

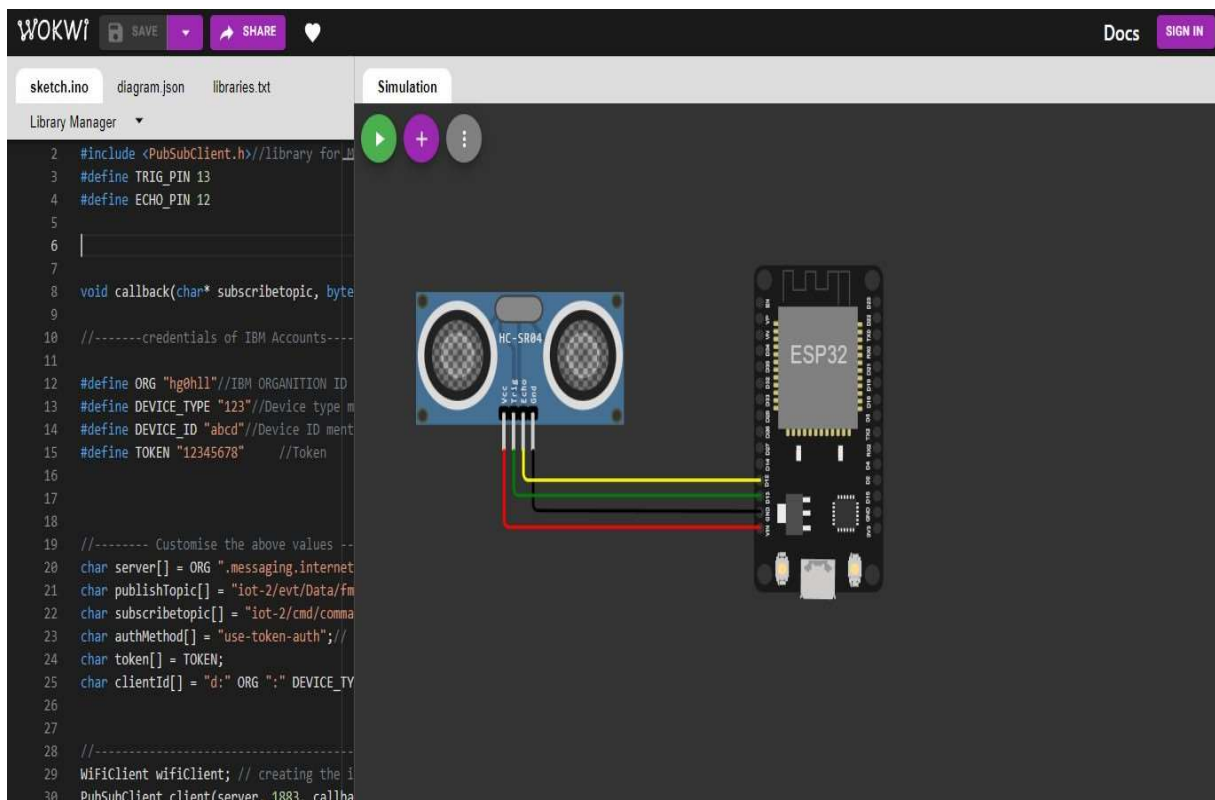
ASSIGNMENT – 4

Name	THIRUMARAN A
Date	05-11-2022
Team ID	PNT2022TMID17427
Project name	Gas leakage monitoring and alerting system

Objective:

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.

Circuit diagram:



Output:

The screenshot displays the Wokwi IoT simulator interface. The top bar includes the Wokwi logo, a 'SAVE' button, a 'SHARE' button, and a 'SIGN UP' button. Below the top bar, there are tabs for 'sketch.ino', 'diagram.json', and 'libraries.txt'. The 'sketch.ino' tab is active, showing a code editor with the following code:

```
2 #include <PubSubClient.h> //library for MQTT
3 #define TRIG_PIN 13
4 #define ECHO_PIN 12
5
6
7
8 void callback(char* subscribetopic, byte
9
10 //-----credentials of IBM Accounts-----
11
12 #define ORG "hg0h11" //IBM ORGANITION ID
13 #define DEVICE_TYPE "123" //Device type m
14 #define DEVICE_ID "abcd" //Device ID meht
15 #define TOKEN "12345678" //Token
16
17
18 //----- Customise the above values -----
19
20 char server[] = ORG ".messaging.internet
21 char publishTopic[] = "iot-2/evt/Data/fi
22 char subscribetopic[] = "iot-2/cmd/comme
23 char authMethod[] = "use-token-auth"; //
24 char token[] = TOKEN;
25 char clientId[] = "d:" ORG ":" DEVICE_TY
26
27
28 //-----
29 WiFiClient wificlient; // creating the
30 PubSubClient client(server, 1883, callba
```

The 'Simulation' tab is also active, showing a visual representation of the hardware. It features a blue HC-SR04 ultrasonic sensor connected to an ESP32 microcontroller via four colored wires (red, yellow, green, and blue). The ESP32 is shown with its pin headers and components. The simulation window includes a 'Simulation' button (a green circle with a white play icon) and a 'Pause' button (a grey square with a white pause icon). A timer in the top right corner of the simulation window shows '00:17.262' and '53%'.

The console output at the bottom of the simulation window shows the following messages:

```
Publish ok
Sending payload: {"Distance":84.98,"MESSAGE":"ALERT"}
Publish ok
Sending payload: {"Distance":84.98,"MESSAGE":"ALERT"}
Publish ok
Sending payload: {"Distance":84.98,"MESSAGE":"ALERT"}
Publish ok
```

