

Assignment -4

Assignment Date	25 October 2022
Student Name	Kalaivani M
Student Roll Number	910419104008
Maximum Marks	2 Marks

Smart Waste Management system in metropolitan cities

Question-1:

Write code and connections in wokwi for ultrasonic sensors. Whenever distance is less than 100 cms send alert to ibm cloud & display in device recent events. Upload document with wokwi share link and images of ibm colud.

Solution:

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "08mif4"
#define DEVICE_TYPE "Kalai"
#define DEVICE_ID "Kalai123"
#define TOKEN "123456789"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/KalaiM/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;
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 automatically of
the loop");
            digitalWrite(led,HIGH);
        }else {
            Serial.println("Publish FAILED");
        }
    }

}

}

```

Output :

The screenshot displays an IDE with an Arduino sketch for an Ultrasonic Distance Sensor. The code includes headers for `PubSubClient` and `WiFiClient`, defines constants for the sensor's pin, distance, and device ID, and sets up an MQTT client. The `loop` function reads the distance and publishes it to a topic.

```
1 #include <Arduino.h>
2 #include <PubSubClient.h>
3 #include <WiFiClient.h>
4 String data;
5 #define TRIGPIN 10
6 #define DEVICE_ID "Kalai"
7 #define DEVICE_TYPE "Kalai"
8 #define TOKEN "123456789"
9 #define speed 0.034
10 #define led 14
11 char server[] = "mqtt://mqtt://192.168.1.1";
12 char publishTopic[] = "iot-2/evt/KalaiM/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:" DEVICE_ID ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, WiFiClient);
18 void publishData();
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
29 void setup()
30 {
31   Serial.begin(115200);
32   pinMode(led, OUTPUT);
33   pinMode(trigpin, OUTPUT);
34   pinMode(echopin, INPUT);
35   WiFi.begin("192.168.1.1");
36   WiFi.waitForConnect();
37   mqttConnect();
38 }
39
40 void loop() {
41   bool isNearby = dist < 100;
42   digitalWrite(led, isNearby);
43 }
```

The IoT platform interface shows a table of devices with columns: Device ID, Status, Device Type, Class ID, Date Added, Descriptive Location, and Added By. The table lists a device named "Kalai123" with status "Connected". Below the table, a section titled "Recent Events" shows a stream of data with columns: Event, Value, Format, and Last Received. The events show the distance being published to the topic.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By
Kalai123	Connected	Kalai	Device	23 Oct 2022 22:52		910419104008@smartintenz.com

Event	Value	Format	Last Received
KalaiM	["Normal Distance":82.94]	json	a few seconds ago
KalaiM	["Normal Distance":82.94]	json	a few seconds ago
KalaiM	["Normal Distance":80.95]	json	a few seconds ago
KalaiM	["Normal Distance":60.95]	json	a few seconds ago
KalaiM	["Normal Distance":65.98]	json	a few seconds ago

The screenshot displays an IDE with an Arduino sketch for an Ultrasonic Distance Sensor. The code includes headers for `PubSubClient` and `WiFiClient`, defines constants for the sensor's pin, distance, and device ID, and sets up an MQTT client. The `loop` function reads the distance and publishes it to a topic.

```
1 #include <Arduino.h>
2 #include <PubSubClient.h>
3 #include <WiFiClient.h>
4 String data;
5 #define TRIGPIN 10
6 #define DEVICE_ID "Kalai"
7 #define DEVICE_TYPE "Kalai"
8 #define TOKEN "123456789"
9 #define speed 0.034
10 #define led 14
11 char server[] = "mqtt://mqtt://192.168.1.1";
12 char publishTopic[] = "iot-2/evt/KalaiM/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:" DEVICE_ID ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, WiFiClient);
18 void publishData();
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
29 void setup()
30 {
31   Serial.begin(115200);
32   pinMode(led, OUTPUT);
33   pinMode(trigpin, OUTPUT);
34   pinMode(echopin, INPUT);
35   WiFi.begin("192.168.1.1");
36   WiFi.waitForConnect();
37   mqttConnect();
38 }
39
40 void loop() {
41   bool isNearby = dist < 100;
42   digitalWrite(led, isNearby);
43 }
```

The IoT platform interface shows a table of devices with columns: Device ID, Status, Device Type, Class ID, Date Added, Descriptive Location, and Added By. The table lists a device named "Kalai123" with status "Connected". Below the table, a section titled "Recent Events" shows a stream of data with columns: Event, Value, Format, and Last Received. The events show the distance being published to the topic.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By
Kalai123	Connected	Kalai	Device	23 Oct 2022 22:52		910419104008@smartintenz.com

Event	Value	Format	Last Received
KalaiM	["Alert distance":108.97]	json	a few seconds ago
KalaiM	["Alert distance":108.92]	json	a few seconds ago
KalaiM	["Alert distance":108.97]	json	a few seconds ago
KalaiM	["Alert distance":108.97]	json	a few seconds ago
KalaiM	["Alert distance":108.95]	json	a few seconds ago

WOKWI

RAYV

SHARE

WOKWI

esp32_dht22.ino • diagram.json • libraries.txt

Library Manager

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wificlient;
4 String data;
5 #define ORG "0Baird"
6 #define DEVICE_TYPE "Kalai"
7 #define DEVICE_ID "Kalai123"
8 #define TOKEN "13456789"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings";
12 char publishTopic[] = "iot-2/evt/kalai/hnt/json";
13 char topic[] = "iot-2/cmd/home/fnt/string";
14 char authMethod[] = "use-token-auth";
15 char token[] = TOKEN;
16 char clientId[] = "d-" ORG "-" DEVICE_TYPE "-";
17 PubSubClient client(server, 1883, wificlient);
18 void publishData();
19
20
21 const int trigpin=5;
22 const int echopin=16;
23 String command;
24 String data="";
25
26 long duration;
27 float dist;
28
29
30 void setup()
31 {
32   Serial.begin(115200);
33   pinMode(led, OUTPUT);
34   pinMode(trigpin, OUTPUT);
35   pinMode(echopin, INPUT);
36   wifiConnect();
37   mqttConnect();
38 }
39
40
41 void loop() {
42   bool isNearby = dist < 100;
43   digitalWrite(led, isNearby);
44
```

Simulation

01:02:765 75%

Editing Ultrasonic Distance Sensor

Distance: 189cm

ESP32

Warning crosses 110cm -- it automatically of the loop

Sending payload: {"Alert distance":108.97}

Warning crosses 110cm -- it automatically of the loop

Sending payload: {"Alert distance":108.97}

Warning crosses 110cm -- it automatically of the loop

Sending payload: {"Alert distance":108.92}

Warning crosses 110cm -- it automatically of the loop

Sending payload: {"Alert distance":108.97}

Warning crosses 110cm -- it automatically of the loop

Sending payload: {"Alert distance":108.97}

Warning crosses 110cm -- it automatically of the loop

IBM Watson IoT Platform

910419104008@smartinternz.com

ID: 0bma14

Browse Action Device Types Interfaces

Add Device

Search by Device ID

Device Simulator

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By
Kalai123	Connected	Kalai	Device	23 Oct 2022 22:52		910419104008@smartinternz.com

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
KalaiM	{"Alert distance":108.97}	json	a few seconds ago
KalaiM	{"Alert distance":108.92}	json	a few seconds ago
KalaiM	{"Alert distance":108.97}	json	a few seconds ago
KalaiM	{"Alert distance":108.97}	json	a few seconds ago
KalaiM	{"Alert distance":108.95}	json	a few seconds ago

Items per page 50 | 1-1 of 1 item

1 of 1 page