

ASSIGNMENT - 4

DATE	17 OCTOBER 2022
TEAM ID	PNT2022TMID06972
NAME	ARUNA S
STUDENT ROLL NUMBER	GCTC1918104
MAXIMUM MARKS	2 MARKS

QUESTION:

Write code and connections in wokwi for the ultrasonic sensor.

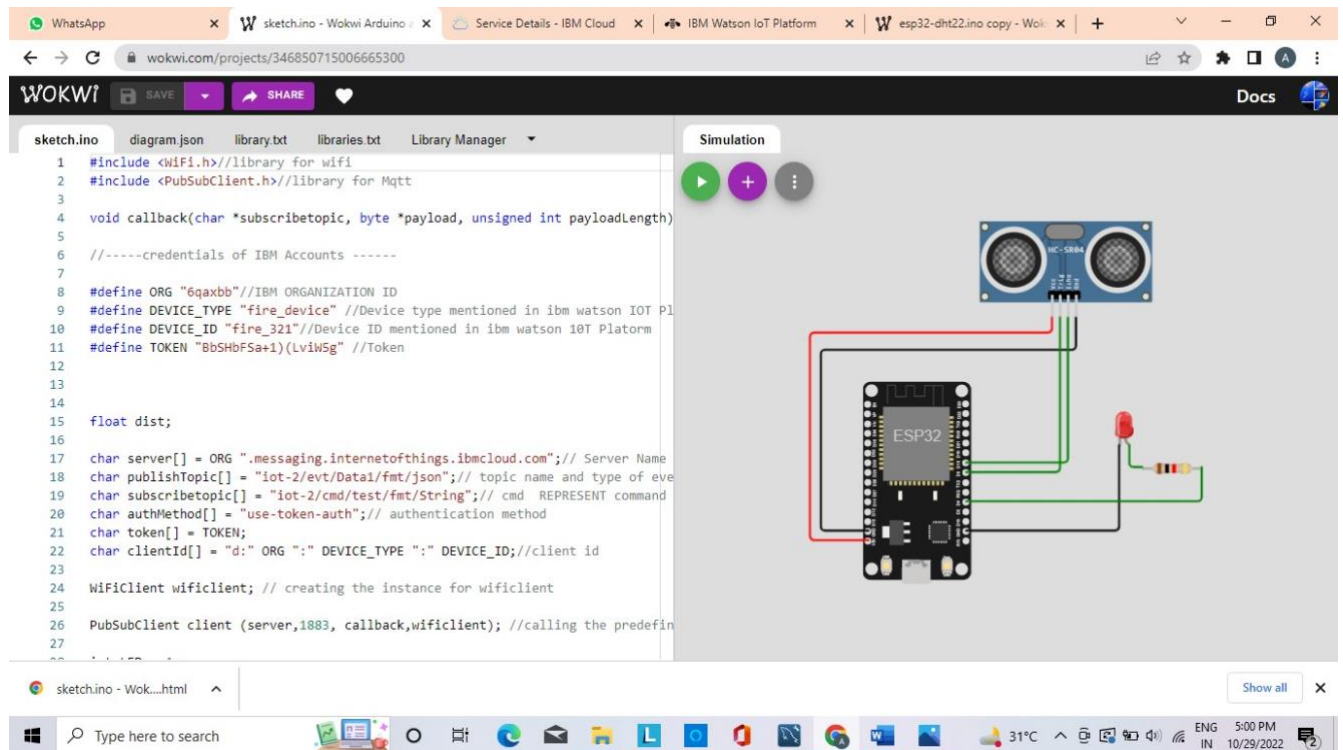
Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events.

Upload document with wokwi share link and images of IBM cloud

WOKWI CODE AND IMPLEMENTATION LINK:

<https://wokwi.com/projects/346850715006665300>

CODE:



The screenshot displays the Wokwi IDE interface. On the left, the 'sketch.ino' file is open, showing an Arduino sketch that configures an ESP32 to connect to IBM Watson IoT Platform via MQTT. The sketch includes necessary libraries, defines IBM credentials, sets up a WiFi client, and registers a callback function for MQTT messages. On the right, a 'Simulation' window shows a virtual circuit. An ESP32 module is connected to an HC-SR04 ultrasonic sensor. The sensor's VCC pin is connected to the ESP32's 5V pin, and its GND pin is connected to the ESP32's GND pin. The sensor's TRIG pin is connected to the ESP32's D4 pin, and its ECHO pin is connected to the ESP32's D5 pin. A red LED is also connected to the ESP32's D2 pin and GND. The bottom of the image shows a Windows taskbar with various application icons and a system clock indicating 5:00 PM on 10/29/2022.

```
1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for Mqtt
3
4 void callback(char *topic, byte *payload, unsigned int payloadLength)
5
6 //-----credentials of IBM Accounts -----
7
8 #define ORG "6qaxbb" //IBM ORGANIZATION ID
9 #define DEVICE_TYPE "fire_device" //Device type mentioned in ibm watson IOT Platform
10 #define DEVICE_ID "fire_321" //Device ID mentioned in ibm watson IOT Platform
11 #define TOKEN "BbSHbFSa+1(LviW5g" //Token
12
13
14
15 float dist;
16
17 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
18 char publishTopic[] = "iot-2/evt/Data1/fmt/json"; // topic name and type of event
19 char subscribetopic[] = "iot-2/cmd/test/fmt/String"; // cmd REPRESENT command
20 char authMethod[] = "use-token-auth"; // authentication method
21 char token[] = TOKEN;
22 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
23
24 WiFiClient wificlient; // creating the instance for wificlient
25
26 PubSubClient client (server,1883, callback,wificlient); //calling the predefined
27
```

WhatsApp x sketchino - Wokwi Arduino x Service Details - IBM Cloud x IBM Watson IoT Platform x esp32-dht22.ino copy - Wok x + -

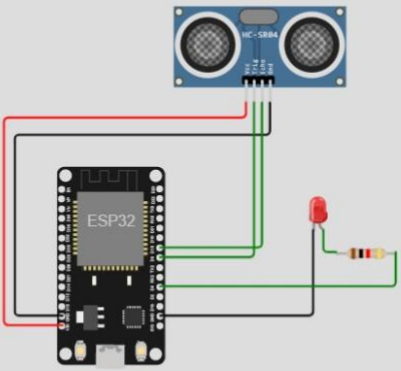
wokwi.com/projects/34685071500665300

WOKWI SAVE SHARE

sketch.ino diagram.json library.txt libraries.txt Library Manager

```
28 int LED = 4;
29
30 int trig =5;
31
32 int echo= 18;
33
34 void setup()
35 {
36
37   Serial.begin(115200);
38   pinMode(trig, OUTPUT);
39   pinMode(echo, INPUT);
40   pinMode(LED, OUTPUT);
41   delay(10);
42
43   wificonnect();
44   mqttconnect();
45
46 }
47
48
49 void loop()// Recursive Function
50 {
51   delayMicroseconds(10);
52   digitalWrite(trig, LOW);
53   digitalWrite(trig, LOW);
54 }
```

Simulation



sketchino - Wok...html

Type here to search

31°C 5:00 PM 10/29/2022

WhatsApp x sketchino - Wokwi Arduino x Service Details - IBM Cloud x IBM Watson IoT Platform x esp32-dht22.ino copy - Wok x + -

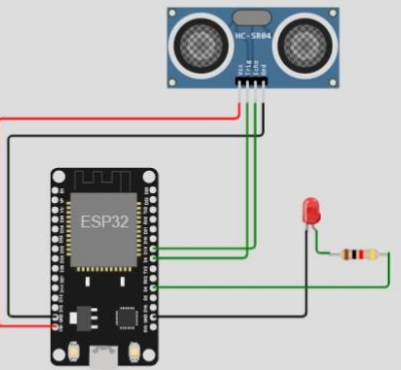
wokwi.com/projects/34685071500665300

WOKWI SAVE SHARE

sketch.ino diagram.json library.txt libraries.txt Library Manager

```
90 String payload="{\"distance\":";
91 payload += dist;
92 payload += "," \"object\":\";
93 payload += object;
94 payload += "\";
95
96 Serial.print("Sending payload: ");
97 Serial.println(payload);
98
99 if (client.publish(publishTopic, (char*) payload.c_str()))
100 {
101   Serial.println("Publish ok"); // if it sucessfully upload
102 }
103 else {
104   Serial.println("Publish failed");
105 }
106 }
107
108 void mqttconnect() {
109   if (!client.connected()) {
110     Serial.print("Reconnecting client to ");
111     Serial.println(server);
112     while (!client.connect(clientId, authMethod, token)) {
113       Serial.print(".");
114       delay(500);
115     }
116 }
```

Simulation



sketchino - Wok...html

Type here to search

31°C 5:01 PM 10/29/2022

WhatsApp x sketchino - Wokwi Arduino x Service Details - IBM Cloud x IBM Watson IoT Platform x esp32-dht22.ino copy - Wok x + -

wokwi.com/projects/346850715006665300

WOKWI SAVE SHARE

sketch.ino diagram.json library.txt libraries.txt Library Manager

```

137 }
138 void initManagedDevice() {
139   if (client.subscribe(subscribetopic)) {
140     Serial.println(subscribetopic);
141     Serial.println("subscribe to cmd OK");
142   }
143   else {
144     Serial.println("subscribe to cmd FAILED");
145   }
146 }
147 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
148 {
149   Serial.print("callback invoked for topic: ");
150   Serial.println(subscribetopic);
151   for (int i = 0; i < payloadLength; i++) {
152     //Serial.print((char)payload[i]);
153     // data3 += (char)payload[i];
154   }
155   // Serial.println("data: "+ data3);
156   //if(data3=="lighton")
157   {
158     //Serial.println(data3);
159     digitalWrite(LED,HIGH);
160   }
161   //else
162   {
163     //Serial.println(data3);
164     digitalWrite(LED,LOW);
165   }
166 }

```

Simulation

sketchino - Wok...html

Type here to search

31°C 5:01 PM 10/29/2022

WhatsApp x sketchino - Wokwi Arduino x Service Details - IBM Cloud x IBM Watson IoT Platform x esp32-dht22.ino copy - Wok x + -

wokwi.com/projects/346850715006665300

WOKWI SAVE SHARE

sketch.ino diagram.json library.txt libraries.txt Library Manager

```

1 {
2   "version": 1,
3   "author": "ARUNA S GCTC1918104",
4   "editor": "wokwi",
5   "parts": [
6     { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 41.94, "left": -151,
7       "parts": [
8         { "type": "wokwi-hc-sr04",
9           "id": "ultrasonic1",
10          "top": -113.83,
11          "left": -37.38,
12          "attrs": { "distance": "217" }
13        },
14        { "type": "wokwi-led",
15          "id": "led1",
16          "top": 64.64,
17          "left": 85.74,
18          "attrs": { "color": "red" }
19        },
20        { "type": "wokwi-resistor",
21          "id": "r1",
22          "top": 119.16,
23          "left": 123.46,
24          "attrs": { "value": "1000" }
25        }
26      ]
27     }
28   ]
29 }

```

Simulation

sketchino - Wok...html

Type here to search

31°C 5:01 PM 10/29/2022

WhatsApp x sketchino - Wokwi Arduino x Service Details - IBM Cloud x IBM Watson IoT Platform x esp32-dht22.ino copy - Wok x + -

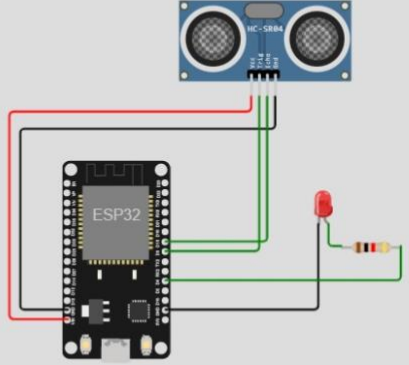
wokwi.com/projects/346850715006665300

WOKWI SAVE SHARE Docs

sketch.ino diagram.json library.txt libraries.txt Library Manager

```
28 ],
29 "connections": [
30 [ "esp:TX0", "$SerialMonitor:RX", "", [] ],
31 [ "esp:RX0", "$SerialMonitor:TX", "", [] ],
32 [ "ultrasonic1:VCC", "esp:VIN", "red", [ "v14.72", "h-238.87", "v203.46" ] ],
33 [ "ultrasonic1:GND", "esp:GND.2", "black", [ "v31.67", "h-251.83", "v180.2" ] ],
34 [ "ultrasonic1:TRIG", "esp:D5", "green", [ "v0" ] ],
35 [ "ultrasonic1:ECHO", "esp:D18", "green", [ "v0" ] ],
36 [ "esp:GND.1", "led1:C", "black", [ "h0" ] ],
37 [ "led1:A", "r1:1", "green", [ "v0" ] ],
38 [ "esp:D4", "r1:2", "green", [ "h0" ] ]
39 ]
40 ]
```

Simulation



sketchino - Wok...html Show all

Type here to search 31°C ENG IN 5:01 PM 10/29/2022

WhatsApp x sketchino - Wokwi Arduino x Service Details - IBM Cloud x IBM Watson IoT Platform x esp32-dht22.ino copy - Wok x + -

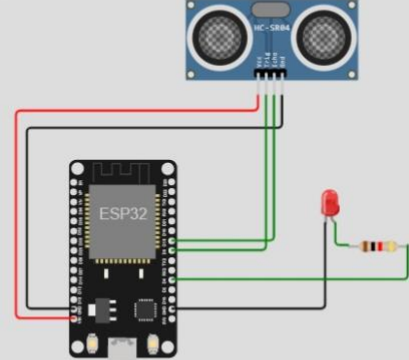
wokwi.com/projects/346850715006665300

WOKWI SAVE SHARE Docs

sketch.ino diagram.json library.txt libraries.txt Library Manager

```
1 Wokwi Library List
2 # See https://docs.wokwi.com/guides/libraries
3
4 PubSubClient
```

Simulation

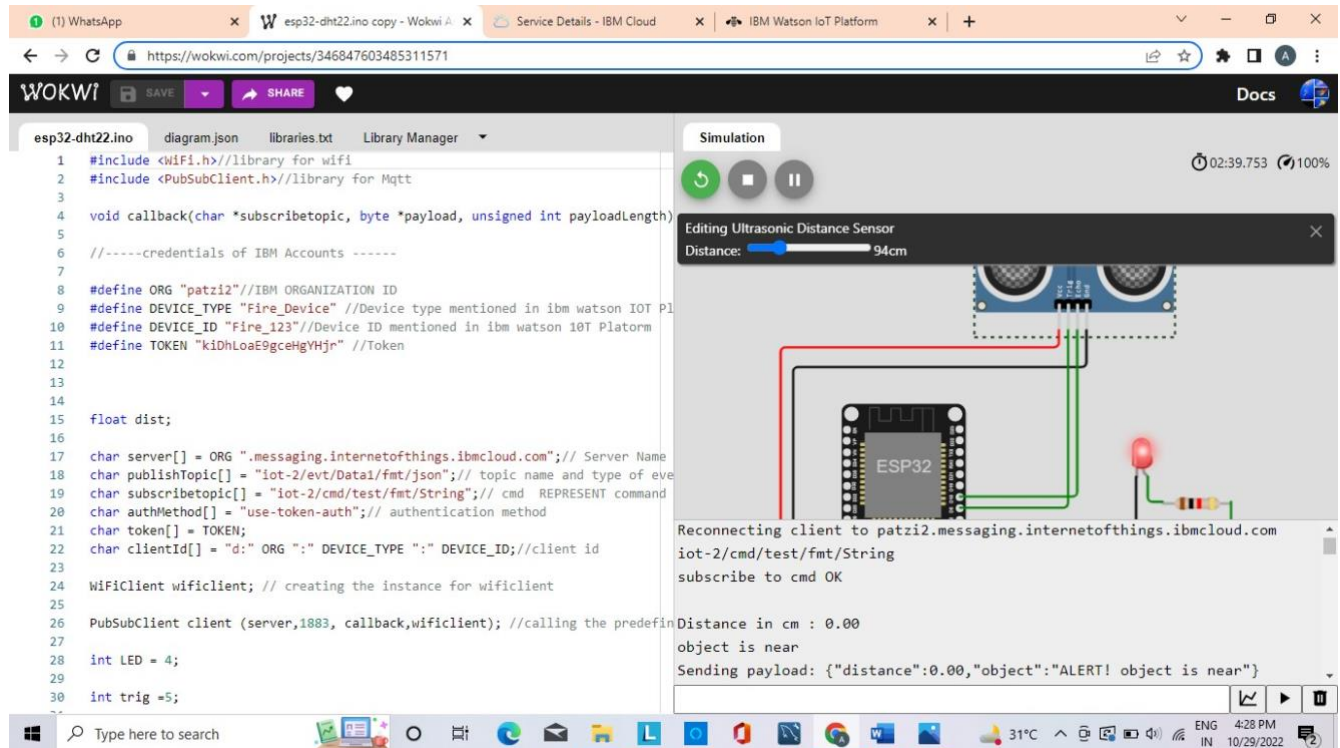


sketchino - Wok...html Show all

Type here to search 31°C ENG IN 5:01 PM 10/29/2022

OUTPUT:

When the distance is less than 100 cms, send an “alert” message to IBM Watson IOT Platform.



IBM CLOUD IMAGE



When the object is far (greater than 100 cms), send “no object found” to the IBM Watson IOT Platform.

The screenshot shows the Wokwi IDE interface. On the left, the code for `esp32-dht22.ino` is displayed. The code includes libraries for WiFi and MQTT, defines IBM Cloud credentials, and sets up an MQTT client to send distance data to the `iot-2/cmd/test/fmt/String` topic. The code also includes an LED and a trig pin. On the right, the simulation window shows an ESP32 board connected to an ultrasonic sensor. The sensor's distance is 157cm. The console output shows the following sequence of events:

```
Sending payload: {"distance":218.85,"object":"No object found"}
Publish ok
Distance in cm : 218.85
no object found
Sending payload: {"distance":218.85,"object":"No object found"}
Publish ok
Distance in cm : 218.85
```

IBM CLOUD IMAGE

The screenshot shows the IBM Cloud IoT Platform interface. The left sidebar contains navigation links for Connection Information, Recent Events, State, Device Information, Metadata, Diagnostics, Connection Logs, and Device Actions. The main area displays the 'Recent Events' section, which shows a list of events received from the device. The table below summarizes the data:

Event	Value	Format	Last Received
Data1	{"distance":218.85,"object":"No object found"}	json	a few seconds ago
Data1	{"distance":218.85,"object":"No object found"}	json	a few seconds ago
Data1	{"distance":218.85,"object":"No object found"}	json	a few seconds ago
Data1	{"distance":218.85,"object":"No object found"}	json	a few seconds ago
Data1	{"distance":218.85,"object":"No object found"}	json	a few seconds ago