

Assignment -4

Assignment Date	29 October 2022
Student Name	S Hariharan
Student Roll Number	722819104039
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.

link- <https://wokwi.com/projects/347026058773529170>

The screenshot displays the Wokwi IDE interface. On the left, the 'sketch.ino' file contains the following code:

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <ArduinoJson.h>
4
5 #define ORG "7cclpf"
6 #define DEVICE_TYPE "iot_session"
7 #define DEVICE_ID "1234"
8 #define TOKEN "123456789"
9 #define speed 0.034
10
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
18 WiFiClient wificlient;
19 PubSubClient client(server, 1883, wificlient);
20
21 void publishData();
22 const int trigpin=18;
23 const int echopin=5;
24 String command;
25 String data="";
26 long duration;
27 int dist;
28
29 void setup()
30 {
```

On the right, the 'Simulation' window shows a visual representation of the hardware. An ESP32 microcontroller is connected to an HC-SR04 ultrasonic sensor. The sensor's VCC pin is connected to the ESP32's 5V pin, and its GND pin is connected to the ESP32's GND pin. The trig pin is connected to pin 18, and the echo pin is connected to pin 5. Below the simulation, the console output shows the following messages:

```
Sending payload: {"AlertDistance":72}
Publish FAILED
Reconnecting MQTT client to 7cclpf.messaging.internetofthings.ibmcloud.com
1
subscribe to cmd OK
72
```

Code :

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#define ORG "7cclpf"
#define DEVICE_TYPE "iot_session"
#define DEVICE_ID "1234"
#define TOKEN "123456789"
#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;
WiFiClient wifiClient;
PubSubClient client(server, 1883, wifiClient);
void publishData();
const int trigpin=18;
const int echopin=5;
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;
    Serial.println(dist);
    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");
        }
    }
}

```

IBM Watson IoT Platform

?

hariharan.s2019cse@secce.ac.in

ID: 7cd1pf

Browse

Action

Device Types

Interfaces

Add Device

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
Data	{"AlertDistance":92}	json	a few seconds ago
Data	{"AlertDistance":36}	json	a few seconds ago
Data	{"AlertDistance":36}	json	a few seconds ago
Data	{"AlertDistance":36}	json	a minute ago

Items per page 50 | 1-1 of 1 item

1 of 1 page

<

1

>