Evaluasi Kinerja Kontrol PI Ziegler-Nichols untuk Pengendalian Level Tangki Air Menggunakan Arduino dan Sensor Ultrasonik

Asepta Surya Wardhana[1], Zafi Azam Maindra[2], Rindu Restu Adinda[3]

[1, 2, 3]Teknik Instrumentasi Kilang, Politeknik Energi dan Mineral Akamigas
[2]Calibration Department, PT. Wifgasindo Dinamika Instrument Engineering
Email: aseptasw@esdm.go.id

Abstract. *Water tanks are needed by the oil and gas industry, generally functioned as a storage feed for subsequent processes. Problems that often occur in water tanks are continuous processes and rapid changes in set points so that level control is needed in the tank to achieve water stability. Control systems generally have components, that is controller, final control element, and transducers. The level control prototype is implemented using Arduino UNO as microcontroller, final control element uses a servo motor, and ultrasonic distance sensor as level sensor. Arduino is widely used as a microcontroller because it is simple and inexpensive to use, as well as servo motors and ultrasonic distance sensors. In this study, PID control with parameters calculated based on the Ziegler-Nichols method was used and a PI controller was selected as the result of the first tuning calculation. The calculated PID parameters are, Kp = 2,96, Ki = 0,44, and Kd = 0. Test results on the prototype show that response time, are time constant = 21,4 seconds, delay time = 1 second, rise time = 38,2 seconds, peak time = 55,4 seconds, settling time = 131 seconds, and overshoot = 7,93 %.*
Keywords: *Tanks, Arduino, Servo, Ultrasonic, Ziegler-Nichols, PID.*

Abstrak. *Tangki air sangat dibutuhkan oleh industri migas, pada umumnya berfungsi sebagai tempat penampungan feed dari beberapa proses selanjutnya. Permasalahan yang sering terjadi pada tangki air adalah proses yang kontinu dan perubahan set point secara cepat sehingga dibutuhkan pengendalian level pada tangki agar tercapai kestabilan air. Sistem pengendalian pada umumnya memiliki komponen-komponen, yaitu controller, final control element, dan transducer. Prototipe pengendalian level, diimplementasikan dengan memanfaatkan Arduino UNO sebagai microcontroller, final control element memanfaatkan motor servo, dan ultrasonic distance sensor sebagai sensor level. Arduino banyak digunakan sebagai salah satu microcontroller dikarenakan mudah dan murah penggunaannya, begitu pula dengan motor servo dan ultrasonic distance sensor. Pada penelitian ini menggunakan PID kontrol dengan parameter yang dihitung berdasarkan metode Ziegler-Nichols dan dipilih PI controller hasil perhitungan tuning pertama. Parameter PID hasil perhitungan, yaitu Kp = 2,96, Ki = 0,44, dan Kd = 0. Hasil pengujian pada prototipe menunjukan respon waktu rata-rata, yaitu time constant = 21,4 detik, delay time = 1 detik, rise time = 38,2 detik, peak time = 55,4 detik, settling time = 131 detik, dan overshoot = 7,93 %.*
Kata Kunci: *Tangki, Arduino, Servo, Ultrasonic, Ziegler-Nichols, PID.*

1. Pendahuluan

Tangki air sangat dibutuhkan oleh industri migas, pada umumnya berfungsi sebagai tempat penampungan *feed* dari beberapa proses selanjutnya. Sebagai contoh uap panas hasil pemanasan oleh *boiler* dari penelitian Prajodan, dkk (Prajodan et al., 2022) yang membutuhkan *feed* berupa air memiliki beberapa fungsi, seperti untuk menjaga suhu *feed crude oil* melalui *heat exchanger* (Sujono et al., 2020; Wardhana et al., 2021), dapat dimanfaatkan untuk penggerak turbin pembangkit atau *steam turbine generator*, atau juga proses penguapan pada *evaporator*. Di dalam tangki air akan terus mengisi tanpa henti, dikarenakan pompa *feed water* diatur agar berhenti ketika terjadi gangguan/*disturbance* pada proses *inlet*. Oleh karena itu dibutuhkan pengendalian *level* pada tangki, agar tercapai kestabilan air. *Control valve* sebagai *final control element* (Asraf et al., 2017) akan dipasang pada pipa *outlet*, yaitu tepatnya sesudah pompa *drain water*. *Control valve* (Mahardhika et al., 2017) akan memiliki aksi *fail close* (FC), ini dikarenakan untuk tujuan *safety*. Ketika terjadi gangguan/*disturbance* pada *level*, *control valve* akan menutup sehingga tidak ada air yang keluar dari dalam tangki.

Penelitian mengenai proses di dalam tangki telah banyak dilakukan, salah satunya yaitu percobaan mengenai pemodelan *heat pump water heater* pada tangki air (Wenzhe and Hrnjak, 2018). Penelitian ini menjelaskan model transient sistem performa HPHW pada tangki air. Dengan model *numerical* untuk memprediksi pergerakan sistem dan model CFD yang menjelaskan *flow* dan temperatur pada tangki air. Pengendalian PID (Ravy et al., 2021) akan diperlukan, karena dapat meningkatkan kualitas produk yang diinginkan. Pengendalian yang direkomendasikan berupa rangkaian *cascade* untuk menangani aliran *diesel* karena dapat meningkatkan performa sistem pengendalian. Dalam beberapa jurnal penelitian membahas mengenai *tuning* PID dengan metode *Ziegler-Nichols* (Yasmin and Wardhana, 2022) dan properti kontrol PID (Gozali, 2015) yang menghasilkan kinerja yang handal untuk pengendalian proses. Hasil penelitian menunjukkan bahwa terdapat korelasi yang sangat baik antara parameter kontrol dan parameter proses yang dikendalikan.

Berdasarkan penelitian sebelumnya maka pada perancangan ini menggunakan metode *Ziegler-Nichols* untuk mendapatkan parameter pengendalian PID, menggunakan *LabView* yang diintegrasikan dengan pengendalian sebagai *monitoring level* tangki, dan untuk memperoleh kestabilan air di dalam tangki, penulis ingin membuat rancang bangun suatu prototipe pengendalian proses pada tangki. Hal tersebut mendasari penulis untuk mengambil judul "Rancang Bangun Prototipe Pengendalian dan *Monitoring Level* Pada Tangki Berbasis PID *Ziegler-Nichols*".

2. Tinjauan Pustaka

Penelitian terkait mengenai suatu tangki telah banyak dilakukan, beberapa yaitu mengenai pengendalian sistem tangki air dengan model matematika dan percobaan mengenai pemodelan *heat pump water heater* pada tangki air (Wenzhe and Hrnjak, 2018). Hasil penelitian oleh Belikov merujuk pada penggunaan *neural network* untuk memodelkan proses secara nyata. Parameter dari hasil pemodelan digunakan juga untuk mendesain pengendalian. Sistem tangki level cairan dipilih sebagai bahan studi untuk mengilustrasikan penerapan pendekatan yang digunakan. Hasil percobaan menunjukkan performa yang bagus pada pendekatan yang telah digunakan. Pengendalian yang didesain dapat merespon terhadap perubahan level pada semua *set point* yang dipercobakan pada tingkatan akurasi yang tinggi dan tanpa menimbulkan *over/undershoot*. Penelitian oleh Li menjelaskan model *transient* sistem performa HPHW pada tangki air. Dengan model *numerical* untuk memprediksi pergerakan sistem dan model CFD yang menjelaskan *flow* dan temperatur pada tangki air. Deviasi kapasitas antara pengukuran dan prediksi model adalah 0,9 % untuk *condenser* dan 2 % untuk *evaporator*. Hasil kerja *compressor* telah diprediksi dengan rata-rata *discrepancy* 2,2 % dan 2,9 % untuk HPF pada HPHW. Suhu refrigerant juga diprediksi dengan discrepancy ±0,5 °C pada *inlet* dan *outlet evaporator* sedangkan pada *condenser* sebesar ±3 °C. Model tangki air CFD memprediksi respon *transient* suhu air pada tempat yang berbeda dengan *error* kurang dari 2 °C.

Sistem pengendalian dengan PID konvensional sangat banyak digunakan di industri migas dikarenakan kemudahan dalam penggunaan dan perancangannya dengan perbandingan antara *controller* PID (Åström et al., 2004) konvensional dan *Fuzzy control*. Penelitian tersebut menyebutkan bahwa pengendalian PID yang digabungkan dengan *self-tuning fuzzy* menghasilkan performa yang lebih baik dan pengendalian dengan presisi yang lebih tinggi.

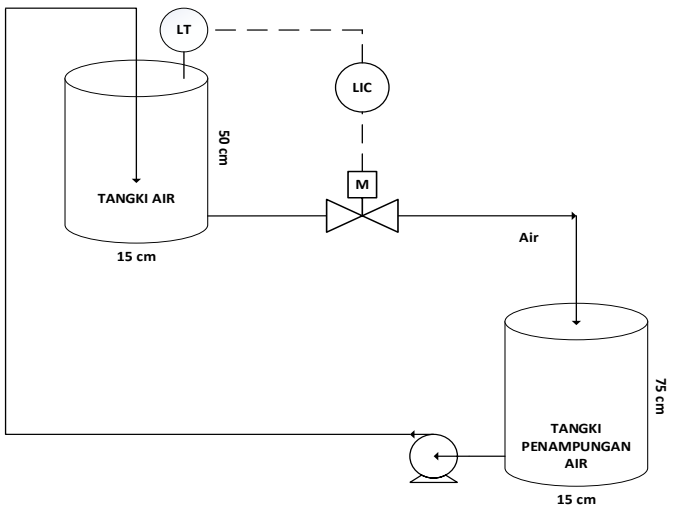
Salah satu tuning pengendalian PID konvensional adalah dengan berdasarkan metode *Ziegler-Nichols* yang telah diteliti (Yasmin and Wardhana, 2022). Berdasarkan aturan tuning yang dijelaskan oleh *Ziegler-Nichols* (Ogata, 2010), perhitungan juga dilakukan melalui MATLAB. Prosedur tuning didapat dengan melakukan analisa secara matematis. Cara ini juga sangat cocok untuk melakukan *tuning* PID secara otomatis. Metode *tuning* PID *controller* berdasarkan *Ziegler-Nichols*, sangat bagus dan mudah dalam perancangannya.

Penelitian mengenai penerapan *Arduino* sebagai *microcontroller* telah banyak dilakukan. Beberapa yaitu mengenai desain PID *controller* dengan *Arduino Mega* 2560 pada motor DC magnet permanen dengan gangguan (Saputra et al., 2022) dan penggunaan sensor tekanan dengan kecepatan motor berhasil dengan error rata-rata 1,3% (Rohiem and Putra, 2021). Implementasi pengendalian kecepatan motor DC magnet permanen berbasis *Arduino Mega* 2560 dan L298N H-*Bridge* sebagai eksperimen menggunakan metode kontrol PID telah divalidasi melalui percobaan secara langsung.

Performa kontrol yang baik, ketahanan, dan stabilitas keseluruhan dapat diharapkan dengan menggunakan *controller* PID, meskipun memiliki respon *rise time* yang lebih rendah. Kemudian penelitian mengenai pengendalian tertutup kecepatan motor DC berbasis syaraf tiruan menggunakan *Arduino* UNO (Pamuji et al., 2021) mendapatkan hasil yang memuaskan. Mereka menjelaskan perancangan dan implementasi sistem pengendalian kecepatan motor DC berbasis *Arduino* UNO menggunakan pengontrol *multilayer neural network* dan PID. Kecepatan motor DC dikontrol dengan dengan memvariasikan *duty cycle* dari sinyal PWM yang diumpankan ke gerbang *mosfet* irf640. Keluaran sistem juga sesuai dengan hasil teoritis yang menunjukkan akurasi sistem yang lebih baik.

3. Metode Penelitian

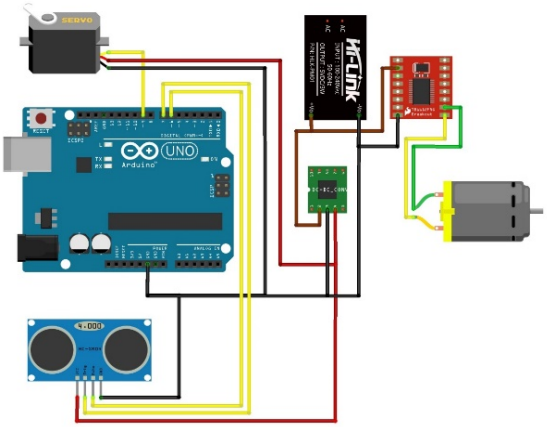
Dalam penelitian ini beberapa metode dilakukan seperti melakukan perancangan prototipe untuk *hardware* dan *software*. Perancangan prototipe *hardware* meliputi, pembuatan tangki pengendalian dan penampungan, pemasangan pipa jaringan, sistem pengkabelan, dan pemasangan *servo valve*. Perancangan prototipe *software* meliputi, pemrograman *Arduino* (Wirasanto and Widodo, 2021; Zainul et al., 2022) sebagai *controller* dan akuisisi data dengan *LabView* (Pangya et al., 2018). Kemudian dilakukan uji coba pada prototipe dan hasil pengujian yang telah sesuai, akan dilakukan analisa terhadap grafik respon perubahan variabel proses, meliputi kestabilan *level* dengan membandingkan hasil respon secara aktual dengan simulasi *MatLab* (Ali, 2004; Elya et al., 2022).



Gambar 1. P&ID Prototipe

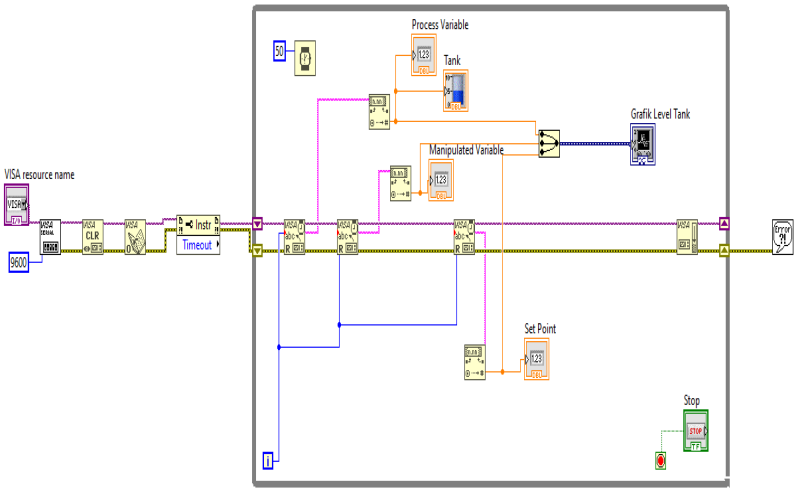
Rancangan prototipe tangki dapat dilihat pada P&ID gambar 1. Pada tangki pengendalian dibuat dua lubang pada bagian atas sebagai *inlet* dan tempat peletakkan *sensor ultrasonic* HY-SRF05, pada sisi-sisi pipa dibuat lubang dengan ketinggian 3 cm dari dasar tangki sebagai *outlet* dan 48 cm dari dasar tangki sebagai *safety*. Pada tangki penampungan dibuat dua lubang pada bagian atas sebagai *inlet* yang berasal dari *outlet* dan *safety* tangki pengendalian. Pada sisi-sisi tangki dibuat lubang *outlet*, *drain*, dan pengisian tangki. Kedua tangki juga terpasang *sight glass* yang berfungsi sebagai indikator. Pipa *outlet* tangki pengendalian dipasang *servo valve*, kemudian dihubungkan ke bagian *inlet* tangki penampungan. Pipa *safety* tangki pengendalian dipasang *manual valve* yang dapat dibuka-tutup ketika terjadi *overload*. Pada *outlet* tangki penampungan terpasang pompa yang berfungsi menghisap air menuju ke tangki pengendalian.

 ISSN: 2579-566X (Online)
ISSN: 2477-5274 (Print)



Gambar 2. Pengkabelan Komponen-komponen Pengendalian

Rancangan pengkabelan dapat dilihat pada gambar 2. Pada prototipe pengendalian dan *monitoring level*, dipasang komponen-komponen, seperti pompa, *Arduino*, *motor servo*, *power supply*, *buck converter*, *speed regulation*, dan *sensor level*. Rangkaian pengkabelan dimulai dengan memasang *power supply* yang terhubung dengan stop kontak. *Power supply* berfungsi merubah dari tegangan AC menjadi tegangan DC. Masukan *power supply* adalah sebesar 220V AC dengan keluaran sebesar 12 VDC. Keluaran *power supply* terhubung dengan *buck converter* dan *speed regulation*. *Buck converter* berfungsi menurunkan tegangan menjadi 6 VDC, sedangkan *speed regulation* yang terhubung dengan pompa berfungsi untuk mengatur kecepatan putaran. Keluaran *buck converter* terhubung dengan masukan *sensor level* dan *motor servo*, dengan arus sebesar 5 A. *Arduino* terhubung dengan PC melalui kabel *serial*, ini juga berfungsi sebagai *supply*. *Pin-pin* yang digunakan adalah *GND*, *pin* 6, *pin* 7, dan *pin* 9. *Pin* 6 dan 7 adalah masukan/keluaran *sensor level* yang bekerja dengan mengirimkan sinyal pulsa kemudian menerima pantulan sinyal dan merubahnya menjadi sinyal digital. Pin 9 adalah keluaran untuk mengatur bukaan *motor servo* melalui PWM (*Pulse With Modulation*), bukaan *servo* bergantung besar sinyal yang dikirim melalui pin 9 dan biasa disebut *manipulated variable*.



Gambar 3. *Block Diagram Monitoring Level* pada *LabView*

Block diagram untuk pembacaan variabel proses dirangkai pada gambar 3. Untuk melakukan komunikasi serial data antara *Arduino* dan *LabView*, diperlukan komponen-komponen yang dirangkai pada *layout block diagram LabView*. Komponen komunikasi yang digunakan adalah *VISA serial*, *VISA clear*, *VISA open*, *VISA Read*, dan penampil data.

Gambar 4. Prototipe Pengendalian dan *Monitoring Level*

Hasil perancangan prototipe pengendalian dan *monitoring level* berbasis PID *Ziegler-Nichols* berupa *plant* dengan skala kecil pada gambar 4. Pada prototipe ini memiliki komponen-komponen utama, yaitu tangki, pompa, *motor servo valve*, *sensor level* HC-SR04, dan *Arduino*. Rangkaian pengendalian adalah berupa *close loop* dengan *controller* PID, *motor servo valve* sebagai *final control element*, dan *ultrasonic distance sensor* HC-SR04 sebagai *sensor level* (Dewi et al., 2020). Sebelum melakukan uji keseluruhan maka metode yang dilakukan melakukan kalibarasi dan uji sensor serta perangkat secara terpisah. Kalibrasi *sensor ultrasonic* dilakukan dengan pengukuran secara langsung, dan dibandingkan dengan pembacaan jarak pada *Arduino*. Pengukuran secara langsung dilakukan dengan cara meletakan penggaris dan *sensor ultrasonic* pada 0 cm. Kemudian *sensor ultrasonic* dihubungkan dengan *Arduino* dan letakan objek didepan *sensor*. Hasil pengukuran dibandingkan dengan hasil pembacaan pada *Arduino* sehingga diperoleh *Error* rata-rata pada pengujian adalah 0,34. Kalibrasi *servo* dilakukan dengan memberikan perintah pada *Arduino* untuk mengatur putaran *servo* (Minariyanto et al., 2020). Pemberian perintah dilakukan dengan mengatur pada 0° dan 90°.

Tahap berikutnya melakukan Bump test dilakukan dengan memberikan perubahan pada *valve (manipulated variable)* pada saat *controller* dalam mode *manual*. Perubahan MV akan mempengaruhi perubahan *level* dalam kurun waktu tertentu. Data hasil bump test akan digunakan sebagai perhitungan fungsi alih. Pada percobaan ini *Bump test* dilakukan sebanyak 3 (tiga) dimana tahap dengan memberi perubahan pada MV sebesar 30° (20° hingga 50°). Perubahan ini mempengaruhi kondisi *level* air di dalam tangki, yang semula 11 % menjadi 48 %. *Bump test* yang kedua dilakukan dengan memberi perubahan pada MV sebesar 30° (30° hingga 60°). Perubahan ini mempengaruhi kondisi *level* air di dalam tangki, yang semula 42 % menjadi 68 %. *Bump test* yang ketiga dilakukan dengan memberi perubahan pada MV sebesar 30° (40° hingga 70°). Perubahan ini mempengaruhi kondisi *level* air di dalam tangki, yang semula 40 % menjadi 97 %. Hasil *bump test* mulai dari yang pertama hingga ketiga, dibuat rata-rata pengukuran untuk mendapatkan fungsi alih yang lebih akurat. Maka hasil rata-rata *bump test* didapat dirubah menjadi persamaan alih fungsi sebagai berikut :

Maka $\boldsymbol{Gs}$: $\boldsymbol{Gs} = \frac{\mathbf{1{,}33}+e^{-2s}}{\mathbf{77}s+\mathbf{1}}$

TUNING PID

Tabel 1. Metode Tuning Ziegler-Nichols

Tipe *Controller*	P_b	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0.9\frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2\frac{T}{L}$	2L	0.5L

 ISSN: 2579-566X (Online)
ISSN: 2477-5274 (Print)

Untuk mendapatkan respon perubahan *level* yang stabil, perlu dilakukan *tuning* terhadap parameter PID. *Tuning* dilakukan berdasarkan metode *Ziegler-Nichols* (Fakhrunnia et al., 2020; Yasmin and Wardhana, 2022) pada tabel 1.

Tuning dilakukan dengan melakukan perhitungan PI *controller* dan PID *controller*. Hasil perhitungan *tuning* PID pertama, kedua dan ketiga, dilihat pada tabel 2.

Tabel 2. *Tuning* PID Pertama, Kedua dan Ketiga

Parameter	*Tuning ke-1*			*Tuning ke-2*			*Tuning ke-3*		
	Kp	***Ki***	***Kd***	***Kp***	***Ki***	***Kd***	***Kp***	***Ki***	***Kd***
PI *Controller*	2,96	0,44	0	2,96	0,44	0	2,74	0,41	0
PID *Controller*	2,22	0,56	2,22	2,22	0,56	2,22	2,06	0,52	2,06

Selanjutnya dilakukan validasi pada parameter PID yang telah didapat. Dengan memasukan parameter PID ke dalam *MatLab* (Megido and Ariyanto, 2016), dapat dilihat *scope* respon pengendalian. Fungsi alih hasil *bump test* rata-rata digunakan sebagai *gain* proses pada *MatLab*. Respon waktu simulasi pada MatLab dengan memasukan parameter PID hasil perhitungan dilihat pada tabel 3. Dari hasil validasi yang pertama hingga yang ketiga, maka akan digunakan PI *controller* dari hasil *tuning* pertama pada implementasi *control Arduino* (Fakhruzzaini and Aprilianto, 2017). Dikarenakan parameter menghasilkan *settling time* paling rendah dibandingkan dengan parameter yang lain.

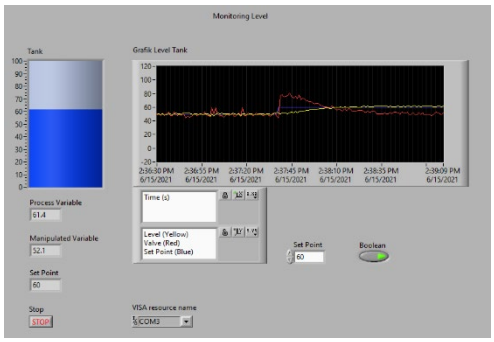
Perancangan prototipe pengendalian dan *monitoring level* pada tahap ini telah mampu melakukan pengendalian terhadap *level* air di dalam tangki. *Level* air di dalam tangki juga telah mampu termonitor melalui *LabView* menampilkan dalam bentuk persentase dan juga grafik perubahan *level*. Bukaan *valve* (*manipulated variable*) dan *set point* juga telah mampu terbaca dan ditampilkan.

Tabel 3. Respon Waktu Simulasi Proses Parameter PID

Respon Waktu	***PI₁***	***PID₁***	***PI₂***	***PID₂***	***PI₃***	***PID₃***
Time Constant	11,17 detik	11,19 detik	11,17 detik	11,19 detik	11,68 detik	11,69 detik
Delay Time	0 detik	0 detik	0 detik	0 detik	0 detik	0 detik
Rise Time	17,6 detik	16,5 detik	17,6 detik	16,5 detik	18,37 detik	17,1 detik
Peak Time	33 detik	29 detik	33 detik	29 detik	33 detik	33 detik
Settling Time	69 detik	97 detik	69 detik	97 detik	73 detik	101 detik
Overshoot	35 %	46 %	35 %	46 %	35 %	47 %

4. Hasil dan Pembahasan

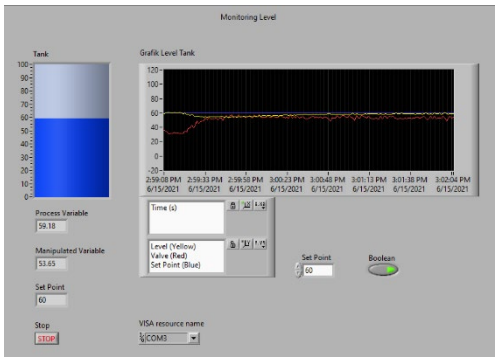
Hasil perancangan dan simulasi secara aktual, ditunjukkan pada gambar 5, menunjukan *level* air stabil pada 60 %. Perubahan *level* juga menunjukan fluktuasi yang tidak terlalu besar dari *set point* yang telah diatur. Parameter-parameter yang dimasukan ke *controller*, yaitu *Set Point* = 60 %, Kp = 2,96, Ki = 0,44, Kd = 0.



Gambar 5. *Monitoring Level* Melalui *LabView*

Hasil pembacaan menunjukan level berada pada 61,4 %, dengan toleransi kestabilan sebesar 2% dan nilai manipulated variable adalah 52,1°. Pada grafik, warna kuning adalah pembacaan level (*process variable*), warna merah adalah pembacaan bukaan *valve* (*manipulated variable*), dan warna biru adalah pembacaan *set point*.

Pengujian keandalan pengendalian dapat dilakukan dengan memberikan gangguan pada sistem, dan kemudian melakukan pengamatan perubahan pada sistem. Pada pengujian ini dilakukan dengan mengatur *set point* sistem pada 60 %, dan menunggu hingga sistem stabil. Setelah sistem stabil beri perubahan laju aliran pada pompa, dalam kasus ini kecepatan laju alir pada pompa dikurangi. Pada gambar 6, terlihat terjadi perubahan pada *manipulated variable*/bukaan *valve*. Sebelum diberi gangguan bukaan *valve* bernilai 30°, kemudian ketika diberi pengurangan pada laju alir pompa, *level* secara perlahan berkurang sebesar 5 %. Ini menyebabkan *valve* harus menutup agar *level* Kembali berada pada titik *set point*. Bukaan *valve* semakin menutup, ini berarti nilai *manipulated variable* semakin tinggi, yaitu 50°.



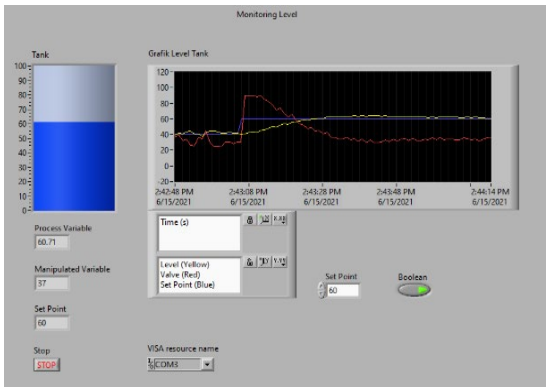
Gambar 6. Respon Pengendalian Ketika Diberi Gangguan

Untuk menguji kemampuan *controller* dalam melakukan pengendalian terhadap *level* air di dalam tangki dapat dilakukan dengan memberi perubahan pada *set point* (Muzaki et al., 2021; Prajodan et al., 2022). Pengujian dilakukan beberapa kali pengulangan.

4.1 Set point 40 % hingga 60 %

Pengujian yang pertama adalah dengan memberikan perubahan *set point* dari yang semula 40 % hingga 60 %. Ini dilakukan dalam kondisi *level* telah stabil berada disekitar *set point* semula. Kemudian dilakukan pengamatan terhadap perubahan *level* yang terjadi, dilihat pada gambar 7.

Hasil pengamatan menunjukan perubahan *level* dalam kurun waktu 60 detik. *Level* stabil pada *set point* 60 % dengan *overshoot* maksimal sebesar 5,63 %. Dapat dilihat pada gambar 7, saat ini *overshoot* adalah sebesar 0,71 % sehingga proses dapat dikatakan telah stabil. Respon waktu perubahan *level* pada pengujian pertama ini dapat dilihat pada tabel 4.



Gambar 7. Respon Pengujian Pertama

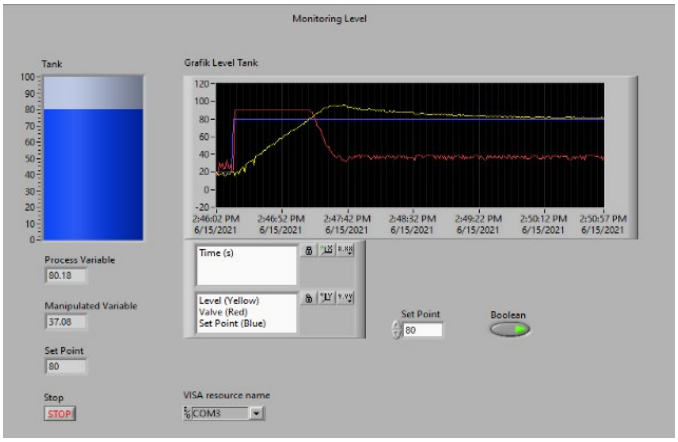
 ISSN: 2579-566X (Online)
ISSN: 2477-5274 (Print)

Tabel 4. Respon Waktu Pengujian Pertama

Respon Waktu	
Time Constant	10 detik
Delay Time	1 detik
Rise Time	22 detik
Peak Time	36 detik
Settling Time	56 detik
Overshoot	5,63 %

4.2 Set point 20 % hingga 80 %

Pengujian yang kedua adalah dengan memberikan perubahan *set point* dari yang semula 20 % hingga 80 %. Ini dilakukan dalam kondisi *level* telah stabil berada disekitar *set point* semula. Kemudian dilakukan pengamatan terhadap perubahan *level* yang terjadi, dilihat pada gambar 8.



Gambar 8. Respon Pengujian Kedua

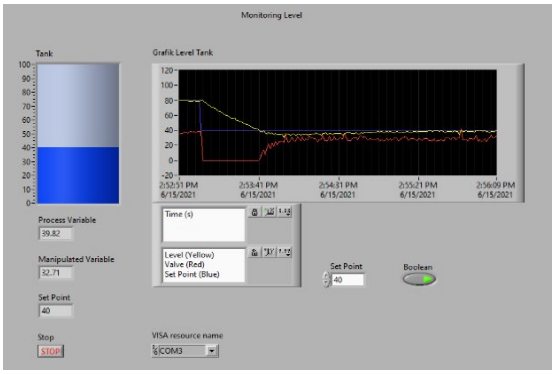
Tabel 5. Respon Waktu Pengujian Kedua

Respon Waktu	
Time Constant	29 detik
Delay Time	1 detik
Rise Time	53 detik
Peak Time	79 detik
Settling Time	263 detik
Overshoot	10,51 %

Hasil pengamatan menunjukan perubahan *level* dalam kurun waktu 270 detik. *Level* stabil pada *set point* 80 % dengan *overshoot* maksimal sebesar 10,51 %. Dapat dilihat pada gambar 8, saat ini *overshoot* adalah sebesar 0,18 % sehingga proses dapat dikatakan telah stabil. Respon waktu perubahan *level* pada pengujian kedua ini dapat dilihat pada tabel 5.

4.3 Set point 80 % hingga 40 %

Pengujian yang ketiga adalah dengan memberikan perubahan *set point* dari yang semula 80 % hingga 40 %. Ini dilakukan dalam kondisi *level* telah stabil berada disekitar *set point* semula. Kemudian dilakukan pengamatan terhadap perubahan *level* yang terjadi, dilihat pada gambar 9.



Gambar 9. Respon Pengujian Ketiga

Tabel 6. Respon Waktu Pengujian Ketiga

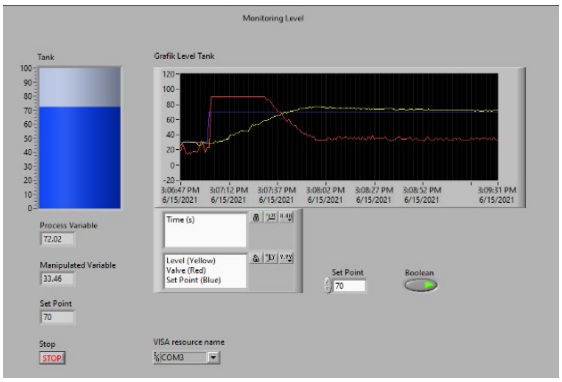
Respon Waktu	
Time Constant	24 detik
Delay Time	1 detik
Rise Time	36 detik
Peak Time	41 detik
Settling Time	136 detik
Overshoot	7,21 %

Hasil pengamatan menunjukan perubahan *level* dalam kurun waktu 140 detik. *Level* stabil pada *set point* 40 % dengan *overshoot* maksimal sebesar 7,21 %. Dapat dilihat pada gambar 9, saat ini *overshoot* adalah sebesar 0,18 % sehingga proses dapat dikatakan telah stabil. Respon waktu perubahan *level* pada pengujian ketiga ini dapat dilihat pada tabel 6.

4.4 Set point 30 % hingga 70 %

Pengujian yang ketiga adalah dengan memberikan perubahan *set point* dari yang semula 30 % hingga 70 %. Ini dilakukan dalam kondisi *level* telah stabil berada disekitar *set point* semula. Kemudian dilakukan pengamatan terhadap perubahan *level* yang terjadi, dilihat pada gambar 10.

Hasil pengamatan menunjukan perubahan *level* dalam kurun waktu 110 detik. *Level* stabil pada *set point* 70 % dengan *overshoot* maksimal sebesar 10,03 %. Dapat dilihat pada gambar 10, saat ini *overshoot* adalah sebesar 0,02 % sehingga proses dapat dikatakan telah stabil. Respon waktu perubahan *level* pada pengujian keempat ini dapat dilihat pada tabel 7.



Gambar 10. Respon Pengujian Keempat

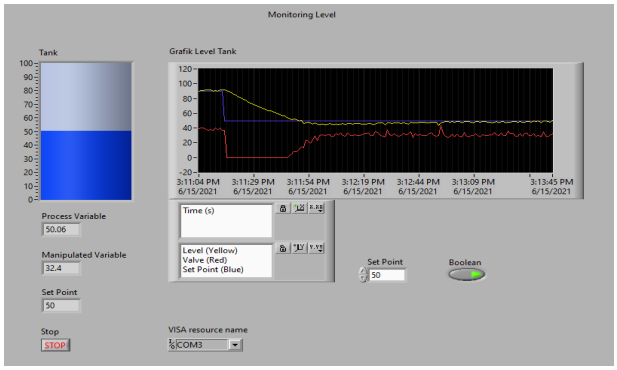
ISSN: 2579-566X (Online)
ISSN: 2477-5274 (Print)

Tabel 7. Respon Waktu Pengujian Keempat

Respon Waktu	
Time Constant	24 detik
Delay Time	1 detik
Rise Time	47 detik
Peak Time	55 detik
Settling Time	107 detik
Overshoot	10,03 %

4.5 Set point 90 % hingga 50 %

Pengujian yang ketiga adalah dengan memberikan perubahan *set point* dari yang semula 90 % hingga 50 %. Ini dilakukan dalam kondisi *level* telah stabil berada disekitar *set point* semula dilihat pada gambar 11. Hasil pengamatan menunjukan perubahan level dalam kurun waktu 100 detik. Level stabil pada set point 50 % dengan *overshoot* maksimal sebesar 6,28 %. Dapat dilihat pada gambar 11, saat ini *overshoot* adalah sebesar 0,06 % sehingga proses dapat dikatakan telah stabil. Respon waktu perubahan *level* pada pengujian kelima ini dapat dilihat pada tabel 8.



Gambar 11. Respon Pengujian Kelima

Tabel 8. Respon Waktu Pengujian Kelima

Respon Waktu	
Time Constant	20 detik
Delay Time	1 detik
Rise Time	33 detik
Peak Time	66 detik
Settling Time	93 detik
Overshoot	6,28 %

Dari berbagai percobaan dengan berbagai set point dengan mode kontroller yang paling tepat Kp = 2,96, Ki = 0,44, Kd = 0. maka berikut ini perbandingan hasil secara keseluruhan pada tabel 9.

Tabel 9. Respon Waktu Seluruh Pengujian

Set Point	*40 - 60 %*	*20 - 80 %*	*80 - 40 %*	*30 – 70 %*	*90-50 %*
Time Constant	10 detik	29 detik	24 detik	24 detik	20 detik
Delay Time	1 detik	1 detik	1 detik	1 detik	1 detik
Rise Time	22 detik	53 detik	36 detik	47 detik	33 detik
Peak Time	36 detik	79 detik	41 detik	55 detik	66 detik
Settling Time	56 detik	263 detik	136 detik	107 detik	93 detik
Overshoot	5,63 %	10,51 %	7,21 %	10,03 %	6,28 %

Secara keseluruhan dari percobaan set point diperoleh hasil perbandingan antara pengendalian secara aktual dengan simulasi melalui *MatLab* menunjukan bahwa rata-rata respon waktu aktual lebih besar. Namun *overshoot* pada simulasi menunjukan nilai yang lebih besar. Dapat dilihat pada tabel 10.

Tabel 10. Rata-rata Respon Waktu

Respon Waktu	*Aktual*	*Simulasi*
Time Constant	21,4 detik	11,08 detik
Delay Time	1 detik	0 detik
Rise Time	38,2 detik	17,3 detik
Peak Time	55,4 detik	32,14 detik
Settling Time	131 detik	101,34 detik
Overshoot	7,93 %	13,69 %

5. Kesimpulan

Hardware prototipe pengendalian dan *monitoring level* pada tangki berbasis PID *Ziegler-Nichols* dapat diaplikasikan dengan baik. Dengan memanfaatkan prototipe tangki dan pipa jaringan, *motor servo* sebagai *final control element*, *Arduino UNO* sebagai *controller*, *ultrasonic distance sensor* sebagai *sensor level*, dan *LabView* sebagai *monitor* pembacaan *level*, prototipe telah mampu merespon ketika terjadi gangguan. Hasil kalibrasi menunjukan rata-rata *error* pembacaan oleh *ultrasonic distance sensor* adalah 0,34 cm. Pengujian *motor servo* putaran pada titik 90° adalah *full close*. Hasil perhitungan parameter PID berdasarkan metode *Ziegler-Nichols* akan digunakan hasil *tuning* berdasarkan *bump test* pertama, yaitu: Kp = 2,96, Ki = 0,44, dan Kd = 0. Pemilihan ini didasarkan pada hasil simulasi yang menunjukan bahwa nilai *overshoot* dan *settling time* parameter ini adalah yang paling kecil, yaitu 35 % dan 69 detik. Hasil percobaan mendapatkan rata-rata respon waktu perubahan *level*, yaitu: *time constant* = 21,4 detik, *delay time* = 1 detik, *rise time* = 38,4 detik, *peak time* = 55,4 detik, *settling time* = 131 detik, dan *overshoot* = 7,93 %. Sedangkan rata-rata respon waktu pada simulasi *MatLab* menunjukan hasil yang lebih cepat, namun memiliki *overshoot* yang lebih besar.

Referensi

Ali, M., 2004. Pembelajaran Perancangan Sistem Kontrol PID Dengan Software Matlab. Jurnal Edukasi Elektro 1, 2.

Asraf, H.M., Nur Dalila, K.A., Muhammad Hakim, A.W., Muhammad Faizzuan Hon, R.H., 2017. Development of experimental simulator via arduino-based PID temperature control system using LabVIEW. Journal of Telecommunication, Electronic and Computer Engineering 9, 53–57.

Åström, Johan, K., Hägglund, T., 2004. Revisiting the Ziegler–Nichols step response method for PID control. Journal of process control 14, 635–650. https://doi.org/10.1016/j.jprocont.2004.01.002

Dewi, A.K., Sahaya, A.A.B.A., Sugiman, W., 2020. Level and Temperature Monitoring System in Blending Process Using Zigbee Wireless Sensor Network 436, 372–375. https://doi.org/10.2991/assehr.k.200529.077

Elya, M.N., Azrena, A.B., Yasmin, N.S., Suhaimi, N.S., 2022. A Simulation Study of Wave Disturbance Model for Underwater Vehicle. 2022 IEEE 9th International Conference on Underwater System Technology: Theory and Applications, USYS 2022. https://doi.org/10.1109/USYS56283.2022.10072423

Fakhrunnia, B.R.R., Fauziyah, M., Dewatama, D., 2020. Kontrol Suhu Menggunakan Metode PID untuk Proses Pemasakan Nira Pada Alat Pembuat Gula Merah Tebu. Jurnal Elektronika dan Otomasi Industri 3, 27. https://doi.org/10.33795/elkolind.v3i2.76

Fakhruzzaini, M., Aprilianto, H., 2017. Sistem Otomatisasi Pengontrolan Volume Dan PH Air Pada Hidroponik. Jutisi : Jurnal Ilmiah Teknik Informatika dan Sistem Informasi 6, 1335–1344.

Gozali, R.B.M., 2015. Desain Kontrol Pid Dengan Metoda Tuning Direct Synthesis Untuk Pengaturan Kecepatan Motor Dc. Teknoin 10, 283–293.

Mahardhika, W.P., Soeprijanto, A., Syaiin, M., Wibowo, S., Kurniawan, R., Herijono, B., Adhitya, R.Y., Zuliari, E.A., Setiawan, D.K., Rinanto, N., Kaloko, B.S., 2017. Design of deaerator storage tank level control system at industrial steam power plant with comparison of Neural Network (NN) and Extreme Learning Machine (ELM) method. 2017 International Symposium on Electronics and Smart Devices, ISESD 2017 2018-Janua, 40–45. https://doi.org/10.1109/ISESD.2017.8253302

Megido, A., Ariyanto, E., 2016. Sistem Kontrol Suhu Air Menggunakan Pengendali Pid. Dan Volume Air Pada Tangki Pemanas Air Berbasis Arduino Uno. Gema Teknologi 18, 21. https://doi.org/10.14710/gt.v18i4.21912

Minariyanto, A., Mardiono, M., Lestari, S.W., 2020. Perancangan Prototype Sistem Pengendali Otomatis Pada Greenhouse Untuk Tanaman Cabai Berbasis Arduino Dan Internet Of Things (IoT). Jurnal Teknologi 7, 121–135. https://doi.org/10.31479/jtek.v7i2.50

Muzaki, A.K., Wardhana, A.S., Dewi, A.K., 2021. Pengendalian dan Monitoring Level Air pada Tangki Pemisah Dua Fasa dengan Metode Internal Model Control, in: In Prosiding Seminar Nasional Teknologi Energi Dan Mineral. pp. 1078–1087.

Ogata, K., 2010. Modern control engineering, Lima. ed. Prentice Hall, New Jersey USA.

Pamuji, Agung, F., Danier, D., Sudarmanta, B., Guntur, H.L., Praskosa, P.R., Waskito, I.S., 2021. Comparison of BLDC Motor Controller Design for Electric Vehicles Using Fuzzy Logic Controller and Artificial Neural Network. Przegląd Elektrotechniczny 1, 3–11. https://doi.org/10.15199/48.2021.06.01

Pangya, T., Thedsakhulwong, A., Ekwongsa, C., 2018. Development of the PID controller and real-time monitoring system for a low-temperature furnace. Journal of Physics: Conference Series 1144. https://doi.org/10.1088/1742-6596/1144/1/012160

Prajodan, R.S.L., Wardhana, A.S., Triyanto, R.H., 2022. Analisis Kestabilan Sistem Pengendalian Level pada Steam Drum Boiler 1 dengan Metode Nyquist Stability, in: Prosiding Seminar Nasional Teknologi Energi Dan Mineral. pp. 1228–1239.

Ravy, J.U., Septiani, N.A., Dewi, A.K., Wardhana, A.S., 2021. Evaluasi Kinerja Controller Design PI Sistem Pengendalian Level Pada Centrifugal Preparation Tank, in: Prosiding Seminar Nasional Teknologi Energi Dan Mineral. pp. 1–11.

Rohiem, N.H., Putra, N.P.U., 2021. Sistem Monitoring Kecepatan Motor dan Tekanan pada Saluran Air Berbasis Internet of Things (IoT). INTEGER: Journal of Information Technology 6, 74–80. https://doi.org/10.31284/j.integer.2021.v6i1.1835

Saputra, D.N., Evelina, Sari, D.P., 2022. Analisa Sensor Infrared pada Alat Sortir Otomatis Berdasarkan Tinggi dengan Sistem Kendali Software HMI Haiwell Scada Berbasis PLC Outseal. TEKNIKA 16, 31–35.

Sujono, S., Dewi, A.K., Soegiarto, T.S., 2020. Evaluating and optimizing performance of shell and tube heat exchanger using excel-solver. IOP Conference Series: Materials Science and Engineering 830. https://doi.org/10.1088/1757-899X/830/4/042029

Wardhana, A.S., Azizah, H.A., Hamdani, C.N., 2021. Pengujian Sistem Pengendalian Temperatur pada Prototipe Heat Exchanger Berbasis PID. Jurnal Otomasi Kontrol dan Instrumentasi 13, 81–91. https://doi.org/10.5614/joki.2021.13.2.3

Wenzhe, L., Hrnjak, P., 2018. Experimentally Validated Model of Heat Pump Water Heater With a Water Tank in Heating-Up Transients. International Journal of Refrigeration 88, 420–431.

Wirasanto, I.F., Widodo, W., 2021. Rancang Bangun Monitoring Dan Penetralisir Kadar Asap Didalam Ruangan Menggunakan Sensor MQ-2 Berbasis Intenet Of Things (IoT). INTEGER: Journal of Information Technology 6, 42–47. https://doi.org/10.31284/j.integer.2021.v6i1.1435

Yasmin, A., Wardhana, A.S., 2022. Perbandingan Tuning Menggunakan Metode Ziegler Nichols Dan Chien Regulator 2 Pada Production Separator. Prosiding Seminar Nasional Teknologi Energi dan Mineral 2, 1156–1166. https://doi.org/10.53026/sntem.v2i1.964

Zainul, R., Rahmad, E.U., Aledresi, K.A.M.S., Nursiwan, W.A., Guspatni, Mukdasai, S., Mandeli, R.S., 2022. Electrolyte Optimization on Dry Cell Generator Electrolysis System for Producing Hydrogen Gas Using Rsm Method (Response Surface Method). Rasayan Journal of Chemistry 15, 116–123. https://doi.org/10.31788/rjc.2022.1516632