

TUGAS SIB ARKATAMA MULTI SOLUSINDO
IOT PRAKTIK

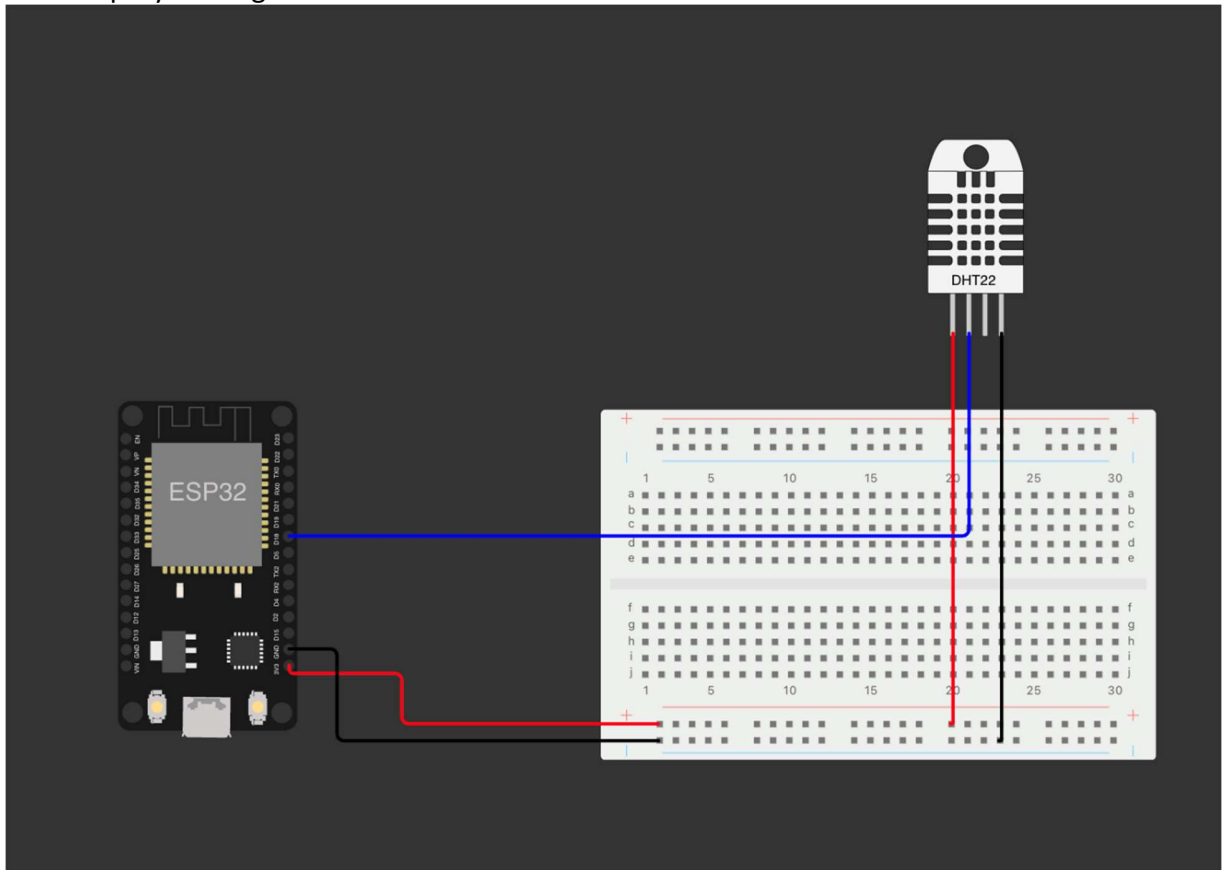
ID Kegiatan : 7582873
Nama : Evy Nur Imamah
Kelas : IoT1

TOOLS!

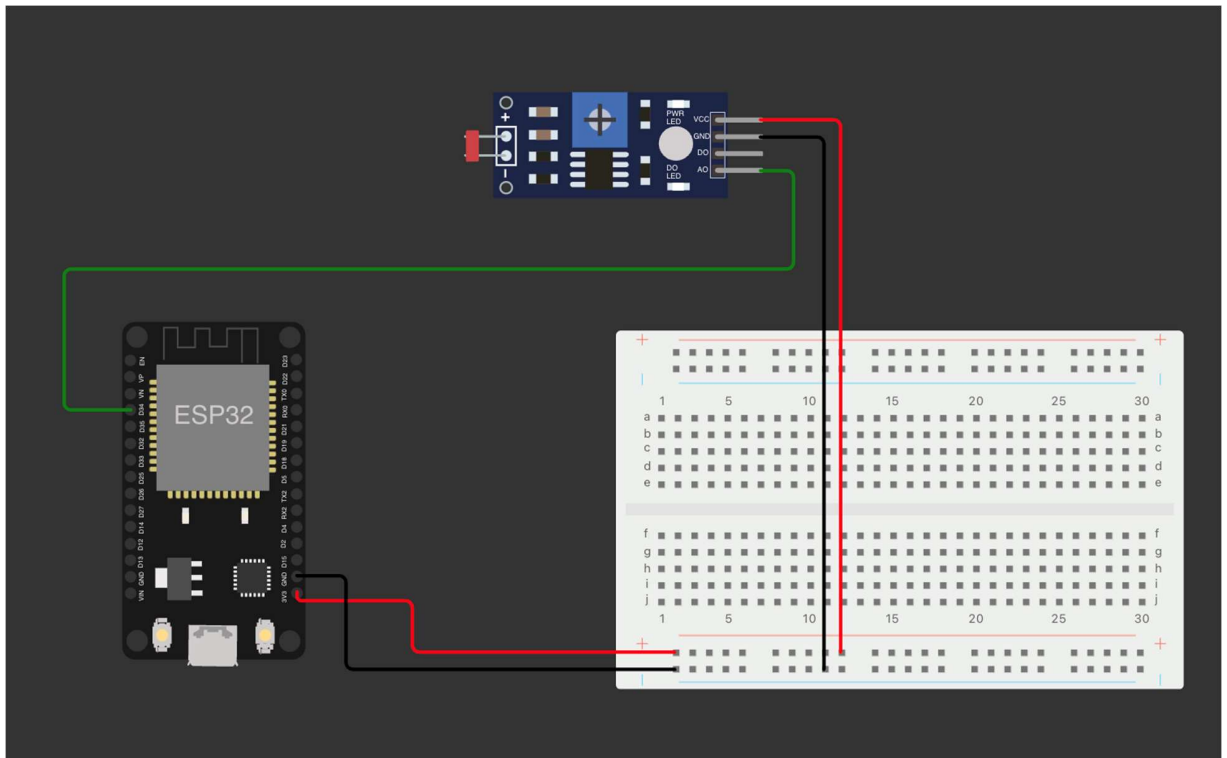
1. <https://wokwi.com/>
2. <https://emqx.com/>
3. <https://mqttx.app/>

Soal!

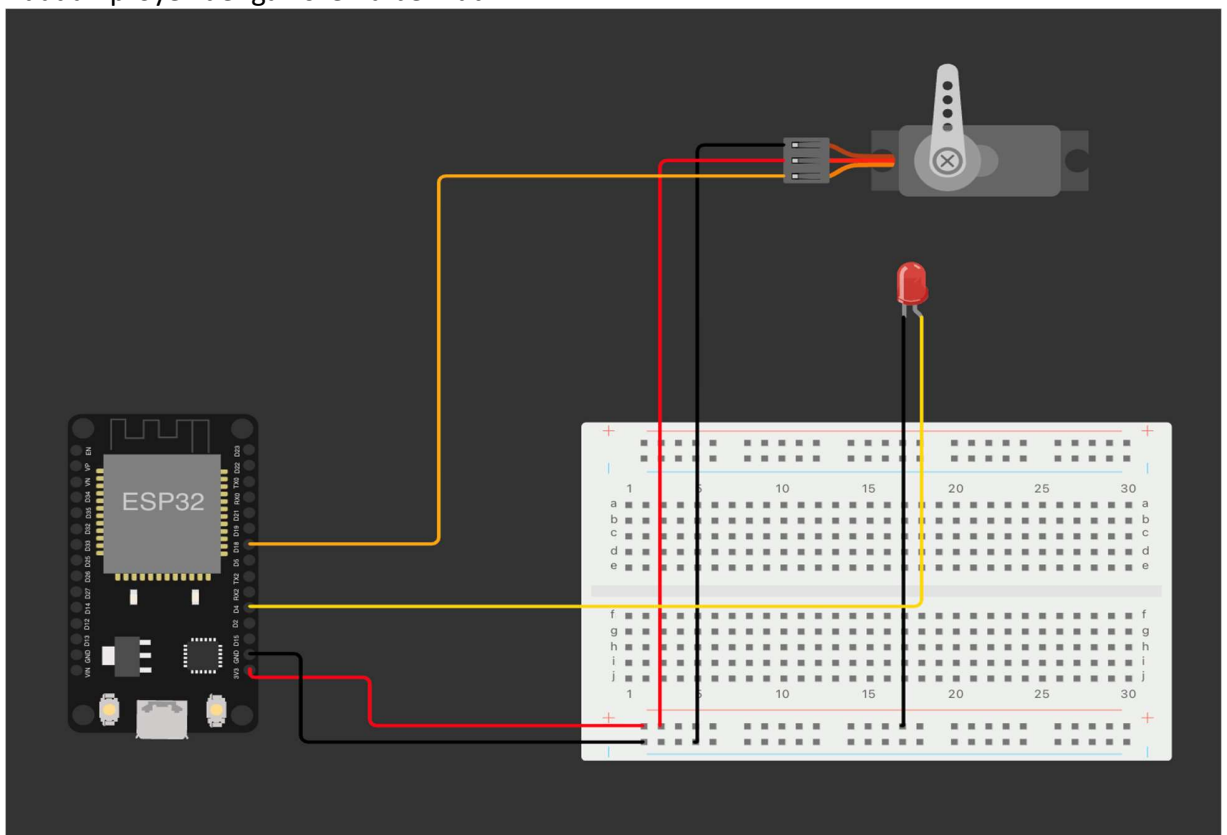
1. Buatlah proyek dengan skema berikut :



2. Buatlah proyek dengan skema berikut :



3. Buatlah proyek dengan skema berikut :



Deskripsi Proyek:

Proyek ini bertujuan untuk membuat tiga buah node yang saling terhubung ke satu MQTT server. Node pertama akan digunakan untuk memonitor suhu dan kelembapan menggunakan sensor DHT22 pada ESP32, node kedua akan memonitor kecerahan/cahaya, dan node ketiga akan bertindak sebagai aktuator untuk menyiram tanaman menggunakan servo pada batas suhu diatas 25 dan kelembapan dibawah 60 dan menhidupkan lampu jika kecerahan/cahaya melewati batas 800.

Spesifikasi Proyek:

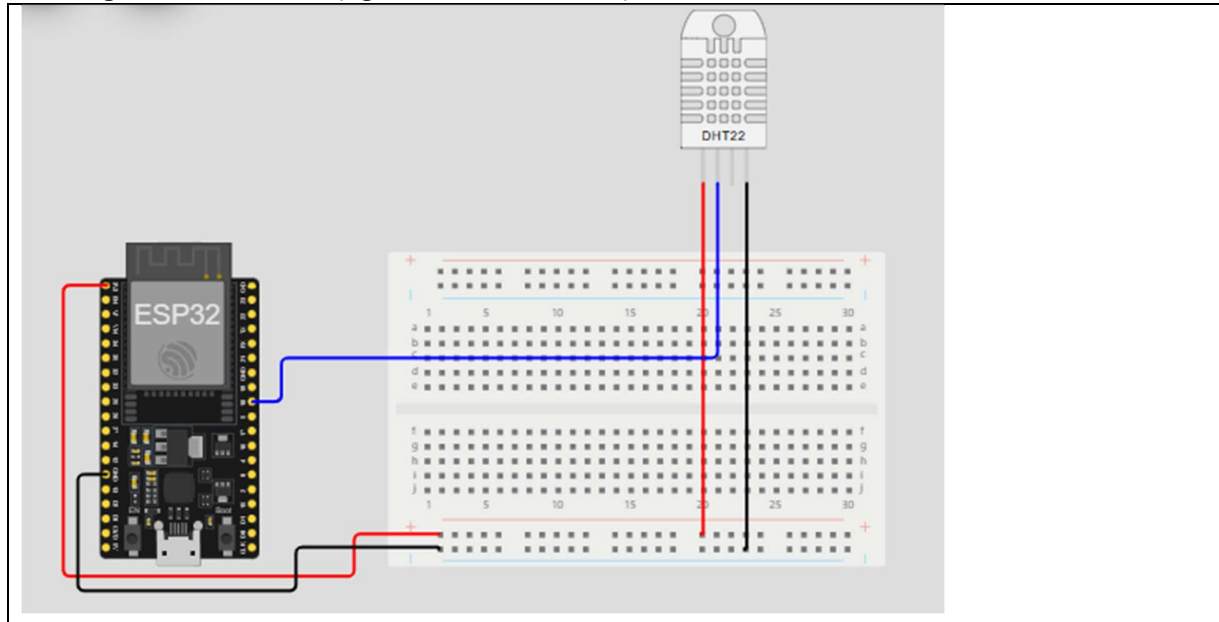
- a. Node Pertama: Monitoring Suhu dan Kelembapan
 - Menggunakan sensor DHT22 untuk memonitor suhu dan kelembapan lingkungan.
 - Data suhu dan kelembapan akan dikirimkan ke MQTT server dengan topik **monitoring/sensor/kelembapan** dan **monitoring/sensor/suhu**.
- b. Node Kedua: Monitoring Cahaya/Kecerahan
 - Memantau tingkat cahaya atau kecerahan lingkungan.
 - Data cahaya akan dikirimkan ke MQTT server dengan topik **monitoring/sensor/cahaya**.
- c. Node Ketiga: Kontrol Aktuator
 - Menerima data suhu dan kelembapan dari MQTT server.
 - Menyiram tanaman menggunakan servo jika suhu atau kelembapan berada di bawah batas tertentu.
 - Menghidupkan lampu jika tingkat kecerahan/cahaya melewati batas tertentu.

Langkah-langkah Proyek:

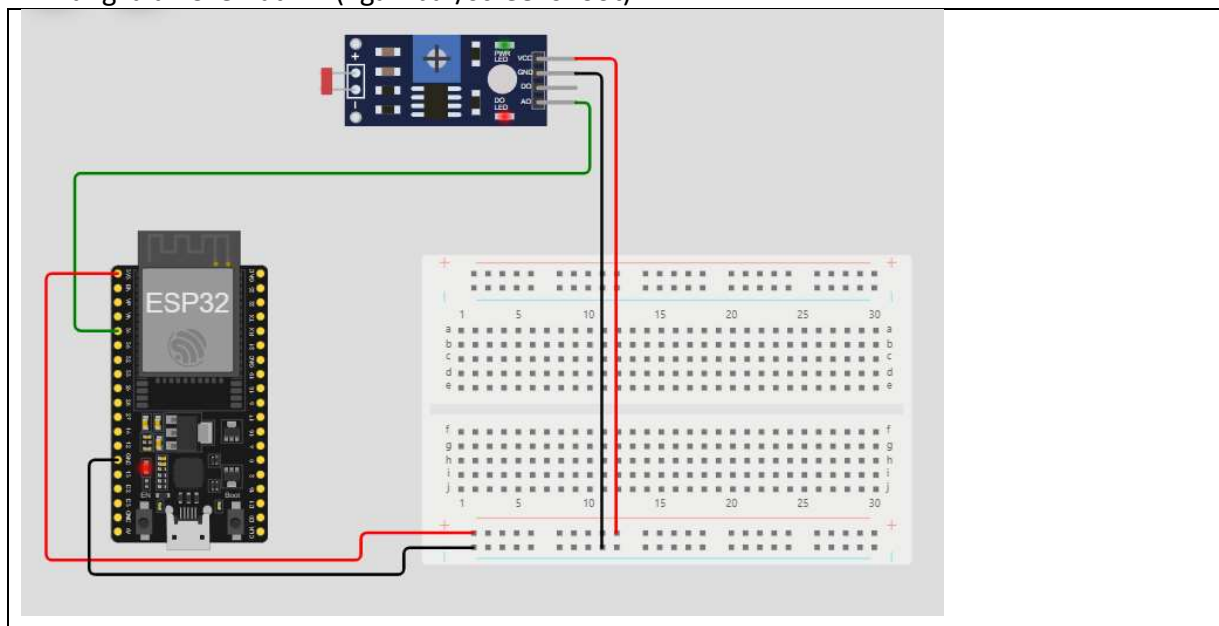
1. Persiapan Perangkat:
 - Persiapkan tiga buah perangkat ESP32 dengan sensor DHT22 untuk node pertama, sensor cahaya untuk node kedua, dan servo serta lampu untuk node ketiga.
 - Pastikan setiap perangkat terhubung ke jaringan WiFi dan memiliki akses ke MQTT server yang sama.
2. Pengembangan Kode Program:
 - Tulis kode program untuk masing-masing node menggunakan Arduino IDE atau platform yang sesuai.
 - Implementasikan logika untuk membaca data dari sensor DHT22 dan sensor cahaya.
 - Gunakan MQTT library untuk mengirimkan dan menerima pesan dari MQTT server.
 - Terapkan logika kontrol untuk aktuator (servo dan lampu) berdasarkan kondisi suhu, kelembapan, dan tingkat cahaya yang terukur.
3. Konfigurasi MQTT Server:
 - Pastikan MQTT server telah dikonfigurasi dengan benar untuk menerima dan mengirimkan pesan dari dan ke tiga node.
 - Buat topik-topik yang sesuai untuk menerima data suhu, kelembapan, dan cahaya, serta mengirimkan instruksi ke node ketiga.
4. Pengujian dan Debugging:
 - Uji coba setiap node secara terpisah untuk memastikan sensor berfungsi dengan baik dan dapat terhubung ke MQTT server.
 - Periksa apakah node ketiga dapat merespons dengan benar terhadap perubahan kondisi suhu, kelembapan, dan tingkat cahaya.

Jawab!

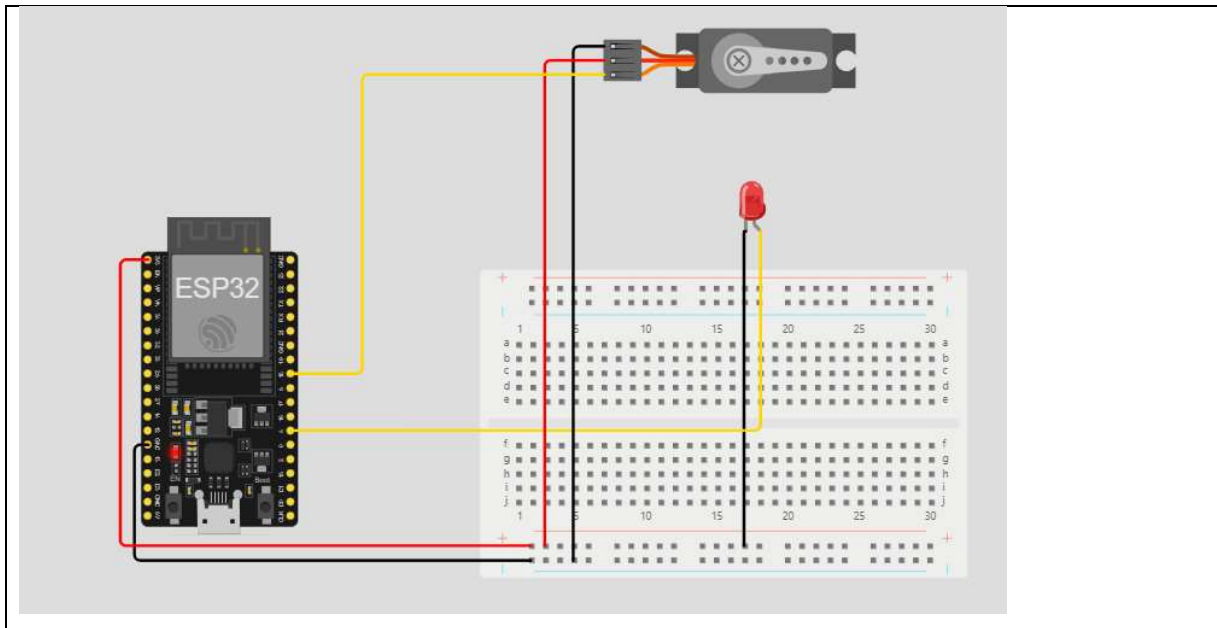
1. Rangkaian Skematik 1 (*gambar/screenshoot)



2. Rangkaian Skematik 2 (*gambar/screenshoot)



3. Rangkaian Skematik 3 (*gambar/screenshoot)



4. Program 1 (.ino,.c)

<https://wokwi.com/projects/394508490815732737>

```
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <PubSubClient.h>
#include <Arduino_JSON.h>
#include "DHT.h"

#define DHTPIN 18      // Pin yang terhubung dengan output sensor DHT22
#define DHTTYPE DHT22 // Jenis sensor DHT (DHT22)

const char *ssid = "Wokwi-GUEST";
const char *password = "";
const char *mqttServer = "dae791f7.ala.asia-southeast1.emqxsl.com";
const char *topic = "iot-6/suhu";

const char *mqtt_username = "imaima";
const char *mqtt_password = "imaima";
int port = 8883;

static const char *root_ca PROGMEM = R"EOF(
-----BEGIN CERTIFICATE-----
MIIDrzCCApegAwIBAgIQCDvgVpBCRrGhdWrJWZHHSjANBgkqhkiG9w0BAQUFADBh
MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYDVQQLExB3
d3cuZGlnaWNlcnQyY29tMSAwHgYDVQQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBD
d3cuZGlnaWNlcnQyY29tMSAwHgYDVQQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBD
```

```
QTAeFw0wNjExMTAwMDAwMDBaFw0zMTEwMTAwMDAwMDBaMGExCzAJBgNVBAYTA1VT
MRUwEwYDVQQKEwxEaWdpQ2VydCBJbmMxGTAXBgNVBAsTEHd3dy5kaWdpY2VydC5j
b20xIDAeBgNVBAMTF0RpZ2lDZXJ0IEdsb2JhbCBSb290IENBMIIBIjANBgkqhkiG
9w0BAQEFAAOCAQ8AMIIBCgKCAQEA4jvhEXLeqKTTolEqUKKPC3eQyaKl7hLOllsB
CSDMAZOnTjC3U/dDxGkAV53ijSLdhwZAAIEJzs4bg7/fzTtxRuLWZscFs3YnFo97
nh6Vfe63SKMI2tavegw5BmV/S10fvBf4q77uKNd0f3p4mVmFaG5cIzJLv07A6Fpt
43C/dxC//AH2hdmoRBBYMql1GNXRor5H4idq9Joz+EkIYIvUX7Q6hL+hqkpMfT7P
T19sdl6gSzeRntwi5m3OFBqOasv+zbMUZBfHWymeMr/y7vrTC0LUq7dBMtoM10/4
gdW7jVg/tRvoSSiicNoxBN33shbyTAPOB6jtSj1etX+jkMOvJwIDAQABo2MwYTAO
BgNVHQ8BAf8EBAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUA95QNVbR
TLtm8KPiGxvDl7I90VUwHwYDVR0jBBgwFoAUA95QNVbRTLtm8KPiGxvDl7I90VUw
DQYJKoZIhvcNAQEFBQADggEBAMucN6pIExIK+t1EnE9SsPTfrgT1eXkIoyQY/Esr
hMATudXH/vTBH1jLuG2cenTnmCmrEbXjckChzUyImZOMkXDiqw8cvpOp/2PV5Adg
06O/nVsJ8dWO41P0jMP6P6fbtGbYmbW0W5BjfiTtep3Sp+dWOIrWcBAI+0tKIJF
PnlUkiaY4IBIqDfv8NZ5YBberOgOzW6sRbc4L0na4UU+Krk2U886UAb3LujEV0ls
YSEYlQSteDwsOoBrp+uvFRTP2InBuThs4pFsiv9kuXclVzDAGySj4dzp30d8tbQk
CAUw7C29C79Fv1C5qfPrmAESrciIxpG0X40KPMbp1ZWVbd4=
```

-----END CERTIFICATE-----

) EOF";

DHT dht(DHTPIN, DHTTYPE);

WiFiClientSecure espClient;

PubSubClient client(espClient);

unsigned long lastTime = 0;

unsigned long intervalTime = 1000 * 60 * 5; // Publish data setiap 5 menit

void mqtt_init();

void publishData();

void setup() {

Serial.begin(115200);

delay(1000);

WiFi.begin(ssid, password);

Serial.print("Connecting");

while (WiFi.status() != WL_CONNECTED) {

delay(100);

Serial.print('.');

}

Serial.println("Connected");

dht.begin();

```

    espClient.setCACert(root_ca);
    client.setServer(mqttServer, port);

    while (!client.connected()) {
        mqtt_init();
    }
}

void loop() {
    client.loop();

    unsigned long currentMillis = millis();
    if (currentMillis - lastTime >= intervalTime) {
        publishData();
        lastTime = currentMillis;
    }
}

void mqtt_init() {
    espClient.setCACert(root_ca);
    client.setServer(mqttServer, port);

    String client_id = "sub-esp32-iot-6-";
    client_id += WiFi.macAddress();
    if (client.connect(client_id.c_str(), mqtt_username, mqtt_password)) {
        Serial.println("Connected to MQTT broker");
        client.subscribe(topic);
    } else {
        Serial.print("Failed to connect to MQTT broker, rc=");
        Serial.println(client.state());
        delay(2000);
    }
}

void publishData() {
    float temperature = dht.readTemperature();
    float humidity = dht.readHumidity();

    if (isnan(temperature) || isnan(humidity)) {
        Serial.println("Failed to read data from DHT22!");
        return;
    }

    JSONVar data;

```



```
data["temperature"] = temperature;
data["humidity"] = humidity;

String jsonString = JSON.stringify(data);

if (!client.publish(topic, jsonString.c_str())) {
    Serial.println("Failed to publish data to MQTT topic!");
} else {
    Serial.println("Data published to MQTT topic:");
    Serial.println(jsonString);
}
}
```

5. Program 2 (.ino,.c)

<https://wokwi.com/projects/394509150748609537>

```
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <PubSubClient.h>
#include <Arduino_JSON.h>

const char *ssid = "Wokwi-GUEST";
const char *password = "";
const char *mqttServer = "dae791f7.ala.asia-southeast1.emqxsl.com";
const char *topic = "iot-6/cahaya";

const char *mqtt_username = "imaima";
const char *mqtt_password = "imaima";
int port = 8883;

static const char *root_ca PROGMEM = R"EOF(
-----BEGIN CERTIFICATE-----
MIIDrzCCApegAwIBAgIQCDvgVpBCRRrGhdWrJWZHHSjANBgkqhkiG9w0BAQUFADBh
MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYDVQQLExB3
d3cuZGlnaWNlcnQyY29tMSAwHgYDVQQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBD
QTAeFw0wNjExMTAwMDAwMDBaFw0zMTEwMTAwMDAwMDBaMGExCzAJBgNVBAYTA1VT
MRUwEwYDVQQKEwxEaWdpQ2VydCBHbG9iYWwgNVBASTEHD3dy5kaWdpY2VydC5j
b20xIDAeBgNVBAMTF0RpZ2lDZXJ0IEUdsb2JhbCBSb290IENBMIIBIjANBgkqhkiG
9w0BAQEFAAOCAQ8AMIIBCgKCAQEA4jvhEXLeqKTTolEqUKKPC3eQyaKl7hLOllsB
CSDMAZOnTjC3U/dDxGkAV53ijSLdhwZAAIEJzs4bg7/fzTtxRuLWZscFs3YnFo97
nh6Vfe63SKMI2tavegw5BmV/S10fvBf4q77uKNd0f3p4mVmFaG5cIzJLv07A6Fpt
43C/dxC//AH2hdmoRBBYMql1GNXRor5H4idq9Joz+EkIYIvUX7Q6hL+hqkpMfT7P
T19sdl6gSzeRntwi5m3OFBqOasv+zbMUZBFHWymeMr/y7vrTC0LUq7dBMtoM10/4
gdW7jVg/tRvoSSiicNoxBN33shbyTAPOB6jtSj1etX+jkMOvJwIDAQABO2MwYTAO
BgNVHQ8BAf8EBAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUA95QNVbR
TLtm8KPiGxvDl7I90VUwHwYDVR0jBBgwFoAUA95QNVbRTLtm8KPiGxvDl7I90VUw
DQYJKoZIhvcNAQEFBQADggEBAMucN6pIExIK+t1EnE9SsPTfrgT1eXkIoyQY/Esr
hMATudXH/vTBH1jLuG2cenTnmCmrEbXjckChzUyImZOMkXDiqw8cvpOp/2PV5Adg
060/nVsJ8dWO41P0jnP6P6fbtGbfYmbW0W5BjfIttep3Sp+dWOIrWcBAI+0tKIJF
PnlUkiaY4IBIqDfv8NZ5YBberOgOzW6sRBC4L0na4UU+Krk2U886UAb3LuJEV0ls
YSEY1QSteDwsOoBrp+uvFRTP2InBuThs4pFsiv9kuXclVzDAGySj4dzp30d8tbQk
CAUw7C29C79Fv1C5qfPrmAESrciIxpgoX40KPMbp1ZWVbd4=
-----END CERTIFICATE-----
)EOF";

const int ldrPin = 34; // Pin yang terhubung dengan sensor LDR

WiFiClientSecure espClient;
```

```

PubSubClient client(espClient);

unsigned long lastTime = 0;
unsigned long intervalTime = 1000 * 60 * 5; // Publish data setiap 5
menit

void mqtt_init();
void publishData();

void setup() {
    Serial.begin(115200);
    delay(1000);

    WiFi.begin(ssid, password);
    Serial.print("Connecting");
    while (WiFi.status() != WL_CONNECTED) {
        delay(100);
        Serial.print('.');
    }
    Serial.println("Connected");

    pinMode(ldrPin, INPUT);

    espClient.setCACert(root_ca);
    client.setServer(mqttServer, port);

    while (!client.connected()) {
        mqtt_init();
    }
}

void loop() {
    client.loop();

    unsigned long currentMillis = millis();
    if (currentMillis - lastTime >= intervalTime) {
        publishData();
        lastTime = currentMillis;
    }
}

void mqtt_init() {
    espClient.setCACert(root_ca);
    client.setServer(mqttServer, port);
}

```

```
String client_id = "sub-esp32-iot-6-";
client_id += WiFi.macAddress();
if (client.connect(client_id.c_str(), mqtt_username, mqtt_password)) {
    Serial.println("Connected to MQTT broker");
    client.subscribe(topic);
} else {
    Serial.print("Failed to connect to MQTT broker, rc=");
    Serial.println(client.state());
    delay(2000);
}
}

void publishData() {
    int lightLevel = analogRead(ldrPin);

    JSONVar data;
    data["light_level"] = lightLevel;

    String jsonString = JSON.stringify(data);

    if (!client.publish(topic, jsonString.c_str())) {
        Serial.println("Failed to publish data to MQTT topic!");
    } else {
        Serial.println("Data published to MQTT topic:");
        Serial.println(jsonString);
    }
}
```

6. Program 3 (.ino,.c)

<https://wokwi.com/projects/394509557568873473>

```
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <PubSubClient.h>
#include <Arduino_JSON.h>
#include <ESP32Servo.h> // Tambahkan header library ESP32Servo

const char *ssid = "Wokwi-GUEST";
const char *password = "";
const char *mqttServer = "dae791f7.ala.asia-southeast1.emqxsl.com";
const char *topic = "iot-6/servo";

const char *mqtt_username = "imaima";
const char *mqtt_password = "imaima";
int port = 8883;

static const char *root_ca PROGMEM = R"EOF(
-----BEGIN CERTIFICATE-----
MIIDrzCCApegAwIBAgIQCDvgVpBCRRrGhdWrJWZHHSjANBgkqhkiG9w0BAQUFADBh
MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYDVQQLExB3
d3cuZGlnaWNlcnQuY29tMSAwHgYDVQQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBD
QTAeFw0wNjExMTAwMDAwMDBaFw0zMTEwMTAwMDAwMDBaMGExCzAJBgNVBAYTA1VT
MRUwEwYDVQQKEwxEaWdpQ2VydCBJbG9jaW4uY290aW4uY290aW4uY290aW4uY290
b20xIDAeBgNVBAMTF0RpZ2l0eXJ0IEUdsb2JhbCBSc290IENBMIIIBIjANBgkqhkiG
9w0BAQEFAAOCAQ8AMIIBCgKCAQEAA4jvhEXLeqKTT0leqUKKPC3eQyaKl7hL0llsB
CSDMAZOnTjC3U/dDxGkAV53ijSLdhwZAAIEJzs4bg7/fzTtxRuLWZscFs3YnFo97
nh6Vfe63SKMI2tavegw5BmV/S10fvBf4q77uKNd0f3p4mVmFaG5cIzJLv07A6Fpt
43C/dxC//AH2hdmORBBYMq1lGNXRor5H4idq9Joz+EkIYIvUX7Q6hL+hqkpMfT7P
T19sdl6gSzeRntwi5m3OFBqOasv+zbMUZBfHWymeMr/y7vrTC0LUq7dBMtoM10/4
gdW7jVg/trVoSSiicNoxBN33shbyTAPOB6jtSj1etX+jkMOvJwIDAQABO2MwYTAO
BgNVHQ8BAf8EBAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUA95QNVbR
TLtm8KPiGxvDl7I90VUwHwYDVR0jBBgwFoAUA95QNVbRTLtm8KPiGxvDl7I90VUw
DQYJKoZIhvcNAQEFBQADggEBAMucN6pIEIXIK+t1EnE9SsPTfrgT1eXkIoyQY/Esr
hMATudXH/vTBH1jLuG2cenTnmCmrEbXjcKChzUyImZOMkXDiqw8cvpOp/2PV5Adg
06O/nVsJ8dWO41P0jMP6P6fbtGbYmbW0W5BjfIttep3Sp+dWOIrWcBAI+0tKIJF
PnlUkiaY4IBIqDfv8NZ5YBberOgOzW6sRBc4L0na4UU+Krk2U886UAb3LujEV0ls
YSEY1QSteDwsOoBrp+uvFRTP2InBuThs4pFsiv9kuXclVzDAGySj4dzp30d8tbQk
CAUw7C29C79Fv1C5qfPrmAESrciIxpG0X40KPMbp1ZWVbd4=
-----END CERTIFICATE-----
) EOF";

WiFiClientSecure espClient;
PubSubClient client(espClient);
```

```
unsigned long lastTime = 0;
unsigned long intervalTime = 1000 * 60 * 5; // Publish data setiap 5
menit

void mqtt_init();
void publishData();

// Inisialisasi servo pada pin 18
Servo myServo;

void setup() {
    Serial.begin(115200);
    delay(1000);

    WiFi.begin(ssid, password);
    Serial.print("Connecting");
    while (WiFi.status() != WL_CONNECTED) {
        delay(100);
        Serial.print('.');
    }
    Serial.println("Connected");

    espClient.setCACert(root_ca);
    client.setServer(mqttServer, port);

    // Attach servo pada pin 18
    myServo.attach(18);

    while (!client.connected()) {
        mqtt_init();
    }
}

void loop() {
    client.loop();

    unsigned long currentMillis = millis();
    if (currentMillis - lastTime >= intervalTime) {
        publishData();
        lastTime = currentMillis;
    }
}

void mqtt_init() {
```

```

espClient.setCACert(root_ca);
client.setServer(mqttServer, port);

String client_id = "sub-esp32-iot-6-";
client_id += WiFi.macAddress();
if (client.connect(client_id.c_str(), mqtt_username, mqtt_password)) {
    Serial.println("Connected to MQTT broker");
    client.subscribe(topic);
} else {
    Serial.print("Failed to connect to MQTT broker, rc=");
    Serial.println(client.state());
    delay(2000);
}
}

void publishData() {
    // Menggerakkan servo secara acak antara 0 dan 180 derajat
    int pos = random(0, 180);
    myServo.write(pos);
    delay(500); // Memberikan waktu untuk servo bergerak

    // Mendapatkan nilai suhu dan kelembaban
    float temperature = random(20, 40); // Ganti dengan nilai suhu yang
sebenarnya
    float humidity = random(40, 70);    // Ganti dengan nilai kelembaban
yang sebenarnya

    JSONVar data;
    data["temperature"] = temperature;
    data["humidity"] = humidity;

    String jsonString = JSON.stringify(data);

    if (!client.publish(topic, jsonString.c_str())) {
        Serial.println("Failed to publish data to MQTT topic!");
    } else {
        Serial.println("Data published to MQTT topic:");
        Serial.println(jsonString);
    }
}

```

Output 1

W penyiraman - Wokwi ESP32, STM x W cahaya - Wokwi ESP32, STM32, A x W suhu - Wokwi ESP32, STM32, Ard x +

WOKWI

SAVE

SHARE

suhu

Docs

sketch.ino

diagram.json

libraries.txt

Library Manager

```
6
7 #define DHTPIN 18 // Pin yang terhubung dengan output sensor DHT2
8 #define DHTTYPE DHT22 // Jenis sensor DHT (DHT22)
9
10 const char *ssid = "Wokwi-GUEST";
11 const char *password = "";
12 const char *mqttServer = "dae791f7.ala.asia-southeast1.emqxsl.com";
13 const char *topic = "iot-6/suhu";
14
15 const char *mqtt_username = "imaima";
16 const char *mqtt_password = "imaima";
17 int port = 8883;
18
19 static const char *root_ca PROGMEM = R"EOF(
20 -----BEGIN CERTIFICATE-----
21 MIIDrzCCApegAIBAgIQCVdPBCRRGhdiwJWZHHSjANBgkqhkiG9w0BAQUFA
22 MQsQCWQVQ0Q6EwEJUVzEVMBMGA1UEChMwRGlnaUNlcnQgSW5jMRkwFwYD
23 dScUg1naUlnInQy29tHSAWgVdVQ0Q6EkdAwdPQ2VydCBHbG91cmVudC
24 QTAeFw0wMjEwMTA0MDAwMDAaMBEwMTEwMTA0MDAwMDAwMDAwMDAwMDAw
25 MRUwEwYDQ0Q6EwEwEwMDPQ2VydCB3bmR0AGBgbG9wMTEwMTA0MDAwMD
26 b2QwIDAeBgNVBAMTF0RlPz21LDZXIEdsbnJhcCBSc29lbnB1IiEjANBgkqhkiG
27 9w0BAQEEAAOCQA8IAIIBCBGCAQEA4jvHkEqkT01eqUKPC3eQyK17hL011sB
28 CSDMAZ0nTjCSU/dXGkAV53iJSLdhwZAAIEJzs4bg7/fzTtxRuLNZscFs3YnF097
29 nh5Vfe63SKI21avegw58mV/S10fVbF4q77uKNd0f3p4mVmFag5cIz3lv07A6Fpt
30 43C/dx/c/AH2hdmoRBVYm1GNXRorSH4Idq9Joz+EKIIVIXUQ6hL+hqkPMFT7P
31 T19sd16gSzeRntwi5m30F8Qoasv+zbMUZBFHlymeMr/y7vrTC0LUq7dBMtoM10/4
32 gdw7Jvg/trVosSiiNoxBN33shbyTap086t5j1etX+JkMoYJwIDAQABo2MwYTAO
```

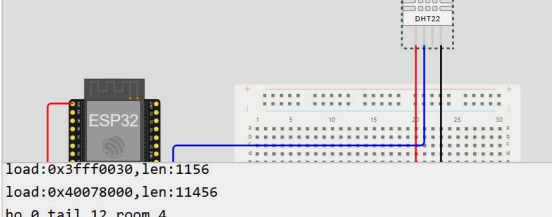
Simulation

00:18.565 22%

Editing DHT22

Temperature: 18.8°C

Humidity: 40.0%



load:0x3ffff0030, len:1156

load:0x40078000, len:11456

ho 0 tail 12, room 4

load:0x40080400, len:2972

entry 0x400805dc

Connected.....Connected

Connected to MQTT broker

DOW -1.09%

Windows Taskbar

System Clock: 15:27 11/04/2024

Output 2

penyiraman - Wokwi ESP32, STM32

W cahaya - Wokwi ESP32, STM32

+

WOKWI

SAVE

SHARE

cahaya

Docs

sketch.ino

diagram.json

libraries.txt

Library Manager

```
1 #include <WiFi.h>
2 #include <WiFiClientSecure.h>
3 #include <PubSubClient.h>
4 #include <Arduino_JSON.h>
5
6 const char *ssid = "Wokwi-GUEST";
7 const char *password = "";
8 const char *mqttServer = "dae791f7.ala.asia-southeast1.emqxsl.com";
9 const char *topic = "iot-6/cahaya";
10
11 const char *mqtt_username = "imaima";
12 const char *mqtt_password = "imaima";
13 int port = 8883;
14
15 static const char *root_ca PROGMEM = R"EOF(
16 -----BEGIN CERTIFICATE-----
17 MIIDrzcCApgeAwIBAgIQCDvGpBCRRGhdwRrWJZHSHJANBgkqhkiG9w0BAQUFADBh
18 MQswCQYDVQQGEwVudm8wMDE1UEUwMDE1UEUwMDE1UEUwMDE1UEUwMDE1UEUwMDE1
19 d3cuZGlnaWlnQnQy29tSAsHwGVDDVQDQEdEaWdpQ2VydCBhbg9iYm9vU9vdCBD
20 QTAEFw0wNjExMTAwMDAwMDBaFw0zMTExMTAwMDAwMDBaMGExCzA3bGVhYTA1VT
21 MRUwEwYDVQKKEwEaWdpQ2VydCBjbm9kZXQwMDE1UEUwMDE1UEUwMDE1UEUwMDE1
22 b2x0IAEABGVhYTA1VTB0ZDQ2VydCBjbm9kZXQwMDE1UEUwMDE1UEUwMDE1UEUwMDE1
23 9w0BAQEFAOCQAQ8AMIIBGqCAQEAA4vhELeqKTT01eqUKPC3eQyAK17hL011sB
24 CSDMAZontJ3Csu/d0xGkAV53ijLdshwZAAIEJzs4bg7/fzTtXrULNZscfs3YnFo97
25 nh6Vfe3SKMI2taveg58BmV/S10FvBF4q77UKn0dF3p4mVmfAG5CI2JLV07A6Fpt
26 43C/dxC//AH2hdmoRB9yH1q1NqXorSH41dQ9Joz+EKIYIvUX7Q6hL+hqkPmFT7P
27 T19sd16SzeRntwi5m30F8Qoasv+zbMUZBFHymeMr/y7vrTC0L7dBMtoM10/4
```

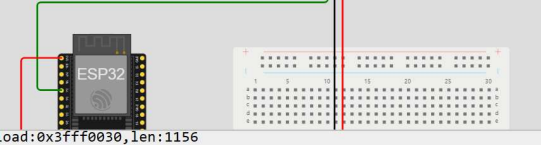
Simulation

00:11.699 19%

Photoresistor (LDR)

ILLUMINATION (LUX)

550 lux



load:0x3ffff0030, len:1156
load:0x40078000, len:11456
ho 0 tail 12 room 4
load:0x40080400, len:2972
entry 0x400805dc
Connecting.....Connected
Connected to MQTT broker

30°C

Cuaca panas

Cari

11/04/2024

Output 3

WOKWI

penyiraman - Wokwi ESP32, STM x

https://wokwi.com/projects/394509557568873473

SAVE SHARE

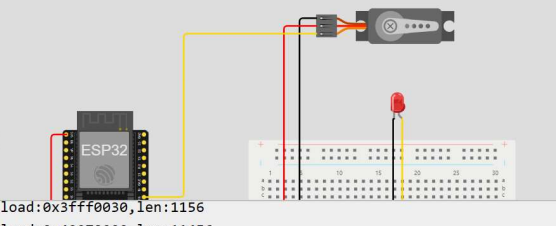
penyiraman Docs

sketch.ino diagram.json libraries.txt Library Manager

```
1 #include <WiFi.h>
2 #include <WiFiClientSecure.h>
3 #include <PubSubClient.h>
4 #include <Arduino_JSON.h>
5 #include <ESP32Servo.h> // Tambahkan header library ESP32Servo
6
7 const char *ssid = "Wokwi-GUEST";
8 const char *password = "";
9 const char *mqttServer = "dae791f7.ala.asia-southeast1.emqxsl.com";
10 const char *topic = "iot-6/servo";
11
12 const char *mqtt_username = "imaima";
13 const char *mqtt_password = "imaima";
14 int port = 8883;
15
16 static const char *root_ca PROGMEM = R"EOF(
17 -----BEGIN CERTIFICATE-----
18 MIIDrzCCApegAwIBAgIQCDvgVpBCRRrGhdwRjWZHSjANBgkqhkiG9w0BAQUFADBh
19 MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW45MRkwFwYDVQQLExB3
20 d3cuZGlnaWlnaUNlcnQy29tMSAwHgYDVQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBk
21 QTAeFw0wNjExMTAwMDAwMDAwFw0zMTExMTAwMDAwMDAwMDAwGGEwCzAJBgNVBAYTA1VT
22 MRUwEwYDVQQKEwxEaWdpQ2VydCBjbmMxGTAXBgNVBAsTEHd3dy5kaWdpY2VydC5j
23 b20xIDAeBgNVBAMTF0R5d3cuZGlnaWlnaUNlcnQy29tMSAwHgYDVQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBk
24 QTAeFw0wNjExMTAwMDAwMDAwFw0zMTExMTAwMDAwMDAwMDAwGGEwCzAJBgNVBAYTA1VT
25 MRUwEwYDVQQKEwxEaWdpQ2VydCBjbmMxGTAXBgNVBAsTEHd3dy5kaWdpY2VydC5j
26 b20xIDAeBgNVBAMTF0R5d3cuZGlnaWlnaUNlcnQy29tMSAwHgYDVQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBk
27 43C/dxC//AH2hdmoRBBYMQ1GNXRorSH4Idq9Joz+EkIYlvUX7Q6hL+haqpmFT7P
```

Simulation

00:24.240 43%



load:0x3fff0030,len:1156
load:0x40078000,len:11456
ho 0 tail 12 room 4
load:0x40080400,len:2972
entry 0x400805dc
Connecting.....Connected
Connected to MQTT broker

15.36 11/04/2024