Sethu Institute Of Technology - Kariapatti

ASSIGNMENT -4

NAME : Sri Harini DEPT :
CSE

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.

```
#include <WiFi.h> #include <PubSubClient.h> void callback(char*
subscribetopic, byte* payload, unsigned int payloadLength);
//-------credentials of IBM Accounts------
#define ORG "jesccj"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32_Controller"//Device type mentioned in ibm watson IOT
Platform
#define DEVICE_ID "BME280_Sensor"//Device ID mentioned in ibm watson IOT
Platform
#define TOKEN "KImxM@H69iIBD&JYZC" //Token String data3; char
server[] = ORG ".messaging.internetofthings.ibmcloud.com"; char
publishTopic[] = "iot-2/evt/Data/fmt/json"; char subscribetopic[]
= "iot-2/cmd/test/fmt/String"; char authMethod[] = "use-token-
auth"; char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback ,wifiClient);
const int trigPin = 5; const int echoPin = 18; #define
SOUND_SPEED 0.034 long duration; float distance; void setup()
{ Serial.begin(115200);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT); wificonnect();
mqttconnect();
} void loop() {
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW); duration
```

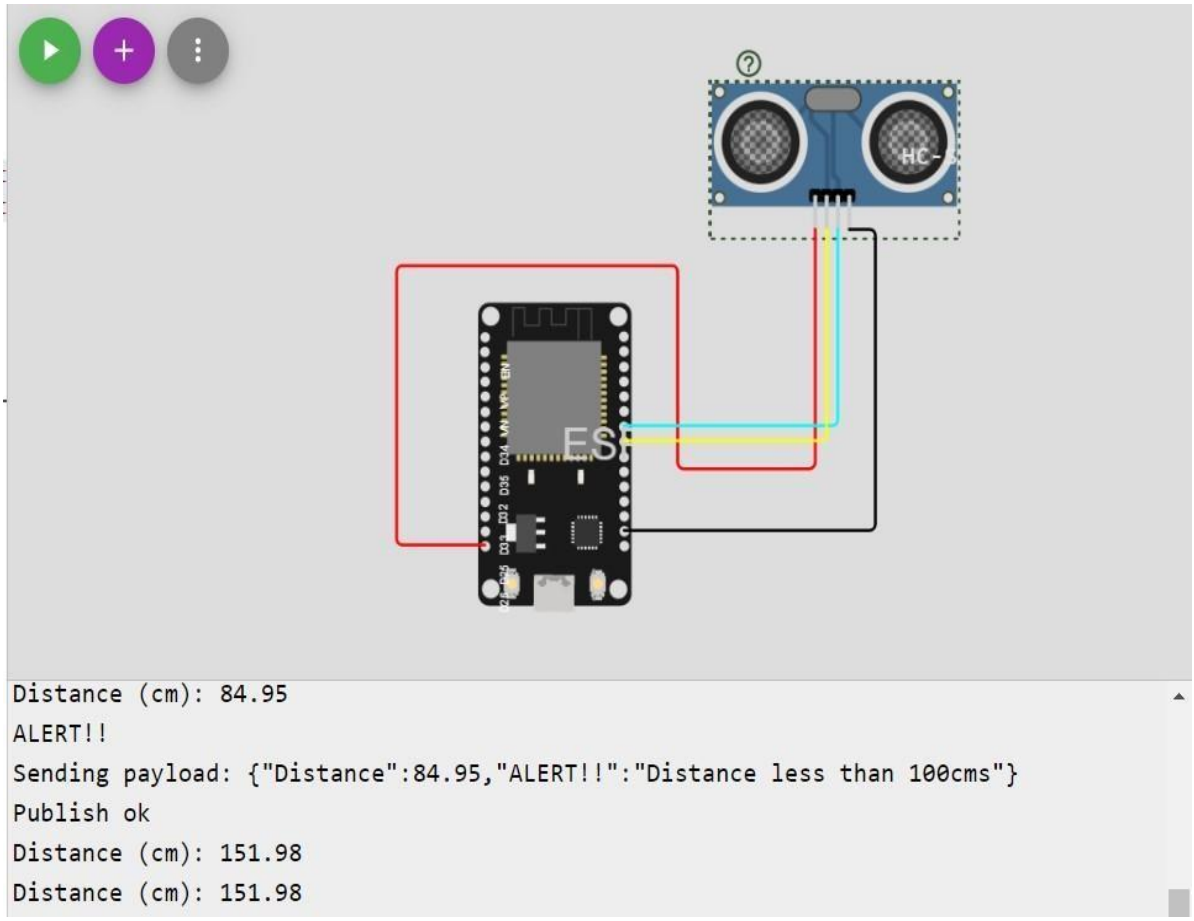
```
= pulseIn(echoPin, HIGH); distance =
duration * SOUND_SPEED/2;
Serial.print("Distance (cm): ");
Serial.println(distance);
if(distance<100)
{
Serial.println("ALERT!!");
delay(1000);
PublishData(distance);
delay(1000); if (!client.loop())
{ mqttconnect();
} } delay(1000); } void
PublishData(float dist) {
mqttconnect();
String payload = "{\"Distance\":";
payload += dist;
payload += ",\"ALERT!!\":""\"Distance less than 100cms\"";
payload += "}";
Serial.print("Sending payload: ");
Serial.println(payload);

if (client.publish(publishTopic, (char*) payload.c_str())) {
Serial.println("Publish ok");
} else {
Serial.println("Publish failed");
} } void mqttconnect() { if
(!client.connected()) {
Serial.print("Reconnecting client to ");
Serial.println(server); while
(!!!client.connect(clientId, authMethod, token)) {
Serial.print("."); delay(500); }  initManagedDevice();
Serial.println();  }
} void
wificonnect()
{
Serial.println();
Serial.print("Connecting     to     ");
WiFi.begin("Wokwi-GUEST", "", 6); while
(WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
} void initManagedDevice()
{  if
(client.subscribe(subscribetopic)) {
Serial.println((subscribetopic));
```

```
Serial.println("subscribe to cmd OK");
} else {
Serial.println("subscribe to cmd FAILED");
} } void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength) {
Serial.print("callback invoked for topic: ");
Serial.println(subscribetopic); for (int i =
0; i < payloadLength; i++) {
//Serial.print((char)payload[i]); data3 +=
(char)payload[i];
}
Serial.println("data: "+ data3);
data3="";
}
```

DIAGRAM.JSON

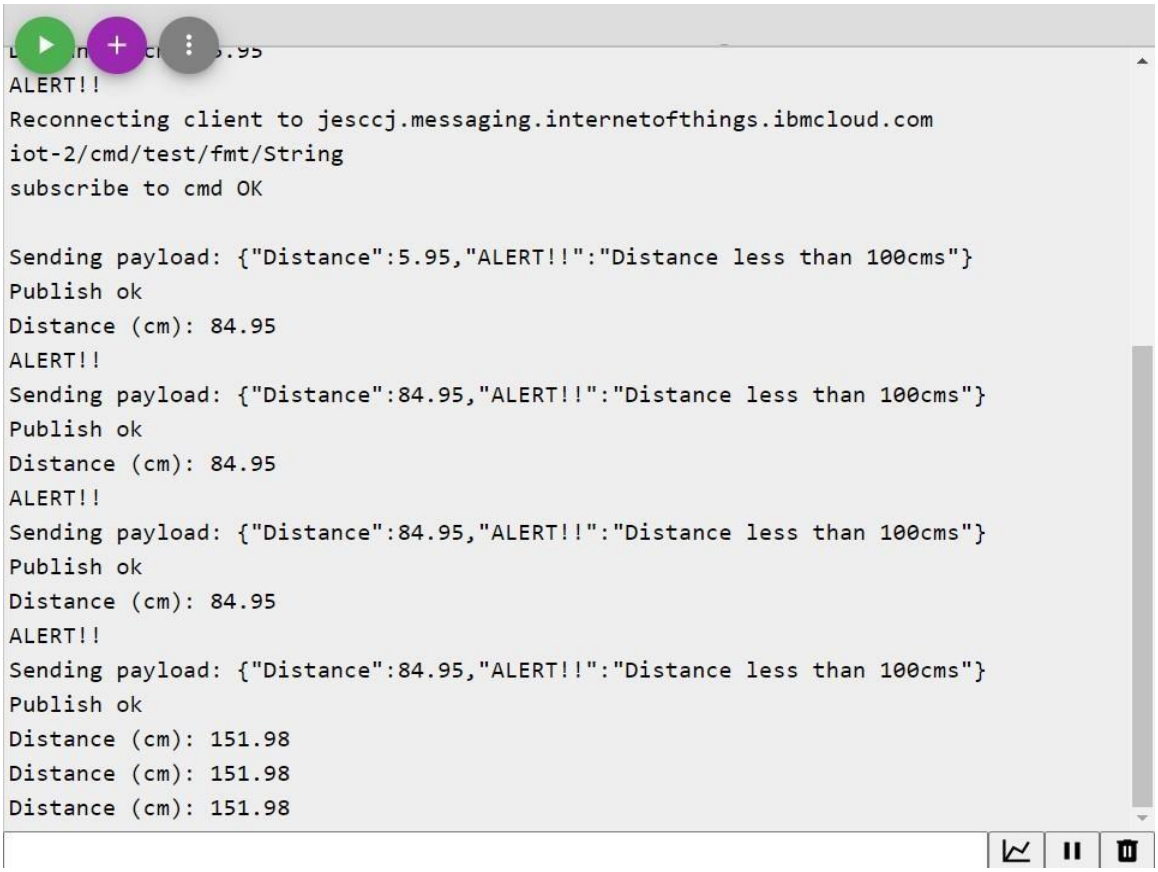
```
{
  "version": 1,
  "author": "Anonymous maker",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -106.69, "left": 1567.63,
"attrs": {} },
    { "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -245.94, "left":
1729.41, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [
      "esp:VIN",
      "ultrasonic1:VCC",
      "red",
      [ "h-60.86", "v-179.91", "h194.33", "v130.8", "h91.18" ]
    ],
    [ "esp:D18", "ultrasonic1:ECHO", "cyan", [ "h0" ] ],
    [ "esp:D5", "ultrasonic1:TRIG", "yellow", [ "h0" ] ],
    [ "esp:GND.1", "ultrasonic1:GND", "black", [ "h173.94", "v-9.72" ] ]
  ] }
```

DIAGRAM:

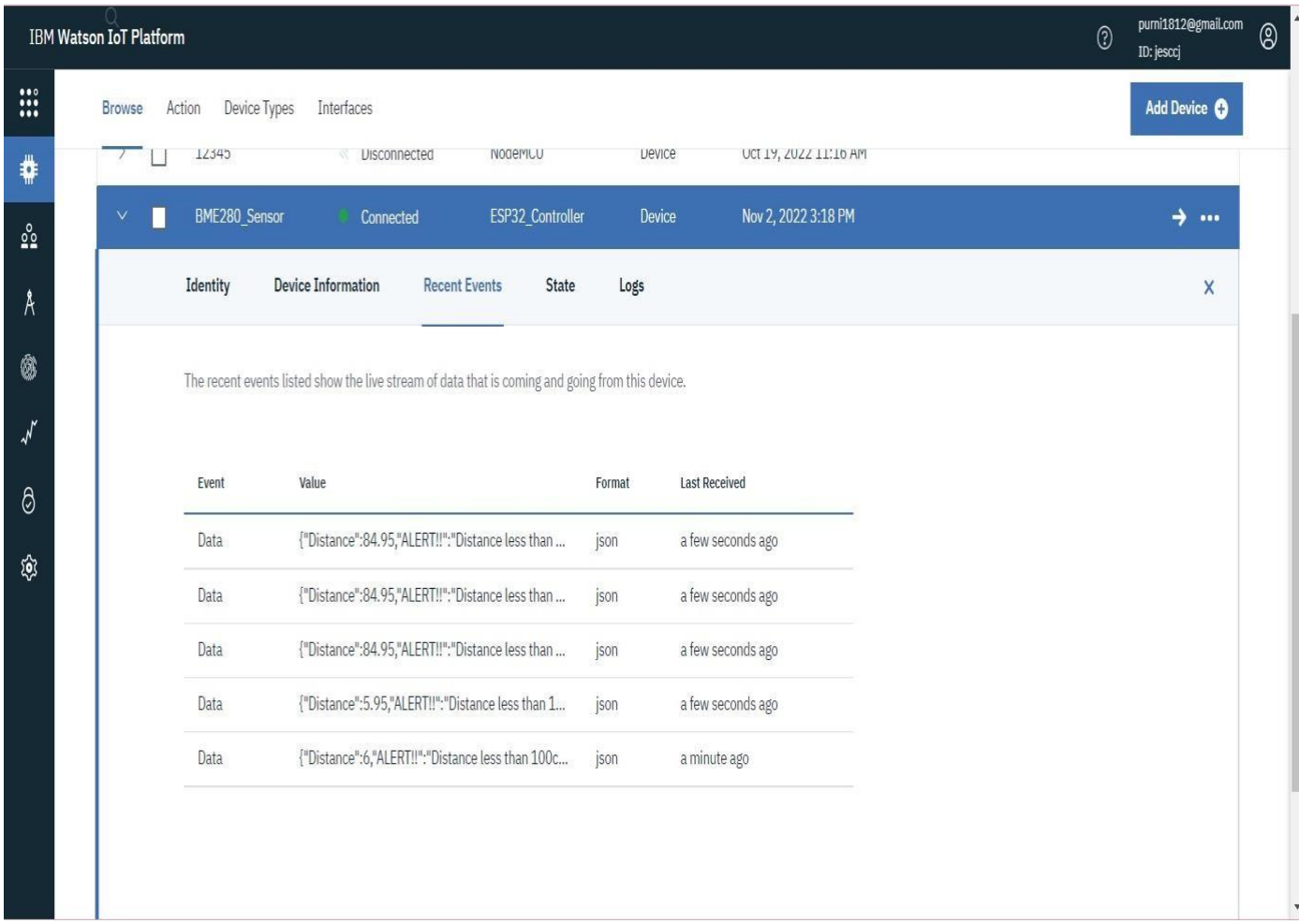


WOKWI OUTPUT:


Simulation
Connecting to ...
WiFi connected
IP address:
10.10.0.2
Reconnecting client to jesccj.messaging.internetofthings.ibmcloud.com
iot-2/cmd/test/fmt/String
subscribe to cmd OK
Distance (cm): 6.00
ALERT!!
Sending payload: {"Distance":6.00,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 5.95
ALERT!!
Reconnecting client to jesccj.messaging.internetofthings.ibmcloud.com
iot-2/cmd/test/fmt/String
subscribe to cmd OK
Sending payload: {"Distance":5.95,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 84.95


ALERT!!
Reconnecting client to jesccj.messaging.internetofthings.ibmcloud.com
iot-2/cmd/test/fmt/String
subscribe to cmd OK
Sending payload: {"Distance":5.95,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 84.95
ALERT!!
Sending payload: {"Distance":84.95,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 84.95
ALERT!!
Sending payload: {"Distance":84.95,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 84.95
ALERT!!
Sending payload: {"Distance":84.95,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 151.98
Distance (cm): 151.98
Distance (cm): 151.98

IBM OUTPUT:



Wokwi simulation link:

https://wokwi.com/projects/347210401624097363