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

Nim : 225510017

Matkul : Prak System IOT

**# PRAKTEK**

**A. Scan Access Point Yang Ada**

1. Code program

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | #include "WiFi.h" | | void setup() | | { | | Serial.begin(115200); | | WiFi.mode(WIFI\_STA);//station | | WiFi.disconnect();// disconnected | | delay(100); | | Serial.println("Setup selesai"); | | } | | void loop() | | { | | Serial.println("Scan dimulai"); | |  | | int n = WiFi.scanNetworks(); //mulai scan | | Serial.println("Scan selesai"); | | if (n == 0) { | | Serial.println("Tidak ada Wifi"); | | } else { | | Serial.print(n); | | Serial.println(" jaringan terciduk"); | | Serial.println("Nr | SSID| RSSI | CH | Encryption"); | | for (int i = 0; i < n; ++i) { | | Serial.printf("%2d",i + 1); | | Serial.print(" | "); | | Serial.printf("%-32.32s",WiFi.SSID(i).c\_str()); | | Serial.print(" | "); | | Serial.printf("%4d", WiFi.RSSI(i)); | | Serial.print(" | "); | | Serial.printf("%2d", WiFi.channel(i)); | | Serial.print(" | "); | | switch (WiFi.encryptionType(i)) | | { | | case WIFI\_AUTH\_OPEN: | | Serial.print("open"); | | break; | | case WIFI\_AUTH\_WEP: | | Serial.print("WEP"); | | break; | | case WIFI\_AUTH\_WPA\_PSK: | | Serial.print("WPA"); | | break; | | case WIFI\_AUTH\_WPA2\_PSK: | | Serial.print("WPA2"); | | break; | | case WIFI\_AUTH\_WPA\_WPA2\_PSK: | | Serial.print("WPA+WPA2"); | | break; | | case WIFI\_AUTH\_WPA2\_ENTERPRISE: | | Serial.print("WPA2-EAP"); | | break; | | case WIFI\_AUTH\_WPA3\_PSK: | | Serial.print("WPA3"); | | } | | break; | | case WIFI\_AUTH\_WPA2\_WPA3\_PSK: | | Serial.print("WPA2+WPA3"); | | break; | | case WIFI\_AUTH\_WAPI\_PSK: | | Serial.print("WAPI"); | | break; | | default: | | Serial.print("unknown"); | | } | | Serial.println(); | | delay(10); | | } | | } | | Serial.println(""); | | WiFi.scanDelete(); | |  | | delay(5000);  } | |

2. . Catat access point yang paling kuat signalnya!

Scan dimulai

Scan selesai

5 jaringan terciduk

Nr | SSID| RSSI | CH | Encryption

1 | RPLA\_2.4 | -50 | 3 | WPA2

2 | harjolukito | -59 | 5 | WPA2

3 | UTDI (Lab-Akuntansi) | -79 | 1 | open

4 | Interaktif\_2,4 | -90 | 1 | WPA2

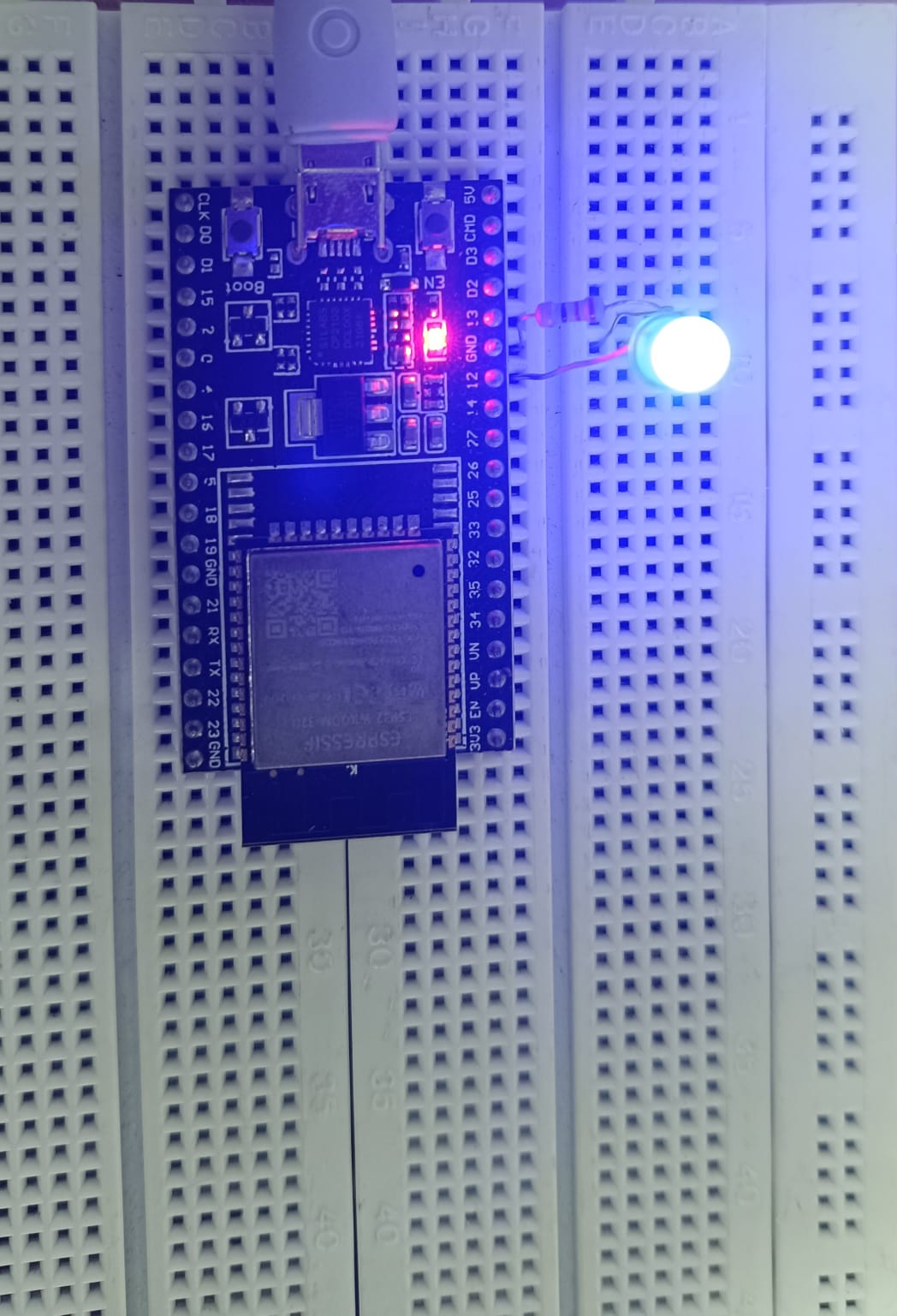
5 | IoT\_2.4 | -93 | 3 | WPA2



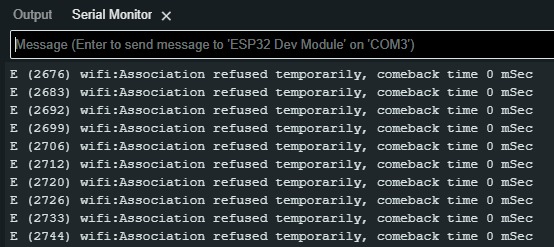
**B. Koneksi ke Access Point**

- Code program

|  |
| --- |
| #include "WiFi.h" |
| #define LEDStatus 12 |
| **const char\* ssid = "RPLA\_2.4";** |
| **const char\* password = "utdijogja";** |
| void initWiFi() { |
| WiFi.mode(WIFI\_STA); |
| WiFi.begin(ssid, password); |
| Serial.print("Connecting to WiFi .."); |
| while (WiFi.status() != WL\_CONNECTED) { |
| Serial.print('.'); |
| delay(1000); |
| } |
| Serial.print("IP Address: "); |
| Serial.println(WiFi.localIP(); |
| } |
| void setup() |
| { |
| Serial.begin(115200); |
| pinMode(LEDStatus,HIGH); |
| initWiFi(); |
| } |
| void loop() |
| { |
|  |
| } |



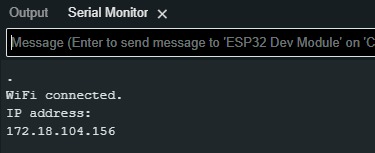
**C. Konfigurasi Interface Secara Statis**

****

1. Code rogram

|  |
| --- |
|  |
| #include "WiFi.h" |
| const char\* ssid = "RPLA\_2.4"; |
| const char\* password = "utdijogja"; |
| //Set IPAddrress sesuaikan dengan jaringan lokal |
| IPAddress local\_IP(192, 168, 56, 1); |
| IPAddress gateway(17,18, 104, 254); |
| IPAddress subnet(255, 255, 255, 0); |
| IPAddress primaryDNS(8, 8, 8, 8); |
| void setup() |
| { |
| Serial.begin(115200); |
| WiFi.mode(WIFI\_STA);//station |
| Serial.print("Connecting to "); |
| Serial.println(ssid); |
| WiFi.disconnect(); |
| WiFi.begin(ssid, password); |
| while (WiFi.status() != WL\_CONNECTED) { |
| delay(500); |
| Serial.print("."); |
| } |
| Serial.println(""); |
| Serial.println("WiFi connected."); |
| Serial.println("IP address: "); |
| Serial.println(WiFi.localIP()); |
| } |
| void loop() |
| { |
| } |
|  |

2. Cek koneksi dari PC kel ESP32 menggunakan ping.



**# LATIHAN**

1. Buat program untuk menentukan Access Point yang paling kuat signalnya.

|  |
| --- |
| import subprocess |
| import re |
|  |
| def scan\_access\_points(): |
| # Melakukan pemindaian jaringan WiFi |
| result = subprocess.run(['nmcli', 'dev', 'wifi'], capture\_output=True, text=True) |
| return result.stdout |
|  |
| def parse\_access\_points(data): |
| ap\_list = [] |
| for line in data.split('\n')[1:]: |
| if line.strip(): |
| parts = re.split(r'\s+', line.strip()) |
| ssid = parts[0] |
| signal = int(parts[7]) # Indeks signal berada di 7 |
| ap\_list.append((ssid, signal)) |
| return ap\_list |
|  |
| def get\_strongest\_ap(ap\_list): |
| return max(ap\_list, key=lambda x: x[1]) |
|  |
| data = scan\_access\_points() |
| ap\_list = parse\_access\_points(data) |
| strongest\_ap = get\_strongest\_ap(ap\_list) |
|  |
| print(f"Access Point Terkuat: {strongest\_ap[0]} dengan sinyal {strongest\_ap[1]}%") |

2. Buat program (atau menambahkan Latihan 1) agar melakukan koneksi ulang jika terputus.

|  |
| --- |
| import time |
|  |
| def connect\_to\_ap(ssid): |
| subprocess.run(['nmcli', 'dev', 'wifi', 'connect', ssid]) |
|  |
| def check\_connection(): |
| result = subprocess.run(['nmcli', 'dev'], capture\_output=True, text=True) |
| return 'connected' in result.stdout |
|  |
| while True: |
| if not check\_connection(): |
| print("Koneksi terputus, mencoba menyambung ulang...") |
| connect\_to\_ap(strongest\_ap[0]) |
| time.sleep(5) # Tunggu beberapa detik sebelum memeriksa kembali |
| else: |
| print("Koneksi aktif.") |
| time.sleep(10) # Periksa status setiap 10 detik |
|  |

3. Tambahkan kode program ke latihan 2 untuk indikator status Wifi (terutama connect dan disconnect)

|  |
| --- |
| def check\_wifi\_status(): |
| if check\_connection(): |
| print(f"Koneksi aktif dengan Access Point: {strongest\_ap[0]}") |
| else: |
| print("Tidak terhubung ke WiFi.") |
|  |
| while True: |
| check\_wifi\_status() |
| if not check\_connection(): |
| print("Koneksi terputus, mencoba menyambung ulang...") |
| connect\_to\_ap(strongest\_ap[0]) |
| time.sleep(10) # Periksa status setiap 10 detik |
|  |

**# TUGAS**

1. Buat diagram alir untuk hasil latihan.

