

Assignment - 4

Wowki & IBM Cloud

Assignment Date	13 November 2022
Student Name	Thenmozhi.A
Student Roll Number	723619106017
Maximum Marks	2 Marks

Question-1:

Write code and connections in wowki for the ultrasonic sensor. Whenever the distance is less than 100cms sent "alert" to IBM cloud and display in device recent events.

Code:

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

WiFiClient wifiClient;

#define ORG "oa3490"
#define DEVICE_TYPE "TestDeviceType"
#define DEVICE_ID "12345"
#define TOKEN "-A)0raS44f)fdjYBVS"
#define speed 0.034

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

```

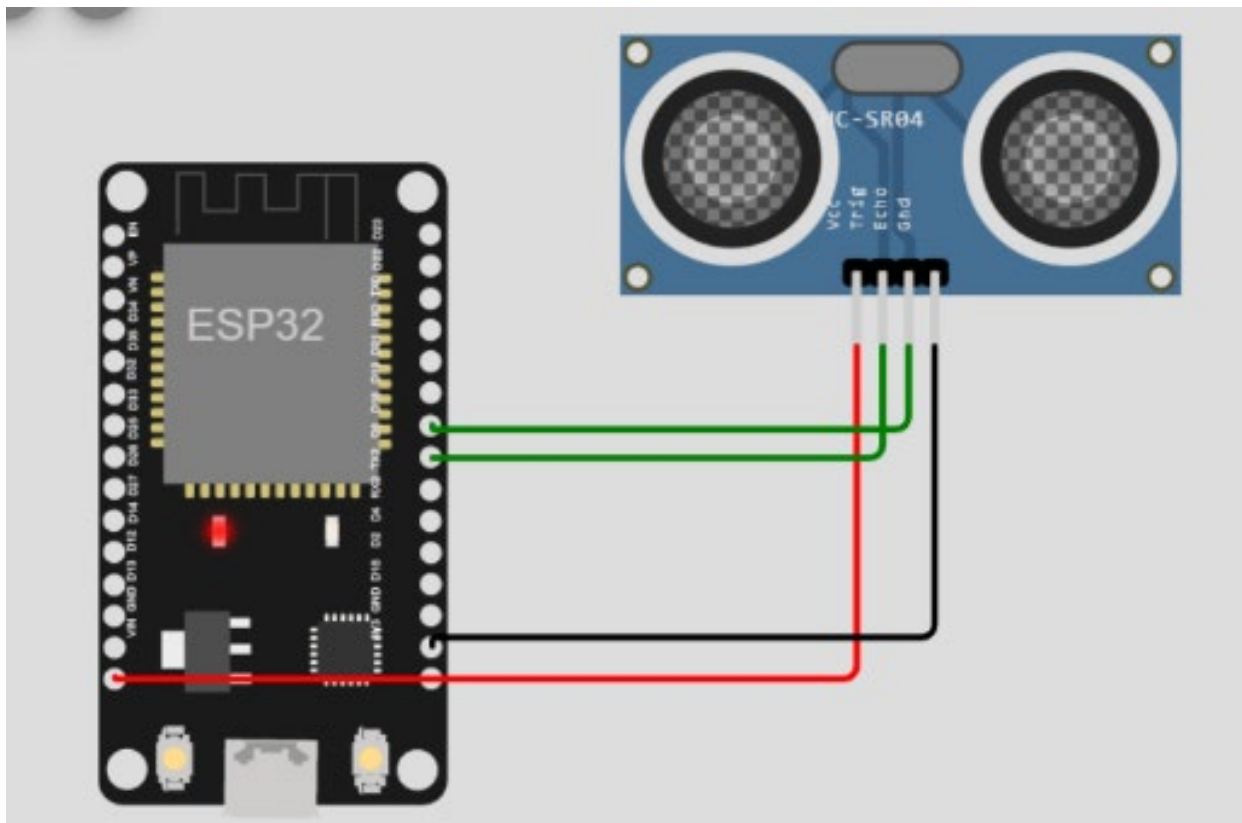
```

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:



Output:

The screenshot shows the Wokwi IDE interface. On the left, the sketch code is displayed, which includes headers for WiFi, PubSubClient, and ArduinoJson. It defines an organization, device type, device ID, token, and speed. The code sets up an ESP32 client and publishes distance data to IBM Cloud IoT Platform. On the right, the simulation window shows an ESP32 board connected to an HC-SR04 ultrasonic sensor. The sensor's distance is 34cm. Below the simulation, the published payload is shown as a JSON object.

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <ArduinoJson.h>
4
5 WiFiClient wificlient;
6
7 #define ORG "oa3490"
8 #define DEVICE_TYPE "TestDeviceType"
9 #define DEVICE_ID "12345"
10 #define TOKEN "-A0ra544f)fdjYBVS"
11 #define speed 0.034
12
13 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
14 char publishTopic[] = "iot-2/evt/abcd_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=18;
24 String command;
25 String data="";
26 String lat="14.167589";
27 String lon="80.248510";
28 String name="point2";
29 String icon="";
30
31 long duration;
32 int dist;
33
34 void setup()
35 {
```

Simulation window shows: Editing Ultrasonic Distance Sensor, Distance: 34cm. The published payload is: {"Name": "point2", "Latitude": "14.167589", "Longitude": "80.248510", "Icon": "fa-trash", "FillPercent": 6}.

Output :(IBM Cloud)

The screenshot shows the IBM Watson IoT Platform dashboard. The device with ID 12345 is listed as disconnected. The recent events table shows a stream of distance data.

Event	Value	Format	Last Received
event_1	{"Alert Distance":8}	json	a few seconds ago
event_1	{"Alert Distance":81}	json	a few seconds ago
event_1	{"Alert Distance":56}	json	a few seconds ago
event_1	{"Alert Distance":98}	json	a few seconds ago
event_1	{"Alert Distance":72}	json	a few seconds ago