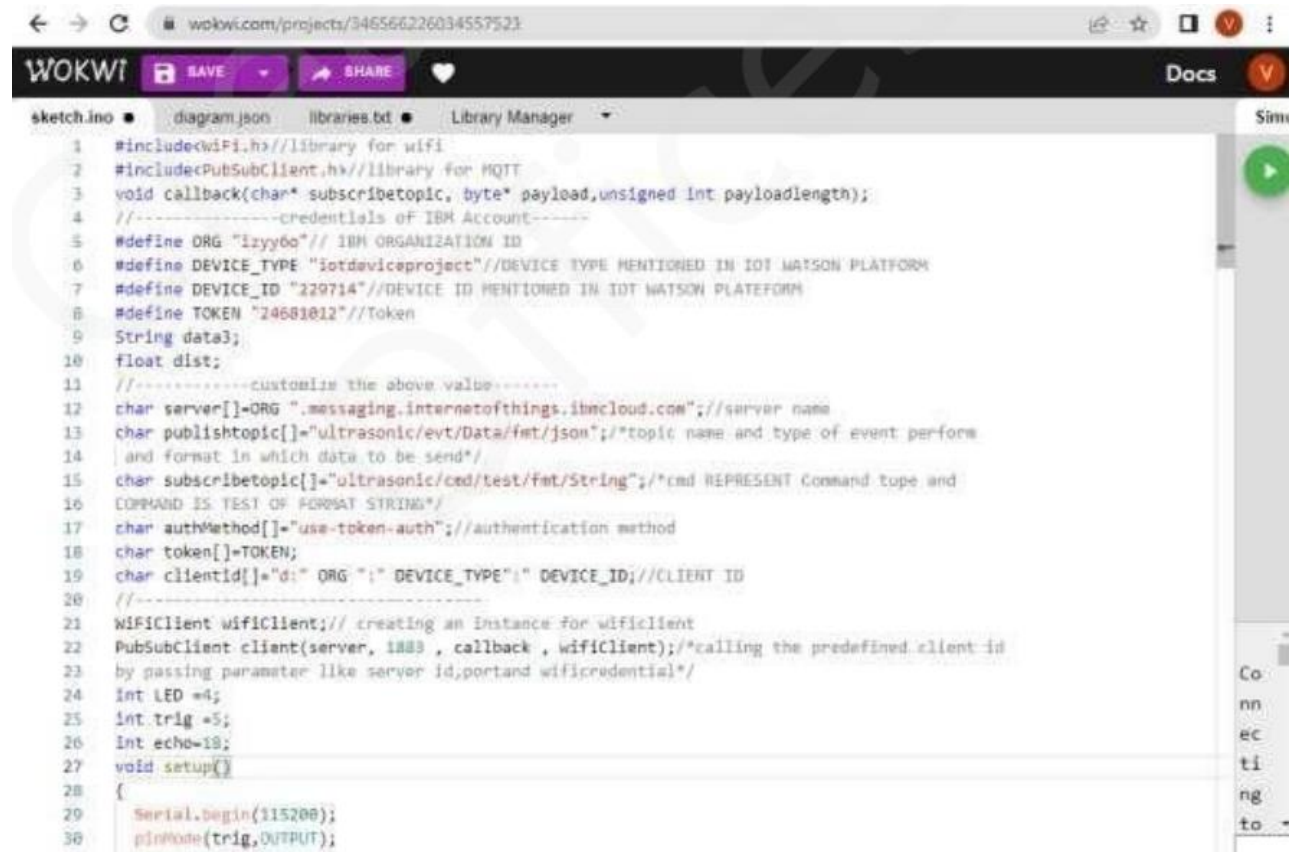


Question-1:

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:



```
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 "izyy6o" // 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 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 wificlient*/
24 int LED = 4;
25 int trig = 5;
26 int echo = 19;
27 void setup()
28 {
29   Serial.begin(115200);
30   pinMode(trig, OUTPUT);
```

← → 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 definition for wificonnect
98 {
99 Serial.println();
100 Serial.print("Connecting to ");
101 WiFi.begin("Wokwi.GUEST", "",0);//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);

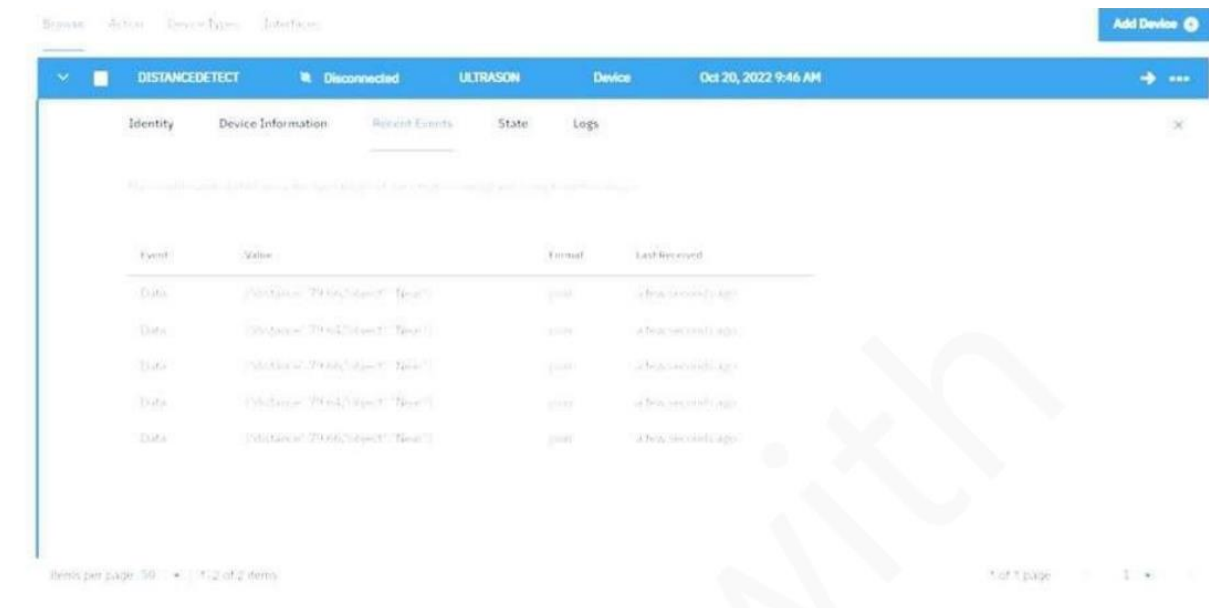
Simu
Co
nn
ec
ti
ng
to

OUTPUT:

DATA IS SENT TO IBM CLOUD WHEN NO OBJECT IS DETECTED

Event	Value	Format	Last Received
Data	No Object Detected	json	14 hrs, 56 mins, 46 sec ago
Data	No Object Detected	json	14 hrs, 56 mins, 46 sec ago
Data	No Object Detected	json	14 hrs, 56 mins, 46 sec ago
Data	No Object Detected	json	14 hrs, 56 mins, 46 sec ago
Data	No Object Detected	json	14 hrs, 56 mins, 46 sec ago

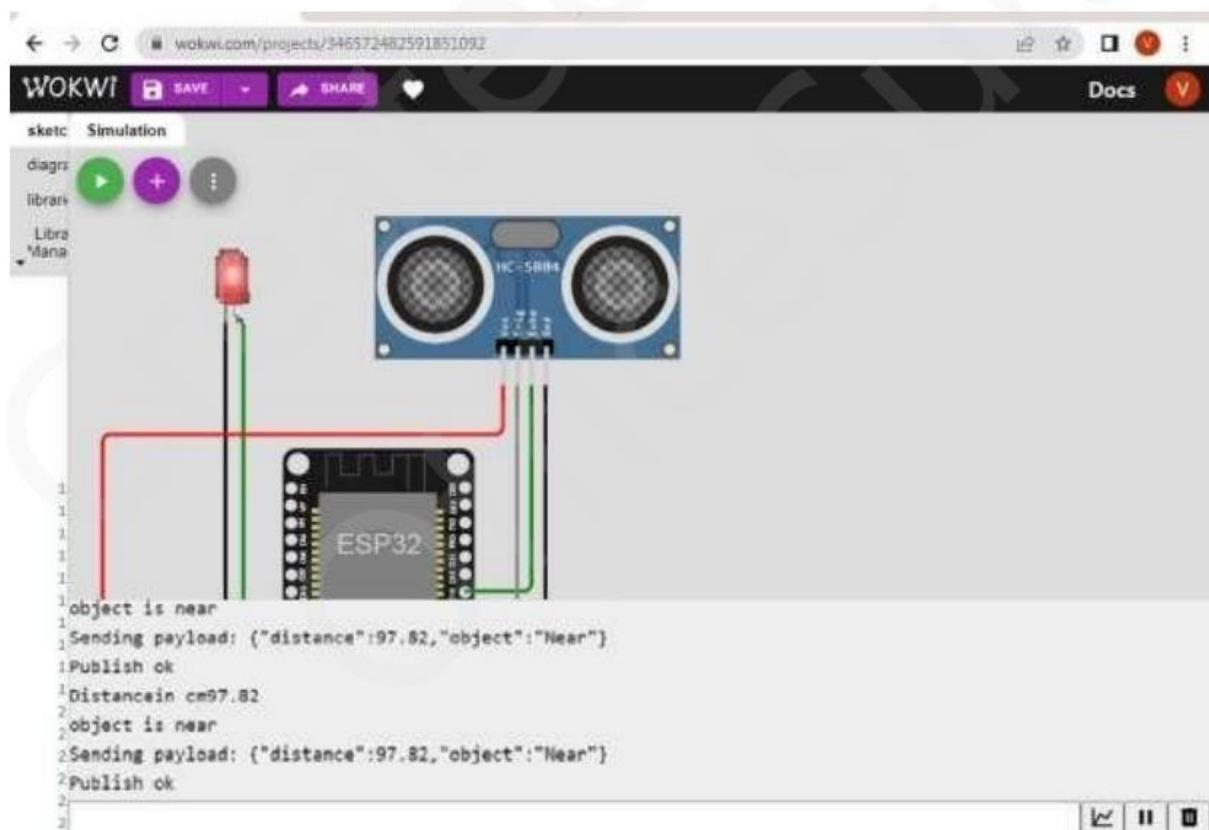
When no object is detected



The screenshot shows a web interface for a device named 'DISTANCEDetect'. The device is currently 'Disconnected'. The interface has tabs for 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' tab is active, displaying a table of events. The table has columns for 'Event', 'Value', 'Format', and 'Last Received'. There are five events listed, all with the value 'Distance of 29462 Subject: "Near"' and the format 'json'. The 'Last Received' column shows timestamps like '4 hrs, 26 mins, 41 secs ago'.

Event	Value	Format	Last Received
Data	Distance of 29462 Subject: "Near"	json	4 hrs, 26 mins, 41 secs ago
Data	Distance of 29462 Subject: "Near"	json	4 hrs, 26 mins, 41 secs ago
Data	Distance of 29462 Subject: "Near"	json	4 hrs, 26 mins, 41 secs ago
Data	Distance of 29462 Subject: "Near"	json	4 hrs, 26 mins, 41 secs ago
Data	Distance of 29462 Subject: "Near"	json	4 hrs, 26 mins, 41 secs ago

When object is detected in ultrasonic detector



The screenshot shows a Wokwi simulation of an ESP32 microcontroller connected to an HC-SR04 ultrasonic sensor. The sensor is connected to the ESP32 via a breadboard. The simulation is running, and the console output shows the following messages:

```
1 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
```