



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



TRAINER KIT KENDALI MOTOR DC DUAL-CONTROL PID- FUZZY BERBASIS ESP32 DAN VISUALISASI DIGITAL TWIN

3D

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Diploma Tiga**

**POLITEKNIK
NEGERI
JAKARTA**

Ghani Putra Nuari

2303321033

**PROGRAM STUDI ELEKTRONIKA INDUSTRI
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI JAKARTA**

2026



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**IMPLEMENTASI ALGORITMA DUAL-CONTROL PID-
FUZZY PADA *TRAINER KIT* KENDALI MOTOR DC
BERBASIS ESP32**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar
Diploma Tiga**

**POLITEKNIK
NEGERI
JAKARTA**

Ghani Putra Nuari

2303321033

**PROGRAM STUDI ELEKTRONIKA INDUSTRI
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI JAKARTA**

2026



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

HALAMAN PERNYATAAN ORISINALITAS

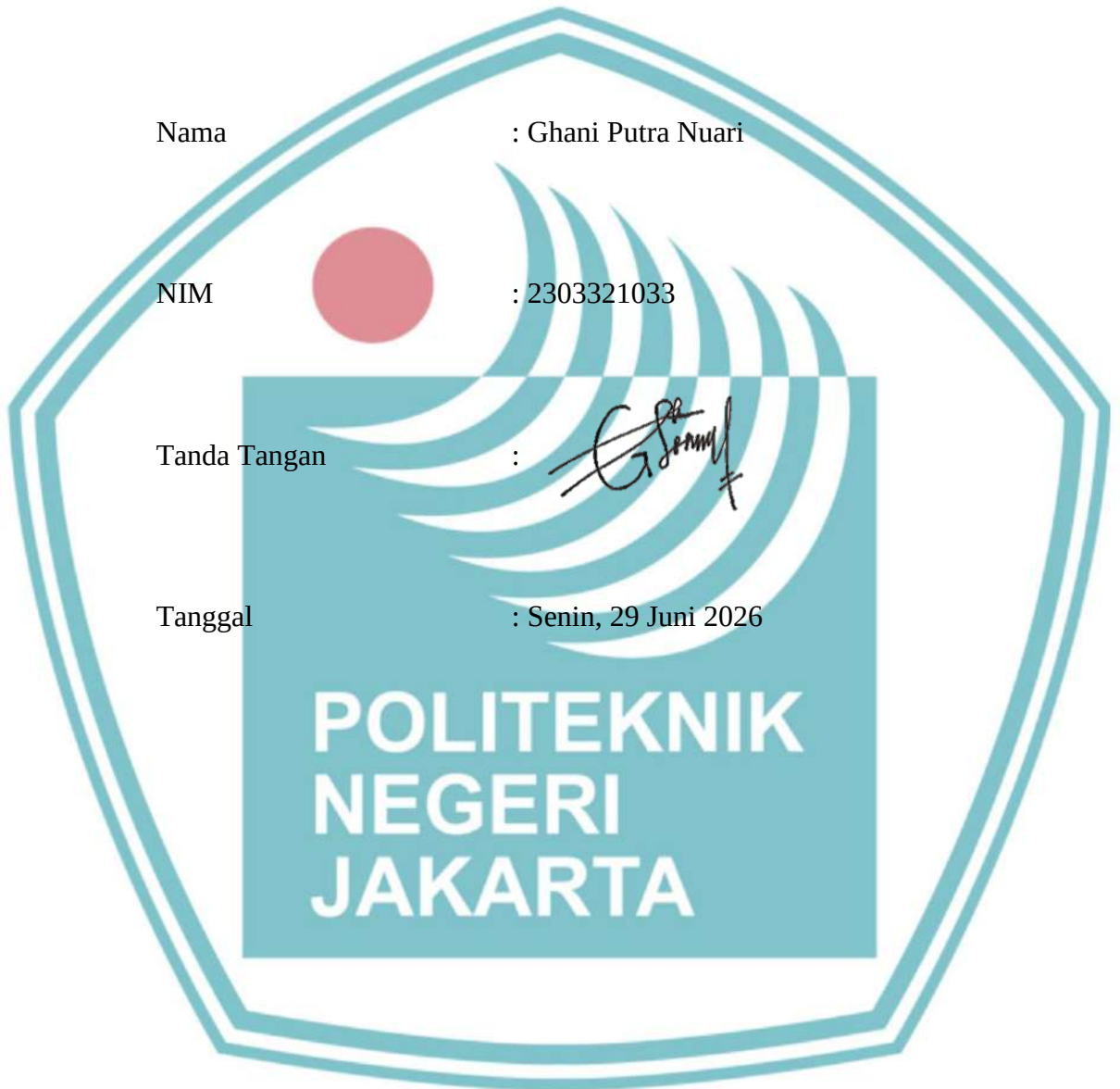
Tugas Akhir ini adalah hasil karya saya sendiri dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Ghani Putra Nuari

NIM : 2303321033

Tanda Tangan : 

Tanggal : Senin, 29 Juni 2026





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**LEMBAR PENGESAHAN
TUGAS AKHIR**

Tugas Akhir diajukan oleh

Nama : Ghani Putra Nuari
NIM : 2303321033
Program Studi : Elektronika Industri
Judul Tugas Akhir : *Trainer Kit Kendali Motor DC Dual – Control PID–Fuzzy Berbasis ESP32 dengan Visualisasi Digital Twin 3D*
Sub Judul Tugas Akhir : Implementasi Algoritma *Dual–Control* PID–Fuzzy pada *Trainer Kit* Kendali Motor DC Berbasis ESP32

Telah diuji oleh tim penguji dalam Sidang Tugas Akhir pada Senin, 29 Juni 2026 dan dinyatakan **LULUS**.

Pembimbing I: Nana Sutarna, S.T., M.T., Ph.D. ()

NIP. 197007122001121001

Depok, 29 Juni 2026

Disahkan Oleh

Ketua Jurusan Teknik Elektro



Dr. Murie Dwiyanti, S.T., M.T.

NIP. 197803312003122002



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat – Nya, saya dapat menyelesaikan Tugas Akhir ini. Penulisan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Diploma Tiga. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan Tugas Akhir ini. Oleh karena itu, penulis mengucapkan terimakasih kepada :

1. Ibu Dr. Murie Dwiyanti, S.T., M.T. Selaku Ketua Jurusan Teknik Elektro Politeknik Negeri Jakarta.
2. Bapak Ihsan Auditia Akhinov, S.T., M.T. Selaku Kepala Program Studi Elektronika Industri Politeknik Negeri Jakarta.
3. Bapak Nana Sutarna, S.T., M.T., Ph.D. Selaku Dosen Pembimbing yang telah menyediakan waktu dan tenaganya untuk membimbing dan mendukung saya dalam penyelesaian Tugas Akhir ini.
4. Orang Tua dan Keluarga yang telah memberikan bantuan dan dukungan semangat sehingga penulis dapat menyelesaikan laporan Tugas Akhir ini.
5. Rekan-Rekan “Padepokan Dimas Kanjeng” Tiar, Revaldi, Radhifka, Hasyim & Helmi yang telah menjadi bagian tak terpisahkan dari perjalanan kuliah penulis.
6. Zulhelmi Syahtiar serta Tiar Rizki Budiyo sebagai rekan satu tim yang telah banyak memberi bantuan dan dukungan sehingga Laporan Tugas Akhir ini dapat diselesaikan.

Akhir kata, penulis berharap kepada Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Penulis menyadari bahwa masih banyak kekurangan dalam penulisan laporan ini sehingga penulis sangat terbuka dalam kritik dan saran yang membangun. Semoga Laporan Tugas Akhir ini membawa manfaat bagi pengembang ilmu baik penulis maupun pembaca.

Depok, 29 Juni 2026

Penulis



Implementasikan Algoritma Dual-Control PID-Fuzzy pada Trainer Kit Kendali Motor DC Berbasis ESP32

ABSTRAK

Penelitian ini merancang dan mengimplementasikan Trainer Kit Kendali Motor DC dengan algoritma Dual-Control PID-Fuzzy berbasis ESP32 sebagai media pembelajaran. Sub-sistem yang dikembangkan, yaitu “Implementasi Algoritma Dual-Control PID-Fuzzy pada Trainer Kit Kendali Motor DC Berbasis ESP32”, menempatkan dua motor DC dalam konfigurasi paralel dengan satu setpoint tunggal, di mana motor PID memungkinkan tuning gain K_p , K_i , dan K_d secara manual melalui rotary encoder sementara motor Fuzzy beroperasi dengan parameter pre-tuned sebagai benchmark kinerja ideal. Hasil pengujian pada rentang setpoint 25-250 RPM menunjukkan bahwa kontrol Fuzzy secara performa unggul dengan rata-rata Overshoot 1,74% dibandingkan PID 4,47%, rata-rata Settling Time 4,20 detik dibandingkan PID 5,52 detik, serta rata-rata Rise Time 3,28 detik dibandingkan PID 2,96%, sementara kedua metode sama-sama mampu mencapai Steady-State Error 0 RPM pada kondisi optimal. Validasi akurasi sensor menggunakan tachometer menghasilkan akurasi rata-rata 98,99% untuk PID dan 99,07 untuk Fuzzy, yang memastikan keandalan data feedback. Sistem komunikasi data real-time ke platform web melalui Firebase Realtime Database teruji dengan latensi rata-rata 41,3 ms dan sinkronisasi data hardware – software tanpa selisih. Penelitian ini membuktikan bahwa konfigurasi dual-control efektif sebagai media pembelajaran interaktif yang memungkinkan mahasiswa memahami secara visual dan kuantitatif perbedaan karakteristik respons antara kontrol konvensional PID dan kontrol cerdas Fuzzy.

Kata Kunci: Fuzzy Logic Controller, PID, Motor DC, Trainer Kit, ESP32.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Implementation of Dual-Control PID-Fuzzy Algorithm on ESP32-Based DC Motor Control Trainer Kit

ABSTRACT

This research designs and implements a DC Motor Control Trainer Kit with a Dual-Control PID-Fuzzy algorithm based on ESP32 as a learning media. The developed sub-system, entitled “Implementation of Dual-Control PID-Fuzzy Algorithm on ESP32-Based DC Motor Control Trainer Kit”, places two DC motor in a parallel configuration with a single setpoint, where the PID motor alws manual tuning of Kp, Ki, and Kd gains through a rotary encoder while the Fuzzy motor operates with pre-tuned parameters as an ideal performance benchmark. Testing results in the setpoint range of 25-250 RPM show that Fuzzy control performs better with an average Overshoot of 1.74% compared to PID at 4.47%, an average Settling Time of 4.20 seconds compared to PID at 5.52 seconds, and an average Rise Time of 3.28 seconds compared to PID at 2.96 seconds, while both methods are equally able to achieve Steady-State Error of 0 RPM under optimal conditions. Sensor accuracy validation using a tachometer yields an average accuracy of 98.99% for PID and 99.07% for Fuzzy, ensuring reliable feedback data. The real-time data communication system to the web platform via Firebase Realtime Database is tested with an average latency of 41.3 ms and hardware-software data synchronization without any difference. This research proves that the dual-control configuration is effective as an interactive learning media that allows students to understand visually and quantitatively the difference in response characteristics between conventional PID control and intelligent Fuzzy control.

Keywords: Fuzzy Logic Controller, PID, Motor DC, Trainer Kit, ESP32.

POLITEKNIK
NEGERI
JAKARTA

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN JUDUL.....	ii
HALAMAN PERNYATAAN ORISINALITAS.....	iii
LEMBAR PENGESAHAN TUGAS AKHIR	iv
KATA PENGANTAR	v
ABSTRAK	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
DAFTAR RUMUS	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan.....	3
1.5 Luaran.....	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 State of The Art	4
2.2 Trainer Kit	5
2.3 PID.....	5
2.4 Fuzzy	6
2.5 IoT & Website	7
2.6 Mikrokontroler ESP32.....	7
2.7 Motor DC & Rotary Encoder	7
2.8 Arduino IDE	8
2.9 MATLAB & Simulink.....	9
BAB III PERENCANAAN DAN REALISASI.....	10
3.1 Rancangan Alat.....	10

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3.1.1 Deskripsi Alat	10
3.1.2 Deskripsi Umum Alat	10
3.1.3 Spesifikasi Alat	11
3.1.4 Diagram Blok.....	12
3.1.5 Flowchart Cara Kerja Alat	13
3.1.6 Flowchart Kode Program	14
3.2 Realisasi Alat	15
3.2.1 Nilai Transfer Function Motor DC	15
3.2.2 Pengujian PID pada Permodelan Simulink.....	16
3.2.3 Realisasi Program Kontrol PID pada Arduino IDE	16
3.2.4 Pembuatan <i>Membership Function Input dan Output Fuzzy</i>	17
3.2.5 Pembuatan <i>Rule Base Fuzzy</i>	19
3.2.6 Pengujian Simulasi <i>Viewer Fuzzy</i>	19
3.2.7 Realisasi Program Kontrol Fuzzy pada Arduino IDE.....	21
3.2.8 Metode Tuning PID	23
3.2.9 Sistem Komunikasi Data.....	23
BAB IV PEMBAHASAN.....	26
4.1 Pengujian Komunikasi Data	26
4.1.1 Deskripsi Pengujian	26
4.1.2 Prosedur Pengujian	26
4.1.3 Data Hasil Pengujian.....	27
4.1.4 Analisis Data / Evaluasi	28
4.2 Pengujian Algoritma Kontrol PID.....	29
4.2.1 Deskripsi Pengujian	29
4.2.2 Prosedur Pengujian	29
4.2.3 Data Hasil Pengujian.....	30
4.2.4 Analisis Data / Evaluasi	34
4.3 Pengujian Algoritma Kontrol Fuzzy.....	35



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4.3.1 Deskripsi Pengujian	35
4.3.2 Prosedur Pengujian	35
4.3.3 Data Hasil Pengujian.....	36
4.3.4 Analisis Data / Evaluasi	36
4.4 Pengujian <i>Dual-control</i> PID-Fuzzy dengan Perubahan Setpoint	37
4.4.1 Deskripsi Pengujian	37
4.4.2 Prosedur Pengujian	38
4.4.3 Data Hasil Pengujian.....	38
4.4.4 Analisis Data / Evaluasi	39
4.5 Pengujian Akurasi Sensor Encoder Setiap Motor DC dengan Tachometer.....	40
4.5.1 Deskripsi Pengujian	40
4.5.2 Prosedur Pengujian	40
4.5.3 Data Hasil Pengujian.....	40
4.5.4 Analisis Data / Evaluasi	41
BAB V PENUTUP.....	42
5.1 Kesimpulan.....	42
5.2 Saran	42
DAFTAR PUSTAKA	43
LAMPIRAN.....	45
L-1. Daftar Riwayat Hidup	45
L-2. Poster Alat.....	46
L-3. SOP Alat.....	47
L-3. Tampilan Alat.....	48
L-4. Tampilan Website.....	49
L-5. Program Algoritma Sistem <i>Trainer Kit</i>	50
L-6. Datasheet Motor JGA250-370	72
L-7. Datasheet Driver L298N	73
L-8. Datasheet ESP32	75



DAFTAR TABEL

Tabel 2. 1 State Of The Art.....	4
Tabel 3. 1 Spesifikasi Software.....	11
Tabel 3. 2 Parameter Input Error.....	17
Tabel 3. 3 Parameter Input Delta Error.....	17
Tabel 3. 4 Parameter Output PWM.....	18
Tabel 3. 5 Rule Base	19
Tabel 4. 1 Alat dan Bahan Pengujian Komunikasi Data.....	26
Tabel 4. 2 Uji Latensi Komunikasi Data.....	27
Tabel 4. 3 Uji Sinkronisasi Data Hardware - Software.....	28
Tabel 4. 4 Alat dan Bahan Pengujian Algoritma Kontrol PID	29
Tabel 4. 5 Pengujian Respon Kontrol Proportional	30
Tabel 4. 6 Pengujian Respon Kontrol Integral	31
Tabel 4. 7 Pengujian Respon Kontrol Derivative	31
Tabel 4. 8 Pengujian Respon Kontrol PI.....	32
Tabel 4. 9 Pengujian Respon Kontrol PD	33
Tabel 4. 10 Pengujian Respon Kontrol PID.....	33
Tabel 4. 11 Alat dan Bahan Pengujian Algoritma Kontrol Fuzzy	35
Tabel 4. 12 Pengujian Algoritma Fuzzy	36
Tabel 4. 13 Alat dan Bahan Pengujian Dual-Control Beragam Setpoint.....	37
Tabel 4. 14 Data Pengujian Perubahan Setpoint.....	38
Tabel 4. 15 Alat dan Bahan Pengujian Akurasi Sensor Encoder	40
Tabel 4. 16 Data Akurasi RPM Motor	41

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



DAFTAR GAMBAR

Gambar 2. 1 Diagram Blok Kontrol PID	6
Gambar 2. 2 Diagram Blok Kontrol Fuzzy	6
Gambar 2. 3 Mikrokontroler ESP32	7
Gambar 2. 4 Motor DC Encoder JGA25-370	8
Gambar 2. 5 Rotary Encoder KY-040	8
Gambar 2. 6 Software Arduino IDE	9
Gambar 2. 7 MATLAB & Simulink	9
Gambar 3. 1 Diagram Blok Sistem Trainer Kit	12
Gambar 3. 2 Flowchart Cara Kerja Alat	13
Gambar 3. 3 Flowchart Kode Program	14
Gambar 3. 4 Grafik Transfer Function Ciancone	15
Gambar 3. 5 Konfigurasi Blok Permodelan PID	16
Gambar 3. 6 Grafik Permodelan PID	16
Gambar 3. 7 Membership Function Input Error	17
Gambar 3. 8 Membership Function Input Delta Error	18
Gambar 3. 9 Membership Function Output PWM	18
Gambar 3. 10 Rule Editor	19
Gambar 3. 11 Rule Viewer	19
Gambar 3. 12 Surface Viewer	20
Gambar 4. 1 Uji Latensi Pengiriman Data	27
Gambar 4. 2 Uji Sinkronisasi Data Hardware – Software	28
Gambar 4. 3 Grafik Pengujian Kontrol Proportional	30
Gambar 4. 4 Grafik Pengujian Kontrol Integral	31
Gambar 4. 5 Grafik Pengujian Kontrol Derivative	31
Gambar 4. 6 Grafik Pengujian Kontrol PI	32
Gambar 4. 7 Grafik Pengujian Kontrol PD	33
Gambar 4. 8 Grafik Pengujian Kontrol PID	33
Gambar 4. 9 Pengujian Algoritma Fuzzy	36
Gambar 4. 10 Hasil Pengujian Perubahan Setpoint	38

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR RUMUS

(3. 1) Transfer Function Ciancone	16
(3. 2) Rumus PID	16





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR LAMPIRAN

L-1. Daftar Riwayat Hidup	45
L-2. Poster Alat	46
L-3. SOP Alat.....	47
L-3. Tampilan Alat.....	48
L-4. Tampilan Website.....	49
L-5. Program Algoritma Sistem <i>Trainer Kit</i>	50
L-6. Datasheet Motor JGA250-370	72
L-7. Datasheet Driver L298N	73
L-8. Datasheet ESP32	75





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB I PENDAHULUAN

1.1 Latar Belakang

Motor DC merupakan salah satu aktuator elektromekanis yang sering dijumpai serta masih digunakan dalam dunia Manufaktur dalam berbagai aplikasi otomasi seperti konveyor, mesin CNC (*Computer Numerical Control*), hingga Robotik (Al Fikram & Anggara, 2024). Motor DC bekerja dengan cara mengonversi energi listrik menjadi energi gerak rotasi. Motor DC memiliki karakteristik yang responsif terhadap perubahan sinyal kontrol namun dapat bersifat non-linear akibat gesekan mekanis, variasi beban, dan perubahan kondisi operasi. Tantangan inilah yang kemudian menjadi titik masuk bagi penerapan metode kontrol modern, terlebih sejak Revolusi Industri 4.0 yang mendorong tuntutan presisi dan kecerdasan sistem kendali ke tingkat yang jauh lebih tinggi dari sebelumnya.

Dalam konteks penerapan metode kontrol yang umum digunakan dalam dunia industri dan manufaktur, lebih dari 90% aplikasi kendali industri secara global masih mengandalkan algoritma kontrol *Proportional – Integral – Derivative* (PID) sebagai metode kontrol utama yang efektif dalam menjaga kestabilan pada kondisi *Steady-State*, serta mudah dalam implementasi pada sistem linear (Supriyono dkk., 2024). Namun, ketika sistem menghadapi non-linieritas, perubahan beban mendadak, atau gangguan transien yang ekstrem, PID yang bergantung pada model matematis yang presisi cenderung menghasilkan nilai *Overshoot* yang tinggi dan *Settling Time* yang panjang. Kondisi ini membuka ruang bagi *Fuzzy Logic Controller* (FLC) sebagai metode komplementer yang beroperasi berdasarkan logika linguistik dan aturan *IF-THEN* sehingga lebih adaptif terhadap ketidakpastian sistem.

Keandalan kedua metode kontrol tersebut bergantung pada presisi data *feedback* yang diperoleh dari sensor. Internal sensor encoder pada Motor DC, digunakan sebagai sensor elektromekanis yang mengonversi gerakan rotasi poros motor menjadi sinyal pulsa digital berupa nilai *Revolution Per Minutes* (RPM), yang digunakan sebagai nilai *Process Variable* (PV). Perkembangan Industri 4.0 kemudian memperluas metode pemantauan dengan menciptakan teknologi Digital Twin 3D sebagai visualisasi atau replika virtual suatu sistem fisik yang simultan dan *real-time* dengan kondisi aktual motor (Santos dkk., 2023). Integrasi ini memungkinkan pemantauan, analisis, dan visualisasi kinerja sistem kendali secara komprehensif dalam satu platform yang terhubung secara utuh.



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Di tengah transformasi digital yang pesat tersebut, Indonesia justru menghadapi kesenjangan kapasitas SDM yang kritis dengan 90,45% tenaga kerja sektor manufaktur memiliki tingkat keterampilan rendah, sementara hanya 6.52% yang memiliki keterampilan menengah dan 3.03% yang memiliki keterampilan tinggi. Lebih dari 63% pekerja di sektor elektronik Indonesia berisiko tinggi terkena otomasi akibat kurangnya keterampilan teknik yang memadai (International Labour Organization (ILO), 2024). Kondisi ini menempatkan Politeknik Negeri Jakarta (PNJ) sebagai institusi vokasi pada posisi strategis dalam mencetak lulusan yang kompeten. Namun, fasilitas praktikum yang ada saat ini belum mampu menjawab tantangan tersebut. Ketiadaan alat peraga fisik yang mumpuni untuk sistem kendali PID dan Fuzzy di bengkel Elektronika Industri PNJ memaksa mahasiswa hanya mengandalkan simulasi perangkat lunak MATLAB/Simulink sebagai satu-satunya media belajar. Akibatnya, mahasiswa kehilangan pengalaman secara langsung.

Berdasarkan seluruh uraian di atas, penelitian ini merancang dan merealisasikan *Trainer Kit* Kendali Motor DC *Dual-control* PID-Fuzzy Berbasis ESP32 dengan Visualisasi Digital Twin 3D sebagai solusi integratif. Sistem ini menempatkan dua motor dalam konfigurasi paralel dengan satu Setpoint tunggal, di mana Motor PID dapat di-tuning secara manual melalui dua Rotary encoder dengan Rotary encoder pertama untuk mengatur nilai Setpoint (RPM motor) dan Rotary encoder kedua mengatur nilai parameter Kp, Ki, dan Kd secara bergantian melalui pemilihan switching. Selain itu, terdapat Motor Fuzzy yang beroperasi dengan parameter pre-tuned sebagai *benchmark* kinerja ideal bagi Motor PID. Seluruh data respons sistem dikirim secara *real-time* ke platform web yang menampilkan grafik komparatif SP dan PV kedua motor dan Visualisasi Digital Twin 3D yang berputar secara sinkron dengan motor fisik. Dengan kontribusi ini, penelitian diharapkan mampu menjadi referensi pengembangan media pembelajaran sistem kendali di institusi vokasi, sekaligus mempersiapkan lulusan yang kompetitif dalam menghadapi tuntutan transformasi digital industri.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana merancang serta mengimplementasikan algoritma *Dual-control* PID-Fuzzy berbasis ESP32 dengan input Rotary Encoder untuk mengendalikan kecepatan dua Motor DC secara paralel?
2. Bagaimana membangun sistem komunikasi data *real-time* berbasis ESP32 untuk melihat respons sistem Motor DC *Dual-control* PID-Fuzzy pada platform website?



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

3. Bagaimana hasil perbandingan pengujian kinerja setiap metode kontrol pada Motor DC dengan data *Overshoot*, *Settling Time*, *Rise Time*, dan *Steady-State Error*?

1.3 Batasan Masalah

Adapun Batasan Masalah dalam penelitian Tugas Akhir ini adalah sebagai berikut:

1. Algoritma program *Dual-control* PID-Fuzzy untuk kendali Motor DC berbasis ESP32.
2. Komunikasi data *real-time* hasil respons Motor DC pada website.
3. Uji perbandingan kinerja metode kontrol Motor DC.
4. Evaluasi performa metode kontrol dengan data *Overshoot*, *Settling Time*, *Rise Time*, dan *Steady-State Error*

1.4 Tujuan

Adapun tujuan yang ingin dicapai dalam penelitian Tugas Akhir ini adalah sebagai berikut:

1. Merancang serta merealisasikan *Trainer Kit* Motor DC dengan algoritma *Dual-control* PID-Fuzzy berbasis ESP32.
2. Membangun sistem komunikasi data *real-time* pada platform website untuk memvisualisasikan grafik respon sistem Motor DC *Dual-control* PID-Fuzzy.
3. Menguji hasil perbandingan kinerja Motor DC terhadap setiap metode kontrol dengan data *Overshoot*, *Settling Time*, *Rise Time*, dan *Steady-State Error*.

1.5 Luaran

Adapun luaran yang diharapkan dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Perangkat keras *Trainer Kit* Motor DC *Dual-Control* PID-Fuzzy yang dapat digunakan pada Lab Elektronika Industri untuk pembelajaran bagi Mahasiswa.
2. Perangkat lunak Embedded berupa program Arduino IDE pada Firmware ESP32 yang berisi algoritma *Dual-control* PID-Fuzzy untuk kontrol Motor DC.
3. Sistem Website sebagai aplikasi kontrol Motor DC yang dilengkapi dengan grafik respons sistem serta Visualisasi Digital Twin 3D untuk menampilkan model 3D Motor DC secara sinkron dengan kondisi aktual.
4. Laporan Teknis yang akan memuat metodologi perancangan, hasil pengujian, analisis kinerja setiap metode kontrol serta dokumentasi perancangan alat.
5. Modul pembelajaran yang akan memuat panduan penggunaan alat serta Jobsheet praktikum.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

BAB V

PENUTUP

5.1 Kesimpulan

1. Trainer Kit berhasil diimplementasikan dengan dua Motor DC yang dirancang secara paralel yang dikendalikan secara simultan oleh ESP32. Motor PID memungkinkan tuning *gain* Kp, Ki, Kd melalui rotary encoder dengan indikator LED, sementara Motor Fuzzy beroperasi dengan parameter *pre-tuned* sebagai benchmark. Algoritma kedua metode berjalan stabil pada *firmware* ESP32 dengan *interval sampling* 50 ms dan kalibrasi RPM otomatis.
2. Sistem komunikasi berbasis *Firebase Realtime Database* dengan protokol HTTPS REST API berhasil mengirim data *real-time* dengan latensi rata-rata 41,3 ms (dengan kategori sangat baik) dan sinkronisasi *hardware-software* tanpa selisih perbedaan data. Dua mode operasi kontrol PID-Fuzzy tersedia secara Online (Perangkat fisik terhubung dengan *Web monitoring*) dan Simulasi (Berjalan di *browser* tanpa *hardware*).
3. Perbandingan hasil pengujian Fuzzy unggul dengan rata-rata *Overshoot* 1,74% vs PID 4,47 dan *Settling Time* 4,20 detik vs PID 5,52 detik, sementara kedua metode sama-sama mencapai *Steady-State Error* 0 RPM. Akurasi sensor encoder tervalidasi 98,99% (PID) dan 99,07% (Fuzzy) via alat ukur Tachometer. Konfigurasi dual-control ini efektif sebagai media pembelajaran interaktif yang memungkinkan mahasiswa membandingkan secara visual kuantitatif karakteristik kedua metode kontrol.

5.2 Saran

Berdasarkan hasil pengujian dan keterbatasan yang teridentifikasi pada Trainer Kit ini, disarankan untuk pengembangan selanjutnya agar menambahkan fitur time/div dan volt/div pada panel grafik respons sistem di website, sehingga pengguna dapat mengukur nilai waktu dan amplitudo secara mandiri dengan menggeser cursor horizontal dan/atau vertikal pada grid grafik, mirip dengan pengoperasian osiloskop digital, yang akan memperkuat pemahaman mahasiswa terhadap konsep pengukuran *Rise Time*, *Settling Time*, dan *Overshoot* secara visual dan interaktif.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Abdel-Salam, A.-A. S. (2022). Comparison between FLC and PID controller for speed control of DC motor. *International Robotics & Automation Journal*, 8(2), 40–45. <https://doi.org/10.15406/iratj.2022.08.00242>
- Al Fikram, M., & Anggara, N. (2024). *Implementasi Rotary Encoder Pada Sistem Pengaturan Kecepatan Motor Dc Dengan Kontrol Pid Berbasis Plc Dan Hmi Siemens*. Politeknik Manufaktur Negeri Bangka Belitung.
- Almawla, A. M., Hussein, M. J., & Abdullah, A. T. (2024). A Comparative Study of DC Motor Speed Control Techniques Using Fuzzy, SMC and PID. *Journal Europeen des Systemes Automatises*, 57(2), 397–406. <https://doi.org/10.18280/jesa.570209>
- Auditiya Marizka, D., Lastiyan Ashila, A., Agustina Solihah, A., Adi Prasetyo, H., Riza, F., & Wicaksono, R. (2025). Rancang Bangun Sistem Kendali PID Motor DC Berbasis Blynk untuk meningkatkan Efisiensi dan Fleksibilitas. *Jurnal Inovasi Teknologi Terapan*, 03(2), 277.
- Diono, Rezky Pratama, Y., & Kurniawan, B. (2025). Perbandingan metode PID dan Fuzzy Logic Control dalam sinkronisasi dua motor DC. Dalam *Jurnal Integrasi* | (Vol. 17, Nomor 2).
- Dwiyanti, M., & Moro, K. N. (2013). *Tuning Parameter Pid Dengan Metode Ciancone Pada Plant Heat Exchanger*.
- Firdaus, F., Priatna, E., Hiron, N., & Busaeri, N. (2022). Prototype Sistem Kendali Kecepatan Motor Dc Dengan Proportional Integral Derivative (Pid) Controller. *Journal Of Energy And Electrical Engineering (JEEE)*, 32(1), 2022.
- International Labour Organization (ILO). (2024). *The Skills development and employment situation in Indonesia's electronics sector: Navigating technological changes and labour transitions ilo.org/indonesia Supported by From the People of Japan*. www.ilo.org/publns.
- Pratama, A. R., Ullah, A., Zarory, H., & Maria, P. S. (2024). Rancang Bangun Alat Peraga Trainer Fuzzy Logic Controller Pada Kecepatan Motor Dc Berbasis Arduino Dan Labview. *Transmisi: Jurnal Ilmiah Teknik Elektro*, 26(3), 154–164. <https://doi.org/10.14710/transmisi.26.3.154-164>



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Reza, M., Putera, A. N., Hidayat, R., & Karawang, S. (2022). Kendali Kecepatan Motor Dc Menggunakan Pengendali Pid Dengan Encoder Sebagai Feedback. *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*.

Santos, J. F. D., Tshoombe, B. K., Santos, L. H. B., Araujo, R. C. F., Manito, A. R. A., Fonseca, W. S., & Silva, M. O. (2023). Digital Twin-Based Monitoring System of Induction Motors Using IoT Sensors and Thermo-Magnetic Finite Element Analysis. *IEEE Access*, 11, 1682–1693. <https://doi.org/10.1109/ACCESS.2022.3232063>

Shuraiji, A. L., & Shneen, S. W. (2022). Fuzzy Logic Control and PID Controller for Brushless Permanent Magnetic Direct Current Motor: A Comparative Study. *Journal of Robotics and Control (JRC)*, 3(6), 762–768. <https://doi.org/10.18196/jrc.v3i6.15974>

Soedjarwanto, N., Arinto Setyawan, F., Harahap, C. R., Adjie Riantama, N., Teknik Elektro, J., Lampung, U., & Sumantri Brojonegoro No, J. (2023). Pengendalian Kecepatan Motor Dc Menggunakan Buck-Boost Converter Berbasis IoT. *Jurnal Informatika dan Teknik Elektro Terapan*, 11(3), 2830–7062. <https://doi.org/10.23960/jitet.v11i3%20s1.3399>

Sulistiyo, W., Rizal, A., Valina, S. N., & Yesitasari, D. (2025). Pengembangan Trainer Pembelajaran Kendali Posisi Motor Dc Dengan Algoritma Pid. Dalam *ORBITH* (Vol. 21, Nomor 3).

Supriyono, H., Alanro, F. F., & Supardi, A. (2024). Development of DC Motor Speed Control Using PID Based on Arduino and Matlab For Laboratory Trainer. *Jurnal Nasional Teknik Elektro*, 36–41. <https://doi.org/10.25077/jnte.v13n1.1155.2024>

Zheng, L., & Chen, M. (2025). The Design and Implementation of a Robotic Arm Digital Twin System Based on ESP32. *International Journal of Advanced AI Applications*, 1(6).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LAMPIRAN

L-1. Daftar Riwayat Hidup



Ghani Putra Nuari

Anak kedua dari tiga bersaudara, lahir di Bekasi 29 Agustus 2005. Lulus dari SD IT Ummy Abiyyi Tahun 2017, SMPN 5 Cikarang Selatan tahun 2020, dan SMK Mitra Industri MM2100 tahun 2023. Gelar Diploma Tiga (D3) diperoleh pada tahun 2026 dari Jurusan Teknik Elektro, Program Studi Elektronika Industri, Politeknik Negeri Jakarta

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta milik Politeknik Negeri Jakarta

L-2. Poster Alat

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



TRAINER KIT KENDALI MOTOR DC DUAL – CONTROL PID – FUZZY BERBASIS ESP32 DAN VISUALISASI DIGITAL TWIN 3D



Tiar Risky Budiyo
(2303321038)



Ghani Putra Nuari
(2303321033)



Zuiheimi Syahtiar
(2303321047)

Dosen Pembimbing:
Nana Sutarna, S.T., M.T., Ph.D.
(NIP.197007122001121001)



1 | LATAR BELAKANG

Perkembangan Industri 4.0 mendorong penerapan teknologi IoT, website monitoring, dan Digital Twin dalam pendidikan vokasi. Namun, pembelajaran sistem kendali di laboratorium masih didominasi simulasi perangkat lunak sehingga pengalaman praktik secara langsung masih terbatas. Oleh karena itu, penelitian ini mengembangkan Trainer Kit Kendali Motor DC Dual-Control PID-Fuzzy berbasis ESP32 yang terintegrasi dengan website monitoring dan visualisasi Digital Twin 3D untuk menampilkan data dan performa sistem secara real-time. Sistem ini diharapkan dapat menjadi media pembelajaran yang lebih interaktif, adaptif, dan sesuai dengan kebutuhan industri modern.

2 | CARA KERJA

1. **Inisialisasi Sistem:** ESP32 menginisialisasi seluruh komponen, seperti LCD 20x4, encoder motor, driver motor, dan koneksi Wi-Fi.
2. **Pengaturan Setpoint:** Pengguna mengatur target kecepatan (RPM) kedua motor menggunakan rotary encoder, sedangkan tombol encoder digunakan untuk mereset sistem.
3. **Penyetelan Parameter PID:** Pengguna mengatur nilai Kp, Ki, dan Kd pada Motor PID menggunakan rotary encoder dengan indikator LED (merah: Kp, kuning: Ki, hijau: Kd).
4. **Proses Kendali:**
 - **Motor 1 (Fuzzy):** Mengolah nilai error dan delta error menggunakan rule base Fuzzy untuk menghasilkan PWM.
 - **Motor 2 (PID):** Mengolah nilai error menggunakan algoritma PID untuk menghasilkan PWM.
5. **Pemantauan Sistem:** LCD 20x4 menampilkan nilai setpoint, PV, dan parameter PID secara real-time.
6. **Monitoring Website dan Digital Twin 3D:** Data sistem dikirim ke Firebase dan ditampilkan pada website berupa grafik respons sistem serta visualisasi Digital Twin 3D secara real-time.

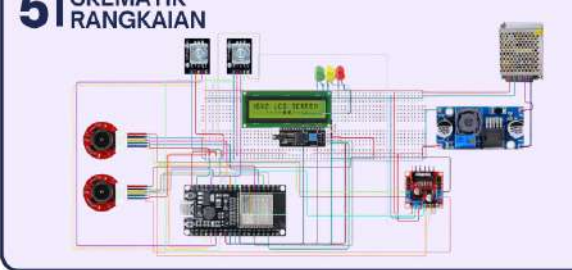
3 | TUJUAN

1. Merealisasikan trainer kit kendali motor DC dual-control PID-Fuzzy berbasis ESP32 yang portable dan ergonomis.
2. Mengimplementasikan kontroler PID dan Fuzzy Mamdani pada sistem kendali motor DC.
3. Membangun website monitoring real-time untuk menampilkan grafik perbandingan respons sistem.
4. Mengembangkan visualisasi Digital Twin 3D sebagai replika virtual sistem fisik.
5. Melakukan analisis komparatif performa PID dan Fuzzy pada sistem kendali.

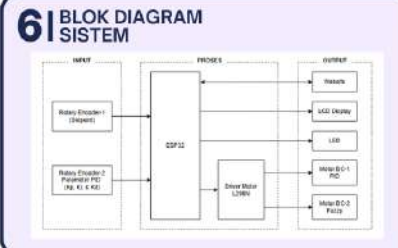
4 | FLOWCHART ALAT



5 | SKEMATIK RANGKAIAN



6 | BLOK DIAGRAM SISTEM



QR CODE MODUL PEMBELAJARAN



Modul Pembelajaran

QR CODE WEB MONITORING



Web Monitoring



L-3. SOP Alat

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



SOP PENGGUNAAN ALAT

TRAINER KIT KENDALI MOTOR DC DUAL – CONTROL PID – FUZZY BERBASIS ESP32 DAN VISUALISASI DIGITAL TWIN 3D

CARA PENGOPERASIAN ALAT

1. **Persiapan Sistem:** Pastikan trainer kit, komputer atau laptop, dan koneksi internet dalam kondisi siap digunakan. Periksa seluruh kabel dan sumber catu daya agar terhubung dengan baik. Pastikan perangkat yang digunakan untuk mengakses website telah terhubung ke jaringan internet yang stabil.
2. **Menyalakan Trainer Kit:** Hubungkan trainer kit ke sumber listrik, kemudian aktifkan sakelar utama pada trainer kit. Tunggu beberapa saat hingga ESP32 berhasil terhubung ke jaringan internet dan sistem siap digunakan.
3. **Mengakses Website Monitoring:** Buka browser pada komputer atau laptop, kemudian masukkan alamat website yang telah disediakan. Tunggu hingga seluruh komponen antarmuka, grafik monitoring, dan visualisasi Digital Twin 3D berhasil dimuat dengan sempurna.
4. **Memilih Mode Operasi:** Pilih mode operasi yang akan digunakan sesuai kebutuhan. Mode ESP32 digunakan untuk melakukan monitoring pada trainer kit fisik, sedangkan Mode Simulasi digunakan untuk menjalankan simulasi sistem tanpa memerlukan perangkat keras.
5. **Mengatur Parameter Sistem:** Masukkan nilai setpoint RPM dan parameter PID yang diinginkan. Setelah seluruh parameter diatur, jalankan sistem dan pastikan data monitoring mulai ditampilkan pada website.
6. **Melakukan Monitoring Sistem:** Amati perubahan grafik respons sistem, grafik PWM, serta visualisasi Digital Twin 3D yang ditampilkan pada website. Pastikan seluruh komponen sistem berjalan secara normal dan data yang ditampilkan berubah secara real-time.
7. **Mengakhiri Pengoperasian:** Setelah proses pengujian selesai, hentikan sistem, tutup website, kemudian matikan trainer kit. Pastikan seluruh perangkat telah dinonaktifkan dengan benar sebelum meninggalkan area praktikum.

ALAT & BAHAN

- ESP32
- Motor DC JGA25 –370 (2 BUAH)
- Driver Motor L298N
- Rotary encoder KY-040 (2 BUAH)
- LCD 20X4
- LED 10MM (3 BUAH)
- Buck Converter 12V to 5V
- Fuse
- Switch
- Kabel 220 VAC
- Power Supply 12V 5A
- Resistor
- Mini Fan DC 12V 5CM

DIBUAT OLEH



Tiar Rizky Budiyo
(2303321038)



Ghani Putra Nuari
(2303321033)



Zulhelmi Syahtiar
(2303321047)

DOSEN PEMBIMBING

Nana Sutarna, S.T., M.T., Ph.D.
(NIP.197007122001121001)

TAMPILAN WEBSITE



QR CODE MODUL PEMBELAJARAN



Modul Pembelajaran

QR CODE WEB MONITORING



Web Monitoring
<https://trainersdtdc.web.app>

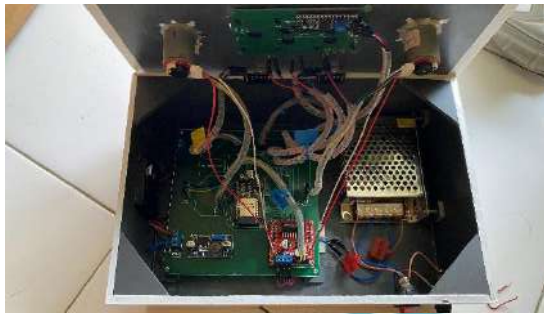


© Hak Cipta milik Politeknik Negeri Jakarta

L-3. Tampilan Alat

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



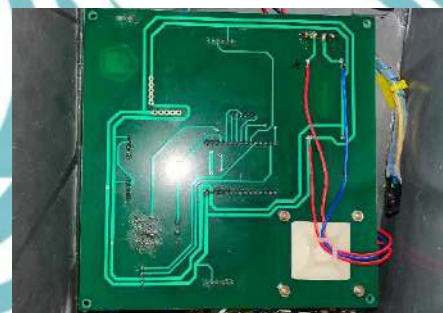
Tampak Dalam Trainer Kit



Tampak Atas Trainer Kit



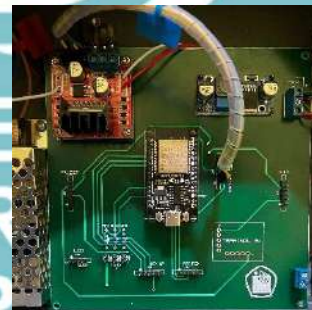
Tampak Depan Trainer Kit



PCB Bottom Layer



Tampak Samping Trainer Kit



PCB Top Layer



© Hak Cipta milik Politeknik Negeri Jakarta

L-4. Tampilan Website

Hak Cipta :

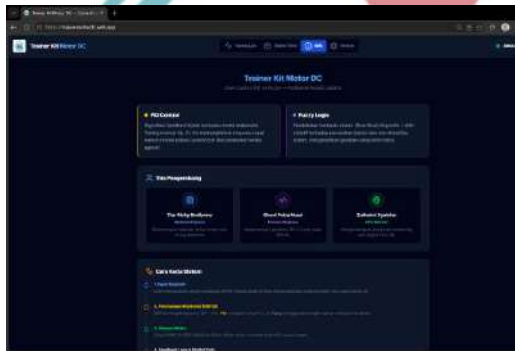
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



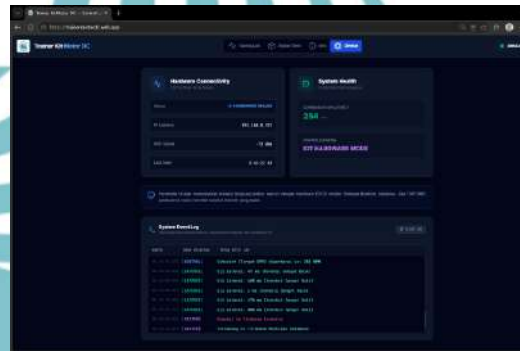
Tampilan Lab



Tampilan Digital Twin 3D



Tampilan Info



Tampilan Device

POLITEKNIK
NEGERI
JAKARTA



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

L-5. Program Algoritma Sistem *Trainer Kit*

```
#include <Arduino.h>

#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#include <soc/gpio_struct.h>

#include <WiFi.h>

#include <HTTPClient.h>

#include <WiFiClientSecure.h>

#include <ArduinoJson.h>

#include <WiFiManager.h>

const String FIREBASE_HOST = "https://trainermotordc-default-rtdb.asia-southeast1.firebaseio.com";

#define ENC_SP_A 34
#define ENC_SP_B 35
#define ENC_SP_BTN 32
#define ENC_PID_A 25
#define ENC_PID_B 26
#define ENC_PID_BTN 27
#define MOTOR1_IN1 16
#define MOTOR1_IN2 17
#define MOTOR1_ENA 4
#define MOTOR1_ENC_A 18
#define MOTOR1_ENC_B 19
#define MOTOR2_IN3 33
#define MOTOR2_IN4 12
#define MOTOR2_ENB 5
#define MOTOR2_ENC_A 23
#define MOTOR2_ENC_B 15
#define LED_MERAH 13
#define LED_KUNING 14
#define LED_HIJAU 2
#define PWM_FREQ 1000
```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
#define PWM_RESOLUTION 8

#define ENCODER_PPR 341 // PPR asli encoder

#define ENCODER_CPR (ENCODER_PPR * 4) // Quadrature 4x = 1364

#define SAMPLE_TIME_S 0.05f
#define SAMPLE_TIME_MS 50

#define MAX_SETPOINT 250
#define MIN_SETPOINT 0

#define DEBOUNCE_MS 250
#define LCD_INTERVAL 300
#define SERIAL_INTERVAL 100
#define SYNC_INTERVAL 1000
#define PWM_DEADBAND_START 18
#define PWM_MIN_EFFECTIVE 22
#define PWM_SMOOTH_ZONE 40

#define GPIO_READ(pin) ((GPIO.in >> (pin)) & 0x1)
#define GPIO_READ_H(pin) ((GPIO.in1.val >> ((pin) - 32)) & 0x1)

float smoothDeadband(float pwm);
float trimf(float x, float a, float b, float c);
float trapmf(float x, float a, float b, float c, float d);
float outNB(float x);
float outNS(float x);
float outZE(float x);
float outPS(float x);
float outPB(float x);
float defuzzify(float* act);
float fuzzyControl(float rpm_target, float actual_rpm);
float pidControl(float rpm_target, float actual_rpm);
float calcRPM(volatile long &encCount, unsigned long dt_ms);

void muError(float e, float &f_NB, float &f_NS, float &f_ZE, float &f_PS, float &f_PB);
void muDeltaError(float de, float &f_NB, float &f_NS, float &f_ZE, float &f_PS, float &f_PB);
void setMotor1(int pwm);
void setMotor2(int pwm);
void updateLCD();
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
void lcdPrintLine(int row, const char* text);
void processEncSP();
void processEncPID();
void processButtons();
void calibrateMaxRPM();
void updateModeLED();
void cloudTask(void* pv);
```

```
LiquidCrystal_I2C lcd(0x27, 20, 4);
int setpoint = 0;
int pidParamMode = 0;
bool powerState = true;
float Kp = 3.0f;
float Ki = 1.2f;
float Kd = 0.08f;
float pid_integral = 0.0f;
float pid_lastError = 0.0f;
float pid_pTerm = 0.0f;
float pid_iTerm = 0.0f;
float pid_dTerm = 0.0f;
float fuzzy_output = 0.0f;
float fuzzy_lastError = 0.0f;
float pwm1_out = 0.0f;
float pwm2_out = 0.0f;
float motor1_rpm_filt = 0.0f;
float motor2_rpm_filt = 0.0f;
float pv_pid = 0.0f;
float pv_fuzzy = 0.0f;
float MAX_RPM_AKTUAL = 156.0f;
volatile long motor1_enc_count = 0;
volatile long motor2_enc_count = 0;
volatile uint8_t enc1_last_state = 0;
volatile uint8_t enc2_last_state = 0;
volatile int enc_sp_delta = 0;
```





© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
volatile int enc_sp_last_a = 0;
volatile int enc_pid_delta = 0;
volatile int enc_pid_last_a = 0;

const int8_t QEM_TABLE[16] = {
    0, -1, 1, 0,
    1, 0, 0, -1,
    -1, 0, 0, 1,
    0, 1, -1, 0
};

volatile bool wifiConnected = false;
volatile bool cloudConnected = false;
volatile bool hardwareChanged = false;
volatile unsigned long lastHardwareChange = 0;

char lcd_last[4][21];
bool lcd_ready = false;

unsigned long lastControlTime = 0;
unsigned long lastLCDTime = 0;
unsigned long lastSerialTime = 0;
unsigned long lastBtnSPTIME = 0;
unsigned long lastBtnPIDTime = 0;

void IRAM_ATTR isr_enc_sp() {
    int a = GPIO_READ_H(ENC_SP_A);
    int b = GPIO_READ_H(ENC_SP_B);
    if (a != enc_sp_last_a) {
        enc_sp_last_a = a;
        enc_sp_delta += (b != a) ? 1 : -1;
    }
}

void IRAM_ATTR isr_enc_pid() {
    int a = GPIO_READ(ENC_PID_A);
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

int b = GPIO_READ(ENC_PID_B);
if (a != enc_pid_last_a) {
    enc_pid_last_a = a;
    enc_pid_delta += (b != a) ? 1 : -1;
}
}

void IRAM_ATTR isr_motor1_enc() {
    uint8_t curr = (GPIO_READ(MOTOR1_ENC_A) << 1) | GPIO_READ(MOTOR1_ENC_B);
    motor1_enc_count += QEM_TABLE[(enc1_last_state << 2) | curr];
    enc1_last_state = curr;
}

void IRAM_ATTR isr_motor2_enc() {
    uint8_t curr = (GPIO_READ(MOTOR2_ENC_A) << 1) | GPIO_READ(MOTOR2_ENC_B);
    motor2_enc_count += QEM_TABLE[(enc2_last_state << 2) | curr];
    enc2_last_state = curr;
}

void calibrateMaxRPM() {
    lcd.clear();
    lcd.setCursor(0, 0); lcd.print(" Kalibrasi... ");
    lcd.setCursor(0, 1); lcd.print(" Harap Tunggu ");
    Serial.println("KALIBRASI:Mulai");
    int pwm_test = 0;
    for (pwm_test = 5; pwm_test <= 50; pwm_test += 2) {
        digitalWrite(MOTOR1_IN1, HIGH); digitalWrite(MOTOR1_IN2, LOW);
        digitalWrite(MOTOR2_IN3, HIGH); digitalWrite(MOTOR2_IN4, LOW);
        ledcWrite(MOTOR1_ENA, pwm_test);
        ledcWrite(MOTOR2_ENB, pwm_test);
        delay(200);
        portDISABLE_INTERRUPTS();
        long c1 = motor1_enc_count; motor1_enc_count = 0;
        long c2 = motor2_enc_count; motor2_enc_count = 0;
        portENABLE_INTERRUPTS();

        if (abs(c1) > 2 || abs(c2) > 2) break;
    }
}

```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

}
Serial.print("KALIBRASI:PWM_START="); Serial.println(pwm_test);

ledcWrite(MOTOR1_ENA, 255);
ledcWrite(MOTOR2_ENB, 255);
delay(1500);

portDISABLE_INTERRUPTS();
motor1_enc_count = 0;
motor2_enc_count = 0;
portENABLE_INTERRUPTS();
delay(500);
portDISABLE_INTERRUPTS();
long count1 = motor1_enc_count;
long count2 = motor2_enc_count;
motor1_enc_count = 0;
motor2_enc_count = 0;
portENABLE_INTERRUPTS();
ledcWrite(MOTOR1_ENA, 0);
ledcWrite(MOTOR2_ENB, 0);
digitalWrite(MOTOR1_IN1, LOW); digitalWrite(MOTOR1_IN2, LOW);
digitalWrite(MOTOR2_IN3, LOW); digitalWrite(MOTOR2_IN4, LOW);
float rpm1_cal = fabsf((float)count1 / (float)ENCODER_CPR / 0.5f * 60.0f);
float rpm2_cal = fabsf((float)count2 / (float)ENCODER_CPR / 0.5f * 60.0f);
float rpm_avg = (rpm1_cal + rpm2_cal) / 2.0f;
if (rpm_avg > 30.0f && rpm_avg < 500.0f) {
    MAX_RPM_AKTUAL = rpm_avg;
} else {
    MAX_RPM_AKTUAL = 156.0f;
}

char buf[21];

lcd.clear();

lcd.setCursor(0, 0); lcd.print(" Kal. Selesai! ");

snprintf(buf, sizeof(buf), " MaxRPM: %5.1f ", MAX_RPM_AKTUAL);

lcd.setCursor(0, 1); lcd.print(buf);

snprintf(buf, sizeof(buf), " PWMstart: %2d ", pwm_test);

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

lcd.setCursor(0, 2); lcd.print(buf);

Serial.print("KALIBRASI:MAX_RPM="); Serial.println(MAX_RPM_AKTUAL, 1);

delay(2000);
}

float smoothDeadband(float pwm) {
    if (pwm <= 0.0f) return 0.0f;
    if (pwm < PWM_DEADBAND_START) return 0.0f;
    if (pwm < PWM_SMOOTH_ZONE) {
        float ratio = (pwm - PWM_DEADBAND_START) / (PWM_SMOOTH_ZONE -
        PWM_DEADBAND_START);
        return PWM_MIN_EFFECTIVE + ratio * (PWM_SMOOTH_ZONE -
        PWM_MIN_EFFECTIVE);
    }
    return pwm;
}

void processEncSP() {
    if (enc_sp_delta == 0) return;
    portDISABLE_INTERRUPTS();
    int delta = enc_sp_delta;
    enc_sp_delta = 0;
    portENABLE_INTERRUPTS();
    int new_sp = setpoint + delta;
    if (new_sp > MAX_SETPOINT) new_sp = MAX_SETPOINT;
    if (new_sp < MIN_SETPOINT) new_sp = MIN_SETPOINT;
    if (new_sp != setpoint) {
        setpoint = new_sp;
        pid_integral = 0.0f;
        pid_lastError = 0.0f;
        fuzzy_output = 0.0f;
        fuzzy_lastError = 0.0f;
        hardwareChanged = true;
        lastHardwareChange = millis();
    }
}

void processEncPID() {

```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

if (enc_pid_delta == 0) return;
portDISABLE_INTERRUPTS();
int delta    = enc_pid_delta;
enc_pid_delta = 0;
portENABLE_INTERRUPTS();

float step;
if  (pidParamMode == 0) step = 0.10f;
else if (pidParamMode == 1) step = 0.01f;
else      step = 0.001f;
float change = (float)delta * step;
float oldKp = Kp, oldKi = Ki, oldKd = Kd;
if (pidParamMode == 0) {
    Kp += change;
    if (Kp < 0.0f) Kp = 0.0f;
    if (Kp > 50.0f) Kp = 50.0f;
} else if (pidParamMode == 1) {
    Ki += change;
    if (Ki < 0.0f) Ki = 0.0f;
    if (Ki > 20.0f) Ki = 20.0f;
    pid_integral = 0.0f;
} else {
    Kd += change;
    if (Kd < 0.0f) Kd = 0.0f;
    if (Kd > 5.0f) Kd = 5.0f;
}
if (Kp != oldKp || Ki != oldKi || Kd != oldKd) {
    hardwareChanged = true;
    lastHardwareChange = millis();
}
}

void updateModeLED() {
    digitalWrite(LED_MERAH, (pidParamMode == 0) ? HIGH : LOW);
    digitalWrite(LED_KUNING, (pidParamMode == 1) ? HIGH : LOW);
    digitalWrite(LED_HIJAU, (pidParamMode == 2) ? HIGH : LOW);
  
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

    }

void processButtons() {
    unsigned long now = millis();

    if (digitalRead(ENC_SP_BTN) == LOW) {
        if ((now - lastBtnSPTime) > DEBOUNCE_MS) {
            lastBtnSPTime = now;

            setpoint = 0;
            pid_integral = 0.0f;
            pid_lastError = 0.0f;
            fuzzy_output = 0.0f;
            fuzzy_lastError = 0.0f;
            pwm1_out = 0.0f;
            pwm2_out = 0.0f;
            pv_pid = 0.0f;
            pv_fuzzy = 0.0f;
            hardwareChanged = true;
            lastHardwareChange = now;
            Serial.println("RESET");
        }
    }

    if (digitalRead(ENC_PID_BTN) == LOW) {
        if ((now - lastBtnPIDTime) > DEBOUNCE_MS) {
            lastBtnPIDTime = now;
            pidParamMode = (pidParamMode + 1) % 3;
            updateModeLED();
            const char* nama[] = {"P", "I", "D"};
            Serial.print("MODE:"); Serial.println(nama[pidParamMode]);
        }
    }
}

float calcRPM(volatile long &encCount, unsigned long dt_ms) {
    portDISABLE_INTERRUPTS();

    long count = encCount;

```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

encCount = 0;

portENABLE_INTERRUPTS();

if (dt_ms == 0) return 0.0f;

return fabsf(((float)count / (float)ENCODER_CPR
              / (dt_ms / 1000.0f) * 60.0f);
}

//Fuzzy

float trimf(float x, float a, float b, float c) {
    if (x <= a || x >= c) return 0.0f;
    if (x <= b) return (x - a) / (b - a);
    return (c - x) / (c - b);
}

float trapmf(float x, float a, float b, float c, float d) {
    if (x <= a || x >= d) return 0.0f;
    if (x >= b && x <= c) return 1.0f;
    if (x < b) return (x - a) / (b - a);
    return (d - x) / (d - c);
}

void muError(float e,
              float &f_NB, float &f_NS,
              float &f_ZE, float &f_PS, float &f_PB) {
    f_NB = trapmf(e, -250, -250, -100, -50);
    f_NS = trimf (e, -100, -20, 0);
    f_ZE = trimf (e, -20, 0, 20);
    f_PS = trimf (e, 0, 20, 100);
    f_PB = trapmf(e, 20, 100, 250, 250);
}

void muDeltaError(float de,
                  float &f_NB, float &f_NS,
                  float &f_ZE, float &f_PS, float &f_PB) {
    f_NB = trapmf(de, -50, -50, -20, -7);
    f_NS = trimf (de, -20, -7, 0);
    f_ZE = trimf (de, -7, 0, 57);

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
f_PS = trimf (de, 0, 7, 20);
f_PB = trapmf(de, 7, 25, 50, 50);
}

float outNB(float x){ return trapmf(x,-30,-30,-15,-7); }
float outNS(float x){ return trimf (x,-15, -7, 0); }
float outZE(float x){ return trimf (x, -7, 0, 7); }
float outPS(float x){ return trimf (x, 0, 7, 15); }
float outPB(float x){ return trapmf(x, 7, 15, 30, 30); }

const uint8_t RULE_BASE[5][5] = {
    { 0, 0, 0, 1, 2 },
    { 0, 0, 1, 2, 3 },
    { 0, 1, 2, 3, 4 },
    { 1, 2, 3, 4, 4 },
    { 2, 3, 4, 4, 4 },
};

float defuzzify(float* act) {
    const int N = 61;
    const float X0 = -30.0f;
    const float X1 = 30.0f;
    float num = 0.0f, den = 0.0f;

    for (int i = 0; i < N; i++) {
        float x = X0 + (X1 - X0) / (float)(N - 1) * (float)i;
        float mu = 0.0f;

        float c0 = (act[0] > 0.0f) ? fminf(act[0], outNB(x)) : 0.0f;
        float c1 = (act[1] > 0.0f) ? fminf(act[1], outNS(x)) : 0.0f;
        float c2 = (act[2] > 0.0f) ? fminf(act[2], outZE(x)) : 0.0f;
        float c3 = (act[3] > 0.0f) ? fminf(act[3], outPS(x)) : 0.0f;
        float c4 = (act[4] > 0.0f) ? fminf(act[4], outPB(x)) : 0.0f;

        if (c0 > mu) mu = c0;
        if (c1 > mu) mu = c1;
        if (c2 > mu) mu = c2;
        if (c3 > mu) mu = c3;
    }
}
```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

if (c4 > mu) mu = c4;
num += x * mu;
den += mu;
}
return (den < 0.0001f) ? 0.0f : (num / den);
}

float fuzzyControl(float rpm_target, float actual_rpm) {
    float e = rpm_target - actual_rpm;
    float de = e - fuzzy_lastError;
    fuzzy_lastError = e;
    float eNB, eNS, eZE, ePS, ePB;
    muError(e, eNB, eNS, eZE, ePS, ePB);
    float eArr[5] = { eNB, eNS, eZE, ePS, ePB };
    float dNB, dNS, dZE, dPS, dPB;
    muDeltaError(de, dNB, dNS, dZE, dPS, dPB);
    float dArr[5] = { dNB, dNS, dZE, dPS, dPB };
    float activation[5] = { 0.0f, 0.0f, 0.0f, 0.0f, 0.0f };
    for (int i = 0; i < 5; i++) {
        for (int j = 0; j < 5; j++) {
            float strength = fminf(eArr[i], dArr[j]);
            if (strength > 0.0f) {
                uint8_t idx = RULE_BASE[i][j];
                if (strength > activation[idx])
                    activation[idx] = strength;
            }
        }
    }
    float deltaPWM = defuzzify(activation);
    float newOutput = fuzzy_output + deltaPWM;
    if (newOutput > 255.0f) {
        newOutput = 255.0f;
        deltaPWM = 0;
    }
    if (newOutput < 0.0f) {

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

newOutput = 0.0f;
deltaPWM = 0;
}
fuzzy_output = newOutput;
return fuzzy_output;
}

//PID
float pidControl(float rpm_target, float actual_rpm) {
    float error = rpm_target - actual_rpm;
    pid_pTerm = Kp * error;
    pid_integral += error * SAMPLE_TIME_S;
    float safeKi = (fabsf(Ki) < 1e-6f) ? 1e-6f : Ki;
    float integralMax = 300.0f / safeKi; // Naik dari 200
    if (pid_integral > integralMax) pid_integral = integralMax;
    if (pid_integral < -integralMax) pid_integral = -integralMax;
    pid_iTerm = Ki * pid_integral;
    float safeDt = (SAMPLE_TIME_S > 0.0f) ? SAMPLE_TIME_S : 0.05f;
    pid_dTerm = Kd * ((error - pid_lastError) / safeDt);
    pid_lastError = error;
    float output = pid_pTerm + pid_iTerm + pid_dTerm;
    float output_clamped = output;
    if (output_clamped > 255.0f) output_clamped = 255.0f;
    if (output_clamped < 0.0f) output_clamped = 0.0f;
    if (output != output_clamped) {
        pid_integral -= error * SAMPLE_TIME_S;
        pid_iTerm = Ki * pid_integral;
    }
    return output_clamped;
}

void setMotor1(int pwm) {
    pwm = constrain(pwm, 0, 255);
    float smoothed = smoothDeadband((float)pwm);
    int finalPwm = (int)roundf(smoothed);
    digitalWrite(MOTOR1_IN1, finalPwm > 0 ? HIGH : LOW);
}

```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
digitalWrite(MOTOR1_IN2, LOW);
ledcWrite(MOTOR1_ENA, finalPwm);
}

void setMotor2(int pwm) {
    pwm = constrain(pwm, 0, 255);
    float smoothed = smoothDeadband((float)pwm);
    int finalPwm = (int)roundf(smoothed);
    digitalWrite(MOTOR2_IN3, finalPwm > 0 ? HIGH : LOW);
    digitalWrite(MOTOR2_IN4, LOW);
    ledcWrite(MOTOR2_ENB, finalPwm);
}

void lcdPrintLine(int row, const char* text) {
    char nb[21];
    memset(nb, ' ', 20); nb[20] = '\0';
    int len = strlen(text);
    if (len > 20) len = 20;
    memcpy(nb, text, len);
    if (!lcd_ready || memcmp(lcd_last[row], nb, 20) != 0) {
        lcd.setCursor(0, row);
        lcd.print(nb);
        memcpy(lcd_last[row], nb, 20);
    }
}

void updateLCD() {
    char buf[21];
    sprintf(buf, sizeof(buf), "SP:%3d  PV_FUZ:%3d",
        setpoint, (int)roundf(pv_fuzzy));
    lcdPrintLine(0, buf);
    // Baris 1
    if (pidParamMode == 0)
        sprintf(buf, sizeof(buf), "Kp>%-6.2f PV_PID:%3d", Kp, (int)roundf(pv_pid));
    else
        sprintf(buf, sizeof(buf), "Kp:%-6.2f PV_PID:%3d", Kp, (int)roundf(pv_pid));
    lcdPrintLine(1, buf);
}
```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
// Baris 2
char wc = (wifiConnected && cloudConnected) ? '*' : (wifiConnected ? 'W' : '!');

if (pidParamMode == 1)
    snprintf(buf, sizeof(buf), "Ki>%-15.3f    %c", Ki, wc); // 15 + 1 = 16, wc di pos 16
else
    snprintf(buf, sizeof(buf), "Ki:%-15.3f    z%c", Ki, wc); // 15 + 1 = 16, wc di pos 16
lcdPrintLine(2, buf);

// Baris 3
if (pidParamMode == 2)
    snprintf(buf, sizeof(buf), "Kd>%-7.4f    %s", Kd, powerState ? "[ON]" : "[OFF]");
else
    snprintf(buf, sizeof(buf), "Kd:%-7.4f    %s", Kd, powerState ? "[ON]" : "[OFF]");
lcdPrintLine(3, buf);
lcd_ready = true;
}

void cloudTask(void* pv) {
    unsigned long lastSync = 0;
    for (;;) {
        unsigned long now = millis();
        if (WiFi.status() == WL_CONNECTED && (now - lastSync) >= SYNC_INTERVAL) {
            lastSync = now;
            WiFiClientSecure sc;
            sc.setInsecure();
            sc.setTimeout(3);
            HTTPClient http;
            http.setReuse(true);
            http.setConnectTimeout(3000);
            http.setTimeout(3000);

            // 1. Kirim telemetri
            http.begin(sc, FIREBASE_HOST + "/monitoring.json");
            http.addHeader("Content-Type", "application/json");
            http.addHeader("Connection", "keep-alive");

            String jp = "{\"rpmPID\": " + String((int)roundf(pv_pid)) +
                ", \"rpmFuzzy\": " + String((int)roundf(pv_fuzzy)) +
```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

        ", \"pwmPID\": \" + String((int)pwm1_out) +
        ", \"pwmFuzzy\": \" + String((int)pwm2_out) +
        ", \"setpoint\": \" + String(setpoint) +
        ", \"maxRpm\": \" + String(MAX_RPM_AKTUAL, 1) + \"}";

int rc = http.PUT(jp);

cloudConnected = (rc == 200 || rc == 201);

http.end();

// 2. Heartbeat + Latency Ping
http.begin(sc, FIREBASE_HOST + "/status.json");
http.addHeader("Content-Type", "application/json");
http.addHeader("Connection", "keep-alive");

String sp2 = "{\"lastSeen\":{\"sv\":\"timestamp\"}, \"ip\": \"\" + WiFi.localIP().toString() +
        \"\", \"rssi\": \" + String(WiFi.RSSI()) + \"}";

http.PUT(sp2);
http.end();

// 3. Upload hardware changes
if (hardwareChanged) {
    http.begin(sc, FIREBASE_HOST + "/control.json");
    http.addHeader("Content-Type", "application/json");
    http.addHeader("Connection", "keep-alive");
    String cp = "{\"setpoint\": \" + String(setpoint) +
        ", \"kp\": \" + String(Kp, 2) +
        ", \"ki\": \" + String(Ki, 3) +
        ", \"kd\": \" + String(Kd, 4) +
        ", \"power\": \" + String(powerState ? \"true\" : \"false\") + \"}";

    rc = http.PATCH(cp);
    if (rc == 200 || rc == 201) hardwareChanged = false;
    http.end();
}

// 4. Baca dari cloud
else if ((now - lastHardwareChange) > 3000) {
    http.begin(sc, FIREBASE_HOST + "/control.json");
    http.setTimeout(2000);
    rc = http.GET();

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

if (rc == 200) {
    String payload = http.getString();
    #if ARDUINOJSON_VERSION_MAJOR >= 7
    JsonDocument doc;
    #else
    StaticJsonDocument<512> doc;
    #endif
    if (!deserializeJson(doc, payload)) {
        if (doc.containsKey("power")) {
            bool newPower = doc["power"].as<bool>();
            if (newPower != powerState) {
                powerState = newPower;
                if (!powerState) {
                    pid_integral = 0.0f; pid_lastError = 0.0f;
                    fuzzy_output = 0.0f; fuzzy_lastError = 0.0f;
                }
            }
        }
        if (doc.containsKey("setpoint")) {
            int wSP = doc["setpoint"].as<int>();
            if (abs(wSP - setpoint) > 1) {
                setpoint = constrain(wSP, MIN_SETPOINT, MAX_SETPOINT);
                pid_integral = 0.0f; pid_lastError = 0.0f;
                fuzzy_output = 0.0f; fuzzy_lastError = 0.0f;
                hardwareChanged = true;
                lastHardwareChange = now;
            }
        }
        if (doc.containsKey("kp")) {
            float newKp = constrain(doc["kp"].as<float>(), 0.0f, 50.0f);
            if (fabsf(newKp - Kp) > 0.01f) { Kp = newKp; hardwareChanged = true;
lastHardwareChange = now; }
        }
        if (doc.containsKey("ki")) {
            float newKi = constrain(doc["ki"].as<float>(), 0.0f, 20.0f);

```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

        if (fabsf(newKi - Ki) > 0.001f) { Ki = newKi; pid_integral = 0.0f; hardwareChanged = true;
lastHardwareChange = now; }

    }

    if (doc.containsKey("kd")) {

        float newKd = constrain(doc["kd"].as<float>(), 0.0f, 5.0f);

        if (fabsf(newKd - Kd) > 0.0001f) { Kd = newKd; hardwareChanged = true;
lastHardwareChange = now; }

    }

    }

    http.end();

}

}

else if (WiFi.status() != WL_CONNECTED) {

    cloudConnected = false;

}

vTaskDelay(pdMS_TO_TICKS(50));

}

}

void setup() {

    Serial.begin(115200);

    delay(300);

    // LCD

    Wire.begin(21, 22);

    lcd.init();

    lcd.backlight();

    lcd.clear();

    lcd.setCursor(0, 0); lcd.print("  TRAINER KIT  ");

    lcd.setCursor(0, 1); lcd.print(" PID-FUZZY ESP32  ");

    lcd.setCursor(0, 2); lcd.print(" Dual Control DC  ");

    lcd.setCursor(0, 3); lcd.print(" Motor Trainer  ");

    delay(2000);

    lcd.clear();

    pinMode(LED_MERAH,      OUTPUT);      pinMode(LED_KUNING,      OUTPUT);
pinMode(LED_HIJAU, OUTPUT);

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

updateModeLED();

pinMode(MOTOR1_IN1, OUTPUT); pinMode(MOTOR1_IN2, OUTPUT);
pinMode(MOTOR2_IN3, OUTPUT); pinMode(MOTOR2_IN4, OUTPUT);

digitalWrite(MOTOR1_IN1, LOW); digitalWrite(MOTOR1_IN2, LOW);
digitalWrite(MOTOR2_IN3, LOW); digitalWrite(MOTOR2_IN4, LOW);

ledcAttach(MOTOR1_ENA, PWM_FREQ, PWM_RESOLUTION);
ledcAttach(MOTOR2_ENB, PWM_FREQ, PWM_RESOLUTION);

ledcWrite(MOTOR1_ENA, 0);
ledcWrite(MOTOR2_ENB, 0);

pinMode(MOTOR1_ENC_A, INPUT_PULLUP);      pinMode(MOTOR1_ENC_B,
INPUT_PULLUP);
pinMode(MOTOR2_ENC_A, INPUT_PULLUP);      pinMode(MOTOR2_ENC_B,
INPUT_PULLUP);

enc1_last_state = (digitalRead(MOTOR1_ENC_A) << 1) | digitalRead(MOTOR1_ENC_B);
enc2_last_state = (digitalRead(MOTOR2_ENC_A) << 1) | digitalRead(MOTOR2_ENC_B);
attachInterrupt(digitalPinToInterrupt(MOTOR1_ENC_A), isr_motor1_enc, CHANGE);
attachInterrupt(digitalPinToInterrupt(MOTOR1_ENC_B), isr_motor1_enc, CHANGE);
attachInterrupt(digitalPinToInterrupt(MOTOR2_ENC_A), isr_motor2_enc, CHANGE);
attachInterrupt(digitalPinToInterrupt(MOTOR2_ENC_B), isr_motor2_enc, CHANGE);
pinMode(ENC_SP_A, INPUT_PULLUP); pinMode(ENC_SP_B, INPUT_PULLUP);
pinMode(ENC_SP_BTN, INPUT_PULLUP);
pinMode(ENC_PID_A, INPUT_PULLUP); pinMode(ENC_PID_B, INPUT_PULLUP);
pinMode(ENC_PID_BTN, INPUT_PULLUP);
enc_sp_last_a = GPIO_READ_H(ENC_SP_A);
enc_pid_last_a = GPIO_READ(ENC_PID_A);
attachInterrupt(digitalPinToInterrupt(ENC_SP_A), isr_enc_sp, CHANGE);
attachInterrupt(digitalPinToInterrupt(ENC_PID_A), isr_enc_pid, CHANGE);

memset(lcd_last, ' ', sizeof(lcd_last));

lcd_ready = false;

lcd.clear();

lcd.setCursor(0, 0); lcd.print("Connecting WiFi... ");
lcd.setCursor(0, 1); lcd.print("Timeout: 30s ");
Serial.println("WIFI:Connecting");

WiFiManager wm;

wm.setConnectTimeout(30);

```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

wm.setConfigPortalTimeout(60);
if (wm.autoConnect("TrainerKit_AP")) {
    wifiConnected = true;
    lcd.clear();
    lcd.setCursor(0, 0); lcd.print("WiFi Connected!  ");
    lcd.setCursor(0, 1); lcd.print(WiFi.localIP().toString());
    Serial.print("WIFI:"); Serial.println(WiFi.localIP());
    delay(1500);
    xTaskCreatePinnedToCore(cloudTask, "CloudTask", 8192, NULL, 2, NULL, 0);
}
else {
    wifiConnected = false;
    cloudConnected = false;
    lcd.clear();
    lcd.setCursor(0, 0); lcd.print("WiFi Gagal!  ");
    lcd.setCursor(0, 1); lcd.print("Mode Offline  ");
    Serial.println("WIFI:Offline");
    delay(1500);
}
calibrateMaxRPM();
lastControlTime = millis();
lastLCDTime = millis();
lcd.clear();
Serial.println("READY");
Serial.println("HEADER:ms,SP,PV_PID,PV_FUZZY,RPM1,RPM2,PWM1,PWM2,Kp,Ki,Kd");
}

void loop() {
    unsigned long now = millis();
    processEncSP();
    processEncPID();
    processButtons();

    if ((now - lastControlTime) >= SAMPLE_TIME_MS) {
        unsigned long dt = now - lastControlTime;

```



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

lastControlTime = now;

float rpm1 = calcRPM(motor1_enc_count, dt);

float rpm2 = calcRPM(motor2_enc_count, dt);

motor1_rpm_filt = 0.85f * motor1_rpm_filt + 0.15f * rpm1;
motor2_rpm_filt = 0.85f * motor2_rpm_filt + 0.15f * rpm2;

pv_pid = (motor1_rpm_filt / MAX_RPM_AKTUAL) * (float)MAX_SETPOINT;
pv_fuzzy = (motor2_rpm_filt / MAX_RPM_AKTUAL) * (float)MAX_SETPOINT;
if (pv_pid > (float)MAX_SETPOINT) pv_pid = (float)MAX_SETPOINT;
if (pv_fuzzy > (float)MAX_SETPOINT) pv_fuzzy = (float)MAX_SETPOINT;
if (pv_pid < 0.0f) pv_pid = 0.0f;
if (pv_fuzzy < 0.0f) pv_fuzzy = 0.0f;
float rpm_target = ((float)setpoint / (float)MAX_SETPOINT) * MAX_RPM_AKTUAL;
if (powerState && setpoint > 0) {
    pwm2_out = fuzzyControl(rpm_target, motor2_rpm_filt);
    setMotor2((int)pwm2_out);
    pwm1_out = pidControl(rpm_target, motor1_rpm_filt);
    setMotor1((int)pwm1_out);
} else {
    pwm1_out = 0.0f; pwm2_out = 0.0f;
    pid_integral = 0.0f;
    fuzzy_output = 0.0f;
    setMotor1(0);
    setMotor2(0);
}

if ((now - lastSerialTime) >= SERIAL_INTERVAL) {
    lastSerialTime = now;
    const char* modeNama[] = {"P", "I", "D"};
    Serial.print("DATA:");
    Serial.print(now);          Serial.print(",");
    Serial.print(setpoint);      Serial.print(",");
    Serial.print((int)roundf(pv_pid));    Serial.print(",");
    Serial.print((int)roundf(pv_fuzzy));  Serial.print(",");
    Serial.print(motor1_rpm_filt, 1); Serial.print(",");
  
```




© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

Serial.print(motor2_rpm_filt, 1); Serial.print(",");
Serial.print((int)pwm1_out);   Serial.print(",");
Serial.print((int)pwm2_out);   Serial.print(",");
Serial.print(Kp, 2);           Serial.print(",");
Serial.print(Ki, 3);           Serial.print(",");
Serial.println(Kd, 4);
Serial.print("SP:");   Serial.print(setpoint);   Serial.print(" ");
Serial.print("PV_PID:"); Serial.print((int)roundf(pv_pid));   Serial.print(" ");
Serial.print("PV_FUZ:"); Serial.print((int)roundf(pv_fuzzy));   Serial.print(" ");
Serial.print("RPM1:");   Serial.print(motor1_rpm_filt, 1); Serial.print(" ");
Serial.print("RPM2:");   Serial.print(motor2_rpm_filt, 1); Serial.print(" ");
Serial.print("PWM1:");   Serial.print((int)pwm1_out);   Serial.print(" ");
Serial.print("PWM2:");   Serial.print((int)pwm2_out);   Serial.print(" ");
Serial.print("Kp:");   Serial.print(Kp, 2);   Serial.print(" ");
Serial.print("Ki:");   Serial.print(Ki, 3);   Serial.print(" ");
Serial.print("Kd:");   Serial.print(Kd, 4);   Serial.print(" ");
Serial.print("PWR:");   Serial.print(powerState ? "ON" : "OFF"); Serial.print(" ");
Serial.print("MODE:");   Serial.println(modeNama[pidParamMode]);
}
}
if ((now - lastLCDTime) >= LCD_INTERVAL) {
    lastLCDTime = now;
    updateLCD();
}
}

```

**POLITEKNIK
NEGERI
JAKARTA**



L-6. Datasheet Motor JGA250-370



MODEL:25GA370



输出功率: 约0.3~10.0W
精密金属齿轮减速电机

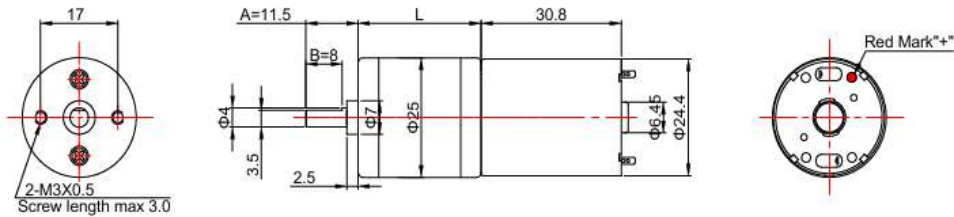
典型应用: 智能家居
智能机器人
医疗器械
电动玩具

OUTPUT: APPROX 0.3~10.0W
Precision Metal Gear Motor

Typical Applications: Smart Home
Smart Robot
Medical Instruments
Electric Toys

1. OUTLINE DRAWING:

UNIT:MM



Customizable Parameters:

1. Length L, A and B
2. Electric Performance
3. Output shaft specifications
4. Reduction Ratio

WEIGHT: 95g(Approx)

2. GEARBOX REDUCTION RATIO:

Gearbox Length (mm)	Stage Number	Reduction Ratio						Rated Torque	Bearable Maximum Instant Torque
								kg.cm	
19	4	15.5/1	20.4/1					1.0	3.0
21	5	26/1	30/1	34/1	45/1			2.0	6.0
23	6	57/1	75/1	98/1				2.5	7.0
25	7	125/1	165/1	217/1				3.0	8.0
27	8	274/1	362/1	478/1				3.0	8.0

3. MOTOR PERFORMANCE :

1N.m ≈ 10.2kg.cm ≈ 14.2oz.in

MODEL	REDUCTION RATIO	VOLTAGE		NO LOAD		AT MAXIMUM EFFICIENCY				STALL	
		OPERATING RANGE	NOMINAL	SPEED	CURRENT	SPEED	CURRENT	TORQUE	OUTPUT	TORQUE	CURRENT
			V	rpm	A	rpm	A	kg.cm	W	kg.cm	A
25GA370-26-12231	26/1	3.0~12.0	12.0	231	0.09	162	0.43	1.1	1.83	3.7	1.2
25GA370-20.4-12460	20.4/1	3.0~12.0	12.0	400	0.05	340	0.36	0.4	1.39	2.6	2.1
25GA370-98-0650	98/1	3.0~6.0	6.0	50	0.05	41	0.21	1.6	0.67	9.8	1.06
25GA370-165-1288	165/1	3.0~12.0	12.0	88	0.26	43	1.27	6.0	2.65	35.3	6.2
25GA370-478-1207.9	478/1	3.0~12.0	12.0	7.8	0.03	7	0.07	2.0	0.14	19.5	0.52

<http://www.tronsunmotor.com>

Email: info@tronsunmotor.com

+86(755)23771495/96/97

17

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

L-7. Datasheet Driver L298N

L298N DC Motor Driver

Specs

Input Voltage: 6-35 VDC

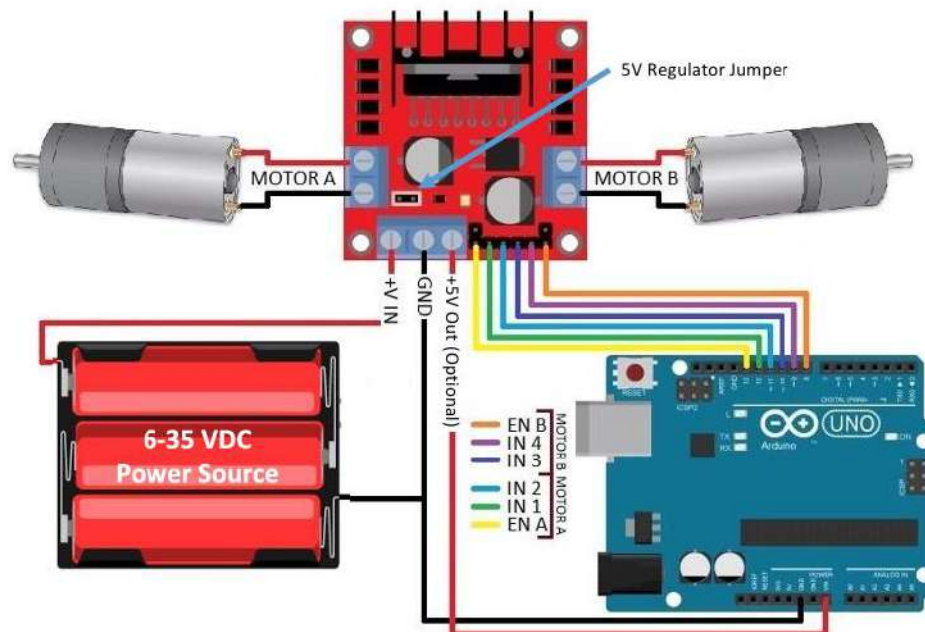
Motor Drive Voltage: 0-35 VDC

Driver Current: 2A MAX Continuous per channel (25W max power dissipation)

Logic Voltage: 2-5V On/High, <2V Off/Low

Max Current Output of 5V OUT: 500mA*** (See important note at bottom)

The L298N motor driver can control one or two motors. It requires a power source (6-35V DC) and PWM / digital signals to control rotational speed and direction. To use the motor driver, follow the wiring diagram below (any micro controller will work, not just an Arduino).



Minimum Connections

- Connect terminals labeled "+12V" and "GND" to 6-35V power supply positive and negative. Remember to connect the microcontroller ground as well. The red LED light indicates that the board is receiving power and should illuminate
- Connect the motor output Positive and Negative terminals to the motor
- Logic connections as detailed in the tables below (5V MAX)



Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Fixed Speed Operation (Full speed, forward or reverse)

- Verify that the 5V Regulator Jumper and both ENA and ENB jumpers are in place
- For forward rotation, connect pin IN1 (or IN3 for motor B) to a +5V source (such as a digital pin on a microcontroller or a power supply). For reverse rotation, connect pin IN2 (or IN4 for motor B) to +5V source. This will operate the motor at full speed either forward or reverse.

IN1 (IN3)	IN2 (IN4)	ENABLE A/B	Motor Output
5 volts	0 volts	5 volts*	Full speed Forward
0 volts	5 volts	5 volts*	Full speed Reverse

*Use the provided jumpers to pull the front Enable A/B pin up to 5 volts using the 5 volts provided by the rear pin.

Variable Speed Operation

To control motor speed, you must use a PWM signal. Speed is proportional to % Duty Cycle. You cannot use analog voltage to control speed. Connect logic pins as shown in tables below

Control method 1: Speed proportional to PWM on IN1/IN2. (Method 1 is better for slower speeds but requires two PWM signals per motor for Forward/Reverse)

IN1 (IN3)	IN2 (IN4)	ENABLE A/B	Motor Output
PWM (1.5 kHz, 5V Amp., 2.5 V Offset)	0 volts	5 volts*	Variable speed Fwd.
0 volts	PWM (1.5 kHz, 5V Amp., 2.5 V Offset)	5 volts*	Variable speed Rev.

*Use the provided jumpers to pull the front Enable A/B pin up to 5 volts using the 5 volts provided by the rear pin.

Control method 2: Speed proportional to PWM on ENABLE A/B. (Method 2 requires only one PWM signal per motor, however there is significant power loss causing you to lose the bottom half of the motor RPM range)

IN1 (IN3)	IN2 (IN4)	ENABLE A/B	Motor Output
5 volts	0 volts	PWM (1.5 kHz, 5V Amp., 2.5 V Offset)	Variable speed Forward
0 volts	5 volts	PWM (1.5 kHz, 5V Amp., 2.5 V Offset)	Variable speed Reverse

*Remove the provided jumpers on the Enable A/B pins and instead connect the PWM signal on the front pin.

Troubleshooting

Motor makes a "whining" noise and does not spin

This usually occurs when the motor is not receiving sufficient voltage to move. Try increasing your duty cycle.

Motor is weak at low speeds or low Duty Cycle

If you are currently controlling the motor using Method 2 above, try using Method 1. Method 1 allows for a lower Duty Cycle before the motor cuts off. If using Method 1 is not possible, try decreasing the voltage supplied to the board so the motor receives a lower voltage at a higher duty cycle.

*****NOTE ON 5V OUTPUT MAX CURRENT:** If you are using the 5V OUTPUT to supply power to some external load, you *must* not exceed 12V on the +VIN connection. You *cannot* go up to the 35V listed above. Doing so will overheat the 5V regulator and temporarily disable the chip. Current capacity at 12V is 500mA.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

L-8. Datasheet ESP32

2 Pin Definitions

2.1 Pin Layout

The pin diagram below shows the approximate location of pins on the module. For the actual diagram drawn to scale, please refer to Figure 9 *Module Dimensions*.

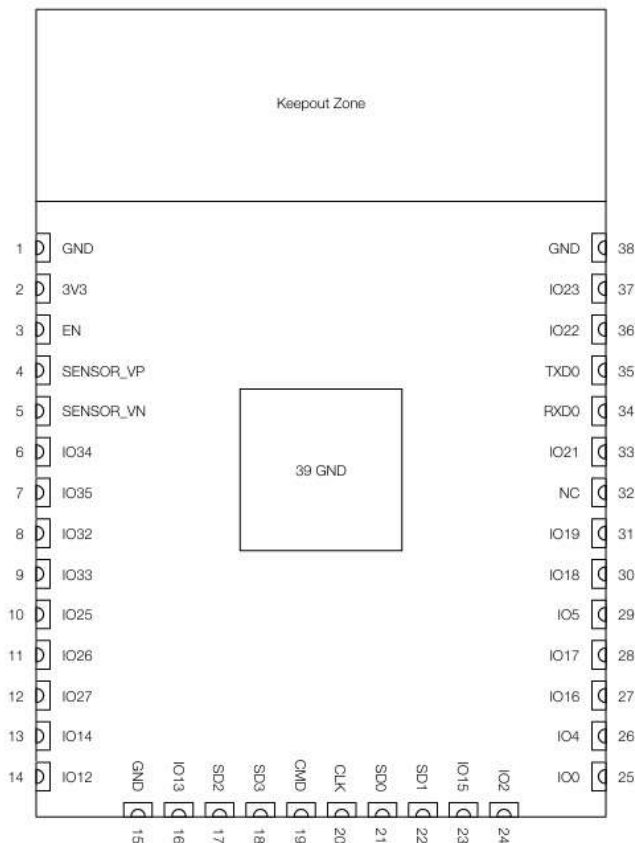


Figure 1: ESP32-WROOM-32 Pin Layout (Top View)

2.2 Pin Description

ESP32-WROOM-32 has 38 pins. See pin definitions in Table 2.

For peripheral pin configurations, please refer to Section 4.2 *Digital Peripherals*.



Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Table 2: Pin Definitions

Name	No.	Type ¹	Function
GND	1	P	Ground
3V3	2	P	Power supply
EN	3	I	Module-enable signal. Active high.
SENSOR_VP	4	I	GPIO36, ADC1_CH0, RTC_GPIO0
SENSOR_VN	5	I	GPIO39, ADC1_CH3, RTC_GPIO3
IO34	6	I	GPIO34, ADC1_CH6, RTC_GPIO4
IO35	7	I	GPIO35, ADC1_CH7, RTC_GPIO5
IO32	8	I/O	GPIO32, XTAL_32K_P (32.768 kHz crystal oscillator input), ADC1_CH4, TOUCH9, RTC_GPIO9
IO33	9	I/O	GPIO33, XTAL_32K_N (32.768 kHz crystal oscillator output), ADC1_CH5, TOUCH8, RTC_GPIO8
IO25	10	I/O	GPIO25, DAC_1, ADC2_CH8, RTC_GPIO6, EMAC_RXD0
IO26	11	I/O	GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1
IO27	12	I/O	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
IO14	13	I/O	GPIO14, ADC2_CH6, TOUCH6, RTC_GPIO16, MTMS, FSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2
IO12	14	I/O	GPIO12, ADC2_CH5, TOUCH5, RTC_GPIO15, MTDI, HSPICQ, HS2_DATA2, SD_DATA2, EMAC_TXD3
GND	15	P	Ground
IO13	16	I/O	GPIO13, ADC2_CH4, TOUCH4, RTC_GPIO14, MTCK, HSPID, HS2_DATA3, SD_DATA3, EMAC_RX_ER
SHD/SD2 ²	17	I/O	GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD
SWP/SD3 ²	18	I/O	GPIO10, SD_DATA3, SWP, HS1_DATA3, U1TXD
SCS/CMD ²	19	I/O	GPIO11, SD_CMD, SPICSO, HS1_CMD, U1RTS
SCK/CLK ²	20	I/O	GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS
SDO/SDO ²	21	I/O	GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS
SDI/SD1 ²	22	I/O	GPIO8, SD_DATA1, SPID, HS1_DATA1, U2CTS
IO15	23	I/O	GPIO15, ADC2_CH3, TOUCH3, MTD0, HSPICSO, RTC_GPIO13, HS2_CMD, SD_CMD, EMAC_RXD3
IO2	24	I/O	GPIO2, ADC2_CH2, TOUCH2, RTC_GPIO12, HSPICP, HS2_DATA0, SD_DATA0
IO0	25	I/O	GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK
IO4	26	I/O	GPIO4, ADC2_CH0, TOUCH0, RTC_GPIO10, HSPICD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
IO16	27	I/O	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
IO17	28	I/O	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
IO5	29	I/O	GPIO5, VSPICSO, HS1_DATA6, EMAC_RX_CLK
IO18	30	I/O	GPIO18, VSPICLK, HS1_DATA7
IO19	31	I/O	GPIO19, VSPICQ, UOCTS, EMAC_TXD0
NC	32	-	-
IO21	33	I/O	GPIO21, VSPICD, EMAC_TX_EN
RXD0	34	I/O	GPIO3, U0RXD, CLK_OUT2
TXD0	35	I/O	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Stresses beyond the absolute maximum ratings listed in Table 12 *Absolute Maximum Ratings* below may cause permanent damage to the device. These are stress ratings only, and do not refer to the functional operation of the device that should follow the Table 13 *Recommended Operating Conditions*.

Table 12: Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Unit
VDD33	Power supply voltage	-0.3	3.6	V
I_{output}^1	Cumulative IO output current	-	1,100	mA
T_{store}	Storage temperature	-40	105	°C

1. The module worked properly after a 24-hour test in ambient temperature at 25 °C, and the IOs in three domains (VDD3P3_RTC, VDD3P3_CPU, VDD_SDIO) output high logic level to ground. Please note that pins occupied by flash and/or PSRAM in the VDD_SDIO power domain were excluded from the test.
2. Please see Appendix IO_MUX of [ESP32 Datasheet](#) for IO's power domain.

5.2 Recommended Operating Conditions

Table 13: Recommended Operating Conditions

Symbol	Parameter	Min	Typical	Max	Unit
VDD33	Power supply voltage	3.0	3.3	3.6	V
I_{VDD}	Current delivered by external power supply	0.5	-	-	A
T	Operating ambient temperature	-40	-	85	°C

5.3 DC Characteristics (3.3 V, 25 °C)

Table 14: DC Characteristics (3.3 V, 25 °C)

Symbol	Parameter	Min	Typ	Max	Unit
C_{IN}	Pin capacitance	-	2	-	pF
V_{IH}	High-level input voltage	$0.75 \times VDD^1$	-	$VDD^1 + 0.3$	V
V_{IL}	Low-level input voltage	-0.3	-	$0.25 \times VDD^1$	V
I_{IH}	High-level input current	-	-	50	nA
I_{IL}	Low-level input current	-	-	50	nA
V_{OH}	High-level output voltage	$0.8 \times VDD^1$	-	-	V
V_{OL}	Low-level output voltage	-	-	$0.1 \times VDD^1$	V
I_{OH}	High-level source current (VDD ¹ = 3.3 V, V _{OH} >= 2.64 V, output drive strength set to the maximum)	VDD3P3_CPU power domain ^{1, 2}	40	-	mA
		VDD3P3_RTC power domain ^{1, 2}	40	-	mA
		VDD_SDIO power domain ^{1, 3}	20	-	mA



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

6 RF Characteristics

This section contains tables with RF characteristics of the Espressif product.

The RF data is measured at the antenna port, where RF cable is connected, including the front-end loss.

Devices should operate in the center frequency range allocated by regional regulatory authorities. The target center frequency range and the target transmit power are configurable by software. See [ESP RF Test Tool and Test Guide](#) for instructions.

Unless otherwise stated, the RF tests are conducted with a 3.3 V ($\pm 5\%$) supply at 25 °C ambient temperature.

6.1 Wi-Fi Radio

Table 16: Wi-Fi RF Characteristics

Name	Description
Center frequency range of operating channel	2412 ~ 2484 MHz
Wi-Fi wireless standard	IEEE 802.11b/g/n

6.1.1 Wi-Fi RF Transmitter (TX) Characteristics

Table 17: TX Power with Spectral Mask and EVM Meeting 802.11 Standards

Rate	Min (dBm)	Typ (dBm)	Max (dBm)
802.11b, 1 Mbps	—	19.5	—
802.11b, 11 Mbps	—	19.5	—
802.11g, 6 Mbps	—	18.0	—
802.11g, 54 Mbps	—	14.0	—
802.11n, HT20, MCS0	—	18.0	—
802.11n, HT20, MCS7	—	13.0	—
802.11n, HT40, MCS0	—	18.0	—
802.11n, HT40, MCS7	—	13.0	—

Table 18: TX EVM Test[†]

Rate	Min (dB)	Typ (dB)	Limit (dB)
802.11b, 1 Mbps, DSSS	—	-25.0	-10.0
802.11b, 11 Mbps, CCK	—	-25.0	-10.0
802.11g, 6 Mbps, OFDM	—	-24.0	-5.0
802.11g, 54 Mbps, OFDM	—	-28.0	-25.0
802.11n, HT20, MCS0	—	-24.0	-5.0
802.11n, HT20, MCS7	—	-30.0	-27.0

Cont'd on next page