

Assignment - 4

Assignment Date	6 November 2022
Team ID	PNT2022TMID27875
Register Number	IoT Based Safety Gadget for Child Safety Monitoring and Notification
Maximum Marks	2 Marks

ASSIGNMENT TOPIC:

Write code and connections in wokwi for ultrasonic sensor. 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 "dluuhi"
#define DEVICE_TYPE "Assignment_4"
#define DEVICE_ID "23323850"
#define TOKEN "12345678"
#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=2;
const int echopin=15;
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);
}

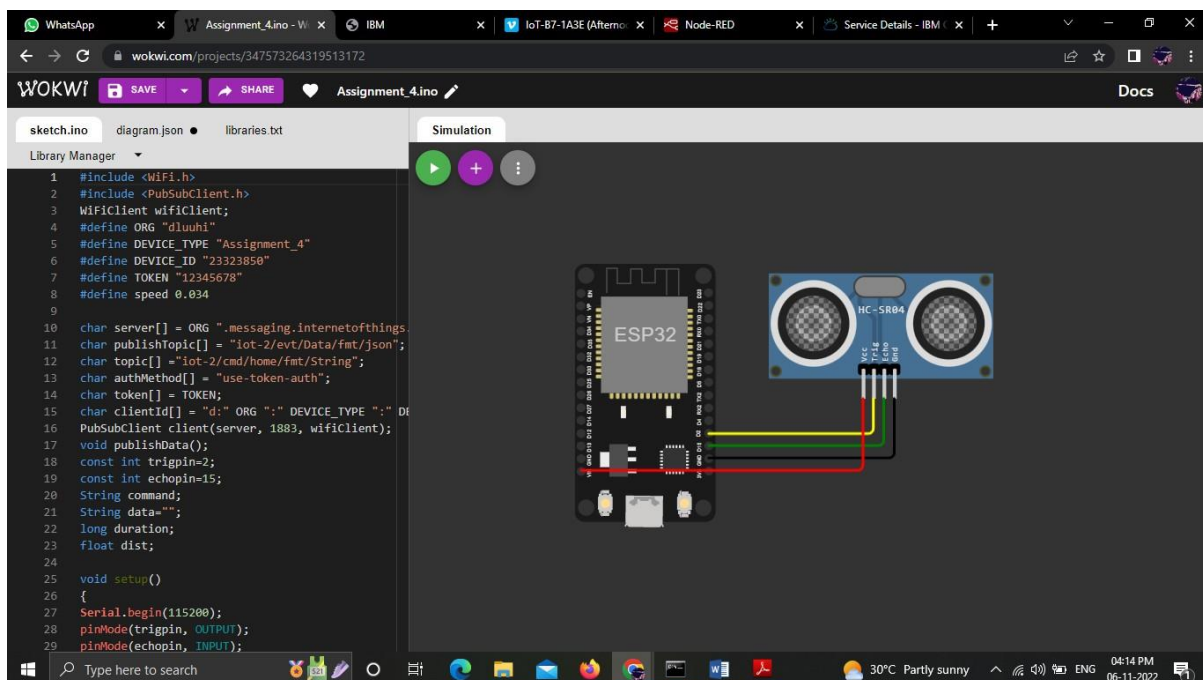
```

```

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:



OUTPUT:

The screenshot displays the Wokwi IoT simulator interface. On the left, the 'sketch.ino' file is open, showing the following code:

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 WiFiClient wificlient;
4 #define ORG "dluuhi"
5 #define DEVICE_TYPE "Assignment_4"
6 #define DEVICE_ID "23323850"
7 #define TOKEN "12345678"
8 #define speed 0.034
9
10 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
11 char publishTopic[] = "iot-2/evt/Data/fmt/json";
12 char topic[] = "iot-2/cmd/home/fmt/String";
13 char authMethod[] = "use-token-auth";
14 char token[] = TOKEN;
15 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
16 PubSubClient client(server, 1883, wificlient);
17 void publishData();
18 const int trigpin=2;
19 const int echopin=15;
20 String command;
21 String data="";
22 long duration;
23 float dist;
24
25 void setup()
26 {
27   Serial.begin(115200);
28   pinMode(trigpin, OUTPUT);
29   pinMode(echopin, INPUT);
30   wifiConnect();
31 }
```

The simulation window on the right shows the hardware components: an ESP32 microcontroller and an HC-SR04 ultrasonic distance sensor. The console output indicates the following sequence of events:

- Connecting to Wifi..Wifi connected, IP address: 10.10.0.2
- Reconnecting MQTT client to dluuhi.messaging.internetofthings.ibmcloud.com
- subscribe to cmd OK
- Sending payload: {"Alert distance":67.95}
- Publish OK

The second screenshot shows the simulation at a later point in time. The console output now includes:

- Publish OK
- Sending payload: {"Alert distance":11.97}
- Publish OK
- Sending payload: {"Alert distance":11.98}
- Publish OK

The 'Editing Ultrasonic Distance Sensor' dialog box is also visible, showing a distance of 12cm.

OUTPUT IN IOT CLOUD PLATFORM:

The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains various icons for navigation. The main content area shows a device with ID '23323850' in a 'Connected' state, assigned to 'Assignment_4'. Below this, a 'Recent Events' tab is active, displaying a table of events. The table has columns for 'Event', 'Value', 'Format', and 'Last Received'. All events are of type 'Data' with a value of '{"Alert distance":11.98}' in 'json' format, received 'a few seconds ago'. A status bar at the bottom indicates '0 Simulations running'.

Event	Value	Format	Last Received
Data	{"Alert distance":11.98}	json	a few seconds ago
Data	{"Alert distance":11.98}	json	a few seconds ago
Data	{"Alert distance":11.98}	json	a few seconds ago
Data	{"Alert distance":11.98}	json	a few seconds ago
Data	{"Alert distance":11.98}	json	a few seconds ago

WOKWI LINK:

<https://wokwi.com/projects/347573264319513172>