Nama : Ludang Prasetyo Nugroho [Teknik Komputer ( S1)](https://www.utdi.ac.id/)

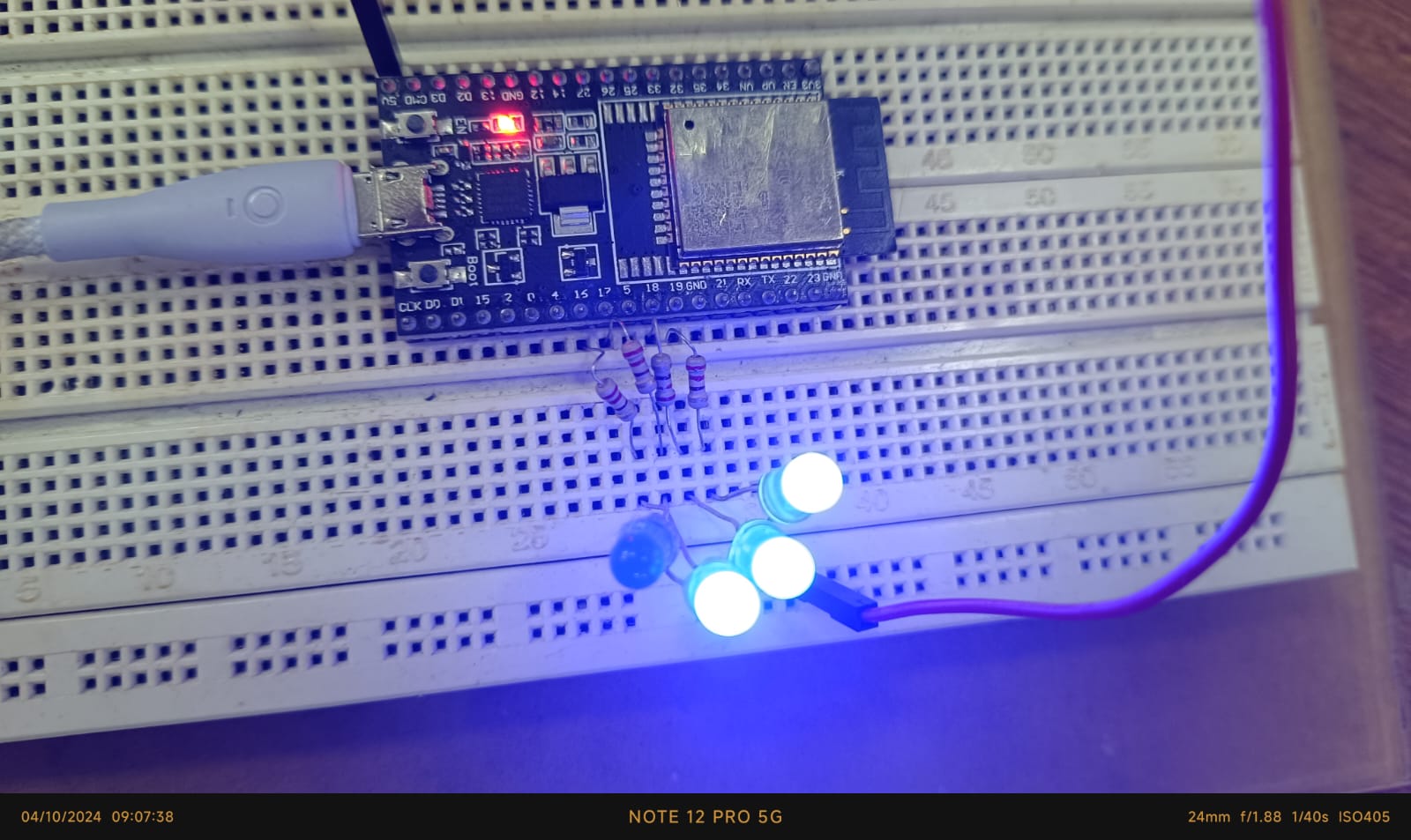
Nim : 225510017

Matkul : Prak system IOT

**PRAKTEK**

# Praktek 1

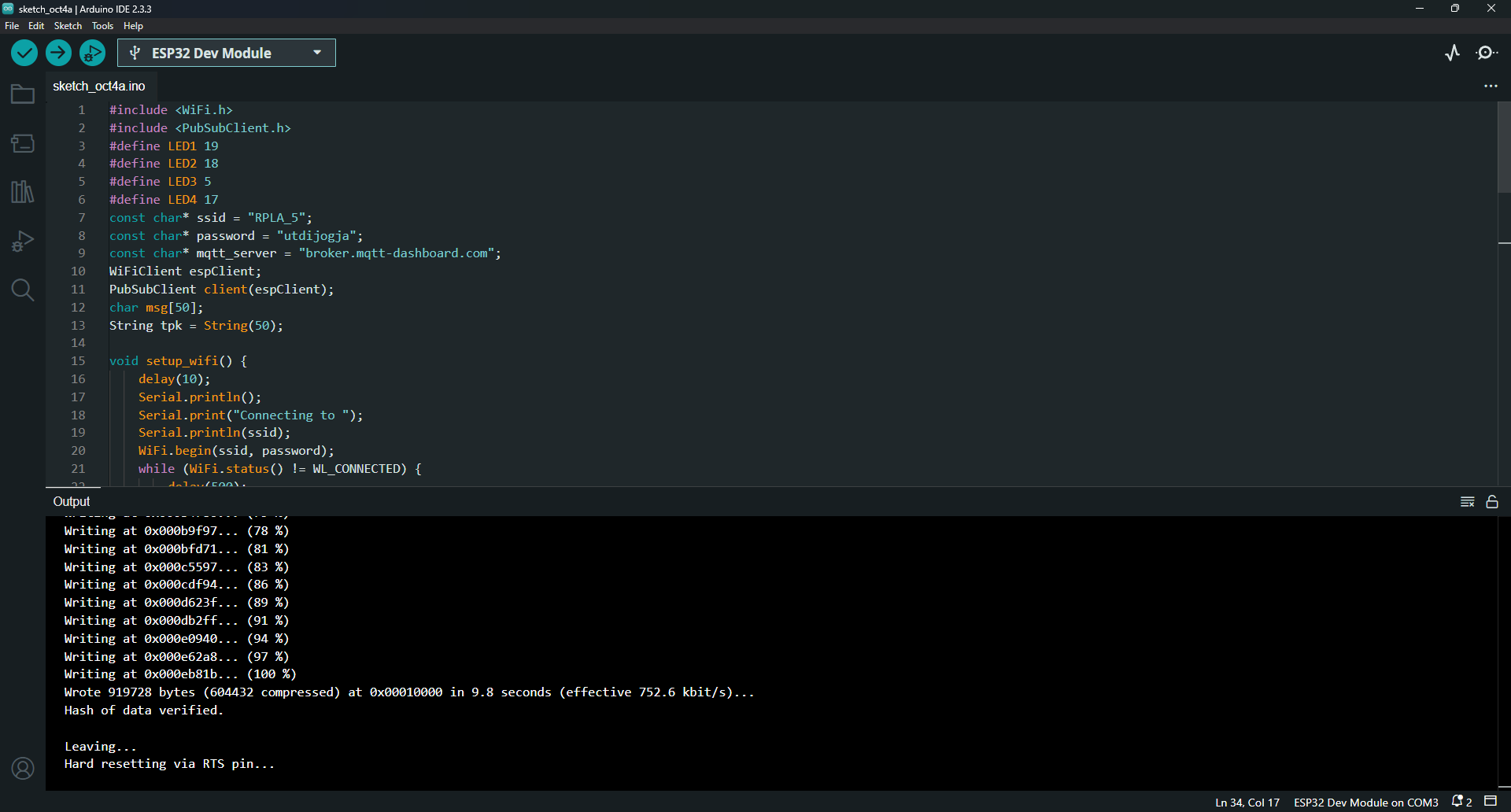
* Bentuk rangkaia



Lampu yang paling timur matisoalnya di dalam code lampunya tidak di nyalakan

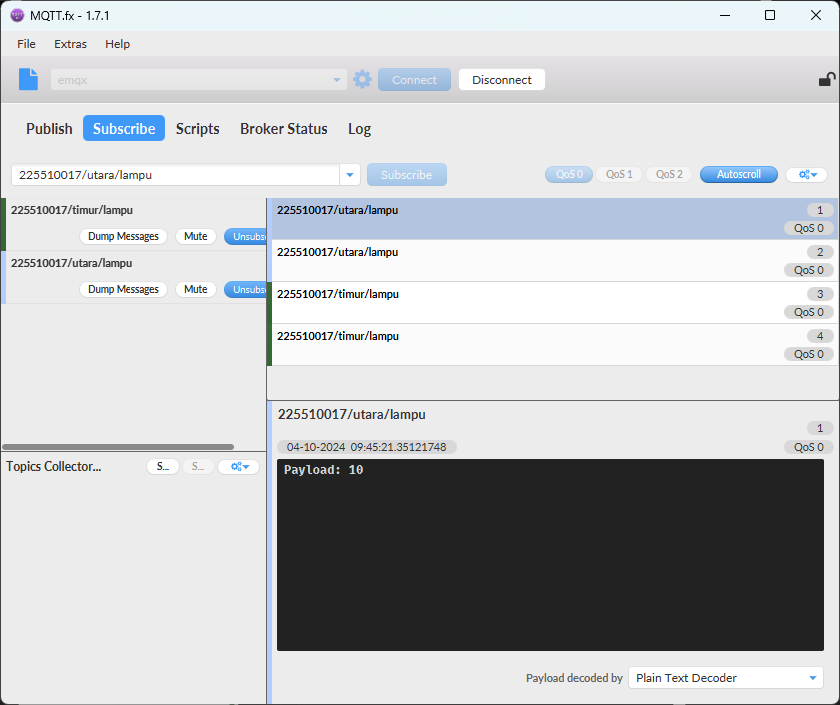
* Tulis program berikut ini dan upload terus dievaluasi. Coba dari Mqtt.Fix untuk publish sesuai Subcribe pada method reconnect();

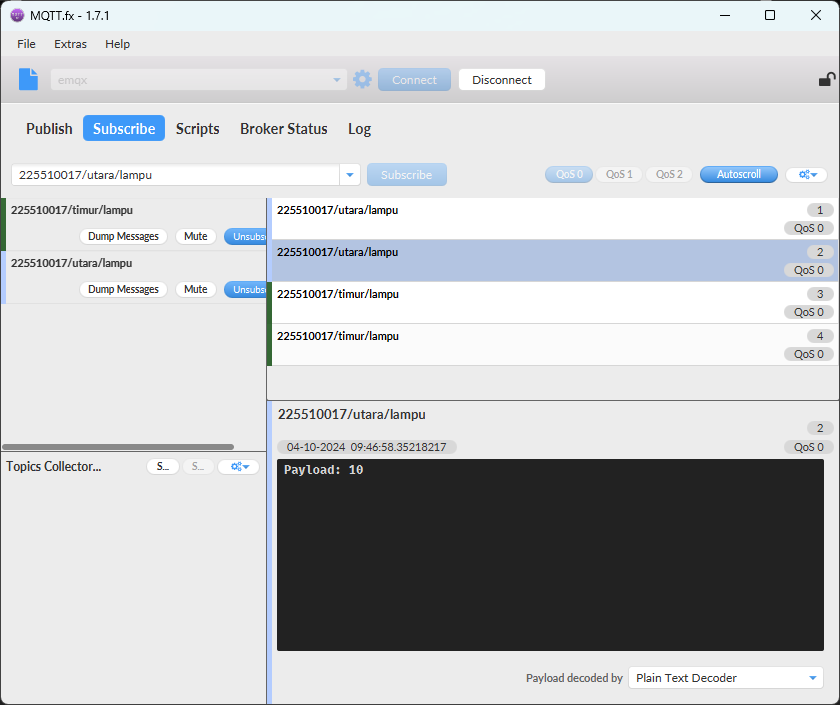
|  |
| --- |
| #include <WiFi.h> |
| #include <PubSubClient.h> |
| #define LED1 19 |
| #define LED2 18 |
| #define LED3 5 |
| #define LED4 17 |
| const char\* ssid = "RPLA\_5"; |
| const char\* password = "utdijogja"; |
| const char\* mqtt\_server = "broker.mqtt-dashboard.com"; |
| WiFiClient espClient; |
| PubSubClient client(espClient); |
| char msg[50]; |
| String tpk = String(50); |
|  |
| void setup\_wifi() { |
| delay(10); |
| Serial.println(); |
| Serial.print("Connecting to "); |
| Serial.println(ssid); |
| WiFi.begin(ssid, password); |
| while (WiFi.status() != WL\_CONNECTED) { |
| delay(500); |
| Serial.print("."); |
| } |
| randomSeed(micros()); |
| Serial.println(""); |
| Serial.println("WiFi connected"); |
| Serial.println("IP address: "); |
| Serial.println(WiFi.localIP()); |
| } |
|  |
| void callback(char\* topic, byte\* payload, unsigned int length) { |
| String action; |
| tpk = topic; |
|  |
| if (tpk == "yogya/utara/lampu") { |
| switch (char(payload[0])) { |
| case '1': |
| action = (char(payload[1]) == '1') ? "LED1 ON" : "LED1 OFF"; |
| digitalWrite(LED1, (char(payload[1]) == '1') ? HIGH : LOW); |
| break; |
| case '2': |
| action = (char(payload[1]) == '1') ? "LED2 ON" : "LED2 OFF"; |
| digitalWrite(LED2, (char(payload[1]) == '1') ? HIGH : LOW); |
| break; |
| } |
| } |
|  |
| if (tpk == "225510017/timur/lampu") { |
| switch (char(payload[0])) { |
| case '1': |
| action = (char(payload[1]) == '1') ? "LED3 ON" : "LED3 OFF"; |
| digitalWrite(LED3, (char(payload[1]) == '1') ? HIGH : LOW); |
| break; |
| case '2': |
| action = (char(payload[1]) == '1') ? "LED4 ON" : "LED4 OFF"; |
| digitalWrite(LED4, (char(payload[1]) == '1') ? HIGH : LOW); |
| break; |
| } |
| } |
|  |
| Serial.print("Message arrived ["); |
| Serial.print(topic); |
| Serial.print("] Payload: "); |
| for (int i = 0; i < length; i++) { |
| Serial.print((char)payload[i]); |
| } |
| Serial.print(" Action: "); |
| Serial.println(action); |
| } |
|  |
| void setup() { |
| Serial.begin(115200); |
| pinMode(LED1, OUTPUT); |
| pinMode(LED2, OUTPUT); |
| pinMode(LED3, OUTPUT); |
| pinMode(LED4, OUTPUT); |
| setup\_wifi(); |
| client.setServer(mqtt\_server, 1883); |
| client.setCallback(callback); |
| } |
|  |
| void loop() { |
| if (!client.connected()) { |
| // Reconnect logic |
| } |
| client.loop(); |
| } |

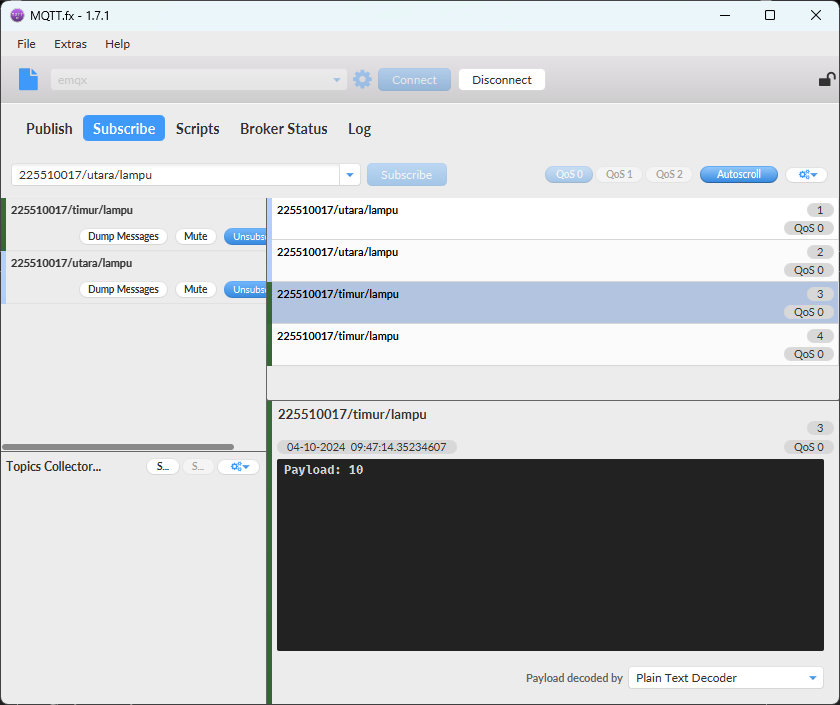


* Keluaran di MQTT

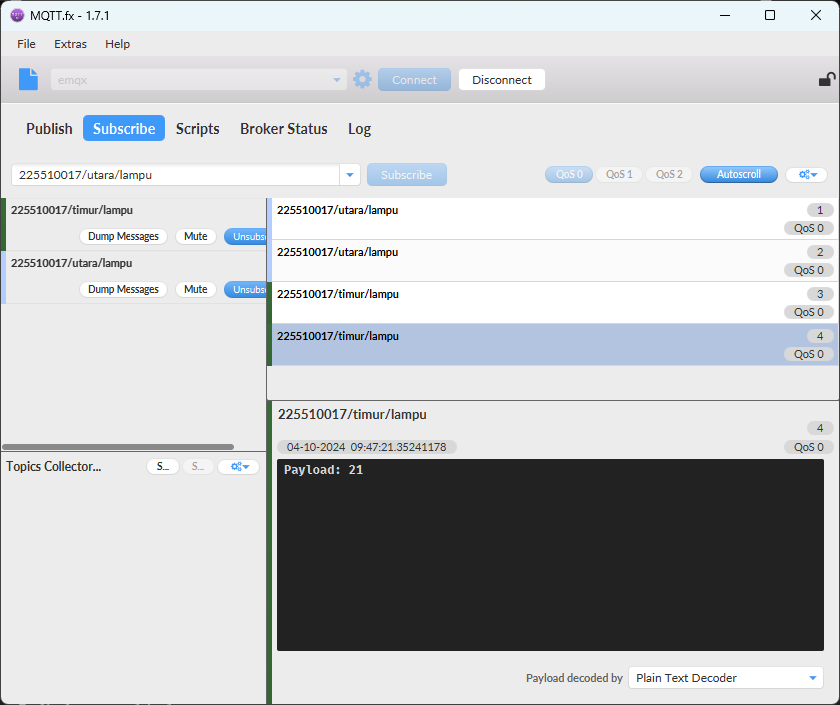
Saat lampu menyala

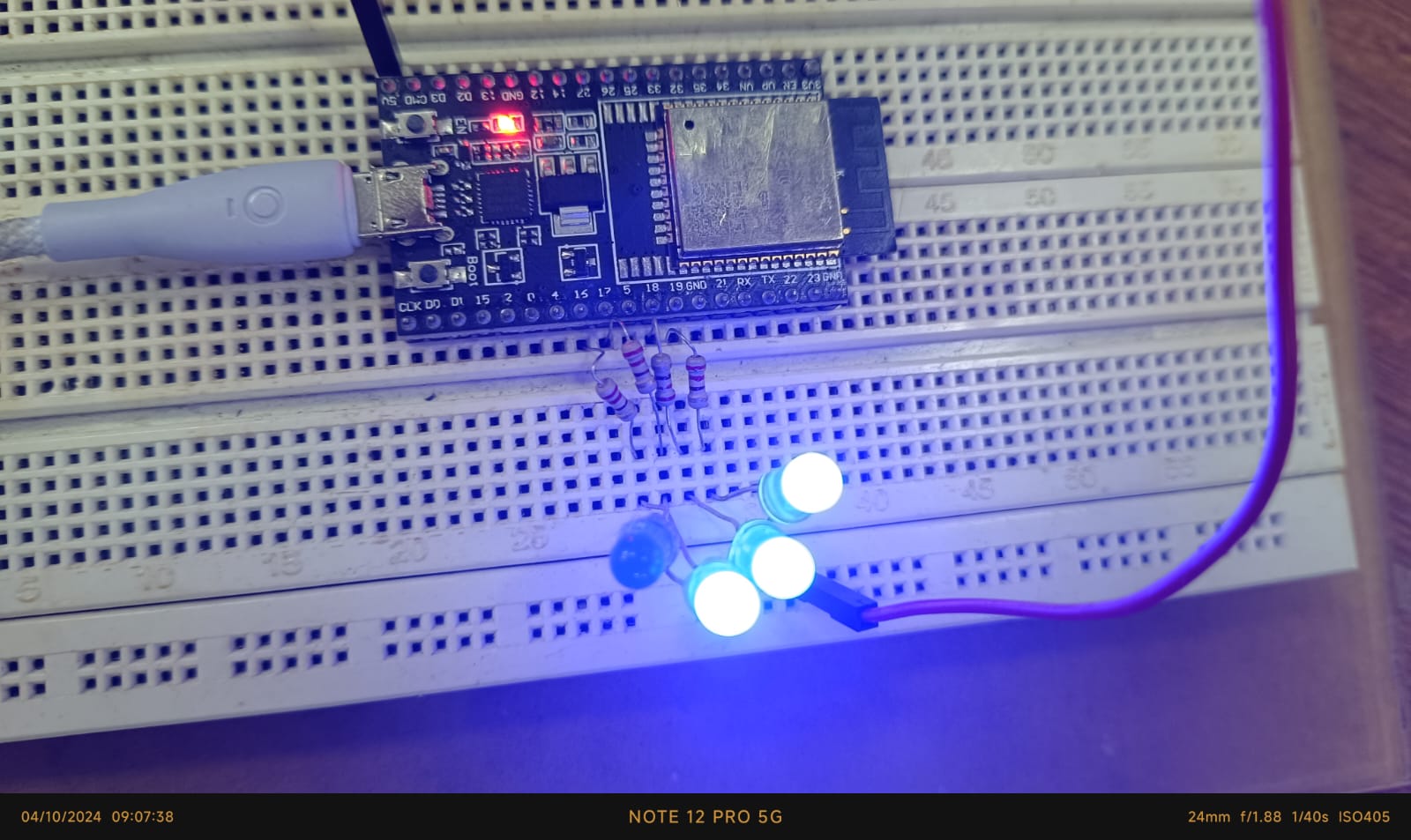






Saat memasukan Perintah untuk mematikan Lampu

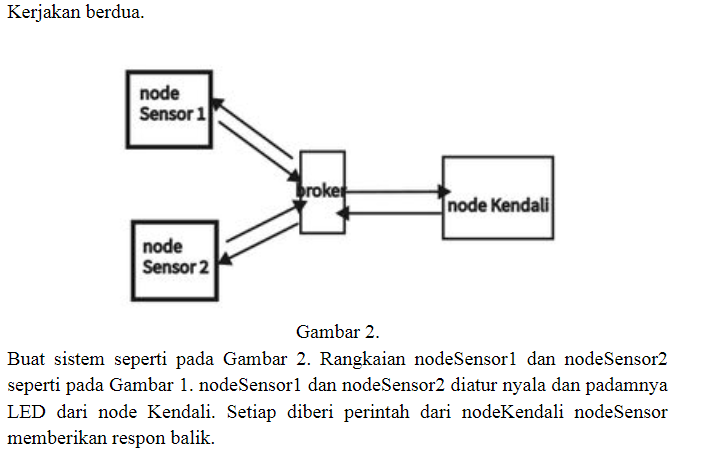




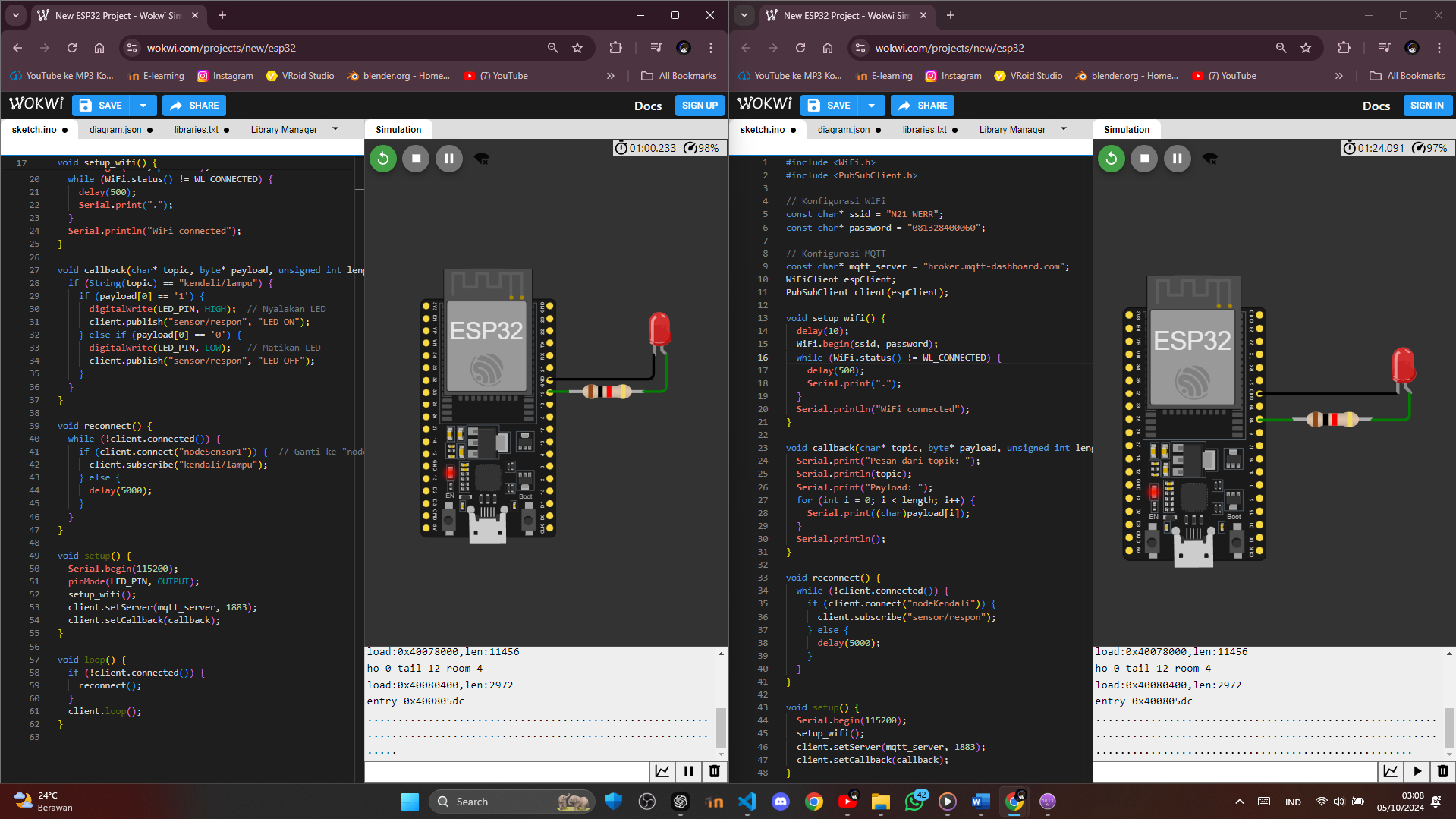
Lampu bagian kiri mati

**LATIHAN**

# Latihan 1



* Bentuk rangkaian



Ynag kiri Seneor respon 1 yang kana sensor respon 2

Link wokwi sensor respon 1 <https://wokwi.com/projects/410846457680444417>

* Code

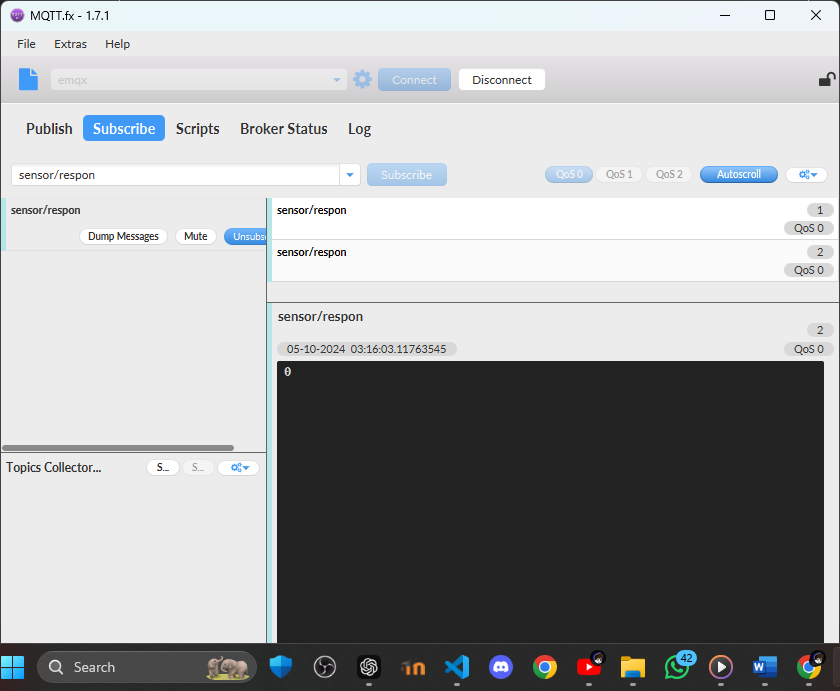
|  |
| --- |
| #include <WiFi.h> |
| #include <PubSubClient.h> |
|  |
| // Konfigurasi WiFi |
| const char\* ssid = "N21\_WERR"; |
| const char\* password = "081328400060"; |
|  |
| // Konfigurasi MQTT |
| const char\* mqtt\_server = "broker.mqtt-dashboard.com"; |
| WiFiClient espClient; |
| PubSubClient client(espClient); |
|  |
| // Pin LED nodeSensor1 (gunakan GPIO berbeda untuk nodeSensor2) |
| #define LED\_PIN 19 // Untuk nodeSensor1 |
| // #define LED\_PIN 18 // Untuk nodeSensor2 |
|  |
| void setup\_wifi() { |
| delay(10); |
| WiFi.begin(ssid, password); |
| while (WiFi.status() != WL\_CONNECTED) { |
| delay(500); |
| **Serial**.print("."); |
| } |
| **Serial**.println("WiFi connected"); |
| } |
|  |
| void callback(char\* topic, byte\* payload, unsigned int length) { |
| if (String(topic) == "kendali/lampu") { |
| if (payload[0] == '1') { |
| digitalWrite(LED\_PIN, HIGH);  // Nyalakan LED |
| client.publish("sensor/respon", "LED ON"); |
| } else if (payload[0] == '0') { |
| digitalWrite(LED\_PIN, LOW);   // Matikan LED |
| client.publish("sensor/respon", "LED OFF"); |
| } |
| } |
| } |
|  |
| void reconnect() { |
| while (!client.connected()) { |
| if (client.connect("nodeSensor1")) {  // Ganti ke "nodeSensor2" untuk node kedua |
| client.subscribe("kendali/lampu"); |
| } else { |
| delay(5000); |
| } |
| } |
| } |
|  |
| void setup() { |
| **Serial**.begin(115200); |
| pinMode(LED\_PIN, OUTPUT); |
| setup\_wifi(); |
| client.setServer(mqtt\_server, 1883); |
| client.setCallback(callback); |
| } |
|  |
| void loop() { |
| if (!client.connected()) { |
| reconnect(); |
| } |
| client.loop(); |
| } |

Link wokwi sensor respon 2 <https://wokwi.com/projects/410846540635400193>

* Code

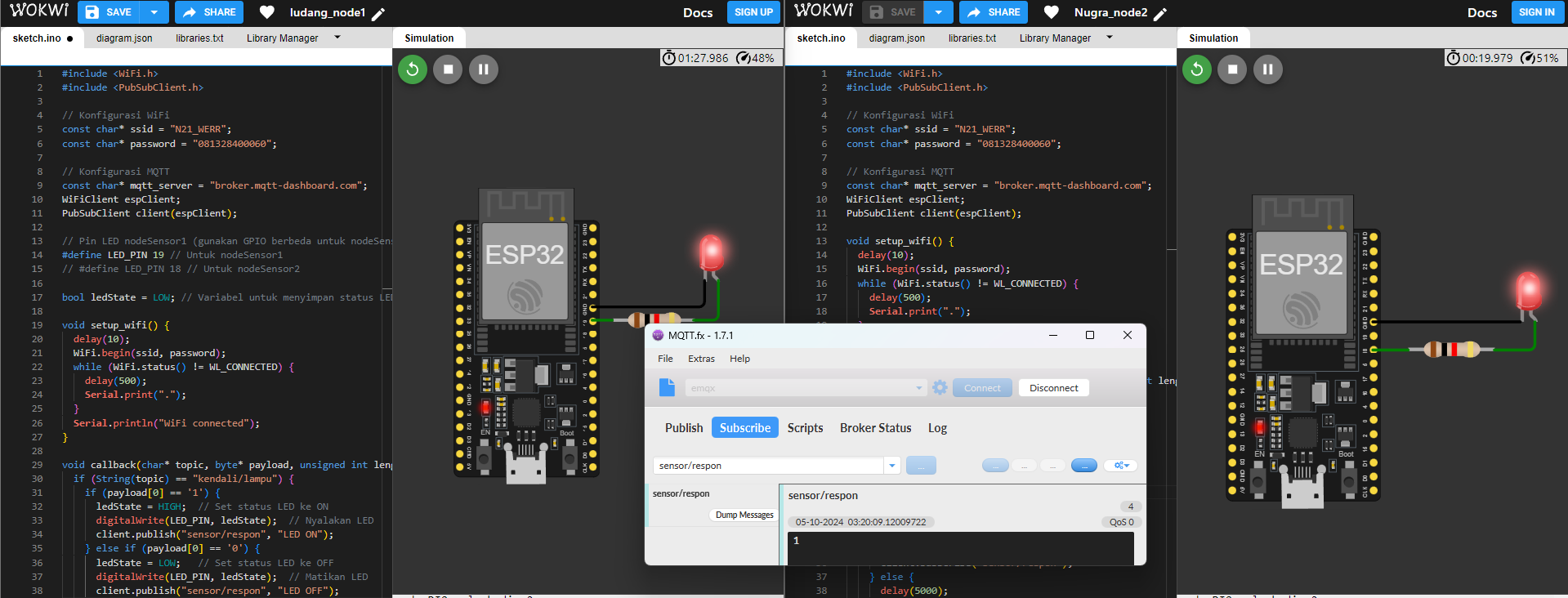
|  |
| --- |
| #include <WiFi.h> |
| #include <PubSubClient.h> |
|  |
| // Konfigurasi WiFi |
| const char\* ssid = "N21\_WERR"; |
| const char\* password = "081328400060"; |
|  |
| // Konfigurasi MQTT |
| const char\* mqtt\_server = "broker.mqtt-dashboard.com"; |
| WiFiClient espClient; |
| PubSubClient client(espClient); |
|  |
| void setup\_wifi() { |
| delay(10); |
| WiFi.begin(ssid, password); |
| while (WiFi.status() != WL\_CONNECTED) { |
| delay(500); |
| **Serial**.print("."); |
| } |
| **Serial**.println("WiFi connected"); |
| } |
|  |
| void callback(char\* topic, byte\* payload, unsigned int length) { |
| **Serial**.print("Pesan dari topik: "); |
| **Serial**.println(topic); |
| **Serial**.print("Payload: "); |
| for (int i = 0; i < length; i++) { |
| **Serial**.print((char)payload[i]); |
| } |
| **Serial**.println(); |
| } |
|  |
| void reconnect() { |
| while (!client.connected()) { |
| if (client.connect("nodeKendali")) { |
| client.subscribe("sensor/respon"); |
| } else { |
| delay(5000); |
| } |
| } |
| } |
|  |
| void setup() { |
| **Serial**.begin(115200); |
| setup\_wifi(); |
| client.setServer(mqtt\_server, 1883); |
| client.setCallback(callback); |
| } |
|  |
| void loop() { |
| if (!client.connected()) { |
| reconnect(); |
| } |
|  |
| // Kirim perintah nyala/mati ke node sensor |
| client.publish("kendali/lampu", "1");  // Perintah nyalakan LED |
| delay(5000);  // Tunggu 5 detik |
| client.publish("kendali/lampu", "0");  // Perintah matikan LED |
| delay(5000); |
|  |
| client.loop(); |
| } |

* MQTT

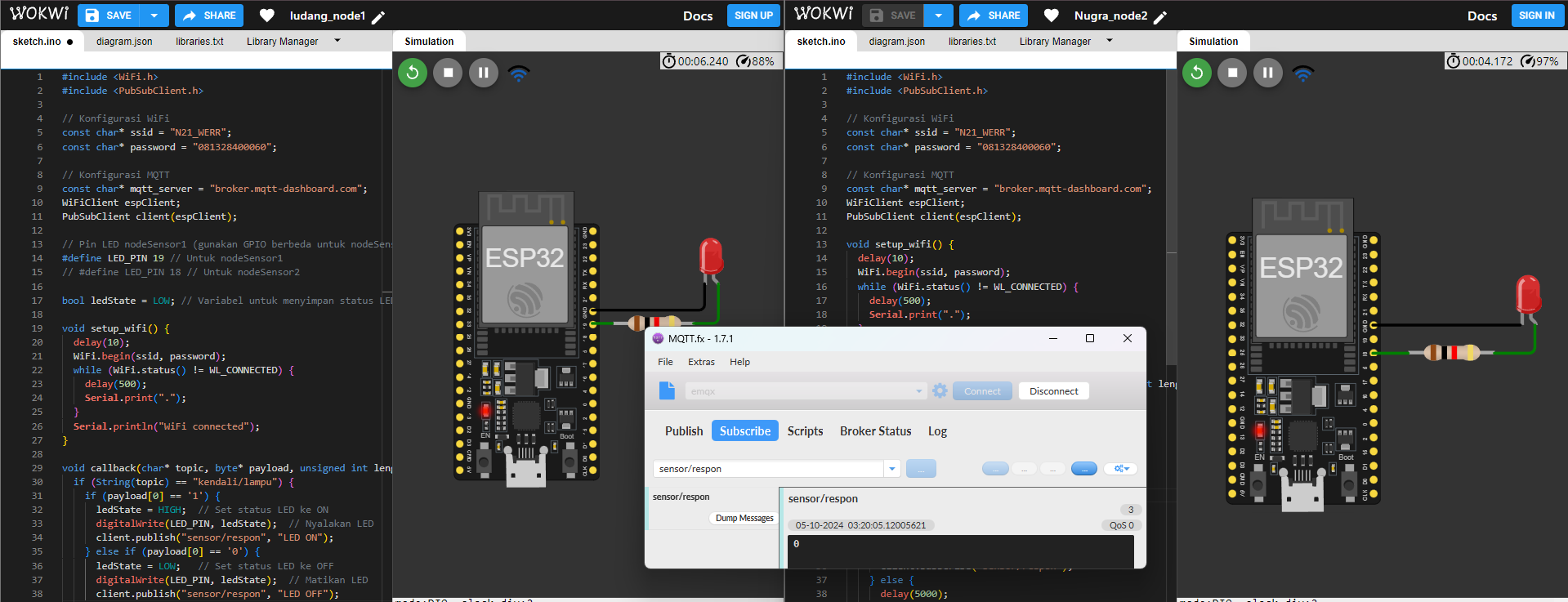


Subscribe ke sensor/respon

* Saat memasukan angka ( 1 ) 🡨 Untuk mematikan LED



* Saat memasukan angka ( 0 ) 🡨 Untuk Menyalakan LED



**TUGAS**

# Tugas

* Diagram Alir

