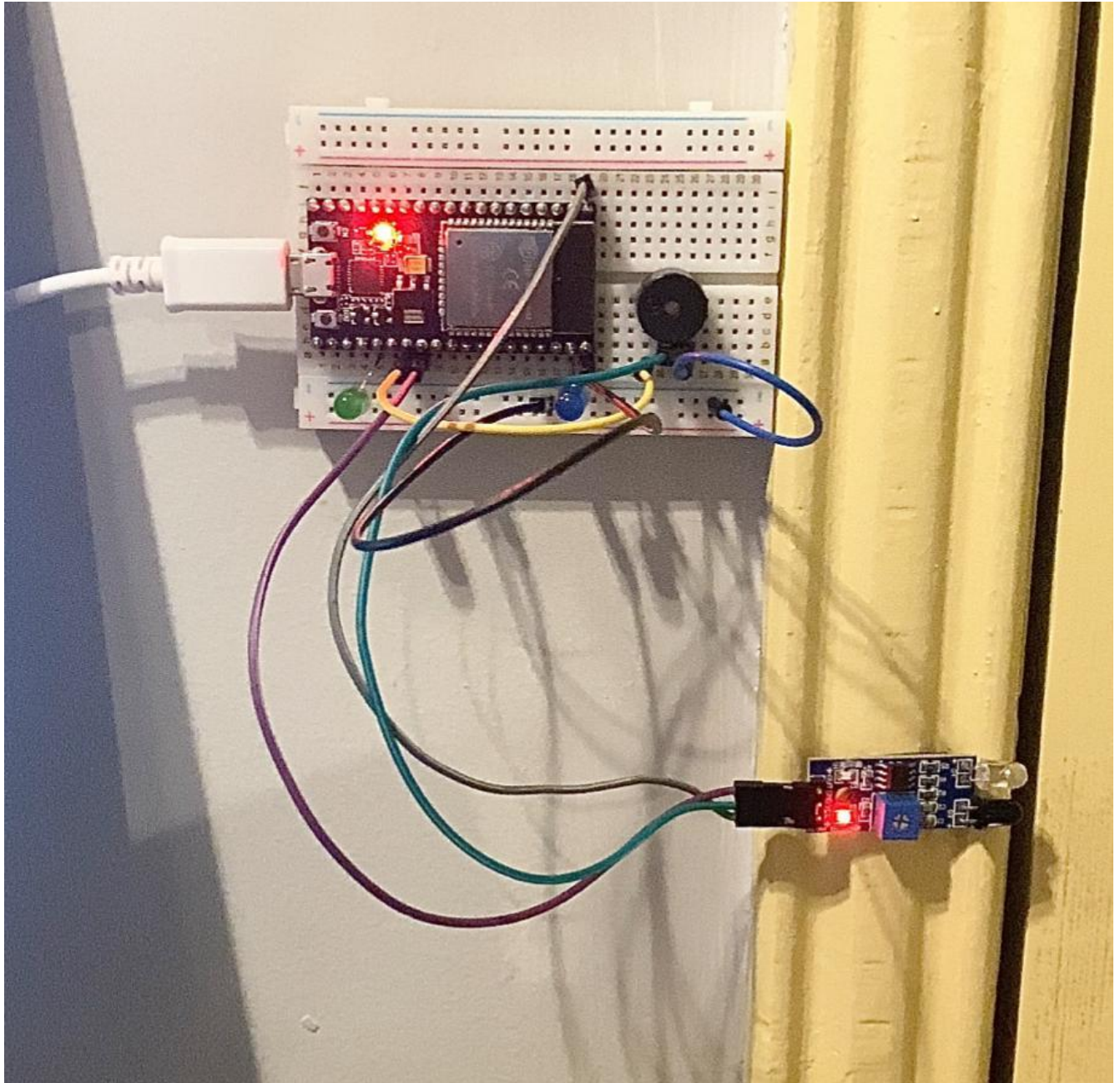
- **This function is a callback that is automatically triggered when the state of the "led" property changes in the Arduino IoT Cloud.**
- **If the "led" property is set to true, it sets digital pin 2 to HIGH (turning on an LED).**
- **If the "led" property is set to false, it sets digital pin 2 to LOW (turning off the LED).**

B) Program Source Code :



```
1 #include "thingProperties.h"
2
3 int IRPin = 16;
4 int buzzerPin = 4;
5 const int l = 23 ;
6
7 void setup()
8 {
9   Serial.begin(9600);
10  pinMode(2, OUTPUT);
11  pinMode(1, OUTPUT);
12  pinMode(IRPin, INPUT);
13  pinMode(buzzerPin, OUTPUT);
14  delay(1500);
15  initProperties();
16  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
17  setDebugMessageLevel(2);
18  ArduinoCloud.printDebugInfo();
19 }
20
21 void loop()
22 {
23   ArduinoCloud.update();
24   int motionDetected = digitalRead(IRPin);
25   if (motionDetected == LOW)
26   {
27     digitalWrite(buzzerPin, HIGH);
28     Serial.print( " OK" );
29     text = " Door unlock  " ;
30     digitalWrite(1, HIGH);
31     delay(1000);
32     digitalWrite(buzzerPin, LOW);
33     digitalWrite(1, LOW);
34   }
35 }
36
37 void onLedChange()
38 {
39   if(led==true)
40   {
41     digitalWrite(2,HIGH);
42   }
43   else
44   {
45     digitalWrite(2,LOW);
46   }
47 }
48
```

Implementation :



fritzing

