

## ASSIGNMENT – 4

**Name** : Maruthamalaiayyanraja.A  
**Date** : 31.10.2001  
**Register Number** : 922119106055  
**Project Title** : Smart Solutions for Railways

### Question:

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.

Upload document with wokwi share link and images of IBM cloud.

### Program:

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
WiFiClient wifiClient;
#define ORG "akewo7"
#define DEVICE_TYPE "IBM"
#define DEVICE_ID "mmmkp"
#define TOKEN "31121110"
#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=13;
const int echopin=12;
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["Distance Alert:"]=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");
}
}
}
}

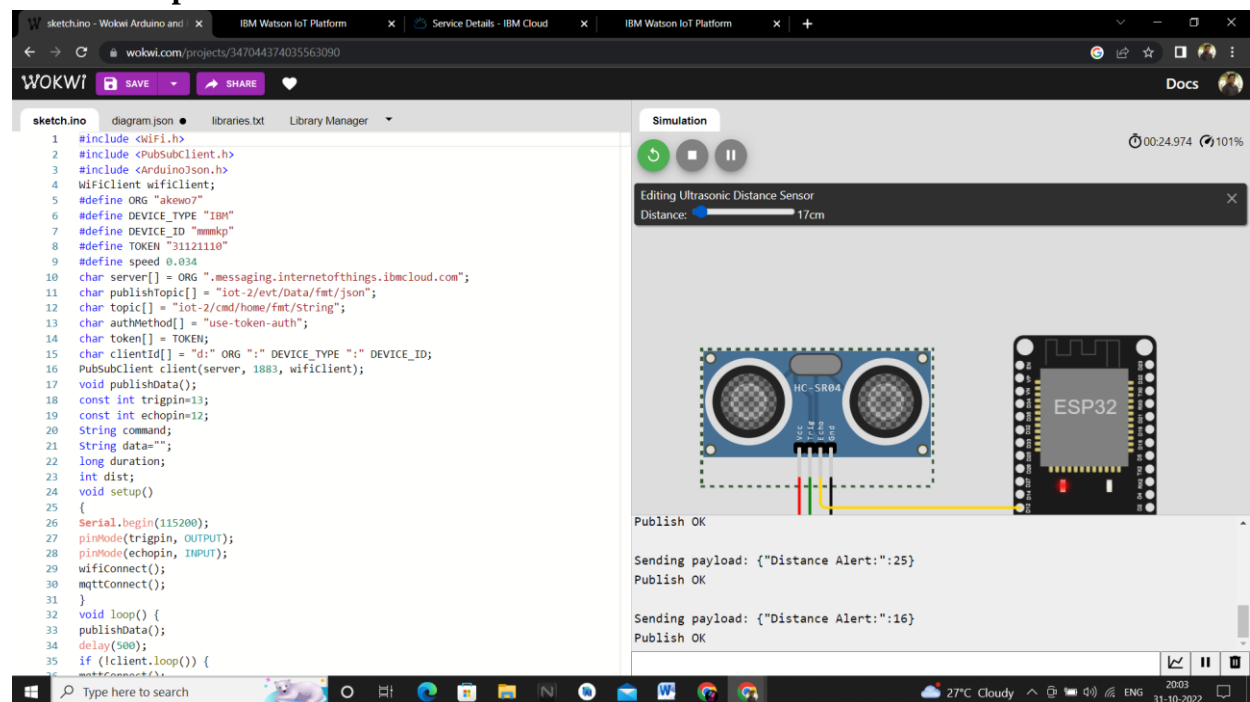
```

## Wokwi Link:

<https://wokwi.com/projects/347044374035563090>

## Output Images:

### Wokwi Output Screen



## IBM Cloud Result:

The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. The main content area shows a list of devices, with one device selected and its details expanded. The device 'mimikp' is shown as 'Disconnected' with an 'IBM' device type and 'Device' class. The 'Recent Events' tab is active, displaying a table of events.

Event	Value	Format	Last Received
Data	{"Distance Alert":16}	json	a few seconds ago
Data	{"Distance Alert":16}	json	a few seconds ago
Data	{"Distance Alert":25}	json	a few seconds ago
Data	{"Distance Alert":61}	json	a few seconds ago
Data	{"Distance Alert":61}	json	a few seconds ago

At the bottom of the dashboard, there is a status bar showing 'Items per page: 50' and '1 of 1 page'. The system tray at the very bottom indicates a temperature of 27°C, cloudy weather, and the date 31-10-2022.