

Implementasi Kendali PID pada Kecepatan Motor DC Sebagai Media Pembelajaran Berbasis Arduino dan LabVIEW

Implementation of PID Control on DC Motor Speed as Learning Media Based on Arduino and LabVIEW

Fauzan Azim¹, Aulia Ullah^{2*}, Jufrizel³, dan Ahmad Faizal⁴

^{1,2,3,4}*Program Studi SI Teknik Elektro, Fakultas Sains dan Teknologi,
Universitas Islam Negeri Sultan Syarif Kasim Riau
Jl. HR. Soebrantas Km. 15, Pekanbaru, Indonesia*

*.²Corresponding author: aulia.ullah@uinsuska.ac.id

Received on 23-05-2024, accepted on 01-07-2024, published on 29-07-2024

Abstrak

Pemahaman konsep kendali PID merupakan elemen penting dalam elektronika instrumentasi, terutama di sektor industri. Pembelajaran yang masih menggunakan metode simulasi, tentu hasilnya tidak sepenuhnya mencerminkan kondisi sebenarnya. Penelitian ini bertujuan untuk mengembangkan alat media pembelajaran sistem kendali PID praktis dengan tampilan antarmuka yang inovatif dan informatif untuk meningkatkan pemahaman mahasiswa di bidang teknik kendali. Alat ini menggunakan Arduino Uno sebagai perangkat akuisisi data dan perangkat lunak LabVIEW sebagai antarmuka. Sensor IR Optocoupler LM393, sensor tegangan, dan sensor ACS712 digunakan untuk membaca kecepatan putaran, tegangan, dan arus motor DC. Hasil pengujian menunjukkan bahwa sistem tanpa kendali menghasilkan kesalahan sebesar 16% sedangkan sistem dengan kendali PID menghasilkan kesalahan sebesar 4.4%. Selanjutnya, hasil pengujian terhadap gangguan kendali PID sistem mampu menjaga kestabilannya dengan baik dibandingkan tanpa kendali. Melalui uji evaluasi *usability* dengan mahasiswa sebagai responden menunjukkan nilai rata-rata sebesar 92,3 %. Jadi dengan demikian, alat ini dapat menjadi solusi efektif bagi mahasiswa untuk mempelajari konsep kendali PID secara langsung.

Kata kunci: Arduino, Kendali PID, LabVIEW, Media Pembelajaran

Abstract

Understanding the concept of PID control is a crucial element in instrumentation electronics, especially in the industrial sector. Learning that still relies on simulation methods does not fully reflect actual conditions. This study aims to develop a practical learning tool for PID control systems with an innovative and informative interface to enhance students' understanding of control engineering. The tool uses Arduino Uno for data acquisition and LabVIEW software as the interface. The IR Optocoupler LM393 sensor, voltage sensor, and ACS712 sensor are used to measure the rotational speed, voltage, and current of the DC motor. Testing results show that a system without control produces an error of 16%, while a system with PID control produces an error of 4.4%. Additionally, the PID control system maintains stability better than the system without control when subjected to disturbances. Usability evaluation tests with students as respondents showed an average score of 92.3%. Thus, this tool can be an effective solution for students to learn the concept of PID control directly.

Keywords: Arduino, PID Control, LabVIEW, Learning Media

I. PENDAHULUAN

Motor DC (*Direct Current*) adalah perangkat elektromekanis yang berfungsi mengubah energi listrik arus searah menjadi energi mekanis. Prinsip kerjanya bergantung pada suplai arus searah pada kumparan medan. Motor DC akan berputar ketika diberikan tegangan dengan polaritas tertentu pada kedua terminalnya, sedangkan kecepatan putaran ditentukan oleh besar tegangan yang diberikan pada kedua terminal [1].

Motor DC telah banyak dimanfaatkan dalam berbagai bidang kendali karena memiliki sejumlah keunggulan. Beberapa keunggulan tersebut meliputi karakteristik torsi yang kuat, kinerja respon yang tinggi, kemudahan dalam pengendalian secara linear, serta kemampuan untuk mempertahankan kestabilan pada nilai referensi yang ditentukan [2][3]. Secara umum, ada dua variabel utama yang dikendalikan pada motor DC, yaitu posisi sudut yang diukur dalam satuan derajat dan kecepatan sudut yang diukur dalam satuan RPM (*Revolutions Per Minute*) atau RPS (*Revolutions Per Second*) [4]. Sehingga motor DC dapat dijadikan sebagai bidang untuk mempelajari algoritma kendali tingkat lanjut, karena konsepnya dapat diproyeksikan pada jenis motor lain [5].

Untuk menjaga kecepatan motor DC berada pada nilai referensi, dibutuhkan metode pengendalian yang sesuai salah satunya adalah kendali PID (*Proportional-Integral-Derivative*) [3]. Di sektor industri, kendali PID masih mendominasi, dengan tingkat penggunaan mencapai 90 % [6]. Hal ini dikarenakan kendali PID memiliki kelebihan mudah dipahami, mudah diimplementasikan, dan mampu memberikan stabilitas sistem yang baik dari sistem non-linear hingga sistem linear, selain itu kendali ini mampu mengoreksi kesalahan suatu sistem dan memberikan respon cepat dengan tingkat kesalahan yang kecil [7]. Penyetelan parameter PID yang tepat, dapat meningkatkan efisiensi kendali. Metode yang sering digunakan adalah *trial and error*. Namun, hasil yang diperoleh tentu belum optimal meskipun kinerjanya yang dihasilkan baik [8].

Berdasarkan hasil observasi dan wawancara terhadap dosen pengampu mata kuliah sistem kendali di program studi Teknik Elektro, UIN Suska Riau. Pada salah satu mata kuliah sistem kendali membahas tentang pengaplikasian sistem kendali *Proportional Integral Derivative* (PID) di industri. Namun, materi tersebut masih disajikan dalam bentuk simulasi menggunakan perangkat lunak seperti Matlab, sehingga hasil yang diperoleh tidak sepenuhnya mencerminkan kondisi sebenarnya [9]. Dengan adanya media pembelajaran yang lebih menarik, tentu dapat meningkatkan komunikasi antara mahasiswa dengan dosen, sehingga materi pembelajaran tersampaikan dengan lebih efektif dan efisien [10][11]. Di Laboratorium Elektronika Instrumentasi, terdapat alat praktikum sistem kendali yang digunakan memiliki harga operasional yang cukup tinggi seperti *analog and digital motor control teaching set*. Selain itu, dibutuhkan waktu merakit setiap modul agar dapat digunakan, hal ini tentu hanya akan merugikan dari segi waktu dan kualitas pembelajaran. Untuk meningkatkan pengalaman pembelajaran baru kepada mahasiswa dalam memahami teori sistem kendali, dibutuhkan suatu media pembelajaran yang praktis dengan tampilan antarmuka yang inovatif dan informatif.

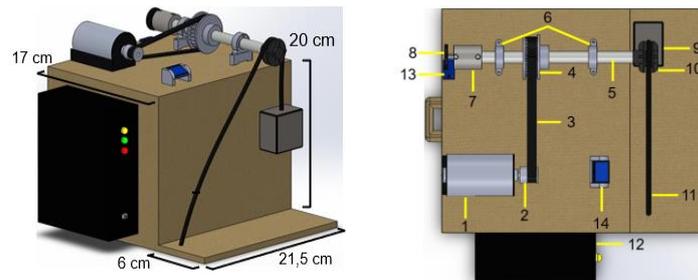
Terdapat beberapa penelitian sebelumnya yang membuat alat sebagai media pembelajaran sistem kendali PID, seperti penelitian yang dilakukan oleh [12], tahun 2020, penelitian ini menerapkan kendali PID pada motor DC sebagai media pembelajaran menggunakan LabVIEW sebagai antarmuka dan Arduino Uno sebagai perangkat akuisisi data. Perubahan parameter kendali PID dapat dilakukan melalui tampilan antarmuka LabVIEW secara langsung. Lalu [13], tahun 2022, peneliti ini telah mengembangkan alat peraga kendali PID pada motor DC menggunakan LabVIEW sebagai antarmuka dan NI-MyRIO sebagai perangkat akuisisi data. Penelitian ini menyajikan pengaruh dari nilai parameter PID terhadap perubahan *setpoint* dan pengujian kestabilan sistem ketika menerima gangguan yang baik. Namun, perangkat NI MyRIO memiliki harga yang cukup tinggi. kemudian, [9], tahun 2023, membuat implementasi kendali PID pada *brightness* cahaya lampu sebagai modul pembelajaran praktikum menggunakan Arduino Uno sebagai perangkat akuisisi data. Namun, parameter PID masih dilakukan melalui perubahan program dan tidak disertai dengan tampilan antarmuka di dalamnya.

Berdasarkan uraian diatas, dirancang alat media pembelajaran praktis untuk memahami konsep sistem kendali PID menggunakan Arduino Uno sebagai perangkat akuisisi data dan perangkat lunak LabVIEW sebagai tampilan antarmuka dan tempat membangun algoritma kendali PID. Untuk pemantauan lebih lanjut dari pengendalian kecepatan motor DC, peneliti menambahkan sensor arus dan tegangan untuk memantau kondisi pada motor DC saat beroperasi. Melalui antarmuka LabVIEW, perubahan nilai *setpoint* dan parameter PID dapat dilakukan secara *real-time*. Desain mekanik ditempatkan di atas kotak kayu untuk memudahkan dalam pemberian gangguan berupa beban gantung. Garis bantu pada grafik RPM ditambahkan untuk membantu analisis respon sistem, yang dapat diaktifkan atau dinonaktifkan secara manual. Untuk mengetahui kualitas media pembelajaran sistem kendali PID yang dibuat, peneliti menambahkan uji evaluasi terhadap mahasiswa/i sebagai responden menggunakan kuesioner.

II. METODE PENELITIAN

A. Desain Sistem

Desain sistem ini dirancang untuk menggambarkan penempatan perangkat keras dan mekanisme kerja motor DC berbeban. Gambar 1 menunjukkan ukuran alat yang dirancang agar lebih portabel, sehingga mudah dibawa dan lebih praktis digunakan dalam berbagai situasi pembelajaran. Penempatan ini dimaksudkan untuk memudahkan dalam melakukan perbaikan jika terjadi masalah pada sistem. Dengan desain seperti ini, pemeliharaan perangkat menjadi lebih efisien dan praktis. Untuk meningkatkan kekuatan torsi pada motor DC maka kecepatan motor DC harus dikurangi dengan menggunakan rasio pada pulley. Sehingga memungkinkan motor DC untuk dapat menahan beban hingga 1 kg.



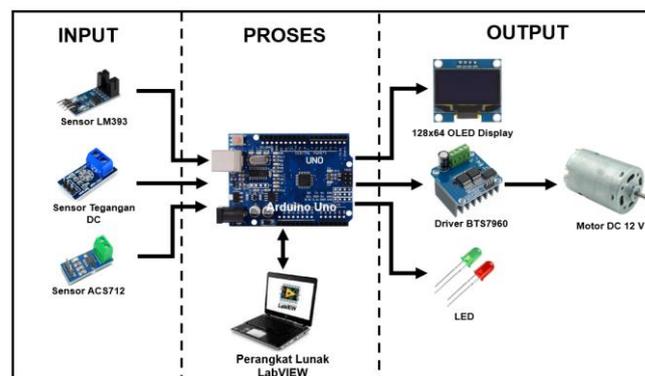
Gambar 1. Desain Tiga Dimensi (3D) Ukuran Alat Media Pembelajaran Sistem Kendali PID

Keterangan gambar 1:

- | | |
|---|--------------------------|
| 1. Motor DC RS545 21V | 8. Encoder disk 5 lubang |
| 2. Timing Pulley GT2 20 teeth Bore | 9. Beban |
| 3. Timing belt 6 mm | 10. Pembatas tali beba |
| 4. Timing Pulley GT2 60 teeth Bore | 11. Tali beban 320 mm |
| 5. Batang As Shaft Stainless Steel 304 8 mm | 12. Box perangkat sistem |
| 6. Pillow Block KP08 8 mm | 13. Sensor LM393 |
| 7. Flexible Coupling D19 L25 5x8 mm | 14. 128x64 OLED Display |

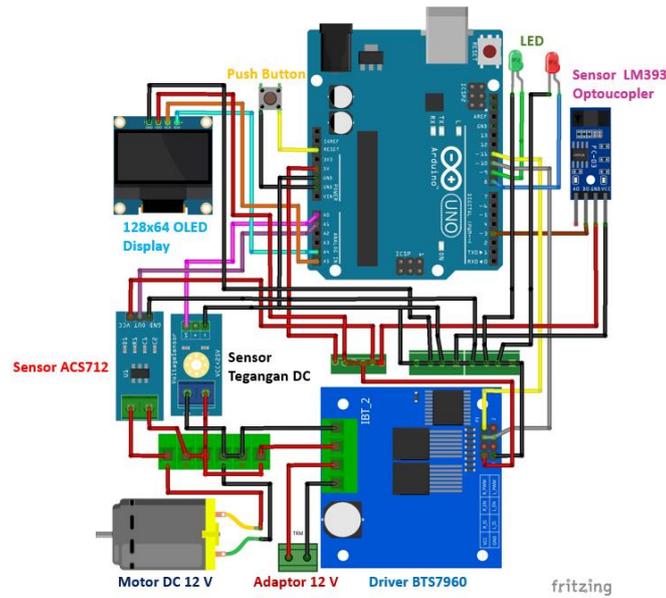
B. Perancangan Perangkat Keras (*Hardware*)

Dalam perancangan perangkat keras sistem, terdapat tiga proses utama yaitu *input*, proses, dan *output*, yang masing-masing memiliki peran penting dalam mempresentasikan cara kerja keseluruhan sistem dan hubungan antar perangkat. Perancangan sistem alat dapat dilihat pada Gambar 2 dibawah ini.



Gambar 2. Perancangan Sistem Alat

Berdasarkan Gambar 2, Data *input* berasal dari sensor LM393, sensor tegangan DC, dan sensor ACS712. Data *input*, akan diproses pada Mikrokontroler Arduino Uno dan perangkat lunak LabVIEW. Sedangkan *Output* sistem berupa sinyal PWM akan diterima oleh driver BTS7960 untuk menggerakkan motor DC. Serta LED dan OLED display digunakan sebagai lampu indikator dan untuk menampilkan hasil dari pengolahan sistem secara langsung.



Gambar 3. Skema Rangkaian Perangkat Keras

Gambar 3 diatas adalah skema rangkaian yang didesain menggunakan perangkat lunak fritzing. Skema ini memudahkan pengkoneksian antar komponen agar sistem dapat berfungsi dengan baik. Untuk menghubungkan setiap komponen digunakan kabel jumper yang memiliki konektor atau pin pada kedua ujungnya. Kabel jumper memiliki jenis yaitu *male* sebagai ujung yang menusuk dan *female* sebagai ujung yang penghubung [14]. Berikut adalah fungsi dari masing-masing komponen pada rangkaian diatas:

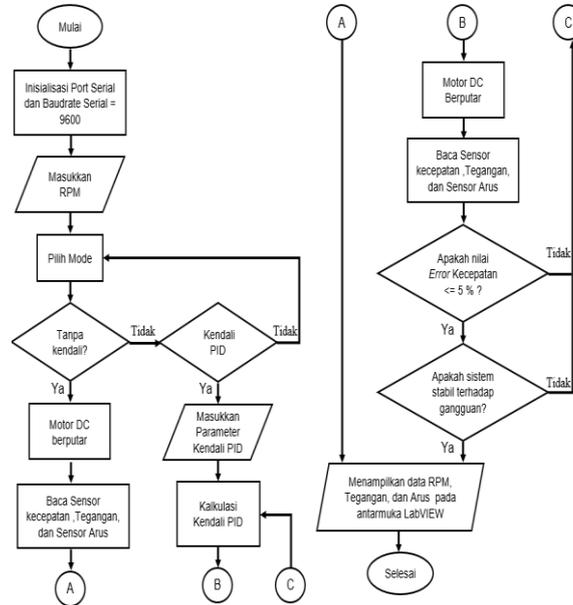
1. Sensor LM393 berfungsi untuk membaca kecepatan putaran motor DC. Sensor bekerja jika LED pada sensor akan menyala sebagai tanda ada benda yang melewati celah pada sensor. Pemancar IR akan menangkap pembacaan dari objek yang kemudian dikirimkan ke penerima IR [15].
2. Sensor tegangan DC 0-25 volt berfungsi untuk mengukur tegangan yang masuk ke motor DC dengan prinsip pembagi tegangan menggunakan dua resistor 30 K Ω dan 7,5K Ω [16].
3. Sensor ACS712 menggunakan prinsip *Hall Effect* untuk mengukur arus motor DC hingga 5 A. Sensor ini bekerja dengan mengubah medan magnet dari arus yang mengalir menjadi tegangan linear yang sebanding dengan tegangan keluar sebesar 185 mV per 1 A yang melewati sensor [17].
4. Arduino Uno berfungsi sebagai prosesor utama dalam proses akuisisi data.
5. Motor DC RS 545 21 V berfungsi sebagai penggerak pada sistem untuk mendapatkan variabel proses.
6. Driver BTS7960 berfungsi sebagai pengendali kecepatan motor DC berdasarkan nilai sinyal PWM dari Arduino Uno. Modul driver ini mampu beroperasi pada tegangan 27V dengan arus 43A [18].
7. 128x64 OLED Display berfungsi sebagai penampil hasil pengukuran sensor kecepatan, sensor tegangan, dan sensor arus selama sistem berjalan.
8. Adaptor 12 volt 5 Ampere berfungsi sebagai tegangan sumber untuk menggerakkan motor DC melalui driver BTS7960.
9. LED 5mm berfungsi sebagai indikator sistem
10. Push Button berfungsi sebagai tombol reset program.

C. Perancangan Perangkat Lunak (*Software*)

1. *Flowchart Sistem*

Dalam perancangan sistem, *flowchart* atau diagram alir berperan dalam memudahkan pemahaman proses kerja sistem. Pada diagram alir tersebut, data sensor-sensor yang telah diproses pada Arduino Uno akan dikirimkan ke perangkat lunak LabVIEW. Pada Gambar 4, hal pertama yang dilakukan adalah menentukan Port Serial dan Baudrate Serial yang sesuai, sehingga perangkat lunak LabVIEW dapat berkomunikasi dengan Arduino Uno secara efektif. Proses berikutnya adalah menentukan nilai *setpoint* RPM, diikuti dengan pemilihan mode sistem pengendalian kecepatan motor DC, baik tanpa kendali maupun menggunakan kendali PID. Jika mode tanpa kendali dipilih, motor DC yang berputar hanya menampilkan nilai sensor kecepatan, tegangan, dan arus pada antarmuka LabVIEW. Jika mode kendali PID yang dipilih,

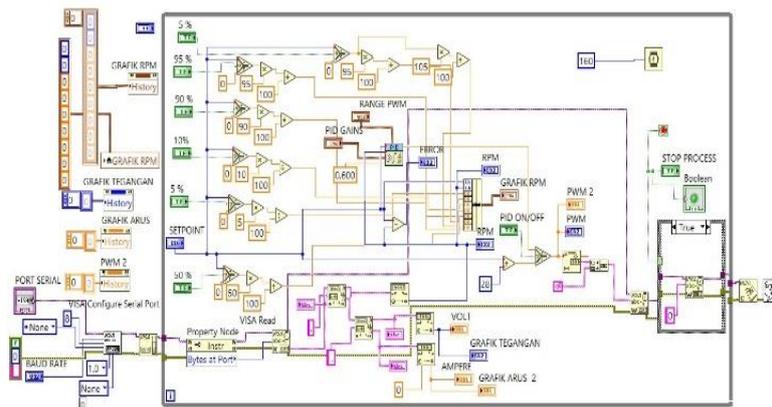
parameter kendali PID harus ditentukan terlebih dahulu. Ketika motor DC berputar, hasil pembacaan sensor kecepatan kemudian diproses kembali oleh algoritma kendali PID untuk memastikan bahwa nilai *error* antara hasil pembacaan sensor kecepatan dan *setpoint* kurang dari 5%. Ketika terdapat gangguan, kendali PID akan memperbaiki *error* yang terjadi hingga sistem kembali ke keadaan stabil. Semua data sensor yang diproses akan ditampilkan pada antarmuka LabVIEW dalam bentuk grafik dan numerik.



Gambar 4. Flowchart Alur Sistem

2. Perancangan Program LabVIEW

Dalam penelitian ini, perangkat lunak yang digunakan adalah LabVIEW 2020 64 bit. LabVIEW menggunakan model pemrograman berbasis aliran data dari setiap suatu blok diagram ke blok diagram lain. Blok diagram program LabVIEW dapat dilihat pada Gambar 5 dibawah ini.

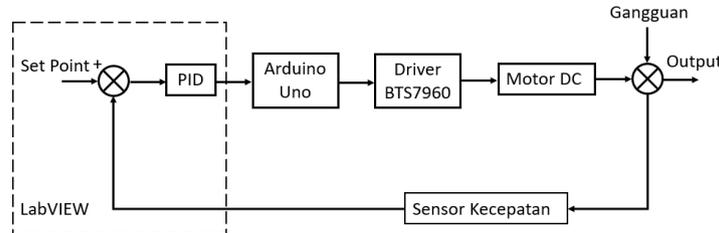


Gambar 5. Tampilan Blok Diagram LabVIEW

Program ini beroperasi secara berulang sesuai dengan waktu *looping* yang telah ditetapkan. digunakan *Virtual Instrument Software Architecture (VISA) Serial* sebagai interkoneksi antara LabVIEW dengan mikrokontroler Arduino Uno [19]. Hasil dari program ini berupa *Graphical User Interface (GUI)*, yang merupakan tampilan antarmuka grafis antara pengguna dengan komputer. Pada GUI semua parameter *input* dapat diubah, sehingga memberikan kesempatan kepada mahasiswa agar bisa melakukan pengujian nilai berbeda [20].

3. Perancangan Kendali PID

Pada penelitian ini, sistem kendali PID dirancang berdasarkan pada *toolkit* perangkat lunak LabVIEW. Setiap parameter kendali PID memiliki kelebihan masing-masing seperti Kendali *proportional* mampu mempercepat *rise time*, kendali *integral* mampu memperkecil nilai *error*, dan kendali *derivative* mampu memperkecil *error* atau meredam *overshoot/undershoot*. Dengan menggabungkan ketiga parameter ini, kendali PID dapat menghasilkan nilai keluaran dengan *rise time* yang cepat dan nilai *error* yang kecil [21]. Blok diagram sistem kendali PID dapat dilihat pada Gambar 6 dibawah ini.



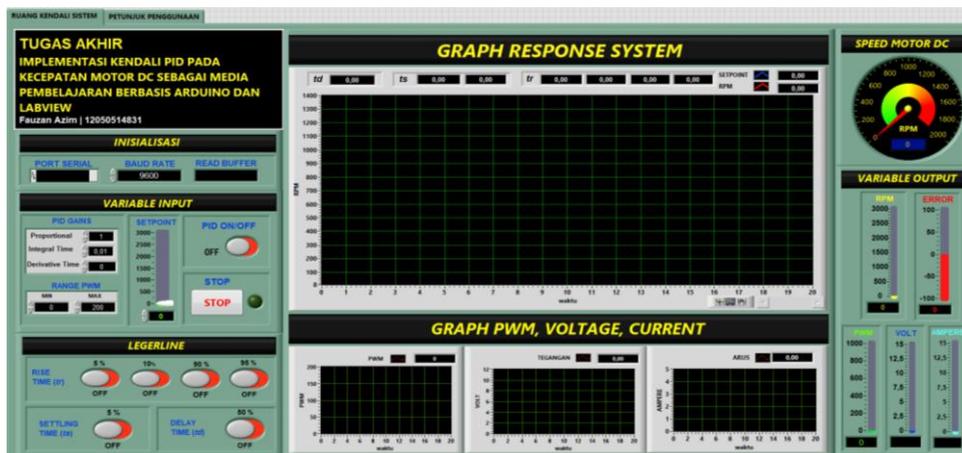
Gambar 6. Diagram Blok Sistem Kendali PID

Pada gambar 6, diagram blok sistem kendali PID dimulai dengan pemberian nilai setpoint ke dalam LabVIEW. Hasil pembacaan dari sensor kecepatan digunakan sebagai nilai aktual dari motor dan mengirimkan data ini ke LabVIEW melalui antarmuka data akuisisi. Dalam LabVIEW, kendali PID akan menghitung nilai *error* antara *setpoint* dan nilai aktual, kemudian menghasilkan sinyal kontrol berdasarkan komponen *proportional*, *integral*, dan *derivative*. Sinyal kontrol ini kemudian dikonversi menjadi sinyal PWM yang dikirim ke driver motor. Driver motor akan mengatur nilai arus dan tegangan sehingga motor dapat bergerak sesuai dengan sinyal PWM. Selama sistem berjalan, kendali PID secara otomatis menyesuaikan sinyal kontrol untuk mengurangi nilai *error* dan mempertahankan kecepatan motor pada *setpoint* yang diberikan meskipun ada gangguan.

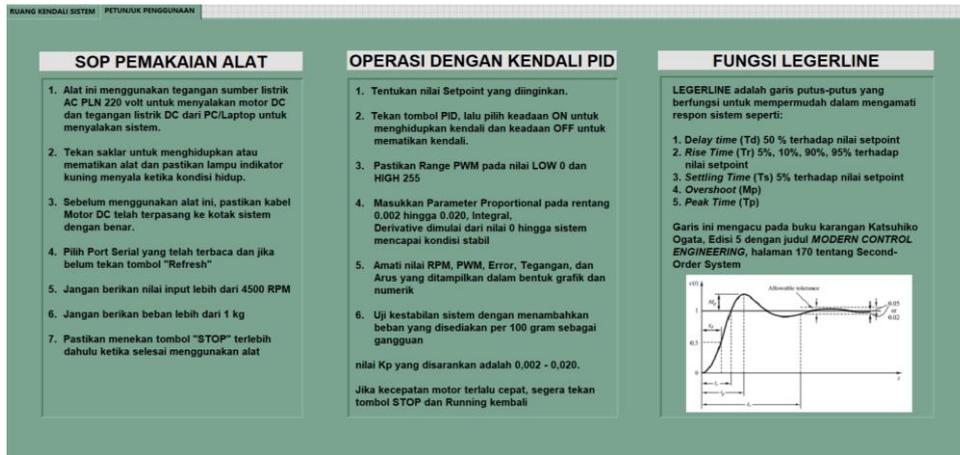
III. HASIL DAN PEMBAHASAN

A. Tampilan GUI Labview

Setelah blok diagram program LabVIEW dibuat, selanjutnya membuat tampilan GUI yang lebih menarik dan informatif agar memudahkan mahasiswa dalam menjalankan tampilan antarmuka. Terdapat dua bagian pada tampilan antarmuka LabVIEW yaitu tampilan GUI ruang kendali sistem dapat dilihat pada Gambar 7 dan tampilan GUI petunjuk penggunaan dapat dilihat pada Gambar 8.

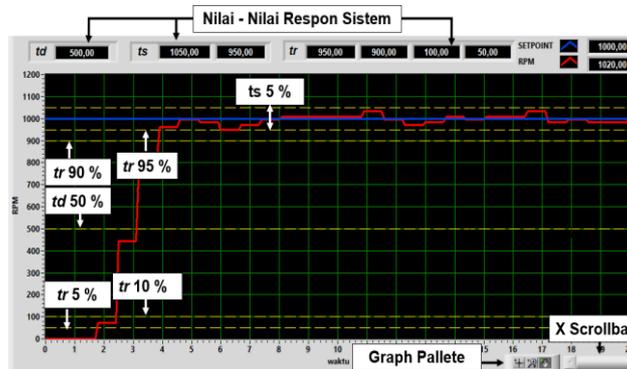


Gambar 7. Hasil Tampilan GUI Ruang Kendali Sistem



Gambar 8. Hasil Tampilan GUI Petunjuk Penggunaan

Pada Gambar 7, menampilkan visualisasi data hasil pembacaan sensor-sensor dalam bentuk grafik dan *vertical process bar*. Fitur *Variable Input* berfungsi sebagai data masukan seperti nilai *setpoint*, nilai parameter PID, dan *range* PWM. Selain itu, tersedia tombol untuk mengaktifkan kendali PID dan menghentikan program. Fitur *variable output* berperan dalam menampilkan hasil pemrosesan sistem, seperti nilai RPM, *error*, PWM, tegangan, dan Arus dalam bentuk *vertical process bar*. Sementara itu, data yang ditampilkan dalam bentuk grafik adalah nilai RPM, PWM, tegangan, dan arus. Pada Fitur *Legerline*, ketika diaktifkan maka pada tampilan grafik RPM akan muncul garis putus-putus berwarna kuning yang berguna bagi membantu mahasiswa dalam menentukan nilai dari *rise time* (T_r), *settling time* (T_s), dan *time delay* (T_d), seperti terlihat pada Gambar 9 dibawah ini.



Gambar 9. Fitur *Legerline* ketika keadaan On

Garis dan nilai respon sistem yang ditampilkan pada grafik RPM mengikuti nilai *setpoint* yang diberikan. Untuk memperbesar grafik respon sistem dapat menggunakan *graph pallette* sehingga didapatkan nilai pembacaan yang lebih akurat. Sedangkan pergeseran horizontal grafik dapat dilakukan dengan menggunakan *x scrollbar*. Selain itu, grafik juga dapat di *export* dalam bentuk format *clipboard*, *excel*, dan *simplified image*. Pada Gambar 9, menampilkan petunjuk penggunaan alat media pembelajaran sistem kendali PID yang terdiri dari SOP pemakaian alat, operasi dengan kendali PID, dan *fungsi Legerline*. Petunjuk ini dirancang untuk membimbing mahasiswa dalam menggunakan alat tersebut, sehingga dapat membantu untuk menjaga kinerja produk dan menghindari kerusakan.

B. Hasil Rancangan

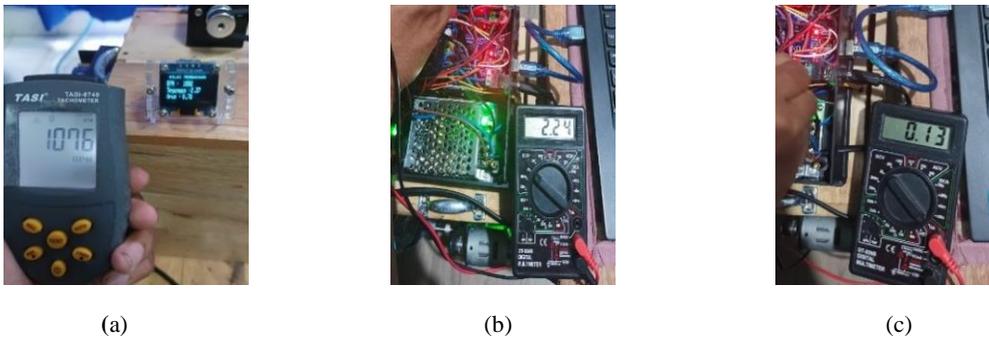
Dalam mempertahankan kinerja sistem dari terjadinya kerusakan, semua komponen perangkat keras yang telah terkoneksi satu sama lain akan ditempatkan dalam *project box*, kecuali sensor LM393 dan OLED Display karena bertujuan mempermudah dalam melakukan pembacaan kecepatan motor dan menampilkan hasil pembacaan semua sensor secara langsung. Laptop/PC digunakan sebagai media antarmuka antara mahasiswa dengan perangkat lunak LabVIEW seperti terlihat pada Gambar 10.



Gambar 10. Alat Media Pembelajaran Kendali PID Pada Kecepatan Motor DC Berbasis Arduino dan LabVIEW

C. Pengujian Kemampuan Sensor

Pada pengukuran sensor IR Optocoupler LM393 dengan satuan RPM, sensor Tegangan DC dengan satuan Volt, dan sensor ACS712 dengan satuan Ampere, dilakukan ketika motor DC menyala dan membandingkan hasilnya dengan alat ukur Tachometer dan Multimeter. Hal ini bertujuan untuk mengevaluasi pembacaan sensor dengan nilai *error* yang sekecil mungkin, seperti terlihat pada Gambar 11 dibawah ini.



Gambar 11. (a) Pengukuran Kecepatan pada Tachometer, (b) Pengukuran Tegangan pada Multimeter, (c) Pengukuran Arus pada Multimeter

Motor DC akan berputar saat Arduino Uno menerima perintah data analogWrite (0 - 255) melalui Serial Monitor. Kemudian driver motor mengonversi data tersebut dalam bentuk *pulse width modulation* (PWM), yang kemudian mengatur tegangan input ke motor DC untuk memvariasi kecepatannya. Sinyal PWM berperan dalam mengendalikan on/off pada MOSFET, yang bertindak sebagai saklar elektrik pada driver motor sehingga dapat mengubah tegangan digital (PWM) menjadi nilai analog dengan level tegangan dari 1-12 volt.

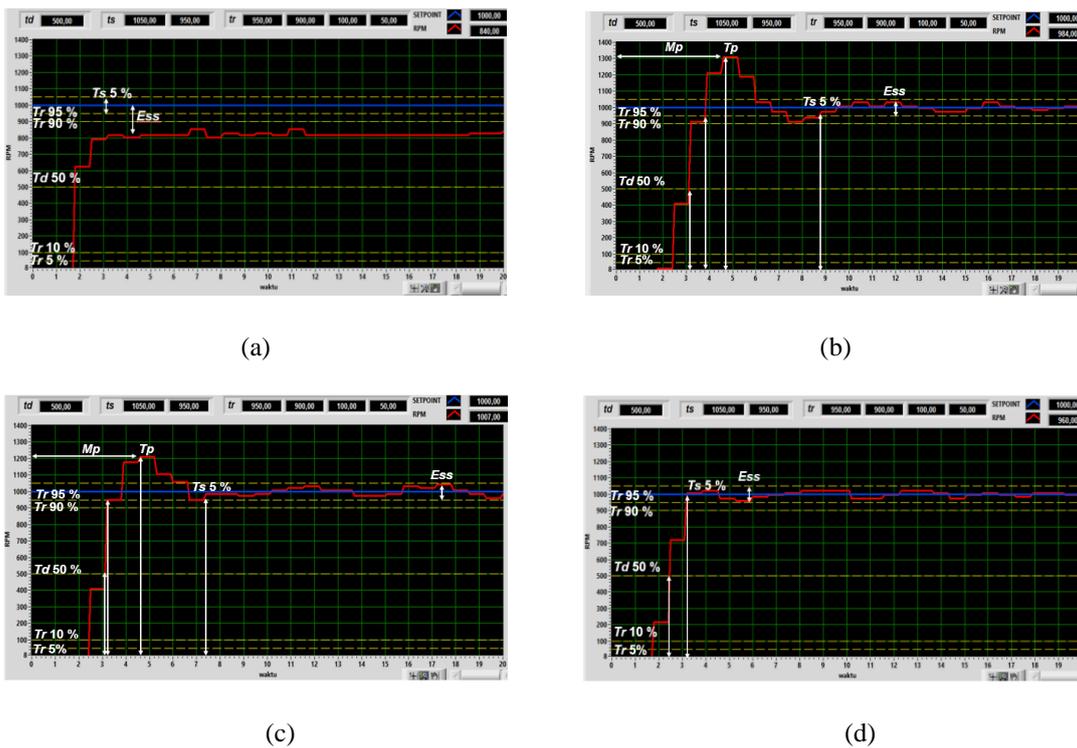
Tabel 1. Perhitungan *error* nilai sensor IR Optocoupler LM393, sensor Voltage Divider DC dan sensor ACS712

PWM (8-bit)	Sensor			Tachometer	Multimeter		Error (%)			
	RPM	V	A	RPM	V	A	RPM	V	A	
20	156	1,24	0,89	160	1,2	0,89	2,5	3,333	4,494	
40	922	2,83	1,47	920	2,56	1,2	0,217	10,546	22,5	
60	1599	4,4	2,02	1600	4,19	1,45	0,062	5,011	39,310	
80	2226	5,56	2,03	2220	5,53	1,63	0,270	0,542	24,539	
100	2723	6,93	2,29	2729	6,68	2,05	0,219	3,742	11,707	
120	3156	7,69	2,4	3155	7,65	2,23	0,031	0,522	7,623	
140	3433	8,21	2,44	3436	8,25	2,31	0,087	0,484	5,627	
160	3677	8,47	2,33	3685	8,92	2,32	0,217	5,044	0,431	
180	3878	9,56	2,4	3904	9,54	2,35	0,665	0,209	2,127	
200	4129	9,92	2,34	4150	9,85	2,37	0,506	0,710	1,265	
220	4306	1,37	2,47	4320	10,35	2,41	0,324	0,193	2,489	
250	4458	11,02	2,35	4450	10,96	2,45	0,179	0,547	4,081	
Rata-rata error								0,440	2,574	10,516

Berdasarkan data Tabel 1, motor DC mulai berputar ketika PWM melebihi 20, dengan kecepatan awal sekitar 160 RPM, dan kecepatan maksimum sekitar 4450 RPM. Nilai *error* berdasarkan selisih antara pembacaan sensor dengan alat ukur, kemudian dibagi dengan nilai alat ukur dan dikalikan 100%. Nilai rata-rata *error* yang terbaca pada sensor kecepatan sebesar 0.440 %. Hasil ini menunjukkan bahwa sensor mampu mengukur kecepatan motor dengan sangat baik. Selanjutnya, hasil pembacaan sensor tegangan dan sensor arus memiliki nilai yang beragam. Hal ini disebabkan oleh modulasi kecepatan putaran motor DC yang dikendalikan melalui sinyal PWM dari driver motor, yang merupakan sinyal digital. Sinyal digital memiliki dua kondisi yaitu *HIGH* (tinggi) dan *LOW* (rendah), yang dapat menyebabkan nilai pembacaan tegangan dan arus mengalami fluktuasi. Rata-rata *error* yang terbaca pada sensor tegangan sebesar 2.574 %, sementara sensor arus adalah sebesar 10.516 %. Hasil ini menunjukkan bahwa sensor tegangan cukup baik dalam mengukur tegangan pada motor DC sedangkan pada sensor arus memiliki *error* yang cukup tinggi pada kondisi tertentu. Hal ini disebabkan oleh fluktuasi dari siklus kerja PWM driver motor.

D. Pengujian Respon Sistem

Pada pengujian sistem, akan dilakukan dua kondisi yaitu ketika sistem tanpa kendali dan sistem dengan kendali PID. Pada pengujian ini, nilai *setpoint* yang ditetapkan adalah 1000 RPM dan untuk analisa respon sistem tombol Legerline diaktifkan. Untuk pengujian sistem dengan kendali PID akan dilakukan penalaan parameter PID menggunakan metode *trial and error*. Pada pengujian ini dilakukan 3 kali perubahan nilai parameter kendali PID. Pengujian ini bertujuan untuk mengamati perbedaan respon sistem antara menggunakan kendali atau tidak sekaligus membantu mahasiswa mengamati bagaimana dari karakteristik kendali PID. Grafik respon sistem dapat dilihat pada gambar 12 dan analisa respon sistem pengujian dikompilasi dalam Tabel 2 dibawah ini:



Gambar 12. Grafik Pengujian Respon: (a) Sistem Tanpa Kendali, (b) Kendali PID dengan nilai $K_p = 0,008$, $T_i = 0,054$, dan $T_d = 0$, (c) Sistem Kendali PID dengan nilai $K_p=0,010$, $T_i= 0,070$, dan $T_d=0,012$, (d) Sistem Kendali PID dengan nilai $K_p=0,012$, $T_i=0,096$, dan $T_d=0,018$

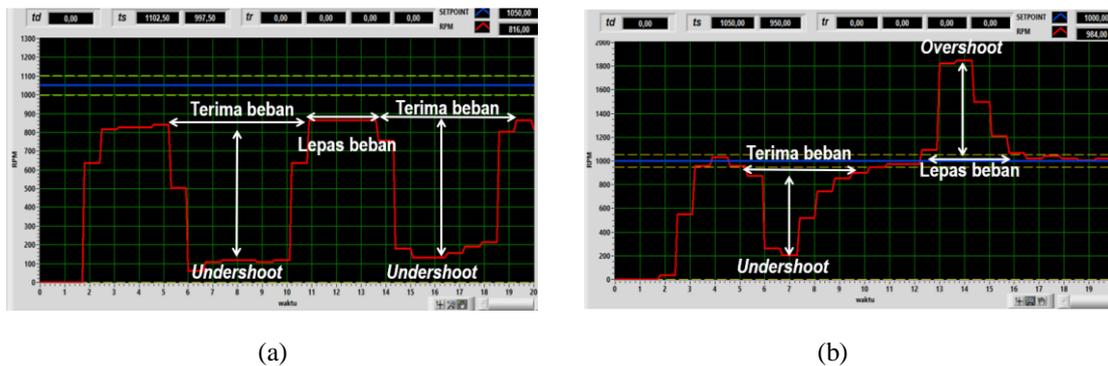
Tabel 2. Data Pengujian Respon Sistem Tanpa Kendali

Mode	Parameter PID			T_s (s)	T_d (s)	Tr 95 % (s)	T_p (s)	Ess (%)	M_p (%)
	K_p	K_i	K_d						
Tanpa kendali	-	-	-	-	1.8	-	-	16	-
Kendali PID	0,008	0,070	0,012	8.8	3.2	3.8	4.6	4.4	30.8
	0.010	0.070	0.012	7.4	3.1	3.2	4.6	4.4	19.5
	0.014	0.096	0.022	3.8	2.5	3.2	-	4.4	-

Berdasarkan Gambar 12 dan Tabel 2 diatas, pada Gambar (a) dapat diamati bahwa pada kondisi sistem tanpa kendali meskipun kecepatan motor DC dapat mencapai keadaan stabil dalam waktu yang relatif singkat, namun nilai *process variable* tidak mencapai *setpoint* yang diinginkan. Pembacaan nilai *process variable* sekitar 840 RPM, sehingga didapatkan nilai *error* melebihi batas toleransi respon sistem yaitu 5% dengan nilai stabilitas sistem sebesar 84%. Selain itu terdapat informasi tambahan yang ditampilkan pada GUI ruang kendali yaitu nilai PWM 36, serta pembacaan nilai tegangan sekitar 1.78V dan arus yang terbaca sekitar 0.78A. Sedangkan pada Gambar (b), Gambar (c), dan Gambar (d) kondisi sistem dengan kendali PID, nilai *process variable* telah mampu mencapai nilai *setpoint* yang diberikan dengan nilai *error* yang dihasilkan kurang dari 5% dengan tingkat stabilitas sistem sebesar 93.59 % - 93.5 %. Pada pengujian dengan parameter PID ($K_p = 0.008$, $T_i = 0.054$, $T_d = 0$), waktu yang diperlukan untuk mencapai kondisi *steady state* adalah yang paling lama, yaitu 11s dengan tingkat *overshoot* tertinggi sebesar 34.4 %. Sementara itu, pengujian dengan parameter PID ($K_p = 0.012$, $T_i = 0.96$, $T_d = 0.018$), waktu yang diperlukan untuk mencapai kondisi *steady state* adalah yang paling cepat, yaitu 3.8s serta tanpa adanya *overshoot*. Selain itu terdapat informasi tambahan yang ditampilkan pada GUI ruang kendali yaitu nilai PWM 38, serta pembacaan nilai tegangan sekitar 2.10V dan arus yang terbaca sekitar 0.83A. Dengan menggunakan metode *trial and error*, mahasiswa dapat mengamati bagaimana perubahan pada parameter K_p , T_i , dan T_d mempengaruhi respon sistem. Pada pengujian yang telah dilakukan, parameter PID yang optimal untuk mencapai stabilitas sistem dalam waktu yang singkat adalah $K_p = 0.014$, $T_i = 0.96$, dan $T_d = 0.022$.

E. Pengujian Terhadap Gangguan

Pengujian dilakukan terhadap gangguan, tombol *legerline* di nonaktifkan. Nilai *setpoint* yang digunakan adalah 1000 RPM kemudian diberikan beban 300 gram. Grafik respon sistem terhadap gangguan dapat dilihat pada Gambar 13 dan analisa respon sistem pengujian dikompilasi dalam Tabel 4 dibawah ini:



Gambar 13. Respon Gangguan 300 gram (a) Sistem Tanpa Kendali (b) Sistem Dengan Kendali PID

Tabel 3. Analisa Respon Sistem Terhadap Gangguan

Berat (gram)	Tanpa Kendali		Kendali PID	
	Terima	Lepas	Terima	Lepas
300	Turun 90 %, tidak kembali stabil	-	Turun 80 %, kembali stabil setelah 5 s	Naik 82,6 %, kembali stabil setelah 4.2 s

Berdasarkan Gambar 13 dan Tabel 3 diatas, pada Gambar (a) saat beban diterapkan pada motor DC tanpa sistem kendali, terjadi penurunan kecepatan yang signifikan atau *undershoot*. Namun, ketika beban dilepaskan, kecepatan kembali ke keadaan awal. Ini menunjukkan bahwa sistem tanpa kendali tidak efektif dalam mempertahankan nilai *setpoint* telah yang telah diberikan. Sementara itu, dengan kendali PID pada Gambar (b), saat beban diterapkan masih terjadi *undershoot* namun tidak sebesar pada sistem tanpa kendali. Selain itu, kendali PID memberikan kemampuan pada motor DC untuk kembali ke keadaan stabil setelah beberapa detik. Saat beban dilepaskan, terjadi *overshoot* di mana kecepatan motor DC melebihi nilai *setpoint* dalam sementara waktu. Hal ini disebabkan oleh sifat dinamis sistem dan karakteristik responsif dari kendali PID.

F. Evaluasi Usability

Evaluasi usability adalah proses penilaian kualitas penggunaan suatu sistem atau produk oleh pengguna. untuk mengidentifikasi dan memperbaiki masalah yang menghambat efisiensi, efektivitas, dan kepuasan pengguna. Dalam penelitian ini, tujuan dilakukannya evaluasi usability adalah untuk mengetahui kualitas alat yang telah dirancang dan perannya sebagai penunjang pada mata kuliah sistem kendali tentang pengaplikasian kendali PID di industri. Penyebaran kuesioner dilakukan pada 39 mahasiswa/I jurusan Teknik Elektro, konsentrasi elektronika instrumentasi. Sebanyak 15 mahasiswa/I dipilih sebagai responden untuk mengisi kuesioner dengan 15 pernyataan. Setiap Poin kuesioner dinilai menggunakan skala, dengan 5 kategori yaitu sangat baik (SB) = 5, baik (B) = 4, cukup (C) 3, kurang (K) = 2, dan sangat kurang (SK) = 1.

Tabel 4. Hasil Evaluasi Usability

No.	Poin - poin yang dinilai	Nilai (%)
1.	Sebagian besar PC/Laptop yang digunakan mahasiswa kompatibel dengan aplikasi antarmuka LabVIEW	98,33
2.	Pengguna merasa pemilihan desain, warna, dan layout pada antarmuka memiliki kualitas yang bagus sehingga mudah dipahami dan menyerupai tampilan HMI yang biasa digunakan di Industri	100
3.	Fitur-fitur atau fungsi-fungsi yang pada antarmuka mudah dimengerti.	88,33
4.	Pengguna dapat memodifikasi dan menghentikan setiap aktivitas yang terjadi pada antarmuka.	93,33
5.	Pengguna dapat mengubah data pada grafik dalam bentuk excel untuk dianalisis lebih lanjut.	93,33
6.	Setiap tombol yang ditekan pada antarmuka mampu merespon dalam waktu yang cepat	97,67
7.	Pengguna merasa grafik respon sistem yang ditampilkan memiliki nyaman untuk dipandang	93,33
8.	Motor DC mampu berputar pada kecepatan maksimal tanpa mengalami overheating	90
9.	Kotak alat terbuat dari bahan kayu yang ringan sehingga mudah dibawa.	90
10.	LED memudahkan pengguna untuk mengetahui kondisi alat sebelum dan ketika beroperasi	95
11.	Data yang dapat ditampilkan pada OLED Display mudah dimengerti	96,67
12.	Pengguna dapat dengan mudah membuka/memasang project box jika terjadi kesalahan sistem.	95
13.	Pengguna dapat dengan mudah memberikan gangguan pada sistem menggunakan beban yang telah disediakan.	91,67
14.	Panduan penggunaan alat rancang bangun mudah dimengerti	83,33
15.	Pengguna merasa dengan rancang bangun alat pengendalian kecepatan motor DC berbasis LabVIEW mampu meningkatkan kompetensi mahasiswa khususnya dalam pemahaman kendali PID	93,33
Rata-rata		93,22

Dari Tabel 4 di atas, secara keseluruhan nilai rata-rata persentase yang diperoleh terhadap tampilan antarmuka dan cara kerja sistem sebagai media pembelajaran sebesar 93,22 %. Dengan persentase nilai terendah terdapat pada “Buku panduan penggunaan alat rancang bangun mudah dimengerti” sebesar 83,33 % dan nilai persentase tertinggi terdapat pada “Pengguna merasa pemilihan desain, warna, dan layout pada antarmuka memiliki kualitas yang bagus sehingga mudah dipahami dan menyerupai tampilan HMI yang biasa digunakan di Industri” sebesar 100%. Hasil ini menunjukkan bahwa penelitian ini valid, praktis, dan efektif untuk meningkatkan minat belajar mahasiswa dalam mengenal konsep kendali PID secara nyata.

IV. KESIMPULAN

Alat media pembelajaran sistem kendali PID praktis pada kecepatan motor DC menggunakan Arduino Uno sebagai akuisisi data dan perangkat lunak LabVIEW 2020 telah berhasil dirancang. Alat ini memiliki tampilan antarmuka yang lebih menarik dengan penambahan *Legerline* sehingga memudahkan dalam analisis sistem serta dilengkapi dengan sensor tegangan dan arus, sehingga memberikan informasi yang lengkap dibandingkan penelitian sebelumnya. Berdasarkan hasil pengujian, respon sistem tanpa kendali menghasilkan *error* sebesar 16%, sedangkan dengan kendali PID menghasilkan dengan *error* sebesar 4,4%. Dan pengujian terhadap gangguan menunjukkan respon menunjukkan sistem dengan kendali PID, mampu menjaga stabilitas respon sistem dengan lebih baik daripada tanpa menggunakan kendali. Kemudahan dalam operasionalnya dibandingkan dengan alat *analog and digital motor control teaching set* membuat alat ini dapat menjadi solusi efektif bagi mahasiswa untuk mempelajari konsep kendali PID secara langsung. Berdasarkan hasil evaluasi usability, alat ini memperoleh nilai rata-rata keseluruhan sebesar 92,3 %.

ACKNOWLEDGMENT

Penulis mengucapkan terimakasih kepada petugas Laboratorium Fakultas Sains dan Teknologi UIN Suska Riau yang telah mengizinkan penulis melakukan penelitian di Laboratorium Elektronika Instrumentasi sehingga penelitian ini berjalan lancar dengan semestinya.

REFERENCES

- [1] H. Budiarto, V. Triwidyaningrum, F. Umam, and A. Dafid, "Implementation of Automatic DC Motor Braking PID Control System on (Disc Brakes)," vol. 4, no. 3, pp. 371–387, 2023, doi: 10.18196/jrc.v4i3.18505.
- [2] S. Gupta, P. Kumar, V. Gupta, Salim, and A. Kanungo, "Speed and Position Control of DC Motor using LabVIEW," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. August, 2021.
- [3] A. Ma'arif, N. R. Setiawan, and E. S. Rahayu, "Embedded Control System of DC Motor Using Microcontroller Arduino and PID Algorithm," *IT J. Res. Dev.*, vol. 6, no. 1, pp. 30–42, 2021, doi: 10.25299/itjrd.2021.vol6(1).6125.
- [4] M. D. I. Putri, A. Ma'arif, and R. Puriyanto, "PENGENDALI KECEPATAN SUDUT MOTOR DC MENGGUNAKAN KONTROL PID DAN TUNING ZIEGLER NICHOLS," *J. TECHNOL.*, vol. 23, no. 1, pp. 09–18, 2022.
- [5] S. N. Al-bargothi, G. M. Qaryouti, and Q. M. Jaber, "Speed Control of DC Motor Using Conventional and Adaptive PID Controllers," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 16, no. 3, pp. 1221–1228, 2019, doi: 10.11591/ijeecs.v16.i3.pp1221-1228.
- [6] C. N. Hamdani, A. Azizy, R. H. Triyanto, and H. K. Wardana, "Rancang Bangun Prototype Sistem Kontrol Bertingkat Menggunakan Fuzzy-Pid Berbasis Arduino," *Transmisi*, vol. 24, no. 3, pp. 98–105, 2022, doi: 10.14710/transmisi.24.3.98-105.
- [7] A. Ma'arif, H. Nabila, Iswanto, and O. Wahyunggoro, "Penerapan Algoritma Pencarian Cerdas di Kontrol Proporsional-Integral-Derivatif dari Direct- Sistem Motorik Saat Ini," *J. Phys. Conf. Ser.*, vol. 1373, no. 1, 2019, doi: 10.1088/1742-6596/1373/1/012039.
- [8] A. Ma'arif, H. Nabila, Iswanto, and O. Wahyunggoro, "Application of Intelligent Search Algorithms in Proportional-Integral-Derivative Control of Direct-Current Motor System," *J. Phys. Conf. Ser.*, vol. 1373, no. 1, 2019, doi: 10.1088/1742-6596/1373/1/012039.
- [9] M. Aulia, Kurniawati, P. Ali Topan, T. Andriani, and D. Maulidyawati, "Implementasi Sistem Kendali Proporsional Integral Derivatif (PID) pada Porototype Pendeteksi Brightness Cahaya Ruangan," *J. Bumigora Inf. Technol.*, vol. 4, no. 2, pp. 217–230, 2022, doi: 10.30812/bite.v4i2.2129.
- [10] I. Y. Basri, A. Arsyfadhillah, D. Irfan, and T. Thamrin, "Rancang Bangun Media Pembelajaran Mini Trainer IC 555," *INVOTEK J. Inov. Vokasional dan Teknol.*, vol. 18, no. 2, pp. 65–76, 2018, doi: 10.24036/invotek.v18i2.332.
- [11] F. Eliza, D. E. Myori, and F. G. Wireksi, "Trainer Sistem Kendali Elektronik Untuk Pembelajaran Mengoperasikan Sistem Kendali Elektronik," *J. Edukasi Elektro*, vol. 1, no. 2, pp. 110–118, 2017, doi: 10.21831/jee.v1i2.17414.
- [12] N. R. Wibowo, Aminuddin, and M. N. A. Syaputra, "Rancang Bangun Sistem Kendali Kecepatan Motor Dc Sebagai Media Pembelajaran Praktikum Sistem Kendali Menggunakan Labview," *JST (Jurnal Sains Ter.)*, vol. 6, no. 2, 2020, doi: 10.32487/jst.v6i2.775.
- [13] A. F. Aji, Sihono, B. Supriyo, V. S. Kartika, A. Ali, and A. R. Ashriyati, "Kendali PID Posisi Sudut Putar Motor DC Disertai Gangguan Berbasis MYRIO," *orbith*, vol. 18, no. 2, pp. 109–118, 2022.
- [14] C. A. Siagian, A. D. A. Nur Utomo, and I. Kresna A., "Sistem Pemantauan Suhu, Kelembapan, Cahaya, dan pH Air pada Rumah Walet Berbasis Internet of Things," *J. Telecommun. Electron. Control Eng.*, vol. 6, no. 1, pp. 1–12, 2024, doi: 10.20895/jtece.v6i1.988.
- [15] E. N. Afifah Amatullah, R. Ekawita, and E. Yuliza, "Comparison of Infrared and Optocoupler Sensors Performance for Lab-Scale Rpm Measurement System," *Indones. Phys. Rev.*, vol. 5, no. 2, pp. 130–136, 2022, doi: 10.29303/ipr.v5i2.150.
- [16] A. El Hammoumi, S. Motahhir, A. Chalh, A. El Ghzizal, and A. Derouich, "Real-time Virtual Instrumentation of Arduino and LabVIEW based PV Panel Characteristics," *IOP Conf. Ser. Earth Environ. Sci. Pap.*, vol. 161, no. 1, p. 012019, 2018.
- [17] A. J. Ali, A. M. T. Ibraheem, and O. T. Mahmood, "Design of a Smart Control and Protection System for Three-Phase Generator Using Arduino," *IOP Conf. Ser. Mater. Sci. Eng. Pap.*, vol. 745, no. 1, p. 012027, 2020, doi: 10.1088/1757-899X/745/1/012027.
- [18] A. N. Halisyah, D. Adiputra, and A. Al Farouq, "Field oriented control driver development based on BTS7960 for physiotherapy robot implementation," vol. 14, no. 2, pp. 1486–1495, 2024, doi: 10.11591/ijece.v14i2.pp1486-1495.
- [19] M. Saad, A. H. Amhedb, and M. Al Sharqawi, "Real time DC motor position control using PID controller in LabVIEW," *J. Robot. Control*, vol. 2, no. 5, pp. 342–348, 2021, doi: 10.18196/jrc.25104.

- [20] A. Afham, M. Muhtadan, and A. Suntoro, "Rancang Bangun Simulator Sistem Interlock Pada Iradiator Merah Putih Berbasis Labview," *J. Forum Nukl.*, vol. 12, no. 1, p. 7, 2018, doi: 10.17146/jfn.2018.12.1.3632.
- [21] R. A. N. Apriyanto, E. Purwanto, H. Oktavianto, and G. Prabowo, "Metode Kontrol Skalar dengan Penala Parameter PID Otomatis Menggunakan Algoritma PSO Sebagai Pengendali Kecepatan Motor Induksi Tiga Fasa Berbasis LabView," vol. 6, no. 1, 2020.