

ASSIGNMENT 4

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

code:

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifClient;
String data3;
#define ORG "x0fxss"
#define DEVICE_TYPE "Noder"
#define DEVICE_ID "1234"
#define TOKEN "987654321"
#define speed 0.034
#define led 14

char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; char
publishTopic[] = "iot-2/evt/shanmugam_assignment4/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, wifClient);

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);  
    wifConnect();  
    mqttConnect();  
}
```

```
void loop() {
```

```
    bool isNearby = dist < 100;  
    digitalWrite(led, isNearby);
```

```
    publishData();  
    delay(500);
```

```
    if (!client.loop()) {  
        mqttConnect();  
    }  
}
```

```
void wifConnect() {
```

```
    Serial.print("Connecting to "); Serial.print("Wif");  
    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:- i) When distance greater than 100 cm

WOKWI

SAVE

SHARE

♥

Docs

sketch.ino

diagram.json

libraries.txt

Library Manager

```

1  {
2    "version": 1,
3    "author": "Keerthika J",
4    "editor": "wokwi",
5    "parts": [
6      { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 92.67, "left": 45.3
7      {
8        "type": "wokwi-led",
9        "id": "led1",
10       "top": 194.54,
11       "left": 309.26,
12       "attrs": { "color": "red" }
13     },
14     {
15       "type": "wokwi-hc-sr04",
16       "id": "ultrasonic1",
17       "top": 60.71,
18       "left": 185.64,
19       "attrs": { "distance": "139" }
20     },
21     {
22       "type": "wokwi-resistor",
23       "id": "r1",
24       "top": 269.89,
25       "left": 260.39,
26       "attrs": { "value": "100" }
27     }
28   ],
29   "connections": [
30     [ "esp:TX0", "$serialMonitor:RX", "", [] ],

```

Simulation

⏮

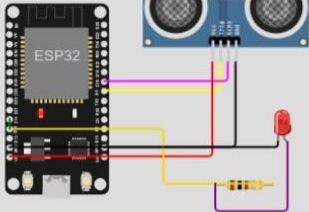
⏪

⏸

⏩

⏭

00:43.263 69%



Publish OK

Sending payload: {"Distance":138.96}

Publish OK

Sending payload: {"Distance":138.98}

Publish OK

The screenshot shows the IBM Watson IoT Platform interface. The top navigation bar includes the IBM logo and the text 'IBM Project-21602-1659785'. The main header displays 'IBM Watson IoT Platform' and the user 'keerthikaj.cse19@veltechmultitech.org' with ID '9pjim1'. The left sidebar contains icons for various functions. The main content area is titled 'Browse' and shows the 'Recent Events' tab for a device named '9pjim1'. The events are listed in a table with columns: Event, Value, Format, and Last Received. The events show a stream of distance data from a Node. The status bar at the bottom indicates '1 Simulation running'.

Event	Value	Format	Last Received
Node	{"distance":17}	json	a few seconds ago
Node	{"distance":153}	json	a minute ago
Node	{"distance":154}	json	a minute ago
Node	{"distance":155}	json	a minute ago
Node	{"distance":170}	json	a minute ago

Items per page 50 | 1-1 of 1 item

1 Simulation running

ii) When distance less than 100 cms.

The screenshot shows the Wokwi IDE interface. On the left, the 'diagram.json' file is open, displaying a JSON configuration for a simulation. The configuration includes parts like 'wokwi-esp32-devkit-v1', 'wokwi-led', 'wokwi-hc-sr04', and 'wokwi-resistor', along with their positions and attributes. The right pane shows a simulation window with a circuit diagram of an ESP32 board connected to an ultrasonic sensor and an LED. A console output shows the following messages:

```
Publish OK
Sending payload: {"Alert Distance":93.96}
Publish OK
Sending payload: {"Alert Distance":93.96}
Publish OK
```

The screenshot shows the IBM Watson IoT Platform dashboard. The 'Recent Events' tab is selected, displaying a table of events. The table has columns for Event, Value, Format, and Last Received. The events are as follows:

Event	Value	Format	Last Received
Node	{"distance":91}	json	a few seconds ago
Node	{"distance":45}	json	a few seconds ago
Node	{"distance":4}	json	a few seconds ago
Node	{"distance":19}	json	a few seconds ago
Node	{"distance":94}	json	a few seconds ago

At the bottom of the dashboard, it indicates '1 Simulation running'.