

TEAM ID : PNT2022TMID12767

PROJECT TITLE : Industry-Specific Intelligent Fire Management System

Name:VIJAYALAKSHMI K

RollNo:717819L255

CODE:

```
#include<WiFi.h>
#include<PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "sg5c1o"
#define DEVICE_TYPE "assignment4"
#define DEVICE_ID "4"
#define TOKEN "90785634"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/event2/fmt/json";
char topic[] = "iot-2/cmd/home/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
const int trigpin = 5;
const int echopin = 18;
String command;
String data = "";
long duration;
float dist;
void setup()
{
  Serial.begin(115200);
  pinMode(led, OUTPUT);
```

```
pinMode(trigpin,OUTPUT);
pinMode(echopin,INPUT);
wifiConnect();
mqttConnect();
}
void loop(){
bool isNearby = dist <100;
digitalWrite(led, isNearby);
publishData();
delay(500);
if(!client.loop()){
mqttConnect();
}
}
void wifiConnect(){
Serial.print("Connecting to ");Serial.print("Wifi");
WiFi.begin("Wokwi-GUEST","",6);
while(WiFi.status()!= WL_CONNECTED){
delay(500);
Serial.print(".");
}
Serial.print("WiFi connected, IP address: ");Serial.println(WiFi.localIP());
}
void mqttConnect(){
if(!client.connected()){
Serial.print("Reconnecting MQTT client to ");Serial.println(server);
while(!client.connect(clientId, authMethod, token)){
Serial.print(".");
delay(500);
}
initManagedDevice();
Serial.println();
}
}
```

```

void initManagedDevice(){
  if(client.subscribe(topic)){
    // Serial.println(client.subscribe(topic));
    Serial.println("IBM subscribe to cmd OK");
  }else{
    Serial.println("subscribe to cmd FAILED");
  }
}

void publishData()
{
  digitalWrite(trigpin,LOW);
  digitalWrite(trigpin,HIGH);
  delayMicroseconds(10);
  digitalWrite(trigpin,LOW);
  duration=pulseIn(echopin,HIGH);
  dist=duration*speed/2;
  if(dist<100){
    String payload = "{\"Alert!! Alert!! Distance\"":";
    payload += dist;
    payload += "}";

    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if(client.publish(publishTopic,(char*) payload.c_str())){
      Serial.println("Publish OK");
    }

  }

  if(dist>100){
    String payload = "{\"Distance\"":";
    payload += dist;
    payload += "}";

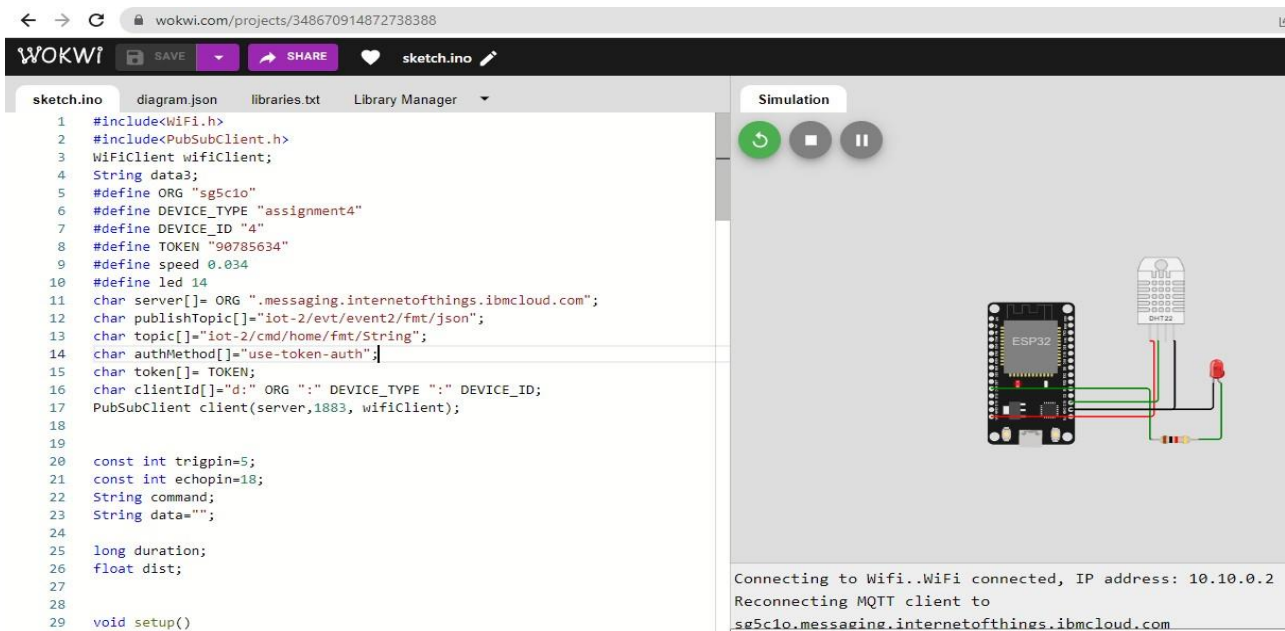
```

```

Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if(client.publish(publishTopic,(char*) payload.c_str())){
Serial.println("Publish OK");
}else{
Serial.println("Publish FAILED");
}
}
}
}

```

CIRCUIT:



The screenshot shows the Wokwi web IDE interface. On the left, the 'sketch.ino' file is open, displaying the following code:

```

1 #include<Wifi.h>
2 #include<PubSubClient.h>
3 WiFiClient wifiClient;
4 String data3;
5 #define ORG "sg5cio"
6 #define DEVICE_TYPE "assignment4"
7 #define DEVICE_ID "4"
8 #define TOKEN "90785634"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/event2/fmt/json";
13 char topic[] = "iot-2/cmd/home/fmt/String";
14 char authMethod[] = "use-token-auth";
15 char token[] = TOKEN;
16 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server,1883, wifiClient);
18
19
20 const int trigpin=5;
21 const int echopin=18;
22 String command;
23 String data="";
24
25 long duration;
26 float dist;
27
28
29 void setup()

```

On the right, the 'Simulation' tab is active, showing a circuit diagram of an ESP32 microcontroller board connected to a DHT22 temperature and humidity sensor and a red LED. The sensor is connected to the ESP32's VCC, GND, and data pins. The LED is connected to the ESP32's GND and a digital pin.

Below the circuit diagram, the simulation log shows the following output:

```

Connecting to Wifi..WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to
sg5cio.messaging.internetofthings.ibmcloud.com

```