

Assignment -2
Python Programming

Assignment Date	21 OCTOBER 2022
Student Name	Abirami P
Student Roll Number	513419106001
Maximum Marks	2

Question

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

CODE :

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "udeo3r"
#define DEVICE_TYPE "abi"
#define DEVICE_ID "131313"
#define TOKEN "Wja?4cr1fbZIxqT2JY"
#define speed 0.034
#define led 14
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=2;
const int echopin=4;
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 = "{\"Normal 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>101 && dist<111){
        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("Warning crosses 110cm -- it automaticaly of the loop");
        digitalWrite(led,HIGH);
    }else {
        Serial.println("Publish FAILED");
    }

}

}

void callback(char* subscribeTopic, byte* payload, unsigned int
payloadLength){
    Serial.print("callback invoked for topic:");
    Serial.println(subscribeTopic);
    for(int i=0; i<payloadLength; i++){
        dist += (char)payload[i];
    }
    Serial.println("data:"+ data3);
    if(data3=="lighton"){
        Serial.println(data3);
        digitalWrite(led,HIGH);
    }
    data3="";
}

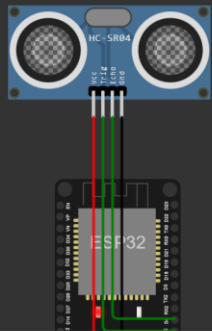
```

WOKWI

sketch.ino

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wificlient;
4 String data3;
5 #define ORG "udeo3r"
6 #define DEVICE_TYPE "abi"
7 #define DEVICE_ID "131313"
8 #define TOKEN "Wja74cr1fbZIXq123Y"
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/home/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 void publishData();
19
20
21 const int trigpin=2;
22 const int echopin=4;
23 String command;
24 String data="";
25
26 long duration;
27 float dist;
28
29
30
31 void setup()
32 {
33   Serial.begin(115200);
34 }
```

Simulation



Publish OK

Sending payload: {"Normal Distance":58.99}

Publish OK

Sending payload: {"Normal Distance":58.99}

Publish OK

27°C Cloudy

11:32 16-11-2022

github - Yahoo...IBM-EPBL...ibm cloud - Y...Service Detail...IBM Watson I...wokwi - Yahoo...New ESP32 P...whatsapp we...WhatsApp...+...udeo3r.internetofthings.ibmcloud.com/dashboard/devices/browse...Untitled mural • ou...

IBM Watson IoT Platformpalanlabirami39@gmail.comID: udeo3r

BrowseActionDevice TypesInterfaces

131313ConnectedabiDevice16 Nov 2022 11:20

IdentityDevice InformationRecent EventsStateLogs

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

Event	Value	Format	Last Received
Data	{*Normal Distance*:58.99}	json	a few seconds ago
Data	{*Normal Distance*:58.99}	json	a few seconds ago
Data	{*Normal Distance*:58.92}	json	a few seconds ago
Data	{*Normal Distance*:58.99}	json	a few seconds ago
Data	{*Normal Distance*:58.99}	json	a few seconds ago

Items per page 50 | 1-1 of 1 item

0 Simulations running

27°C Cloudy

ENG IN11:32 16-11-2022