

LAPORAN PRAKTIK PENGUJIAN ESP32 DENGAN REAL HARDWARE

Yasiva Nurul Ramadhan – 233140707111067 – T4A – Teknologi Informasi

Fakultas Vokasi, Universitas Brawijaya

Email: ysvlhn@student.ub.ac.id

Praktikum ini membahas penggunaan ESP32 secara langsung (real hardware) untuk menguji pengendalian LED dan pemindaian jaringan Wi-Fi. Tujuan dari percobaan ini adalah memahami proses upload kode dari VSCode melalui PlatformIO ke ESP32 serta mengecek keberhasilan koneksi dan pengendalian perangkat melalui kode yang ditulis. Hasil praktikum menunjukkan bahwa ESP32 dapat dikenali oleh komputer dan berhasil menjalankan fungsinya, baik untuk menyalakan LED maupun mendeteksi jaringan Wi-Fi di sekitarnya.

Kata kunci— ESP32, LED, Wi-Fi, PlatformIO, Real Hardware

1. Pendahuluan

1.1 Latar belakang

ESP32 adalah salah satu mikrokontroler populer dalam pengembangan proyek IoT. Dalam tahap pengembangan sistem berbasis IoT, penting untuk memastikan bahwa perangkat keras dapat berfungsi dengan baik dan dapat berkomunikasi dengan komputer serta jaringan internet. Melalui praktikum ini, dilakukan pengujian terhadap pengendalian perangkat (dalam hal ini LED) dan pemindaian jaringan Wi-Fi menggunakan ESP32 secara langsung melalui PlatformIO.

1.2 Tujuan Eksperimen

Praktikum ini bertujuan untuk:

1. Memastikan ESP32 dapat dikenali oleh komputer.
2. Menguji pengendalian LED melalui program yang ditulis di VSCode.
3. Menguji konektivitas ESP32 terhadap jaringan Wi-Fi.

2. Metodologi

2.1 Alat dan Bahan

Dalam praktikum ini, alat dan bahan yang digunakan meliputi:

1. Hardware ESP32
2. Kabel USB Micro
3. Komputer/Laptop
4. Visual Studio Code dengan PlatformIO

5. Breadboard dan LED
6. Akses internet untuk koneksi Wi-Fi

2.2 Langkah Implementasi

Berikut langkah-langkah yang dilakukan dalam praktikum ini:

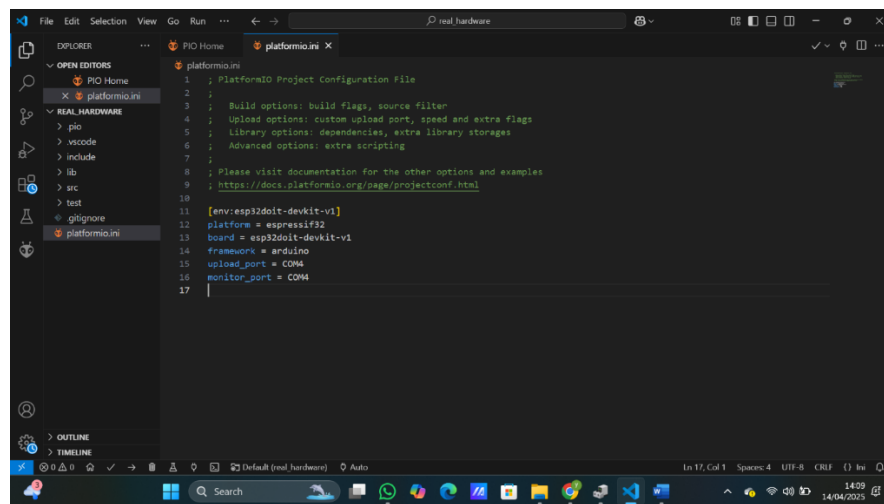
1. Menyambungkan ESP32 ke komputer dan memastikan device dikenali (Silicon Labs CP210x driver).
2. Membuat folder project baru di VSCode menggunakan PlatformIO.
3. Mengatur file platformio.ini agar sesuai dengan device (COM4).
4. Menulis program LED di main.cpp dan melakukan upload ke ESP32.
5. Menulis program scanning Wi-Fi dan melakukan upload kembali.
6. Melihat hasil pada serial monitor.

3. Hasil Dan Pembahasan

3.1 Hasil Eksperimen

Hasil dari praktikum ini adalah sebagai berikut:

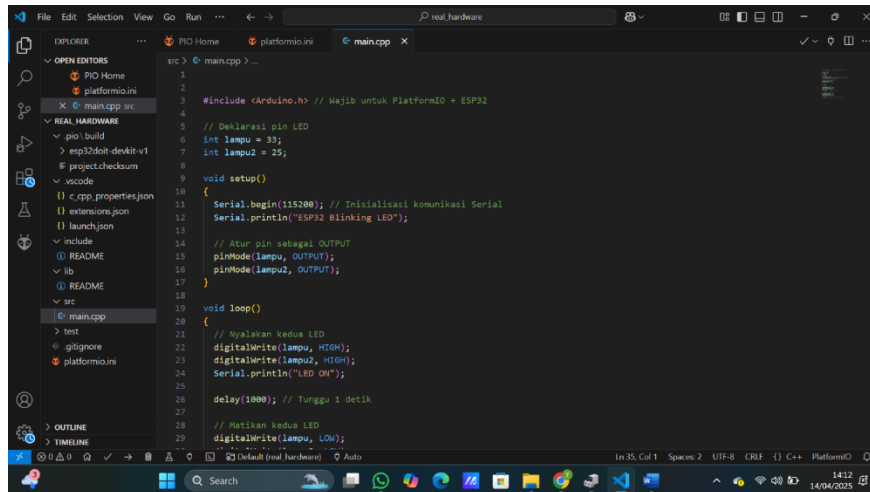
1. Buat project baru di platformio. Ubah file platformio.ini dan modifikasi sebagai berikut



The screenshot shows the VS Code editor with the PlatformIO extension. The Explorer sidebar on the left shows the project structure with 'platformio.ini' selected. The main editor window displays the content of 'platformio.ini'. The file is a PlatformIO Project Configuration File. It includes comments for build options, upload options, library options, and advanced options. The configuration is set for an ESP32 using the espressif32 platform, the esp32doit-devkit-v1 board, and the arduino framework. The upload port is set to COM4 and the monitor port is also set to COM4.

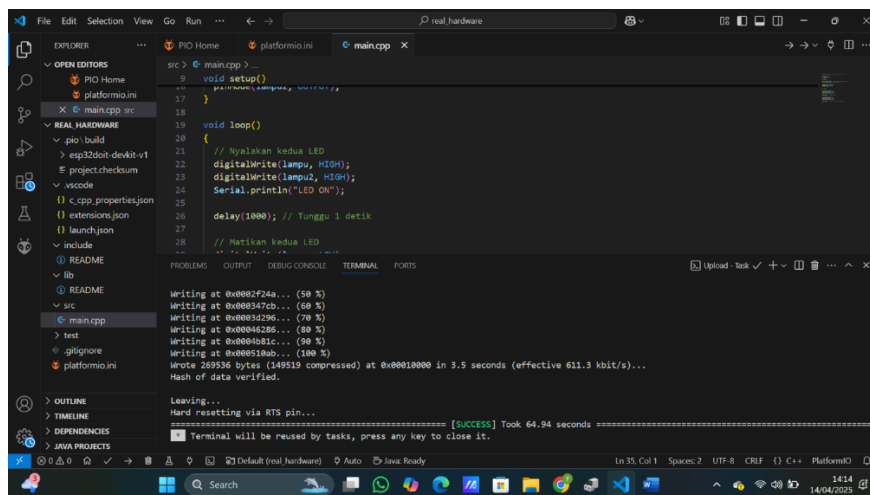
```
1 ; PlatformIO Project Configuration File
2 ;
3 ; Build options: build flags, source filter
4 ; Upload options: custom upload port, speed and extra flags
5 ; Library options: dependencies, extra library storages
6 ; Advanced options: extra scripting
7 ;
8 ; Please visit documentation for the other options and examples
9 ; https://docs.platformio.org/page/projectconf.html
10
11 [env:esp32doit-devkit-v1]
12 platform = espressif32
13 board = esp32doit-devkit-v1
14 framework = arduino
15 upload_port = COM4
16 monitor_port = COM4
17
```

-
2. Pada file main.cpp masukkan koding lampu LED yang telah dibuat di bab 8



```
1 2
3 #include <Arduino.h> // Wajib untuk PlatformIO + ESP32
4
5 // Deklarasi pin LED
6 int lampu = 33;
7 int lampu2 = 25;
8
9 void setup()
10 {
11   Serial.begin(115200); // Inisialisasi komunikasi Serial
12   Serial.println("ESP32 Blinking LED");
13
14   // Atur pin sebagai OUTPUT
15   pinMode(lampu, OUTPUT);
16   pinMode(lampu2, OUTPUT);
17 }
18
19 void loop()
20 {
21   // Nyalakan kedua LED
22   digitalWrite(lampu, HIGH);
23   digitalWrite(lampu2, HIGH);
24   Serial.println("LED ON");
25
26   delay(1000); // Tunggu 1 detik
27
28   // Matikan kedua LED
29   digitalWrite(lampu, LOW);
```

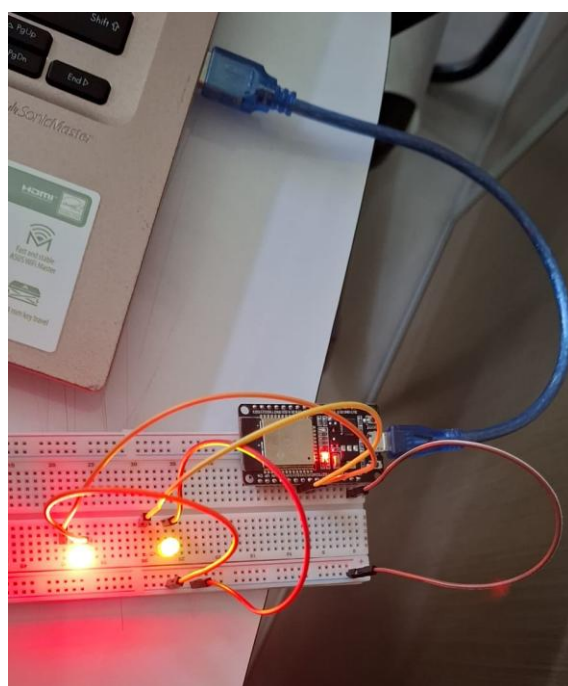
-
-
3. Lakukan Upload pada menu Upload



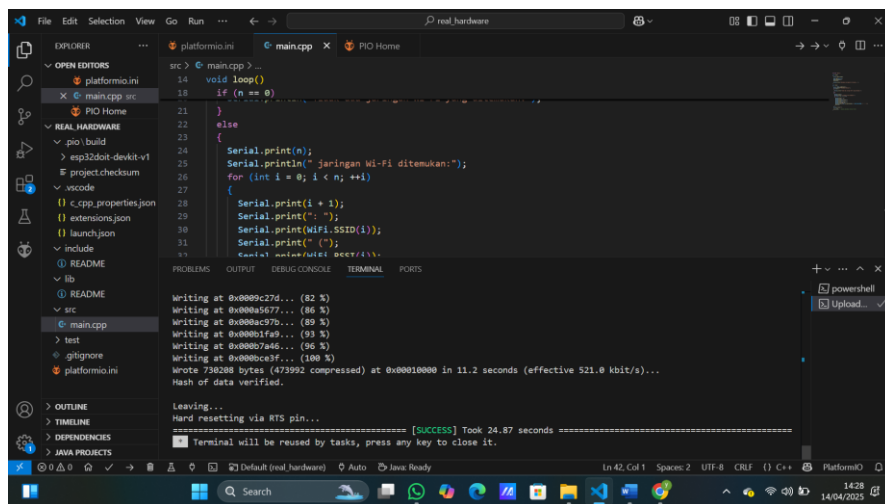
```
9 void setup()
10 {
11   Serial.begin(115200); // Inisialisasi komunikasi Serial
12   Serial.println("ESP32 Blinking LED");
13
14   // Atur pin sebagai OUTPUT
15   pinMode(lampu, OUTPUT);
16   pinMode(lampu2, OUTPUT);
17 }
18
19 void loop()
20 {
21   // Nyalakan kedua LED
22   digitalWrite(lampu, HIGH);
23   digitalWrite(lampu2, HIGH);
24   Serial.println("LED ON");
25
26   delay(1000); // Tunggu 1 detik
27
28   // Matikan kedua LED
29   digitalWrite(lampu, LOW);
```

Writing at 0x000024a... (50 %)
Writing at 0x000047cb... (60 %)
Writing at 0x00004296... (70 %)
Writing at 0x00004286... (80 %)
Writing at 0x00004082... (90 %)
Writing at 0x000018ab... (100 %)
Wrote 269536 bytes (149519 compressed) at 0x00010000 in 3.5 seconds (effective 611.3 kbit/s)...
Hash of data verified.

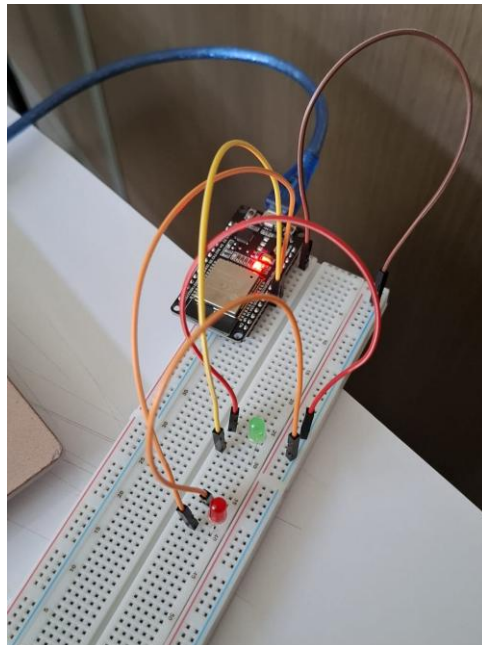
Leaving...
Hard resetting via RTS pin...
[SUCCESS] Took 64.94 seconds
Terminal will be reused by tasks, press any key to close it.



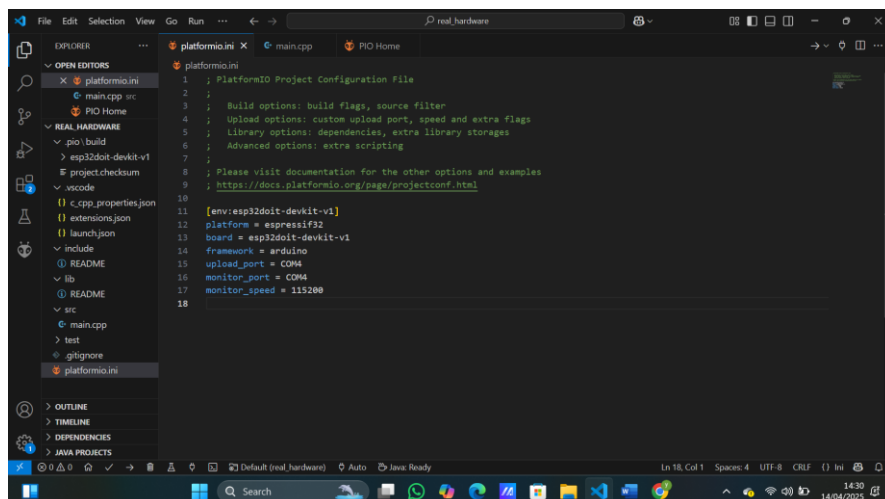
4. Mengecek Koneksi WIFI pada Hardware ESP32



```
platformio.ini  main.cpp  PIO Home
src > main.cpp > ...
14 void loop()
15 {
16     if (n == 0)
17     {
18         Serial.print(n);
19         Serial.println(" Jaringan Wi-Fi ditemukan!");
20         for (int i = 0; i < n; ++i)
21         {
22             Serial.print(i + 1);
23             Serial.print(": ");
24             Serial.print(WiFi.SSID(i));
25             Serial.print(" ");
26             Serial.print("\n");
27             Serial.println(WiFi.BSSID(i));
28         }
29     }
30     n++;
31     delay(1000);
32 }
33
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Writing at 0x00000000... (82 %)
Writing at 0x00000001... (86 %)
Writing at 0x00000002... (89 %)
Writing at 0x00000003... (93 %)
Writing at 0x00000004... (96 %)
Writing at 0x00000005... (100 %)
Wrote 738208 bytes (473992 compressed) at 0x00010000 in 11.2 seconds (effective 521.0 kbit/s)...
Hash of data verified.
Leaving...
Hard resetting via RTS pin...
[SUCCESS] Took 24.87 seconds
Terminal will be reused by tasks, press any key to close it.
```

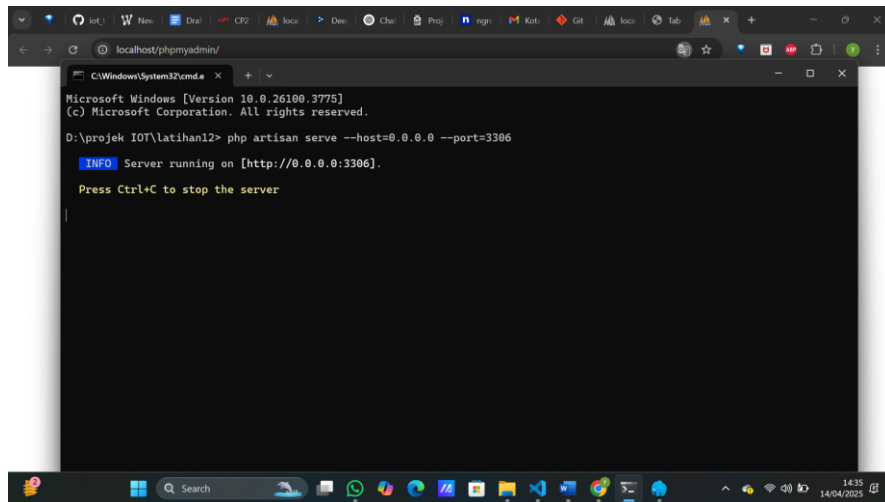


5. Ubah kembali file platformio.ini sebagai berikut

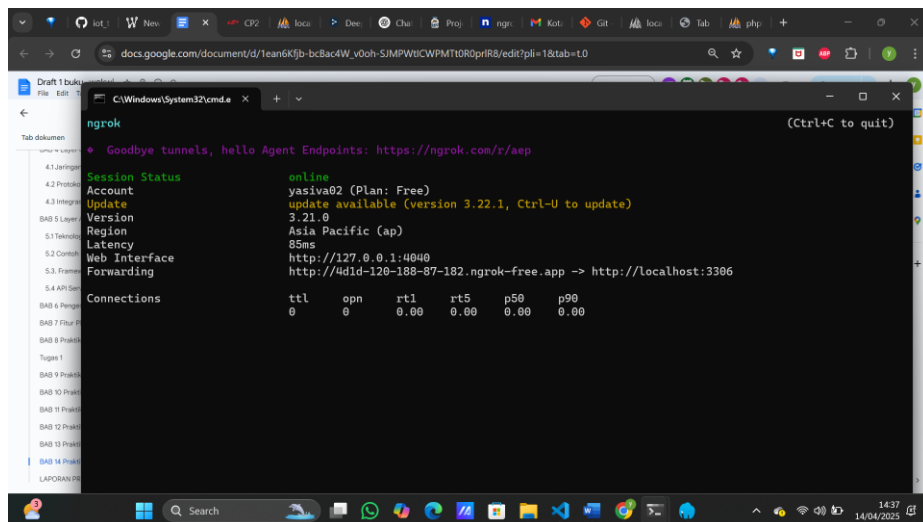


```
platformio.ini
; PlatformIO Project Configuration File
;
; Build options: build flags, source filter
; Upload options: custom upload port, speed and extra flags
; Library options: dependencies, extra library storages
; Advanced options: extra scripting
;
; Please visit documentation for the other options and examples
; https://docs.platformio.org/page/projectconf.html
[env:esp32dev]
platform = espressif32
board = esp32dev
framework = arduino
upload_port = COM4
monitor_port = COM4
monitor_speed = 115200
```

6. Pemindaian wifi berhasil



7. Pemindaian NGROK berhasil



8. Proses Berhasil

