

TEAM ID : PNT2022TMID12767

PROJECT TITLE : Industry-Specific Intelligent Fire Management System

NAME : Ahalya Preethi R

ROLL NO : 717819L201

CODE :

```
#include<WiFi.h>
```

```
include<PubSubClient.h>
```

```
WiFiClient wifiClient;
```

```
Stringdata3;
```

```
#defineORG"sg5c1o"
```

```
#define DEVICE_TYPE "assignment4"
```

```
#defineDEVICE_ID"4"
```

```
#defineTOKEN"90785634"
```

```
#definespeed 0.034
```

```
#defineled14
```

```
char server[] = ORG ".messaging.internet of things.ibmcloud.com";
```

```
charpublishTopic[]="iot-2/evt/event2/fmt/json";
```

```
char topic[]="iot-2/cmd/home/fmt/String";
```

```
char auth Method[]="use-token-auth";
```

```
char token[]=TOKEN;
```

```
charclientId[]="d:"ORG":DEVICE_TYPE ":"DEVICE_ID;
```

```
PubSubClient client(server,1883, wifi Client);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 in it Managed Device(){if(client.subscribe(topic)){
  //Serial.println(client.subscribe(topic));
  Serial.println("IBM subscribe to cmd OK");
}else{

```

```
Serial.println("subscribe to cmd FAILED");
}
}
void publish Data()
{
digital Write(trigpin,LOW);
digital Write(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(publish Topic,(char*)payload.c_str())){
Serial.println("PublishOK");
}
}
if(dist>100){
String payload="{\"Distance\":";
payload+=dist;
```

```

payload+="}";

Serial.print("\n");

Serial.print("Sending payload:");

Serial.println(payload);

if(client.publish(publish Topic,(char*)payload.c_str())){

Serial.println("Publish OK");

}else{

Serial.println("Publish FAILED");

}

}

}

}

```

CIRCUIT:

The screenshot displays the Wokwi online IDE interface. On the left, the 'sketch.ino' file is open, showing a C++ program for an ESP32 microcontroller. The code includes libraries for WiFi and MQTT, defines constants for organization, device type, device ID, token, speed, and LED pin, and sets up MQTT client and server details. It also defines pins for a trigpin and echopin, and declares variables for command, data, duration, and distance. The setup function is partially visible.

On the right, the 'Simulation' tab is active, showing a visual representation of the ESP32 board connected to a DHT22 temperature and humidity sensor. Below the simulation, a console log shows the following output:

```

Connecting to Wifi..WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to
se5cio.messaging.internetofthings.ibmcloud.com

```