OPTIMASI PENGENDALIAN LEVEL VESSEL MENGGUNAKAN PID CONTROLLER BERBASIS ESP32 DENGAN MONITORING IOT

Sofya Suci Anjarwati[1], Mochamad Rizky Pradana[1*], Suka Handaja Budi[1]
[1]Teknik Instrumentasi Kilang, Politeknik dan Mineral Akamigas, Jl. Gajah Mada No.37, Blora, 55312
**E-mail*: *Mochamad.Pradana@esdm.go.id*

ABSTRAK

Penelitian ini membahas perancangan dan implementasi sistem pengendalian level cairan pada vessel menggunakan mikrokontroler ESP32 yang terintegrasi dengan teknologi *Internet of Things* (IoT). Sensor ultrasonik digunakan untuk mengukur ketinggian cairan, sementara aktuator berupa servo valve mengatur aliran masuk berdasarkan data yang diproses oleh ESP32. Sistem ini dikendalikan dan dipantau secara real-time melalui aplikasi *Blynk* yang berfungsi sebagai antarmuka pengguna pada perangkat mobile. Selain pemantauan jarak jauh, penggunaan Blynk dapat memungkinkan pengaturan setpoint dan gain dari kontroler secara real time. Sistem dirancang untuk meningkatkan efisiensi dan akurasi dalam pengendalian proses cairan. Selain itu, integrasi IoT memungkinkan pemantauan jarak jauh dan pengambilan keputusan yang lebih responsif. Tiga metode tuning kontroler PID dibandingkan dalam pengujian, yaitu *Cohen-Coon, Ziegler-Nichols*, dan Internal Model Control (IMC). Berdasarkan hasil pengujian, metode *Cohen-Coon* dengan kontroler PI memberikan kinerja paling optimal dengan *rise time* sebesar 17 detik, *settling time* sebesar 62 detik, dan *overshoot* sebesar 5%. Konfigurasi ini direkomendasikan karena mampu memberikan respons cepat, minim osilasi, dan kestabilan tinggi. Skala laboratorium sebagai solusi otomatisasi yang efisien dan hemat biaya.

Kata kunci: Pengendalian Level, PID Controller, ESP32, IoT Blynk, Tuning PID

1. PENDAHULUAN

Level merupakan parameter yang digunakan di industri proses sebagai indikator keadaan suatu proses. Level digunakan sebagai penanda ketersediaan feedstock bagi suatu proses kontinu. Pemantauan atas feedstock menjadi bagian penting agar proses dapat berjalan tanpa adanya gangguan dari suplai bahan baku industri[1]. Selain itu, level suatu fluida bagi tangki menjadi penanda proses penguapan suatu fluida didalam tangki. Proses penguapan pada tangki harus diperhatikan karena agar tidak terjadi degradasi pada crude oil diakibatkan oleh proses oksidasi yang terjadi. Oleh karena itu pengendalian level di dalam tangki menjadi suatu bagian penting bagi industri proses[2].

Di industri saat ini, sudah terdapat beberapa metode untuk melakukan pengendalian level pada tangki, seperti kontroler on-off[3], kontroler proportional[4], [5], kontroler PID[6], [7], [8], [9], [10], hingga kontroler yang menggunakan fuzzy logic[2], [11] maupun beberapa teknik dari artificial intelligence, seperti metode reinforced learning[12], [13]. Kontroler PID, dengan beberapa variasi seperti, PI dan PD, merupakan sensor yang paling sering digunakan di industri proses. Namun untuk mendapatkan kontrol dengan respon yang optimal pada kontroler PID, diperlukan proses optimalisasi berupa tuning pada kontroler PID tersebut.

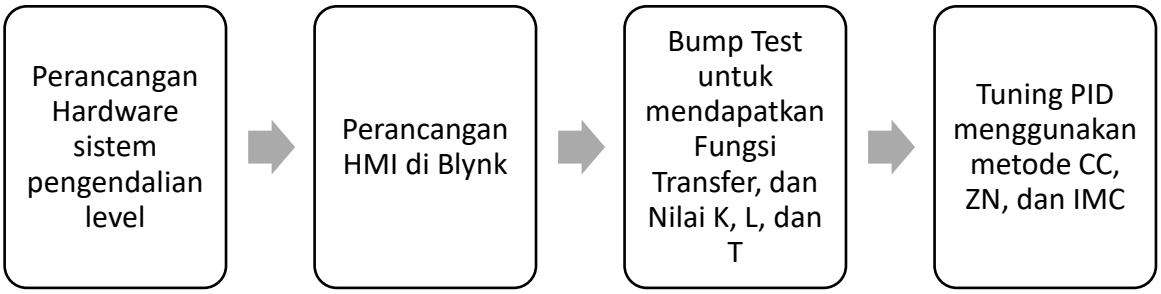
Beberapa penelitian yang telah dilakukan sebelumnya telah mengkaji penggunaan tuning kontroler PID pada pengukuran level. Penelitian yang dilakukan oleh Mukkaromah menunjukkan bahwa pengendalian level pada steam drum boiler dengan menggunakan metode tuning IMC-PID menghasilkan performa yang baik ketika mengalami perubahan set point dan

disturbance dengan pm 5%[14]. Selain itu penelitian yang dilakukan oleh Saad menunjukkan bahwa penggunaan metode tuning Ziegler Nichols kontroler PID pada coupled-tank menghasilkan respon dengan rise yang cepat dan mampu menciptakan kestabilan pada sistem. Namun penggunaan metode tuning Ziegler Nichols pada kontroler PID di penelitian Saad menghasilkan overshoot yang cukup tinggi, yaitu 12%[15].

Berdasarkan beberapa penelitian yang telah dilakukan sebelumnya, penelitian yang akan dibawakan saat ini adalah perbandingan metoda tuning Ziegler Nichols, Cohen Coon, dan IMC pada kontroler PID di pengendalian level. Peneliti ingin membandingkan performa ketiga metode tuning tersebut dan mengetahui respon transien yang diberikan oleh penggunaan metoda tuning. Selain itu peneliti juga ingin mendapatkan respon transien terkait dengan konfigurasi kontroler PID, seperti P, PI, dan PD, terhadap penggunaan ketiga metode tuning tersebut.

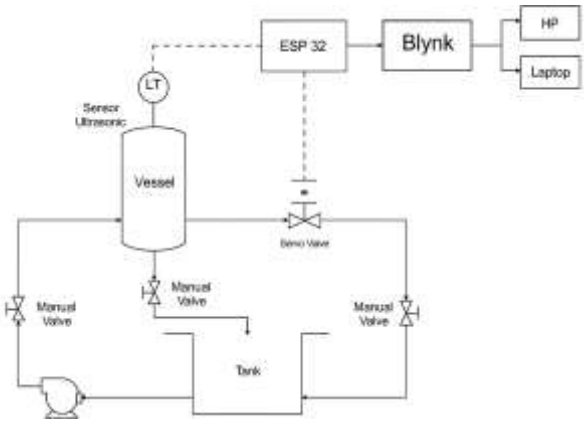
2. METODE

Berikut adalah flowchart yang digunakan sebagai runtutan langkah pada penelitian ini, seperti yang ditunjukan pada gambar 1.



Gambar 1. Flowchart Penelitian Pengendalian Level

Penelitian ini dimulai dari perancangan hardware berupa vessel air yang dilengkapi katup sebagai outlet air dari vessel. Katup dilengkapi motor servo MG996R sebagai aktuator penggerak katup. Aktuator bekerja sebagai pembuka dan penutup katup hingga level mencapai set point yang telah ditentukan. Penentuan set point diatur pada mikrokontroler ESP32. Set point berupa level yang diinginkan oleh user dibandingkan dengan level saat ini yang berada pada vessel. Untuk mendapatkan data real level pada vessel digunakan sensor ultrasonik HC-SR04 sebagai sensing element level didalam tangki. Sensor bekerja dengan rentang maksimal sebesar 4 meter, membuatnya cukup digunakan pada vessel dengan ketinggian 1,5 meter. Setelah level pada vessel mencapai set point, miktrokontroler akan memerintahkan motor servo untuk menutup katup. Sumber air yang masuk dan keluar pada vessel berasal dari bak penampungan air yang didorong oleh pompa 12V, 22 VDC. Inlet pada vessel dihubungkan pada manual valve yang diatur pada keadaan normally open. Sehingga ketika level tangki ingin ditingkatkan, maka ESP32 akan memerintahkan pompa untuk mendorong air dari bak penampungan kepada vessel hingga mencapai level yang diinginkan. Gambar 2 memuat P&ID dari sistem pengendalian level pada penelitian ini.



Gambar 2. P&ID Pengendalian Level

Pemilihan ESP 32, selain sebagai mikrokontroler pengatur dan pembaca sensor, motor servo, dan juga motor DC, dilakukan karena ESP32 memiliki fitur Wi-fi, sehingga dapat dihubungkan ke internet, dan dapat dilakukan pembacaan secara remote. Pembacaan remote pada penelitian ini dilakukan dengan menggunakan Blynk. Alasan penggunaan Blynk, dikarenakan Blynk adalah aplikasi yang memudahkan untuk membuat UI yang digunakan untuk keperluan Human Machine Interface (HMI). Hal ini didukung oleh Blynk telah memiliki library di Arduino IDE sehingga dapat dihubungkan pada pemograman pengendalian level yang telah dibuat sebelumnya. Setelah data dibaca oleh sensor dan dikirimkan ke ESP32 untuk dilakukan pengendalian, data tersebut dikirimkan kepada cloud Blynk untuk disimpan di database. Komunikasi antara ESP32 dan Blynk menggunakan protokol komunikasi TCP/IP dengan menggunakan HTTP. Data yang telah disimpan pada cloud Blynk lalu ditampilkan pada interface yang telah dibuat oleh peneliti seperti yang ditunjukan pada gambar 3. Semua variabel penting seperti status pompa, nilai setpoint, tinggi air saat ini, dan posisi servo ditampilkan secara real-time melalui dashboard *blynk*. Dashboard ini memungkinkan pengguna memantau dan mengendalikan sistem dari jarak jauh melalui internet, termasuk menghidupkan atau mematikan pompa dan mengubah nilai setpoint. Dengan tampilan visual berupa grafik dan indikator, pengguna dapat dengan mudah mengevaluasi performa sistem dan melakukan penyesuaian bila diperlukan.

Gambar 3. Dashboard Blynk Pengendalian Level

Kontroler PID digunakan sebagai metode pengendalian utama pada penelitian ini. Secara umum, error dari sistem akan masuk kepada kontroler PID sebelum digunakan untuk menggerakan aktuator pada tangki. Output dari kontroler dihitung dengan menggunakna persamaan kontroler PID yaitu

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(t)\,dt + K_p T_d \frac{de(t)}{dt} \quad (1)$$

Untuk mendapatkan ketiga parameter K_p, T_i, dan T_d, nilai konstanta waktu (T) dan waktu tunda (L) harus dihitung untuk mendapatkan nilai dari ketiga parameter diatas berdasarkan persamaan yang telah ditetapkan metode-metode tuning kontroler PID. Konstanta waktu dan waktu tunda dapat dicari dengan menggunakan bump test. Metode ini dilakukan dengan cara memberi perubahan pada manipulated variables (MV) yang tidak terlalu besar, lalu dicatatkan perubahan process variable (PV) sebelum dan setelah dilakukan bump test.

Pada penelitian ini digunakan tiga metoda untuk melakukan tuning PID controller. Tiga metode tersebut adalah Cohen Coon, Ziegler Nichols dan Internal Model Control (IMC). Metode Cohen-Coon adalah salah satu metode tuning PID yang digunakan untuk menentukan parameter kontroler PID (Kp, Ti, Td) berdasarkan model proses sistem orde satu dengan waktu tunda (*first-order plus dead time* / FOPDT). Metode ini menggunakan data dari *step response* sistem, yaitu gain proses (K), waktu tunda (L), dan konstanta waktu (T) untuk menghitung parameter PID agar sistem memiliki respons yang cepat dan stabil.

Tabel 1. Metode Cohen Coon

Tipe	Kp	Ti	Td
P	$\frac{1}{K_c}.\left(\frac{T}{L}\right)\left(1+\frac{1}{3}\times\frac{L}{T}\right)$		
PI	$\frac{1}{K_c}.\left(\frac{T}{L}\right)\left(0{,}9+\frac{1}{12}\times\frac{L}{T}\right)$	$L.\left(\frac{30}{9}+\frac{3}{20}\times\frac{\frac{L}{T}}{\frac{L}{T}}\right)$	
PD	$\frac{1}{K_c}.\left(\frac{T}{L}\right)\left(\frac{5}{4}+\frac{1}{6}\times\frac{L}{T}\right)$		$L.\left(\frac{6-2\,(\frac{L}{T})}{22+3\,(\frac{L}{T})}\right)$
PID	$\frac{1}{K_c}.\left(\frac{T}{L}\right)\left(\frac{4}{3}+\frac{L}{4T}\right)$	$L.\left(\frac{32}{13}+\frac{6}{8}\times\frac{\frac{L}{T}}{\frac{L}{T}}\right)$	$\frac{4L}{11+2\times\frac{L}{T}}$

Serupa dengan Metode Cohen Coon, Metode Open-Loop Ziegler-Nichols adalah salah satu metode tuning PID yang berdasarkan pada pengukuran time constant (T), dan dead time (L). Hal yang membedakan Open-Loop Ziegler-Nichols dan Cohen Coon adalah respon yang diharapkan pada penggunaan kedua metode tuning tersebut. Metode Cohen Coon dikenal dengan output yang stabil, sedangkan Metode Ziegler-Nichols digunakan untuk mendapatkan respons yang lebih cepat namun lebih agresif

Tabel 2. Metode Open-Loop Ziegler Nichols

Tipe	Kp	Ti	Td
P	$\frac{T}{L}$	∞	-
PI	$\frac{0{,}9\,T}{L}$	$\frac{L}{0{,}3}$	-
PD	$\frac{1{,}2\,T}{L}$	-	$0.5L$
PID	$\frac{1{,}2\,T}{L}$	$2L$	$0.5L$

Metode IMC (Internal Model Control) adalah pendekatan tuning PID yang didasarkan pada model matematis dari sistem yang dikendalikan. Tujuan utama dari IMC adalah untuk menghasilkan sistem kendali yang stabil, robust, dan mudah diatur, dengan mempertimbangkan ketidaksempurnaan model dan gangguan eksternal.

Tabel 3. Metode Internal Model Control

Tipe	Kp	Ti	Td
P	$\frac{T}{K\,(L+\lambda)}$	-	-
PI	$\frac{T}{K\,(L+\lambda)}$	$T_i = T$	-
PD	$\frac{2T+L}{K\,(L+2\lambda)}$	-	$\frac{T+L}{(L+2\lambda)}$
PID	$\frac{2T+L}{K(L+2\lambda)}$	$\frac{L}{2}+T$	$\frac{T+L}{(L+2\lambda)}$

3. PEMBAHASAN

A. Pemodelan Proses Level

Fungsi alih dari sistem pengendalian level air dalam tangki diperoleh melalui proses identifikasi menggunakan metode *bump test.* Berikut adalah respon dari sistem ketika Bump Test dilakukan, seperti yang ditunjukan oleh Gambar 4:

Gambar 4. Hasil Bump Test Sistem Pengendalian Level

Dalam pengujian ini, Process Variable (PV) awal berada dalam kondisi stabil pada ketinggian 40 cm, dengan kondisi bukaan awal servo valve sebesar 70%. Untuk memicu respon dinamis sistem, dilakukan pemberian gangguan berupa perubahan mendadak (input step) terhadap sinyal kontrol atau Manipulated Variable (MV), dengan menurunkan bukaan servo valve dari 70% menjadi 50%, atau terjadi penurunan sebesar 20%. Dengan berkurangnya bukaan katup, laju pengeluaran air dari tangki menurun, sementara laju pemasukan tetap, sehingga volume air di dalam tangki meningkat dan menyebabkan kenaikan level air. Setelah dilakukan perubahan Manipulated Variable tersebut, didapatkan, sistem dengan MV sebeasar 50% berada pada kondisi stabil dengan ketinggian 58 cm, sehingga memiliki perbedaaan range Process Variable (ΔPV) sebesar 18 cm.

Proses Variable baru mulai merespon terjadinya perubahan variable pada $t_{start} = 15$ detik dengan waktu tunda antara sistem dengan MV 70% hingga diturunkan pada MV 50% sebesar 4 detik. Lalu, kenaikan sebesar 63.2% dari Process Variable baru sebesar 51.4 cm. Waktu yang diperlukan untuk mencapai ketinggian ($t_{63,2}$) tersebut sebesar 86 detik. Selisih antara waktu mulai dengan $t_{63,2}$ merupakan time constant dengan $t_{start} = 71$ detik.

Selanjutnya, parameter gain proses (K), dead time (L), dan Time Constant (T) digunakan untuk menghitung nilai Kp, Ti, dan Td berdasarkan metode tuning masing-masing pada konfigurasi kontroler P, PI, PD, dan PID. Berikut adalah rekap hasil berdasarkan hasil Bump Test sistem pengendalian level:

Tabel 4. Parameter Sistem Pengendalian Level

Parameter	Nilai
Range Process Variable (ΔPV)	18
Range Controller Output (ΔCO)	20
Dead Time (L)	4 s
63,2 % Sistem	51
$t_{63,2}$	86 s
t_{start}	15 s
Time Constant (T)	71 s
Gain (K_c)	0,9

B. Pengujian Metode Cohen Coon

Tabel 5. Nilai Kp, Ti, dan Td pada Metode Tuning Cohen-Coon

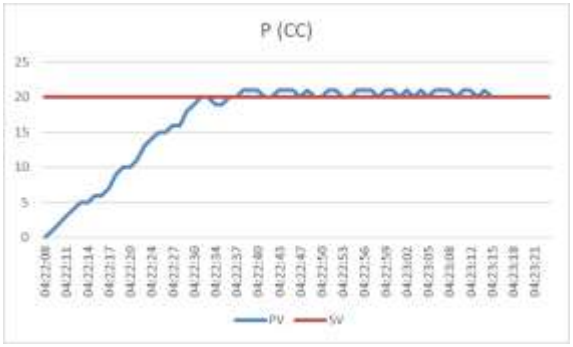
Tipe	Kp	Ti	Td
P	20,90	-	-
PI	17,84	11,916	
PD	24,8334	-	1,06228
PID	26,563	12,846	1,4

Kontroler P mampu mencapai setpoint dengan waktu rise time sebesar 18 s dan mencapai kestabilan pada set point dengan waktu settling time sebesar 64 detik. Overshoot yang dihasilkan dengan menggunakan kontroler P sebesar 5%. Lalu, Kontroler PI menghasilkan respons yang lebih cepat dibandingkan dengan kontroler P dengan rise time sebesar 17 detik dan rise time sebesar 62 detik. Kontroler PI juga memiliki overshoot yang sama dengan kontroler P sebesar 5%. Kemudian, penggunaan kontroler PD menghasilkan waktu rise time

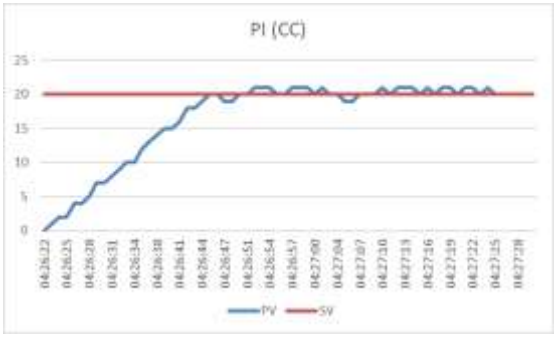
yang lebih cepat dibandingkan kedua kontroler sebelumnya dengan rise time sebesar 16 detik, namun membutukan waktu untuk mencapai kestabilan yang lebih lama sebesar 72 detik, dengan overshoot yang lebih besar yaitu 10%. Dan terakhir kontroler PID menghasilkan respon transien yang lebih lambat dibandingkan kontrol P dan PI dengan rise time sebesar 19 detik dan settling time sebesar 19 detik. Kesimpulan dari pengujian tuning PID menggunakan metode Cohen-Coon pada pengendalian level adalah penggunaan kontroler PI merupakan kontroler dengan respon transien terbaik pada metode tuning Cohen-Coon. Respon dari tiap kontroler PID dengan tuning Cohen-Coon dapat dilihat di gambar 5.

Tabel 6. Respons Transien pada PID Controller dengan Tuning Cohen-Coon

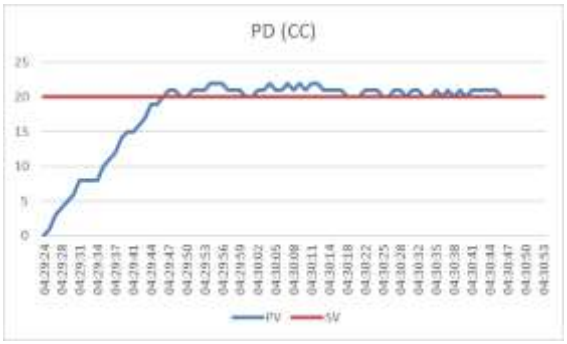
Tipe	P	PI	PD	PID
Settling Time (s)	64s	62s	72s	65s
Rise Time (s)	18s	17s	16s	19s
Overshoot max (%)	5%	5%	10%	5%



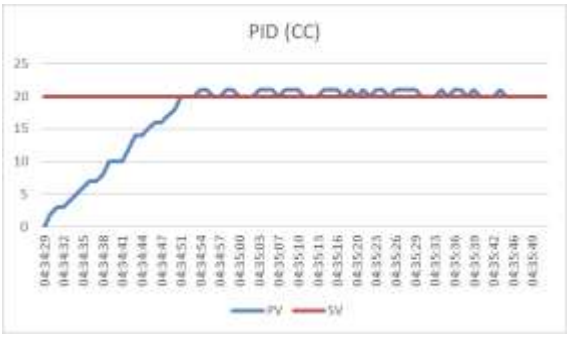
(a)



(b)



(c)



(d)

Gambar 5. Respons Transien pada PID Controller dengan Tuning Cohen-Coon

(a) Respons dengan Konfigurasi P
(b) Respons dengan Konfigurasi PI
(c) Respons dengan Konfigurasi PD
(d) **Respons dengan Konfigurasi PID**

C. Pengujian Metode Open Loop Ziegler Nichlos

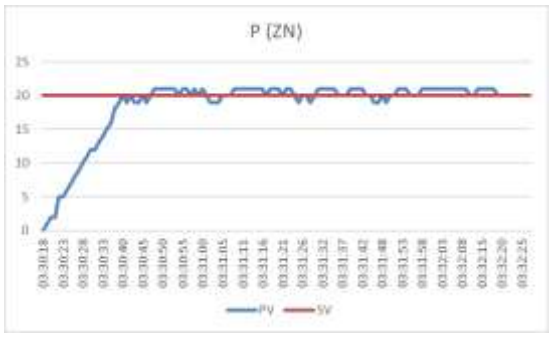
Tabel 7. Nilai Kp, Ti, dan Td pada Metode Tuning Open Loop Ziegler Nichols

Tipe	Kp	Ti	Td
P	17,75	-	-
PI	15,975	13,33	
PD	21,3	-	2
PID	21,3	8	2

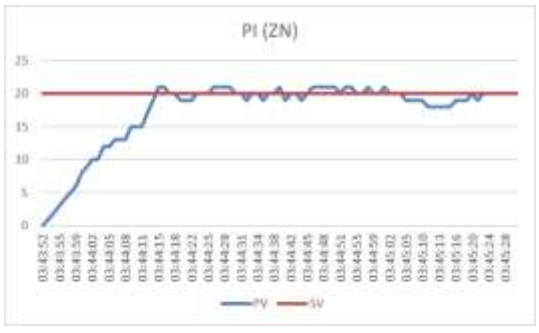
Pada percobaan dengan metode Open Loop Ziegler Nichols, dilakukan pengujian dengan empat jenis konfigurasi, yaitu P, PI, PD, dan PID. Dapat dilihat dari gambar 3, bahwa keempat konfigurasi tersebut mampu mencapai kestabilan. Kontroler P mampu mencapai setpoint dengan waktu rise time sebesar 15 s dan mencapai kestabilan pada set point dengan waktu settling time sebesar 117 detik. Overshoot yang dihasilkan dengan menggunakan kontroler P sebesar 5%. Lalu, Kontroler PI menghasilkan rise time yang lebih lambat dibandingkan kontroler P dengan rise time sebesar 18 detik, namun mampu kestabilan dengan waktu yang lebih cepat yaitu 78 detik. Kontroler PI juga memiliki overshoot yang sama dengan kontroler P sebesar 5%. Kemudian, penggunaan kontroler PD menghasilkan waktu rise time yang lebih cepat dibandingkan kontroler PI dengan rise time sebesar 17 detik, namun membutukan waktu untuk mencapai kestabilan yang lebih lama sebesar 123 detik, dengan overshoot yang lebih besar yaitu 10%. Dan terakhir kontroler PID menghasilkan respon transien yang lebih cepat dibandingkan kontrol P dan PI dengan rise time sebesar 15 detik dan settling time sebesar 79 detik. Kesimpulan dari pengujian tuning PID menggunakan metode Open Loop Ziegler Nichols pada pengendalian level adalah penggunaan kontroler PID merupakan kontroler dengan respon transien terbaik pada metode tuning Open Loop Ziegler Nichols. Respon dari tiap kontroler PID dengan tuning Open Loop Ziegler Nichols dapat dilihat di gambar 6.

Tabel 8. Respons Transien pada PID Controller dengan Tuning Open Loop Ziegler Nichols

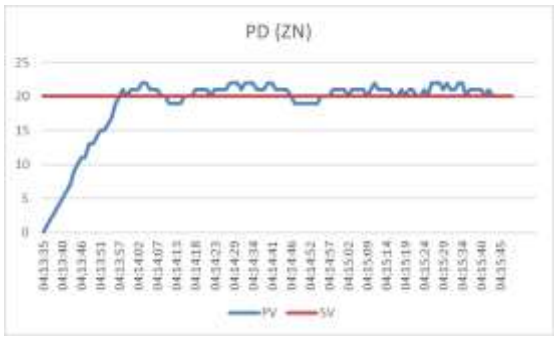
Tipe	P	PI	PD	PID
Settling Time (s)	117s	78s	123s	79s
Rise Time (s)	15s	18s	17s	15s
Overshoot max (%)	5%	5%	10%	5%



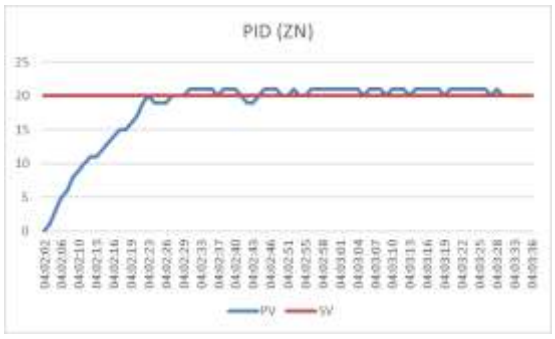
(a)



(b)



(c)



(d)

Gambar 6. Respons Transien pada PID Controller dengan Tuning Open Loop Ziegler Nichols

(a) Respons dengan Konfigurasi P
(b) Respons dengan Konfigurasi PI
(c) Respons dengan Konfigurasi PD
(d) **Respons dengan Konfigurasi PID**

D. Pengujian Metode IMC

Pada pengujian kontroler PID dengan metode tuning IMC, nilai lambda (λ) yang digunakan adalah λ = 10. Hal ini didasarkan untuk mendapatkan respons yang cepat, respons lambda berada pada rentang $0{,}1T < \lambda < 0{,}5T$. Berdasarkan nilai λ ,nilai time constant (T), dead time (L), dan gain (K_c) yang didapatkan dari Bump Test, dapat ditentukan nilai proportional gain (K_p), integral time (T_i), dan derivative time (T_d) untuk Tuning PID Controller dengan Metode IMC sebagai berikut:

Tabel 9. Nilai Kp, Ti, dan Td pada Metode Tuning IMC

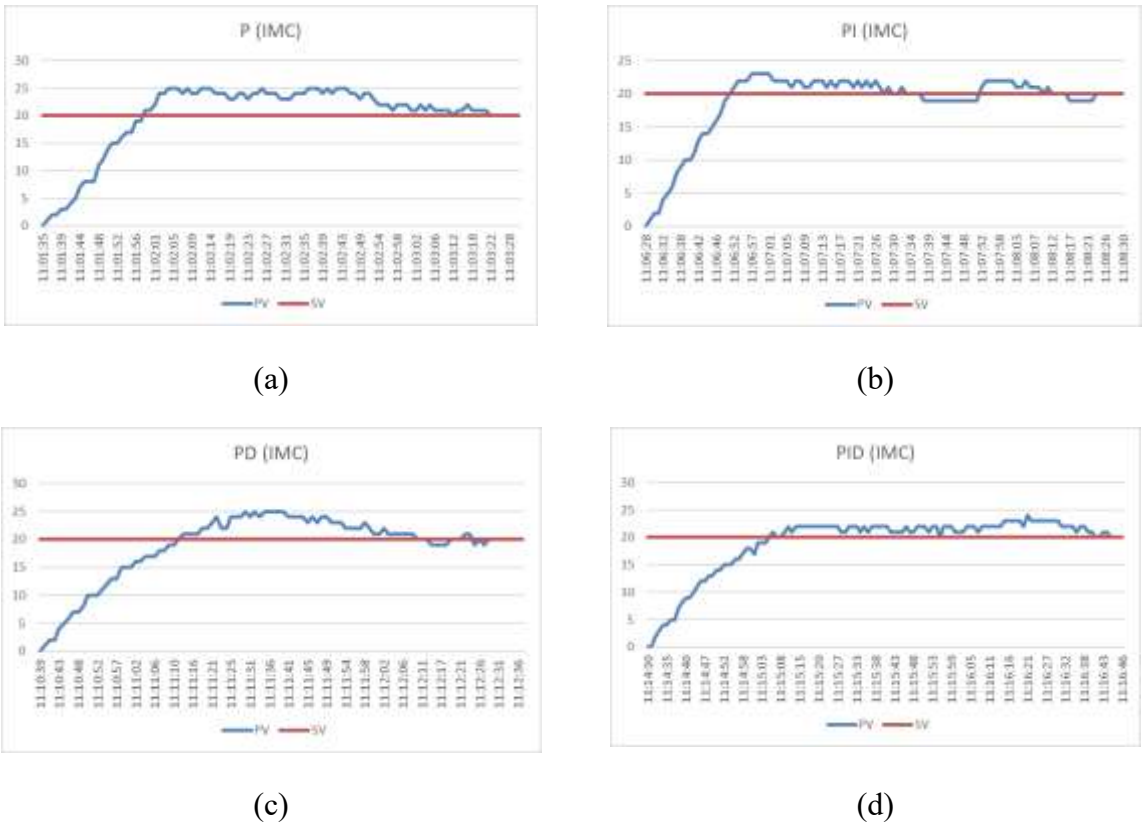
Tipe	Kp	Ti	Td
P	5,63	-	-
PI	5,63	71	
PD	6,76	-	11,83
PID	6,76	73	11,83

Secara umum, semua konfigurasi memiliki overshoot yang lebih besar dibandingkan dengan kedua metode tuning sebelumnya. Hal ini selaras dengan penggunaan λ yang berada pada rentang yang telah ditentukan, sehingga sistem menjadi agresif dan mengakibatkan overshoot menjadi besar. Kontroler P mampu mencapai setpoint dengan waktu rise time sebesar 17 s dan mencapai kestabilan pada set point dengan waktu settling time sebesar 97 detik. Overshoot yang dihasilkan dengan menggunakan kontroler P sebesar 25%. Lalu, Kontroler PI menghasilkan rise time yang lebih cepat dibandingkan kontroler P dengan rise time sebesar 15 detik, namun mampu kestabilan dengan waktu yang lebih lambat yaitu 103 detik. Kontroler PI juga memiliki overshoot yang lebih rendah dibandingkan dengan kontroler P sebesar 15%. Kemudian, penggunaan kontroler PD menghasilkan waktu rise time yang lebih lambat dibandingkan kontroler PI dengan rise time sebesar 23 detik, namun membutukan waktu untuk mencapai kestabilan yang lebih cepat sebesar 95 detik, dengan overshoot yang lebih besar yaitu

25%. Dan terakhir kontroler PID menghasilkan respon transien yang lebih lambat dibandingkan kontrol P dan PI dengan rise time sebesar 26 detik dan settling time sebesar 123 detik. Kesimpulan dari pengujian tuning PID menggunakan metode IMC pada pengendalian level adalah penggunaan kontroler PI merupakan kontroler dengan respon transien terbaik pada metode tuning IMC dengan rise time dan overshoot yang lebih rendah dibandingkan dengan kontroler lainnya. Respon dari tiap kontroler PID dengan tuning Cohen-Coon dapat dilihat di gambar 7.

Tabel 10. Respons Transien pada PID Controller dengan Tuning IMC

Tipe	P	PI	PD	PID
Settling Time (s)	97s	103s	95s	123s
Rise Time (s)	17s	15s	23s	26s
Overshoot max (%)	25%	15%	25%	20%



(a) (b)

(c) (d)

Gambar 7. Respons Transien pada PID Controller dengan Tuning IMC

(a) Respons dengan Konfigurasi P
(b) Respons dengan Konfigurasi PI
(c) Respons dengan Konfigurasi PD
(d) **Respons dengan Konfigurasi PID**

E. Perbandingan Metode Cohen-Coon, Metode Ziegler Nichols dan Metode IMC

Secara umum, Metode Cohen Coon memberika respon transien terbaik dibandingkan dengan metode tuning lainnya pada pengendalian level di penelitian ini. Hal ini dapat dilihat dari rise time antara ketiga metode tuning memiliki waktu yang kurang lebih sama, namun dapat memberikan waktu kestabilan yang jauh lebih cepat dan overshoot yang lebih rendah dibandingkan IMC, dan waktu kestabilan yang lama pada Ziegler Nichols

Salah satu alasan pengaplikasian metode Cohen Coon lebih baik pada pengendalian level di penelitian ini karena sistem ini memiliki waktu tunda (L) yang jauh lebih kecil dibandingkan konstanta waktu (T). Metode Cohen Coon mampu mencapai kestabilannya pada waktu yang lebih cepat dibandingkan Ziegler Nichols pada rasio T/L besar. Hal ini disebabkan oleh pada rasio T/L besar, Ziegler Nichols cenderung untuk membuat sistem menjadi lebih agresive sehingga mencapai kestabilan dalam waktu yang lebih lama. Hal ini didukung oleh hasil dari rise time yang didapatkan oleh metode Ziegler Nichols memiliki waktu sekitar empat detik lebih cepat dibanding metode Cohen Coon. Namun sebagai kerugian dari penggunaan metode Ziegler Nichols, waktu kestabilan menjadi lebih lama. Sedangkan pada pengimplentasian metode tuning IMC, lambda yang digunakan sangat kecil dengan dead time yang juga kecil, menjadikan sistem terlalu agresif sehingga sulit untuk mencapai kestabilan dan memiliki overshoot yang lebih besar dibandingkan kedua metode tuning lainnya.

Selanjutnya, konfigurasi kontroler PI pada metode Cohen Coon memiliki respon transien yang lebih baik dibandingkan konfigurasi lainnya. Hal ini disebabkan oleh penggunaan elemen derivative pada kontroller menyebabkan kontroler lebih sensitif terhadap noise, terutama pada dead time yang rendah, sehingga sistem membutuhkan waktu yang lebih lama untuk naik diakibatkan oleh usaha untuk meredam noise tersebut sebelum akhirnya stabil.

Tabel 11. Perbandingan Metode Cohen-Coon, Metode Ziegler Nichols dan Metode IMC

Parameter	Cohen Coon				Ziegler Nichols				IMC			
	P	PI	PD	PID	P	PI	PD	PID	P	PI	PD	PID
Rise Time (s)	18	17	16	19	15	18	17	15	17	15	23	26
Settling Time (s)	64	62	72	65	117	78	123	79	97	103	95	123
Overshoot (%)	5	5	10	5	5	5	10	5	25	15	25	20

4. SIMPULAN

Pada percobaan pengendalian level dengan menggunakan kontroler PID ini, digunakan ESP32 sebagai mikrokontroler yang menentukan level pada vessel dengan mengatur bukaan pada servo valve, yang berfungsi sebagai aktuator. Optimasi kontroler PID yang dilakukan pada percobaan ini metode tuning Cohen-Coon, Open Loop Ziegler Nichols, dan IMC. Semua metode tuning diujikan pada empat konfigurasi kontroler PID, yaitu P, PI, PD, dan PID. Berdasarkan respon transien yang diberikan pada setiap metode tuning dan konfigurasi, didapatkan bahwa respon transien terbaik berada pada metode Cohen-Coon dengan kontroler PI, dengan kecepatan respon rise time sebesar 17 detik, kestabilan settling time sebesar 62 detik, dan kehalusan respon overshoot sebesar 5%.

5. DAFTAR PUSTAKA

[1] P. Rajalakshmi, M. Ponni Bala, C. Aravind, and S. J. Suji Prasad, “Monitoring and Control of Cooler Tank Level Measurement In Gas Cleaning Plant (GCP),” *IOP Conf Ser Mater Sci Eng*, vol. 995, no. 1, p. 012039, Nov. 2020, doi: 10.1088/1757-899X/995/1/012039.

[2] S. Sünkün, B. O. Parlak, and T. B. Akca, "Level Control of a Crude Oil Tank using Fuzzy-PID Controller," in *INTERNATIONAL ARTIFICIAL INTELLIGENCE & DATA PROCESSING SYMPOSIUM*, 2023.

[3] W. L. Luyben, "Liquid level control: Simplicity and complexity," *J Process Control*, vol. 86, pp. 57–64, Feb. 2020, doi: 10.1016/j.jprocont.2019.12.008.

[4] S. Faradisa and P. W. Rusimamto, "PERANCANGAN KONTROLER PI DENGAN METODE TUNING COHEN-COON UNTUK KENDALI SUHU PADA INKUBATOR BAYI BERBASIS LABVIEW 2014," *Jurnal Teknik Elektro*, vol. 09, no. 02, pp. 293–301, 2020.

[5] M. Huba, S. Chamraz, P. Bistak, and D. Vrancic, "Making the PI and PID Controller Tuning Inspired by Ziegler and Nichols Precise and Reliable," *Sensors*, vol. 21, no. 18, p. 6157, Sep. 2021, doi: 10.3390/s21186157.

[6] M. Ali, A. N. Afandi, A. Parwati, R. Hidayat, and C. Hasyim, "DESIGN OF WATER LEVEL CONTROL SYSTEMS USING PID AND ANFIS BASED ON FIREFLY ALGORITHM," *JEEMECS (Journal of Electrical Engineering, Mechatronic and Computer Science)*, vol. 2, no. 1, Feb. 2019, doi: 10.26905/jeemecs.v2i1.2804.

[7] P. Jiang, "Summary of PID Control System of Liquid Level of a Single-Capacity Tank," *J Phys Conf Ser*, vol. 1865, no. 2, p. 022061, Apr. 2021, doi: 10.1088/1742-6596/1865/2/022061.

[8] T. Dlabač, S. Antić, M. Ćalasan, A. Milovanović, and N. Marvučić, "Nonlinear Tank-Level Control Using Dahlin Algorithm Design and PID Control," *Applied Sciences*, vol. 13, no. 9, p. 5414, Apr. 2023, doi: 10.3390/app13095414.

[9] S. Bhattacharya, M. Chakraborty, P. K. Sadhu, and B. Bhadra, "Study of A PID Control System for Liquid Level Measurement Using Non Contact Type Sensor," in *Proceedings of International Conference on Computational Intelligence & IoT*, 2018, pp. 555–559.

[10] A. K. Shah, M. Anilkumar, and N. Parikh, "Performance Analysis of IMC based PID Controller Tuning on Approximated Process Model," *NIRMA UNIVERSITTY JOURNAL OF ENGINEERING AND TECHNOLOGY*, vol. 1, no. 2, pp. 51–54, 2010.

[11] I. R. Imaduddin, "Design of Water Level Control in Tank Based on Fuzzy Logic," *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 2, no. 3, p. 145, Jan. 2021, doi: 10.12928/biste.v2i3.3307.

[12] B. B. V and H. G. Chandrakanth, "An Intelligent Approach for Adaptive Liquid Level Control System Using Machine Learning," *Int J Res Appl Sci Eng Technol*, vol. 10, no. 10, pp. 1489–1502, Oct. 2022, doi: 10.22214/ijraset.2022.47214.

[13] M. M. Noel and B. J. Pandian, "Control of a nonlinear liquid level system using a new artificial neural network based reinforcement learning approach," *Appl Soft Comput*, vol. 23, pp. 444–451, Oct. 2014, doi: 10.1016/j.asoc.2014.06.037.

[14] L. Mukarromah, P. D. Permatasari, and E. Tridianto, "Modeling of Level Control System using IMC-PID Tuning Method with Filter Tuning Factor Variation on Steam Drum Boiler of Power Plant," in *2022 14th International Conference on Information Technology and Electrical Engineering (ICITEE)*, IEEE, Oct. 2022, pp. 36–41. doi: 10.1109/ICITEE56407.2022.9954108.

[15] M. Saad, M. Alshara, and K. Mustafa, "Fuzzy PID Controller Design for a Coupled Tank Liquid Level Control System," *WSEAS TRANSACTIONS ON SYSTEMS AND CONTROL*, vol. 18, pp. 401–408, Nov. 2023, doi: 10.37394/23203.2023.18.42.