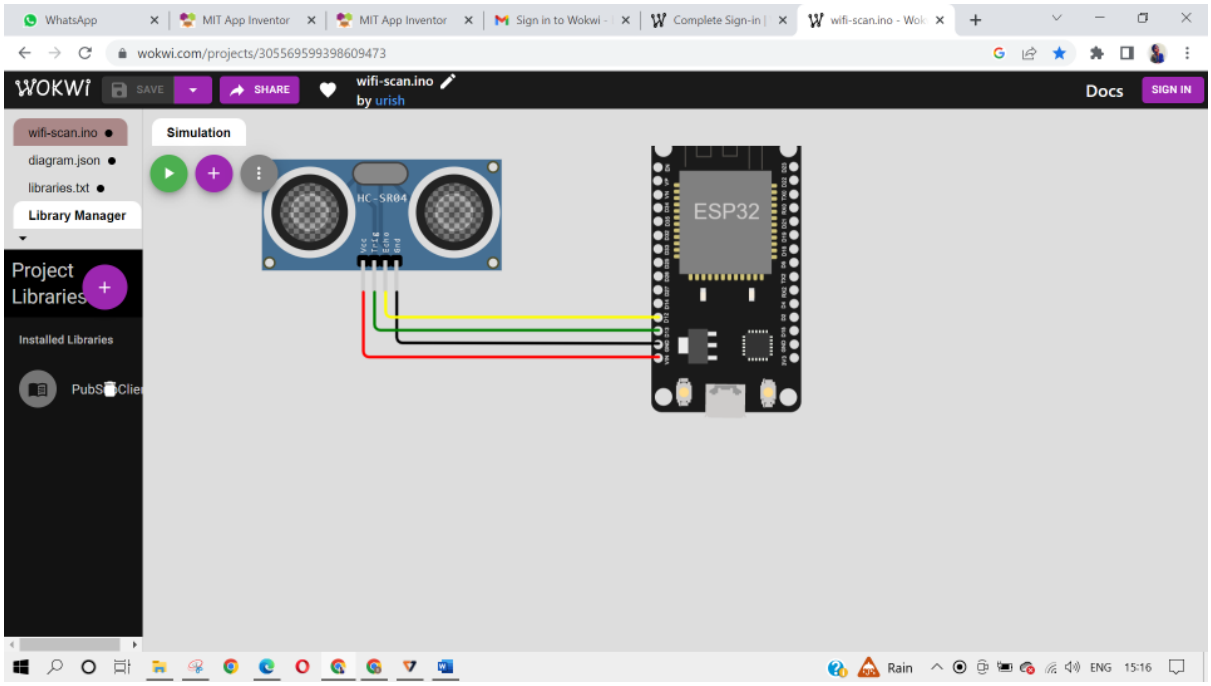
ASSIGNMENT – 4

Write a code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100 cm and send an “alert” to the IBM cloud and display in the device recent events.

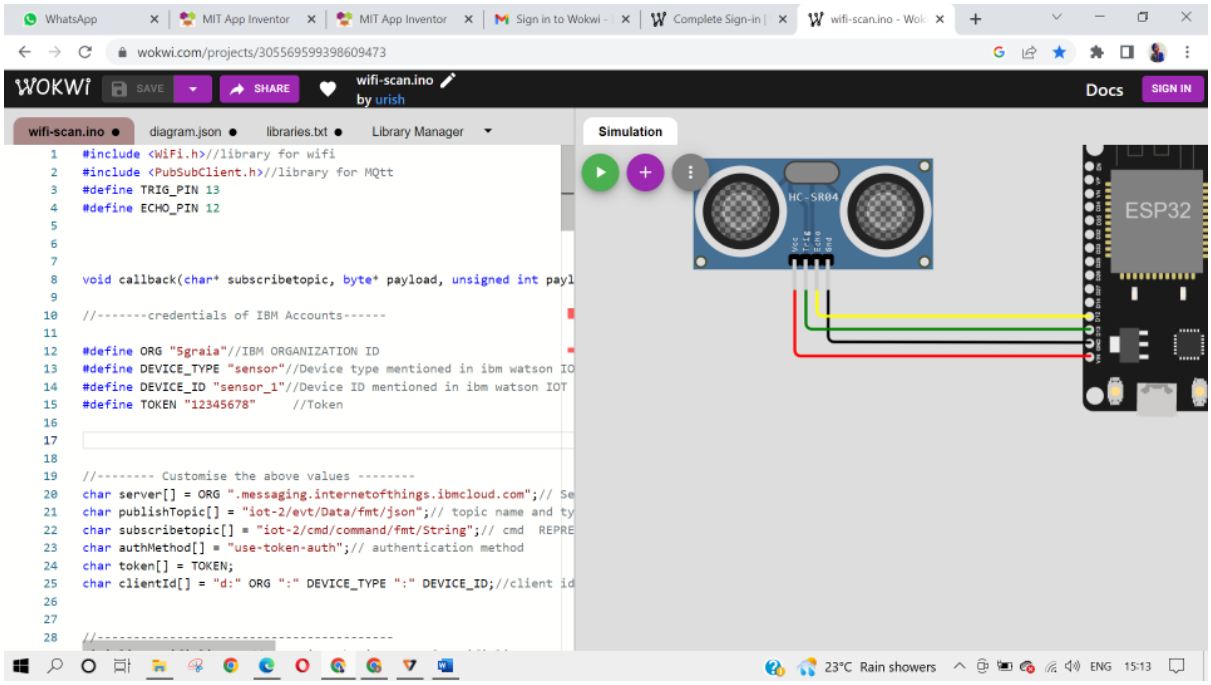
CIRCUIT DIAGRAM:

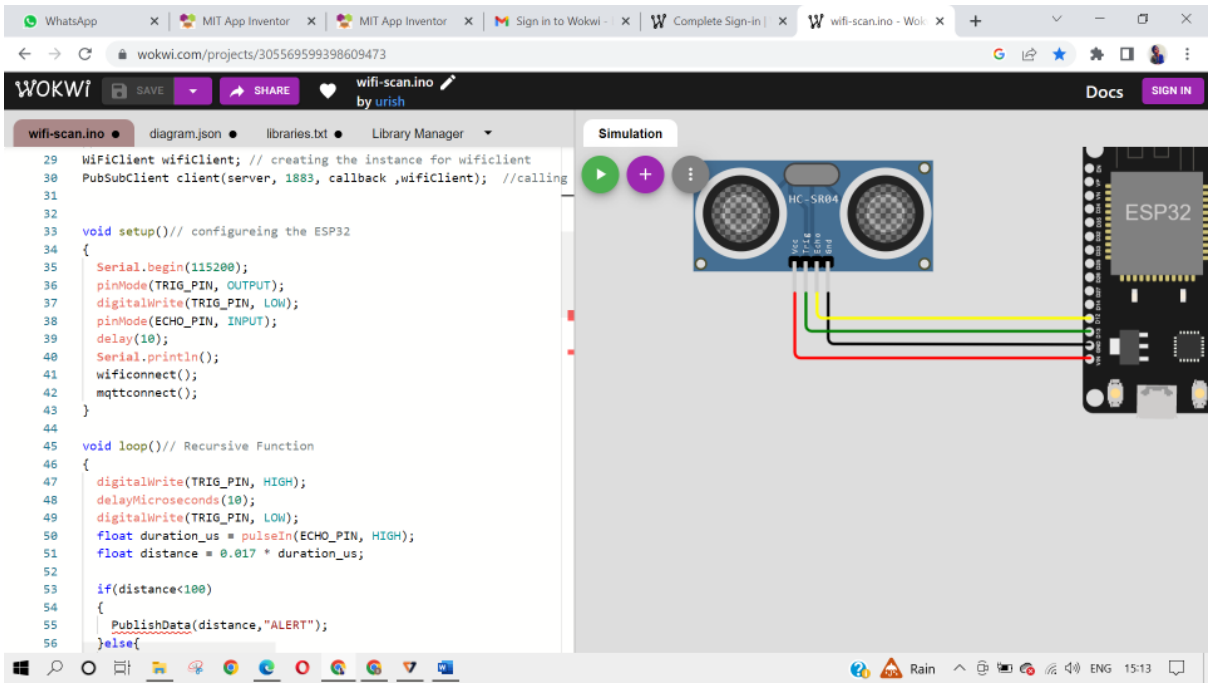


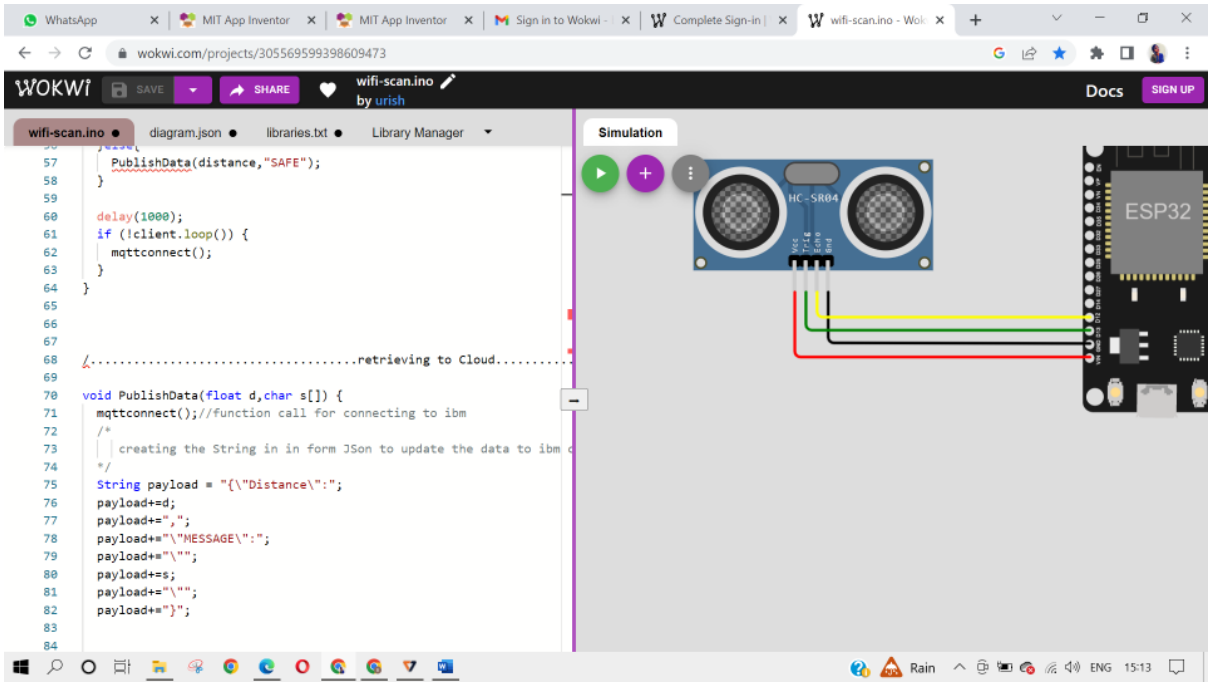
LINK:

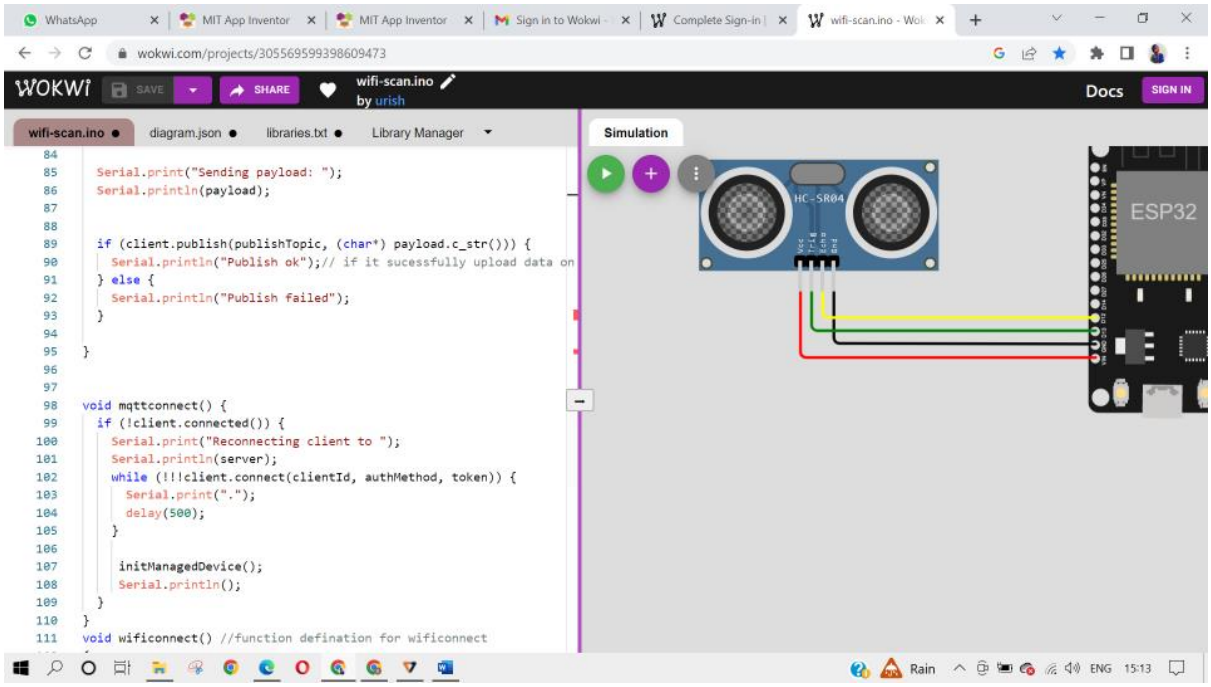
https://wokwi.com/projects/305569599398609473

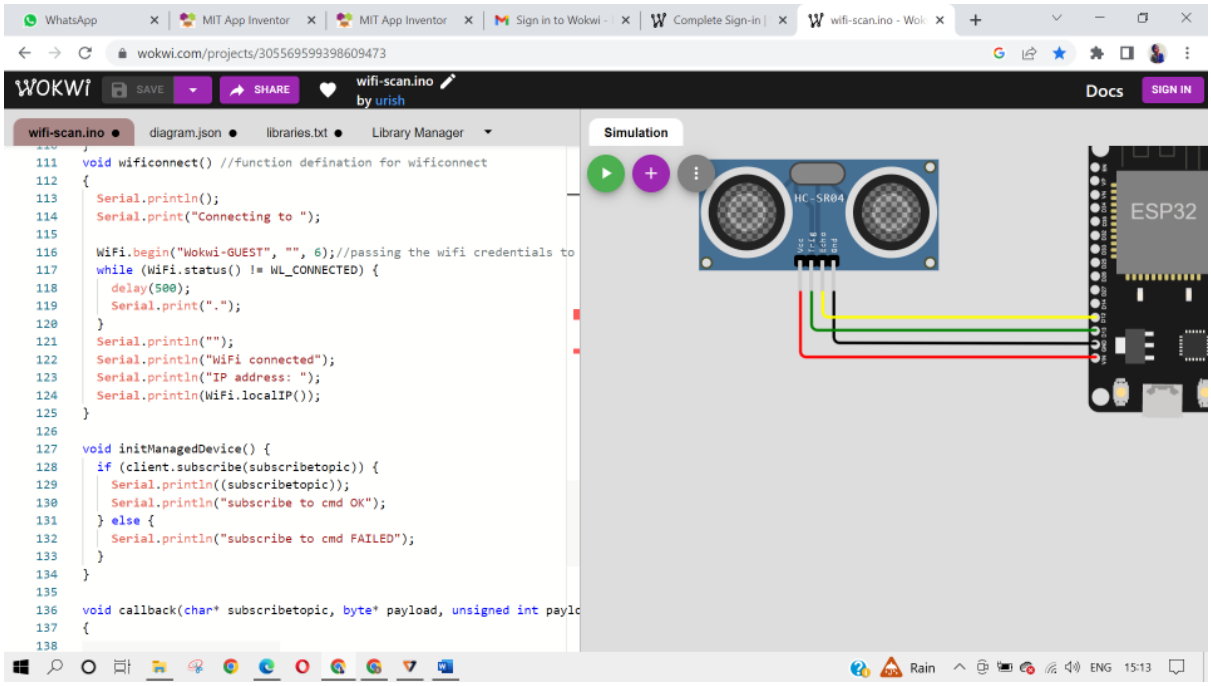
OUTPUT SCREENSHOT AND CODE:

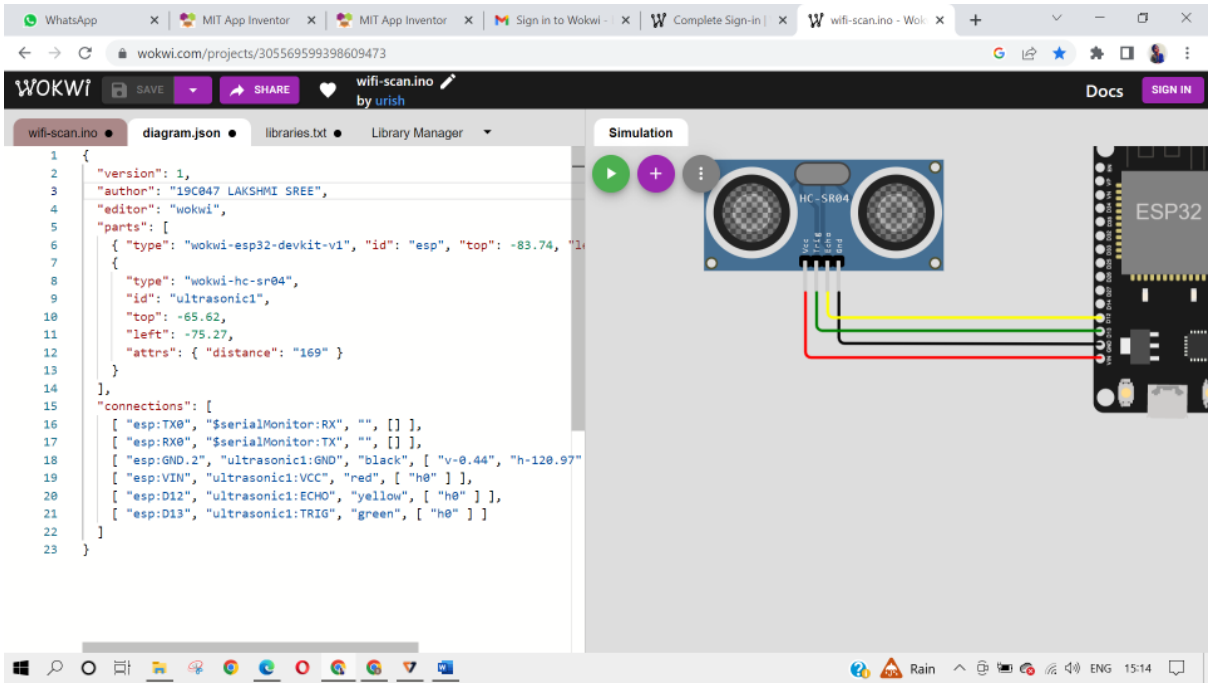


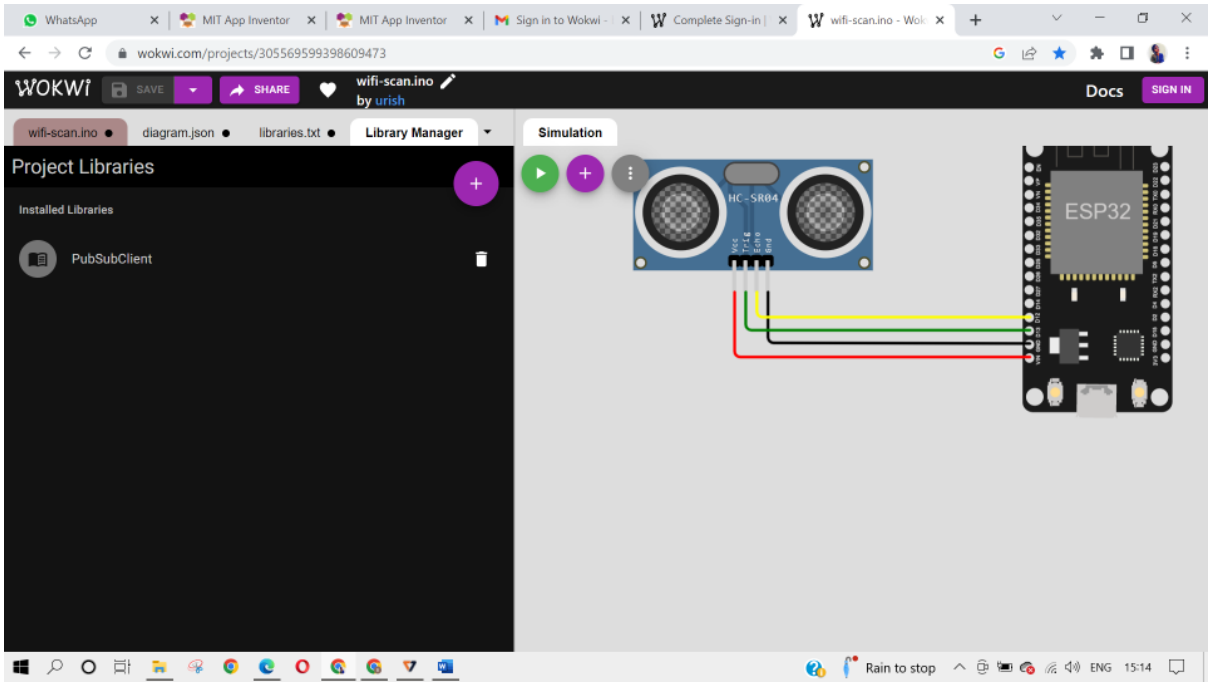


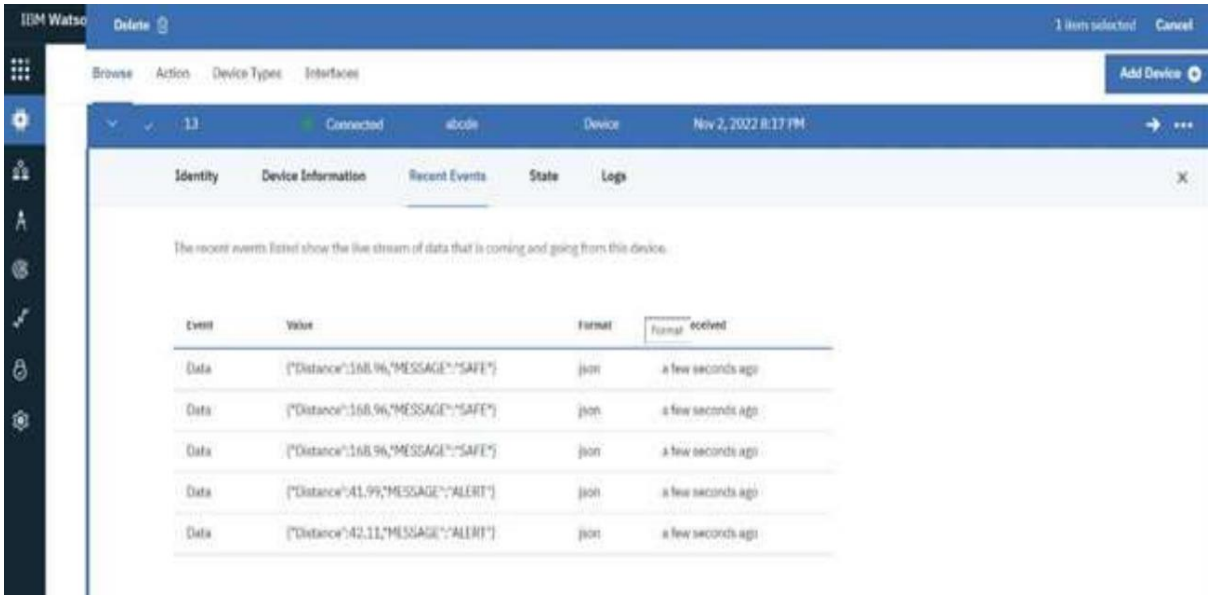

WOKWI
SAVE
SHARE
wifi-scan.ino
by urish
Docs
SIGN UP
wifi-scan.ino
diagram.json
libraries.txt
Library Manager
Simulation
HC-SR04
ESP32
PublishData(distance,"SAFE");
delay(1000);
if (!client.loop()) {
mqttconnect();
}
}
/.....retrieving to Cloud.....
void PublishData(float d,char s[]) {
mqttconnect();//function call for connecting to ibm
/*
creating the String in in form JSon to update the data to ibm
*/
String payload = "{\"Distance\":";
payload+=d;
payload+=",";
payload+="\"MESSAGE\":";
payload+="\"";
payload+=s;
payload+="\"";
payload+="}";


WOKWI
SAVE
SHARE
wifi-scan.ino
by urish
Docs
SIGN IN
wifi-scan.ino
diagram.json
libraries.txt
Library Manager
Simulation
HC-SR04
ESP32
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) {
Serial.println("Publish ok");// if it sucessfully upload data on
} 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() //function defination for wificonnect


WOKWI
SAVE
SHARE
wifi-scan.ino
by urish
Docs
SIGN IN
wifi-scan.ino
diagram.json
libraries.txt
Library Manager
Simulation
HC-SR04
ESP32
void wificonnect() //function defination for wificonnect
{
Serial.println();
Serial.print("Connecting to ");
WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to
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 paylo
{
Rain
ENG
15:13


WOKWI
SAVE
SHARE
wifi-scan.ino
by urish
Docs
SIGN IN
wifi-scan.ino
diagram.json
libraries.txt
Library Manager
Simulation
HC-SR04
ESP32
{
"version": 1,
"author": "19C047 LAKSHMI SREE",
"editor": "wokwi",
"parts": [
{ "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -83.74,
{
"type": "wokwi-hc-sr04",
"id": "ultrasonic1",
"top": -65.62,
"left": -75.27,
"attrs": { "distance": "169" }
}
],
"connections": [
["esp:TX0", "$serialMonitor:RX", "", []],
["esp:RX0", "$serialMonitor:TX", "", []],
["esp:GND.2", "ultrasonic1:GND", "black", ["v-0.44", "h-120.97"
["esp:VIN", "ultrasonic1:VCC", "red", ["h0"]],
["esp:D12", "ultrasonic1:ECHO", "yellow", ["h0"]],
["esp:D13", "ultrasonic1:TRIG", "green", ["h0"]]
]
}
Rain
ENG
15:14


wokwi.com/projects/305569599398609473
WOKWI
SAVE
SHARE
wifi-scan.ino
by urish
Docs
SIGN IN
wifi-scan.ino
diagram.json
libraries.txt
Library Manager
Project Libraries
Installed Libraries
PubSubClient
Simulation
HC-SR04
ESP32
Rain to stop
ENG
15:14


Delete
1 item selected
Cancel
Browse
Action
Device Types
Interfaces
Add Device
Connected
Device
Nov 2, 2022 8:17 PM
Identity
Device Information
Recent Events
State
Logs
Event
Value
Format
Data
{"Distance":168.96,"MESSAGE":"SAFE"}
json
a few seconds ago
{"Distance":41.99,"MESSAGE":"ALERT"}
{"Distance":42.11,"MESSAGE":"ALERT"}