

Assignment -4

Assignment Date	31 October 2022
Student Name	Arunkumar S.
Student Roll Number	611219106004
Maximum Marks	2 Marks

Question-1:

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100 cm send "alert" to IBM cloud and display in device recent events.

WOWKI LINK:

<https://wokwi.com/projects/347013247322292820>

Solution:

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "2c42b4"
#define DEVICE_TYPE "Assignment-4"
#define DEVICE_ID "ArunID"
#define TOKEN "vZ2OKUj2D!pYfKlzhw"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/manimd/fmt/json";
char topic[] = "iot-2/cmd/led/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 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");
        }

    }

}
```

Output:

The screenshot shows the Wokwi online 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 "2c42b4"
6 #define DEVICE_TYPE "Assignment-4"
7 #define DEVICE_ID "ArunID"
8 #define TOKEN "vZ20KUj2D1pYfK1zhw"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/manimd/fmt/json";
13 char topic[] = "iot-2/cmd/led/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
21 const int trigpin=5;
22 const int echopin=18;
23 String command;
24 String data="";
25
26 long duration;
27 float dist;
```

On the right, the 'Simulation' window shows a virtual circuit with an ESP32 microcontroller and an HC-SR04 ultrasonic sensor. The sensor is connected to the ESP32's pins. The simulation status bar at the bottom indicates 'Publish OK' and 'Sending payload: {"Distance":138.96}'.

The screenshot shows the IBM Watson IoT Platform dashboard. The top navigation bar includes 'Service Details - IBM Cloud' and 'IBM Watson IoT Platform'. The main content area displays the details for a device named 'ArunID'.

Device Details:

- Device ID:** ArunID
- Status:** Connected
- Device Type:** Assignment-4
- Class ID:** Device
- Date Added:** Oct 31, 2022 3:13 PM

Recent Events:

Event	Value	Format	Last Received
manimd	{"Distance":138.96}	json	a few seconds ago
manimd	{"Distance":138.98}	json	a few seconds ago
manimd	{"Distance":138.96}	json	a few seconds ago
manimd	{"Distance":138.96}	json	a few seconds ago
manimd	{"Distance":138.96}	json	a few seconds ago