

Assignment -4
Python Programming

Assignment Date	30 October2022
Student Name	Sajetha M
Student Reg Number	710719106302
Maximum marks	2 Marks

Question-1:

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.

Solution:

← → ↻ wokwi.com/projects/346566226034557523

WOKWI SAVE SHARE Library Manager Docs

sketch.ino diagram.json libraries.txt Library Manager

```
1 #include<WiFi.h> //library for wifi
2 #include<PubSubClient.h> //library for MQTT
3 void callback(char* subscribetopic, byte* payload, unsigned int payloadlength);
4 //-----credentials of IBM Account-----
5 #define ORG "Izzy60" // IBM ORGANIZATION ID
6 #define DEVICE_TYPE "iotedeviceproject" //DEVICE TYPE MENTIONED IN IOT WATSON PLATFORM
7 #define DEVICE_ID "229714" //DEVICE ID MENTIONED IN IOT WATSON PLATFORM
8 #define TOKEN "24681012" //Token
9 String data3;
10 float dist;
11 //-----customize the above value-----
12 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; //server name
13 char publishstopic[] = "ultrasonic/evt/Data/fmt/json"; //topic name and type of event perform
14 //and format in which data to be send*/
15 char subscribetopic[] = "ultrasonic/cmd/test/fmt/String"; //cmd REPRESENT Command type and
16 //COMMAND IS TEST OF FORMAT STRING*/
17 char authMethod[] = "use-token-auth"; //authentication method
18 char token[] = TOKEN;
19 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //CLIENT ID
20 //-----
21 WiFiClient wifiClient; // creating an instance for wifiClient
22 PubSubClient client(server, 1883, callback, wifiClient); //calling the predefined client id
23 //by passing parameter like server id, port and wifi credential*/
24 int LED = 4;
25 int trig = 5;
26 int echo = 18;
27 void setup()
28 {
29   Serial.begin(115200);
30   pinMode(trig, OUTPUT);
```

Conn
ec
ti
ng
to

← → C wokwi.com/projects/346566226034557523

WOKWI SAVE SHARE Docs

sketch.ino diagram.json libraries.txt Library Manager

```
61 Serial.println("no object is near");
62 object="Near";
63 }
64 else
65 {
66   digitalWrite(LED,LOW);
67   Serial.println("no object found");
68   object="No";
69 }
70 String payload="{\"distance\": ";
71 payload +=dist;
72 payload +=", \"object\": \"";
73 payload += object;
74 payload += "\";";
75
76 Serial.print("Sending payload: ");
77 Serial.println(payload);
78 if(client.publish(publishtopic, (char*) payload.c_str())){
79   Serial.println("Publish ok");/* If its successfully upload data on the cloud then it will print
80   publish ok in serial monitor or else it will print publish failed*/
81 } else{
82   Serial.println("Publish failed");
83 }
84 }
85 void mqttconnect(){
86   if(!client.connected()){
87     Serial.print("Reconnecting client to ");
88     Serial.println(server);
89     while(!client.connect(clientid,authMethod, token)){
90       Serial.print(".");
91       delay(500);
```

← → C wokwi.com/projects/346566226034557523

WOKWI SAVE SHARE Docs

sketch.ino diagram.json libraries.txt Library Manager

```
92 }
93 initManagedDevice();
94 Serial.println();
95 }
96 }
97 void wificonnect();//function defenition for wificonnect
98 {
99   Serial.println();
100   Serial.print("Connecting to ");
101   WiFi.begin("Wokwi-GUEST", "",6);//PASSING THE WIFI CREDENTIALS TO ESTABLISH CONNECTION
102   while (WiFi.status() !=WL_CONNECTED){
103     delay(500);
104     Serial.print(".");
105   }
106   Serial.println("");
107   Serial.println("WiFi connected");
108   Serial.println("IP address");
109   Serial.println(WiFi.localIP());
110 }
111 void initManagedDevice(){
112   if(client.subscribe(subscribetopic)){
113     Serial.println((subscribetopic));
114     Serial.println("subscribe to cmd OK");
115   }else{
116     Serial.println("subscribe to cmd failed");
117   }
118 }
119 void callback(char* subscribetopic,byte*payload,unsigned int payloadLength)
120 {
121   Serial.print("callback invoked for topic: ");
122   Serial.println(subscribetopic);
```

```

123 for(int i=0; i< payloadlength; i++){
124   //Serial.print((char)payload[i]);
125   data3 +=(char)payload[i];
126 }
127 //Serial.println("data: "+ data3);
128 //if(data3=="Near")
129 //{
130 //Serial.println(data3);
131 //digitalWrite(LED,HIGH);
132 //}
133 //else
134 //{
135 //Serial.println(data3);
136 //digitalWrite(LED,LOW);
137 //}
138 data3="";
139 }

```

OUTPUT:
DATA IS SENT TO IBM CLOUD WHEN NO OBJECT IS DETECTED

Event	Value	Format	Last received
Data	Distance: 79.64/object: Near	json	4 hours, 50 minutes, 11 seconds ago
Data	Distance: 79.64/object: Near	json	4 hours, 50 minutes, 11 seconds ago
Data	Distance: 79.64/object: Near	json	4 hours, 50 minutes, 11 seconds ago
Data	Distance: 79.64/object: Near	json	4 hours, 50 minutes, 11 seconds ago
Data	Distance: 79.64/object: Near	json	4 hours, 50 minutes, 11 seconds ago

When no object is detected

DistanceDetect

Disconnected

ULTRASON

Device

Oct 20, 2022 9:46 AM

Identity

Device Information

Recent Events

State

Logs

Hardware: Add a new device to the system and connect it to the system.

Event	Value	Format	Last Received
Data	[{"distance": 97.82, "object": "Near"}]	json	A few seconds ago
Data	[{"distance": 97.82, "object": "Near"}]	json	A few seconds ago
Data	[{"distance": 97.82, "object": "Near"}]	json	A few seconds ago
Data	[{"distance": 97.82, "object": "Near"}]	json	A few seconds ago
Data	[{"distance": 97.82, "object": "Near"}]	json	A few seconds ago

Items per page: 50

1 of 2 items

1 of 1 page

When object is detected in ultrasonic detector

wokwi.com/projects/346572482591851092

WOKWI

SAVE

SHARE

Docs

sketch

Simulation

diagram

library

Libra

Mana

HC-SR04

ESP32

object is near

1 Sending payload: {"distance":97.82,"object":"Near"}

1 Publish ok

1 Distance in cm 97.82

2 object is near

2 Sending payload: {"distance":97.82,"object":"Near"}

2 Publish ok