

ASSIGNMENT – IV

DOMAIN: IOT

PROJECT TITLE: Smart Farmer – IOT
Enabled Smart Farming Application

Team ID	PNT2022TMID48496
Student Name	S. Hemalatha
Student Roll Number	920219106010
Maximum Marks	2 Marks

Assignment 4:

Write code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events. Upload document with wokwi share link and images of IBM cloud

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
```

```
#define ORG "4imkkt"
#define DEVICE_TYPE "ass4"
#define DEVICE_ID "ass4ID"
#define TOKEN "123456789"
#define speed 0.034
#define led 14
char server[] = ORG
".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-
2/evt/HEMALATHA/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);
void publishData();
```

```
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 = "{\"Normal 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");
        }
    }
}
```

}

}

Service Details - IBM CloudIBM Watson IoT PlatformFwd: ass4 - dhanushyao47@gmail.comesp32-dht22.ino copy - Wokwi

wokwi.com/projects/348464889765823058

WOKWI

SAVE

SHARE

esp32-dht22.ino copy

Docs

SIGN UP

esp32-dht22.ino

diagram.json

libraries.txt

Library Manager

Simulation

00:09.607101%

```
1 #include <Arduino.h>
2 #include <PubSubClient.h>
3 WiFiClient wificlient;
4 String data3;
5 #define ORG "73yp09"
6 #define DEVICE_TYPE "ass4"
7 #define DEVICE_ID "ass41D"
8 #define TOKEN "123456789"
9 #define speed 0.834
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/HIMALATHA/fmt/json";
13 char topic[] = "iot-2/cmd/home/fmt/String";
14 char authMethod[] = "use-token-auth";
15 char token[] = TOKEN;
16 char clientId[] = "41:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, wificlient);
18 void publishData();
19
20
21 const int trigpin=5;
22 const int echopin=18;
23 String command;
24 String data="";
25
26 long duration;
27 float dist;
28
29
30
31 void setup()
32 {
33   Serial.begin(115200);
34   pinMode(led, OUTPUT);
35   pinMode(trigpin, OUTPUT);
```

Publish OK

Sending payload: {"Normal Distance":89.95}

Publish OK

Sending payload: {"Normal Distance":89.95}

Publish OK

Browse Devices

All DevicesDiagnose

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Q Search by Device IDDevice Simulator

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
> ass41D	Connected	ass4	Device	Nov 16, 2022 12:07 PM	

Items per page 501-1 of 1 item1 of 1 page