

## Assignment-4

### WOWKI SIMULATION

|                     |                               |
|---------------------|-------------------------------|
| Assignment Date     | 4 <sup>th</sup> NOVEMBER 2022 |
| Student Name        | Akshaya. A                    |
| Student Roll Number | 960219106015                  |
| Maximum Marks       | 2 Marks                       |

#### Question-1:

Write a code and make a connection in WOKWI for ultrasonic sensor.  
Whenever distance is less than 100 , send “alert” to IBM cloud and display in device recent events.

PROGRAM:

```
#include <WiFi.h>

#include <PubSubClient.h>

WiFiClient wifiClient;

String data3;

#define ORG "273sie"

#define DEVICE_TYPE "b11m3edevicetype"

#define DEVICE_ID "b11m3deviceid"

#define TOKEN "5pcd+3rRln7d?_2)3"

#define speed 0.034

#define led 14

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";

char publishTopic[] = "iot-2/evt/Akshaya/fmt/json";

char topic[] = "iot-2/cmd/led/fmt/String";

char authMethod[] = "use-token-auth";

char token[] = TOKEN;
```

```
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);

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 = "{\"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("Publish OK");
        }

    }

    if(dist>100){
        String payload = "{\"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");

}else {

Serial.println("Publish FAILED");

}

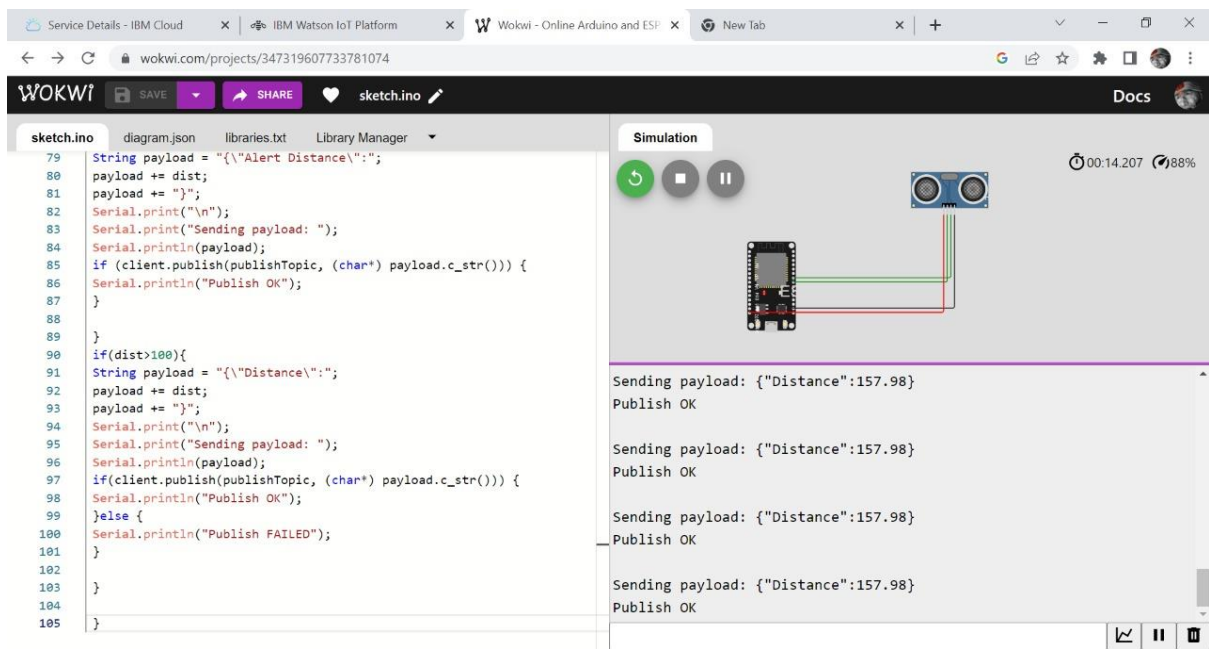
}

}

```

## OUTPUT:

## WOKWI SIMULATION



The screenshot shows the Wokwi online Arduino IDE simulation interface. The left pane displays the sketch code, and the right pane shows the simulation of an Arduino Uno connected to an ultrasonic sensor. The serial monitor displays the output of the code.

**Sketch Code:**

```

79 String payload = "{\"Alert Distance\": ";
80 payload += dist;
81 payload += "}";
82 Serial.print("\n");
83 Serial.print("Sending payload: ");
84 Serial.println(payload);
85 if (client.publish(publishTopic, (char*) payload.c_str())) {
86   Serial.println("Publish OK");
87 }
88
89 }
90
91 if(dist>100){
92   String payload = "{\"Distance\": ";
93   payload += dist;
94   payload += "}";
95   Serial.print("\n");
96   Serial.print("Sending payload: ");
97   Serial.println(payload);
98   if(client.publish(publishTopic, (char*) payload.c_str())) {
99     Serial.println("Publish OK");
100   }else {
101     Serial.println("Publish FAILED");
102   }
103 }
104
105 }

```

**Simulation Output:**

```

Sending payload: {"Distance":157.98}
Publish OK

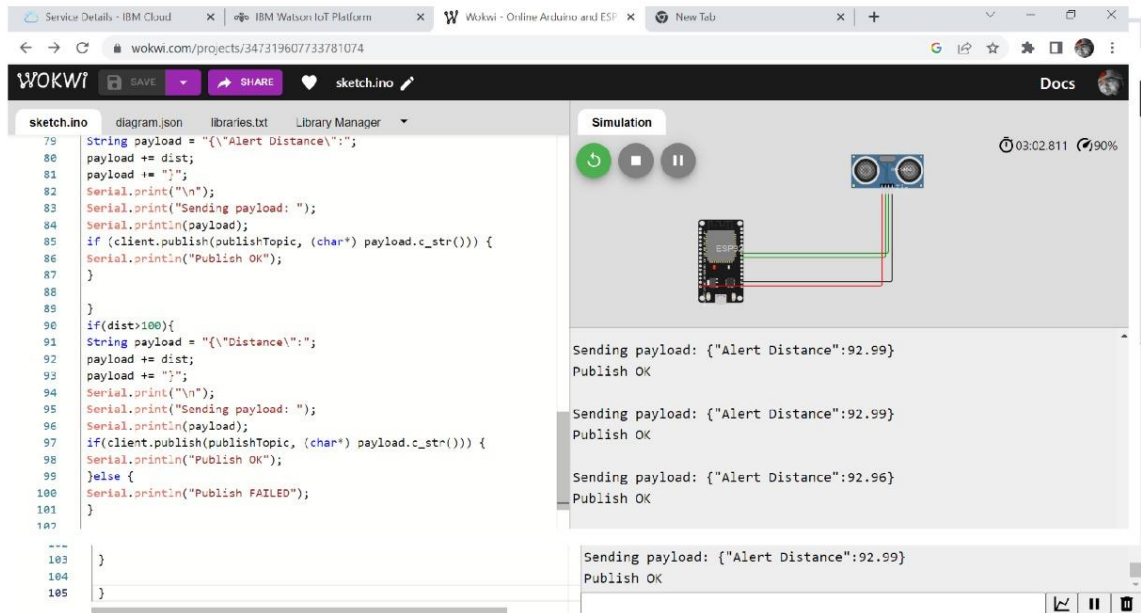
Sending payload: {"Distance":157.98}
Publish OK

Sending payload: {"Distance":157.98}
Publish OK

Sending payload: {"Distance":157.98}
Publish OK

```

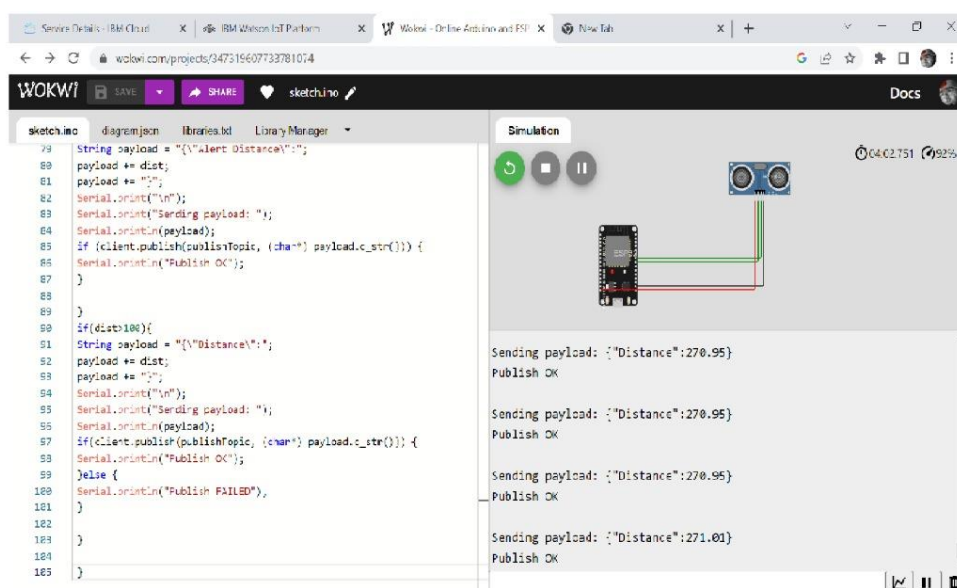
When distance<100:



The screenshot shows the Wokwi online IDE interface. On the left, the 'sketch.ino' file is open, displaying C++ code for an ESP8266. The code includes logic for sending an 'Alert Distance' payload when the distance is less than 100. The right panel shows a simulation of the ESP8266 connected to an ultrasonic sensor. Below the simulation, a log window displays the following output:

```
Sending payload: {"Alert Distance":92.99}
Publish OK
Sending payload: {"Alert Distance":92.99}
Publish OK
Sending payload: {"Alert Distance":92.96}
Publish OK
Sending payload: {"Alert Distance":92.99}
Publish OK
```

When distance>100:



The screenshot shows the Wokwi online IDE interface. On the left, the 'sketch.ino' file is open, displaying C++ code for an ESP8266. The code includes logic for sending a 'Distance' payload when the distance is greater than 100. The right panel shows a simulation of the ESP8266 connected to an ultrasonic sensor. Below the simulation, a log window displays the following output:

```
Sending payload: {"Distance":270.95}
Publish OK
Sending payload: {"Distance":270.95}
Publish OK
Sending payload: {"Distance":270.95}
Publish OK
Sending payload: {"Distance":271.01}
Publish OK
```

## IBM CLOUD OUTPUT

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

| Event   | Value                 | Format | Last Received     |
|---------|-----------------------|--------|-------------------|
| Akshaya | {"Alert Distance":43} | json   | a few seconds ago |
| Akshaya | {"Alert Distance":54} | json   | a few seconds ago |
| Akshaya | {"Alert Distance":52} | json   | a few seconds ago |
| Akshaya | {"Alert Distance":41} | json   | a few seconds ago |
| Akshaya | {"Alert Distance":29} | json   | a few seconds ago |

WOKWI LINK:

<https://wokwi.com/projects/347319607733781074>