

Assignment -4

Assignment Date	20 October 2022
Student Name	PRIYADARSHINI MD
Student Roll Number	111519104104
Maximum Marks	2 Marks

Question-1:

Write code and connections in wokwi for the ultrasonic sensor.

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

Solution :

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

WiFiClient wifiClient;

#define ORG "91w3mh"
#define DEVICE_TYPE "priya14"
#define DEVICE_ID "1429"
#define TOKEN "VTuMa!GUR&jTjA113?"
#define speed 0.034

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/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="";

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){
        DynamicJsonDocument doc(1024);
        String payload;
        doc["AlertDistance:"]=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");
        }
    }
}
}

```

<https://wokwi.com/projects/347219285148435027>

Speed Dial | W sketchino copy - Wokwi | IBM Watson IoT Platform

wokwi.com/projects/347219285148435027

WOKWI | sketchino copy | Docs | SIGN IN

sketchino | diagram.json | libraries.txt | Library Manager

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wificlient;
4 String data3;
5 #define ORG "91w3mh"
6 #define DEVICE_TYPE "priya14"
7 #define DEVICE_ID "1429"
8 #define TOKEN "VTuMaIGUR&JTJA113?"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/Data/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;
28
```

Simulation

02:38.226 92%

ESP32

HC-SR04

Publish OK

Sending payload: {"Distance":138.98}

26°C Raining now

18:35 02-11-2022

Speed Dial | W sketchino copy - Wokwi | IBM Watson IoT Platform

wokwi.com/projects/347219285148435027

WOKWI | sketchino copy | Docs | SIGN IN

sketchino | diagram.json | libraries.txt | Library Manager

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wificlient;
4 String data3;
5 #define ORG "91w3mh"
6 #define DEVICE_TYPE "priya14"
7 #define DEVICE_ID "1429"
8 #define TOKEN "VTuMaIGUR&JTJA113?"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/Data/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;
28
```

Simulation

00:26.514 100%

ESP32

HC-SR04

Sending payload: {"Distance":138.96}

Publish OK

Sending payload: {"Distance":138.96}

Publish OK

Sending payload: {"Distance":138.96}

Publish OK

26°C Raining now

18:25 02-11-2022

The screenshot displays the IBM Watson IoT Platform interface. At the top, the navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search icon is also present. On the right, there's a user profile section with the email '111519104104@smartinternz.com' and ID '91w3mh'. Below the navigation bar, a table lists devices. The selected device is 'priya14', which is 'Connected'. Below the device list, there's a section for 'Recent Events'. A message states: 'The recent events listed show the live stream of data that is coming and going from this device.' Below this message is a table of recent events.

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