

Smart Waste Management System for Metropolitan Cities -ASSIGNMENT 4

Name	MOHANAPRIYA.T
Date	26 October 2022
Team ID	PNT2022TMID38381
Project Name	Smart Waste Management System for Metropolitan Cities

ASSIGNMENT 4:

Write code and connections in wokwi for ultrasonic sensors.

Whenever distance is less than 100 cms send "alert" to ibm cloud and display in device recent events.

Upload document with wokwi share link and images of ibm cloud

CODE:

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;

#define ORG "xkgx5i"
#define DEVICE_TYPE "mohana123"
#define DEVICE_ID "12345678"
#define TOKEN "12345678"
#define speed 0.034

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

long duration;
float 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(500);
        }
        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");
    }
}

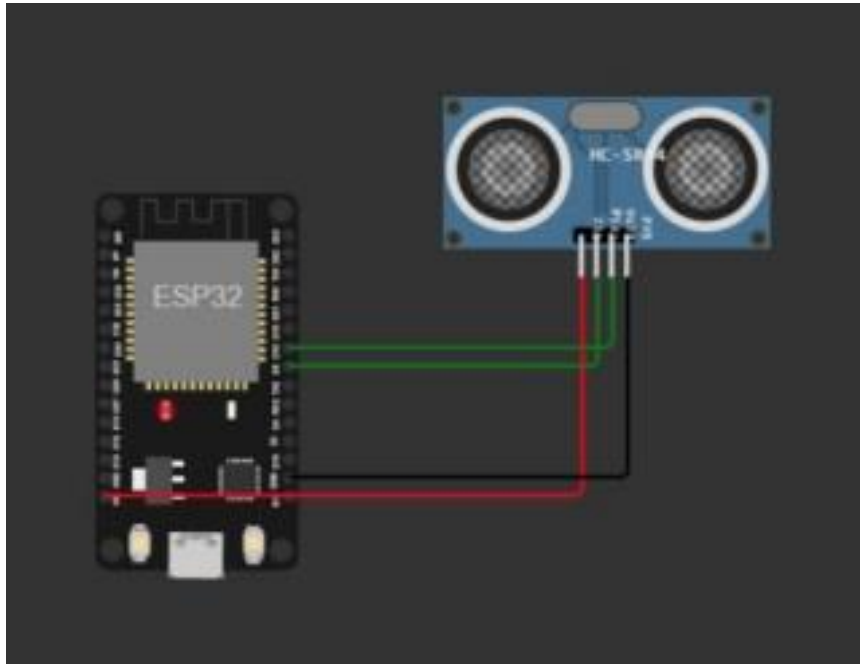
void publishData()
{
    digitalWrite(trigpin, LOW);
    digitalWrite(trigpin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigpin, LOW);
    duration=pulseIn(echopin, HIGH);
    dist=duration*speed/2;
    if(dist<100){
        String payload = "{\"Alert distance\": ";
        payload += dist;
        payload += "}";

        Serial.print("\n");
        Serial.print("Sending payload: ");
        Serial.println(payload);
        if (client.publish(publishTopic, (char*) payload.c_str()))
        { Serial.println("Publish OK");
        } else {
            Serial.println("Publish FAILED");
        }
    }
}

```



CONNECTIONS:



WOKWI LINK:

<https://wokwi.com/projects/346767055901426258>

OUTPUT:

WOKWI! SAVE SHARE sketch.ino copy Docs

sketch.ino diagram.json libraries.txt Library Manager

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wifiClient;
4 #define ORG "xkgx5i"
5 #define DEVICE_TYPE "raspberrypi"
6 #define DEVICE_ID "mohana123"
7 #define TOKEN "12345678"
8 #define speed 0.034
9 char server[] = ORG
10 ".messaging.internetofthings.ibmcloud.com"; char
11 publishTopic[] = "iot-2/evt/raspberrypi_1/fmt/json";
12 char topic[] = "iot-2/cmd/home/fmt/String";
13 char authMethod[] = "use-token-auth";
14 char token[] = TOKEN;
15
16 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, wifiClient);
18 void publishData();
19 const int trigpin=5;
20 const int echopin=18;
21 String command;
22 String data="";
23 long duration;
24 float dist;
25
26 void setup()
27 {
28   Serial.begin(115200);
```

Simulation

Restart the simulation

Connecting to Wifi...WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to
xkgx5i.messaging.internetofthings.ibmcloud.com

00:03.081 41%

IBM Watson IoT Platform

12345678 Disconnected mohana123 Device Nov 3, 2022 3:03 PM

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
event_1	{"Alert Distance":9}	json	a few seconds ago
event_1	{"Alert Distance":55}	json	a few seconds ago
event_1	{"Alert Distance":97}	json	a few seconds ago
event_1	{"Alert Distance":13}	json	a few seconds ago
event_1	{"Alert Distance":99}	json	a few seconds ago

1. Simulation running