Assignment -4

Wokwi Programming

Assignment Date	31 October 2022
Student Name	A.SABNA BEGAM
Student Roll Number	815119106303
Maximum Marks	2 Marks

Question-1:

Write code and connections in wokwi for the ultrasonic sensor.

Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in

the device recent events.

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

PROGRAM

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);
//-------credentials of IBM Accounts------
#define ORG "rv07c6"//IBM ORGANITION ID
#define DEVICE_TYPE "distance_hcsr04"//Device type mentioned in ibm watson IOT
#define DEVICE_ID "6789"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "w_mwV+5NZn*W7Xt)qA" //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 = random(200);
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="";
}
```

JSONCODE:

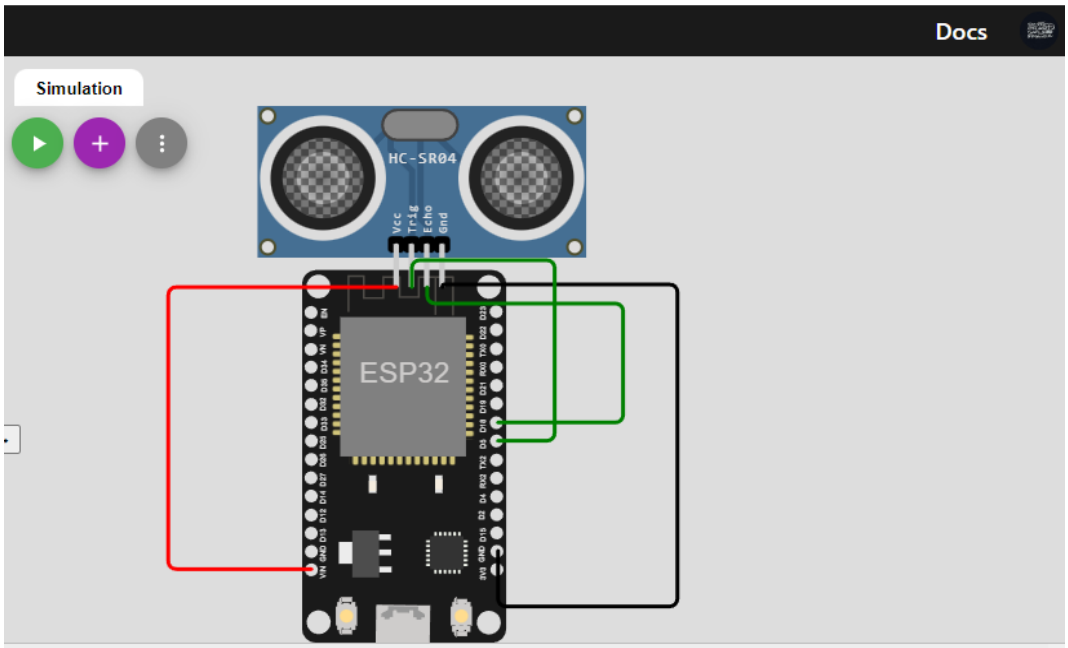
WOKWI SAVE SHARE

sketch.ino | diagram.json ● | libraries.txt | Library Manager

```
{
  "version": 1,
  "author": "K. Annapooraneshwari",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 56, "left": -13.34, "attrs": {} },
    { "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -27.7, "left": -36.17, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "ultrasonic1:GND", "esp:GND.1", "black", [ "v-1.43", "h121.22", "v168.67", "h-14" ] ],
    [ "esp:VIN", "ultrasonic1:VCC", "red", [ "h-73.83", "v-173.8" ] ],
    [ "ultrasonic1:ECHO", "esp:D18", "green", [ "v8.57", "h101.11", "v88" ] ],
    [ "esp:D5", "ultrasonic1:TRIG", "green", [ "h29.21", "v-94.6", "h-78" ] ]
  ]
}
```

sketch.ino - Wok....html

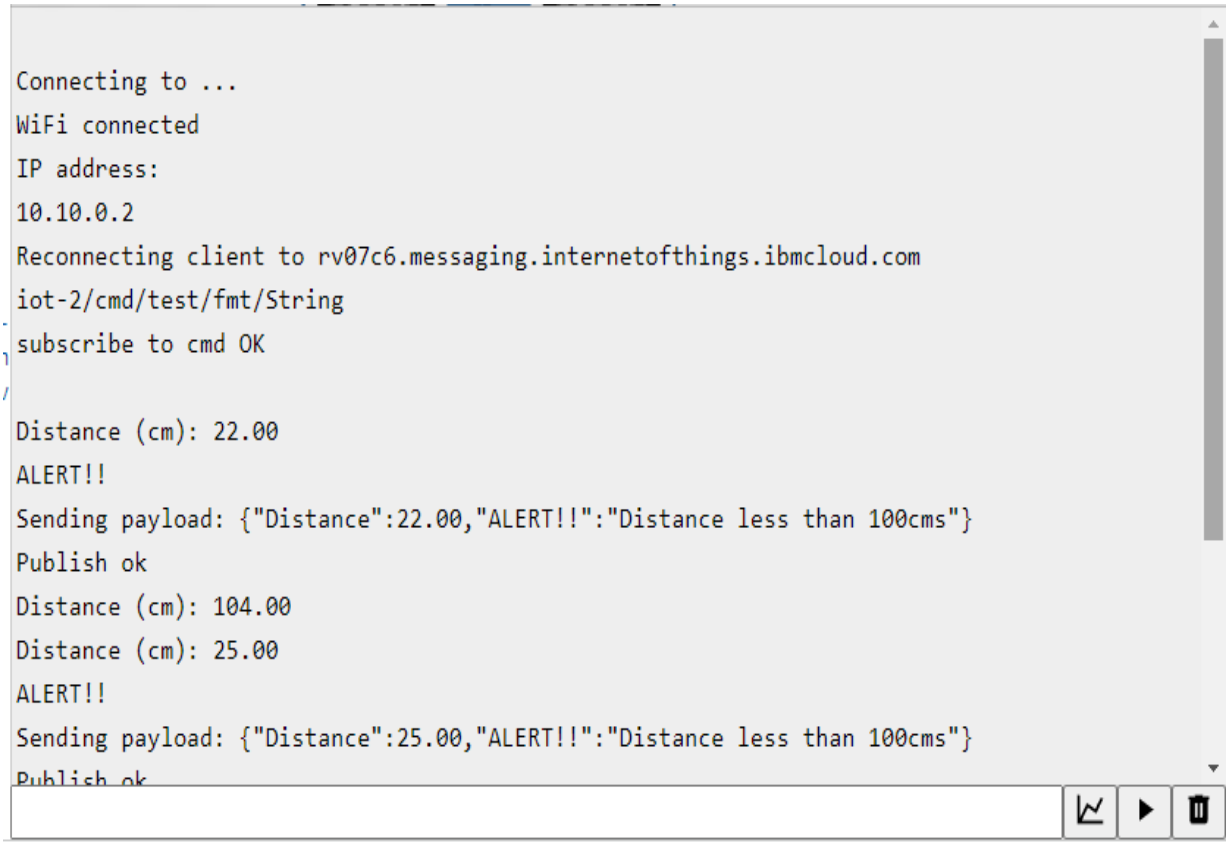
CIRCUIT DIAGRAM:

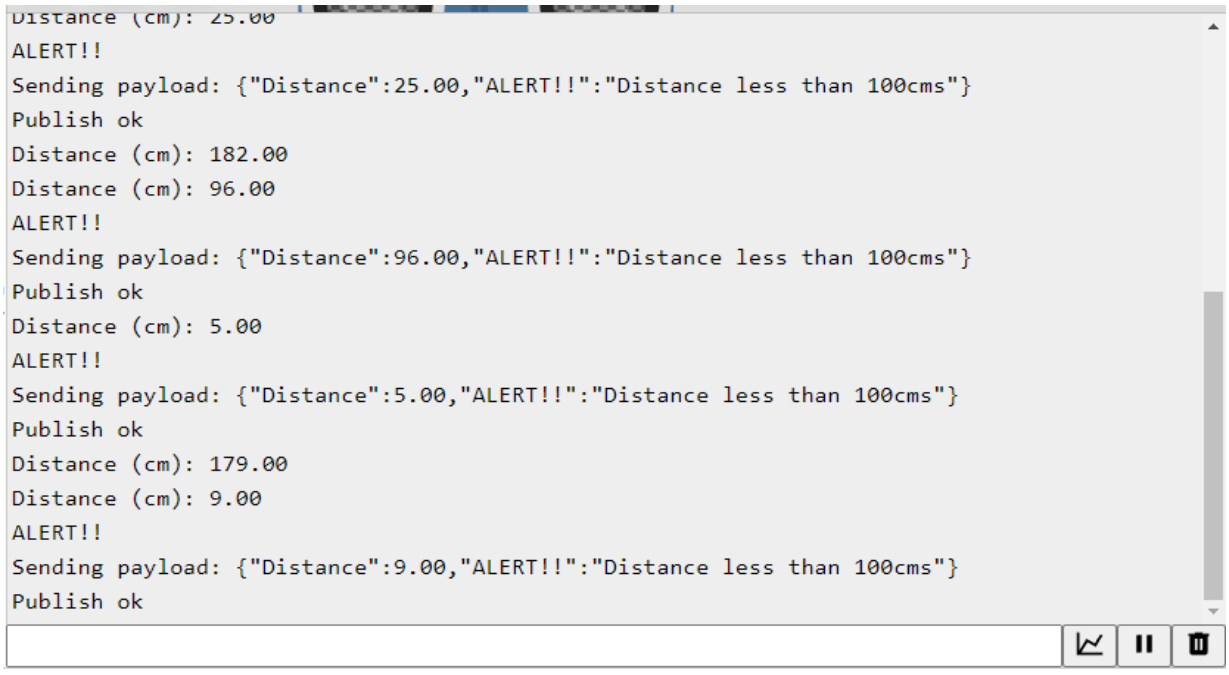


WOKWI URL:

https://wokwi.com/projects/347009925106369108

WOKWI OUTPUT:





IBM CLOUD OUTPUT: