Assignment-4

Write code and connections in wokwi for ultrasonic sensors. Whenever distance is less than 100 cms send "alert" to ibm cloud and display in device recent events.

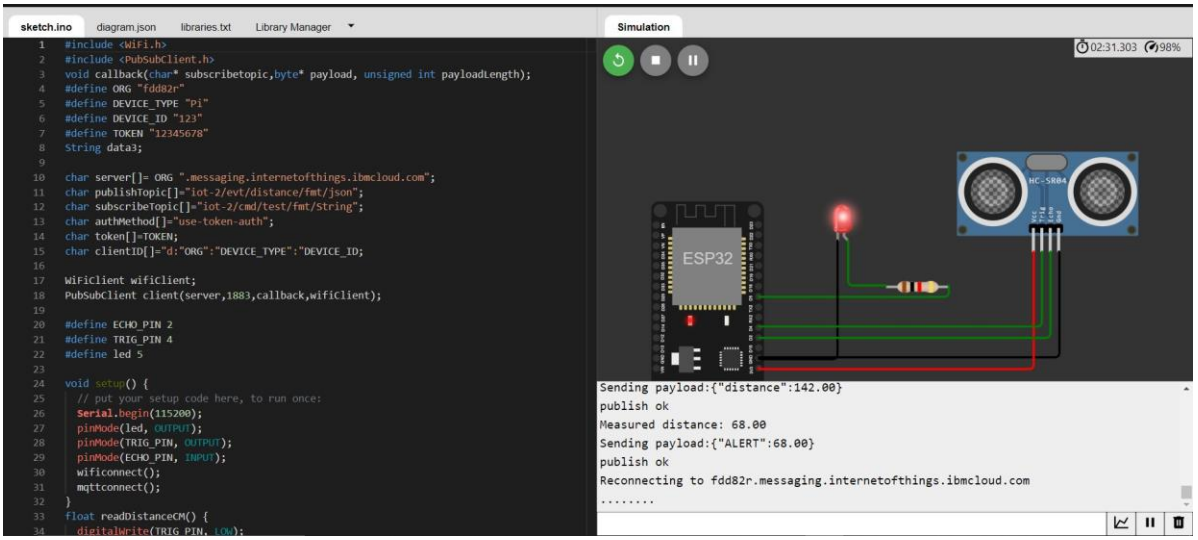
PROGRAM:

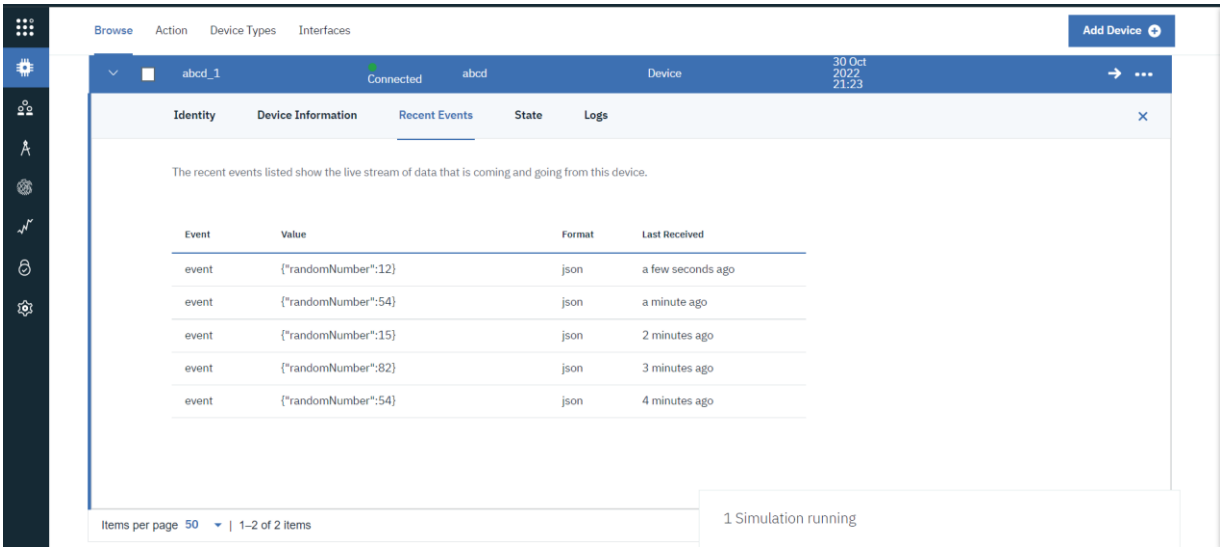
```
#include  <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribe topic,byte* payload, unsigned int payloadLength);
#define ORG "fdd82r"
#define DEVICE_TYPE "Pi"
#define DEVICE_ID "123"
#define TOKEN "12345678"
String data3;

char server[]= ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[]="iot-2/evt/distance/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);

#define ECHO_PIN 2
#define TRIG_PIN 4
#define led 5

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  wifi connect();
  mqtt connect();
}
float readDistanceCM() {
```

```
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration=random(1,200);
  //Serial.println(duration);
  //duration = pulseIn(ECHO_PIN, HIGH);
  return duration ;
  //Serial.println(duration);

}

void loop() {
  float distance = readDistanceCM();
  //Serial.println(distance);

  bool isNearby = distance < 100;
  digitalWrite(led, isNearby);

  Serial.print("Measured distance: ");
  Serial.println(distance);
  if(distance<100){
  PublishData2(distance);

  }else{
    PublishData1(distance);

  }
  //PublishData(distance);
  delay(1000);
  if(!client.loop()){
  mqttconnect();
  }

  //delay(2000);
}
void PublishData1(float dist){
  mqttconnect();
  String payload= "{\"distance\":";
  payload += dist;
  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 PublishData2(float dist){
  mqttconnect();
  String payload= "{\"ALERT\":";
  payload += dist;
  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 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++){
  data3 += (char)payload[i];
  }
  Serial.println("data:"+ data3);
  if(data3=="lighton"){
  Serial.println(data3);
  digitalWrite(led,HIGH);
  }else{
    Serial.println(data3);
    digitalWrite(led,LOW);
  }
  data3="";
}
```




Browse
Action
Device Types
Interfaces
Add Device
abcd_1
Connected
abcd
Device
30 Oct 2022 21:23
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
event
{"randomNumber":12}
json
a few seconds ago
event
{"randomNumber":54}
json
a minute ago
event
{"randomNumber":15}
json
2 minutes ago
event
{"randomNumber":82}
json
3 minutes ago
event
{"randomNumber":54}
json
4 minutes ago
Items per page 50
1–2 of 2 items
1 Simulation running