

LAPORAN PROJECT AKHIR MIKROKONTROLER

Dosen Pengampu: Dr. Basuki Rahmat, S.Si., MT.



Disusun Oleh:

Ayudya Aldi Setiawan (23081010031)

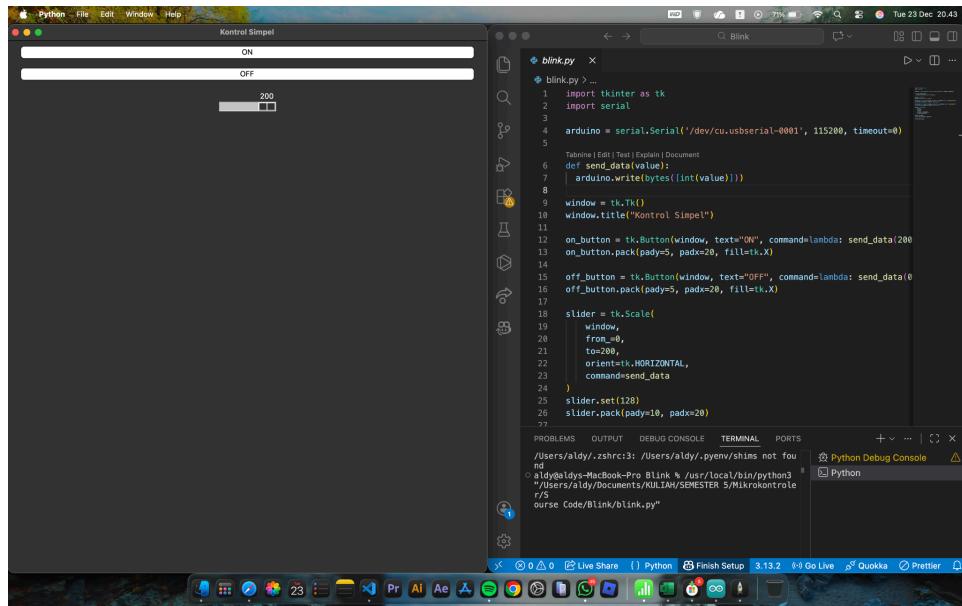
PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PEMBANGUNAN NASIONAL VETERAN
JAWA TIMUR
2025

1. Python On-Off

Fokus Utama: Komunikasi Serial UART dan Modulasi Lebar Pulsa (PWM).

Analisis Cara Kerja: Pada tahap ini, laptop berfungsi sebagai pusat kendali utama (Master). Antarmuka Python yang dibuat menggunakan library Tkinter mengirimkan sinyal perintah melalui kabel USB (Serial Communication).

- **Mekanisme:** Saat slider digeser, Python mengirimkan nilai duty cycle (0-255).
- **Respon:** Kit iMCLab menerima data tersebut dan mengkonversinya menjadi sinyal listrik PWM ke motor, sehingga kecepatan motor berubah sesuai posisi slider di layar laptop secara real-time.



Gambar 1. UI On/Off Python

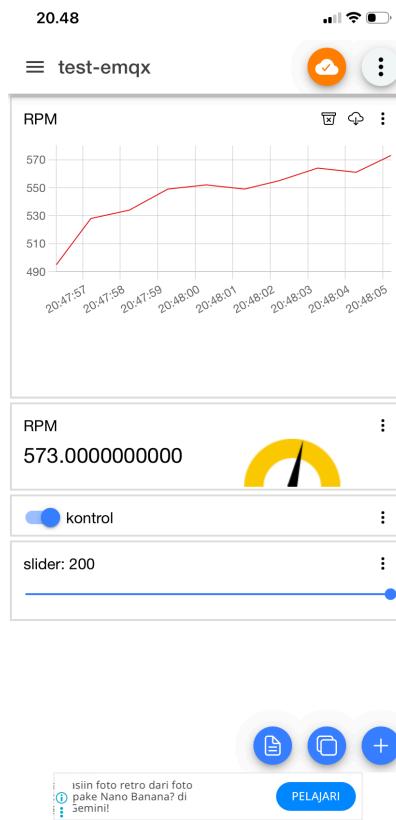
2. IoT On-Off

Fokus Utama: Penerapan Protokol Jaringan MQTT

Analisis Cara Kerja: Percobaan ini memindahkan kendali dari kabel USB ke jaringan internet nirkabel.

- **Arsitektur:** HP bertindak sebagai Publisher yang mengirim perintah ke Broker MQTT publik (EMQX). Python di laptop bertindak sebagai Subscriber yang mendengarkan perintah tersebut.
- **Alur Data:** HP - Internet - Python - Motor.

- **Hasil:** Motor dapat dikendalikan dari jarak jauh tanpa batasan fisik kabel, namun mulai terlihat adanya jeda waktu (latency) tergantung kualitas koneksi internet.



Gambar 2. Proses On/Off dari HP

3. PID Arduino

Fokus Utama: Komputasi Kendali Terdistribusi (Embedded Control).

Analisis Cara Kerja: Berbeda dengan sebelumnya, logika perhitungan PID (Proportional, Integral, Derivative) kini ditanam langsung di dalam chip mikrokontroler (ESP32).

- **Proses:** Mikrokontroler membaca sensor encoder, menghitung selisih error (Target - Aktual), dan menentukan besaran tenaga motor secara mandiri.
- **Observasi:** Melalui Serial Monitor, terlihat nilai RPM berusaha mengejar target secara otomatis. Respon sistem sangat cepat karena tidak ada delay komunikasi dengan laptop.

```

pid_arduino.ino
1 // constants
2 const int baud = 115200;           // serial baud rate
3
4 int motorPin1 = 27;
5 int motorPin2 = 26;
6 int enable1Pin = 12;
7
8 // Setting PWM properties
9 const int freq = 30000;
10 const int pwmChannel = 1;
11 const int resolution = 8;
12
13 byte pin_rpm = 13;
14 volatile unsigned long rev = 0;      // Changed to unsigned long
15 unsigned long last_rev_count = 0;    // Track last revolution count
16 unsigned long last_pmt_time = 0;     // Track last RPM calculation time
17
18 float rpm = 0;                     // Current RPM value
19 float rpm_filtered = 0;            // Filtered RPM value
20
21 unsigned long cur_time, old_time;
22
23 //float P, I, D, Kd, tauI, tauD;
24
25
26
27
28
29
30
31
32
33
34
35
36

```

Output:

```

Message (Enter to send message to 'DOIT ESP32 DEVKIT V1' on '/dev/cu.usbserial-0001')
Time: 100ms, Pulses: 0, RPM: 4998.00, Filtered RPM: 4998.00
sp=5000.00 pv=4998.00 dt=1.00 op=176.00 P=0.02 I=176.06 D=-0.00
PWM: 176
Time: 100ms, Pulses: 0, RPM: 5005.00, Filtered RPM: 4997.92
sp=5000.00 pv=5005.00 dt=1.00 op=176.03 P=-0.00 I=176.06 D=-0.02
PWM: 176
Time: 100ms, Pulses: 0, RPM: 5034.96, Filtered RPM: 5009.03
sp=5000.00 pv=5034.96 dt=1.00 op=175.96 P=-0.03 I=176.02 D=-0.03
PWM: 175
Time: 100ms, Pulses: 0, RPM: 4975.02, Filtered RPM: 4998.83
sp=5000.00 pv=4975.02 dt=1.00 op=176.13 P=0.02 I=176.05 D=0.06
PWM: 176
Time: 100ms, Pulses: 0, RPM: 4950.00, Filtered RPM: 4984.18
sp=5000.00 pv=4950.00 dt=1.00 op=176.17 P=0.05 I=176.10 D=0.03
PWM: 176

```

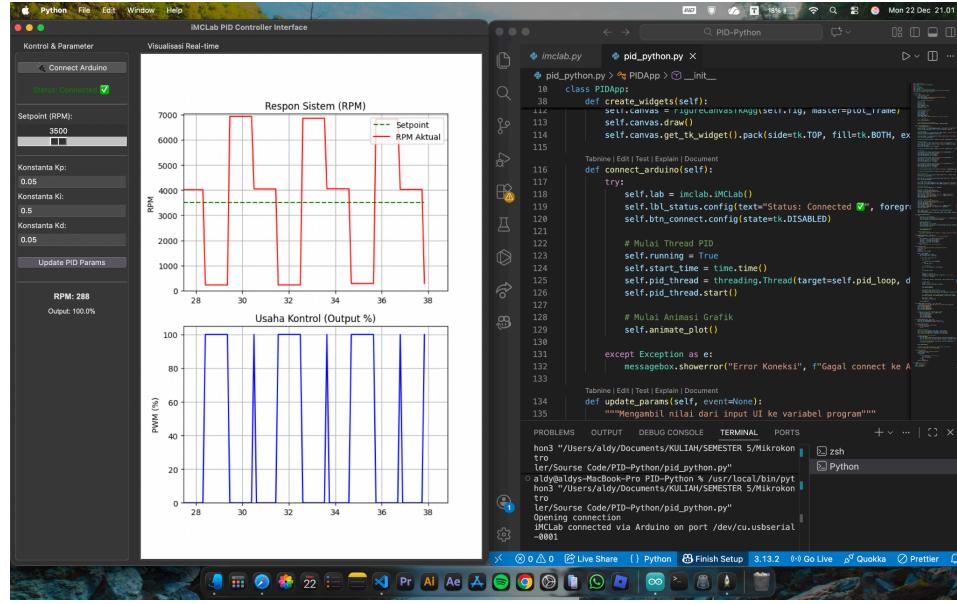
Gambar 3. Serial Monitor PID Arduino

4. PID Python

Fokus Utama: Visualisasi Respon Sistem

Analisis Cara Kerja: Logika PID dipindahkan kembali ke Python untuk memudahkan pemantauan grafik.

- **Visualisasi:** Grafik menampilkan dua garis penting: Garis Target (Setpoint) dan Garis Aktual (Respon Motor).
- **Analisis Grafik:** Terlihat adanya fenomena overshoot (motor berputar terlalu kencang di awal) dan settling time (waktu yang dibutuhkan untuk stabil) sangat lama. Hal ini membuktikan bahwa parameter PID (K_p , K_i , K_d) manual sangat mempengaruhi kestabilan sistem.



Gambar 4. UI PID dengan Python

5. PID Python Menggunakan AI

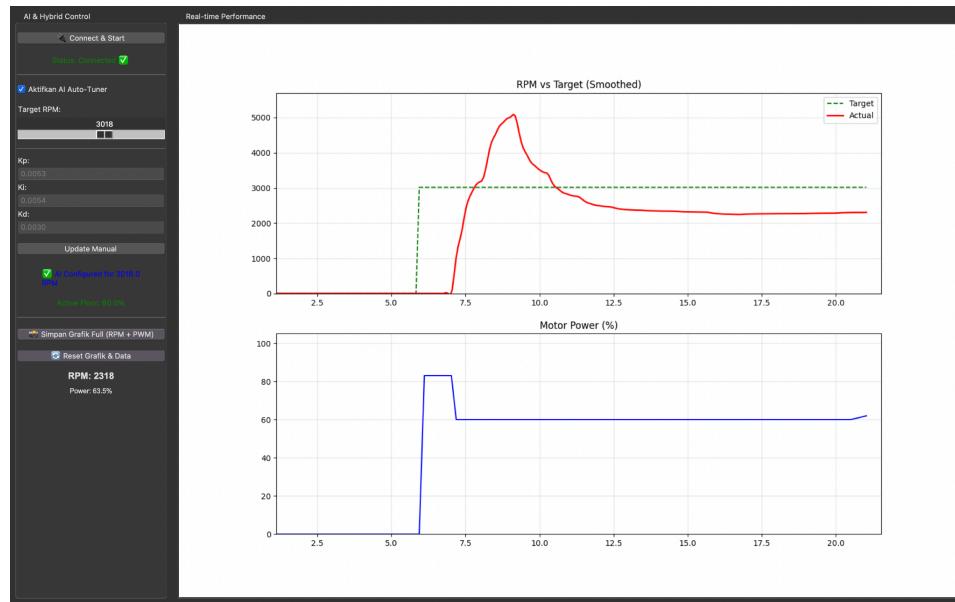
Fokus Utama: Optimasi Parameter Otomatis berbasis Machine Learning.

Analisis Cara Kerja: Ini adalah inti kecerdasan sistem. Menggantikan metode trial-and-error manusia dengan algoritma XGBoost.

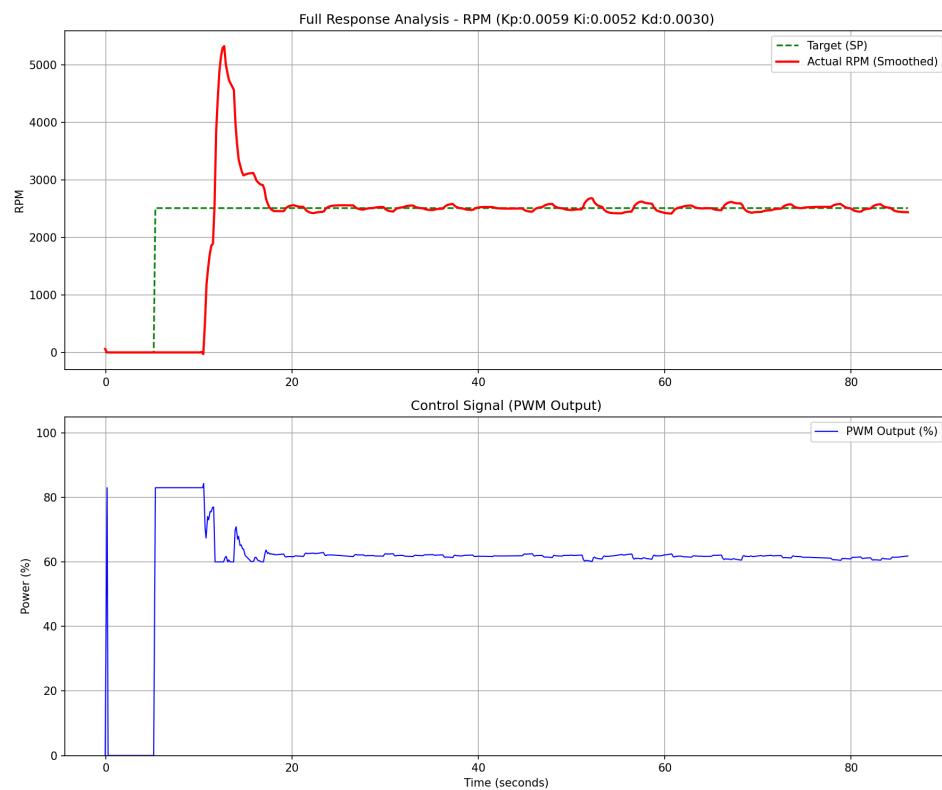
- **Mekanisme Cerdas:** Berdasarkan data eksperimen sebelumnya (Tiered Data Collection), model AI memprediksi kombinasi Kp, Ki, Kd terbaik untuk setiap target RPM.
- **Hasil Pengujian:**
 - **2500 RPM:** AI memberikan parameter yang lembut agar motor tidak berosilasi di kecepatan rendah.
 - **3000 - 4000 RPM:** AI memberikan parameter yang lebih agresif untuk respon cepat. Grafik menunjukkan motor mencapai target stabil kurang lebih 10 detik tanpa bantuan tangan manusia.

Gambar 5. Proses Pencarian Data

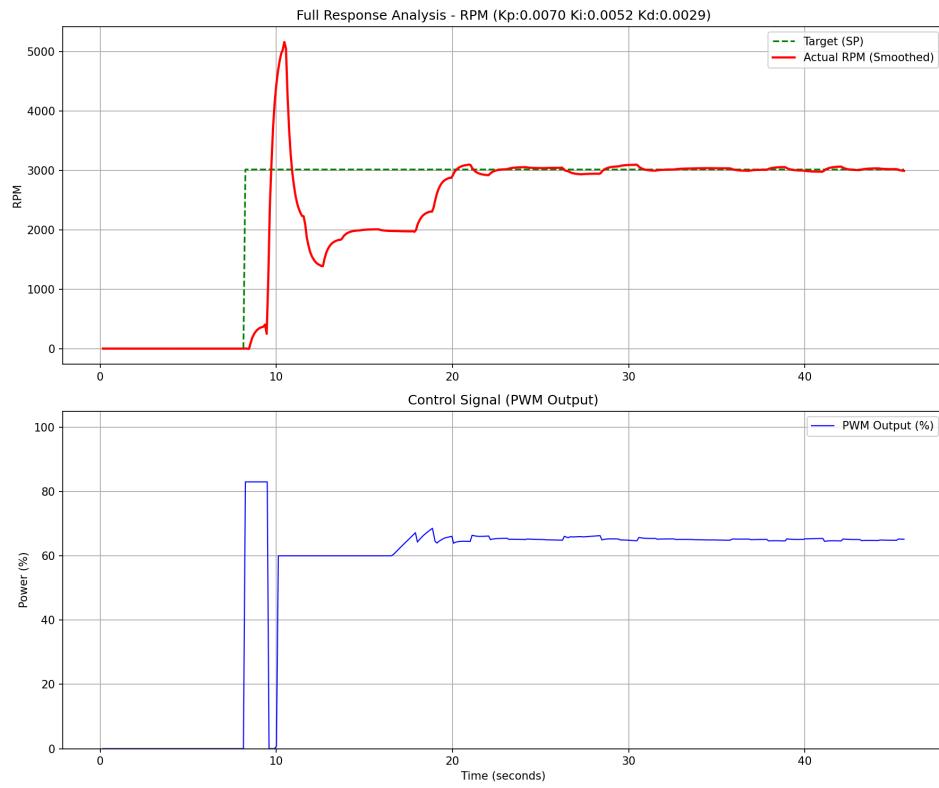
Gambar 6. Hasil Training Data



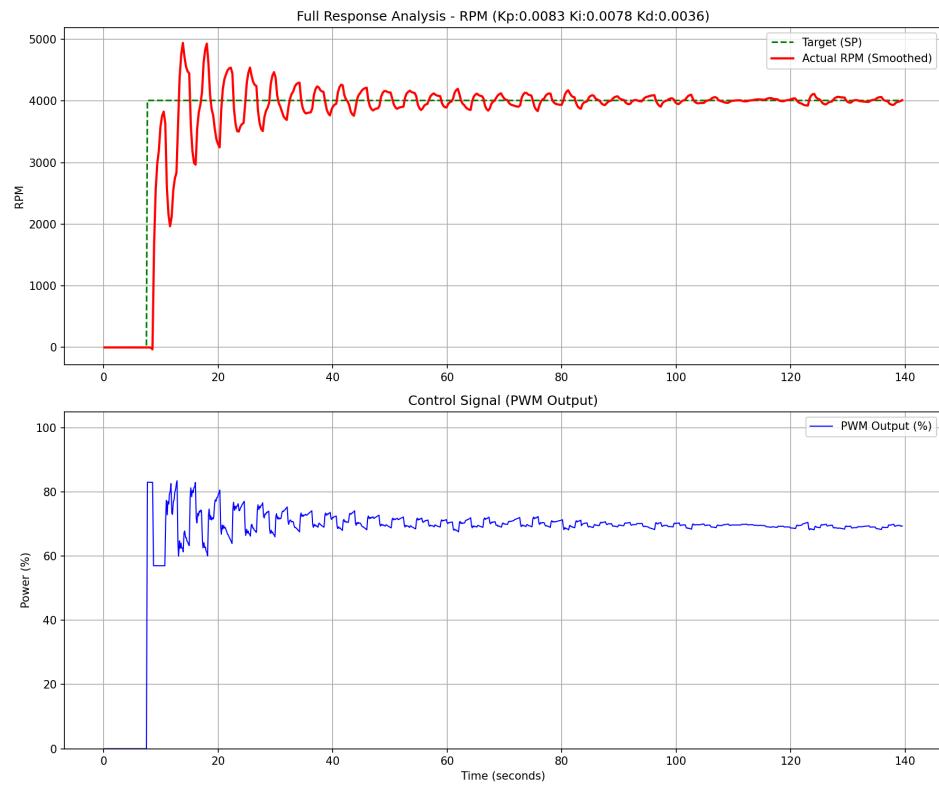
Gambar 7. Proses Pencarian PID Optimal dengan AI



Gambar 8. Hasil AI pada Setpoint 2500 RPM



Gambar 9. Hasil AI pada Setpoint 3000 RPM



Gambar 10. Hasil AI pada Setpoint 4000 RPM

6. IoT PID Python (Manual)

Fokus Utama: Integrasi Sistem Kendali Lokal dengan Antarmuka Cloud.

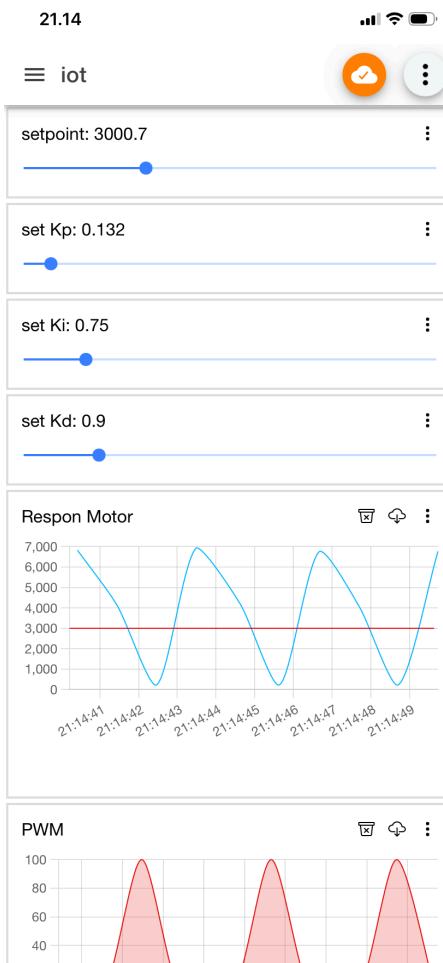
Analisis Cara Kerja: Menggabungkan keandalan PID Python dengan fleksibilitas IoT.

- **Sinkronisasi:** Saat slider di dashboard HP digeser, Python menerima target baru dan PID lokal langsung bekerja mengejar target tersebut.
- **Monitoring:** Nilai RPM aktual dikirim balik dari Python ke HP, memungkinkan pemantauan dua arah. Pengguna bisa melihat performa motor secara *live* dari layar HP.

```
IoT-PID-Python.ipynb
iot-pid-python.ipynb > C:\ --- MAIN PID LOOP (FINAL FIX: Setpoint Line) -->
Generate + Code + Markdown | ⚡ Interrupt ⚡ Restart ⚡ Clear All Outputs ⚡ Go To | Jupyter Variables | Outline ... | Python 3.13.2
finally:
    lab.stop()
    print("Motor dimatikan.")

31.8s
PID Loop Dimulai (Update IP setup ls)
T: 2.3 | SP: 1 | RPM: 1073 | Out: 0.0%
T: 2.3 | SP: 1 | RPM: 231 | Out: 0.0%
T: 3.1 | SP: 1 | RPM: 261 | Out: 0.0%
T: 4.2 | SP: 1 | RPM: 0 | Out: 0.0%
T: 5.2 | SP: 1 | RPM: 288 | Out: 0.0%
T: 6.2 | SP: 1 | RPM: 0 | Out: 0.0%
Setpoint Diperbarui: 3000.7 RPM
Setpoint Diperbarui: 1.0 RPM
T: 8.3 | SP: 1 | RPM: 0 | Out: 0.0%
T: 8.3 | SP: 1 | RPM: 0 | Out: 0.0%
T: 9.4 | SP: 1 | RPM: 299 | Out: 0.0%
T: 10.4 | SP: 1 | RPM: 0 | Out: 0.0%
T: 11.4 | SP: 1 | RPM: 233 | Out: 0.0%
T: 12.5 | SP: 1 | RPM: 0 | Out: 0.0%
Setpoint Diperbarui: 4300.57 RPM
T: 13.5 | SP: 4300 | RPM: 0 | Out: 100.0%
Setpoint Diperbarui: 4300.7 RPM
T: 14.6 | SP: 3001 | RPM: 4753 | Out: 0.0%
T: 15.6 | SP: 3001 | RPM: 3545 | Out: 0.0%
T: 16.6 | SP: 3001 | RPM: 232 | Out: 100.0%
T: 17.7 | SP: 3001 | RPM: 6792 | Out: 0.0%
T: 18.7 | SP: 3001 | RPM: 4189 | Out: 0.0%
T: 19.8 | SP: 3001 | RPM: 261 | Out: 100.0%
T: 20.8 | SP: 3001 | RPM: 6819 | Out: 0.0%
...
T: 28.1 | SP: 3001 | RPM: 4215 | Out: 0.0%
T: 29.1 | SP: 3001 | RPM: 233 | Out: 100.0%
T: 30.2 | SP: 3001 | RPM: 6750 | Out: 0.0%
T: 31.2 | SP: 3001 | RPM: 4185 | Out: 0.0%
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

Gambar 11. Proses Penerimaan Pesan dari HP



Gambar 12. Proses Setting PID dari HP

7. IoT PID Menggunakan AI

Fokus Utama: Sistem Cerdas Terintegrasi Penuh

Analisis Cara Kerja: Menggabungkan seluruh komponen: Kendali Motor, Algoritma AI, dan Jaringan IoT.

- **Fitur Unggulan:**

1. **Auto-Tuning Real-time:** Saat target diubah lewat HP, AI di laptop langsung menghitung ulang parameter PID dalam hitungan milidetik.
2. **Stabilisasi Data:** Diterapkan algoritma *Smoothing* (pembulatan data) dan *Rate Limiting* pada pengiriman data ke HP. Hasilnya, grafik di HP terlihat stabil dan rapi (tidak bergerigi/jitter) meskipun kondisi jaringan internet naik-turun.

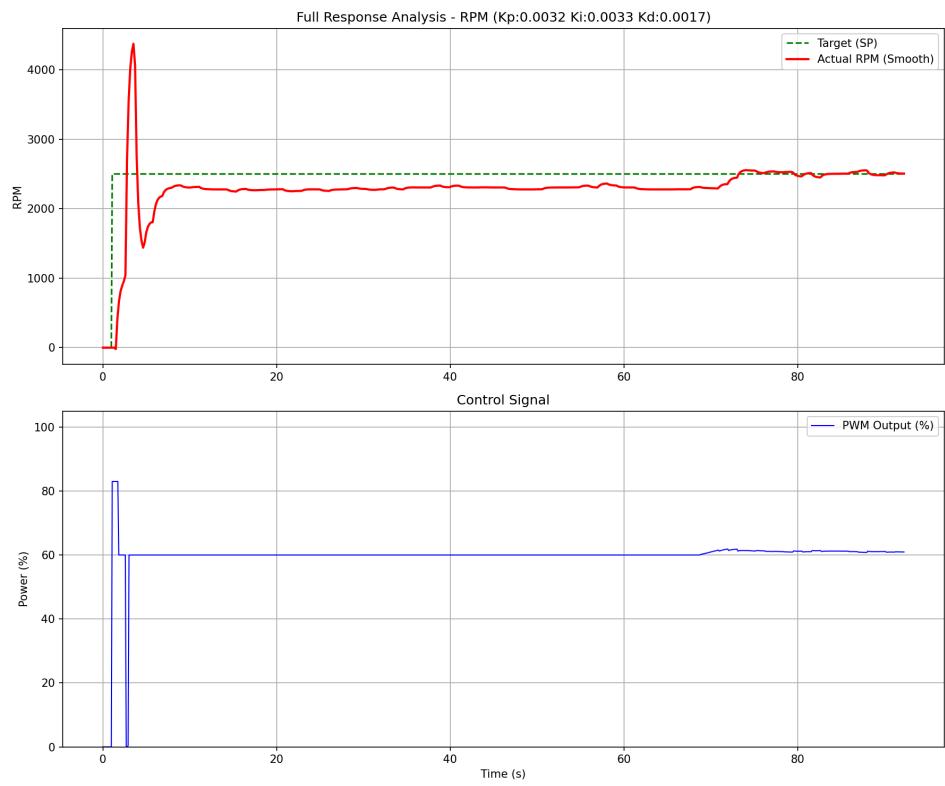
- **Kesimpulan:** Sistem mampu beradaptasi secara cerdas terhadap perubahan target kecepatan dari jarak jauh dengan kestabilan tinggi.



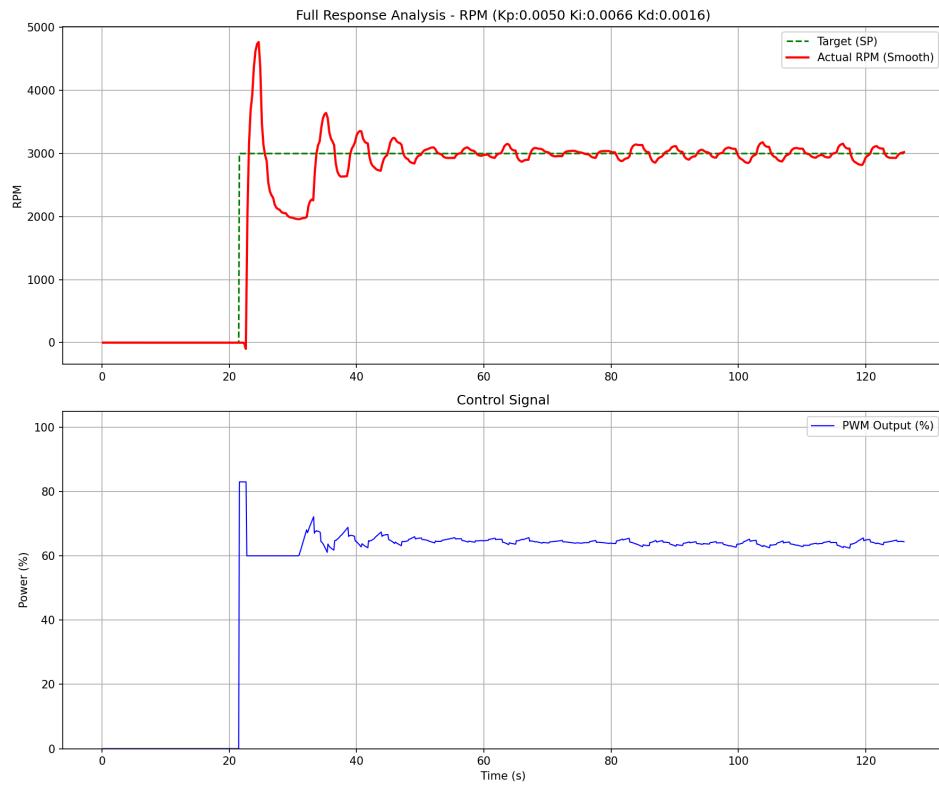
Gambar 13. Proses Monitoring PID Melalui Python



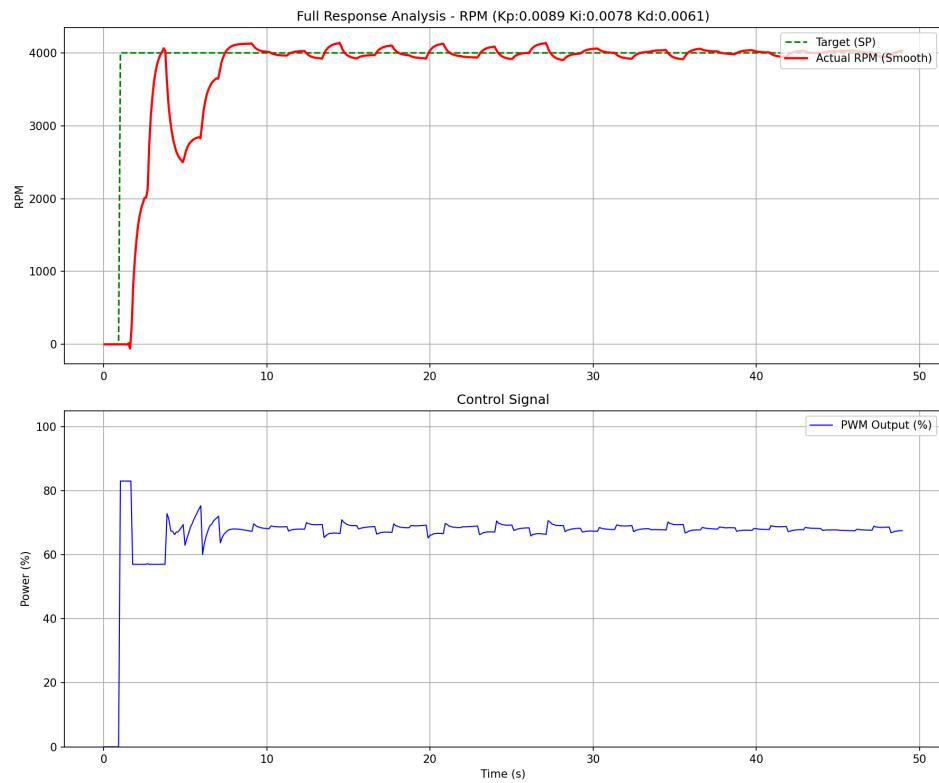
Gambar 14. Proses Setting Setpoint dari HP



Gambar 15. Hasil AI pada Setpoint 2500 RPM



Gambar 16. Hasil AI pada Setpoint 3000 RPM



Gambar 17. Hasil AI pada Setpoint 4000 RPM