

**Assignment -4**  
Wowki Simulation with IBM Watson

Assignment Date	19 November 2022
Student Name	Kowsik M
Student Roll Number	111519106076
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.

**Wowki Code:**

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

```
WiFiClient wifiClient;
```

```
#define ORG "luicm6"
#define DEVICE_TYPE "ThisDevice"
#define DEVICE_ID "12309"
#define TOKEN "V2oLr-jJK48pMO8rHx"
#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;
PubSubClient client(server, 1883, wifiClient); void
publishData();
```

```
const int trigpin=5;
const int echopin=18;
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;

  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 Cloud Output:

The screenshot displays the IBM Watson IoT Platform dashboard. The main view shows a table of devices with columns for Device ID, Status, Device Type, Class ID, and Date Added. A device with ID 12309 is highlighted, showing its status as 'Connected' and its device type as 'ThisDevice'. Below the table, the 'Recent Events' tab is active, displaying a list of events with columns for Event, Value, Format, and Last Received. The events show 'AlertDistance' values ranging from 40 to 86.

An overlay window titled '1 item selected' is visible, showing options to 'Create Simulated Device' or 'Use Registered Device'. The overlay also indicates '1/50 Simulations Running' and '777 events sent'.

Event	Value	Format	Last Received
event_hand	{"AlertDistance":44}	json	a few seconds ago
event_hand	{"AlertDistance":86}	json	a few seconds ago
event_hand	{"AlertDistance":52}	json	a few seconds ago
event_hand	{"AlertDistance":23}	json	a few seconds ago
event_hand	{"AlertDistance":40}	json	a few seconds ago

## WOWKI OUTPUT:

IBM Watson IoT Platform x Node-RED Dashboard x IBM x Node-RED : node-red-key: x esp32-dht22.ino - Wokwi x esp32-dht22.ino - Wokwi x

wokwi.com/projects/322410731508073042

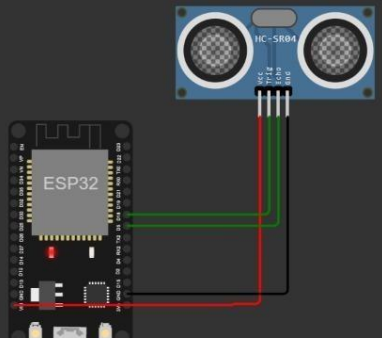
WOKWI SAVE SHARE esp32-dht22.ino by urish Docs

esp32-dht22.ino • diagram.json • libraries.txt • Library Manager

```
1
2 #include <WiFi.h>
3 #include <PubSubClient.h>
4 #include <ArduinoJson.h>
5
6 WiFiClient wifiClient;
7
8 #define ORG "luicmg"
9 #define DEVICE_TYPE "thisdevice"
10 #define DEVICE_ID "12309"
11 #define TOKEN "V2oLr-jJK48pM08rHx"
12 #define speed 0.034
13
14 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
15 char publishTopic[] = "iot-2/evt/data/fmt/json";
16 char topic[] = "iot-2/cmd/home/fmt/string";
17 char authMethod[] = "use token-auth";
18 char token[] = TOKEN;
19 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
20 PubSubClient client(server, 1883, wifiClient);
21 void publishData();
22
23 const int trigpin=5;
24 const int echopin=18;
25 String command;
26 String data="";
27
28 long duration;
29 int dist;
30
31 void setup()
32 {
33   Serial.begin(115200);
34   pinMode(trigpin, OUTPUT);
35   pinMode(echopin, INPUT);
36 }
```

Simulation

01:19.312 100%



Publish OK

Sending payload: {"AlertDistance":0}

Publish OK

Sending payload: {"AlertDistance":0}

Publish OK