

# LAPORAN

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )

Nama : Ludang Prasetyo Nugroho  
Nim : 225510017  
Matkul : Prak System IOT

[Teknik Komputer \( S1 \)](#)

---

## # 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;
```

# LAPORAN

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )

```
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();

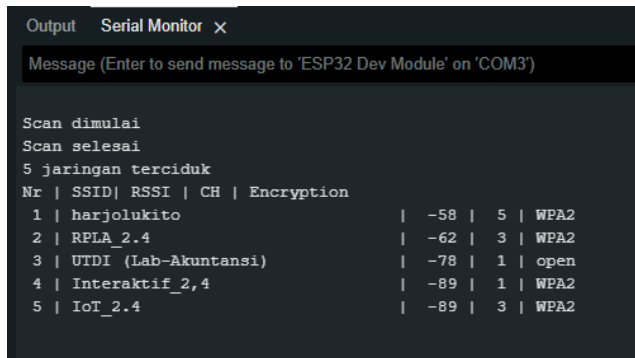
delay(5000);
}
```

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
```

# LAPORAN

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )



```
Output Serial Monitor x
Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')

Scan dimulai
Scan selesai
5 jaringan terciduk
Nr | SSID| RSSI | CH | Encryption
1 | harjolukito | -58 | 5 | WPA2
2 | RPLA_2.4 | -62 | 3 | WPA2
3 | UTDI (Lab-Akuntansi) | -78 | 1 | open
4 | Interaktif_2,4 | -89 | 1 | WPA2
5 | IoT_2.4 | -89 | 3 | WPA2
```

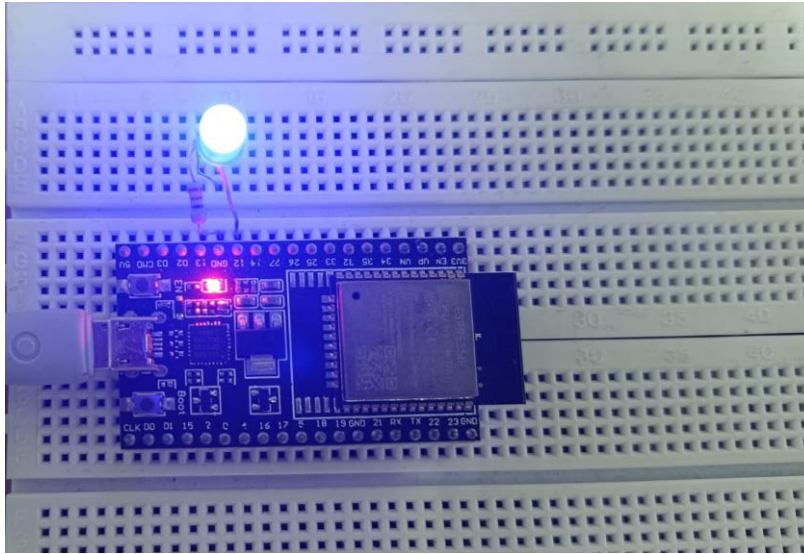
## B. Koneksi ke Access Point

- 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()
{
}
```

# LAPORAN

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )



## C. Konfigurasi Interface Secara Statis

```
Output  Serial Monitor x
Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')

E (2676) wifi:Association refused temporarily, comeback time 0 mSec
E (2683) wifi:Association refused temporarily, comeback time 0 mSec
E (2692) wifi:Association refused temporarily, comeback time 0 mSec
E (2699) wifi:Association refused temporarily, comeback time 0 mSec
E (2706) wifi:Association refused temporarily, comeback time 0 mSec
E (2712) wifi:Association refused temporarily, comeback time 0 mSec
E (2720) wifi:Association refused temporarily, comeback time 0 mSec
E (2726) wifi:Association refused temporarily, comeback time 0 mSec
E (2733) wifi:Association refused temporarily, comeback time 0 mSec
E (2744) wifi:Association refused temporarily, comeback time 0 mSec
```

### 1. Code rogram

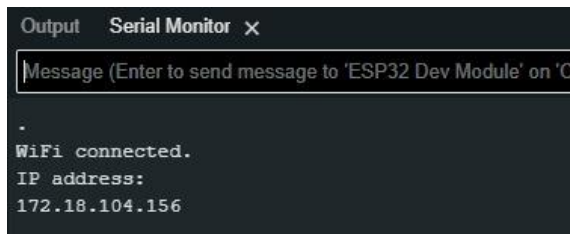
```
#include "WiFi.h"
const char* ssid = "RPLA_2.4";
const char* password = "utdijogja";
//Set IPAddress sesuaikan dengan jaringan lokal
IPAddress local_IP(192, 168, 56, 1);
IPAddress gateway(17,18, 104, 254);
IPAddress subnet(255, 255, 255, 0);
IPAddress primaryDNS(8, 8, 8, 8);
void setup()
{
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);//station
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.disconnect();
```

# LAPORAN

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )

```
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 ke ESP32 menggunakan ping.



The screenshot shows a 'Serial Monitor' window with a title bar that includes 'Output', 'Serial Monitor', and a close button. The input field contains the text 'Message (Enter to send message to 'ESP32 Dev Module' on 'C'. The output area displays the following text: a single dot on the first line, 'WiFi connected.' on the second line, 'IP address:' on the third line, and '172.18.104.156' on the fourth line.

## # LATIHAN

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

```
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))
```

# LAPORAN

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )

```
    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])
```

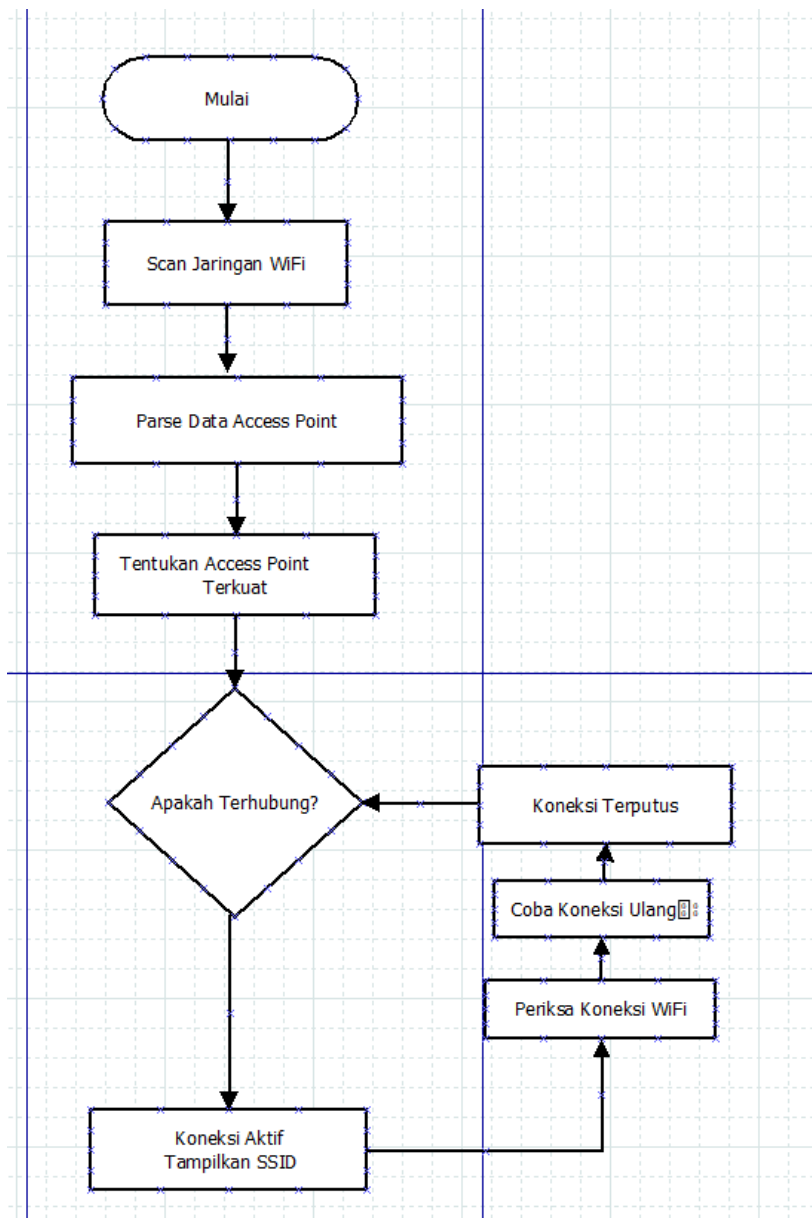
# LAPORAN

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )

```
time.sleep(10) # Periksa status setiap 10 detik
```

## # TUGAS

1. Buat diagram alir untuk hasil latihan.



# **LAPORAN**

Laporan praktikum Systemn Internet Of Thinks ( Jumat 20 September )