

ASSIGNMENT-4

TEAM ID: PNT2022TMID32030

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100 cms 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 <PubSubClient.h>

WiFiClient wifiClient;

String data3;

#define ORG "xhlz7n"//IBM ORGANITION ID

#define DEVICE_TYPE "ultrasonic_sensor"//Device type mentioned in ibm watson IOT Platform

#define DEVICE_ID "ultrasonic"//Device ID mentioned in ibm watson IOT Platform

#define TOKEN "od3sDQXgze)--k7P7p"

#define speed 0.034

#define led 14

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";

char publishTopic[] = "iot-2/evt/shreedharen/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");
```

```
}
```

```
}
```

```
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");
```

```
}
```

```
}
```

```
}
```

Output:

The screenshot shows the Wokwi 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 "xhlz7n"//IBM ORGANITION ID
6 #define DEVICE_TYPE "ultrasonic_sensor"//Device type mentioned in ibm watson IOT Platf
7 #define DEVICE_ID "ultrasonic"//Device ID mentioned in ibm watson IOT Platform
8 #define TOKEN "od3sDQXgze)--k7P7p"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/shreedharen/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 visual representation of the ESP32 microcontroller connected to an HC-SR04 ultrasonic sensor. The sensor's VCC pin is connected to the ESP32's 5V pin, and its GND pin is connected to the ESP32's GND pin. The simulation status bar at the top right indicates a runtime of 01:12.382 and 99% battery level.

The screenshot shows the Wokwi IDE interface with the simulation output window open. The output displays the following messages:

```
Connecting to Wifi...WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to xhlz7n.messaging.internetofthings.ibmcloud.com
IBM subscribe to cmd OK

Sending payload: {"Distance":399.92}
Publish OK

Sending payload: {"Distance":399.92}
Publish OK

1 Sending payload: {"Distance":399.96}
1 Publish OK

1
1 Sending payload: {"Distance":399.96}
1 Publish OK

1
1 Sending payload: {"Distance":399.94}
2 Publish OK
```

The simulation status bar at the top right indicates a runtime of 00:50.199 and 99% battery level.

Cloud upload:

The screenshot shows the IBM Watson IoT Platform dashboard. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains icons for various IoT functions. The main content area displays a table of devices. One device, 'ultrasonic', is selected and expanded to show its 'Recent Events'.

Device ID	Status	Device Type	Class ID	Date Added
ultrasonic	Connected	ultrasonic_sensor	Device	9 Nov 2022 10:05 PM

Identity Device Information Recent Events State Logs

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

Event	Value	Format	Last Received
shreedharen	{"Distance":399.98}	json	a few seconds ago
shreedharen	{"Distance":399.96}	json	a few seconds ago
shreedharen	{"Distance":399.96}	json	a few seconds ago
shreedharen	{"Distance":399.96}	json	a few seconds ago
shreedharen	{"Distance":399.92}	json	a few seconds ago

0 Simulations running

Wokwi link:

<https://wokwi.com/projects/347869236072284756>