

**SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY**  
**ASSIGNMENT-4**

<b>NAME</b>	E. ANANDHI
<b>DATE</b>	25 <sup>th</sup> October 2022
<b>TEAM ID</b>	PNT2022TMID38378
<b>PROJECT NAME</b>	Signs with Smart Connectivity for Better Road Safety

**ASSIGNMENT-4**

Write code and connections in wokwi for ultrasonic sensors. 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.

**CODE:**

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

WiFiClient wifiClient;

#define ORG "o1z9pr"
#define DEVICE_TYPE "raspberrypi"
#define DEVICE_ID "12345"
#define TOKEN "12345678"
#define speed 0.034

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/raspberrypi_1/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="";
String lat="14.167589";
String lon="80.248510";
String name="point2";
String icon="";
```

```

long duration;
int dist;

void setup()
{
    Serial.begin(115200);
    pinMode(trigpin, OUTPUT);
    pinMode(echopin, INPUT);
    wifiConnect();
    mqttConnect();
}

void loop() {

    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(1000);
        }
        initManagedDevice();
        Serial.println();
    }
}

void initManagedDevice() {
    if (client.subscribe(topic)) {
        Serial.println(client.subscribe(topic));
    }
}

```

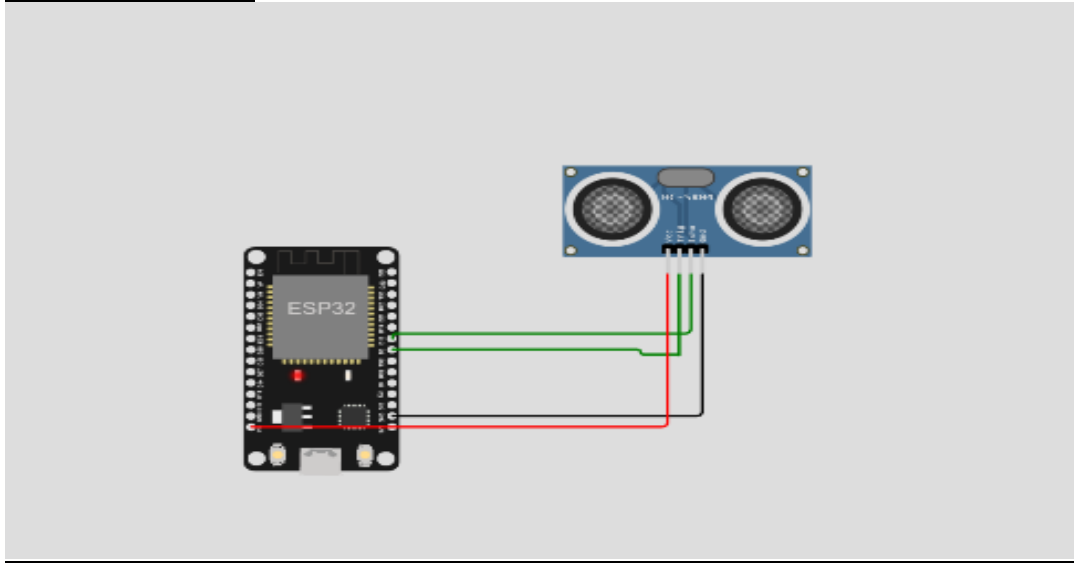
```

        Serial.println("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){
        dist=100-dist;
        icon="fa-trash";
    }else{
        dist=0;
        icon="fa-trash-o";
    }
    DynamicJsonDocument doc(1024);
    String payload;
    doc["Name"]=name;
    doc["Latitude"]=lat;
    doc["Longitude"]=lon;
    doc["Icon"]=icon;
    doc["FillPercent"]=dist;
    serializeJson(doc, payload);
    delay(3000);
    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");
    }
}
}

```

## CONNECTIONS:



## WOKWI LINK:

<https://wokwi.com/projects/346487528804581971>

## OUTPUT:

```
W sketchino - Wokwi Arduino and X W sketchino copy - Wokwi Arduino X +
WOKWI SAVE SHARE Docs
sketchino diagram.json libraries.txt Library Manager
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <ArduinoJson.h>
4
5 WiFiClient wifiClient;
6
7 #define ORG "o1z9pr"
8 #define DEVICE_TYPE "raspberrypi"
9 #define DEVICE_ID "12345"
10 #define TOKEN "12345678"
11 #define speed 0.034
12
13 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
14 char publishTopic[] = "iot-2/evt/raspberrypi_1/fmt/json";
15 char topic[] = "iot-2/cmd/home/fmt/String";
16 char authMethod[] = "use-token-auth";
17 char token[] = TOKEN;
18 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
19 PubSubClient client(server, 1883, wifiClient);
20 void publishData();
21
22 const int trigPin=5;
23 const int echoPin=16;
24 String command;
25 String data="";
26 String lat="14.167589";
27 String lon="80.248510";
28
29 void setup() {
30   pinMode(trigPin, OUTPUT);
31   pinMode(echoPin, INPUT);
32   Serial.begin(115200);
33   Serial.println("Starting");
34   client.setServer(1883);
35   client.setClient(wifiClient);
36   client.setCredentials(authMethod, token, clientId);
37   client.connect();
38   if (!client.connected()) {
39     Serial.println("MQTT client not connected");
40     return;
41   }
42   Serial.println("MQTT client connected");
43   client.subscribe(topic);
44   Serial.println("subscribe to cmd OK");
45 }
46
47 void loop() {
48   if (client.available()) {
49     String message = client.read();
50     Serial.println("Received: " + message);
51     ArduinoJson json;
52     json.parse(message);
53     if (json.success()) {
54       String event = json["event"];
55       String value = json["value"];
56       Serial.println("Event: " + event + " Value: " + value);
57       publishData();
58     }
59   }
60   delay(1000);
61 }
```

Connecting to Wifi...Wifi connected, IP address: 10.10.0.2  
Reconnecting MQTT client to  
o1z9pr.messaging.internetofthings.ibmcloud.com  
1  
subscribe to cmd OK

Browse	Action	Device Types	Interfaces	Add Device
12345	Disconnected	raspberrypi	Device	Oct 6, 2022 8:02 PM
raspberrypi_1	Connected	raspberrypi	Device	Oct 24, 2022 3:57 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	
event_1	{\"Alert distance\":10}	json	a few seconds ago	
event_1	{\"Alert distance\":27}	json	a few seconds ago	
event_1	{\"Alert distance\":49}	json	a few seconds ago	
event_1	{\"Alert distance\":67}	json	a few seconds ago	
event_1	{\"Alert distance\":87}	json	a few seconds ago	
9 Simulations running				

