

## ASSIGNMENT-4

### DISTANCE DETECTION USING ULTRASONIC SENSOR

Date	02 November 2022
Team ID	PNT2022TMID43127
Project Name	Smart Waste Management System for Metropolitan Cities
Marks	2 Marks

#### Question 1:

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100 centimeters it should send "alert" to IBM cloud and display in device recent events.

#### CODE:

```
1  #include <WiFi.h> //library for wifi
2  #include <Pubsubclient.h> //library for Hott
3  void callback(char subscribetopic, byte payload, unsigned int payloadLength);
4
5
6  #define ORG 4h0jp //IBM ORGANITION IO
7  #define DEVICE_TYPE "ULTRASON
8  #define DEVICE_ID "DISTANCEDETECT
9  #define TOKEN "wuo5s7PR)ZSegvk&Rx"
10 String data3;
11 float dist;
12 char server[] = ".messaging.internetofthings.ibmcloud.com"; // Server Name
13 char publishTopic[] = "iot-2/evata/fmt/json";
14 char authMethod[] = "use-token-auth"; // authentication method
15 char token[] = TOKEN;
16 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //client id
17
18 int LED = 4;
19 int trig = 5;
20 int echo = 18;
21 void setic()
22 {
23   Serial.begin(115200);
24 }
```

esp32-blink.ino • diagram.json • libraries.txt • Library Manager ▼

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

```

70 | | creating the String in in form JSON to update the data to ibm cloud
71 */
72 String object;
73 if (dist <100)
74 {
75     digitalWrite(LED,HIGH);
76     Serial.println("object is near");
77     object = "Near";
78 }
79 else
80 {
81     digitalWrite(LED,LOW);
82     Serial.println("no object found");
83     object = "No";
84 }
85
86 String payload = "{\"distance\":";
87 payload += dist;
88 payload += ", \"object\":";
89 payload += object;
90 payload += "\"}";
91
92
93 Serial.print("Sending payload: ");
94 Serial.println(payload);
95
96
97
98

```

```

esp32-blink.ino • diagram.json • libraries.txt • Library Manager
98
99 if (client.publish(publishTopic, (char*) payload.c_str())) {
100     Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will print publish ok in Serial monitor or else it will print publish failed
101 } else {
102     Serial.println("Publish failed");
103 }
104
105 }
106 void mqttconnect() {
107     if (!client.connected()) {
108         Serial.print("Reconnecting client to ");
109         Serial.println(server);
110         while (!client.connect(clientId, authMethod, token)) {
111             Serial.print(".");
112             delay(500);
113         }
114
115         initManagedDevice();
116         Serial.println();
117     }
118 }
119 void wificonnect() //Function defination for wificonnect
120 {
121     Serial.println();
122     Serial.print("Connecting to ");
123
124     WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
125     while (WiFi.status() != WL_CONNECTED) {
126         delay(500);
127         Serial.print(".");
128     }
129     Serial.println("");
130     Serial.println("Wifi connected");
131     Serial.println("IP address: ");
132     Serial.println(WiFi.localIP());

```

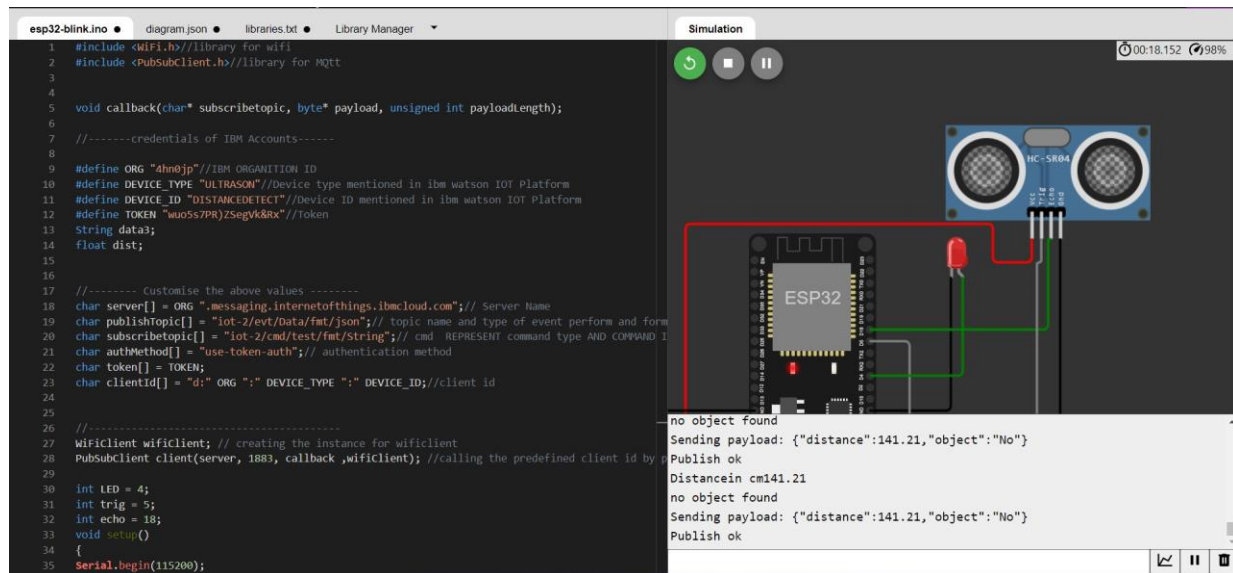
esp32-blink.ino • diagram.json • libraries.txt • Library Manager ▾

```
123
124   WiFi.begin("Wokwi-GUEST", "", 6); //passing the wifi credentials to establish the connection
125   while (WiFi.status() != WL_CONNECTED) {
126       delay(500);
127       Serial.print(".");
128   }
129   Serial.println("");
130   Serial.println("WiFi connected");
131   Serial.println("IP address: ");
132   Serial.println(WiFi.localIP());
133 }
134
135 void initManagedDevice() {
136     if (client.subscribe(subscribetopic)) {
137         Serial.println(subscribetopic);
138         Serial.println("subscribe to cmd OK");
139     } else {
140         Serial.println("subscribe to cmd FAILED");
141     }
142 }
143
144 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
145 {
146
147     Serial.print("callback invoked for topic: ");
148     Serial.println(subscribetopic);
149     for (int i = 0; i < payloadLength; i++) {
150         //Serial.print((char)payload[i]);
151         data3 += (char)payload[i];
152     }
153
154     // Serial.println("data: "+ data3);
155     // if(data3=="Near")
156     // {
157     // Serial.println(data3);
158     // Serial.println("Near detected");
159 }
```

esp32-blink.ino • diagram.json • libraries.txt • Library Manager ▼

```
142 }
143
144 void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
145 {
146     Serial.print("callback invoked for topic: ");
147     Serial.println(subscribetopic);
148     for (int i = 0; i < payloadLength; i++) {
149         //Serial.print((char)payload[i]);
150         data3 += (char)payload[i];
151     }
152
153     // Serial.println("data: "+ data3);
154     // if(data3=="Near")
155     // {
156     //     Serial.println(data3);
157     //     digitalWrite(LED,HIGH);
158     // }
159
160     // else
161     // {
162     //     Serial.println(data3);
163     //     digitalWrite(LED,LOW);
164     // }
165     data3="";
166
167 }
168
169
170
171 }
```

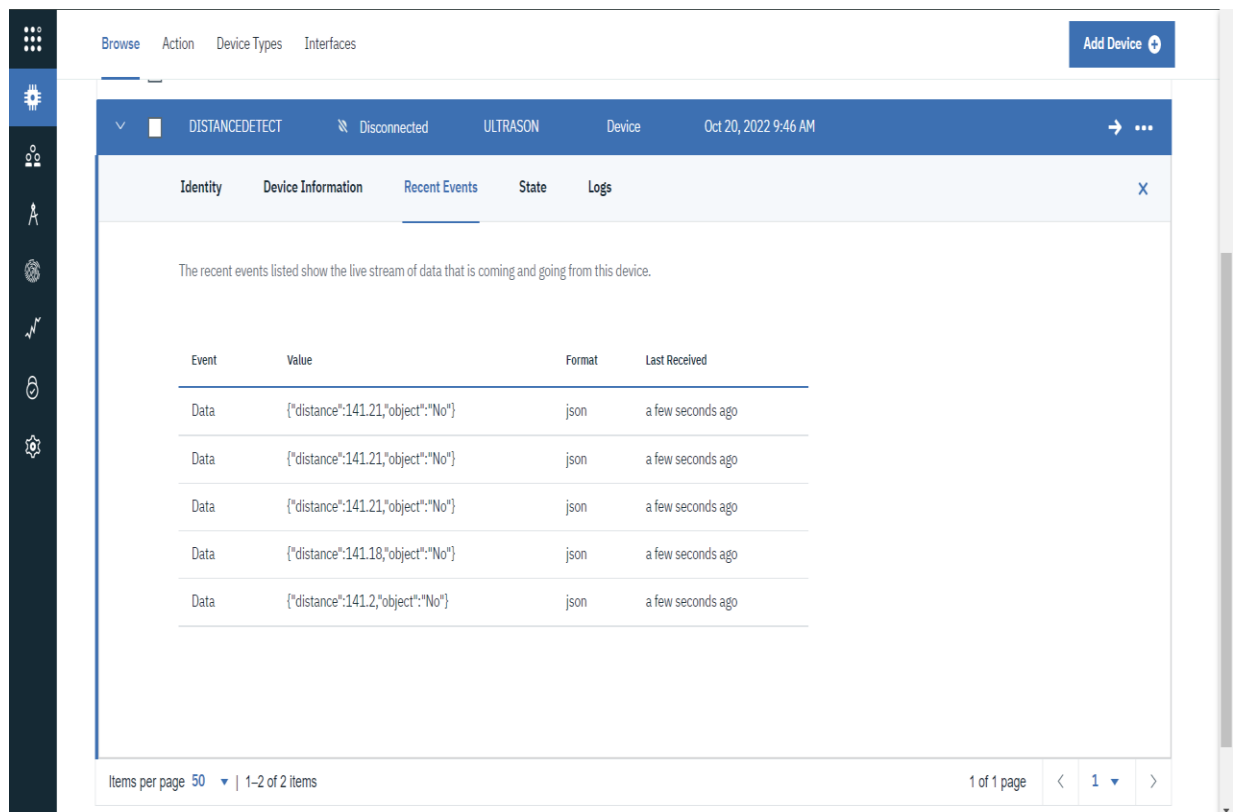
## OUTPUT:



The screenshot shows the Arduino IDE interface with a simulation window. The code on the left defines an ESP32 device connected to an HC-SR04 ultrasonic sensor. The simulation window on the right displays a circuit diagram of the ESP32 and the sensor. Below the diagram, the serial monitor output shows the following sequence of events:

```
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
```

Data send to the IBM cloud device when the object is far



The screenshot shows 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. The main content area displays a table of device data for a device named 'DISTANCEDETECT'.

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

At the bottom of the table, there is a pagination bar showing 'Items per page 50' and '1-2 of 2 items'. The bottom right corner indicates '1 of 1 page'.

when object is near to the ultrasonic sensor

wokwi.com/projects/305566932847821378

esp32-arduino.ino by urish

Simulation

object is near  
Sending payload: {"distance":97.82,"object":"Near"}  
Publish ok  
Distancein cm97.82  
object is near  
Sending payload: {"distance":97.82,"object":"Near"}  
Publish ok

Data sent to the IBM Cloud Device when the object is near

DistanceDetect

Disconnected

ULTRASON

Device

Oct 20, 2022 9:46 AM

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
Data	{"distance":79.66,"object":"Near"}	json	a few seconds ago
Data	{"distance":79.64,"object":"Near"}	json	a few seconds ago
Data	{"distance":79.66,"object":"Near"}	json	a few seconds ago
Data	{"distance":79.64,"object":"Near"}	json	a few seconds ago
Data	{"distance":79.66,"object":"Near"}	json	a few seconds ago

Items per page 50 | 1-2 of 2 items

1 of 1 page