Name: Maheshsriram T (737819ITR045)

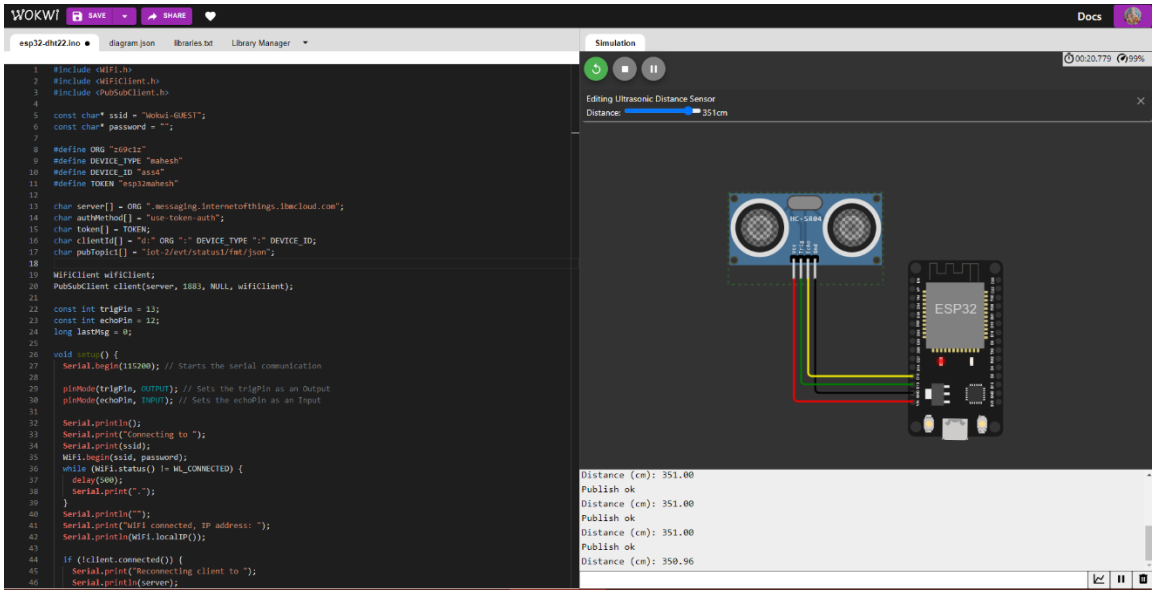
Code:

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <PubSubClient.h>
const char* ssid = "Wokwi-GUEST";
const char* password = "";
#define ORG "z69c1z"
#define DEVICE_TYPE "mahesh"
#define DEVICE_ID "ass4"
#define TOKEN "esp32mahesh"
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
char pubTopic1[] = "iot-2/evt/status1/fmt/json";
WiFiClient wifiClient;
PubSubClient client(server, 1883, NULL, wifiClient);
const int trigPin = 13;
const int echoPin = 12;
long lastMsg = 0;
void setup() {
  Serial.begin(115200); // Starts the serial communication
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
  Serial.println();
  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
```

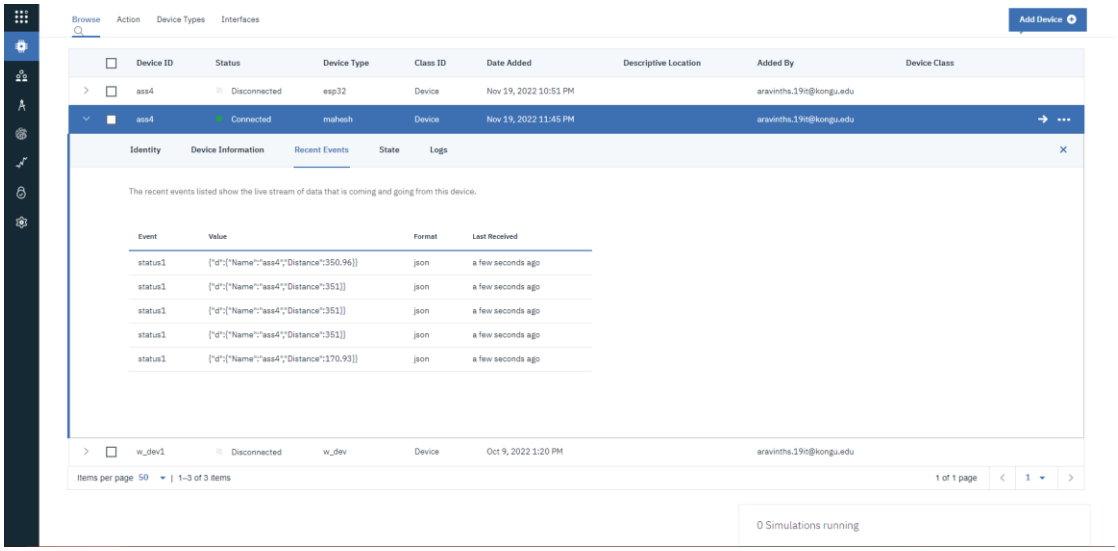
```
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("WiFi connected, IP address: ");
  Serial.println(WiFi.localIP());
  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }
    Serial.println("Bluemix connected");
  }
}
void loop() {
  // Clears the trigPin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  long duration = pulseIn(echoPin, HIGH);
  // Calculate the distance
  float distanceCm = duration * 0.034/2;
  // Prints the distance in the Serial Monitor
  Serial.print("Distance (cm): ");
  Serial.println(distanceCm);
  delay(3000);
```

```
client.loop();
String payload = "{\"d\":{\"Name\":\"" DEVICE_ID "\"";
        payload += ",\"Distance\":";
        payload += distanceCm;
        payload += "}}";
if (client.publish(pubTopic1, (char*) payload.c_str())) {
  Serial.println("Publish ok");
} else {
  Serial.println("Publish failed");
}
}
```

Wokwi Output:



IBM output:



Wokwi link: https://wokwi.com/projects/348780939335172692