

Assignment -4

Python Programming

Assignment Date	26 October 2022
Student Name	Vidyalakshmi.S
Student Roll Number	951219106086
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 devicerecentevents.

Upload document with WOKWi share link and images of IBM cloud.

Solution:

```
sketch.ino | diagram.json | libraries.txt | Library Manager | Simu
1 #include<WiFi.h> //library for wifi
2 #include<PubSubClient.h> //library for MQTT
3 void callback(char* topic, byte* payload, unsigned int payloadlength);
4 //-----credentials of IBM Account-----
5 #define ORG "iyy6o" // IBM ORGANIZATION ID
6 #define DEVICE_TYPE "iotdeviceproject" //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 publishTopic[] = "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 tupe 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,portand wifiCredential*/
24 int LED = 4;
25 int trig = 5;
26 int echo = 10;
27 void setup()
28 {
29   Serial.begin(115200);
30   pinMode(trig, OUTPUT);
```

```

92     }
93     initManagedDevice();
94     Serial.println();
95   }
96 }
97 void wificonnect()//function definition 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);

```



Co
nn
ec
ti
ng
to

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

```
123   for(int i=0; i< payloadLength; i++){
124       //Serial.print((char)payload[i]);
125       data3 +=(char)payload[i];
126   }
127   //Serial.println("dta: "+ 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

WOKWI

SAVE

SHARE

Docs

sketch: Simulation

diagram

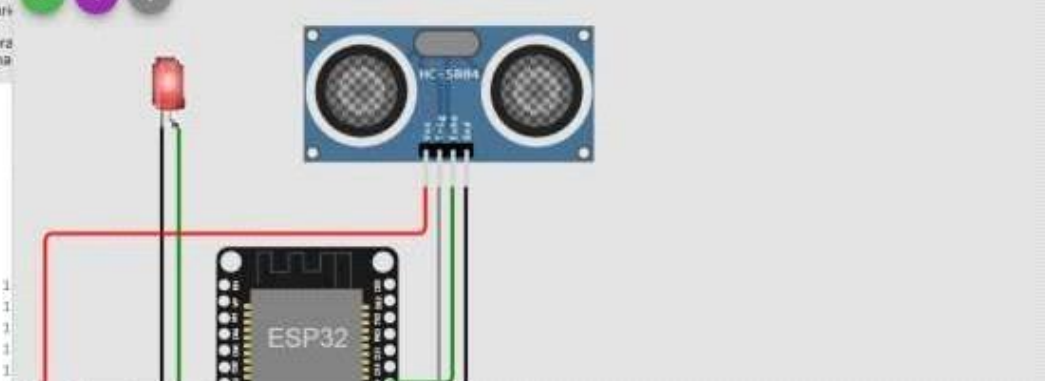
+

⋮

library

Libraries

Manager



The diagram shows an ESP32 microcontroller connected to an HC-SR04 ultrasonic sensor and a red LED. The sensor's VCC pin is connected to the ESP32's VCC pin, and its GND pin is connected to the ESP32's GND pin. The sensor's Trig pin is connected to the ESP32's pin 4, and its Echo pin is connected to the ESP32's pin 5. The red LED's anode is connected to the ESP32's pin 2, and its cathode is connected to the ESP32's GND pin.

```
1 object is near
1
1 Sending payload: {"distance":97.82,"object":"Near"}
1 Publish ok
1 Distance in cm 97.82
2
2 object is near
2
2 Sending payload: {"distance":97.82,"object":"Near"}
2 Publish ok
2
2
```

✓

||