



Sistem Kontrol Konveyor dengan PID menggunakan Perintah Suara dan Monitoring berbasis Visual C#

Tugas Akhir

Oleh:

Wahyu Firdaus (4212101082)

Muhammad Zidni Aufa (4212101088)

Program Studi Teknik Mekatronika

Jurusan Teknik Elektro

Politeknik Negeri Batam

2024

Pernyataan Keaslian Tugas Akhir

Saya yang bertandatangan dibawah ini menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya yang berjudul : "Sistem Kontrol Konveyor dengan PID menggunakan Perintah Suara dan Monitoring berbasis Visual C#" adalah **hasil karya sendiri**, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan, dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri. Semua referensi yang dikutip atau dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan saya ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.



Nama Wahyu Firdaus
NIM: 4212101082



Batam, 30 Januari 2025
Nama Muhammad Zidni Afa
NIM: 4212101088

Lembar Pengesahan

Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Terapan Teknik (S.Tr.T)
di
Politeknik Negeri Batam

Oleh:
Wahyu Firdaus (4212101082)
Muhammad Zidni Aufa (4212101088)

Tanggal Sidang: 14, 01, 2025

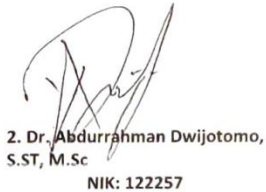
Disetujui oleh :



1. Diong, S.Tr.T., M.Sc
NIK: 120243



1. Adlian Jefiza, S.Pd., M.T.
NIK: 119220



2. Dr. Abdurrahman Dwijotomo,
S.ST, M.Sc
NIK: 122257

[Sistem Kontrol Konveyor dengan PID menggunakan Perintah Suara dan Monitoring berbasis Visual C#]

Abstrak

Dalam era Revolusi Industri 4.0, otomatisasi menjadi elemen penting dalam meningkatkan efisiensi dan produktivitas di berbagai sektor industri. Penelitian ini mengembangkan sistem kontrol konveyor yang mengintegrasikan kendali PID (*Proportional-Integral-Derivative*), perintah suara, dan pemantauan berbasis Visual C#. Penerapan pengendali PID terbukti secara signifikan meningkatkan stabilitas dan akurasi motor DC, dengan rata-rata error tanpa PID sebesar 31.52% yang turun drastis menjadi 0.94% setelah menggunakan PID, menggunakan parameter $K_p = 0.26$, $K_i = 0.009$, dan $K_d = 0.0011$. Sistem ini dirancang untuk meningkatkan fleksibilitas pengoperasian konveyor melalui teknologi pengenalan suara yang memungkinkan interaksi intuitif antara manusia dan mesin, serta pemantauan real-time menggunakan Visual C#. Dengan mengurangi keterlibatan manusia dalam tugas berbahaya dan repetitif, sistem ini tidak hanya meningkatkan keselamatan kerja tetapi juga mengoptimalkan proses produksi. Selain itu, kemampuan untuk mendeteksi lebih awal potensi kerusakan pada konveyor memungkinkan tindakan preventif yang dapat mengurangi downtime produksi.

Kata Kunci : Motor DC, Kontrol PID, ESP32, Protokol ESP-NOW, Kontrol Suara, Monitoring Visual C#.

[Conveyor Control System with PID using Voice Commands and Visual C# based Monitoring]

Abstract

In the era of the Industrial Revolution 4.0, automation is an important element in improving efficiency and productivity in various industrial sectors. This research develops a conveyor control system that integrates PID (Proportional-Integral-Derivative) control, voice commands, and Visual C#-based monitoring. The application of the PID controller was shown to significantly improve the stability and accuracy of the DC motor, with an average error without PID of 31.52% which dropped dramatically to 0.94% after using PID, using parameters $K_p = 0.26$, $K_i = 0.009$, and $K_d = 0.0011$. The system is designed to increase the flexibility of conveyor operations through voice recognition technology that enables intuitive interaction between humans and machines, as well as real-time monitoring using Visual C#. By reducing human involvement in dangerous and repetitive tasks, the system not only improves work safety but also optimizes the production process. In addition, the ability to detect early potential damage to the conveyor enables preventive actions that can reduce production downtime

Keywords : *DC motor, PID control, ESP32, ESP-NOW protocol, voice control, Visual C# monitoring.*

Kata Pengantar

Dengan penuh rasa syukur, penulis mengucapkan terima kasih kepada Allah SWT yang telah memberikan rahmat, hidayah, dan kekuatan-Nya, sehingga Penulis dapat menyelesaikan Tugas Akhir yang berjudul "Sistem Kontrol Konveyor dengan PID menggunakan Perintah Suara dan Monitoring berbasis Visual C#" tepat waktu.

Penulis juga ingin menyampaikan rasa terima kasih yang mendalam kepada semua pihak yang telah memberikan dukungan, ilmu, dan waktu yang berharga selama proses penyusunan Tugas Akhir ini. Tugas Akhir ini merupakan bagian dari syarat untuk meraih gelar S.Tr.T di Program Studi Teknik Mekatronika, Jurusan Teknik Elektro, Politeknik Negeri Batam. Penulis menyadari bahwa penyelesaian Tugas Akhir ini tidak akan terwujud tanpa bimbingan dan bantuan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis mengucapkan terima kasih kepada :

1. Bapak Adlian Jefiza, S.Pd., M.T., sebagai dosen pembimbing, atas bimbingan, arahan, dan motivasinya dalam menyelesaikan tugas akhir ini.
2. Bapak Diono, S.Tr.T., M.Sc, sebagai dosen penguji 1, atas saran, masukan, dan koreksi yang membangun.
3. Bapak Dr. Abdurrahman Dwijotomo, S.ST, M.Sc, sebagai dosen penguji 2, atas saran, masukan, dan koreksi yang diberikan.
4. Orang tua, keluarga, dan teman-teman atas dukungan moral, material, dan doa untuk menyelesaikan tugas akhir ini.
5. Semua pihak yang telah membantu secara langsung maupun tidak langsung dalam penyelesaian tugas akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna karena adanya keterbatasan ilmu dan pengalaman yang dimiliki. Oleh karena itu, semua kritik dan saran yang bersifat membangun akan penulis terima dengan senang hati. Penulis berharap, semoga Tugas Akhir ini dapat bermanfaat bagi semua pihak yang memerlukan.

Batam, 30 Januari 2025



Wahyu Firdaus & Muhammad Zidni Aufa

Daftar Isi

Pernyataan Keaslian Tugas Akhir	i
Lembar Pengesahan	ii
Abstrak	iii
<i>Abstract</i>	iv
Kata Pengantar	v
Daftar Isi	vi
Daftar Gambar	ix
Daftar Tabel	x
Bab 1. Pendahuluan	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan	3
1.4. Manfaat	3
1.5. Batasan	3
1.6. <i>Work Breakdown Structure</i> (Opsional)	4
Bab 2. Tinjauan Pustaka	5
2.1 Penelitian Terdahulu	5
2.2 Konveyor	6
2.3 Motor DC	6
2.4 PID (<i>Proportional, Integral, Derivative</i>)	7
2.5 Speech Recognition	7
2.6 <i>Monitoring</i>	8
2.8 ESP32	8
2.9 ESP-NOW	9
2.10 Database	9
2.11 C#	9

2.12 XAMPP.....	10
2.13 Motor Driver.....	10
2.14 Sensor ACS712.....	11
2.15 Sensor Tegangan	11
Bab 3. Metodologi Penelitian/Metode Pelaksanaan	18
3.1 Alur Penelitian	18
3.1.1 Perancangan Sistem	19
3.1.2 Perancangan Elektrikal	21
3.1.3 Perancangan Mekanikal	24
3.1.4 Perancangan Program	26
3.2. Alat dan Bahan	29
3.3. Pengujian.....	30
3.3.1 Pengujian Mekanikal	30
Bab 4. Hasil dan Pembahasan.....	33
4.1 Hasil Pengujian Mekanikal.....	33
4.2 Hasil Pengujian Elektrikal	34
4.2.1 Hasil Pengujian Menggunakan Sensor.....	35
4.2.2 Hasil Pengujian Menggunakan Alat Ukur	36
4.2.3 Hasil Pengujian Perhitungan Error Sistem	37
4.3 Hasil Pengujian GUI	38
4.3.1 Hasil Pengujian Pada Menu Login.....	38
4.3.2 Hasil Pengujian Menu kontrol	38
4.3.3 Pengujian Speech Recognition	39
4.3.4 Hasil Pengujian Menu Monitoring	40
4.3.3 Hasil Pengujian Management User	41
4.4 Hasil Pengujian <i>PID Controller</i> pada Sistem.....	42
4.4.1 Pengujian terhadap Setpoint Tanpa <i>PID Controller</i>	43
4.4.1.1 Pengujian <i>Tuning</i> PID dengan Metode <i>Trial and Error</i>	46
4.4.1.2 Pengujian Parameter KP	46

4.4.1.3 Pengujian Parameter KI	50
4.4.1.4 Pengujian Parameter KD.....	55
4.4.1.5 Pengujian Validasi Parameter Keseluruhan.....	59
4.4.2 Pengujian terhadap Setpoint dengan <i>PID Controller</i>	60
Bab 5. Kesimpulan dan Saran	65
5.1. Kesimpulan.....	65
5.2. Saran	66
Daftar Pustaka.....	68
Biodata	71
LAMPIRAN	72

Daftar Gambar

Gambar 1. Konveyor.....	6
Gambar 2. Motor DC.....	6
Gambar 3. Persamaan PID.....	7
Gambar 4. ESP32.....	9
Gambar 5. ESP-NOW.....	9
Gambar 6. C#.....	10
Gambar 7. XAMPP.....	10
Gambar 8. Motor Driver L298N.....	11
Gambar 9. Sensor ACS71.....	11
Gambar 10. Sensor Tegangan.....	12
Gambar 11. Alur Penelitian.....	18
Gambar 12. Diagram Blok Sistem.....	19
Gambar 13. Desain Sistem PID.....	20
Gambar 14. Desain Sistem Voice.....	20
Gambar 15. Flowchart Perancangan Elektrikal.....	21
Gambar 16. Perancangan Desain Elektrikal.....	22
Gambar 17. Flowchart Perancangan Mekanikal.....	24
Gambar 18. Perancangan Desain Mekanikal.....	25
Gambar 19. Perancangan Interface.....	26
Gambar 20. Desain Menu Login.....	27
Gambar 21. Desain Menu Kontrol.....	28
Gambar 22. Desain Menu Monitoring.....	28
Gambar 23. Desain Menu Management.....	29
Gambar 24. Hasil Pengujian Mekanik.....	33
Gambar 25. Hasil Pengujian Elektrikal.....	35
Gambar 26. Hasil Pengujian Menu Login.....	38
Gambar 27. Hasil Pengujian Menu Monitoring.....	40
Gambar 28. Hasil Pengujian Menu Management User.....	41
Gambar 29. Blok Diagram Pengujian PID.....	42
Gambar 30. Grafik Respon Setpoint 100 Tanpa PID.....	43
Gambar 31. Grafik Respon Setpoint 200 Tanpa PID.....	44
Gambar 32. Grafik Respon Setpoint 400 Tanpa PID.....	44
Gambar 33. Grafik Respon Setpoint 600 Tanpa PID.....	45
Gambar 34. Pengujian Parameter KP 0.15.....	47
Gambar 35. Pengujian Parameter KP 0.20.....	47
Gambar 36. Pengujian Parameter KP 0.26.....	48
Gambar 37. Grafik Respon Pengujian Kp Bervariasi.....	49
Gambar 38. Pengujian Parameter KI 0.005.....	51
Gambar 39. Pengujian Parameter KI 0.007.....	52

Gambar 40. Pengujian Parameter KI 0.009	53
Gambar 41. Grafik Respon Pengujian Ki Bervariasi	54
Gambar 42. Pengujian Parameter KD 0.005	55
Gambar 43. Pengujian Parameter KD 0.008	56
Gambar 44. Pengujian Parameter KD 0.0011	57
Gambar 45. Grafik Respon Pengujian KD Bervariasi	58
Gambar 46. Grafik Respon Pengujian Validasi PID	59
Gambar 47. Grafik Respon Setpoint 100 dengan PID	60
Gambar 48. Grafik Respon <i>Setpoint</i> 200 dengan PID	61
Gambar 49. Grafik Respon <i>Setpoint</i> 400 dengan PID	61
Gambar 50. Grafik Respon <i>Setpoint</i> 600 dengan PID	62

Daftar Tabel

Tabel 1. Work Breakdown Structure	4
Tabel 2. Tabel Referensi Penulis	12
Tabel 3. Penggunaan Komponen Elektrikal	23
Tabel 4. Dimensi Desain Mekanikal	25
Tabel 5. Estimasi Alat dan Bahan	29
Tabel 6. Hasil Pengujian Mekanikal	34
Tabel 7. Hasil Pembacaan Sensor	36
Tabel 8. Hasil Pembacaan Alat ukur	36
Tabel 9. Hasil Pengujian Perhitungan Error Sistem	37
Tabel 10. Hasil Pengujian Jarak Konektivitas	39
Tabel 11. Hasil Akurasi Berdasarkan Jarak	39
Tabel 12. Hasil Pengujian Parameter KP Bervariasi	49
Tabel 13. Hasil Pengujian Parameter KI Bervariasi	53
Tabel 14. Hasil Pengujian Parameter KD Bervariasi	57
Tabel 15. Hasil Pengujian PID Terhadap Sistem	63
Tabel 16. Hasil Perbandingan Pengujian tanpa Kontrol PID dan dengan Kontrol PID	64

Bab 1. Pendahuluan

1.1. Latar Belakang

Dalam era Revolusi Industri 4.0, otomatisasi telah menjadi elemen penting dalam berbagai sektor, seperti otomotif, manufaktur, dan pertambangan. Teknologi ini mampu meningkatkan efisiensi, produktivitas, dan kualitas hasil produksi, sekaligus mengurangi biaya operasional[1]. Namun, meskipun potensi ini sangat besar, masih terdapat banyak tantangan dalam implementasi otomatisasi secara menyeluruh, terutama pada sistem yang masih bergantung pada metode manual. Salah satu contoh yang sering dihadapi adalah sistem pengendalian konveyor, di mana metode manual masih banyak digunakan, sehingga menimbulkan ketidakefisienan dan keterbatasan aksesibilitas, terutama bagi operator dengan kebutuhan khusus.

Konveyor yang banyak digunakan di lapangan umumnya menggunakan metode manual yang masih menerapkan sistem kendali open-loop, sehingga beban kerja yang berubah-ubah dapat secara signifikan memengaruhi kecepatan putaran motor (RPM). Hal ini menyebabkan sistem kurang efisien, sulit dioptimalkan, dan tidak mampu memberikan kinerja yang konsisten sesuai kebutuhan produksi[2].

Salah satu dampak dari sistem kendali manual ini adalah ketidakefisienan, ketika konveyor digunakan untuk mengangkut benda produksi yang sensitif, seperti proses assembly komponen elektronik. Lonjakan RPM yang tidak terkontrol dapat menyebabkan benda produksi terjatuh dari konveyor atau bahkan saling bertabrakan secara beruntun, yang tidak hanya merusak komponen tetapi juga menghentikan proses produksi sementara.

Keterbatasan aksesibilitas juga menjadi kendala, terutama bagi operator dengan disabilitas yang menghadapi tantangan untuk berinteraksi langsung dengan mesin produksi[3]. Dalam sistem kendali manual, operator sering kali diharuskan melakukan tindakan fisik, seperti menekan tombol pada panel kontrol, memutar tuas, atau bahkan berjalan menuju lokasi tertentu untuk melakukan penyesuaian. Bagi operator dengan keterbatasan mobilitas atau disabilitas lainnya, hal ini menjadi hambatan yang tidak hanya mengurangi efisiensi kerja tetapi juga menurunkan tingkat partisipasi mereka dalam operasional produksi. Seperti seorang operator dengan keterbatasan gerak mungkin memerlukan waktu lebih lama untuk mencapai panel kontrol atau kesulitan dalam mengoperasikan perangkat yang dirancang tanpa mempertimbangkan kebutuhan khusus mereka.

Dalam mengatasi permasalahan tersebut yaitu dengan menerapkan kontrol PID (*Proportional-Integral-Derivative*) pada sistem kendali konveyor dapat mengatasi permasalahan seperti lonjakan RPM akibat perubahan beban kerja, sehingga menjaga kecepatan motor tetap stabil dan mencegah kerusakan benda

produksi. Selain itu, pengendalian berbasis suara memudahkan interaksi manusia-mesin, sehingga operator dapat mengontrol konveyor tanpa perlu kontak fisik langsung. Ini sangat membantu operator dengan disabilitas, yang mungkin kesulitan dengan tugas fisik.

Berdasarkan studi lapangan dan pengalaman penulis dalam proyek di industri manufaktur di Kota Batam, mayoritas pengoperasian konveyor masih dilakukan secara manual[4][5]. Dalam mengatur kecepatan konveyor, operator sering kali harus melakukan tindakan fisik, seperti berjalan menuju panel kontrol untuk melakukan penyesuaian. Selain itu, operator juga sering diharuskan untuk melakukan dua pekerjaan sekaligus, seperti memantau barang cacat dan mengambil hasil produksi pada mesin yang lokasinya cukup berjauhan. Kondisi ini tidak hanya mengurangi efisiensi kerja, tetapi juga menyebabkan *downtime* dalam proses produksi, yang berdampak negatif pada produktivitas secara keseluruhan.

Penelitian ini didasarkan pada sejumlah studi terdahulu yang relevan. Pertama, penelitian oleh Muhammad Reza Aditya Nurkholis Putera dan Rahmat Hidayat (2022) menggunakan metode kontrol PID untuk meningkatkan respons sistem kontrol. Hasil penelitian menunjukkan bahwa kontrol PID dapat menjaga stabilitas sistem dalam menghadapi perubahan beban[6]. Kedua, penelitian oleh Radi Birdayansyah et al. (2015) mengembangkan sistem kendali menggunakan perintah suara untuk mengatur kecepatan motor DC. Sistem ini menunjukkan bahwa teknologi perintah suara dapat meningkatkan efisiensi dan aksesibilitas operasional[7]. Ketiga, penelitian oleh M. Asyroful Ulum dan Subuh Isnur Haryudo (2020) menunjukkan bahwa pemantauan jarak jauh melalui sensor memungkinkan deteksi dini terhadap potensi kerusakan pada mesin[8].

Penelitian ini bertujuan untuk mengembangkan sistem kontrol konveyor berbasis PID yang dikombinasikan dengan perintah suara. Sistem ini diharapkan dapat memberikan solusi yang lebih efisien, meningkatkan aksesibilitas, serta memungkinkan deteksi dini terhadap gangguan pada konveyor. Dengan memanfaatkan kontrol PID, sistem mampu menjaga stabilitas kinerja konveyor meskipun terjadi perubahan beban. Integrasi perintah suara memudahkan operasional bagi operator, terutama yang memiliki keterbatasan fisik.

1.2. Rumusan Masalah

1. Bagaimana mengimplementasikan kontrol PID untuk mendapatkan respon yang akurat dan stabil pada sistem kontrol konveyor?
2. Bagaimana mengintegrasikan perintah suara sebagai input untuk mengontrol konveyor?
3. Bagaimana mengimplementasikan visual C# sebagai interface untuk memonitoring konveyor?

1.3. Tujuan

1. Menganalisis sistem kontrol konveyor menggunakan metode kendali PID (Proportional-Integral-Derivative) untuk mengatur kecepatan motor DC pada konveyor secara stabil.
2. Mengimplementasikan teknologi *Voice Recognition* untuk memudahkan interaksi antara manusia dan mesin dalam mengendalikan operasi konveyor secara fleksibel.
3. Merancang dan membangun sistem pemantauan berbasis *Visual C#* untuk memantau kinerja konveyor secara *real-time*, sehingga memungkinkan deteksi dini terhadap kondisi abnormal dan penerapan tindakan preventif yang diperlukan untuk menjaga kelancaran operasional.

1.4. Manfaat

Manfaat dari penelitian ini adalah sebagai berikut:

1. Memberikan kinerja konveyor yang akurat dan stabil.
2. Memudahkan *User* dalam pengendalian konveyor secara fleksibel dan cepat menggunakan perintah suara.
3. Membantu *User* agar dapat memantau keadaan konveyor secara *realtime*

1.5. Batasan

Adapun beberapa batasan dari penelitian ini adalah sebagai berikut:

1. Konveyor yang digunakan adalah konveyor dengan skala kecil (*mini conveyor*).
2. Pembahasan lebih berfokus pada sistem kontrol dan monitoring.
3. Keterbatasan lingkungan, yang mana alat yang dirancang hanya difokuskan untuk lingkungan yang memiliki tingkat kebisingan menengah ke bawah.
4. Fitur yang di rancang adalah perintah suara dan bukan pengenalan suara secara biometric.

1.6. *Work Breakdown Structure* (Opsional)

Tabel 1. *Work Breakdown Structure*

No	Nama	Tugas dan Tanggung Jawab dalam Tim
1	Wahyu Firdaus	Programmer dan Elektikal
2	Muhammad Zidni Aufa	Rancang Sistem Kendali dan Mekanikal
3	Wahyu Firdaus	Pengujian dan Analisis
4	Muhammad Zidni Aufa	Pengambilan data dan Analisis

Bab 2. Tinjauan Pustaka

2.1 Penelitian Terdahulu

Dalam penelitian ini kami mengambil beberapa referensi dari para peneliti sebelumnya, kegiatan ini dilakukan sebagai bentuk perbandingan mengenai kelebihan atau kekurangan dari penelitian-penelitian sebelumnya. Selain itu kami juga mencari informasi lain dari paper, jurnal, maupun situs website yang berkaitan dengan teori-teori yang akan kami gunakan untuk penelitian ini.

Sistem kontrol konveyor sebelumnya sudah pernah dibuat dan digunakan namun masih dalam segi pengaplikasian yang terpisah, berikut adalah beberapa penelitian yang berkaitan dengan sistem kontrol konveyor.

Penelitian pertama adalah penelitian yang dilakukan oleh Radi Birdayansyah, dkk (2015), Penelitian yang berjudul “Pengendalian Kecepatan Motor DC Menggunakan Perintah Suara Berbasis Mikrokontroler Arduino” [17] pada penelitian ini ditemukan beberapa informasi terkait dengan rumus-rumus yang berkaitan dengan teori yang akan kami gunakan seperti rumus menghitung *PWM*, *RPM*, dan *Duty Cycle*. Namun dalam penelitian ini masih menggunakan modul rakitan sehingga membutuhkan waktu riset yang lama dalam pembuatan modul tersebut, selain itu pemilihan komponen yang terlalu banyak akan menyebabkan kesulitan dalam proses *maintenance*. [19]

Penelitian kedua adalah penelitian yang dilakukan oleh M. Asyroful Ulum dan Subuh Isnur Haryudo (2020), Penelitian yang berjudul “perancangan sistem monitoring kecepatan putar motor dc berbasis internet of things menggunakan aplikasi blynk”. Pada penelitian ini kami mendapatkan beberapa referensi terkait apa saja sensor yang dapat digunakan untuk monitoring motor DC, tingkat akurasi pembacaan sensor, serta pengaruh beban terhadap arus dan tegangan pada motor DC.

Penelitian ketiga adalah penelitian yang dilakukan oleh Mila Diah Ika Putri, dkk (2021), Penelitian yang berjudul “Pengendali Kecepatan Sudut Motor DC Menggunakan Kontrol PID dan Tuning Ziegler Nichols” [20]. Pada penelitian ini kami mengetahui dan mengambil referensi terkait mengendalikan Motor DC menggunakan pengendali PID dengan metode Ziegler Nichols. Serta mengetahui cara mencari nilai Parameter K_p , K_i , dan K_d .

2.2 Konveyor

Konveyor atau mesin konveyor merupakan peralatan sederhana yang dapat bergerak dari satu tempat ke tempat lain sebagai alat angkut untuk berbagai jenis barang, mulai dari kapasitas kecil hingga besar. Konveyor dijadikan sebagai alat transportasi yang cepat dan efisien. Terdapat beberapa jenis konveyor, seperti roller conveyor, belt conveyor, dan lain sebagainya. Gambaran umum mengenai sistem konveyor ini dapat dilihat pada **Error! Reference source not found.**, yang menunjukkan struktur dasar serta komponen utama dari konveyor. Hal ini memberikan pemahaman lebih jelas mengenai mekanisme kerja dan variasi jenis konveyor [11].



Gambar 1. Konveyor

2.3 Motor DC

Motor Listrik DC atau DC Motor adalah suatu perangkat yang mengubah energi listrik menjadi energi kinetik atau Gerakan seperti **Error! Reference source not found.** Motor DC ini juga dapat disebut sebagai Motor Arus Searah. Seperti namanya, DC Motor memiliki dua terminal dan memerlukan tegangan arus searah atau DC (*Direct Current*) untuk dapat menggerakannya[16]



Gambar 2. Motor DC

2.4 PID (*Proportional, Integral, Derivative*)

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

Gambar 3. Persamaan PID

Berdasarkan Gambar 3, Nilai kontrol $u(t)$ yang dihitung oleh kontroler PID didasarkan pada K_p adalah koefisien proporsional, T_i adalah integral waktu konstan dan T_d adalah diferensian time konstan.

Keterangan :

$e(t)$: sinyal error

$u(t)$: output controller

K_p : konstanta proporsional

K_i : konstanta integral

K_d : konstanta derivative

Dalam penelitian ini, metode *trial and error* digunakan untuk menentukan parameter PID yang optimal, yaitu K_p , K_i , dan K_d . Proses ini dimulai dengan menetapkan nilai awal berdasarkan pendekatan empiris atau referensi sebelumnya. Selanjutnya, sistem diuji dengan parameter tersebut, dan responsnya diamati, seperti waktu naik (*rise time*), waktu pemulihan (*settling time*), osilasi, serta kesalahan steady-state. Jika sistem menunjukkan osilasi yang berlebihan atau tidak stabil, nilai K_p dikurangi. Jika terdapat kesalahan steady-state yang besar, nilai K_i ditingkatkan untuk menguranginya. Selain itu, jika respons sistem terlalu lambat atau mengalami overshoot yang signifikan, nilai K_d disesuaikan untuk mengontrol perubahan sinyal error. Proses ini dilakukan secara berulang hingga diperoleh kombinasi parameter yang menghasilkan performa sistem yang stabil dan sesuai dengan kriteria yang diinginkan [12]. Metode *trial and error* dipilih karena memberikan fleksibilitas dalam menyesuaikan parameter PID secara langsung berdasarkan karakteristik sistem yang diuji. Meskipun membutuhkan waktu dan pengujian berulang, metode ini efektif dalam mendapatkan tuning parameter yang optimal tanpa memerlukan model matematis yang kompleks.

2.5 Speech Recognition

Speech recognition merupakan kemampuan mesin untuk mendengarkan kata-kata yang diucapkan dan mengidentifikasinya. Kemampuan untuk dapat mengubah suara yang masuk pada komputer ke dalam bentuk teks. *Speech Recognition* menggabungkan ilmu komputer dan linguistik untuk

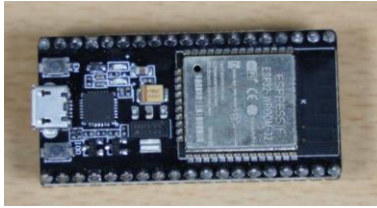
mengidentifikasi kata-kata yang diucapkan dan mengubahnya menjadi teks. Hal ini memungkinkan komputer untuk memahami bahasa manusia[21]. Teknologi speech recognition memungkinkan suatu perangkat untuk mengenali dan memahami kata – kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan pada suatu perangkat.

2.6 Monitoring

Monitoring adalah proses pengumpulan dan analisis informasi berdasarkan indikator yang ditetapkan secara sistematis dan kontinu tentang kegiatan program sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program kegiatan itu selanjutnya. Monitoring adalah pemantauan yang dapat dijelaskan sebagai kesadaran (awareness) tentang apa yang ingin diketahui[22], pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan ke arah tujuan atau menjauh dari itu. Monitoring akan memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu ke waktu, pemantauan umumnya dilakukan untuk tujuan tertentu, untuk memeriksa terhadap proses berikut objek atau untuk mengevaluasi kondisi atau kemajuan menuju tujuan hasil manajemen atas efek tindakan dari beberapa jenis antara lain tindakan untuk mempertahankan manajemen yang sedang berjalan.

2.8 ESP32

ESP32 pada Gambar 4, merupakan mikrokontroler SoC (System on Chip) terpadu dengan WiFi 802.11 b/g/n, Bluetooth versi 4.2, dan berbagai periferal[15]. Chip ini menggunakan mikroprosesor 32 bit Xtensa LX6 dual-core. Ruang alamat untuk data dan instruksi adalah 4 GB dan ruang alamat periferal 512 kB. Memori terdiri atas 448 kB ROM, 520 kB SRAM, dua 8kB RTC memory, dan flash memory 4MB. Chip ini mempunyai 18 pin ADC (12-bit), empat SPI, dan dua I2C[16]. Kelebihan utama mikrokontroler ini ialah harganya yang relatif murah, mudah diprogram, memiliki jumlah pin I/O yang memadai, serta memiliki adapter WiFi internal untuk mengakses jaringan Internet. Board ini memiliki dua versi, yaitu 30 GPIO dan 36 GPIO. Keduanya memiliki fungsi yang sama tetapi versi yang 30 GPIO dipilih karena memiliki dua pin GND. Semua pin diberi label di bagian atas board sehingga mudah untuk dikenali. Board ini memiliki interface USB to UART yang mudah diprogram dengan program pengembangan aplikasi seperti Arduino IDE. Sumber daya board bisa diberikan melalui konektor micro USB.



Gambar 4. ESP32

2.9 ESP-NOW

ESP-NOW adalah sebuah protokol komunikasi yang dikembangkan oleh Espressif eksklusif untuk perangkat ESP saja. Komunikasi dilakukan tanpa harus melakukan prosedur yang selayaknya dilakukan dalam penyambungan perangkat WiFi[17]. Protokol ini lebih mirip dengan protokol komunikasi nirkabel 2.4Ghz seperti Gambar 5 yang umumnya digunakan oleh perangkat komputer mouse, yakni setelah dilakukan pairing antar perangkat, koneksi sudah diasumsikan sebagai aman, tanpa perlu melakukan *handshake*[16].



Gambar 5. ESP-NOW

2.10 Database

Database adalah kumpulan data atau informasi yang kompleks, data-data tersebut disusun menjadi beberapa kelompok dengan tipe data yang sejenis disebut *table/entity*), di mana setiap datanya dapat saling berhubungan satu sama lain atau dapat berdiri sendiri, sehingga mudah diakses.

2.11 C#

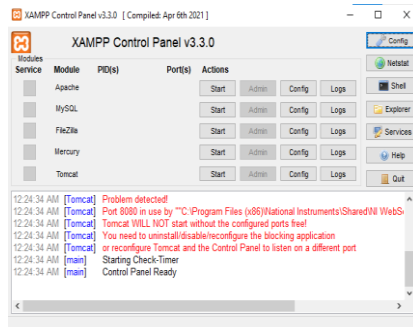
Visual C# merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka *.NET Framework*[15]. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti *Java*, *Delphi*, *Visual Basic*, dan lain-lain dengan beberapa penyederhanaan.



Gambar 6. C#

2.12 XAMPP

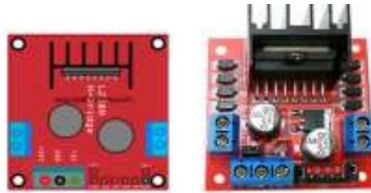
XAMPP adalah sebuah software web server apache yang didalamnya sudah tersedia database server MySQL dan dapat mendukung pemrograman PHP. XAMPP merupakan software yang mudah digunakan, gratis dan mendukung instalasi di Linux dan Windows seperti pada Gambar 7. Keuntungan lainnya adalah cuma menginstal satu kali sudah tersedia Apache Web Server, MySQL Database Server, PHP Support (PHP 4 dan PHP 5) dan beberapa modul lainnya[9].



Gambar 7. XAMPP

2.13 Motor Driver

Driver motor L298N pada Gambar 8, merupakan modul driver motor DC yang digunakan untuk mengontrol kecepatan dan arah perputaran motor DC. Modul ini paling banyak digunakan dalam dunia elektronika dan sering dihubungkan ke mikrokontroler Arduino. IC L298N merupakan sebuah IC tipe H-bridge yang mampu mengendalikan beban induktif pada kumparan seperti solenoid, relay, motor DC dan motor stepper[19]. Motor listrik terdiri dari lilitan kumparan sehingga memiliki beban induktif yang sangat besar. Pada IC L298N terdapat transistor transistor logic (TTL) dengan gerbang NAND yang berfungsi untuk mengubah arah putaran motor suatu motor dc maupun motor stepper.



Gambar 8. Motor Driver L298N

2.14 Sensor ACS712

Sensor arus ACS712 atau Hall Effect Current Sensor yang berfungsi mendeteksi aliran listrik yang melewatinya *Hall effect allegro ACS712* merupakan sensor yang presisi sebagai sensor arus AC atau DC dalam pembacaan arus di dalam dunia industri, otomotif, komersil dan sistem - sistem komunikasi. Pada umumnya aplikasi sensor ini biasa digunakan untuk mengontrol motor, deteksi beban listrik, switched - mode power supplies dan proteksi beban berlebih. Sensor ini dipasang seri dengan beban yang akan diukur.



Gambar 9. Sensor ACS71

2.15 Sensor Tegangan

Sensor tegangan berfungsi membaca nilai tegangan suatu rangkaian. Arduino dapat membaca nilai tegangan dengan memanfaatkan pin analog. Jika range tegangan yang dibaca diantara 0-5 V bisa langsung menggunakan pin analog, sedangkan jika range tegangan yang dibaca >5V harus menggunakan rangkaian tambahan yakni pembagi tegangan karena pin arduino bekerja pada maksimal 5V.



Gambar 10. Sensor Tegangan

Jurnal Referensi sebagai berikut :

Tabel 2. Tabel Referensi Penulis

N o	Tahun	Penulis	Jurnal	Metode	Kelemahan
1	2023	Taufiqur Rohman, Widi Aribowo, Ayusta Lukita Wardani, Reza Rahmadian	Perancangan Sistem Pengendali Kecepatan Motor DC Menggunakan Kontroler Proportional Integral Derivative Pada Palang Pintu Parkir	PID	Jurnal tersebut tidak memberikan informasi yang cukup detail mengenai hasil pengujian yang dilakukan, seperti data numerik yang mendukung kesimpulan yang diambil dari penelitian tersebut.

2	2023	Demas Chandra Permana, Radinald Ferdiansyah, 3Fayza Putri Safira, Zayid Thoriq Ali Gumilang, Ariya Jembar Pangestu, Rama Wijawa Abdul Rozak	OTOMASI INDUSTRI SEBUAH PELUANG ATAU ANCAMAN	Pendekatan Kualitatif	Jurnal tersebut tidak membahas mengenai solusi konkret untuk mengatasi dampak negatif dari adopsi teknologi otomatisasi.
3	2022	Baroroh Dhiya' Ushofa, Lilik Anifah, I Gusti Putu Asto Buditjahjanto, Endryansyah	Sistem Kendali Kecepatan Putaran Motor DC pada Conveyor dengan Metode Kontrol PID	PID	Jurnal tersebut kurang menjelaskan mengenai metode penelitian yang digunakan dan minimnya analisis terhadap faktor-faktor yang mempengaruhi hasil penelitian
4	2023	Tole Sutikno, Jekson	Monitoring Tegangan dan	Kuantitatif	Jurnal tersebut

		Alfahri , Hendril Satrian Purnama	Arus Pada Panel Surya Menggunakan IoT		kurang menjelaskan mengenai metode eksperimen yang digunakan serta tidak disebutkan secara detail mengenai analisis data yang dilakukan. Selain itu, kurang menjelaskan mengenai validitas dan reliabilitas data yang digunakan dalam penelitian.
5	2020	Muhamad M.I. Putra, Sherwin R. U. A. Sompie, Sary Paturusi	Implementasi Speech Recognition pada Aplikasi Pembelajaran Bahasa Inggris untuk Anak	Multimedia Development Life Cycle (MDLC)	Jurnal tersebut adalah tidak disebutkan secara jelas mengenai metode pengumpulan data yang digunakan

					dalam penelitian.
6	2015	Radi Birdayansyah, Noer Sudjarwanto, Osea Zebua	Pengendalian Kecepatan Motor DC Menggunakan Perintah Suara Berbasis Mikrokontroler Arduino	Pendekatan Kualitatif	Jurnal tersebut tidak membahas kecepatan pengenalan suara dalam penelitian ini. Penggunaan suara yang kompleks atau waktu pemrosesan yang lama dapat menghasilkan keterlambatan dalam pengendalian kecepatan motor DC.
7	2022	Muhammad Reza Aditya Nurkholis Putera, Rahmat Hidayat	KENDALI KECEPATAN MOTOR DC MENGGUNAKAN PENGENDALIAN PID DENGAN ENCODER SEBAGAI FEEDBACK	PID	Jurnal tersebut kurang membahas mengenai metode pengujian yang digunakan serta kurangnya

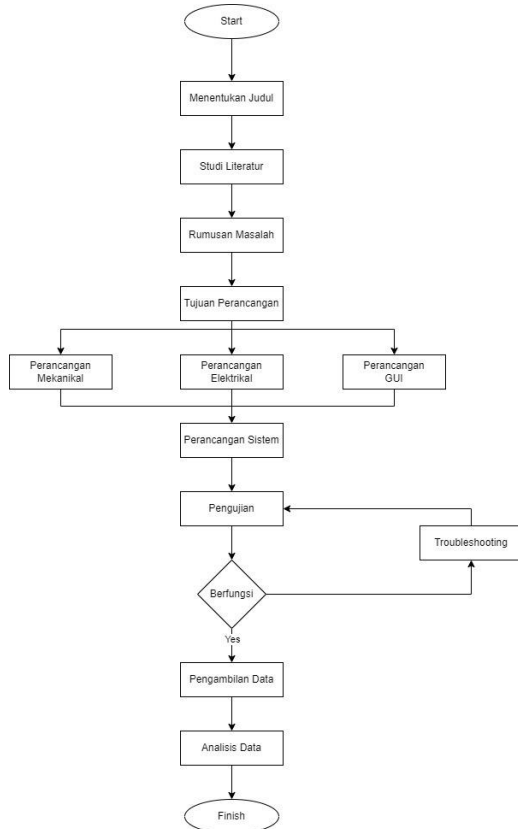
					analisis terhadap faktor-faktor eksternal yang dapat memengaruhi hasil penelitian.
8	2022	Fahrizal Adnan , Ilya Amelia , Sayyid 'Umar Shiddiq	Implementasi Voice Recognition Berbasis Machine Learning	Kuantitatif	Jurnal tersebut kurang menjelaskan mengenai ukuran sampel yang digunakan dalam penelitian.
9	2013	WALUYO, ADITYA FITRIANSYA H, SYAHRIAL	Analisis Penalaan Kontrol PID pada Simulasi Kendali Kecepatan Putaran Motor DC Berbeban menggunakan Metode Heuristik	PID	Jurnal tersebut kurang melakukan analisis statistik yang mendalam, serta referensi belum banyak.

10	2020	M. Asyroful Ulum, Subuh Isnur Haryudo	PERANCANGAN SISTEM MONITORING KECEPATAN PUTAR MOTOR DC BERBASIS INTERNET OF THINGS MENGGUNAKAN APLIKASI BLYNK	Pendekatan Kuantitatif	Jurnal tersebut tidak disebutkan metode pengujian yang digunakan untuk menghitung error sistem secara detail lalu juga tidak membahas informasi mengenai lingkup atau batasan penelitian yang dilakukan.
----	------	--	---	------------------------	--

Bab 3. Metodologi Penelitian/Metode Pelaksanaan

3.1 Alur Penelitian

Metodologi penelitian bertujuan untuk menyelesaikan masalah yang ada secara terstruktur. Berikut pembagian dari flow chart penelitian ditunjukkan dalam Gambar 11 dibawah ini.



Gambar 11. Alur Penelitian

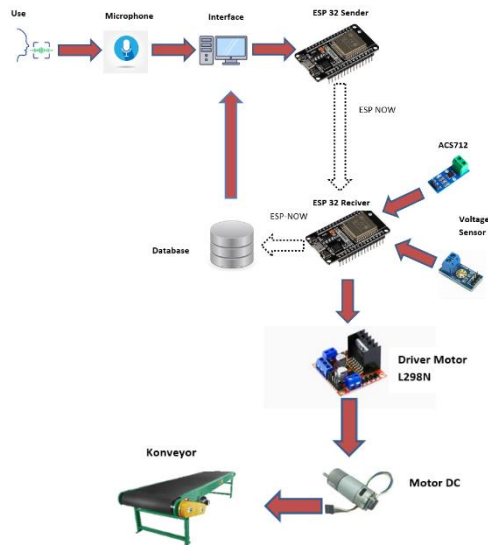
Berdasarkan Gambar 11, Proses dimulai dengan menentukan judul yang relevan dengan tugas akhir yang akan dikerjakan. Setelah judul ditetapkan,

langkah berikutnya adalah mencari referensi dari penelitian sebelumnya yang berkaitan dengan topik yang dipilih. Setelah memperoleh referensi, dilakukan identifikasi rumusan masalah yang sesuai dengan judul tersebut. Selanjutnya, ditetapkan tujuan perancangan sebagai dasar dalam pengembangan alat. Setelah tujuan perancangan ditentukan, dilakukan perancangan sistem yang mencakup aspek elektrikal, mekanikal, desain GUI, serta sistem keseluruhan. Setelah seluruh sistem dirancang dan digabungkan, tahap berikutnya adalah pengujian alat. Jika alat tidak berfungsi sebagaimana mestinya, dilakukan troubleshooting untuk mengatasi permasalahan yang muncul. Namun, jika alat berfungsi sesuai dengan mekanisme yang dirancang, maka proses dilanjutkan dengan pengambilan data. Setelah seluruh data pengujian diperoleh, langkah terakhir adalah melakukan analisis terhadap data yang telah dikumpulkan.

3.1.1 Perancangan Sistem

Dalam Penelitian kami melakukan rancangan sistem menjadi dua aspek, yang pertama adalah Diagram Blok Sistem dan yang kedua adalah Desain Sistem, Adapun penjelasannya sebagai berikut :

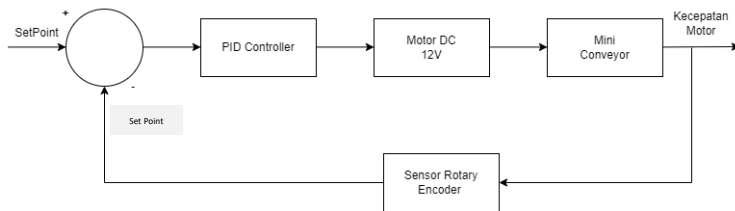
Diagram Blok Sistem



Gambar 12. Diagram Blok Sistem

Berdasarkan Gambar 12, untuk perancangan *hardware* di mulai dengan membuka aplikasi, setelah itu menginputkan perintah menggunakan *voice recognition*, selanjutnya data akan di teruskan oleh *esp sender* ke *esp receiver*, pada *esp receiver* terdapat beberapa komponen yang terhubung yaitu sensor *power supply* dan motor driver, yang mana hasil pembacaan sensor akan di kirim ke database dan motor driver untuk mengatur kecepatan pada motor dc.

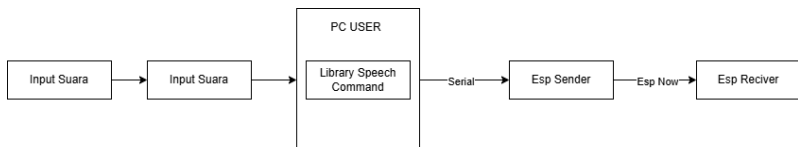
Desain Sistem PID



Gambar 13. Desain Sistem PID

Pada Gambar 13 diatas, merupakan Desain Sistem kendali pada proses pengendalian kecepatan Motor DC. Nilai masukan (*Setpoint*) sistem berupa kecepatan. Kontroler PID akan diproses pada ESP32 yang nantinya akan mengatur kecepatan pada motor. Hasil dari keluaran *Controller* berupa tegangan yang akan masuk ke motor DC 12V melalui driver motor L298N. Sensor *rotary encoder* akan bekerja untuk mengukur kecepatan motor DC kemudian menjadikan nilai pembacaan sebagai nilai umpan balik pada keluaran yang dihasilkan berupa kecepatan.

Desain Sistem Voice

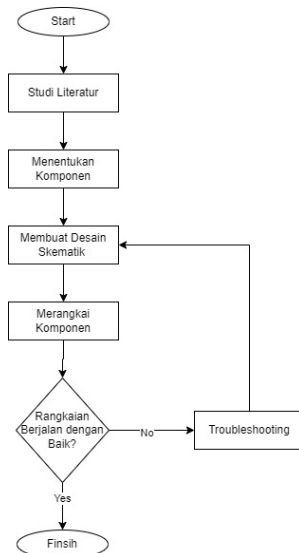


Gambar 14. Desain Sistem Voice

Berdasarkan Gambar 14, Pertama input suara diberikan melalui mikrofon yang mana perintah tersebut terdiri dari On, Off, Low, Mid, High. Suara yang diinput tersebut kemudian diproses oleh antarmuka (interface) yang dikembangkan menggunakan Visual Studio dengan memanfaatkan library yang di sediakan oleh visual studio (System.Speech) . Setelah suara dikenali dan diolah menjadi data perintah, data ini dikirimkan ke modul ESP32 sebagai pengirim (ESP sender). Modul ESP sender berfungsi untuk meneruskan data tersebut ke modul ESP penerima (ESP receiver) menggunakan protokol komunikasi ESP-NOW. Protokol ini memungkinkan pengiriman data secara langsung dan efisien antara perangkat ESP32 tanpa memerlukan jaringan Wi-Fi eksternal. Selanjutnya, data perintah yang diterima oleh ESP receiver digunakan untuk menginputkan parameter set point pada kontrol motor dc

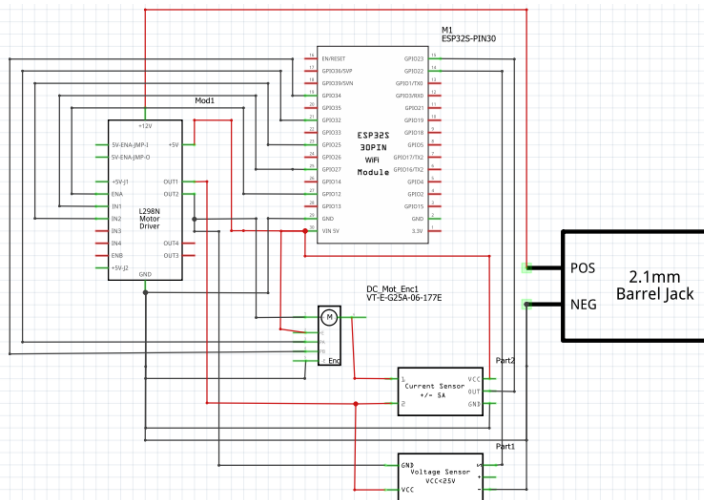
3.1.2 Perancangan Elektrikal

Berikut ini adalah *flowchart* perancangan elektrikal yang akan peneliti lakukan agar dalam pengerjaan aktualnya lebih terstruktur, Adapun beberapa tahapan dan penjelasannya dapat dilihat pada Gambar 15 dibawah ini.



Gambar 15. Flowchart Perancangan Elektrikal

Berdasarkan Gambar 15, pada tahap perancangan elektrikal dimulai dengan melakukan studi literatur terlebih dahulu. langkah selanjutnya yaitu dengan menentukan komponen apa yang akan digunakan. setelah itu dilanjutkan dengan membuat desain skematik, hal ini bertujuan untuk memudahkan dalam proses perakitan tahap selanjutnya yaitu merangkai komponen. setelah semua komponen selesai dirangkai maka akan dilakukan pengecekan terlebih dahulu, apakah rangkaian berjalan dengan baik atau tidak, jika tidak maka akan dilakukan troubleshooting dan proses akan diulang kembali ke langkah membuat desain Berikut merupakan rancangan desain elektrikalnya



Gambar 16. Perancangan Desain Elektrikal

Berdasarkan Gambar 16, Rangkaian ini menggunakan adaptor 220VAC to 12 VDC sebagai sumber tegangannya, yang mana positif adaptor terhubung ke pin +12 dan negatif adaptor ke GND pada driver motor L298N. Kemudian output +5V pada driver motor di hubungkan ke pin Vin pada esp 32 dan juga terhubung ke Encoder (+), voltage sensor dan ACS712. Selanjutnya pin ENB Driver Motor terhubung ke GPIO12, IN3 terhubung ke pin GPIO 27 dan IN4 terhubung ke pin GPIO14 pada esp 32. Output dari L298N terhubung ke wire out pada ACS712 dan vcc pada Voltage sensor sedangkan pin Out4 terhubung ke GND voltage sensor dan Encoder (-). Selanjutnya pin data (S) pada voltage sensor terhubung ke GPIO32 pada esp 32 sedangkan pin OUT pada ACS712 terhubung ke GPIO34 pada esp 32. selanjutnya pin Encoder A Phase pada motor dc terhubung ke GPIO26 dan Encoder B Phase terhubung ke GPIO25 pada esp32.

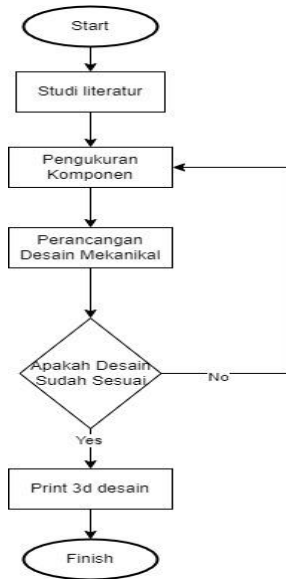
Adapun alasan peneliti menggunakan komponen – komponen diatas dapat dilihat pada Tabel 3 dibawah ini.

Tabel 3. Penggunaan Komponen Elektrikal

Komponen	Alasan Penggunaan
Driver Motor L298N	Driver motor L298N ini mudah diintegrasikan dengan mikrokontroler seperti Arduino, dengan konektor terminal yang memudahkan penyambungan. Dukungan library, dokumentasi, serta adanya LED indikator membuat penggunaan dan troubleshooting lebih mudah, sehingga menjadikannya pilihan populer untuk proyek elektronika dan semacamnya.
ESP 32	Karena esp32 memiliki protokol khusus dalam mengirim data yaitu ESP NOW yang mana kami menggunakan protokol ini untuk mengirim data dari esp sender ke esp receiver.
ACS712	Memiliki akurasi yang cukup baik, Kemudahan dalam pengaplikasian dan rentang pembacaan arus yang cukup bervariasi sampai dengan +- 30A.
Voltage Sensor	Karena untuk pemantauan real-time, proteksi perangkat dari tegangan berlebih atau kurang, efisiensi energi, dan pengendalian otomatis. Sensor ini meningkatkan keamanan dengan mendeteksi kondisi tegangan abnormal dan mempermudah diagnostik serta pemeliharaan sistem elektronik, memastikan performa optimal dan mencegah kerusakan.
Motor With Encoder	Lebih fleksibel karena motor dan encoder dapat di implementasikan secara bersamaan.
Adaptor 220V AC to 12V DC	Menyesuaikan dengan tegangan input dari motor driver yaitu 12V.

3.1.3 Perancangan Mekanikal

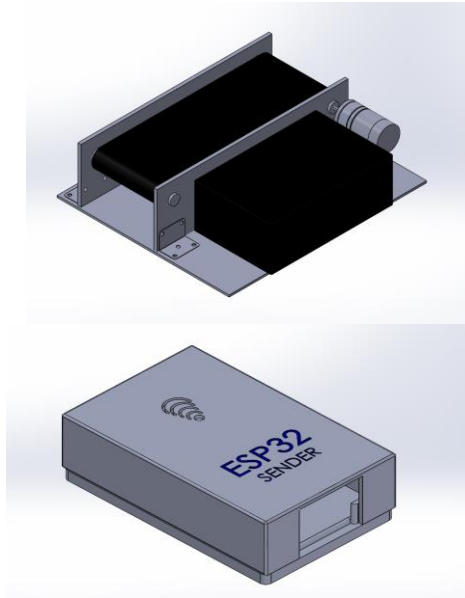
Berikut ini adalah *flowchart* perancangan mekanikal yang akan peneliti lakukan agar dalam pengerjaan aktualnya lebih terstruktur, Adapun beberapa tahapan dan penjelasannya dapat dilihat pada Gambar 17 dibawah ini.



Gambar 17. Flowchart Perancangan Mekanikal

Berdasarkan Gambar 17, dalam perancangan ini peneliti melakukan studi literatur atau mencari referensi terlebih dahulu, lalu mengukur dan melakukan perancangan desain komponen-komponen yang diperlukan. Setelah itu melakukan assembly dan dilihat apakah sudah sesuai dengan yang diinginkan. Akhir dari itu melakukan 3d Printing dan juga melakukan penggabungan terhadap komponen lainnya dan dinyatakan selesai.

Berikut merupakan rancangan desain mekanikalnya, bisa dilihat pada Gambar 18 dibawah ini.



Gambar 18. Perancangan Desain Mekanikal

Keterangan :

1. Desain Conveyor dan Box controller berisikan motor driver L298N, Sensor Arus dan Tegangan, ESP32-Receiver, PCB, dan Perkabelan.
2. Cover ESP32 Sender yang berada dekat dengan Operator.

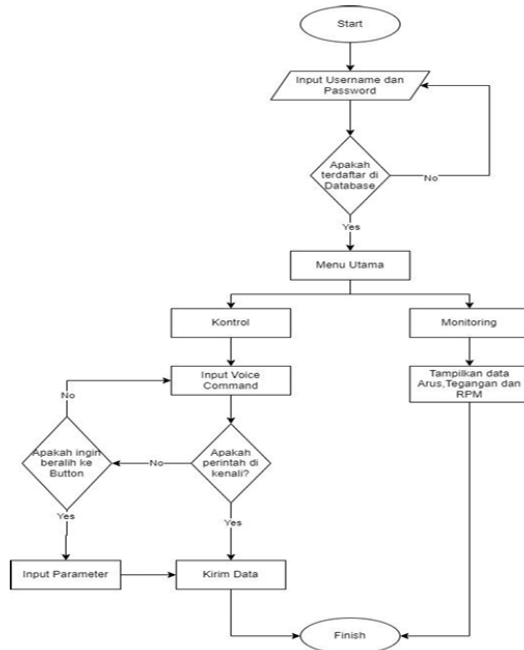
Adapun dimensi pada tiap rancangan yang telah dibuat dapat dilihat pada Tabel 4 dibawah ini.

Tabel 4. Dimensi Desain Mekanikal

NAMA	PANJANG	LEBAR	TINGGI
Konveyor	31 cm	8,5 cm	10 cm
ESP Cover	5,5 cm	3,5 cm	2 cm
Box Controller	18,5 cm	11,5 cm	6,5 cm

3.1.4 Perancangan Program

Berikut ini adalah *flowchart* perancangan program yang akan peneliti lakukan agar dalam pengerjaan aktualnya lebih terstruktur, Adapun beberapa tahapan dan penjelasannya dapat dilihat pada Gambar 19 dibawah ini.

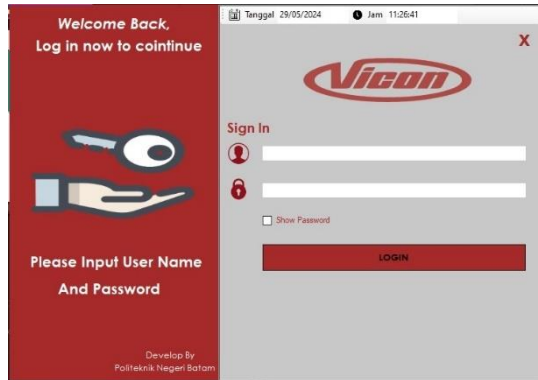


Gambar 19. Perancangan Interface

Berdasarkan Gambar 19, dalam perancangan interace ada beberapa tahapan yang harus di lakukan oleh user, Pertama user harus memasukkan username dan password, apabila password dan username tidak terdaftar maka user akan di minta untuk menginput ulang username dan password namun apabila username dan password sudah terdaftar maka user akun dapat masuk ke menu utama, pada menu utama terdapat 2 fitur yaitu monitoring dan kontrol, pada menu monitoring akan muncul tampilan pembacaan arus, tegangan dan RPM konveyor. Sedangkan pada menu kontrol user dapat mengatur kecepatan motor menggunakan perintah suara, apabila perintah di kenali maka data akan di kirim ke esp receiver (kontrol sistem) namun apabila perintah tidak di kenali maka sistem akan menawarkan input parameter menggunakan button jika user menolak maka akan di minta

untuk menginputkan suara kembali, namun apabila user ingin beralih menggunakan button maka user dapat memilih parameter yang diinginkan yaitu LOW, MID, dan HIGH setelah itu data akan di kirim ke esp receiver (Kontrol Sistem).

Adapun Rancangan GUI (Graphical User Interface) yang sudah kami buat seperti Gambar 20 berikut.



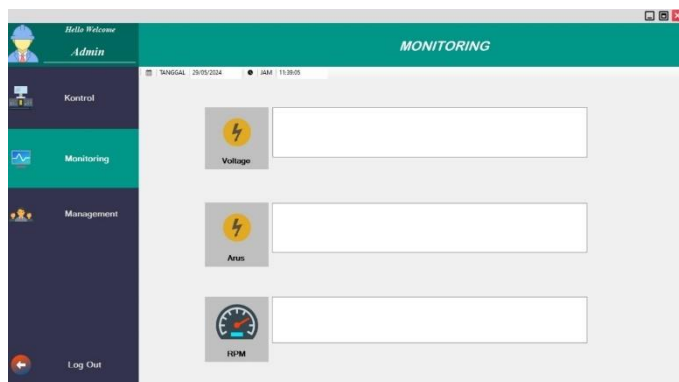
Gambar 20. Desain Menu Login

Berdasarkan Gambar 20, pada menu ini pengguna diminta untuk menginputkan username dan password. Jika username dan password yang dimasukkan terdaftar di dalam database, maka pengguna akan dialihkan ke menu utama. Namun, jika username dan password tidak sesuai atau tidak terdaftar, pengguna akan diminta untuk menginputkan ulang *username* dan *password*. Validasi ini dapat dilihat lebih jelas pada Gambar 20, yang menggambarkan tampilan autentikasi pengguna. Hal ini bertujuan untuk memastikan hanya pengguna yang terverifikasi yang dapat mengakses menu utama.



Gambar 21. Desain Menu Kontrol

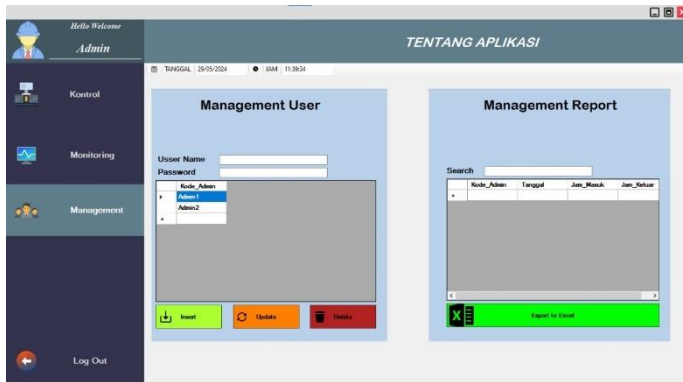
Berdasarkan Gambar 21, pada menu ini pengguna dapat melakukan kontrol pada motor DC menggunakan perintah suara yang terdiri dari parameter start, stop, low, mid, dan high. Namun, apabila terjadi masalah pada pengenalan suara atau lingkungan kerja yang kurang kondusif, tersedia opsi alternatif untuk melakukan kontrol menggunakan push button, pilihan ini memberikan fleksibilitas kepada pengguna dalam mengoperasikan motor DC, terutama saat kondisi suara tidak ideal atau perangkat pengenalan suara mengalami kendala.



Gambar 22. Desain Menu Monitoring

Berdasarkan Gambar 22, menu ini berfungsi untuk memantau RPM dari motor DC secara realtime. Proses monitoring ini memberikan informasi langsung kepada pengguna mengenai kecepatan putaran motor, sehingga dapat digunakan untuk

evaluasi kinerja sistem yang menunjukkan antarmuka monitoring dan data RPM yang ditampilkan secara aktual.



Gambar 23. Desain Menu Management

Berdasarkan Gambar 23, *Menu Management User* berfungsi untuk melakukan penambahan, penghapusan, atau pengeditan data pengguna yang nantinya akan dikirimkan ke database. Sedangkan pada *Menu Management Report*, terdapat fitur untuk melihat riwayat login dari setiap pengguna yang menggunakan aplikasi. Riwayat tersebut mencakup informasi seperti tanggal, nama, status, jam masuk, dan jam keluar. Proses dan tampilan kedua menu ini dapat dilihat lebih jelas pada Gambar 23, yang menunjukkan alur serta antarmuka pengguna untuk masing-masing fungsi tersebut. Hal ini bertujuan untuk mempermudah pengelolaan data dan pelacakan aktivitas pengguna secara efektif.

3.2. Alat dan Bahan

Dalam penelitian ini, ada beberapa alat dan bahan yang digunakan berdasarkan kebutuhan dan referensi yang diambil. Sehingga peneliti membuat rincian dan harga yang bisa dilihat pada Tabel 5 dibawah ini.

Tabel 5. Estimasi Alat dan Bahan

No	Alat/Bahan	Harga Satuan (Rp.)	Jumlah	Total (Rp.)	Keterangan
1	ESP32	76.000	2	152.000	Dana Pribadi
2	Sensor Arus (ACS712)	23.000	1	23.000	Dana Pribadi

3	Sensor Tegangan	4.000	1	12.000	Dana Pribadi
4	Breadboard	12.000	2	24.000	Dana Pribadi
5	Motor driver L298N	23.000	1	23.000	Dana Pribadi
6	Motor DC with Encoder	165.000	1	165.000	Dana Pribadi
7	Case ESP32	20.000	1	20.000	Dana Pribadi
8	Power Adaptor 12V	15.000	1	15.000	Dana Pribadi
7	PC server	-	1	-	Dana Pribadi
Total				435.000	Dana Pribadi

3.3. Pengujian

Pengujian dilakukan untuk memastikan bahwa sistem dapat beroperasi dengan baik dan sesuai dengan tujuan yang telah ditetapkan dalam Tugas Akhir. Tujuan utama sistem ini adalah untuk mengendalikan conveyor secara efektif dan akurat, dengan memanfaatkan teknologi kendali jarak jauh berbasis perintah suara menggunakan protocol ESP-NOW. Sistem ini juga dirancang untuk memonitor arus, tegangan, dan kecepatan (RPM) sesuai dengan kondisi yang telah disesuaikan pada tampilan antarmuka grafis (GUI). Selain itu, penggunaan kontrol PID pada motor DC bertujuan untuk meningkatkan keandalan dan kinerja keseluruhan sistem.

Pengujian sistem kendali conveyor ini melibatkan pemantauan dan pengendalian melalui dua komponen utama, yaitu perintah suara dan sistem visual berbasis C# yang bertindak sebagai antarmuka untuk monitoring dan pengaturan. Pengujian mencakup serangkaian teknik untuk mengevaluasi kinerja sistem secara menyeluruh. Berbagai parameter, seperti arus, tegangan, dan kecepatan motor DC, diuji untuk memastikan bahwa sistem dapat berfungsi dengan optimal pada kondisi yang telah ditentukan. Kontrol PID juga diuji untuk menilai bagaimana algoritma ini dapat meningkatkan responsivitas sistem dan kestabilan operasi motor. Berikut adalah detail tentang pengujian yang dilakukan.

3.3.1 Pengujian Mekanikal

Pada pengujian mekanikal, Penulis memastikan integrasi antara conveyor, box controller, dan cover ESP32 Pengirim dengan desain mekanikal yang kompleks.

Cover ESP32 dibuat menggunakan teknologi 3D printing dengan bahan filamen yang presisi untuk melindungi komponen, sementara box controller dirancang untuk mendukung penempatan ESP32 penerima secara optimal. Prototipe konveyor dibuat dari akrilik dengan roller pipa, dilengkapi engsel dan baut untuk memastikan kekokohan, serta dirancang agar kompatibel dengan seluruh sistem. Layout mekanikal ini memastikan setiap komponen saling terhubung dengan baik, mendukung efisiensi pengujian.

3.3.2.1 Pengujian Sensor Encoder

Dalam pengujian ini, penulis melakukan pengaturan kecepatan motor DC dengan mengubah nilai PWM menggunakan driver L298N yang diberi suplai tegangan sebesar 12V. Pengaturan ini bertujuan untuk mengevaluasi respons motor terhadap variasi nilai PWM yang diberikan serta memastikan bahwa motor berfungsi sesuai dengan setpoint yang diinginkan. Data kecepatan aktual motor diukur menggunakan sensor encoder, yang hasil pembacaannya diamati melalui serial monitor.

3.3.2.2 Pengujian Sensor Tegangan dan Arus

Dalam pengujian ini, penulis melakukan pengecekan terhadap sensor tegangan dan Arus dengan cara mengaplikasikan tegangan DC sebesar 12V langsung dari adapter 12V. Pembacaan tegangan yang dihasilkan oleh sensor kemudian ditampilkan melalui tampilan serial monitor untuk memastikan akurasi dan kestabilannya. Langkah ini bertujuan untuk memastikan bahwa sensor tegangan dapat berfungsi sesuai dengan spesifikasinya dalam kondisi yang telah ditentukan.

3.3.2.3 Pengujian Motor DC 12V

Dalam pengujian ini, penulis melakukan pengecekan motor DC diuji dengan diberi suplai tegangan sebesar 12V untuk memastikan bahwa motor berfungsi dengan baik. Pengujian ini juga dilakukan untuk mengukur kecepatan putaran maksimum (RPM) yang dapat dicapai oleh motor DC tersebut. Selain itu, kecepatan aktual motor diperiksa menggunakan tachometer guna memvalidasi hasil pengukuran dan memastikan akurasi kinerja motor. Pengujian ini bertujuan mengetahui performa motor DC secara keseluruhan, termasuk kestabilannya saat beroperasi pada tegangan suplai yang telah ditentukan.

3.3.3 Pengujian GUI (*Graphical User Interface*)

Dalam pengujian ini, penulis melakukan pengecekan untuk memastikan integrasi dan fungsionalitas berbagai fitur yang dirancang dalam sistem. Pada menu login, pengujian difokuskan untuk memverifikasi koneksi dengan database

MySQL dan memastikan sistem dapat memvalidasi kredensial pengguna secara akurat, sehingga hanya pengguna terdaftar yang dapat mengakses aplikasi. Selanjutnya, pengujian dilakukan pada kemampuan GUI dalam mengirimkan data ke ESP sender menggunakan protokol ESP-NOW, dengan menguji perintah seperti ON, OFF, LOW, MID, dan HIGH. Proses ini melibatkan simulasi pengiriman data untuk memastikan komunikasi berjalan dengan lancar, serta mengevaluasi kecepatan dan keandalan protokol dalam mendukung komunikasi jarak jauh. Pada menu monitoring, pengujian bertujuan memastikan bahwa data terbaru yang dikirimkan oleh ESP receiver melalui sensor dapat ditampilkan secara real-time di GUI dengan pembaruan langsung dari database. Sementara itu, pengujian pada menu manajemen pengguna meliputi kemampuan untuk melakukan operasi Insert, Update, dan Delete pada data pengguna melalui antarmuka yang telah dirancang. Seluruh pengujian ini bertujuan untuk memastikan bahwa sistem tidak hanya berfungsi sesuai spesifikasi, tetapi juga memberikan pengalaman pengguna yang efisien dan andal.

Bab 4. Hasil dan Pembahasan

Pada Bab 4 ini, penulis akan mencantumkan hasil pengujian serta melakukan pembahasan secara mendalam terkait setiap kondisi, parameter, dan keadaan yang telah ditetapkan dalam proses pengujian. Data yang disajikan merupakan hasil dari pengujian menyeluruh pada setiap sub-sistem yang telah dipastikan berfungsi dengan baik. Dengan demikian, penulis dapat mengintegrasikan hasil-hasil tersebut untuk membentuk sistem yang utuh dan fungsional sebagai bagian dari tugas akhir ini. Pembahasan ini tidak hanya mencakup analisis terhadap kinerja setiap komponen yang diuji, tetapi juga memberikan wawasan tentang kelebihan, kekurangan, serta potensi perbaikan yang dapat diterapkan untuk meningkatkan efektivitas dan efisiensi sistem secara keseluruhan. Hasil yang diperoleh akan dibahas secara rinci untuk memastikan bahwa tujuan yang telah ditetapkan dalam penelitian ini tercapai dengan optimal.

4.1 Hasil Pengujian Mekanikal

Ketahanan desain mekanik pada cover ESP32, box controller dan konveyor menjadi fokus utama dalam pengujian ini. Tujuannya adalah untuk melindungi komponen dari kerusakan fisik yang dapat menyebabkan error. Pengujian dilakukan untuk memastikan apakah sistem pengirim dan penerima mampu berfungsi secara optimal.



Gambar 24. Hasil Pengujian Mekanikal

Setelah rancangan desain telah dibuat seperti Gambar 24, penulis melakukan pengujian untuk memastikan sistem dapat mengatur kecepatan dan stabilitas konveyor sesuai dengan perintah yang diberikan, sehingga menggambarkan efektivitas kontrol konveyor jarak jauh dalam mendukung pengoperasian sistem secara keseluruhan. Adapun hasil yang telah dilakukan sebagai berikut :

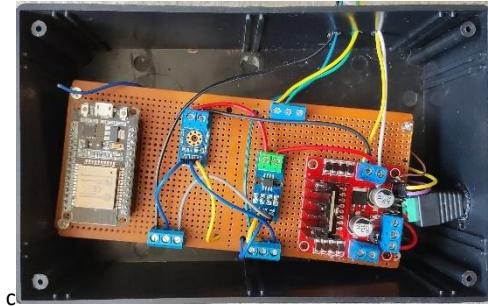
Tabel 6. Hasil Pengujian Mekanikal

Kondisi	Conveyor
On	Berjalan
Low	Berjalan
Mid	Berjalan
High	Berjalan

Berdasarkan hasil pengujian pada Tabel 6, sistem berhasil mengatur kecepatan dan stabilitas konveyor sesuai dengan perintah yang diberikan pada berbagai kondisi (On, Low, Mid, dan High). Konveyor dapat berjalan dengan baik di setiap pengaturan, menggambarkan bahwa sistem pengirim dan penerima mampu menjalankan kontrol jarak jauh secara efektif dan responsif. Hal ini menunjukkan bahwa rancangan desain mekanikal dan integrasi sistem yang diterapkan telah mendukung pengoperasian sistem secara optimal sesuai dengan tujuan tugas akhir.

4.2 Hasil Pengujian Elektrikal

Berdasarkan pengujian yang telah dilakukan sebelumnya, penulis telah menguji beberapa komponen penting, yaitu sensor tegangan, sensor arus ACS 712, dan rotary encoder. Hasil dari pengujian ini mencakup tingkat keakuratan pembacaan yang dihasilkan oleh masing-masing sensor, yang kemudian dibandingkan dengan data referensi yang diambil menggunakan alat ukur yang telah terkalibrasi, seperti multimeter untuk pengukuran tegangan dan arus, serta tachometer untuk mengukur kecepatan putaran. Perbandingan ini bertujuan untuk memastikan bahwa pembacaan sensor yang digunakan dalam sistem sesuai dengan nilai yang sebenarnya, serta untuk memverifikasi bahwa alat ukur yang digunakan dalam pengujian dapat menghasilkan hasil yang konsisten. Adapun rangkaian yang sudah penulis buat bisa dilihat pada Gambar 25 dibawah ini.



Gambar 25. Hasil Pengujian Elektrikal

Pada Gambar 25, rangkaian ini menggunakan adaptor 220VAC ke 12 VDC sebagai sumber daya, dengan terminal positif adaptor terhubung ke pin +12 dan terminal negatif adaptor ke GND pada driver motor L298N. Selanjutnya, output +5V dari driver motor dihubungkan ke pin Vin pada ESP32, serta ke Encoder (+), sensor tegangan, dan ACS712. Pin ENB pada driver motor terhubung ke GPIO12, IN3 ke GPIO27, dan IN4 ke GPIO14 pada ESP32. Output dari L298N dihubungkan ke wire out pada ACS712 dan VCC pada sensor tegangan, sementara pin Out4 terhubung ke GND sensor tegangan dan Encoder (-). Pin data (S) pada sensor tegangan dihubungkan ke GPIO32 pada ESP32, dan pin OUT pada ACS712 dihubungkan ke GPIO34. Terakhir, pin Encoder A Phase pada motor DC terhubung ke GPIO33, dan pin Encoder B Phase terhubung ke GPIO34 pada ESP32.

4.2.1 Hasil Pengujian Menggunakan Sensor

Dalam pengujian ini, penulis melakukan evaluasi terhadap pembacaan arus, tegangan, dan RPM menggunakan sensor ACS 712, sensor tegangan, dan rotary encoder. Pengujian dilakukan dengan mengatur beberapa setpoint RPM yang berbeda, yaitu 100, 200, 400, dan 600 RPM, untuk menilai kinerja sensor pada berbagai kondisi operasional. Tujuan dari pengujian ini adalah untuk memverifikasi apakah sensor yang digunakan dapat memberikan pembacaan yang akurat dan konsisten sesuai dengan nilai yang diharapkan pada setiap setpoint yang diuji. Hasil dari pengujian ini, yang mencakup data pembacaan arus, tegangan, dan RPM dicantumkan pada Tabel 7 dibawah ini.

Tabel 7. Hasil Pembacaan Sensor

Setpoint (RPM)	Hasil Pembacaan Sensor		
	Kecepatan Putar (RPM)	Arus (mA)	Tegangan (V)
100	99 – 102	20,6 mA	1.9 V
200	198 – 201	23.8 mA	2.5 V
400	399 – 403	63,4 mA	4.6 V
600	698 – 602	75.9 mA	7.9 V

Dari Tabel 7 hasil pembacaan sensor, dapat terlihat bahwa sensor mampu mengukur arus, tegangan, serta RPM pada motor DC dengan variasi nilai yang beragam. Pengukuran tersebut dilakukan berdasarkan RPM yang dijadikan sebagai setpoint. Hal ini menunjukkan bahwa sensor memiliki kemampuan untuk memantau berbagai parameter penting motor DC secara akurat, sehingga dapat mendukung analisis kinerja motor dalam berbagai kondisi operasional. Selain itu, kemampuan ini juga membuktikan keandalan sistem dalam menangani perubahan setpoint RPM yang dinamis.

4.2.2 Hasil Pengujian Menggunakan Alat Ukur

Dalam pengujian ini, penulis mengumpulkan data pembacaan arus, tegangan, dan RPM menggunakan alat ukur yang telah terkalibrasi dengan akurat. Pengujian dilakukan dengan mengatur beberapa parameter RPM yang berbeda, yaitu 100, 200, 400, dan 600 RPM, untuk mengevaluasi kinerja sistem pada berbagai tingkat kecepatan. Tujuan dari pengujian ini adalah untuk memastikan bahwa pengukuran yang diperoleh melalui alat ukur yang terkalibrasi mencerminkan kondisi operasional yang sebenarnya, serta untuk memverifikasi keakuratan dan konsistensi hasil pengukuran pada setiap setpoint yang diuji. Hasil pengujian yang diperoleh dapat dilihat pada Tabel 8 dibawah ini.

Tabel 8. Hasil Pembacaan Alat ukur

Setpoint (RPM)	Hasil Pembacaan Manual		
	Kecepatan Putar (RPM)	Arus (mA)	Tegangan (V)
100	97 – 103	20,3 mA	1.8 V
200	198 – 202	23.4 mA	2.8 V
400	399 – 403	63,1 mA	4.9 V
600	697 – 605	75.7 mA	8.1 V

Berdasarkan Tabel 8, hasil pembacaan alat ukur menampilkan data yang diperoleh melalui penggunaan perangkat ukur yang telah terkalibrasi dengan baik.

Perangkat yang digunakan meliputi multimeter untuk mengukur tegangan dan arus, serta tachometer untuk menghitung RPM pada motor DC. Dengan menggunakan alat ukur yang terkalibrasi, data yang dihasilkan dapat dipercaya dalam hal akurasi dan konsistensi. Pendekatan ini memastikan bahwa pengujian dilakukan sesuai dengan standar yang berlaku, memberikan hasil yang valid untuk analisis lebih lanjut terkait kinerja motor DC di berbagai kondisi operasional.

4.2.3 Hasil Pengujian Perhitungan Error Sistem

Dalam pengujian ini, penulis melakukan perbandingan antara hasil pembacaan arus, tegangan, dan RPM yang diperoleh dari sensor dengan pengukuran yang dilakukan menggunakan alat ukur yang terkalibrasi. Pengujian dilakukan sesuai dengan parameter yang telah ditentukan sebelumnya, untuk memastikan bahwa pembacaan sensor dapat diandalkan dan sesuai dengan nilai yang diukur secara manual. Data yang diperoleh dari kedua sumber tersebut kemudian dianalisis menggunakan rumus perhitungan yang relevan untuk mengukur tingkat akurasi dan konsistensi antara sensor dan alat ukur. Adapun rumus dan data yang di dapat adalah sebagai berikut.

$$\text{Error (\%)} = \frac{\text{Hasil Pembacaan Alat Ukur} - \text{Hasil Pembacaan Sensor}}{\text{Hasil Pembacaan Sensor}} \times 100 \quad (1)$$

Tabel 9. Hasil Pengujian Perhitungan Error Sistem

Set Point (RPM)	Error (%)		
	Kecepatan Motor	Arus	Tegangan
100	1 %	1,47 %	5,5 %
200	0,99 %	1,97 %	6,6 %
400	0,49 %	0,47 %	6,1 %
600	0,33 %	0,26 %	2,4 %

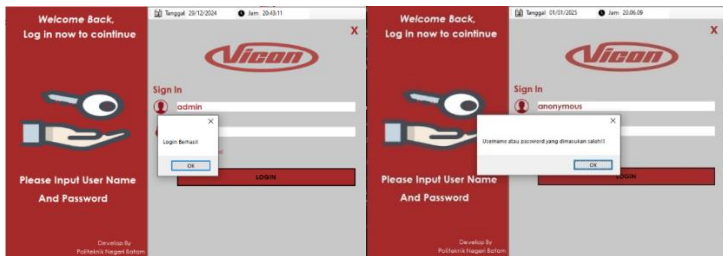
Berdasarkan Tabel 9, dapat disimpulkan bahwa selisih error antara pembacaan menggunakan sensor dan hasil dari alat ukur yang terkalibrasi menunjukkan perbedaan yang tidak signifikan. Hal ini mengindikasikan bahwa sensor yang digunakan mampu membaca parameter dengan tingkat akurasi yang cukup baik. Dengan demikian, sensor tersebut dapat diandalkan untuk mengukur variabel yang diperlukan dalam sistem, seperti arus, tegangan, dan kecepatan (RPM). Kesimpulan ini juga menunjukkan bahwa sensor dapat berkontribusi secara efektif dalam mendukung kinerja sistem secara keseluruhan, baik dalam pengambilan data real-time maupun dalam pengendalian yang presisi.

4.3 Hasil Pengujian GUI

Pada bagian ini, peneliti melakukan pengujian menyeluruh terhadap interface yang telah dirancang, dengan tujuan untuk memastikan bahwa setiap fitur pada aplikasi berfungsi sebagaimana mestinya. Pengujian mencakup beberapa aspek utama yaitu menu login, menu kontrol, menu monitoring dan menu manajemen user.

4.3.1 Hasil Pengujian Pada Menu Login

Pertama menu login pada menu ini di lakukan pengujian login aplikasi menggunakan username yang terdaftar di database dan username yang belum terdaftar. Tujuan dari pengujian ini memastikan interface dapat terhubung dengan database MySQL dan dapat memvalidasi pengguna untuk memastikan hanya user yang sudah terdaftar yang dapat mengakses sistem.



Gambar 26. Hasil Pengujian Menu Login

Berdasarkan Gambar 26, hasil pengujian pada fitur login menunjukkan bahwa sistem telah mampu membedakan antara pengguna yang terdaftar dan yang tidak terdaftar di database. Bagi pengguna yang terdaftar, akan muncul pesan pop-up “Login berhasil,” dan pengguna akan diarahkan ke menu utama. Sementara itu, bagi pengguna yang tidak terdaftar, sistem akan menampilkan pesan “Username atau password salah!!!” dan meminta user untuk menginput ulang data login.

4.3.2 Hasil Pengujian Menu kontrol

Selanjutnya, dilakukan pengujian pada menu kontrol, yang berperan dalam mengirimkan data dari interface ke rangkaian kontrol menggunakan protokol komunikasi yang telah ditentukan yaitu ESP-NOW. Pada pengujian ini penulis melakukan pengujian jarak akses dan akurasi voice kontrol. Untuk jarak akses di lakukan dari jarak 1 – 10 meter di lingkungan indoor dan outdoor yang terdata pada Tabel 10 dibawah ini.

Tabel 10. Hasil Pengujian Jarak Konektivitas

Jarak	Diluar ruangan			Dalam ruangan		
	Terhubung	Tidak	Delay	Terhubung	Tidak	Delay
1 Meter	✓		0.70 ms	✓		0.60 ms
2 Meter	✓		0.72 ms	✓		0.62 ms
3 Meter	✓		0.75 ms	✓		0.65 ms
4 Meter	✓		0.78 ms	✓		0.68 ms
5 Meter	✓		0.81 ms	✓		0.70 ms
6 Meter	✓		0.85 ms	✓		0.85 ms
7 Meter	✓		0.90 ms		✓	-
8 Meter	✓		1 sec		✓	-
9 Meter	✓		1.1 sec		✓	-
10 Meter		✓	-		✓	-

Berdasarkan Tabel 10, hasil pengujian koneksi ESP-NOW menunjukkan bahwa pada rentang jarak 1–5 meter, koneksi tetap stabil baik di dalam maupun di luar ruangan. Namun, koneksi mulai terputus pada jarak 7 meter saat berada di dalam ruangan, sementara di luar ruangan koneksi masih stabil hingga jarak 10 meter sebelum akhirnya terputus. Temuan ini mengindikasikan bahwa performa koneksi ESP-NOW dipengaruhi oleh hambatan fisik seperti dinding dan material bangunan, yang lebih signifikan pada pengujian dalam ruangan. Sebaliknya, di luar ruangan, koneksi dapat menjangkau jarak yang lebih jauh karena minimnya hambatan sinyal, menjadikan teknologi ini lebih efektif untuk aplikasi luar ruangan dengan kebutuhan komunikasi jarak menengah.

4.3.3 Pengujian Speech Recognition

Selanjutnya penulis melakukan pengujian terhadap inputan *speech recognition* untuk mengukur tingkat akurasi fitur kontrol suara. Setiap perintah, yaitu *On*, *Off*, *Low*, *Mid*, dan *High*, diuji sebanyak 10 kali. Pengujian dilakukan pada jarak ½ hingga 1 meter dari perangkat. Hasil pengujian yang diperoleh dapat dilihat pada Tabel 11 dibawah ini.

Tabel 11. Hasil Akurasi Berdasarkan Jarak

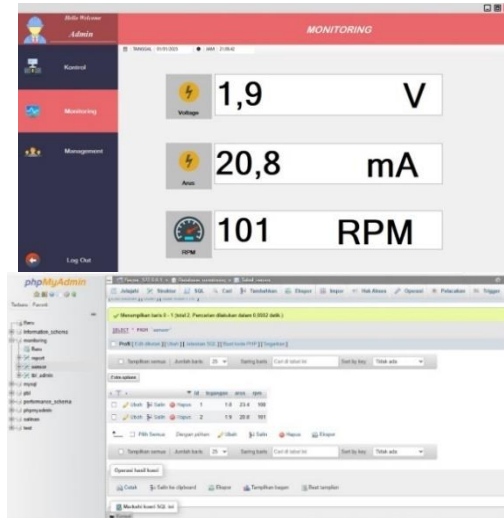
Kata Masukan	Jumlah Pengujian	jarak 1/2 meter	jarak 1 meter
On	10	100%	80%
Off	10	100%	80%
Low	10	100%	60%
Mid	10	80%	50%

High	10	80%	50%
------	----	-----	-----

Berdasarkan Tabel 11, fitur ini berfungsi dengan baik ketika jarak antara pengguna dan antarmuka berada dalam rentang $\frac{1}{2}$ meter. Namun, akurasi cenderung menurun jika pengguna berada pada jarak yang lebih jauh dari antarmuka. Penurunan ini disebabkan oleh penggunaan modul mikrofon internal pada perangkat untuk menangkap data suara. Mikrofon internal memiliki keterbatasan dalam menangkap suara dari jarak jauh, terutama jika terdapat gangguan lingkungan seperti kebisingan latar belakang.

4.3.4 Hasil Pengujian Menu Monitoring

Selanjutnya pengujian menu monitoring bertujuan untuk menampilkan data hasil pembacaan sensor yang dikirimkan oleh sistem kontrol ke database MySQL. Proses ini dilakukan untuk memastikan bahwa data yang dihasilkan oleh sensor dapat diterima, disimpan, dan dikelola dengan baik dalam database. Setelah data berhasil disimpan, langkah berikutnya adalah menampilkan informasi tersebut secara real-time melalui fitur monitoring pada aplikasi. Pengujian ini bertujuan untuk memvalidasi alur kerja sistem mulai dari pengiriman data, penyimpanan dalam database, hingga penyajian informasi kepada pengguna melalui antarmuka monitoring. Hasil pengujian yang diperoleh dari proses ini akan dijelaskan secara rinci pada Gambar 27 dibawah ini.

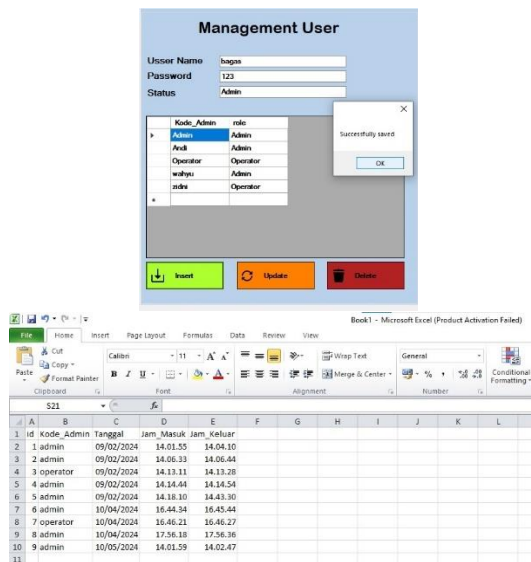


Gambar 27. Hasil Pengujian Menu Monitoring

Berdasarkan Gambar 27, hasil pengujian pada menu monitoring menunjukkan bahwa antarmuka mampu menampilkan data terbaru yang diterima oleh database secara real-time. Hal ini tidak hanya membuktikan bahwa menu monitoring berfungsi dengan baik, tetapi juga memastikan bahwa sistem bekerja secara efisien dalam menyinkronkan data dan memberikan informasi yang akurat kepada pengguna. Kemampuan ini menjadi indikator bahwa sistem telah dirancang dan diimplementasikan dengan optimal untuk mendukung kebutuhan pemantauan secara langsung.

4.3.3 Hasil Pengujian Management User

Selanjutnya, penulis melakukan pengujian pada menu manajemen pengguna meliputi kemampuan untuk melakukan operasi Insert, Update, dan Delete pada data pengguna yang terhubung dengan database serta melakukan rekap dan export data login ke ms excel Adapun hasil yang di dapat adalah sebagai berikut.



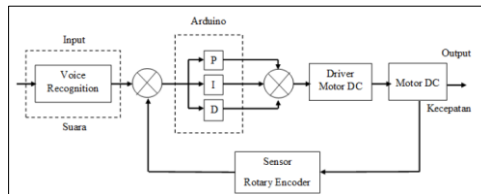
Gambar 28. Hasil Pengujian Menu Management User

Berdasarkan Gambar 28, pengujian ini menunjukkan bahwa antarmuka aplikasi memiliki kemampuan untuk menambahkan pengguna baru dengan cara menginputkan username, password, serta status pengguna. Selain itu, antarmuka juga mampu mencatat dan merekap data login pengguna, yang kemudian dapat diekspor ke dalam format Microsoft Excel. Fitur ini tidak hanya mempermudah

pengelolaan data pengguna, tetapi juga mendukung dokumentasi dan analisis lebih lanjut terkait aktivitas login pada aplikasi, sehingga meningkatkan efisiensi dan akurasi pengelolaan sistem.

4.4 Hasil Pengujian *PID Controller* pada Sistem

Pada bagian ini, penulis melakukan pengujian untuk menentukan parameter optimal pengendali PID (Proportional-Integral-Derivative) pada sistem motor DC. Pengujian ini bertujuan untuk memperoleh nilai konstanta kontrol K_p (gain proporsional), K_i (gain integral), dan K_d (gain derivatif) yang menghasilkan respons sistem terbaik, seperti waktu tunda yang minimal, overshoot yang terkendali, serta kesalahan steady-state yang kecil. Metode Trial and Error digunakan dalam proses penyetelan parameter PID. Metode ini dipilih karena memberikan fleksibilitas dalam menyesuaikan parameter secara langsung berdasarkan pengamatan terhadap respons sistem. Dalam pendekatan ini, penulis secara bertahap mengubah nilai K_p , K_i , dan K_d , kemudian menganalisis perubahan pada karakteristik sistem, seperti waktu naik (rise time), waktu pemulihan (settling time), dan kestabilan sistem.



Gambar 29. Blok Diagram Pengujian PID

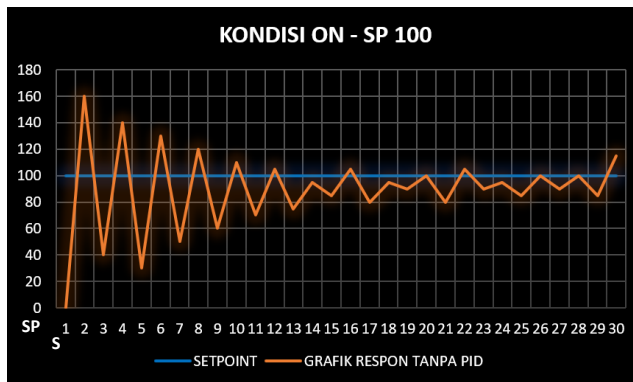
Gambar 29, menunjukkan diagram blok sistem kendali yang digunakan dalam pengendalian kecepatan motor DC. Sistem ini dirancang untuk mengontrol kecepatan motor berdasarkan nilai masukan (*setpoint*) yang telah ditentukan. Masukan tersebut merupakan nilai kecepatan yang diinginkan, yang akan diproses oleh pengendali PID yang diimplementasikan pada modul ESP32. Pengendali PID berfungsi untuk menghasilkan keluaran berupa tegangan kontrol, yang kemudian digunakan untuk mengaturlah driver motor. Driver motor bertugas mengendalikan arah putaran dan kecepatan motor DC berdasarkan sinyal kontrol tersebut. Tegangan yang dihasilkan oleh driver motor akan langsung memengaruhi kecepatan motor, sesuai dengan perhitungan dan penyesuaian yang dilakukan oleh pengendali PID. Selain itu, sistem dilengkapi dengan sensor rotary encoder, yang berperan sebagai komponen umpan balik (*feedback*). Sensor ini secara terus-menerus mengukur kecepatan aktual motor DC, lalu mengirimkan data tersebut

kembali ke ESP32. Data kecepatan aktual ini dibandingkan dengan nilai *setpoint* oleh pengendali PID untuk menghitung selisih error.

4.4.1 Pengujian terhadap Setpoint Tanpa PID Controller

Pada tahap ini, Penulis melakukan pengujian yang bertujuan untuk mendapatkan data dasar kinerja sistem motor DC tanpa adanya pengaruh pengendali PID. Data yang digunakan dalam analisis ini diperoleh dari rata-rata 30 pembacaan serial monitor selama kurang lebih 30 detik. Grafik yang dihasilkan akan menjadi titik acuan untuk membandingkan kinerja sistem setelah diterapkan pengendali PID. Dengan demikian, efektivitas pengendali PID dalam meningkatkan performa sistem dapat diukur secara lebih akurat.

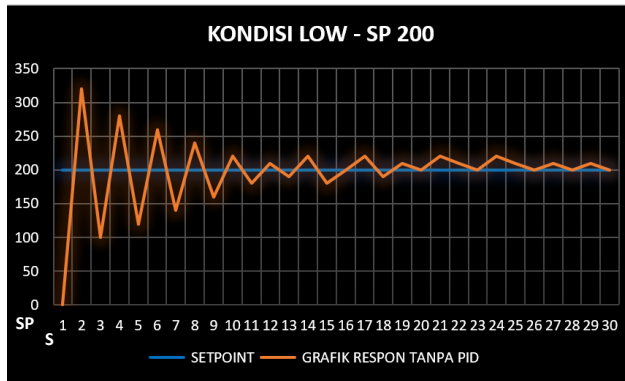
Adapun hasil dan pembahasan berikut pada kondisi On dengan setpoint 100 RPM yang bisa dilihat pada Gambar 30 dibawah ini.



Gambar 30. Grafik Respon Setpoint 100 Tanpa PID

Berdasarkan Gambar 30, grafik respons menunjukkan adanya overshoot yang signifikan, mencapai 160 pada setpoint 100, serta osilasi yang tidak teratur. Hal ini mengindikasikan bahwa sistem mengalami ketidakstabilan yang cukup tinggi, yang menyebabkan kecepatan rotasi motor (rpm) yang dihasilkan jauh dari kondisi stabil yang diharapkan.

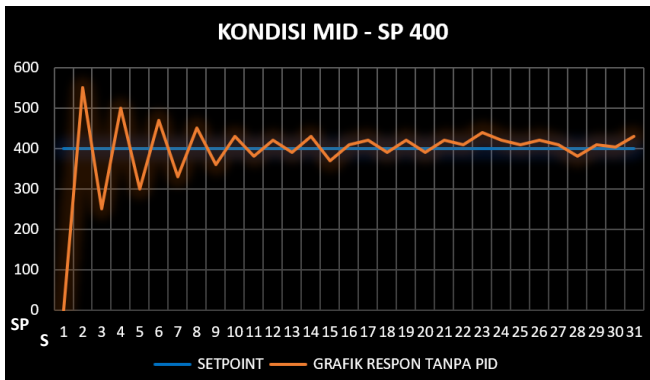
Selanjutnya adalah hasil dan pembahasan berikut pada kondisi Low dengan setpoint 200 RPM yang bisa dilihat pada Gambar 31 dibawah ini.



Gambar 31. Grafik Respon Setpoint 200 Tanpa PID

Berdasarkan Gambar 31, grafik respons memperlihatkan adanya overshoot yang cukup besar, melebihi 300 pada setpoint 200, disertai osilasi yang tidak konsisten. Hal ini menandakan bahwa sistem mengalami ketidakstabilan yang signifikan, yang mengakibatkan kecepatan rotasi motor (rpm) yang dihasilkan jauh dari keadaan stabil yang diinginkan.

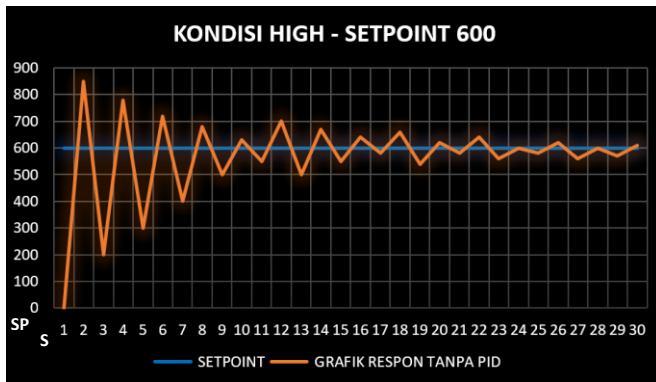
Selanjutnya adalah hasil dan pembahasan berikut pada kondisi Mid dengan setpoint 400 RPM yang bisa dilihat pada Gambar 32 dibawah ini.



Gambar 32. Grafik Respon Setpoint 400 Tanpa PID

Berdasarkan Gambar 32, grafik respons menunjukkan overshoot yang sangat besar, mencapai 550 pada setpoint 400, dengan osilasi yang tidak stabil. Hal ini mengindikasikan bahwa sistem mengalami ketidakstabilan yang serius, yang menyebabkan kecepatan rotasi motor (*rpm*) yang dihasilkan jauh dari kondisi stabil yang diharapkan.

Selanjutnya adalah hasil dan pembahasan berikut pada kondisi High dengan setpoint 600 RPM yang bisa dilihat pada Gambar 33 dibawah ini.



Gambar 33. Grafik Respon Setpoint 600 Tanpa PID

Berdasarkan Gambar 33, grafik respons memperlihatkan overshoot yang sangat tinggi, mencapai 850 pada setpoint 600, disertai dengan osilasi yang tidak stabil. Hal ini menunjukkan bahwa sistem mengalami ketidakstabilan yang signifikan, yang menyebabkan kecepatan rotasi motor (*rpm*) yang dihasilkan jauh dari kondisi stabil yang diinginkan.

Berdasarkan hasil pengujian awal tanpa pengendali PID, kinerja sistem motor DC menunjukkan bahwa kecepatan aktual motor belum mampu mencapai kestabilan pada setpoint yang diinginkan. Sistem yang diuji mengalami ketidakstabilan yang signifikan pada setiap titik setpoint yang berbeda. Setiap grafik respons menunjukkan adanya overshoot yang sangat besar, diikuti oleh osilasi yang tidak teratur, yang menyebabkan kecepatan rotasi motor (*rpm*) jauh dari kondisi stabil yang diharapkan. Ketidakstabilan ini menunjukkan bahwa sistem perlu dilakukan perbaikan atau pengaturan lebih lanjut untuk mencapai performa yang lebih stabil dan optimal.

4.4.1.1 Pengujian *Tuning* PID dengan Metode *Trial and Error*

Dalam pengujian respons PID, data yang dianalisis mencakup *rise time overshoot*, dan *error steady state* terhadap sinyal masukan yang diberikan. Pada pengujian kontrol PID, nilai K_p , K_i , dan K_d diperoleh melalui metode *Trial and Error* dengan melakukan percobaan berulang. Dalam metode ini, pencarian nilai parameter K_p , K_i , dan K_d dilakukan dengan mengamati respons sistem dan memilih hasil yang paling optimal setelah melakukan tiga kali pengujian pada setiap parameter. Setelah memperoleh respons dari setiap variasi parameter, langkah berikutnya adalah menentukan kombinasi parameter yang paling optimal untuk digunakan dalam penelitian. Sebelum menerapkan pengendali PID pada sistem, perlu dilakukan penentuan nilai parameter terlebih dahulu, karena hal ini sangat berpengaruh terhadap keakuratan dan kestabilan sistem. Oleh karena itu, tuning pengendali menjadi langkah penting agar diperoleh parameter PID yang sesuai.

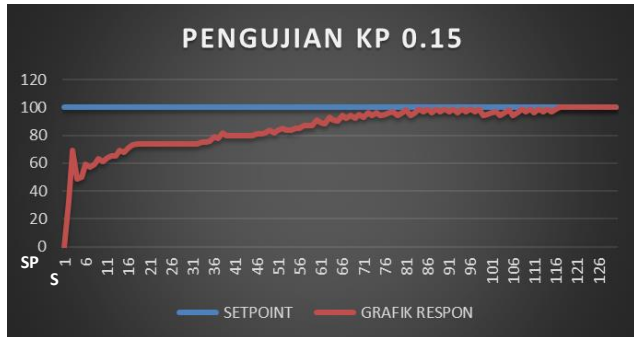
Dalam pengujian ini, proses tuning PID dilakukan secara manual, yaitu dengan menyesuaikan parameter berdasarkan pengamatan langsung terhadap respons sistem setelah setiap pengujian. Adapun tahapan untuk mendapatkan nilai hasil *Tuning* PID sebagai berikut.

4.4.1.2 Pengujian Parameter K_P

Setiap komponen dalam pengendali proporsional (P), integral (I), dan derivative (D) memiliki karakteristik yang berbeda dan memberikan dampak yang unik terhadap respons sistem. Oleh karena itu, perlu dilakukan pengujian terhadap masing-masing parameter untuk memahami pengaruhnya terhadap dinamika sistem kendali. Dalam pengujian ini, setiap parameter diuji secara terpisah dengan menggunakan nilai referensi (set point) yang sama, yaitu 100 RPM. Sebagai langkah awal dalam pengujian parameter PID, pengaruh gain proporsional (K_p) terhadap sistem akan dianalisis terlebih dahulu. Parameter K_p berperan dalam memperkuat sinyal kesalahan dan memberikan respons langsung terhadap perubahan set point.

Pada pengujian pertama yaitu mencari nilai K_P yang cocok untuk sistem, penulis melakukan pengujian dengan nilai $K_P = 0.15$ dengan kondisi K_I dan $K_D = 0.0$ yang merupakan nilai *default* dari sistem yang digunakan.

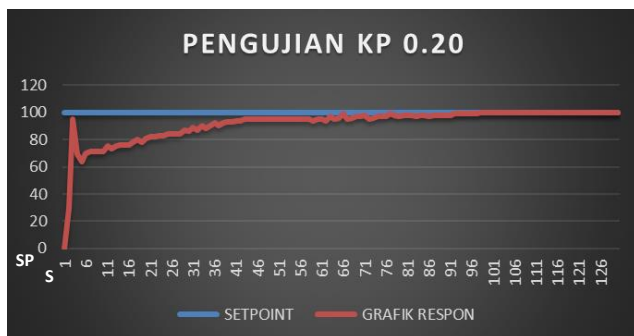
Adapun hasil dan pembahasan jika parameter $K_P 0.15$ dengan setpoint 100 RPM yang bisa dilihat pada Gambar 34 dibawah ini.



Gambar 34. Pengujian Parameter KP 0.15

Berdasarkan Gambar 34, dapat diketahui bahwa dengan nilai $K_p = 0.15$, sistem membutuhkan waktu yang relatif lama untuk mencapai rise time, yaitu sekitar 2 menit 10 detik, hingga akhirnya mendekati nilai set point sebesar 100 RPM. Hal ini menunjukkan bahwa nilai K_p yang digunakan masih tergolong rendah, sehingga respons sistem terhadap perubahan set point berlangsung lebih lambat. Sehingga penulis melakukan pengujian kembali untuk nilai parameter lain.

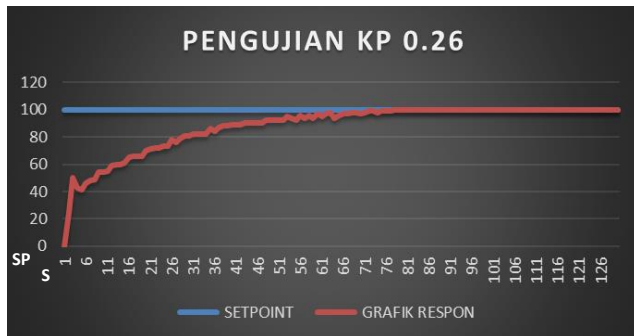
Pada pengujian kedua, penulis melakukan pengujian dengan menaikkan nilai $K_p = 0.20$ dengan kondisi K_I dan $K_D = 0.0$ yang merupakan nilai *default* dari sistem yang digunakan. Selanjutnya adalah hasil pengujian yang bisa dilihat pada Gambar 35 dibawah ini.



Gambar 35. Pengujian Parameter KP 0.20

Berdasarkan Gambar 35, terlihat bahwa dengan nilai $K_p = 0.20$, sistem memerlukan waktu sekitar 1 menit 30 detik untuk mencapai rise time dan mendekati nilai set point sebesar 100 RPM. Hal ini menunjukkan bahwa nilai K_p yang digunakan sudah lebih baik dibandingkan sebelumnya, karena menghasilkan respons yang lebih cepat dalam mencapai set point. Dengan hasil ini, penulis melanjutkan pengujian untuk menentukan nilai parameter lainnya.

Pada pengujian ketiga, penulis meningkatkan nilai K_p menjadi 0.26, dengan kondisi K_i dan K_d tetap pada nilai default, yaitu 0.0. Berikut adalah hasil pengujian yang bisa dilihat pada Gambar 36 dibawah ini.



Gambar 36. Pengujian Parameter K_p 0.26

Berdasarkan Gambar 36, yang terlihat pada gambar, dengan nilai $K_p = 0.26$, sistem memerlukan waktu sekitar 1 menit 10 detik untuk mencapai rise time dan mendekati nilai set point sebesar 100 RPM. Hal ini menunjukkan bahwa nilai K_p yang digunakan telah lebih baik dibandingkan dengan pengujian sebelumnya, karena respons sistem menjadi lebih cepat dalam mencapai set point. Berdasarkan hasil ini, penulis melanjutkan dengan melakukan analisis terhadap tiga kondisi yang telah diuji untuk menentukan nilai K_p yang optimal.

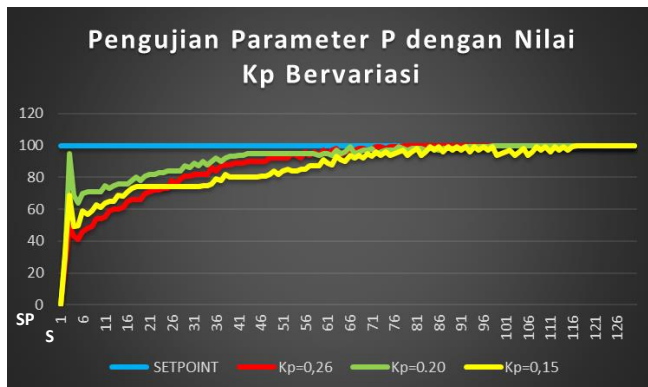
Setelah mengetahui dari 3 kondisi parameter pada pengujian nilai K_p , dapat menghasilkan beberapa keterangan yang dituliskan pada Tabel 12 dibawah ini.

Tabel 12. Hasil Pengujian Parameter KP Bervariasi

Parameter PID			Rise Time	Overshoot
Kp	Ki	Kd	(s)	(%)
0.15	0.0	0.0	130	0
0.20	0.0	0.0	90	0
0.26	0.0	0.0	70	0

Berdasarkan Tabel 12. Hasil Pengujian Parameter KP Bervariasi, parameter Kp yang dilakukan dengan metode Trial and Error, terlihat bahwa peningkatan nilai Kp berpengaruh signifikan terhadap waktu yang dibutuhkan sistem untuk mencapai rise time dan mendekati nilai set point sebesar 100 RPM. Pada pengujian pertama dengan Kp = 0.15, sistem memerlukan waktu sekitar 130 detik untuk mencapai nilai set point. Pada pengujian kedua dengan Kp = 0.20, waktu yang dibutuhkan berkurang menjadi 90 detik, menunjukkan peningkatan respons sistem yang lebih cepat. Terakhir, pada pengujian dengan Kp = 0.26, sistem hanya membutuhkan 70 detik untuk mencapai nilai set point, yang menunjukkan bahwa semakin besar nilai Kp, semakin cepat sistem merespons perubahan input.

Adapun grafik respons gabungan dari variasi pengujian parameter yang telah diuji dapat memberikan gambaran yang lebih komprehensif mengenai kinerja sistem dengan berbagai nilai Kp. Grafik ini menggambarkan bagaimana setiap perubahan pada parameter-parameter tersebut mempengaruhi rise time dan overshoot yang bisa dilihat pada Gambar 37 dibawah ini.



Gambar 37. Grafik Respons Pengujian Kp Bervariasi

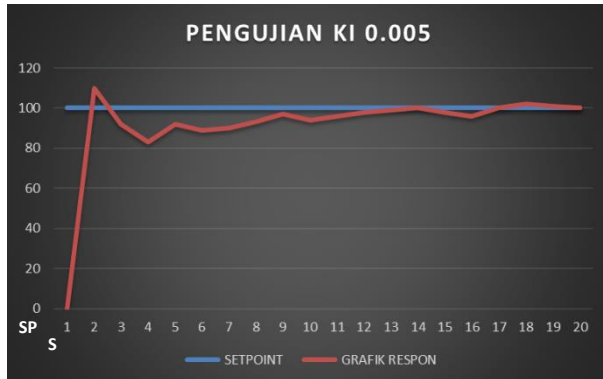
Berdasarkan Gambar 37, dapat disimpulkan bahwa nilai $K_p = 0.26$ memberikan respons terbaik, dengan rise time yang cepat dan tanpa adanya overshoot dibandingkan dengan nilai $K_p = 0.20$ dan $K_p = 0.15$. Pada nilai $K_p = 0.26$, sistem hanya memerlukan waktu sekitar 70 detik untuk mencapai posisi set point yang diinginkan, yaitu 100 RPM, yang menunjukkan respons yang lebih efisien dan stabil. Sedangkan pada $K_p = 0.20$, sistem membutuhkan waktu 90 detik, dan pada $K_p = 0.15$, sistem memerlukan waktu lebih lama, yaitu 130 detik.

Dari hasil ini, dapat disimpulkan bahwa $K_p = 0.26$ adalah nilai yang paling optimal, karena memberikan rise time tercepat tanpa overshoot, sehingga dipilih sebagai nilai K_p yang akan dilanjutkan untuk tuning parameter K_i dan K_d pada tahap selanjutnya menggunakan metode Trial and Error.

4.4.1.3 Pengujian Parameter K_i

Setelah melakukan pengujian terhadap parameter K_p , langkah berikutnya dalam tuning PID adalah menganalisis pengaruh dari parameter K_i (gain integral). Pada pengujian pertama untuk parameter K_i , penulis memulai dengan menggunakan nilai $K_p = 0.26$, yang sebelumnya telah diperoleh dari pengujian parameter K_p pada Gambar 37. Selanjutnya, penulis melakukan pengujian dengan menetapkan nilai $K_i = 0.005$, sementara K_d tetap pada nilai 0.0, yang merupakan pengaturan default untuk parameter derivative. Pengujian ini bertujuan untuk menganalisis pengaruh nilai K_i terhadap respons sistem, dengan K_p yang telah dioptimalkan sebelumnya. Dengan demikian, penulis ingin mengevaluasi bagaimana penambahan integral gain dapat mempengaruhi kestabilan dan akurasi sistem dalam mencapai set point yang diinginkan, yaitu 100 RPM.

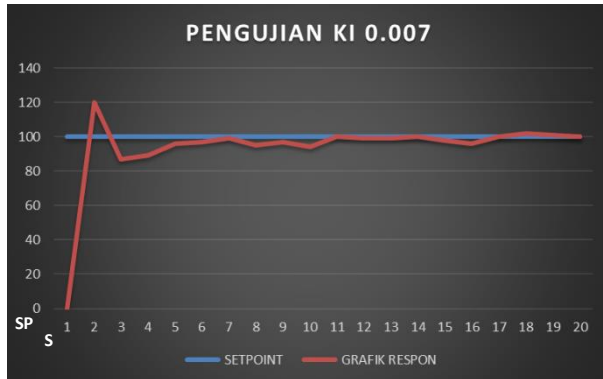
Adapun hasil dan pembahasan jika parameter K_i 0.005 dengan setpoint 100 RPM yang bisa dilihat pada Gambar 38 dibawah ini.



Gambar 38. Pengujian Parameter KI 0.005

Berdasarkan yang terlihat pada Gambar 38, dengan nilai $K_i = 0.005$, sistem menunjukkan adanya overshoot hingga 10 RPM, yang menunjukkan bahwa sistem belum sepenuhnya stabil walau pengurangan risetime sudah baik dari sebelumnya yaitu tanpa pengendali K_i . Selain itu, sistem juga mengalami fluktuasi yang cukup signifikan, dengan adanya overshoot dan undershoot yang cukup besar, serta ketidakmampuan untuk mempertahankan nilai set point 100 RPM secara konstan. Hal ini menunjukkan bahwa nilai K_i yang digunakan terlalu kecil atau tidak optimal, sehingga respons sistem masih belum cukup baik dalam mencapai kestabilan dan akurasi yang diinginkan. Dengan hasil ini, penulis melanjutkan pengujian dengan nilai K_i yang lebih tinggi dan melihat respon selanjutnya.

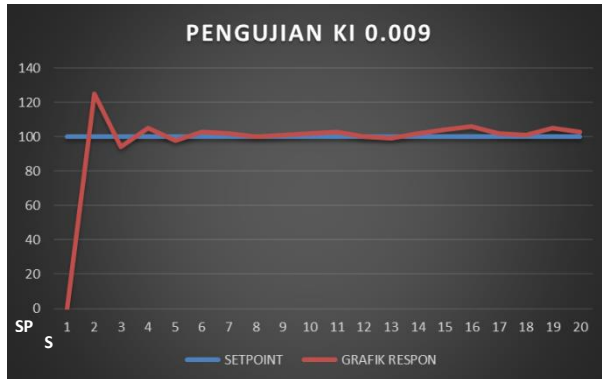
Pada pengujian kedua, penulis melakukan pengujian dengan menaikkan nilai $K_i = 0.007$ dengan kondisi $K_P = 0.26$ dan $K_D = 0.0$ yang merupakan nilai *default* dari sistem yang digunakan. Berikut adalah hasil pengujian yang bisa dilihat pada Gambar 39 dibawah ini.



Gambar 39. Pengujian Parameter KI 0.007

Berdasarkan hasil pengujian yang terlihat pada Gambar 39, dengan nilai $K_i = 0.007$, sistem menunjukkan adanya overshoot hingga 20 RPM, yang menandakan bahwa sistem masih belum sepenuhnya stabil meskipun rise time telah lebih baik dibandingkan nilai K_i sebelumnya. Selain itu, sistem masih mengalami fluktuasi, dengan adanya overshoot dan undershoot yang tidak terlalu besar, serta ketidakmampuan untuk mempertahankan set point 100 RPM secara konstan. Hal ini mengindikasikan bahwa nilai K_i yang digunakan sudah menunjukkan suatu respon yang mendekati walau belum secara keseluruhan, sehingga respons sistem belum mencapai kestabilan dan akurasi yang diinginkan. Berdasarkan hasil ini, penulis melanjutkan pengujian dengan mencoba nilai K_i yang lebih tinggi untuk melihat perubahan respons sistem selanjutnya.

Pada pengujian ketiga, penulis meningkatkan nilai K_i menjadi 0.009, dengan kondisi $K_p = 0.26$ dan $K_d = 0.0$, yang merupakan nilai default dari sistem yang digunakan. Berikut adalah hasil pengujian yang bisa dilihat pada Gambar 40 dibawah ini.



Gambar 40. Pengujian Parameter KI 0.009

Berdasarkan hasil pengujian yang terlihat pada Gambar 40, dengan nilai $K_i = 0.009$, sistem menunjukkan adanya overshoot hingga 25 RPM, yang menandakan bahwa sistem masih belum sepenuhnya stabil meskipun rise time telah lebih baik dibandingkan kondisi tanpa pengendali K_i . Selain itu, nilai undershoot mulai berkurang, dan rise time semakin cepat. Hal ini menunjukkan bahwa dengan peningkatan nilai K_i , respons sistem mulai membaik. Namun, sistem masih belum mampu mempertahankan set point 100 RPM secara konstan. Oleh karena itu, untuk meningkatkan kestabilan sistem dan mengurangi fluktuasi lebih lanjut, penulis memutuskan untuk menambahkan pengendali K_d guna berfungsi sebagai stabilisator dan mengurangi ketidakstabilan yang tersisa.

Setelah mengetahui dari 3 kondisi parameter pada pengujian nilai K_i , dapat menghasilkan beberapa keterangan yang dituliskan pada Tabel 13 dibawah ini.

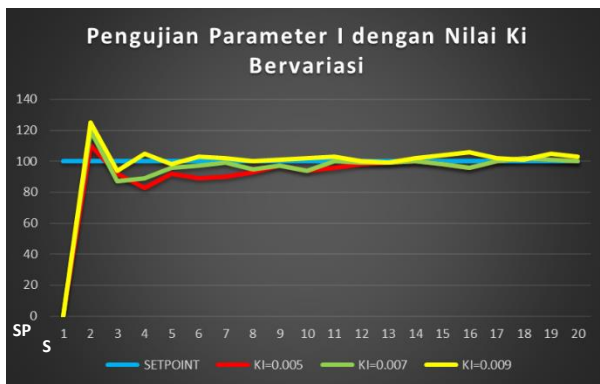
Tabel 13. Hasil Pengujian Parameter KI Bervariasi

Parameter PID			Rise Time	Overshoot
K_p	K_i	K_d	(s)	(%)
0.26	0.005	0.0	2.2	10
0.26	0.007	0.0	2.1	20
0.26	0.009	0.0	1.9	25

Berdasarkan Tabel 13. Hasil Pengujian Parameter KI Bervariasi, dapat disimpulkan bahwa sistem mengalami peningkatan overshoot seiring dengan peningkatan nilai K_i . Pada $K_i = 0.005$, sistem memiliki rise time 2.2 detik dan overshoot 10%, yang menunjukkan respons yang relatif cepat namun masih ada fluktuasi yang cukup besar. Dengan $K_i = 0.007$, rise time sedikit lebih cepat (2.1

detik), namun overshoot meningkat menjadi 20%, yang menunjukkan adanya kecenderungan sistem menjadi lebih responsif namun kurang stabil. Pada $K_i = 0.009$, rise time semakin cepat (1.9 detik) dan overshoot meningkat lagi menjadi 25%, menunjukkan bahwa meskipun respons sistem semakin cepat, kestabilan menurun dan overshoot menjadi lebih signifikan.

Grafik respons gabungan dari berbagai variasi pengujian parameter yang telah dilakukan memberikan gambaran yang lebih menyeluruh tentang kinerja sistem dengan nilai K_i yang berbeda. Grafik ini menunjukkan bagaimana perubahan pada parameter-parameter tersebut mempengaruhi rise time, overshoot, undershoot, dan osilasi yang bisa dilihat pada Gambar 41 dibawah ini.



Gambar 41. Grafik Respon Pengujian K_i Bervariasi

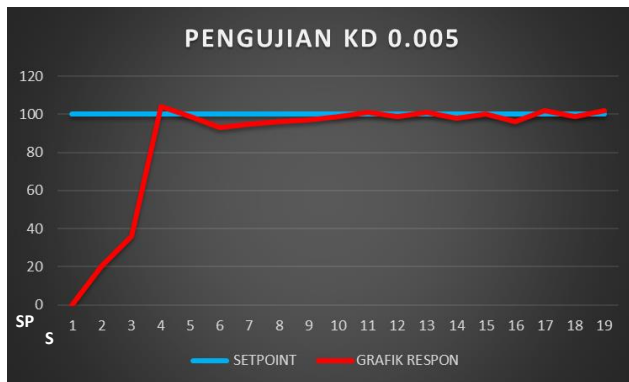
Berdasarkan Gambar 41, pengujian yang terlihat pada gambar, dapat disimpulkan bahwa semakin tinggi nilai K_i , semakin cepat waktu rise time. Pada $K_i = 0.005$, sistem membutuhkan waktu 6 detik untuk mencapai set point, sedangkan pada $K_i = 0.007$, waktu rise time berkurang menjadi 4 detik, dan pada $K_i = 0.009$, waktu rise time semakin cepat menjadi 2 detik. Hal ini menunjukkan bahwa peningkatan K_i mampu mempercepat respons sistem dalam mencapai set point. Selain itu, pada $K_i = 0.009$, sistem mampu dengan cepat menyeimbangkan nilai error terhadap set point yang diinginkan, sebagaimana terlihat pada grafik. Meskipun sempat mengalami undershoot, nilainya relatif kecil, dan sistem segera naik kembali menuju set point. Namun, meskipun rise time semakin cepat, sistem masih mengalami fluktuasi dan belum stabil secara konstan. Hal ini terjadi karena nilai K_i yang lebih tinggi memperbesar pengaruh kesalahan kumulatif, menyebabkan sistem lebih agresif dalam melakukan koreksi, tetapi berisiko meningkatkan overshoot, undershoot, serta osilasi.

Berdasarkan hasil grafik tersebut, $K_i = 0.009$ dipilih untuk digunakan dalam sistem selanjutnya, karena memberikan rise time tercepat dengan respons yang lebih sigap terhadap perubahan error. Namun, karena sistem masih mengalami osilasi, langkah berikutnya adalah melakukan tuning pada parameter K_d untuk meredam fluktuasi dan meningkatkan kestabilan sistem secara keseluruhan.

4.4.1.4 Pengujian Parameter K_d

Pada pengujian setelah menentukan nilai K_p dan K_i yang optimal, langkah selanjutnya adalah melakukan pengujian terhadap parameter K_d (derivative gain). Parameter K_d berperan dalam meredam osilasi yang terjadi akibat efek integral serta membantu meningkatkan stabilitas sistem dengan mengurangi overshoot dan undershoot. Dengan adanya K_d , sistem diharapkan dapat mencapai set point dengan lebih halus dan stabil tanpa fluktuasi yang berlebihan. Nilai K_d akan divariasikan sementara nilai $K_p = 0.26$ dan $K_i = 0.009$ tetap digunakan berdasarkan hasil tuning sebelumnya. Pengujian dilakukan untuk melihat bagaimana pengaruh K_d terhadap respons sistem, terutama dalam mengurangi osilasi dan meningkatkan kestabilan pada set point 100 RPM.

Adapun hasil dan pembahasan jika parameter K_d 0.005 dengan setpoint 100 RPM yang bisa dilihat pada Gambar 42 dibawah ini.

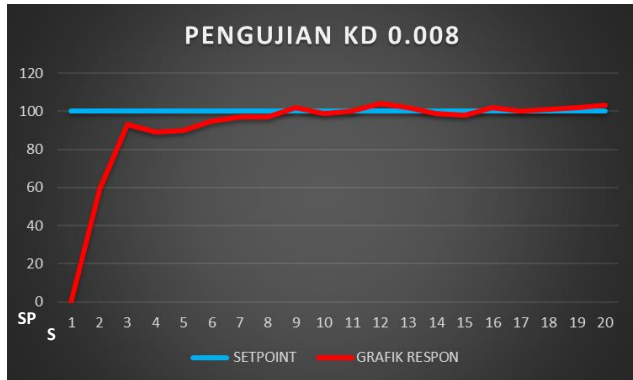


Gambar 42. Pengujian Parameter K_d 0.005

Berdasarkan Gambar 42, dengan $K_d = 0.005$ sistem masih mengalami undershoot lumayan, sekitar 10 RPM setelah rise time. Meskipun sistem mampu menyesuaikan diri dengan set point secara cepat, osilasi yang terjadi masih cukup signifikan, menyebabkan fluktuasi yang cukup sering. Oleh karena itu, pada kondisi ini, sistem belum dapat dikategorikan sebagai akurat dan stabil, sehingga penulis

melakukan peningkatan nilai K_d untuk meredam osilasi lebih lanjut pada pengujian selanjutnya

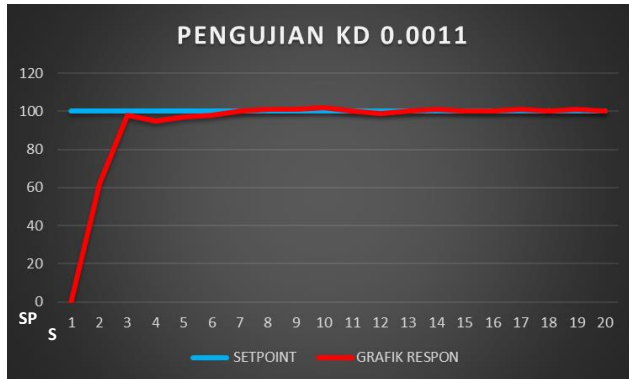
Pada pengujian kedua, penulis menaikkan nilai K_d menjadi 0.008, dengan mempertahankan $K_p = 0.26$ dan $K_i = 0.009$, yang merupakan nilai hasil tuning dari pengujian sebelumnya pada sistem yang digunakan. Berikut adalah hasil pengujian yang bisa dilihat pada Gambar 43 dibawah ini.



Gambar 43. Pengujian Parameter K_D 0.008

Berdasarkan Gambar 43, dengan $K_d = 0.008$, sistem tidak mengalami overshoot yang signifikan dan mampu menyesuaikan diri dengan set point. Namun, osilasi yang terjadi masih belum sepenuhnya teredam dengan baik, dengan fluktuasi sekitar 3-5 RPM dari set point yang diinginkan. Oleh karena itu, penulis melanjutkan pengujian dengan meningkatkan nilai K_d lebih lanjut untuk mencari parameter yang dapat meminimalkan osilasi hingga berada dalam toleransi 1-2 RPM saja.

Pada pengujian ketiga, penulis meningkatkan nilai K_d menjadi 0.011, sementara $K_p = 0.26$ dan $K_i = 0.009$ tetap dipertahankan sebagai hasil tuning dari pengujian sebelumnya pada sistem yang digunakan. Berikut adalah hasil pengujian yang bisa dilihat pada Gambar 44 dibawah ini.



Gambar 44. Pengujian Parameter KD 0.0011

Berdasarkan Gambar 44, dengan $K_d = 0.011$ sistem menunjukkan respons yang lebih baik dalam menyesuaikan diri dengan set point. Tidak terdapat overshoot maupun undershoot yang signifikan, menandakan peningkatan kestabilan sistem. Selain itu, osilasi yang terjadi jauh lebih minimal dibandingkan pengujian sebelumnya, sehingga steady-state error dapat terdefiniskan dengan jelas, yaitu berada dalam kisaran 1 hingga 2 RPM saja. Hasil ini menunjukkan bahwa nilai $K_d = 0.011$ mampu meredam osilasi dengan baik dan meningkatkan akurasi sistem dalam mencapai set point.

Setelah mengetahui dari 3 kondisi parameter pada pengujian nilai K_d , dapat menghasilkan beberapa keterangan yang dituliskan pada Tabel 14 dibawah ini.

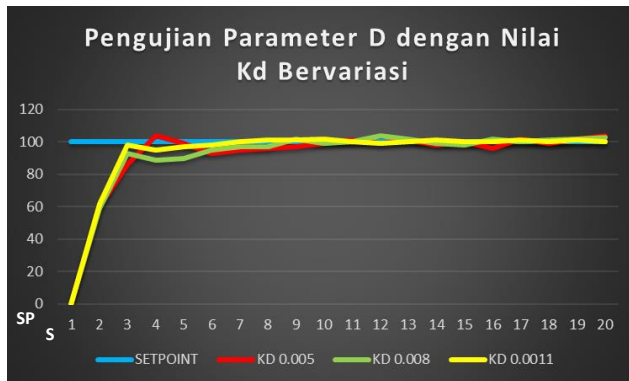
Tabel 14. Hasil Pengujian Parameter KD Bervariasi

Parameter PID			Rise Time	Overshoot	Error Steady State
Kp	Ki	Kd	(s)	(%)	
0.26	0.009	0.005	2.2	5	5
0.26	0.009	0.008	2.1	0	3
0.26	0.009	0.0011	1.8	0	1

Berdasarkan Tabel 14 dengan berbagai nilai parameter PID, dapat disimpulkan bahwa sistem menunjukkan perbaikan seiring peningkatan nilai K_d . Pada $K_d = 0.005$, sistem memiliki rise time 2.2 detik, overshoot 5%, dan steady-state error 5 RPM, yang menunjukkan kestabilan yang kurang optimal. Dengan $K_d = 0.008$, rise time sedikit lebih cepat (2.1 detik) dan overshoot berhasil dihilangkan, namun steady-state error masih 3 RPM. Pada $K_d = 0.011$, sistem mencapai rise time terbaik (1.8 detik), tanpa overshoot, dan steady-state error hanya 1 RPM, yang

menunjukkan kinerja terbaik dalam hal kecepatan respons, kestabilan, dan akurasi.

Setelah melakukan pengujian dengan variasi parameter K_p dan K_i , pengujian dengan variasi nilai K_d memberikan wawasan lebih lanjut tentang pengaruhnya terhadap kinerja sistem. Grafik respons gabungan dari pengujian K_d menunjukkan bagaimana perubahan pada nilai K_d mempengaruhi rise time, overshoot, dan osilasi sistem, serta bagaimana K_d berperan dalam mengurangi fluktuasi dan meningkatkan kestabilan sistem secara keseluruhan yang bisa dilihat pada Gambar 45 dibawah ini.



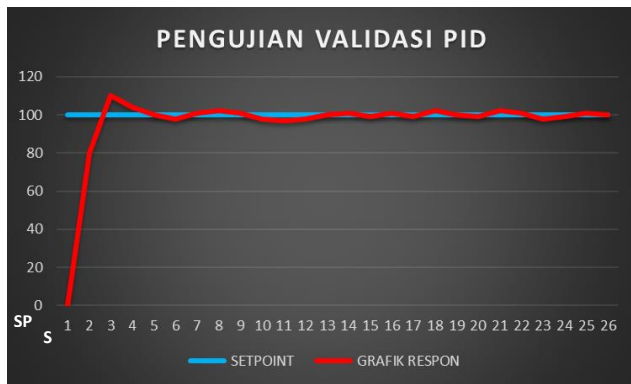
Gambar 45. Grafik Respon Pengujian KD Bervariasi

Berdasarkan Gambar 45, dengan berbagai nilai K_d dapat disimpulkan bahwa sistem mengalami perbaikan seiring dengan peningkatan nilai K_d . Pada $K_d = 0.005$, sistem masih mengalami undershoot sekitar 10 RPM setelah rise time, dengan osilasi yang cukup signifikan dan fluktuasi yang sering, sehingga sistem belum dapat dikategorikan stabil dan akurat. Pada $K_d = 0.008$, meskipun tidak ada overshoot yang signifikan, osilasi masih terjadi dengan fluktuasi sekitar 3-5 RPM dari set point, yang menunjukkan bahwa kestabilan sistem belum optimal. Namun, pada $K_d = 0.011$, sistem menunjukkan respons yang jauh lebih baik, tanpa overshoot maupun undershoot yang signifikan, dan osilasi dapat diminimalkan hingga berada dalam kisaran 1-2 RPM. Hasil ini menunjukkan bahwa nilai $K_d = 0.011$ berhasil meredam osilasi dan meningkatkan akurasi sistem, sehingga penulis memilih nilai tersebut sebagai parameter terbaik untuk mencapai kinerja sistem yang stabil dan akurat.

4.4.1.5 Pengujian Validasi Parameter Keseluruhan

Sebelum melakukan pengujian terhadap set point yang diinginkan, seperti low, mid, dan high RPM, saya terlebih dahulu melakukan validasi terhadap parameter $K_p = 0.26$, $K_i = 0.009$, dan $K_d = 0.0011$ yang telah diperoleh. Validasi ini bertujuan untuk memastikan bahwa kombinasi parameter PID yang telah dioptimalkan dapat menghasilkan kinerja sistem yang akurat dan stabil, serta mampu menjaga kestabilan respons sistem pada berbagai kondisi set point yang berbeda.

Adapun hasil pengujian validasi pada kondisi setpoint 100 RPM yang bisa dilihat pada Gambar 46 dibawah ini.



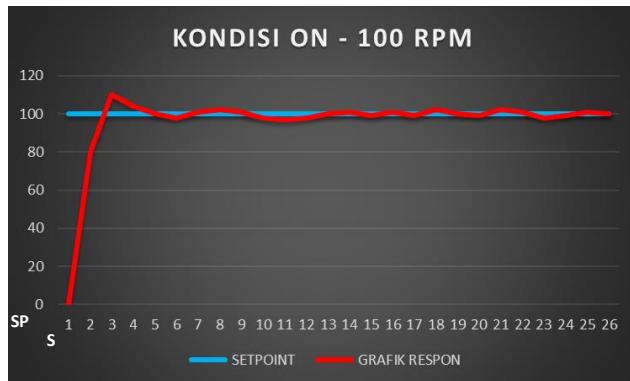
Gambar 46. Grafik Respon Pengujian Validasi PID

Berdasarkan Gambar 46, validasi PID yang dilakukan melalui metode trial and error, didapatkan nilai-nilai RPM yang menunjukkan respons sistem yang cukup baik. Meskipun terdapat overshoot sekitar 10 RPM, sistem berhasil menunjukkan respons yang cepat dengan rise time hanya 1.5 detik dan stabil tanpa adanya undershoot atau osilasi yang signifikan. Hal ini mengindikasikan bahwa parameter PID yang telah dioptimalkan mampu menjaga kestabilan sistem dan memastikan bahwa sistem dapat beradaptasi dengan cepat terhadap perubahan set point, meskipun sedikit fluktuasi masih terjadi. Secara keseluruhan, hasil pengujian menunjukkan kinerja yang memadai dan sistem dapat dikatakan stabil dan akurat.

4.4.2 Pengujian terhadap Setpoint dengan *PID Controller*

Pada pembahasan ini, penulis melakukan pengujian terhadap set point dengan menggunakan PID controller yang telah dituning berdasarkan nilai-nilai K_p , K_i , dan K_d yang diperoleh dari pengujian sebelumnya. Nilai parameter PID yang digunakan adalah $K_p = 0.26$, $K_i = 0.009$, dan $K_d = 0.0011$. Pengujian ini bertujuan untuk mengevaluasi kinerja sistem pada berbagai kondisi set point, yaitu pada kecepatan rendah (100 RPM), sedang (200 RPM), dan tinggi (400 RPM serta 600 RPM), untuk memastikan bahwa sistem dapat mempertahankan kestabilan dan respons yang akurat pada berbagai kecepatan yang berbeda. Pengujian ini dilakukan pada motor DC 12V dengan encoder yang digunakan untuk menggerakkan mini konveyor.

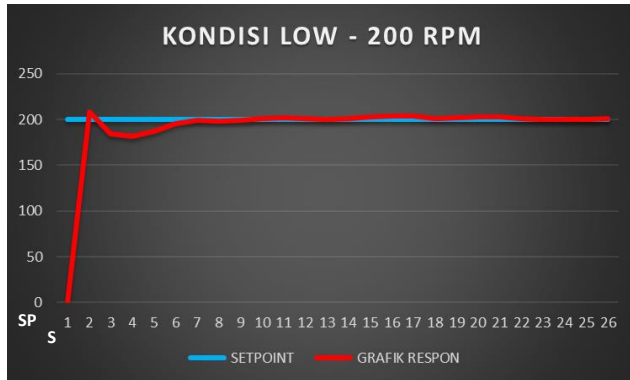
Adapun hasil yang didapatkan dari masing masing setpoint setelah menerapkan kendali PID pada sistem, dapat dilihat pada Gambar 47 adalah hasil pengujian pada kondisi on 100 RPM.



Gambar 47. Grafik Respons Setpoint 100 dengan PID

Berdasarkan Gambar 47, grafik respons menunjukkan bahwa sistem mencapai kestabilan setelah terjadinya overshoot yang melebihi 110 pada awal perubahan kecepatan. Setelah overshoot tersebut, sistem mampu menyesuaikan diri dengan cepat. Dengan PID controller yang diterapkan, umpan balik berupa error berhasil dikoreksi, dan set point yang diinginkan, yaitu 100 RPM, dapat tercapai. Rise time yang tercatat adalah 2 detik, dengan steady-state error yang minimal, yaitu hanya 2 RPM, menandakan respons yang cepat dan akurat dalam penyesuaian kecepatan.

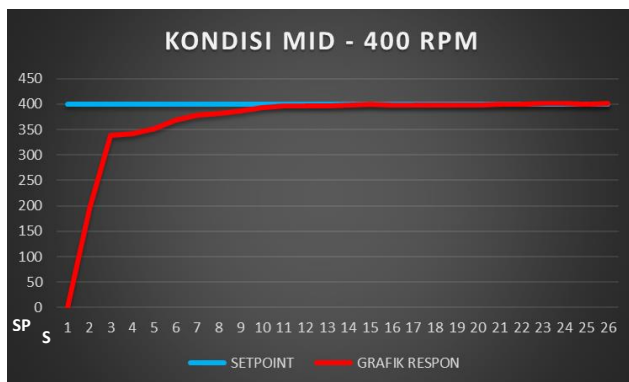
Selanjutnya hasil dan pembahasan berikut yaitu kondisi Low dengan setpoint 200 RPM yang bisa dilihat pada Gambar 48 dibawah ini.



Gambar 48. Grafik Respon *Setpoint* 200 dengan PID

Berdasarkan Gambar 48, grafik respons menunjukkan bahwa sistem mencapai kestabilan setelah terjadi overshoot yang cukup signifikan pada awal perubahan kecepatan, sekitar 8-10 RPM. Setelah overshoot tersebut, sistem sempat mendapatkan undershoot namun berhasil menyesuaikan diri, dan PID controller mampu merespons umpan balik berupa error untuk mencapai setpoint yang diinginkan, yaitu 200 RPM.

Selanjutnya hasil dan pembahasan yaitu pada kondisi Mid dengan setpoint 400 RPM yang bisa dilihat pada Gambar 49 dibawah ini.

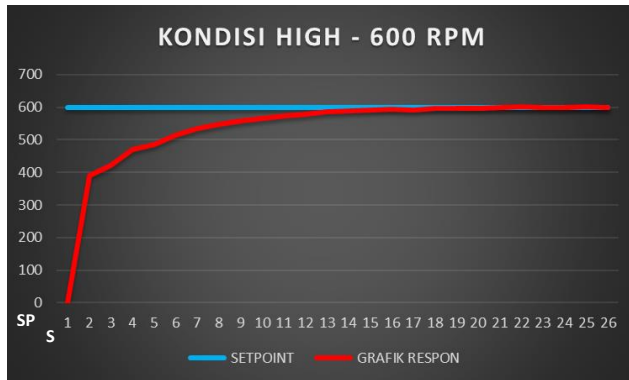


Gambar 49. Grafik Respon *Setpoint* 400 dengan PID

Berdasarkan Gambar 49, grafik respons menunjukkan bahwa tidak terjadi overshoot, yang menandakan bahwa sistem mampu menyesuaikan diri dengan baik meskipun rise time memakan waktu hingga sekitar 12 detik. Meskipun waktu

penyesuaian cukup lama, sistem tetap dapat digolongkan sebagai responsif, karena tidak terjadi osilasi yang berarti. Hal ini menunjukkan bahwa sistem stabil dan dapat mencapai setpoint yang diinginkan dengan akurat, dengan steady-state error yang minimal, yaitu hanya 2 RPM. Kinerja sistem ini mencerminkan kemampuan PID controller untuk menjaga kestabilan meskipun membutuhkan waktu lebih lama dalam proses penyesuaian.

Selanjutnya hasil dan pembahasan yaitu pada kondisi High dengan setpoint 600 RPM yang bisa dilihat pada Gambar 50 dibawah ini.



Gambar 50. Grafik Respon Setpoint 600 dengan PID

Berdasarkan Gambar 50, grafik respons menunjukkan bahwa tidak terjadi overshoot, yang mengindikasikan bahwa sistem dapat menyesuaikan diri dengan baik meskipun *rise time* memakan waktu hingga sekitar 20 detik. Walaupun proses penyesuaian memerlukan waktu yang cukup lama, sistem tetap dapat dikategorikan sebagai responsif karena tidak terjadi osilasi yang signifikan. Hal ini mengindikasikan kestabilan sistem yang baik dalam mencapai *setpoint* 600 RPM, dengan *steady-state error* yang sangat kecil, yaitu hanya 1 RPM. Kinerja ini mencerminkan efektivitas *PID controller* dalam menjaga kestabilan meskipun proses penyesuaian membutuhkan waktu lebih lama.

Berdasarkan hasil pengujian yang dilakukan pada setpoint 100 RPM, 200 RPM, dan 600 RPM, dapat disimpulkan bahwa sistem yang menggunakan PID controller dengan parameter yang telah dioptimalkan menunjukkan kinerja yang baik dalam berbagai kondisi kecepatan. Pada *setpoint* 100 RPM, meskipun terjadi overshoot yang cukup besar pada awal perubahan kecepatan, sistem mampu menyesuaikan diri dengan cepat, dengan *rise time* 2 detik dan *steady-state error* yang hanya 2 RPM, menunjukkan respons yang cepat dan akurat. Pada setpoint 200 RPM, terjadi *overshoot* sekitar 8-10 RPM diikuti dengan sedikit *undershoot*, namun

sistem berhasil menyesuaikan diri dan mencapai setpoint yang diinginkan. Hal ini menunjukkan bahwa *PID controller* mampu mengoreksi error dan menjaga kestabilan meskipun ada fluktuasi kecil. Pada setpoint 600 RPM, sistem tidak mengalami overshoot dan meskipun *rise time* mencapai 20 detik, sistem tetap responsif dan stabil. Tidak adanya osilasi yang signifikan serta *steady-state error* yang sangat kecil (1 RPM) menunjukkan efektivitas *PID controller* dalam menjaga kestabilan, meskipun membutuhkan waktu lebih lama dalam proses penyesuaian. Secara keseluruhan, hasil pengujian ini menunjukkan bahwa *PID controller* yang telah dioptimalkan dengan parameter yang tepat mampu menjaga kestabilan sistem pada berbagai setpoint dan memberikan respons yang akurat dan stabil tanpa adanya osilasi yang berarti.

Adapun nilai hasil pengujian *PID* terhadap sistem yang digunakan oleh penulis sebagai berikut dituliskan dalam Tabel 15 dibawah ini.

Tabel 15. Hasil Pengujian *PID* Terhadap Sistem

<i>Setpoint</i>	<i>Rise Time (s)</i>	<i>Settling Time (s)</i>	<i>Overshoot (%)</i>	<i>Steady State Error</i>
100	2 detik	5 detik	10 %	10
200	2,2 detik	8 detik	8 %	8
400	2,4 detik	12 detik	0 %	3
600	2,5 detik	20 detik	0 %	1

Berdasarkan analisis performa sistem dengan berbagai nilai setpoint (100, 200, 400, dan 600) pada Tabel 15, dapat disimpulkan bahwa semakin tinggi nilai setpoint, semakin lama waktu yang dibutuhkan untuk mencapai 90% dari setpoint (*rise time*) dan semakin lama waktu untuk stabil (*settling time*). Meskipun demikian, overshoot tetap terkendali, dan *steady-state error* tetap kecil, menunjukkan akurasi yang baik. Selain itu pengujian ini juga dilakukan untuk mengetahui nilai rata rata error pada tiap kondisi dengan tidak menggunakan kontrol *PID* dan tentu menggunakan kontrol *PID*. Adapun rumusan yang penulis gunakan untuk mengetahuinya adalah sebagai berikut :

$$\text{Error} = \frac{\sum |\text{Setpoint} - \text{Pembacaan}|}{n} \quad (4)$$

di mana *n* adalah jumlah pembacaan.

Sehingga hasil yang didapatkan dalam perhitungan error dapat dituliskan dalam Tabel 16 dibawah ini.

Tabel 16. Hasil Perbandingan Pengujian tanpa Kontrol PID dan dengan Kontrol PID

Kondisi / Setpoint	TANPA PID	DENGAN PID
ON – 100	25.74 %	0.85 %
LOW - 200	37.78 %	1.04 %
MID - 400	20.89 %	0.92 %
HIGH - 600	41.67 %	0.94 %
RATA RATA	31.52 %	0.94 %

Dari analisis yang dilakukan terhadap sistem kontrol dengan dan tanpa penggunaan PID pada Tabel 16, dapat disimpulkan bahwa penerapan kontrol PID memberikan peningkatan yang sangat signifikan dalam hal akurasi dan stabilitas sistem dalam mencapai setpoint yang diinginkan. Pada pengujian tanpa PID, sistem menunjukkan rata-rata steady-state error yang cukup besar, yaitu sekitar 31.52%, yang mengindikasikan ketidakakuratan serta fluktuasi besar dalam mencapai nilai setpoint yang diharapkan. Kondisi ini menunjukkan bahwa tanpa kendali yang optimal, sistem cenderung mengalami kesalahan yang tinggi, yang dapat berdampak pada kinerja keseluruhan, terutama dalam aplikasi yang memerlukan presisi tinggi.

Namun, setelah penerapan kontrol PID dengan parameter yang telah dioptimalkan, yaitu $K_p = 0.26$, $K_i = 0.009$, dan $K_d = 0.0011$, steady-state error rata-rata turun drastis menjadi hanya 0.94%. Pengurangan error ini menunjukkan bahwa sistem menjadi lebih responsif dan mampu mencapai setpoint dengan lebih cepat serta minim overshoot. Selain itu, fluktuasi yang sebelumnya besar berhasil diredam secara efektif, sehingga kestabilan sistem meningkat secara signifikan.

Secara keseluruhan, hasil analisis ini menunjukkan bahwa kontrol PID merupakan pendekatan yang sangat efektif dalam meningkatkan kinerja sistem kontrol. Dengan kemampuannya dalam mengurangi error, meningkatkan kestabilan, serta memberikan respons yang lebih halus dan presisi, PID menjadi pilihan yang sangat baik untuk berbagai aplikasi yang membutuhkan kontrol presisi tinggi, terutama dalam pengendalian kecepatan motor DC dalam sistem otomatisasi industri.

Bab 5. Kesimpulan dan Saran

5.1. Kesimpulan

Berdasarkan pengujian dan analisis yang telah dilakukan, Tugas Akhir ini menghasilkan beberapa simpulan utama yang mencerminkan keberhasilan implementasi sistem kendali motor DC berbasis ESP32 dengan fitur kontrol PID, validasi keamanan, dan kemampuan kontrol jarak jauh. Adapun kesimpulan yang dapat diambil adalah sebagai berikut :

1. Keandalan Sensor

Sistem ini berhasil menggunakan sensor untuk membaca parameter penting seperti arus, tegangan, dan kecepatan putar (RPM) motor DC. Hasil pengujian menunjukkan bahwa akurasi pembacaan sensor hampir mendekati hasil pengukuran manual menggunakan alat ukur standar yang sudah terkalibrasi, seperti multimeter dan tachometer. Hal ini membuktikan bahwa sensor yang digunakan dapat diandalkan untuk mendukung kendali sistem.

2. Validasi Keamanan Melalui GUI (Interface)

GUI yang dirancang memiliki kemampuan untuk memvalidasi pengguna melalui menu login yang terhubung dengan database MySQL. Fitur ini memberikan proteksi keamanan pada sistem, sehingga hanya pengguna yang terdaftar dapat mengakses dan mengontrol perangkat. Selain itu, pengujian terhadap fitur kontrol, monitoring, dan manajemen pengguna juga menunjukkan hasil yang optimal.

3. Kontrol Jarak Jauh dengan Protokol ESP-NOW

Sistem kontrol jarak jauh berbasis protokol komunikasi ESP-NOW dapat diimplementasikan dengan baik untuk mengirim data antara antarmuka pengguna dan sistem kontrol. Meskipun demikian, jangkauan komunikasi terbatas pada jarak tertentu, terutama di lingkungan indoor dengan hambatan fisik dibandingkan dengan lingkungan outdoor yang lebih terbuka.

4. Efektivitas Kendali PID pada Plant (Motor DC)

Penerapan pengendali PID terbukti meningkatkan stabilitas dan akurasi kinerja motor DC. Hal ini dibuktikan melalui perbandingan antara kondisi sistem tanpa pengendali PID, yang cenderung tidak stabil, dengan sistem yang telah diimplementasikan kendali PID. Hasil pengujian menunjukkan bahwa penerapan kontrol PID secara signifikan meningkatkan akurasi dan stabilitas dalam mencapai setpoint yang diinginkan. Rata-rata error tanpa PID mencapai sekitar 31.52%, sementara dengan penggunaan PID, rata-rata error

turun drastis menjadi hanya 0.94%. Parameter kontrol yang digunakan dalam pengaturan PID adalah $K_p = 0.26$, $K_i = 0.009$, dan $K_d = 0.0011$

5. Keterbatasan Jarak Fitur Voice Control

Penggunaan fitur voice control, yang memanfaatkan modul mikrofon internal perangkat untuk menerima perintah suara, memiliki keterbatasan jarak operasional. Hal ini disebabkan oleh sensitivitas mikrofon yang menurun pada jarak tertentu, sehingga memengaruhi keakuratan pengenalan suara.

5.2. Saran

Penulis menyadari bahwa pengembangan sistem secara keseluruhan masih diperlukan untuk meningkatkan kinerja dan efisiensi sistem di masa depan. Oleh karena itu, penulis memberikan beberapa saran yang dapat menjadi acuan untuk penelitian dan pengembangan lebih lanjut. Adapun saran-saran tersebut adalah sebagai berikut:

1. Optimalisasi Sistem Kendali

Untuk meningkatkan kinerja sistem dan mencapai hasil yang lebih optimal, terdapat beberapa rekomendasi yang dapat diimplementasikan dalam penelitian selanjutnya. Pertama, pengembangan penggunaan metode PID (Proportional-Integral-Derivative) dapat dilakukan dengan mengeksplorasi atau menggabungkan metode pengendalian lainnya untuk menghasilkan respons yang lebih presisi dan adaptif terhadap perubahan kondisi sistem. Selain itu, metode PID yang telah digunakan diharapkan dapat diaplikasikan pada perangkat atau alat lain yang relevan, sehingga penggunaannya dapat lebih luas dan bermanfaat dalam berbagai konteks. Selanjutnya, dalam pencarian respon optimal PID, parameter K_p (gain proporsional), K_i (gain integral), dan K_d (gain derivatif) dapat dikembangkan lebih lanjut melalui metode tuning yang lebih canggih, seperti algoritma genetika, optimasi swarm, atau metode eksperimen lainnya. Pengembangan ini diharapkan mampu meningkatkan akurasi dan efisiensi pengendalian, sekaligus memberikan kontribusi yang signifikan terhadap kemajuan teknologi sistem kontrol.

2. Optimalisasi Protokol Komunikasi untuk Lingkungan Indoor

Penggunaan protokol ESP-NOW dalam ruangan dengan banyak hambatan fisik terbukti kurang efisien karena jangkauan komunikasi yang terbatas. Sebagai solusi, disarankan untuk mengganti protokol komunikasi dengan metode yang lebih sesuai untuk lingkungan indoor, seperti komunikasi serial. Protokol serial dapat memberikan stabilitas yang lebih baik serta

meminimalkan gangguan yang disebabkan oleh penghalang fisik, sehingga mendukung komunikasi data yang lebih andal dan konsisten.

3. Peningkatan Kinerja Voice Control dengan Mikrofon Eksternal

Untuk meningkatkan kinerja fitur voice control, disarankan penggunaan mikrofon eksternal dengan tingkat sensitivitas yang lebih tinggi. Mikrofon semacam ini dapat menangkap suara dengan lebih jelas, bahkan ketika pengguna berada pada jarak yang lebih jauh dari perangkat. Selain itu, penerapan teknologi noise-canceling juga perlu dipertimbangkan untuk mengurangi dampak gangguan suara latar belakang, terutama pada lingkungan dengan tingkat kebisingan tinggi. Dengan implementasi ini, akurasi pengenalan suara dapat ditingkatkan secara signifikan, sehingga fitur voice control lebih responsif dan handal.

Pengoptimalan ini diharapkan tidak hanya meningkatkan efisiensi dan akurasi sistem, tetapi juga memastikan bahwa perangkat dapat beroperasi dengan baik dalam berbagai kondisi lingkungan, baik di dalam ruangan maupun pada situasi dengan tingkat kebisingan tinggi.

Daftar Pustaka

- [1] D. C. Permana, R. Ferdiansyah, F. P. Safira, Z. T. A. Gumilang, A. J. Pangestu, and R. W. Abdul Rozak, "Otomasi Industri Sebuah Peluang Atau Ancaman," *J. Pengabd. Masy. Pemberdayaan, Inov. dan Perubahan*, vol. 3, no. 3, pp. 139–146, 2023, doi: 10.59818/jpm.v3i3.515.
- [2] D. Junaedi, F. Redjeki, and M. D. Priadi, "Pengaruh Promosi dan Kualitas Layanan Terhadap Keputusan Pembelian: studi kasus pada Koperasi Mitra Dhuafa Cabang Mande-Cianjur," *Indones. J. Econ. Bus.*, vol. 1, no. 2, pp. 106–120, 2023, doi: 10.58818/ijeb.v1i2.31.
- [3] Y. Syaifudin and H. Hendriko, "Mesin Insert PIN Dan Pellet Otomatis Berbasis PLC," *J. Indones. Sos. Teknol.*, vol. 4, no. 7, pp. 855–864, 2023, doi: 10.59141/jist.v4i7.649.
- [4] H. P. Batam, "Kunjungan Kerja Direksi PT SIER ke PT Persero Batam," perserobatam. [Online]. Available: <https://perserobatam.com/kunjungan-kerja-direksi-pt-sier-ke-pt-persero-batam/>
- [5] Superadmin, "Jasa Fabrikasi Conveyor Di Batam," PT. Multi Karya Teknik. [Online]. Available: <https://multikaryatehnik.co.id/jasa-fabrikasi-conveyor-di-batam/>
- [6] PT. Environeer, "4 Keuntungan Menggunakan Conveyor," PT. Environeer. [Online]. Available: <https://www.connectautomation.co.id/blog/4-keuntungan-menggunakan-conveyor/>
- [7] Liputan6, "Conveyor Belt Adalah: Panduan Lengkap Sistem Transportasi Material Industri," *Liputan6*, p. 1, 2021. [Online]. Available: <https://www.liputan6.com/feeds/read/5774851/conveyor-belt-adalah-panduan-lengkap-sistem-transportasi-material-industri?page=12>
- [8] PuskoMedia Indonesia, "Mengapa Desain Antarmuka Pengguna Suara (VUI) Penting dalam Era Perangkat Cerdas," p. 1, 2024, [Online]. Available: https://www.puskomedia.id/blog/mengapa-desain-antarmuka-pengguna-suara-vui-penting-dalam-era-perangkat-cerdas/?utm_source=chatgpt.com
- [9] F. Irawan, A. B. Neris, R. A. Marlina, T. Pertambangan, and S. Tinggi Teknologi Industri Padang, "Analisis Produktivitas Belt Conveyor Di Tunnel Main Shaft Pt. Allied Indo Coal Jaya (Aicj) Pambahan Kec. Talawi, Kota Sawahlunto, Sumatera Barat," *J. Sains dan Teknol.*, 2020.
- [10] B. Dhiya' Ushofa, L. Anifah, G. Buditjahjanto, and Endryansyah, "Sistem Kendali Kecepatan Putaran Motor DC pada Conveyor dengan Metode Kontrol PID," *J. Tek. Elektro*, vol. 11, no. Universitas Negeri Surabaya, pp. 332–342, 2022.
- [11] W. WALUYO, A. FITRIANSYAH, and S. SYAHRIAL, "Analisis Penalaan

- Kontrol PID pada Simulasi Kendali Kecepatan Putaran Motor DC Berbeban menggunakan Metode Heuristik,” *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 1, no. 2, p. 79, 2013, doi: 10.26760/elkomika.v1i2.79.
- [12] A. Pane, J., Surya, A., Novita, S., Mazmur, R., Aryza, A., Hamdani, Rizky, “Implementasi PID Dalam Mengendalikan Motor Menggunakan Metode PID dan Mikrokontroler Atmega,” *Sainteks*, vol. 1, no. 1, pp. 196–201, 2019, [Online]. Available: <https://seminar-id.com/prosiding/index.php/sainteks/article/download/155/153>
- [13] K. P. Mentor, “Perancangan Sistem Pengendali Kecepatan Motor DC Menggunakan Kontroler Proportional Integral Derivative Pada Palang Pintu Parkir,” *J. Tek. Elektro*, vol. 12, p. 48, 2023.
- [14] M. M. I. Putra, S. R. U. A. Sompie, and S. Paturusi, “Implementasi Speech Recognition pada Aplikasi Pembelajaran Bahasa Inggris untuk Anak,” *J. Tek. Inform.*, vol. 15, no. 4, pp. 247–256, 2020, [Online]. Available: <https://ejournal.unsrat.ac.id/index.php/informatika/article/view/30426>
- [15] S. STEKOM, “C sharp (bahasa pemograman),” *Ensiklopedia*. Accessed: Mar. 12, 2024. [Online]. Available: [https://p2k.stekom.ac.id/ensiklopedia/C_Sharp_\(bahasa_pemrograman\)](https://p2k.stekom.ac.id/ensiklopedia/C_Sharp_(bahasa_pemrograman))
- [16] M. R. A. Nurkholis Putera and R. Hidayat, “Kendali Kecepatan Motor DC Menggunakan Pengendali PID dengan Encoder sebagai Feedback,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 7, no. 1, p. 50, 2022, doi: 10.30998/string.v7i1.13026.
- [17] R. Birdayansyah, N. Sudjarwanto, and O. Zebua, “Pengendalian Kecepatan Motor DC Menggunakan Perintah Suara Berbasis Mikrokontroler Arduino,” *Electr. – J. Rekayasa dan Teknol. Elektro*, vol. 9, no. 2, pp. 97–107, 2015.
- [18] M. A. Ulum and S. I. Haryudo, “Perancangan Sistem Monitoring Kecepatan Putar Motor DC Berbasis Internet Of Things Menggunakan Aplikasi BLYNK,” *J. Tek. Elektro*, vol. 9, no. 1, pp. 855–862, 2020.
- [19] A. W. Wicesa, “MEMBANGUN APLIKASI E-LIBRARY MENGGUNAKAN HTML, PHP SCRIPT, DAN MYSQL DATABASE Rini Sovia dan Jimmy Febio,” *Processor*, vol. 6, no. 2, pp. 38–54, 2011.
- [20] M. Diah Ika Putri, A. Ma’arif, and R. Dwi Puriyanto, “Pengendali Kecepatan Sudut Motor DC Menggunakan Kontrol PID dan Tuning Ziegler Nichols,” *Techno (Jurnal Fak. Tek. Univ. Muhammadiyah Purwokerto)*, vol. 23, no. 1, 2022, doi: 10.30595/techno.v23i1.10773.
- [21] F. Adnan, I. Amelia, and S. ‘Umar Shiddiq, “Implementasi Voice Recognition Berbasis Machine Learning,” *Implementasi Voice Recognit. Berbas. Mach. Learn.*, vol. 11, no. 1, pp. 24–29, 2022.
- [22] S. Hadi, A. S. Anas, and L. G. R. Putra, “Rancang Bangun Sistem Monitoring

- Penggunaan Daya Listrik Berbasis Internet of Things,” *Circuit J. Ilm. Pendidik. Tek. Elektro*, vol. 6, no. 1, p. 54, 2022, doi: 10.22373/crc.v6i1.10862.
- [23] A. Wagyana, “Prototipe Modul Praktik untuk Pengembangan Aplikasi Internet of Things (IoT),” *Setrum Sist. Kendali-Tenaga-elektronika-telekomunikasi-komputer*, vol. 8, no. 2, p. 238, 2019, doi: 10.36055/setrum.v8i2.6561.
- [24] R. Aulia Nabih, A. Setia Budi, and R. Primananda, “Sistem Pemanggil Pelayan LCLE (Low Cost Low Energy) Berbasis ESP-NOW,” *Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 1, pp. 342–354, 2023, [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/download/12184/5546>
- [25] Z. Lubis *et al.*, “Kontrol Mesin Air Otomatis Berbasis Arduino Dengan Smartphone,” *Cetak) Bul. Utama Tek.*, vol. 14, no. 3, pp. 1410–4520, 2019.
- [26] E. A. Prasetyo, “Driver Motor L298N,” ARDUINOINDONESIA. Accessed: Mar. 12, 2024. [Online]. Available: <https://www.arduinoindonesia.id/2022/10/driver-motor-l298n.html>

Biodata



Nama : Muhammad Zidni Aufa
TTL : Batam, 20 September 2002
Agama : Islam
Alamat : Pondok Graha blok L No 09 Tj Piayu,
Batam, Kepulauan Riau.
Email : akuziddni@gmail.com
Riwayat Pendidikan :
SMA/SMK : SMAN 16 BATAM
SMP : SMPN 16 BATAM



Nama : Wahyu Firdaus
TTL : Bukit tinggi, 20 Oktober 2002
Agama : Islam
Alamat : Pasar Shopping Centre Blok C3 No. 03
Email : Wahyufirdaus2010@gmail.com
Riwayat Pendidikan :
SMA/SMK : SMKN 1 SOLOK SELATAN
SMP : MTSN 1 SOLOK SELATAN

LAMPIRAN

FORMULIR LOGBOOK BIMBINGAN DAN PENGAJUAN SIDANG TUGAS AKHIR

Nama : Muhammad Zidni Aufa
NIM : 4212101088
Pembimbing I : Adlian Jefiza, S.Pd., M.T.
Pembimbing II* : -
Judul : Sistem Kontrol Konveyor dengan PID menggunakan Perintah Suara dan Monitoring berbasis Visual C#

No	Hari/Tgl	Rincian Kegiatan	TTD Pembimbing I
1	28/11/2024	Melakukan Bimbingan terkait bab 3 pada sub bab Pengujian	
2	9/12/2024	Melakukan Bimbingan terkait hasil revisi bab 3 pada sub bab Pengujian	
3	11/12/2024	Melakukan Bimbingan terkait bab 4 tentang hasil pengujian	
4	17/12/2024	Melakukan Bimbingan terkait bab 4 tentang teknis pengambilan data	
5	19/12/2024	Melakukan Bimbingan terkait bab 4 tentang kendala saat pengujian	
6	23/12/2024	Melakukan Bimbingan terkait bab 4 tentang menganalisis tiap parameter	
7	28/12/2024	Melakukan Bimbingan terkait bab 4 terkait revisi penulisan hasil dan analisis	
8	29/12/2024	Melakukan Bimbingan terkait bab 4 terkait revisi penulisan hasil dan analisis	
9	2/01/2025	Melakukan Bimbingan terkait hasil dan revisi bab 5 tentang Kesimpulan dan saran	
10	3/01/2025	Melakukan Bimbingan terkait keseluruhan Buku TA	

Berdasarkan hasil bimbingan yang telah dilaksanakan selama 2 bulan lebih dan telah disetujui oleh dosen pembimbing, maka dengan ini saya mengajukan diri sebagai peserta Sidang Tugas Akhir.

Batam, 03/01/2025
Muhammad Zidni Aufa



NIM: 4212101088

**FORMULIR LOGBOOK BIMBINGAN DAN PENGAJUAN
SIDANG TUGAS AKHIR**

Nama : Wahyu Firdaus
 NIM : 4212101082
 Pembimbing I : Adlian Jefiza, S.Pd., M.T.
 Pembimbing II* : -
 Judul : Sistem Kontrol Konveyor dengan PID menggunakan Perintah Suara dan Monitoring berbasis Visual C#

No	Hari/Tgl	Rincian Kegiatan	TTD Pembimbing I
1	28/11/2024	Melakukan Bimbingan terkait bab 3 pada sub bab Pengujian	
2	9/12/2024	Melakukan Bimbingan terkait hasil revisi bab 3 pada sub bab Pengujian	
3	11/12/2024	Melakukan Bimbingan terkait bab 4 tentang hasil pengujian	
4	17/12/2024	Melakukan Bimbingan terkait bab 4 tentang teknis pengambilan data	
5	19/12/2024	Melakukan Bimbingan terkait bab 4 tentang kendala saat pengujian	
6	23/12/2024	Melakukan Bimbingan terkait bab 4 tentang penganalisan tiap parameter	
7	28/12/2024	Melakukan Bimbingan terkait bab 4 terkait revisi penulisan hasil dan analisis	
8	29/12/2024	Melakukan Bimbingan terkait bab 4 terkait revisi penulisan hasil dan analisis	
9	2/01/2025	Melakukan Bimbingan terkait hasil dan revisi bab 5 tentang Kesimpulan dan saran	
10	3/01/2025	Melakukan Bimbingan terkait keseluruhan Buku TA	

Berdasarkan hasil bimbingan yang telah dilaksanakan selama 2 bulan lebih dan telah disetujui oleh dosen pembimbing, maka dengan ini saya mengajukan diri sebagai peserta Sidang Tugas Akhir.

Batam, 03/01/2025
 Wahyu Firdaus



NIM: 4212101082