

Pengendalian Tegangan pada Simulator Plant Orde Dua dengan Metode PID Berbasis PLC-HMI WinCC

Muhammad Hafiz Aziz¹, Sarjono Wahyu Jadmiko², Sofian Yahya³

¹Jurusan Teknik Elektro, Politeknik Negeri Bandung, Bandung 40012

E-mail : muhammad.hafiz.toi17@polban.ac.id

²Jurusan Teknik Elektro, Politeknik Negeri Bandung, Bandung 40012

E-mail : sarjono_wahyu@polban.ac.id

³Jurusan Teknik Elektro, Politeknik Negeri Bandung, Bandung 40012

E-mail : sofianyahya@polban.ac.id

ABSTRAK

Salah satu metode pengendalian yang masih banyak digunakan adalah metode Proporsional Integral Derivatif (PID). Mengetahui nilai-nilai parameter plant secara *real time* dan pencatatan data plant sangatlah penting, oleh karena itu dibutuhkan *Human Machine Interface* (HMI) dan *database* pada sebuah *plant*. Dalam penelitian ini penulis membuat pengendalian tegangan pada simulator *plant* orde dua dengan metode PID berbasis PLC-HMI. PLC yang digunakan adalah PLC Siemens S7-1200 dengan HMI WinCC, dan Microsoft Excel sebagai *database*. Simulator *plant* dibuat pada SIMULINK MATLAB yang terhubung dengan PLC melalui *Open Platform Communications Unified Architecture* (OPC UA). Berdasarkan hasil pengujian, tegangan pada simulator *plant* orde dua saat diberi *set point* 5 volt dapat mencapai nilai *set point* dengan *error* sebesar 0% dan saat diberi gangguan sebesar -2 volt keluaran sistem mendapat *error* sebesar 5%, HMI dapat menampilkan data dari simulator *plant* dan nilai parameter PID, data dapat disimpan ke *database* dengan *delay* waktu rata-rata sebesar 2.4 detik.

Kata Kunci

Simulator plant orde dua, PID, PLC, HMI, Database

1. PENDAHULUAN

Salah satu metode pengendalian yang masih banyak digunakan adalah metode Proporsional Integral Derivatif (PID). Banyaknya penggunaan PID sebagai komponen kontrol dikarenakan strukturnya yang sederhana dan kemudahan dalam melakukan *tuning* parameter kontrolnya [1]. HMI menjadi penghubung antara operator dengan peralatan yang dikendalikan, berfungsi untuk menampilkan informasi yang mudah dipahami secara visual untuk pengawasan dan mudah dalam melakukan pengendalian sehingga dapat meningkatkan efisiensi tenaga dan waktu. *Database* digunakan untuk menyimpan data yang diperlukan yang berfungsi untuk mempermudah dalam melakukan *troubleshooting* saat mengalami masalah, analisis data, dan fungsi lainnya. Mengintegrasikan PLC dan MATLAB memiliki keuntungan diantaranya memudahkan proses pemodelan, mengurangi kerumitan, menghemat biaya, dengan tidak adanya perangkat keras tambahan maka tidak perlu mengubah I/O analog ke digital atau sebaliknya, karena semua I/O dalam bentuk digital [2].

Berbagai metode pengendalian telah dikembangkan oleh peneliti terdahulu, penelitian yang dilakukan S. W. Jadmiko dkk membandingkan kendali PID dan kendali logika *fuzzy* menggunakan PLC Omron sebagai kontroler diperoleh kesimpulan kinerja kendali PID lebih baik pada *time constant* yang kecil, sedangkan kinerja logika *fuzzy* cukup baik apabila diterapkan pada *plant* yang memiliki *time constant* yang besar [3]. Penelitian Muhtarom dkk membandingkan kendali PID

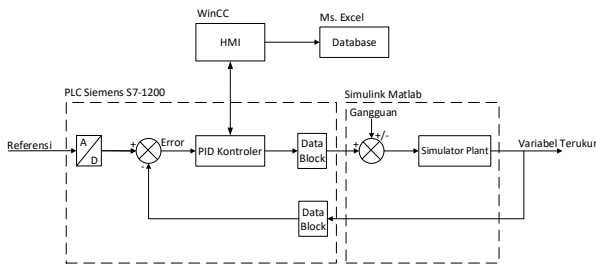
dan kendali logika *fuzzy* pada pengendalian suhu memperoleh kesimpulan bahwa kendali yang lebih baik terdapat pada kendali PID yang memiliki *overshoot*, *error steady state*, *rise time* dan *peak time* yang lebih kecil daripada logika *fuzzy* [4]. Penggunaan kontroler PID pada PLC Siemens dalam penelitian yang dilakukan oleh D. V. Nghiep dkk diperoleh kesimpulan penggunaan kontrol PID pada PLC Siemens efisien karena terdapat *function block* yang beragam [5]. Penggunaan aplikasi HMI WinCC dan *database* Microsoft SQL Server dalam penelitian yang dilakukan Z. Wei menunjukkan *real time monitoring* dan pengontrolan dapat dilakukan dengan aplikasi WinCC pada komputer dan data otomatis tersimpan di *database* [6]. Penelitian yang dilakukan W. Zhang dkk menggunakan aplikasi HMI WinCC dan *database* Microsoft SQL Server pada sistem yang dikendalikannya menunjukkan hasil *monitoring* yang komprehensif dan kontrol presisi [7].

Para peneliti terdahulu telah mengaplikasikan metode PID untuk pengendalian tetapi tidak terdapat HMI dan *database* sehingga untuk melakukan pemantauan data maka harus membuka aplikasi dari kontroler yang digunakan, selain itu data hanya dapat disimpan dengan cara ditulis manual. Penggunaan PLC Siemens dan HMI WinCC pada penelitian terdahulu menunjukkan kinerja yang baik. Berdasarkan kajian rujukan di atas dikembangkanlah sistem "Pengendalian Tegangan Pada Simulator Plant Orde Dua Dengan Metode PID Berbasis PLC-HMI WinCC" agar didapatkan pengendalian yang lebih baik.

2. METODE

2.1 Prinsip Kerja Alat

Gambar 1 memperlihatkan blok diagram dari sistem, tegangan referensi yang diberikan akan masuk ke kontroler PLC Siemens S7-1200 yang sebelumnya dikonversi menjadi sinyal digital oleh *Analog to Digital Converter* (ADC). Keluaran kontroler masuk ke simulator plant orde dua melalui OPC UA, selain sinyal kontroler sinyal gangguan juga diberikan ke simulator *plant* orde dua. Sinyal keluaran simulator *plant* diumpun balik ke kontroler untuk dibandingkan dengan nilai referensi.



Gambar 1. Blok diagram sistem

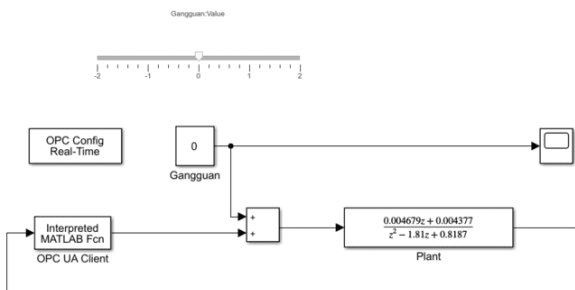
Dengan adanya pengendali PID, tegangan keluaran simulator diharapkan stabil sebesar 5 volt dengan *error* $\pm 5\%$ pada saat diberi gangguan sebesar 0 volt sampai dengan -2 volt, untuk menentukan parameter PID digunakan metode kurva reaksi (*open loop tuning*) yang hasilnya dihitung menggunakan tabel Ziegler-Nichols tipe 1. Selain itu, nilai parameter PID dapat diawasi dan diatur melalui HMI dan dapat disimpan ke *database*.

2.2 Perancangan Simulator Plant Orde Dua

Simulator *plant* orde dua dibuat pada SIMULINK MATLAB yang ditunjukkan pada gambar 2, *plant* yang dibuat memiliki fungsi alih pada persamaan (1), agar dapat bekerja dengan PLC maka fungsi alih terlebih dahulu diubah menjadi fungsi alih diskrit, sehingga fungsi alih diskritnya ditunjukkan pada persamaan (2).

$$\frac{1}{s^2 + 2s + 1} \quad (1)$$

$$\frac{0.004679z + 0.004377}{z^2 - 1.81z + 0.8187} \quad (2)$$



Gambar 2. Simulator *plant*

2.3 Perancangan Komunikasi Antara Simulator Plant Dengan PLC

Komunikasi data antara PLC dengan simulator *plant* menggunakan OPC UA seperti yang ditunjukkan pada gambar 3, nilai keluaran kendali PID dikirim oleh PLC sebagai OPC UA Server, dan nilai keluaran dari *plant* dikirim oleh simulator *plant* sebagai OPC UA Client ke PLC.



Gambar 3. Komunikasi PLC dengan simulator plant

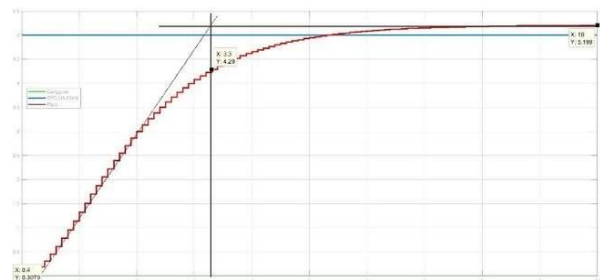
PLC berkomunikasi dengan SIMULINK MATLAB menggunakan kabel ethernet seperti yang ditunjukkan pada gambar 4.



Gambar 4. Hardware

2.4 Perancangan Pengendali PID

Untuk menentukan parameter kendali PID pada penelitian ini digunakan metode kurva reaksi. Simulator *plant* dibuat lup terbuka, lalu diberi nilai 5V dari PLC secara *force*, amati sinyal keluaran *plant* yang ditunjukkan dengan garis berwarna merah pada gambar 5, lalu hitung menggunakan tabel Ziegler-Nichols tipe 1. Perpotongan garis singgung dengan sumbu absis disebut *dead time* (L), sementara perpotongan garis maksimum disebut *delay time* (T), nilai *Ks* ditentukan dari nilai referensi dibagi nilai keluaran.



Gambar 5. Keluaran sistem lup terbuka

Dari respons sistem pada gambar 5 didapatkan nilai *L* = 0.4, nilai *T* = 2.9, dan nilai *Ks* = 1.04.

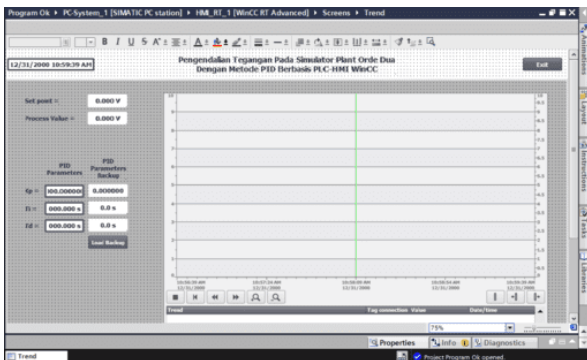
Tabel 1. Rumus parameter PID dengan Ziegler-Nichols tipe 1

Jenis alat kendali	Kp	Ti	Td
P	$\frac{T}{(L * Ks)}$	∞	0
PI	$\frac{0.95 T}{(L * Ks)}$	$\frac{L}{0.3}$	0
PID	$\frac{1.2 T}{(L * Ks)}$	2 L	0.5 L

Dengan menggunakan tabel Ziegler-Nichols pada tabel 1 didapatkan nilai parameter PID dengan nilai Kp = 8.365385, Ti = 0.8, dan Td = 0.2.

2.5 Perancangan HMI

HMI dirancang untuk menampilkan grafik nilai *set point* dan nilai keluaran *plant*, dan juga untuk dapat mengatur parameter PID yang ditunjukkan pada gambar 6. Pada penelitian ini perangkat yang digunakan sebagai HMI adalah laptop.



Gambar 6. Tampilan HMI

2.6 Perancangan Database

Database digunakan untuk menyimpan data dari simulator *plant*. pada penelitian ini data dari simulator *plant* disimpan secara berkala setiap satu menit, *file database* akan dibuat setiap satu hari sekali. Format *database* yang dibuat ditunjukkan pada gambar 7 memiliki informasi waktu penyimpanan data, nilai *set point*, nilai *process value*, nilai Kp, nilai Ti, dan nilai Td.

The screenshot shows an Excel spreadsheet with the following data table:

No.	Date & Time	Set Point (V)	Process Value (V)	Kp	Ti	Td
3				5.5		
4				5		
5				4.5		
6				4		
7				3.5		
8				3		
9				2.5		

The spreadsheet also includes a title 'Rancang Bangun Pengendalian Tegangan Pada Simulator Plant Orde Dua Dengan Metode PID Berbasis PLC-HMI WinCC' and the POLBAN logo.

Gambar 7. File database referensi

3. HASIL DAN PEMBAHASAN

3.1 Pengujian Komunikasi PLC Dengan Simulator Plant

Untuk mengecek data dapat berkomunikasi, berikan nilai terlebih dahulu pada *tag* PLC, pada pengujian ini *tag* Control diberikan nilai 5 dan *tag* PV diberikan nilai 0.

The screenshot shows a table with the following data:

	Name	Data type	Start value	Monitor valu
1	Static			
2	Control	Real	0.0	5.0
3	PV	Real	0.0	0.0

Gambar 8. Memberi nilai pada tag PLC

Lihat nilai tag Control pada SIMULINK MATLAB, nilai tag Control bernilai 5 seperti pada gambar 9.

```

Command Window
>> Var_Node_In = opcuanode(4,2, uaClient)

Var_Node_In =

OPC UA Node object:
    Name: Control
    Description:
    NamespaceIndex: 4
    Identifier: 2
    NodeType: Variable
    Children: 0 nodes.
    ServerDataType: Float
    AccessLevelCurrent: read/write
    AccessLevelHistory: none
    Historizing: 0

>> [val, ~, ~] = readValue(uaClient, Var_Node_In)

val =

single

5
    
```

Gambar 9. Pengujian tag control

Lakukan perubahan nilai *tag* PV pada SIMULINK MATLAB seperti pada gambar 10, kemudian lihat kembali *tag* PV pada PLC seperti pada gambar 11, nilai *tag* PV pada PLC berubah mengikuti nilai pada SIMULINK MATLAB.

```

Command Window
>> Var_Node_Out = opcuanode(4,3, uaClient)

Var_Node_Out =

OPC UA Node object:
    Name: PV
    Description:
    NamespaceIndex: 4
    Identifier: 3
    NodeType: Variable
    Children: 0 nodes.
    ServerDataType: Float
    AccessLevelCurrent: read/write
    AccessLevelHistory: none
    Historizing: 0

>> [val, ~, ~] = readValue(uaClient, Var_Node_Out)

val =

single

0

>> writeValue(uaClient, Var_Node_Out, 5)
>> [val, ~, ~] = readValue(uaClient, Var_Node_Out)

val =

single

5
    
```

Gambar 10. Pengujian tag PV

Data_block_OPCLUA				
	Name	Data type	Start value	Monitor valu
1	Static			
2	Control	Real	0.0	5.0
3	PV	Real	0.0	5.0

Gambar 11. Cek nilai pada tag PLC

3.2 Pengujian Pengendali PID

Untuk kendali PID menggunakan metode Ziegler-Nichols 1 nilai parameter yang digunakan berdasarkan hasil perhitungan pada tahap perancangan pengendalian PID dengan nilai $K_p = 8.365$, $T_i = 0.8$, dan $T_d = 0.2$. Dengan *set point* 5 volt keluaran sistem setelah *steady state* dapat mencapai nilai *set point* dengan *error* 0%, respon awal memiliki *overshoot* sebesar 16% dan saat diberi gangguan sebesar -2 volt keluaran sistem mendapat *error* sebesar 5%.

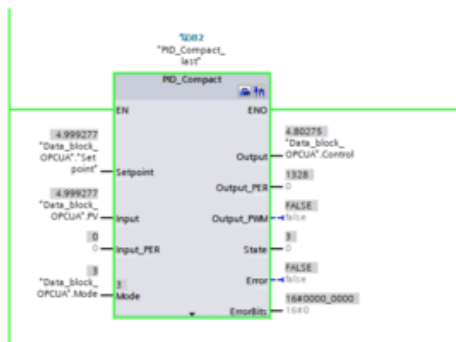


Gambar 12. Keluaran simulator plant

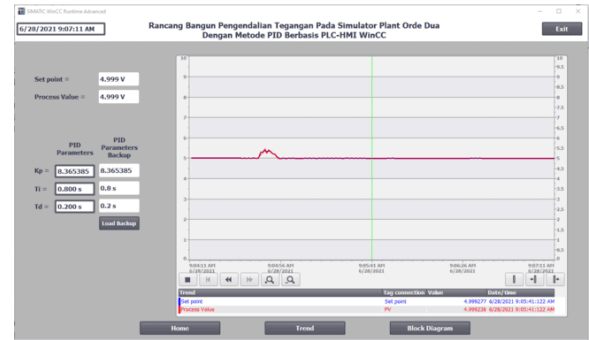
Sinyal berwarna merah merupakan keluaran dari simulator *plant*, dan sinyal berwarna biru merupakan sinyal kontrol keluaran dari PLC.

3.3 Pengujian HMI

Pengujian HMI dilakukan untuk mengetahui keberhasilan HMI menampilkan data dari simulator *plant* dan parameter PID dapat diatur dari HMI. Untuk mengetahui data dari simulator *plant* berhasil ditampilkan HMI, bandingkan nilai pada PLC dengan cara memonitor pada aplikasi TIA Portal dengan *trend* pada HMI. Pada saat *set point* dan *process value* pada PLC bernilai 4.999 pada HMI bernilai sama sebesar 4.999. Pada HMI garis berwarna merah merupakan nilai *process value* dan garis berwarna biru merupakan nilai *set point*.

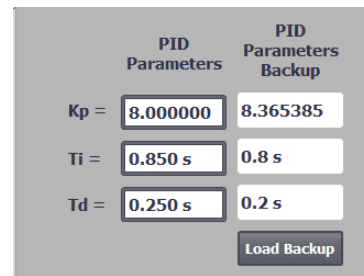


Gambar 13. Nilai pada PLC



Gambar 14. Nilai pada HMI

Untuk mengetahui apakah nilai parameter PID dapat diatur oleh HMI, ubah parameter PID pada HMI dengan nilai K_p sebesar 8, nilai T_i sebesar 0.85 dan nilai T_d sebesar 0.25 seperti pada gambar 15. Selanjutnya buka *tag* dari PID untuk melihat nilai pada PLC seperti pada gambar 16.



Gambar 15. Parameter PID pada HMI

PID_Compact_last (snapshot created: 4/30/2021 8:42:28 AM)				
Name	Data type	Start value	Monitor value	
48	Retain	PID_CompactRetain		
49	CtrlParams	PID_CompactContr...		
50	Gain	Real	8.365385	8.0
51	Ti	Real	0.8	0.85
52	Td	Real	0.2	0.25
53	TdPitRatio	Real	0.2	0.2
54	PWeighting	Real	1.0	1.0
55	DWeighting	Real	1.0	1.0
56	Cycle	Real	0.1	0.1

Gambar 16. Parameter PID pada PLC

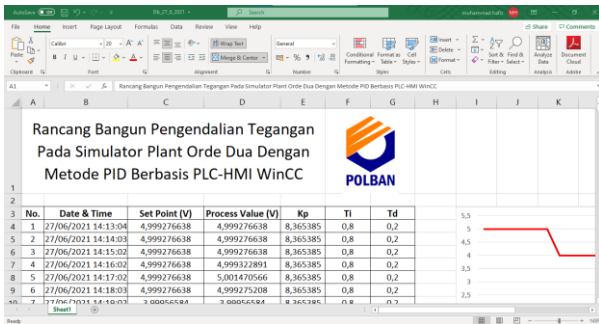
3.4 Pengujian Database

Setelah data dari simulator *plant* dapat ditampilkan di HMI selanjutnya menguji apakah data simulator *plant* tersimpan pada *database*. Buka lokasi folder yang digunakan untuk menyimpan *file database*, untuk melihat apakah data tersimpan.

No.	Date & Time	Set Point (V)	Process Value (V)	Kp	Ti	Td	
1	27/06/2021 14:13:04	4.999276638	4.999276638	8.365385	0.8	0.2	5.5
4							5
5							4.5
6							4
7							3.5
8							3
9							2.5

Gambar 17. File database

Untuk menguji apabila data simulator *plant* berubah apakah semua data tetap tersimpan setiap satu menit, lakukan perubahan nilai *set point* dan nilai parameter PID, kemudian buka kembali *file database*. Data dari *file database* ditunjukkan pada tabel 2.



Gambar 18. Data tersimpan pada file database

Seluruh data dari HMI dapat tersimpan pada Microsoft Excel dengan interval waktu setiap satu menit. Penyimpanan data ke *database* memiliki *delay* waktu rata-rata sebesar 2.4 detik.

Tabel 2. Data pada *database*

Date & Time	Set Point (V)	Process Value (V)	Kp	Ti	Td
27/06/2021 14:13:04	4,999276638	4,999276638	8,365385	0,8	0,2
27/06/2021 14:14:03	4,999276638	4,999276638	8,365385	0,8	0,2
27/06/2021 14:15:02	4,999276638	4,999276638	8,365385	0,8	0,2
27/06/2021 14:16:02	4,999276638	4,999322891	8,365385	0,8	0,2
27/06/2021 14:17:02	4,999276638	5,001470566	8,365385	0,8	0,2
27/06/2021 14:18:03	4,999276638	4,999275208	8,365385	0,8	0,2
27/06/2021 14:19:02	3,99956584	3,99956584	8,365385	0,8	0,2
27/06/2021 14:20:02	3,99956584	3,99956584	8,365385	0,8	0,2
27/06/2021 14:21:03	3,99956584	3,99956584	8,365385	0,8	0,2
27/06/2021 14:22:02	3,99956584	3,99956584	8,365385	0,8	0,2
27/06/2021 14:23:02	3,99956584	3,99956584	8,365385	0,8	0,2
27/06/2021 14:24:02	2,999493837	2,999496937	8,365385	0,8	0,2
27/06/2021 14:25:03	2,999493837	2,999477625	8,365385	0,8	0,2
27/06/2021 14:26:03	2,999493837	2,999382019	8,365385	0,8	0,2
27/06/2021 14:27:02	2,999493837	2,99920249	8,365385	0,8	0,2
27/06/2021 14:28:03	2,998408794	2,999409676	8,365385	0,8	0,2
27/06/2021 14:29:03	1,99978292	1,99978292	8	0,85	0,25
27/06/2021 14:30:02	1,99978292	1,99978292	8	0,85	0,25
27/06/2021 14:31:02	1,99978292	1,99978292	8	0,85	0,25
27/06/2021 14:32:03	1,99978292	1,99978292	8	0,85	0,25
27/06/2021 14:33:02	1,99978292	1,99978292	8	0,85	0,25
27/06/2021 14:34:02	3,555049419	3,555049419	8,365385	0,8	0,2
27/06/2021 14:35:02	3,555049419	3,555049419	8,365385	0,8	0,2
27/06/2021 14:36:03	3,555049419	3,555049419	8,365385	0,8	0,2

27/06/2021 14:37:02	3,555049419	3,555049181	8,365385	0,8	0,2
27/06/2021 14:38:02	4,517867565	4,518830776	8,365385	0,8	0,2
27/06/2021 14:39:02	4,517867565	4,517867088	8,365385	0,8	0,2
27/06/2021 14:40:03	4,517867565	4,517867565	8,365385	0,8	0,2
27/06/2021 14:41:02	4,517867565	4,517867565	8,365385	0,8	0,2
27/06/2021 14:42:02	4,517867565	4,517905235	8,365385	0,8	0,2

4. KESIMPULAN

Berdasarkan hasil pengujian dan analisis, maka dapat diperoleh kesimpulan:

1. PLC dapat mengatur simulator *plant* pada SIMULINK MATLAB.
2. Tegangan pada simulator *plant* orde dua dapat mencapai nilai *set point* 5 volt dengan *error* 0%, respon awal memiliki *overshoot* sebesar 16% dan saat diberi gangguan -2 volt keluaran sistem mendapat *error* sebesar 5%.
3. HMI yang dibuat dengan WinCC dapat menampilkan data dari simulator *plant*, dan dapat mengatur nilai parameter PID.
4. Data dari simulator *plant* dan nilai parameter PID dapat disimpan pada *database* Microsoft Excel dengan *delay* waktu rata-rata sebesar 2.4 detik.

UCAPAN TERIMA KASIH

Tim peneliti mengucapkan terima kasih kepada Politeknik Negeri Bandung, melalui Wakil Direktur Akademik atas bantuan pendanaan penyusunan Tugas Akhir dengan nomor B/402/PL1.R1/EP.00.08/2021 kelompok A1.

DAFTAR PUSTAKA

- [1] I. Setiawan, *Kontrol PID untuk Proses Industri*. Elex Media Komputindo, 2013. [Online]. Available: <https://books.google.co.id/books?id=QeJMDwAAQBAJ>
- [2] E. M. Sartika, T. R. Sarjono, and D. D. Saputra, "Prediction of PID control model on PLC," *TELKOMNIKA Telecommun. Comput. Electron. Control*, vol. 17, no. 1, p. 529, Feb. 2019, doi: 10.12928/telkomnika.v17i1.11589.
- [3] S. W. Jadmiko, S. Yahya, S. Sudrajat, and F. Azizah, "Komparasi Kinerja Kendali PID dan Logika Fuzzy pada Simulator Plant Orde Dua," *JTERA J. Teknol. Rekayasa*, vol. 5, no. 2, p. 237, Dec. 2020, doi: 10.31544/jtera.v5.i2.2020.237-246.
- [4] M. Muhtarom and S. Sujono, "Perbandingan sistem kendali PID dan kendali logika fuzzy pada pengendalian suhu sistem pemanas induksi," *Maest. Vol 2 No 1 2019 Ed. April 2019*, 2019, [Online]. Available: <https://jom.ft.budiluhur.ac.id/index.php/maestro/article/view/150>
- [5] D. V. Nghiep and L. H. Son, "Implementation of Digital PID Controller in Siemens PLC S7-300," *Int. J. Electr. Electron. Eng.*, vol. 5, no. 5, pp. 10–13, May 2018, doi: 10.14445/23488379/IJEEE-V5I5P103.

- [6] Z. Wei and M. Tao, "Design of Continuous Hydrogenation Control System Based on WinCC Configuration Software," in *2019 Chinese Control And Decision Conference (CCDC)*, Nanchang, China, Jun. 2019, pp. 565–570. doi: 10.1109/CCDC.2019.8833354.
- [7] W. Zhang and Y. Li, "Design of Automatic Feeding Control System in Tank Area Based on WinCC Configuration Software," in *2020 Chinese Control And Decision Conference (CCDC)*, Hefei, China, Aug. 2020, pp. 2371–2376. doi: 10.1109/CCDC49329.2020.9164796.