

Rancang Bangun *PID Controller* Pada Level Simulator Berbasis ESP32

Emelia Agnes Wulandari Samosir¹, Roni Heru Triyanto^{2,*}

^{1,2}Program Studi Teknik Instrumentasi Kilang, Politeknik Energi dan Mineral Akamigas, Cepu Jawa Tengah, Indonesia 55312

*Corresponding Author: roniherutriyanto@gmail.com (Roni Heru Triyanto)

Info Artikel

Histori Artikel:

Diterima berkas: 5 Mei 2024

Direvisi: 20 Mei 2024

Disetujui terbit 1 Juni 2024

Keyword:

PID Controller
Tuning Controller
Level Simulator
ESP32

ABSTRAK

Air merupakan salah satu sumber daya alam yang sangat bermanfaat bagi kita makhluk hidup. Untuk menjaga air yang digunakan tidak berlebihan atau berkekurangan maka kita perlu mengendalikan ketinggian dari air tersebut dalam sebuah bak. Pengendalian ini disebut dengan pengendalian level, pengendalian level ini merupakan salah satu pengendalian yang sering dilakukan dalam dunia industry.. Pengendalian dilakukan agar mencapai nilai set point yang diinginkan. Dalam pengendalian level ini digunakan mikrokontroler ESP32 dan juga teknologi IoT sebagai blynk sebagai interfacenya. Pengendalian level ini menggunakan metode PID (proportional, integral, derivative) dan menggunakan metode tuning Ziegler Nichols sebagai uji kestabilan proses tersebut. Setelah dilakukan bump test maka didapatkan L 3s. dan T 146s. Untuk menentukan nilai PID maka dilakukan metode tuning menggunakan Ziegler Nichols dan didapatkan lah hasil Kp 58.4, Ki 9.7 dan Kd 39. Hasil implementasi menunjukkan bahwa simulasi sebanyak 3 kali perubahan set point dari 20 cm ke 25 cm, dari 25 cm ke 30 cm, dan dari 30 cm ke 35 cm menunjukkan bahwa respon mampu stabil walaupun tidak stabil secara sempurna.

ABSTRACT

Water is a natural resource that is very beneficial for us living creatures. To ensure that the water used is not excessive or insufficient, we need to control the height of the water in a tank. This control is called level control, this level control is one of the controls that is often carried out in the industrial world. Control is carried out to achieve the desired set point value. In controlling this level, the ESP32 microcontroller and IoT technology as blynk are used as the interface. This level control uses the PID (proportional, integral, derivative) method and uses the Ziegler - Nichols tuning method as a stability test for the process. After carrying out the bump test, it was found that L3s, and T 146s. To determine the PID value, the tuning method was carried out using Ziegler Nichols and the results were Kp58.4, Ki9.7 and Kd39. The implementation results showed that the simulation of changing the set point from 20 cm to 25 cm, from 25 cm to 30 cm, and from 30 cm to 35 cm showed that the response can be stable even though it is not perfectly stable.

This is an open access article under the CC BY-SA license.



1. PENDAHULUAN

Di era globalisasi, bisnis sangat penting, terutama di Indonesia. Banyak bisnis berkembang dengan pesat. Industri membutuhkan sistem kontrol yang baik untuk mendukung operasi industri dan meningkatkan efisiensi proses produksi. Air merupakan kebutuhan yang sangat penting baik dalam industri maupun rumah tangga. Namun, ketinggian air melebihi kapasitas industri, yang berarti air terbuang sia-sia. Ini disebabkan oleh fakta bahwa pengontrolan level industri masih dilakukan secara manual oleh manusia. Pengontrolan level manual memiliki banyak kekurangan, salah satunya adalah kemungkinan kesalahan manusia dalam menentukan

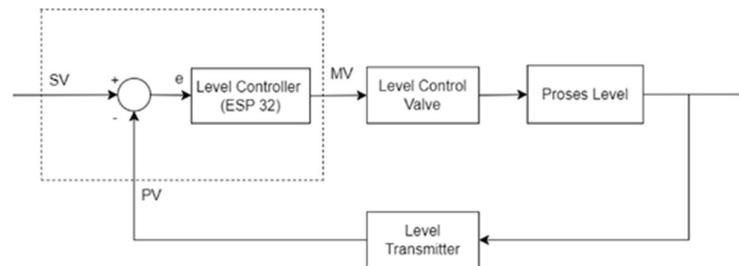
ketinggian level. Walaupun air ialah renewable energy, cadangan clean water juga dapat berkurang seiring bertambahnya waktu dan jumlah penduduk.

Sistem pengendalian berbasis PID digunakan untuk mengendalikan level industri. Mode kontrol proporsional dan derivatif digunakan sebab setiap mode punya keunggulan untuk membuat proses menjadi stabil. Kontroler proporsional membantu mempercepat menyentuh waktu naik, kontroler integral membantu mengurangi nilai kesalahan, dan kontroler derivatif membantu meredam dan mengurangi overshoot. Demi memperoleh hasil terbaik, gabungkan semua aksi mode kontrol PID. Kontrol PID sangat bermanfaat untuk memperoleh keluaran yang diinginkan karena mampu membuat sistem yang sensitif terhadap gangguan lainnya. Dengan sistem PID, kita dapat mengatur respons yang kita ingin gunakan dalam sistem yang akan dibuat. Nilai parameter K_p , K_i , dan K_d yang tepat menentukan metode PID bekerja dengan baik.

ESP32 adalah sebuah mikrokontroler yang berfungsi untuk memudahkan manusia dalam melakukan berbagai hal. ESP32 akan menerima input dari sensor yang kemudian diolah dan menghasilkan output yang sesuai dengan yang telah diperintahkan. Digunakan ESP 32 untuk mengontrol PID dan memantau proses pengisian air. Dengan bantuan aplikasi blynk, pengendalian ketinggian air dapat dilakukan dari jarak jauh melalui smartphone. Ini membuatnya lebih mudah dan praktis.

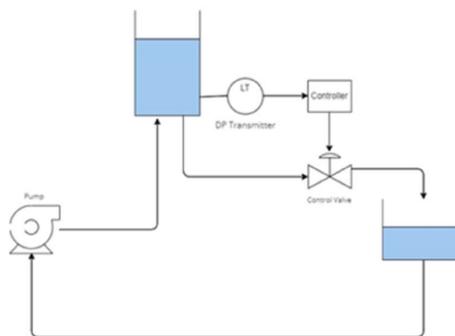
2. MATERIAL DAN METODA

Dalam penelitian ini, proses pengendalian level dikendalikan oleh PID Controller yang dirancang menggunakan mikrokontroler ESP32. Jenis kontrol PID ini merupakan gabungan dari tiga pengendalian yaitu Proportional, Integral, dan Derivative. Feed yang berada di industri dapat dikontrol secara automatic maupun manual melalui aplikasi blynk ataupun langsung di field. User dapat memilih antara dua mode yaitu manual ataupun automatic. Apabila memilih mode manual maka user dapat langsung mengatur MV, dan apabila mode automatic maka program akan langsung berjalan dengan melibatkan perhitungan PID. Baik dari mode manual ataupun automatic outputnya sama yaitu menuju control valve yang akan mengatur bukan dari control valve itu sendiri.



Gambar 1. Block Diagram System

Alur perancangan software disajikan dalam bentuk flowchart. Software yang digunakan pada perancangan ini yaitu Arduino IDE sebagai programming sistem pengendalian level. Untuk menjalankan alat dimulai dengan mendesain flowchart sistem dari program yang akan dibuat. Berikut adalah flowchart sistem pada gambar 1.



Gambar 2. Flowchart Cara Kerja Sistem

Dalam merancang suatu prototype tentunya diperlukan adanya diagram alir proses yang menyatakan hubungan antar komponen dan juga aliran proses yang jelas. Untuk itu penulis telah membuat PFD tentang rancang bangun PID controller pada level simulator berbasis ESP 32 seperti terlihat di gambar 3.2 diatas ini.

2.1. METODA

2.1.1 Sistem Kendali

Sistem pengendali adalah suatu sistem yang menghasilkan nilai tertentu sebagai keluarannya melalui pengendalian ataupun perubahan ketentuan dari masukan sistem. Sistem ini berfungsi untuk mengendalikan, memerintah, dan mengatur keadaan dari suatu sistem untuk menghasilkan keluaran yang diinginkan[1]. Hal ini dilakukan agar karakteristik dinamis sistem yang dihasilkan dari desain memenuhi spesifikasi kinerja tertentu. Hal pertama yang harus dilakukan dengan sebuah sistem kendali adalah. menentukan strukturnya dengan benar. Perencanaan ini biasanya dilakukan agar memenuhi terhadap spesifikasi berikut yaitu ketelitian, kecepatan, settling time paling lama, dan stabilitas.

Spesifikasi kontrol harus digunakan sebagai dasar untuk setiap perancangan pengendali atau spesifikasi pengendali. Spesifikasi kinerja untuk respons transient digunakan sebagai referensi untuk memilih controller yang baik ketika mengatur valve keluaran tank untuk mengetahui level air. Dalam praktik, respons transien sistem kontrol selalu menunjukkan osilasi teredam sebelum mencapai keadaan stabil.

2.1.2 PID Controller

Pengendali PID sangat banyak penggunaannya dalam dunia industri karena memiliki nilai presisi yang cukup tinggi, mudah diaplikasikan dan handal. Pengendalian dilakukan dengan mengurangi sinyal kesalahan pada saat sistem bekerja, serta mampu memberikan keluaran sinyal yang memiliki respon yang cepat, error kecil, dan meminimalisir terjadinya overshoot. Semakin kecil nilai error yang terjadi maka semakin lancar unjuk kerja sistem kendali yang digunakan[2]. Nilai variable proses diterima oleh kontroler, yang kemudian membandingkannya dengan set point untuk mengetahui apakah proses telah mencapai atau mendekati nilai yang diinginkan. Selanjutnya, algoritma yang disebut sebagai mode kontrol digunakan untuk menghitung perbedaan nilai ini. Nilai yang dihasilkan oleh algoritma ini dikirim ke elemen kontrol terakhir. Gain yang diterima oleh kontroler dipengaruhi oleh mode kontrolnya.

Jika sinyal hasil pengukuran yang dimasukkan ke kontroler naik (naik) dan outputnya juga naik (naik).. maka kontroler tersebut memiliki aksi langsung. Jika sinyal hasil pengukuran turun (decrease) tetapi output kontroler juga naik (naik), maka kontroler tersebut memiliki aksi reverse. Istilah "mode kontrol" atau "algoritma kontrol" digunakan dalam sistem kontrol. Menurut [3], ada enam (enam) kategori: proportional (P), integral (I), proportional integral (PI), proportional derivative (PD), dan proportional integral derivative (PID).

2.1.3 Metode Tuning

Metode tuning adalah proses mengatur parameter kontroler untuk mendapatkan nilai respon kontrol yang optimal. Dalam beberapa penelitian, metode tuning PID digunakan untuk mengatur parameter kontroler PID seperti koefisien proporsional (K_p), integral (K_i), dan derivat (K_d)[4]. Metode tuning yang umum digunakan adalah metode Ziegler-Nichols dan metode Cohen-Coon. Metode Ziegler-Nichols melibatkan penggunaan gain margin dan phase margin untuk menentukan robustness suatu sistem, sedangkan metode Cohen-Coon menggunakan persamaan yang lebih kompleks untuk menghitung nilai parameter kontroler yang optimal [5]. Penentuan parameter kontroler PID untuk membuat sistem memenuhi syarat yang diinginkan. dikenal sebagai pengaturan kontroler. Tuning PID dilakukan secara manual dengan menggunakan prinsip Kerja loop tertutup. Ini dilakukan untuk memaksimalkan fungsi pengontrol pada parameter PID dan penalaan sistem

Ada beberapa cara berbeda untuk melakukan penyetelan pengontrol PID secara manual, berikut merupakan step to step cara menggunakan tuning manual

1. Atur system dalam keadaan manual kontrol.
2. Lalu setting sistem dengan K_p , K_i , dan K_d pada 0
3. Pada Langkah ketiga, kita mengatur K_p yaitu dengan cara meningkatkan K_p hingga error pada kondisi tunak sangat rendah
4. Setelah merendahkan nilai K_p , maka Langkah selanjutnya ialah mengatur nilai K_i dengan cara meningkatkan hingga error kondisi tunak tersebut hilang seluruhnya
5. Langkah terakhir ialah kita mengatur nilai K_d dengan cara meningkatkannya sampai osilasi dihilangkan. 10
6. Atur pengambilan sampel dan juga sampel yang diinginkan.
7. Mulai perekaman data untuk mendapatkan data tersebut
8. Amati perubahan data tersebut.

9. Setelah mendapatkan data, kemudian masukkan dapat tersebut ke Microsoft excel untuk dilakukan Analisa.

Menemukan nilai K_p yang baik untuk memulai bisa sangat membantu. Untuk melakukan ini, output maksimum dari sistem yang dibuat harus diketahui dan juga satuan yang digunakan dalam mengukur sistem ini.

2.2 MATERIAL

2.2.1 DP Transmitter

Differential Pressure pada dasarnya adalah salah satu metode pengukuran tekanan yang tidak mengacu pada referensi tekanan khusus[6]. Prinsip "hidrostatis head" mengatakan bahwa pada setiap titik di dalam fluida yang diam (statik), gaya yang bekerja padanya adalah sama untuk semua arah dan tidak tergantung pada volume fluida, bentuk ruang, atau lokasi fluida, tetapi hanya pada tinggi kolom fluida di atas titik tersebut. Ini adalah dasar untuk pengukuran level jenis tekanan diferensial (DP). Oleh karena itu, kepala hidrostatik sering digunakan dalam satuan tekanan.

Beberapa tahun sebelumnya, penggunaan metode perbedaan tekanan atau tekanan hidrostatik untuk mengukur level telah mengalami kemajuan besar. Peralatan D/P ini dapat mengukur level dalam rentang yang luas pada layanan yang bersih, korosif, slurry, dan tinggi viscous. Jika peralatan tersebut tersedia dalam rentang yang diperlukan untuk level yang dimaksud, hampir semua jenis peralatan D/P dapat digunakan untuk mengukur level tersebut. Range pengukuran D/P umumnya berkisar antara (10 hingga 150) inci suhu udara.

2.2.2 ESP 32

Mikrokontroler ESP32 merupakan mikrokontroler SoC (System on Chip) terpadu dengan dilengkapi WiFi 802.11 b/g/n, Bluetooth versi 4.2, dan berbagai peripheral. ESP32 adalah chip yang cukup lengkap, terdapat prosesor, penyimpanan dan akses pada GPIO (General Purpose Input Output). ESP32 bisa digunakan untuk rangkaian pengganti pada Arduino, ESP32 memiliki kemampuan untuk mendukung terkoneksi ke WI-FI secara langsung [7]. Karena dilengkapi dengan fitur WiFi dan Bluetooth 4.2 pada boardnya, ESP 32 memiliki keunggulan dibandingkan dengan Arduino.

2.2.3 Internet Of Things

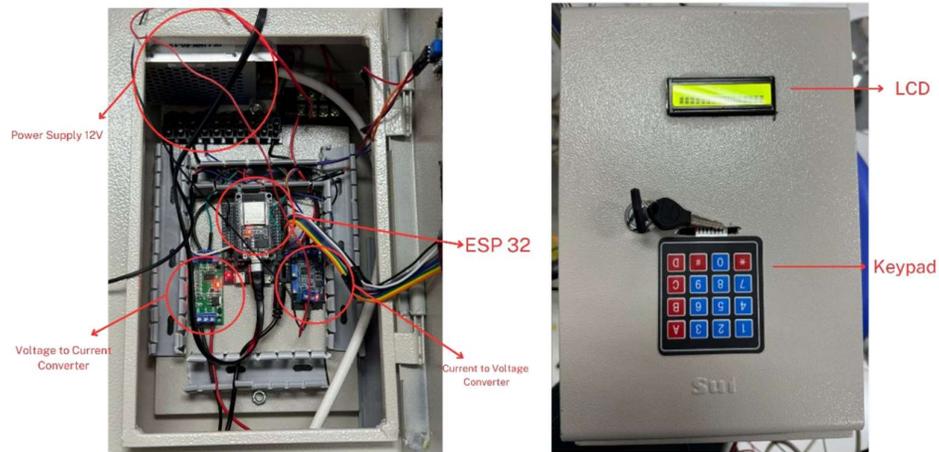
Internet of things atau bisa disebut juga dengan IoT adalah sebuah teknologi canggih yang memiliki konsep yang bertujuan untuk memperluas dan memperkembang manfaat dari konektivitas internet yang tersambung terus menerus menghubungkan benda-benda di sekitar agar aktivitas sehari-hari menjadi lebih mudah dan efisien yang sangat membantu segala pekerjaan manusia. Pentingnya internet of things dapat dilihat dengan semakin banyaknya diterapkan dalam berbagai kehidupan saat ini. Menurut metode identifikasi RFID (Radio Frequency Identification), istilah IoT tergolong dalam metode komunikasi, meskipun IoT juga dapat mencakup teknologi sensor lainnya, teknologi nirkabel atau kode QR (Quick Response) [8].

Untuk melakukan fungsi Internet of Things, internet menghubungkan dua interaksi mesin, sedangkan pengguna hanya bertindak sebagai pengatur dan pengawas alat secara langsung. Internet of thing (IoT) merupakan suatu konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus[9]. Menurut [10], konsep Internet of Things, juga dikenal sebagai "IoT", bertujuan untuk meningkatkan manfaat dari konektivitas internet yang tersambung secara terus-menerus. yang memungkinkan kita untuk menghubungkan sensor jaringan dan aktuator ke mesin, peralatan, dan benda fisik lainnya untuk memperoleh data dan mengelola kinerjanya sendiri, yang memungkinkan mesin untuk berkolaborasi dan bahkan bertindak berdasarkan apa yang mereka ketahui. Blynk adalah aplikasi berbasis Android dan iOS yang berfungsi sebagai modul kontrol berbasis internet untuk berbagai perangkat, termasuk tetapi tidak terbatas pada ESP32, Arduino, dan Raspberry Pi..

3. HASIL DAN PEMBAHASAN

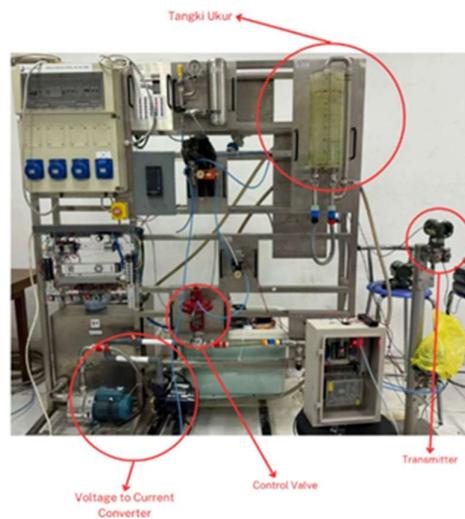
3.1 Implementasi Sistem

Tujuan dari pengontrol PID pada level simulator berbasis ESP 32 ini adalah untuk mengukur ketinggian level tangki dari jarak jauh melalui IoT. Prototype yang dibuat merupakan bagian dari sistem pompa yang menggunakan ESP 32 sebagai mikrokontroler untuk mengukur ketinggian level air pada tangki, yang berasal dari bak penampungan. Perhitungan dan monitoring ketinggian level dapat dilakukan dengan menggunakan sistem telemetri berbasis IoT. Hasil implementasi dari rancang bangun PID controller pada level simulator berbasis ESP 32 ini terdiri dari level simulator, panel box, ESP32, voltage to current converter, current to voltage converter, power supply 12V & 24V, LCD 16x2, dan keypad.



Gambar 3. *Prototype Controller*

Gambar 3 diatas menunjukkan tampak depan dan juga tampak dalam dari controller Pada bagian depan dari panel box ini terdapat LCD dan juga keypad. Lalu pada bagian dalam panel box terdapat ESP32, voltage to current converter, current to voltage converter, dan juga power supply. Sedangkan pada level simulator terdiri dari beberapa bagian yaitu tangki penyimpanan atau pembuangan air, tangki perhitungan, transmitter, pompa, dan juga control valve. Gambar 4.3 dibawah ini menunjukkan bagian bagian dari level simulator itu sendiri.



Gambar 4. *Wiring Diagram*



Gambar 5. *Blynk untuk Monitoring Process*

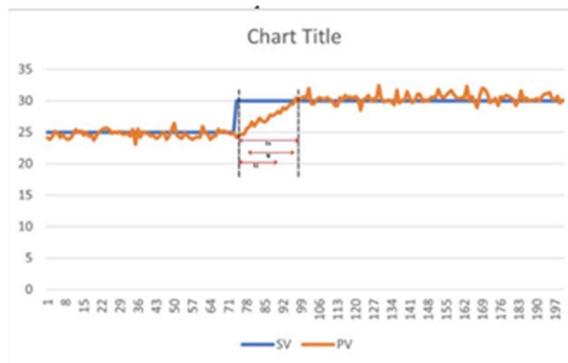
Dalam rancang bangun prototype ini, menggunakan software yaitu Arduino IDE. Software Arduino IDE digunakan untuk memprogram mikrokontroler ESP32 agar bisa bekerja sesuai dengan konsep prototype yang akan dijalankan. Disana terdapat inialisasi pin dari sensor dan beban yang akan digunakan serta pengkonversian nilai bacaan sensor menjadi nilai yang dapat dipahami. Sensor adalah digunakan untuk mengukur level. Blyn dipakai untuk user interface pada sistem pengendalian ini agar lebih mudah dipahami. Tampilannya meliputi monitoring hasil pengukuran sensor dan perubahan nilai setpoint dari masing-masing variabel proses yang diukur, seperti yang terlihat pada Gambar 5, dari sana bisa dilihat perubahan yang terjadi dalam proses secara real time.

3.2 Analisis Data

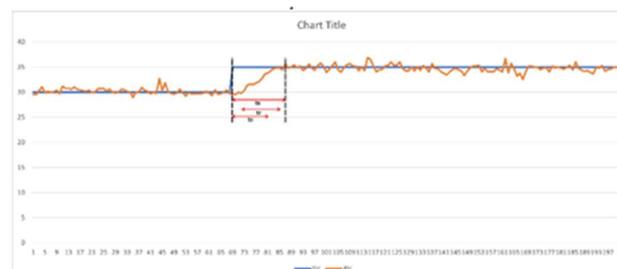
Pada tahapan ini dilakukan beberapa pengamatan dari hasil simulasi yaitu mengamati seberapa lama level akan stabil saat set point tertentu. Dilakukan tiga kali pengujian perubahan set point dari 20 cm ke 25 cm, 25 cm ke 30 cm, dan dari 30 cm ke 35 cm. Sebelum melakukan pengujian sudah dilakukan bump test untuk mendapatkan nilai K_p , K_i , dan K_d dari proses ini. Setelah dilakukan pengujian maka didapatkan hasil berupa grafik seperti gambar 6 dibawah ini.



Gambar 6. Setpoint 20 cm ke 25 cm



Gambar 7. Setpoint 25 cm ke 30 cm



Gambar 8. Setpoint 30 cm ke 35 cm

Pada analisis data maka dilakukan analisis dari hasil pengujian dengan menggunakan metode tuning PID Ziegler Nichols yang telah dilakukan, sehingga menghasilkan data sebagai berikut :

No	Spesifikasi	20 – 25 cm	25 - 30 cm	30 - 35 cm
1	Setting time	24 s	22 s	17 s
2	Rise time	20 s	20 s	14 s
3	Time constant	15 s	18 s	11 s

Pada tabel diatas dapat ditarik analisis data berdasarkan pengujian sistem menggunakan metode Ziegler Nichols yaitu dari hasil pengujian sebanyak tiga kali perubahan set point dari 20 cm ke 25 cm, dari 25 cm ke 30 cm. dan dari 30 cm ke 35 cm menggunakan metode tuning ziegler Nichols menunjukkan bahwa respon dapat stabil dan dapat menyesuaikan dengan set point yang diinginkan meskipun tidak stabil secara sempurna, namun system dapat merespon dengan baik tuning yang digunakan. Ada keterlambatan dalam komunikasi dengan blynk. Ini disebabkan oleh waktu yang terbuang untuk mengirimkan data ke blynk, efek dari koneksi hotspot, dan efek dari respon blynk. Dengan kata lain pembacaan sensor dan interface yang digunakan sangat dipengaruhi oleh jaringan internet. Sehingga berdampak pada perbedaan spesifikasi set point yang dibuat.

4. KESIMPULAN

Berdasarkan tujuan penelitian dan hasilnya. Pada level simulator berbasis ESP32 dengan kontrol PID dan menggunakan metode tuning Ziegler Nichols ini, beberapa poin penting terkait desain kontrol PID dapat disimpulkan sebagai berikut: Prototipe rancang bangun PID controller pada level simulator berbasis ESP32 telah berhasil dibuat. Jenis system pengendalian yang digunakan yakni control PID dan metode tuning yang digunakan ialah metode Ziegler - Nichols.

Berdasarkan bump test yang dilakukan untuk mendapatkan nilai PID maka didapatkan nilai L sebesar 3s dan T sebesar 146s. Lalu dengan menggunakan metode Ziegler Nichols maka didapatkan nilai $K_p = 58,4$ nilai $K_i 9,7$ dan nilai $K_d = 39$.

Berdasarkan hasil pengujian sebanyak tiga kali yaitu perubahan set point dari 20 cm ke 25 cm, dari 25 cm ke 30 cm, 30 cm ke 35 cm maka didapatkan bahwa semakin besar set point maka settling time akan semakin berkurang. Penggunaan Blynk sebagai interface berjalan sesuai dengan tujuan yaitu memonitoring proses yang ada di plant sehingga kita dapat mengetahui besaran proses seperti process variable, set value, dan manipulated variable pada proses dapat dipantau secara real time.

DAFTAR PUSTAKA

- [1] J. Rantung. "KARAKTERISTIK PENGENDALI ON-OFF UNTUK APLIKASI PADA SISTEM PENGENDALIAN TEMPERATUR," 2015.
- [2] T. Pujiati, "Penerapan Kontroler PID pada Sistem Kendali Level Cairan dengan Metode Ziegler- Nichols Berbasis Arduino," 2021.
- [3] Katsuhiko. Ogata, Modern control engineering. Prentice-Hall, 2010.
- [4] H. Wicaksono, "Analisa Performansi dan Robustness Beberapa Metode Tuning Kontroler PID pada Motor DC," 2004. [Online]. Available: <http://puslit.petra.ac.id/journals/electrical/70>
- [5] H. Supriyanto et al., "IMPLEMENTASI KONTROLER PID DENGAN METODE TUNING ZIEGLER-NICHOLS DAN COHEN-COON PADA SISTEM SCADA KENDALI LEVEL AIR." Jurnal Teknologi Terapan) |, vol. 8, no. 2, 2022.
- [6] "ELO_DIV_TEI_18130024_Muhammad Difa Syafti_2022".
- [7] A. Wagiyana, J. Teknik Elektro, P. Negeri Jakarta, J. G. Siwabessy, and K. U. Depok, "Prototipe Modul Praktik untuk Pengembangan Aplikasi Internet of Things (IoT)," Jurnal Ilmiah Setrum Article In Press, vol. 8, no. 2, pp. 238 247, 2019.
- [8] A. Selay et al., "INTERNET OF THINGS," 2022.
- [9] F. Panduardi, S. Haq, P. Studi, T. Informatika, and P. N. Banyuwangi, "WIRELESS SMART HOME SYSTEM MENGGUNAKAN RASPBERRY PI BERBASIS ANDROID," 2016.
- [10] S. Arafat, M. Kom, and Kom, "SISTEM PENGAMANAN PINTU RUMAH BERBASIS Internet Of Things (IoT) Dengan ESP8266," Oktober-Desember, 2016.