

## Assignment - 4

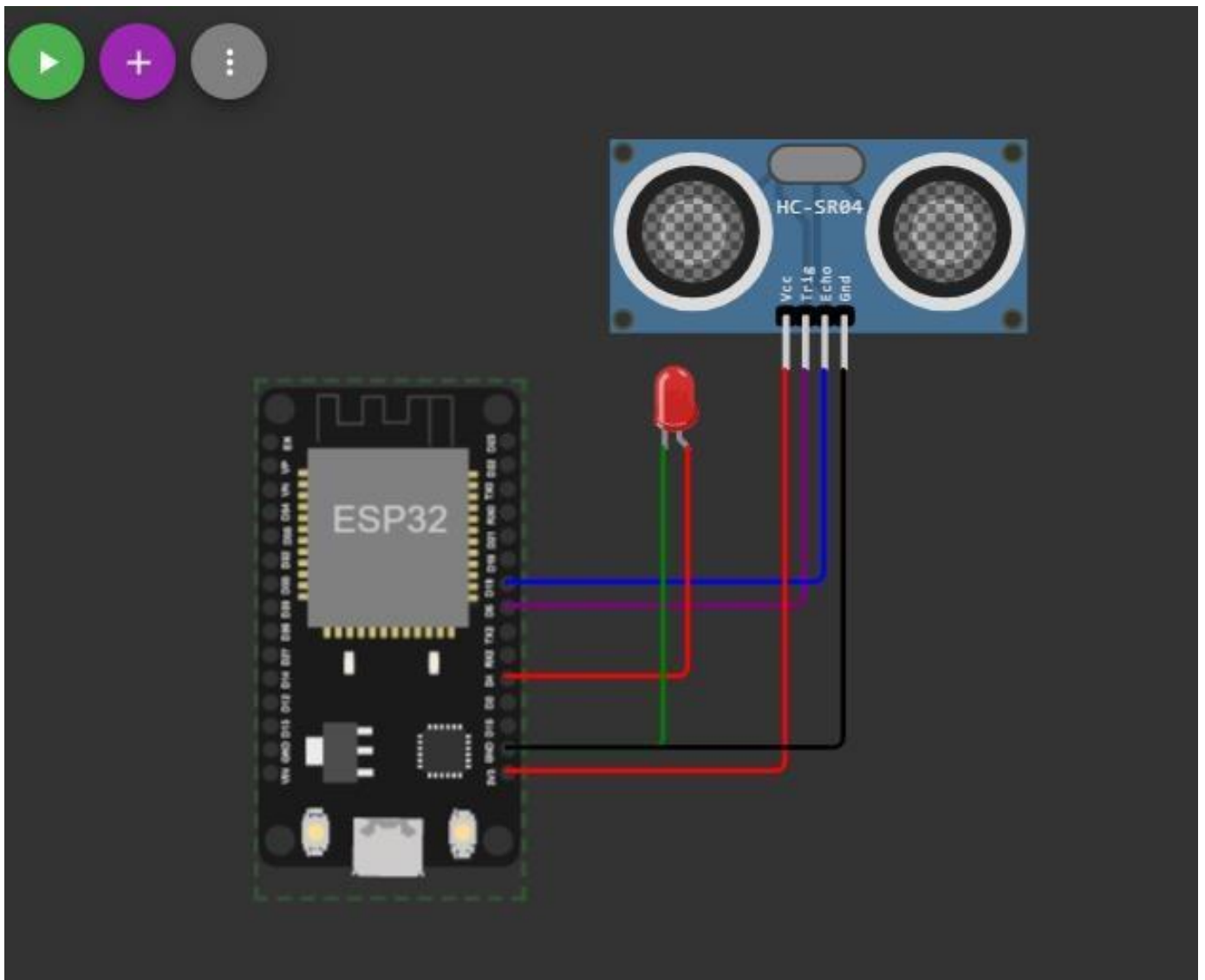
Assignment Date	16/11/2022
Student Name	K.Jaya surya
Student Roll Number	212219060140

### Question:

Write code and connections in WokWi for Ultrasonic Sensor. Whenever the distance is less 100cm, send "Alert" to IBM cloud and display in device recent events.

### Solution:

### Circuit Diagram:



### Code:

```

1  #include <WiFi.h>
2  #include <PubSubClient.h>
3  void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
4
5  //-----credentials of IBM Accounts-----
6
7  #define ORG "um5y3e"//IBM ORGANITION ID
8  #define DEVICE_TYPE "ESP32" //Device type mentioned in ibm watson IOT Platform
9  #define DEVICE_ID "ESP3240P"//Device ID mentioned in ibm watson IOT Platform
10 #define TOKEN "Sv*Ygwum-RMJXioBy?"//Token
11 String data3;
12 float dist;
13
14 //-----Customise the above values-----
15 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // Server Name
16 char publishTopic[] = "iot-2/evt/Data/fat/json"; // topic name and type of event perform and format in which data to be send
17 char subscribetopic[] = "iot-2/cmd/test/fmt/String"; // cmd REPRESENT command type AND COMMAND IS TEST OF FORMAT STRING
18 char authMethod[] = "use-token-auth"; // authentication method
19 char token[] = TOKEN;
20 char clientid[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
21
22
23 WiFiClient wificlient; // creating the instance for wificlient
24 PubSubClient client(server, 1883, callback,wificlient); //calling the predefined client id by passing parameter like server id, portand wificredential
25
26 int LED = 4;
27 int trig = 5;
28 int echo = 18;
29 void setup()
30 {
31     Serial.begin(115200);
32     pinMode(trig, OUTPUT);
33     pinMode(echo, INPUT);
34     pinMode(LED, OUTPUT);
35     delay(10);
36     wificlient.connect();

```

```

36     wificlient.connect();
37     mqttconnect();
38 }
39
40 void loop()// Recursive Function
41 {
42     digitalWrite(trig, LOW);
43     digitalWrite(trig,HIGH);
44     delayMicroseconds(10);
45     digitalWrite(trig, LOW);
46     float dur = pulseIn(echo,HIGH);
47     float dist = (dur * 0.0343)/2;
48     Serial.print ("Distance in cm");
49     Serial.println(dist);
50
51
52     PublishData(dist);
53     delay(1000);
54     if (!client.loop()) {
55         mqttconnect();
56     }
57 }
58
59
60 //..retrieving to cloud...
61
62 void PublishData(float dist) {
63     mqttconnect();//function call for connecting to ibm
64     //Creating the string in the form of Json to update the data to ibm cloud
65
66     String object;
67     if (dist <100)
68     {
69         digitalWrite(LED, HIGH);
70         Serial.println("object is near");

```

```

70     Serial.println("object is near");
71     object = "Near";
72 }
73 else
74 {
75     digitalWrite(LED, LOW);
76     Serial.println("no object found");
77     object = "No";
78 }
79
80 String payload="{\"distance\": ";
81 payload += dist;
82 payload += "," "object\": \"";
83 payload += object;
84 payload += "\"}";
85
86 Serial.print("Sending payload: ");
87 Serial.println(payload);
88 if (client.publish(publishTopic, (char*) payload.c_str())) {
89     Serial.println("Publish ok");// if it successfully upload data on the cloud then it will print publish ok in Serial monitor or else it will print publish failm
90 }
91 else
92 {
93     Serial.println("Publish failed");
94 }
95 }
96
97 void mqttconnect() {
98     if (!client.connected()) {
99         Serial.print("Reconnecting client to ");
100         Serial.print(server);
101         while (!client.connect(clientid, authMethod, token)) {
102             Serial.print(".");
103             delay(500);
104         }

```

```

104     }
105     initManagedDevice();
106     Serial.println();
107 }
108 }
109
110 void wificonnect() //function defination for wificonnect
111 {
112     Serial.println();
113     Serial.print("Connecting to ");
114     WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
115     while (WiFi.status() != WL_CONNECTED) {
116         delay(500);
117         Serial.print(".");
118     }
119     Serial.println("");
120     Serial.println("Wifi connected");
121     Serial.print("IP address: ");
122     Serial.println(WiFi.localIP());
123 }
124
125 void initManagedDevice() {
126     if (client.subscribe(subscribetopic)) {
127         Serial.println((subscribetopic));
128         Serial.println("subscribe to cmd OK");
129     } else {
130         Serial.println("subscribe to cmd FAILED");
131     }
132 }
133
134 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength){
135     Serial.print("callback invoked for topic: ");
136     Serial.println(subscribetopic);
137     for (int i=0; i < payloadLength; i++) {
138         //Serial.print((char)payload[i]);

```

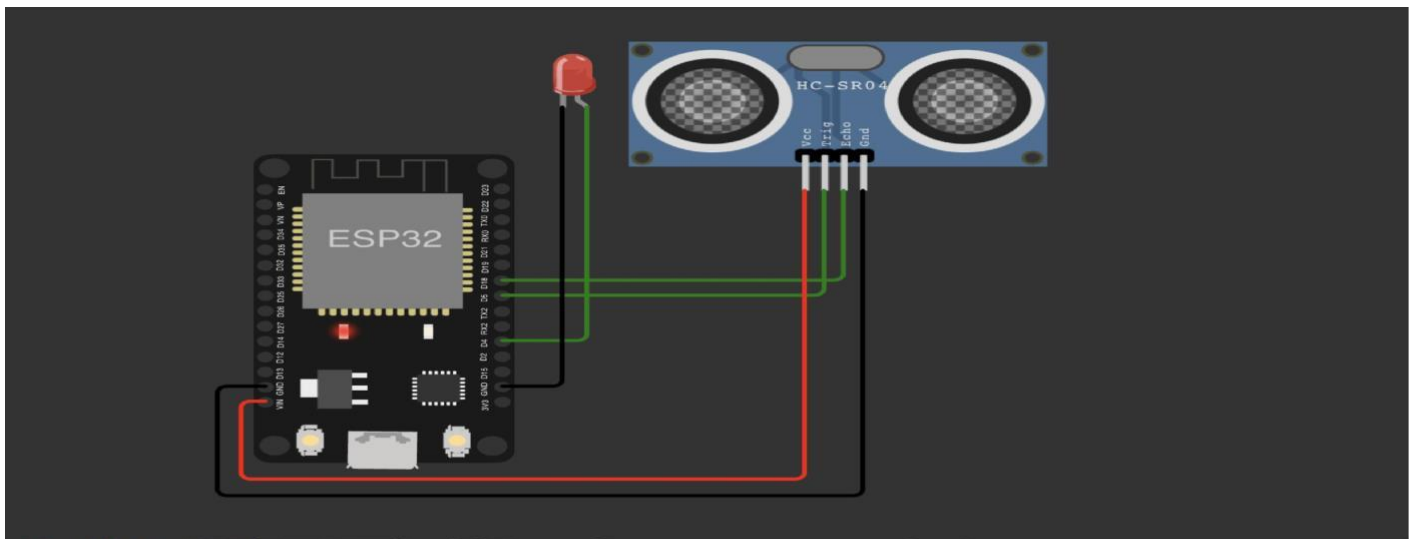
```

134 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength){
135     Serial.print("callback invoked for topic: ");
136     Serial.println(subscribetopic);
137     for (int i=0; i < payloadLength; i++) {
138         //Serial.print((char)payload[i]);
139         data3 += (char)payload[i];
140     }
141     data3 = "";
142 }

```

**OUTPUT:**

## When the Object is Far:



```
no object found
Sending payload: {"distance":141.21,"object":"No"}
Publish ok
Distancein cm141.21
no object found
Sending payload: {"distance":141.21,"object":"No"}
Publish ok
```

Interface showing device status and recent events for the DISTANCEDETECT device.

Device: DISTANCEDETECT, Status: Disconnected, Type: ULTRASON, Device: Device, Oct 20, 2022 9:46 AM

Recent Events:

Event	Value	Format	Last Received
Data	{"distance":141.21,"object":"No"}	json	a few seconds ago
Data	{"distance":141.21,"object":"No"}	json	a few seconds ago
Data	{"distance":141.21,"object":"No"}	json	a few seconds ago
Data	{"distance":141.18,"object":"No"}	json	a few seconds ago
Data	{"distance":141.2,"object":"No"}	json	a few seconds ago

## When the Object is Near:

