

Assignment-4

Student Name	ASHWIN KUMAR S
Roll Number	211719106501
Team ID	PNT2022TMID26547
Date	30-October 2022
Project Name	Project -Smart farmer-IOT enabled smart Farming Application

Question:

Write code and connections in wokwi for ultrasonic sensor.

Whenever distance is less than 100cm send "alert" to IBM cloud and display in device recent events.

Upload document with wokwi share link and images of IBM cloud.

CODE:

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

#define ORG "17lsro"
#define DEVICE_TYPE "MyDeviceType"
#define DEVICE_ID "12345"
#define TOKEN "GkatKdiUS?UVHKvnAD"

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char pubTopic1[] = "iot-2/evt/ASHWIN KUMAR S/fmt/json";
char pubTopic2[] = "iot-2/evt/status2/fmt/json";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
const int DHT_PIN = 15;
WiFiClient wifiClient;
PubSubClient client(server, 1883, NULL, wifiClient);

#define ECHO_PIN 12
#define TRIG_PIN 13

float readDistanceCM() ;
void setup() {
    Serial.begin(115200);
    pinMode(15, OUTPUT);
    pinMode(TRIG_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);

    Serial.println();
    Serial.print("Connecting to ");

    WiFi.begin("Wokwi-GUEST", "", 6);
```

```

while (WiFi.status() != WL_CONNECTED) {
    delay(50);
    Serial.print(".");
}
Serial.println("");

Serial.print("WiFi connected, IP address: ");
Serial.println(WiFi.localIP());

if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!client.connect(clientId, authMethod, token)) {
        Serial.print(".");
        delay(500);
    }
    Serial.println("Bluemix connected");
}
}

float readDistanceCM()
{
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    int duration = pulseIn(ECHO_PIN, HIGH);
    return duration * 0.034 / 2;
}

long lastMsg = 0;
void loop() {
    float distance = readDistanceCM();

    bool isNearby = distance < 100; //checking whether the distance is less than 100
    digitalWrite(15, isNearby);

    Serial.print("Measured distance: ");
    Serial.println(readDistanceCM());

    delay(100);
    if(isNearby) //Whenever the distance is less than 100 cms send an "alert" to the IBM
cloud
    {

        client.loop();
        long now = millis();
        if (now - lastMsg > 3000) {
            lastMsg = now;

            String payload = "{\"distance\":";

```

```
payload += distance;
```

```
payload += "}";
```

```
Serial.print("Sending payload: ");  
Serial.println(payload);
```

```
if (client.publish(pubTopic1, (char*) payload.c_str())) {  
    Serial.println("Publish ok");  
} else {  
    Serial.println("Publish failed");  
}
```

```
}
```

```
}  
}
```

OUTPUT:

Case: 1

When Distance Is Above 100 Cm

The screenshot displays the WOKWI IoT simulator interface. On the left, the 'sketch.ino' file is open, showing the following code:

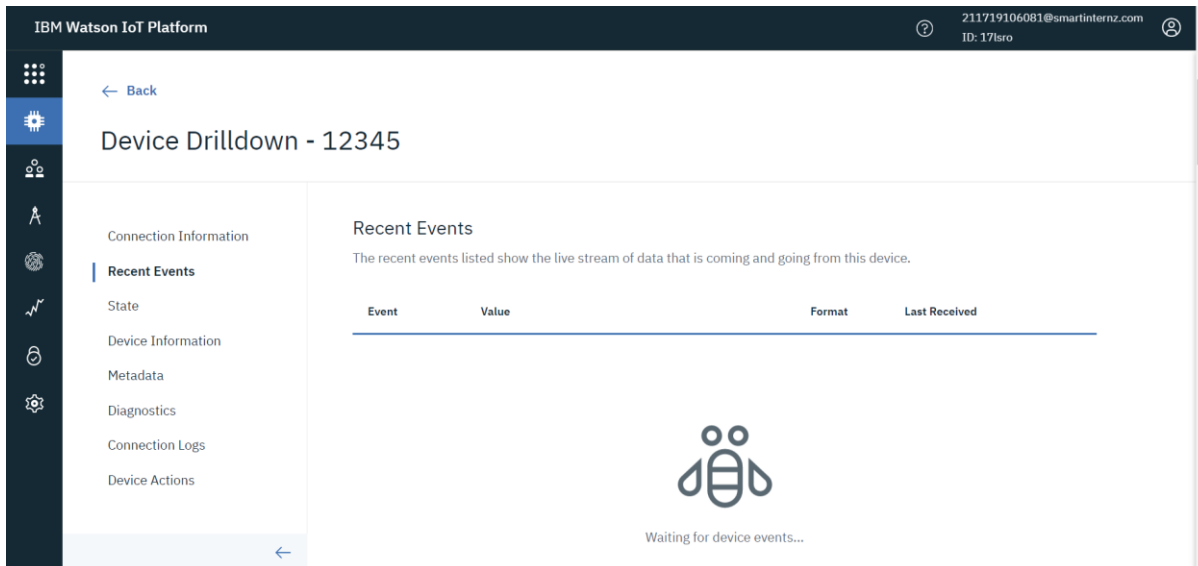
```
1 #include <WiFi.h>  
2 #include <WiFiClient.h>  
3 #include <PubSubClient.h>  
4  
5 #define ORG "17lsro"  
6 #define DEVICE_TYPE "MyDeviceType"  
7 #define DEVICE_ID "12345"  
8 #define TOKEN "GkatKdiUS?UVHKvnAD"  
9  
10 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";  
11 char pubTopic1[] = "iot-2/evt/ASHWIN KUMAR S/fmt/json";  
12 char pubTopic2[] = "iot-2/evt/status2/fmt/json";  
13 char authMethod[] = "use-token-auth";  
14 char token[] = TOKEN;  
15 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;  
16 const int DHT_PIN = 15;  
17 WiFiClient wifiClient;  
18 PubSubClient client(server, 1883, NULL, wifiClient);  
19  
20  
21 #define ECHO_PIN 12  
22 #define TRIG_PIN 13  
23  
24 float readDistanceCM();  
25 void setup() {  
26     Serial.begin(115200);  
27     pinMode(15, OUTPUT);  
28     pinMode(TRIG_PIN, OUTPUT);  
29 }
```

On the right, the 'Simulation' tab is active, showing a 3D model of an ESP32 module connected to an HC-SR04 ultrasonic sensor. The sensor's VCC pin is connected to the ESP32's VCC pin, and its GND pin is connected to the ESP32's GND pin. The sensor's TRIG pin is connected to the ESP32's TRIG_PIN (pin 13), and its ECHO pin is connected to the ESP32's ECHO_PIN (pin 12). A red LED is also connected to the ESP32's VCC and GND pins.

The simulation log shows the following output:

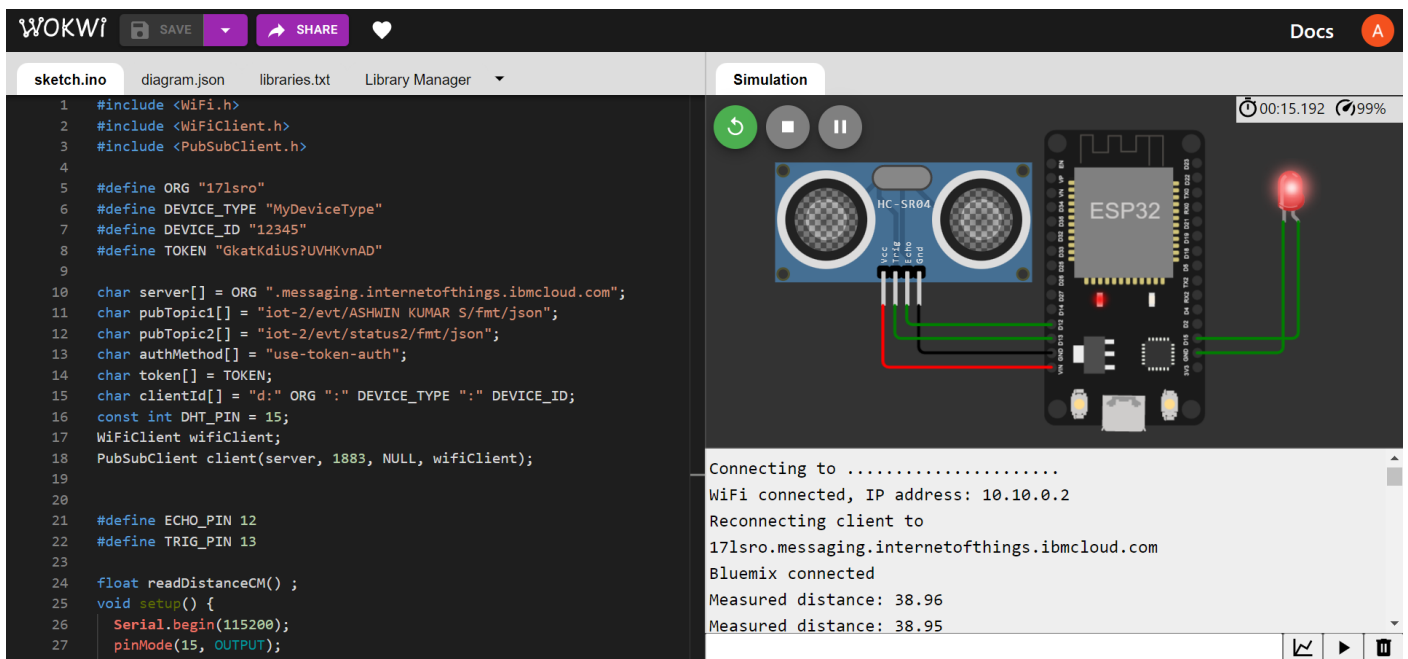
```
Connecting to .....  
WiFi connected, IP address: 10.10.0.2  
Reconnecting client to  
17lsro.messaging.internetofthings.ibmcloud.com  
Bluemix connected  
Measured distance: 258.96  
Measured distance: 259.03
```

Data Is Not Send to IBM IOT PLATFORM If Distance Is Above 100 Cm



Case:2

When Distance Is Below 100 Cm



When The Distance Is Below 100Cm Data Is Sent To IBM Iot Platform

IBM Watson IoT Platform

?

211719106081@smartinternz.com

ID: 17lsro

← Back

Device Drilldown - 12345

Connection Information

Recent Events

State

Device Information

Metadata

Diagnostics

Connection Logs

Device Actions

←

Recent Events

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
ASHWIN KU...	{"distance":38.96}	json	a few seconds ago
ASHWIN KU...	{"distance":38.96}	json	a few seconds ago
ASHWIN KU...	{"distance":38.96}	json	a few seconds ago