

Perancangan Sistem Kontrol Suhu Ruangan Menggunakan Metode Proportional, Integral, dan Derivatif Berbasis Arduino Uno

^{1*}Febrian M Saputra, ²Ryan Fikri, ³Winda Agustiarmi, Zulwishi

¹ Program Studi Teknologi Rekayasa Sistem Elektronika, Departemen Teknik Elektronika, Fakultas Teknik, Universitas Negeri Padang, Indonesia

*Corresponding Author e-mail: febriannsaputraa0102@gmail.com

Received: May 2026; Revised: July 2026; Published: August 2026

Abstrak

Pengendalian suhu secara otomatis diperlukan untuk menjaga kondisi termal tetap mendekati nilai setpoint tanpa pengaturan manual terus-menerus. Penelitian ini bertujuan merancang dan menguji sistem kontrol suhu berbasis Arduino Uno menggunakan metode Proportional, Integral, and Derivative (PID). Sistem menggunakan sensor DHT11 sebagai pembaca suhu, lampu 12 V sebagai aktuator pemanas, kipas DC 12 V sebagai aktuator pendingin, MOSFET IRF520 sebagai driver aktuator, dan kotak akrilik sebagai ruang pengujian. Penelitian menggunakan pendekatan Research and Development (R&D) melalui tahap perancangan perangkat keras, pemrograman, perakitan, dan pengujian sistem. Nilai setpoint ditetapkan sebesar 26 °C. Nilai parameter PID yang digunakan adalah K_p = [isi sesuai program], K_i = [isi sesuai program], dan K_d = [isi sesuai program]. Pengujian dilakukan sepuluh kali, terdiri atas lima pengujian dari suhu awal 24 °C menuju 26 °C dan lima pengujian dari suhu awal 29 °C menuju 26 °C. Pada proses pemanasan, waktu respons berada pada rentang 2 menit 35 detik hingga 4 menit 30 detik dengan rata-rata sekitar 3 menit 6 detik. Pada proses pendinginan, waktu respons berada pada rentang 3 menit 45 detik hingga 4 menit 26 detik dengan rata-rata sekitar 4 menit 10 detik. Seluruh percobaan mampu mengarahkan suhu menuju setpoint 26 °C. Keluaran aktuator berkurang ketika suhu semakin mendekati setpoint sehingga perubahan suhu menjadi lebih terkendali. Hasil tersebut menunjukkan bahwa sistem yang dikembangkan mampu menjalankan fungsi pemanasan dan pendinginan secara otomatis serta mempertahankan suhu di sekitar setpoint pada kondisi pengujian prototipe.

Kata Kunci: Kontrol Suhu Ruangan; PID; Arduino Uno; Sensor DHT11; Pemanas Dan Pendingin

How to Cite: Saputra, F. M., Fikri, R., Agustiarmi, W., & Zulwishi. (2026). Perancangan Sistem Kontrol Suhu Ruangan Menggunakan Metode Proportional, Integral, dan Derivatif Berbasis Arduino Uno. *Journal of Authentic Research*, 5(3), 7006–7017. <https://doi.org/10.36312/jar.v5i3.6861>



<https://doi.org/10.36312/jar.v5i3.6861>

Copyright© 2026, Saputra et al.

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



PENDAHULUAN

Kenyamanan termal merupakan aspek krusial dalam menciptakan lingkungan ruangan yang mendukung produktivitas dan kesehatan penghuninya. Menurut Hadi et al. (2020), pencapaian kondisi nyaman ini sangat bergantung pada pengukuran parameter lingkungan yang akurat, seperti suhu, kelembapan, dan kecepatan angin di dalam ruangan (Visioli, 2006). Ketidakstabilan suhu ruangan tidak hanya berdampak pada konsentrasi, tetapi juga dapat memengaruhi kondisi fisiologis manusia, seperti peningkatan detak jantung dan kelelahan dini akibat mekanisme termoregulasi tubuh yang bekerja ekstra. Persepsi kenyamanan setiap individu juga bersifat subjektif sehingga dibutuhkan sistem yang responsif terhadap perubahan lingkungan (Hadi et al., 2020). Namun, pada kenyataannya, banyak

perangkat pendingin udara seperti kipas angin yang masih beroperasi secara manual dengan kecepatan putar yang statis tanpa mempertimbangkan kondisi lingkungan sekitarnya. Hal ini menyebabkan ketidakseimbangan energi dan ketidaknyamanan bagi pengguna karena harus melakukan penyesuaian kecepatan secara berulang saat terjadi fluktuasi suhu (Franklin et al., 2019).

Secara konvensional, pengaturan suhu ruangan sering dilakukan secara manual, misalnya dengan menyalakan atau mematikan pendingin/pemanas berdasarkan perkiraan atau pengamatan langsung oleh manusia (D-Robotics, 2010). Cara ini kurang efisien karena rentan terhadap kesalahan, respons yang lambat, serta tidak dapat bekerja secara terus-menerus tanpa pengawasan (Baskoro et al., 2024). Pengembangan teknologi mikrokontroler membuka peluang untuk merancang sistem kendali otomatis yang lebih presisi, salah satunya menggunakan metode kendali Proportional Integral Derivative (PID), yang mampu menghasilkan respons sistem lebih cepat mencapai setpoint, meminimalkan error tunak, dan mengurangi overshoot dibandingkan metode kontrol on/off sederhana (Megido & Ariyanto, 2016; Ogata, 2010).

Metode tuning PID yang umum digunakan antara lain Ziegler-Nichols, Cohen-Coon, dan trial and error (Ogata, 2010; Nise, 2020). Pada penelitian ini, metode trial and error dipilih karena sistem masih berupa prototipe, jumlah data terbatas, dan penyesuaian langsung diperlukan untuk melihat respons aktuator pemanas dan pendingin secara empiris (Åström & Murray, 2021). Pengaturan suhu secara manual juga memiliki keterbatasan, di antaranya respons yang lambat, tingkat akurasi yang rendah, serta ketergantungan terhadap pengawasan manusia secara terus-menerus (Åström & Hägglund, 2006). Seiring dengan berkembangnya teknologi mikrokontroler, berbagai sistem kendali otomatis mulai dikembangkan untuk mengatasi permasalahan tersebut. Salah satu metode pengendalian yang banyak diterapkan dalam sistem kontrol adalah metode Proportional Integral Derivative (PID) (Banzi & Shiloh, 2022). Metode PID bekerja dengan menghitung selisih (error) antara nilai suhu aktual dan suhu yang diinginkan (setpoint), kemudian menghasilkan sinyal kendali yang mampu mempercepat pencapaian setpoint, memperkecil steady-state error, serta mengurangi overshoot dibandingkan metode pengendalian on/off konvensional (Megido & Ariyanto, 2016).

Dalam penelitian ini, sistem pengendalian suhu dirancang menggunakan mikrokontroler Arduino dengan memanfaatkan sensor DHT11 sebagai sensor untuk mengukur suhu dan kelembapan ruangan. Meskipun DHT11 memiliki tingkat akurasi yang lebih rendah dibandingkan beberapa sensor lainnya, sensor ini memiliki keunggulan biaya yang ekonomis, kemudahan integrasi dengan mikrokontroler, serta kemampuan pengukuran yang memadai untuk aplikasi pemantauan dan pengendalian suhu pada skala laboratorium maupun media uji sederhana. Data suhu yang diperoleh dari sensor DHT11 digunakan sebagai masukan bagi algoritma PID untuk menentukan aksi pengendalian terhadap aktuator sistem (Ibrahim, 2019).

Dalam sistem pengendalian suhu, sensor merupakan komponen utama yang berfungsi mendeteksi kondisi lingkungan secara real-time. Penelitian ini menggunakan sensor DHT11 sebagai perangkat masukan untuk mengukur suhu udara di dalam ruangan. Sensor DHT11 dipilih karena memiliki antarmuka digital, konsumsi daya yang rendah, harga yang ekonomis, serta mudah diintegrasikan dengan Arduino Uno. Selain mampu mengukur suhu, sensor ini juga dapat mengukur kelembapan udara sehingga memberikan informasi lingkungan yang lebih lengkap. Walaupun tingkat akurasi masih berada di bawah sensor kelas industri, DHT11 telah banyak digunakan dalam penelitian sistem monitoring dan pengendalian suhu karena memiliki performa yang memadai untuk aplikasi prototipe dan media pembelajaran. Penelitian terbaru juga menunjukkan bahwa sensor DHT11 masih relevan digunakan pada sistem kendali suhu berbasis mikrokontroler karena memberikan hasil pembacaan yang stabil pada rentang suhu lingkungan normal (Akbar et al., 2024).

Selain sensor, aktuator merupakan komponen yang menentukan keberhasilan sistem dalam mengendalikan suhu ruangan. Pada penelitian ini digunakan kipas DC 12 Volt sebagai media pendingin dan lampu pijar sebagai sumber panas (heater). Kipas DC bekerja dengan meningkatkan sirkulasi udara sehingga mampu mempercepat proses pelepasan panas ketika suhu ruangan melebihi nilai setpoint. Sebaliknya, lampu pijar menghasilkan energi panas yang dimanfaatkan untuk meningkatkan suhu ruangan apabila suhu berada di bawah nilai yang telah ditentukan. Kombinasi kedua aktuator tersebut memungkinkan sistem melakukan pengendalian suhu secara dua arah (heating dan cooling), sehingga suhu ruangan dapat dipertahankan lebih stabil dibandingkan sistem yang hanya menggunakan satu jenis aktuator. Implementasi sistem pemanas dan pendingin berbasis Arduino telah banyak diterapkan pada berbagai penelitian karena mampu menghasilkan sistem yang sederhana, ekonomis, dan mudah dikembangkan (Noviyani & Sumariyah, 2019).

Keberhasilan suatu sistem kontrol tidak hanya dipengaruhi oleh sensor dan aktuator, tetapi juga sangat ditentukan oleh metode pengendalian yang digunakan. Metode pengendalian yang paling sederhana adalah On-Off Control, yaitu aktuator hanya bekerja dalam dua kondisi, yaitu hidup (ON) dan mati (OFF). Meskipun implementasinya mudah, metode ini sering menghasilkan fluktuasi suhu di sekitar nilai setpoint karena aktuator akan terus berganti kondisi setiap kali suhu melewati batas yang telah ditentukan. Kondisi tersebut menyebabkan sistem mengalami osilasi sehingga kurang sesuai untuk aplikasi yang membutuhkan kestabilan suhu tinggi (Kuo, 1995).

Untuk mengatasi kelemahan tersebut, salah satu metode yang paling banyak diterapkan pada sistem kendali adalah Proportional Integral Derivative (PID). Metode PID merupakan algoritma pengendalian umpan balik (feedback control) yang menggabungkan aksi proporsional (Proportional), integral (Integral), dan derivatif (Derivative). Aksi proporsional berfungsi memberikan respons berdasarkan besar kesalahan (error) antara suhu aktual dan nilai setpoint. Aksi integral bertujuan menghilangkan steady-state error sehingga keluaran sistem

semakin mendekati nilai yang diinginkan. Sementara itu, aksi derivatif berfungsi mempercepat respons dan mengurangi overshoot dengan memberikan prediksi terhadap perubahan error (Ogata, 2010; Nise, 2020).

METODE

Rancangan Penelitian

Penelitian ini menggunakan metode Research and Development (R&D) karena berfokus pada perancangan dan pengembangan prototipe sistem kontrol suhu berbasis Arduino Uno dengan metode Proportional, Integral, and Derivative (PID). Tahapan penelitian meliputi penentuan kebutuhan sistem, perancangan perangkat keras dan perangkat lunak, perakitan komponen, penerapan program pengendalian pada Arduino Uno, serta pengujian sistem. Kinerja sistem selanjutnya dianalisis berdasarkan perubahan suhu dan karakteristik respons dalam mencapai serta mempertahankan suhu di sekitar nilai setpoint yang telah ditentukan.

Prinsip Kerja Sistem

DHT11 membaca suhu aktual di dalam kotak akrilik dan mengirimkan data ke Arduino Uno. Suhu aktual dibandingkan dengan setpoint 26 °C untuk memperoleh error. Error diproses oleh algoritma PID dan hasilnya digunakan sebagai dasar pengaturan keluaran aktuator. Ketika suhu berada di bawah setpoint, lampu bekerja untuk menaikkan suhu. Ketika suhu berada di atas setpoint, kipas bekerja untuk menurunkan suhu. Perubahan suhu akibat kerja aktuator kembali dibaca oleh sensor sehingga terbentuk sistem kontrol tertutup.

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt}$$

dengan $u(t)$ sebagai keluaran pengendali, $e(t)$ sebagai selisih antara setpoint dan suhu aktual, serta K_p , K_i , dan K_d sebagai parameter pengendali PID. Nilai K_p , K_i , dan K_d yang digunakan adalah $K_p = [\text{isi sesuai program}]$, $K_i = [\text{isi sesuai program}]$, dan $K_d = [\text{isi sesuai program}]$. Keluaran pengendali diterapkan melalui sinyal Pulse Width Modulation (PWM) untuk menyesuaikan tingkat kerja aktuator.

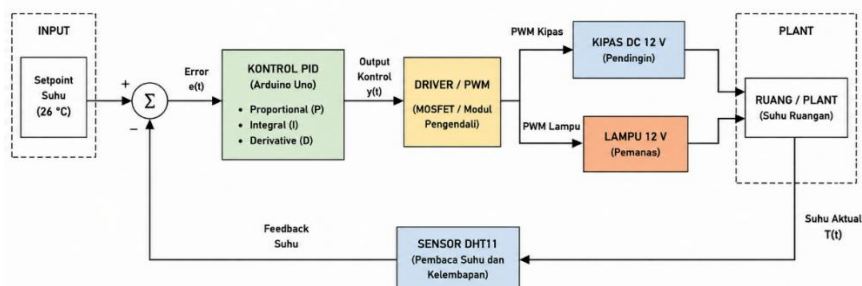
Prosedur Pengujian

Pengujian dilakukan secara bertahap terhadap sensor DHT11, kipas DC 12 V, lampu pemanas, dan sistem PID secara keseluruhan. Pada pengujian sistem PID, setpoint ditetapkan sebesar 26 °C. Lima percobaan dilakukan dari suhu awal 24 °C untuk mengevaluasi proses pemanasan dan lima percobaan dari suhu awal 29 °C untuk mengevaluasi proses pendinginan. Data yang diamati mencakup suhu awal, suhu akhir, waktu respons, serta perilaku suhu ketika mendekati setpoint.

Analisis respons sistem menggunakan waktu respons menuju setpoint sebagai parameter utama yang tersedia secara kuantitatif. Overshoot, undershoot, settling time, steady-state error, dan kestabilan diamati dari pola respons suhu. Nilai numerik parameter tersebut hanya dapat ditetapkan apabila data deret waktu setiap percobaan tersedia secara lengkap; karena itu artikel ini tidak memberikan angka yang tidak tercatat pada data sumber.

Blok Diagram

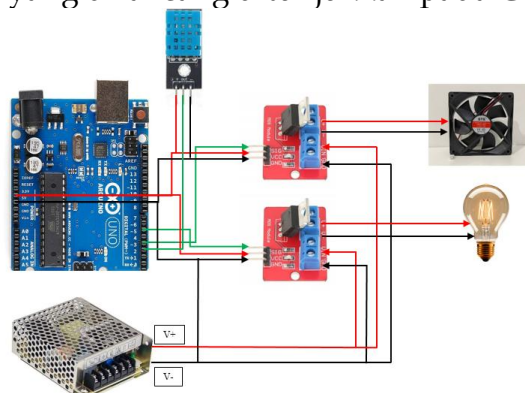
Dalam penelitian ini, perancangan alat mengacu pada blok diagram sistem yang disajikan pada Gambar 1 di bawah ini. Diagram tersebut menggambarkan hubungan antar komponen sekaligus mekanisme kerja sistem secara keseluruhan.



Gambar 1. Blok Diagram

Rangkaian Sistem

Berdasarkan blok diagram yang telah dijelaskan, bentuk skematik dari sistem kontrol suhu ruangan yang dirancang ditunjukkan pada Gambar 2 di bawah ini.

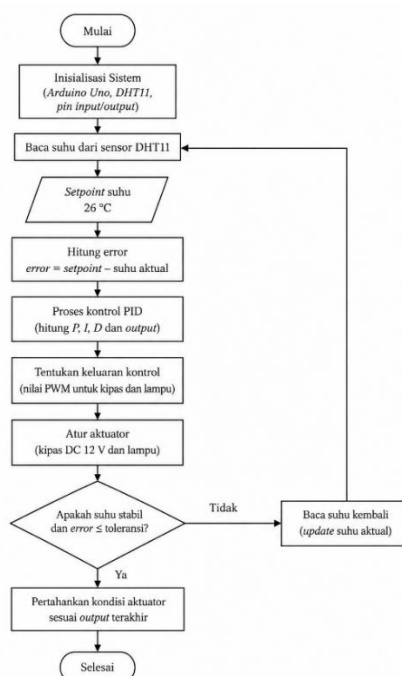


Gambar 2. Rangkaian Sistem

Pada Gambar 2 di atas menjelaskan hubungan antara input (sensor), proses (Arduino), dan output (aktuator). Sensor DHT11 berperan sebagai input untuk mengambil angka suhu saat ini dan dikirim ke Arduino Uno yang tersambung dengan pin 2. Arduino Uno sebagai mikrokontroler menjalankan empat tahap PID yang hasilnya akan dikirim ke aktuator kipas dan lampu dengan modul IRF520 sebagai jembatannya dengan pin 5 dan 10. Selanjutnya IRF520 mengirim pesan dari Arduino Uno dengan data yang sudah berupa sinyal PWM, dan kecepatan serta besar kecilnya intensitas cahaya bergantung pada besar kecilnya sinyal PWM yang dikirim lewat IRF520.

Diagram Alir

Proses kerja dan pengambilan keputusan pada sistem kontrol suhu ini dipresentasikan dalam bentuk flowchart pada Gambar 3 di bawah ini.



Gambar 3. Diagram Alir

HASIL DAN PEMBAHASAN

Hasil Pengujian Komponen

Pengujian awal menunjukkan sensor DHT11, kipas, dan lampu dapat bekerja sebagai bagian dari sistem. Tegangan terukur pada sensor DHT11 sebesar 3,3 V, kipas DC sekitar 12,5 V, dan lampu pemanas 12 V. Kipas dapat menyala dan berhenti sesuai perintah pendinginan, sedangkan lampu dapat merespons perintah pemanasan. Hasil tersebut menjadi dasar untuk melanjutkan pengujian sistem PID secara terintegrasi.

Pengujian Pendinginan

Pengujian pendinginan dimulai pada suhu 29 °C dan diarahkan menuju setpoint 26 °C. Seluruh lima percobaan mencapai suhu akhir 26 °C. Waktu respons berkisar antara 225 sampai 266 detik. Percobaan kedua memberikan waktu tercepat sebesar 225 detik, sedangkan percobaan ketiga memberikan waktu terlama sebesar 266 detik.

Tabel 1. Pengujian Pendinginan

Percobaan	Suhu awal (°C)	Setpoint (°C)	Suhu akhir (°C)	Waktu respons
1	29	26	26	3 menit 59 detik
2	29	26	26	3 menit 45 detik
3	29	26	26	4 menit 26 detik
4	29	26	26	4 menit 20 detik
5	29	26	26	4 menit 20 detik

Pengujian pendinginan dilakukan sebanyak lima kali dari suhu awal 29 °C menuju setpoint 26 °C. Waktu respons yang diperoleh berturut-turut adalah 3 menit 59 detik, 3 menit 45 detik, 4 menit 26 detik, 4 menit 20 detik, dan **4 menit 20 detik**.

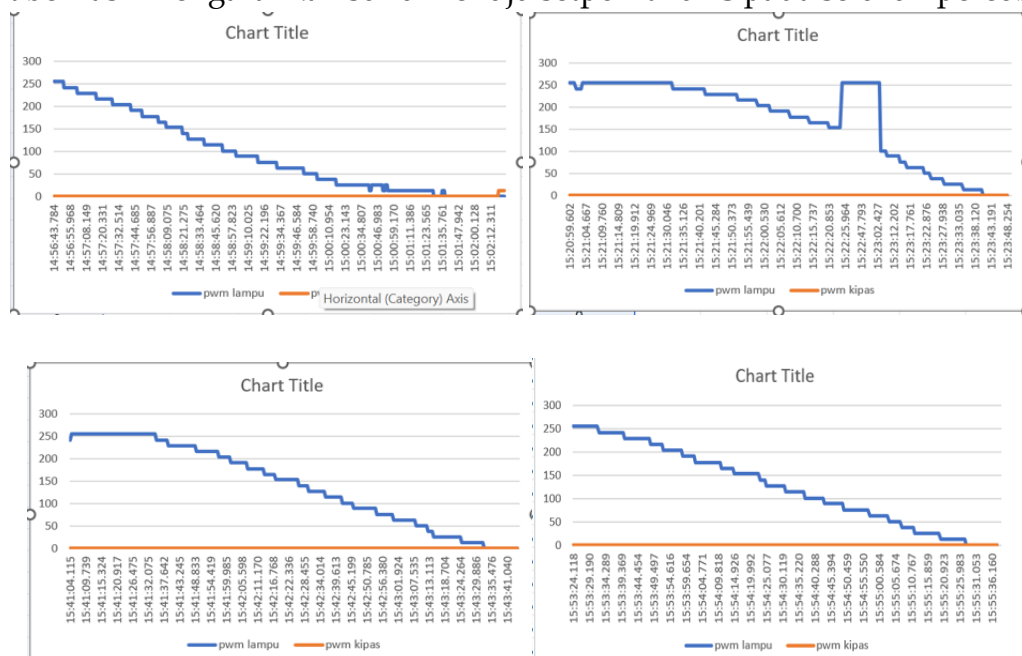
Rata-rata waktu yang dibutuhkan sistem untuk menurunkan suhu sebesar 3 °C adalah sekitar 4 menit 10 detik.

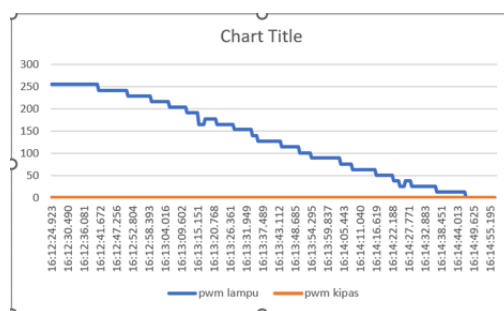
Waktu respons tercepat diperoleh pada percobaan kedua sebesar 3 menit 45 detik, sedangkan waktu terlama terjadi pada percobaan ketiga sebesar 4 menit 26 detik. Selisih antara waktu tercepat dan terlama sebesar 41 detik. Rentang tersebut lebih kecil dibandingkan dengan pengujian pemanasan, sehingga pada proses pendinginan menunjukkan hasil yang lebih berdekatan. Kesamaan waktu respons pada lima percobaan menunjukkan bahwa kerja kipas dalam menurunkan suhu memiliki kecenderungan yang lebih konsisten.

Ketika suhu masih jauh dari setpoint, sistem memberikan keluaran yang lebih besar kepada kipas. Keluaran tersebut kemudian berkurang secara bertahap ketika suhu semakin mendekati 26 °C. Pola kerja ini membuat proses pendinginan berlangsung secara bertahap hingga suhu mencapai nilai yang ditentukan.

Rata-rata waktu pendinginan sebesar 4 menit 10 detik memang lebih lama dibandingkan rata-rata waktu pemanasan sebesar 3 menit 6 detik. Namun, kedua nilai tersebut tidak dapat dibandingkan hanya berdasarkan waktu total karena suhu pada kedua pengujian berbeda. Pada proses pemanasan, sistem hanya menaikkan suhu sebesar 2 °C dari 24 °C menjadi 26 °C, sedangkan pada proses pendinginan sistem harus menurunkan suhu sebesar 3 °C dari 29 °C menjadi 26 °C. Jika dilihat berdasarkan perubahan suhu rata-rata terhadap waktu, proses pemanasan memiliki laju rata-rata sekitar 0,64 °C per menit, sedangkan proses pendinginan sekitar 0,72 °C per menit. Nilai tersebut menunjukkan bahwa meskipun waktu total pendinginan lebih lama, perubahan suhu per satuan waktu pada proses pendinginan tidak lebih lambat daripada proses pemanasan.

Hasil pengujian secara keseluruhan menunjukkan bahwa sistem kontrol mampu memberikan respons terhadap dua kondisi yang berbeda. Ketika suhu berada di bawah setpoint, sistem mengaktifkan proses pemanasan, sedangkan ketika suhu berada di atas setpoint, sistem menjalankan proses pendinginan. Kedua kondisi tersebut berhasil mengarahkan suhu menuju setpoint 26 °C pada seluruh percobaan.





Gambar 4. Grafik Pengujian Pendinginan

Keterangan: sumbu x = waktu; sumbu y = nilai PWM aktuator.

Pengujian Pemanasan

Pengujian pemanasan dimulai pada suhu 24 °C menuju setpoint 26 °C.

Tabel 2. Pengujian Pemanasan

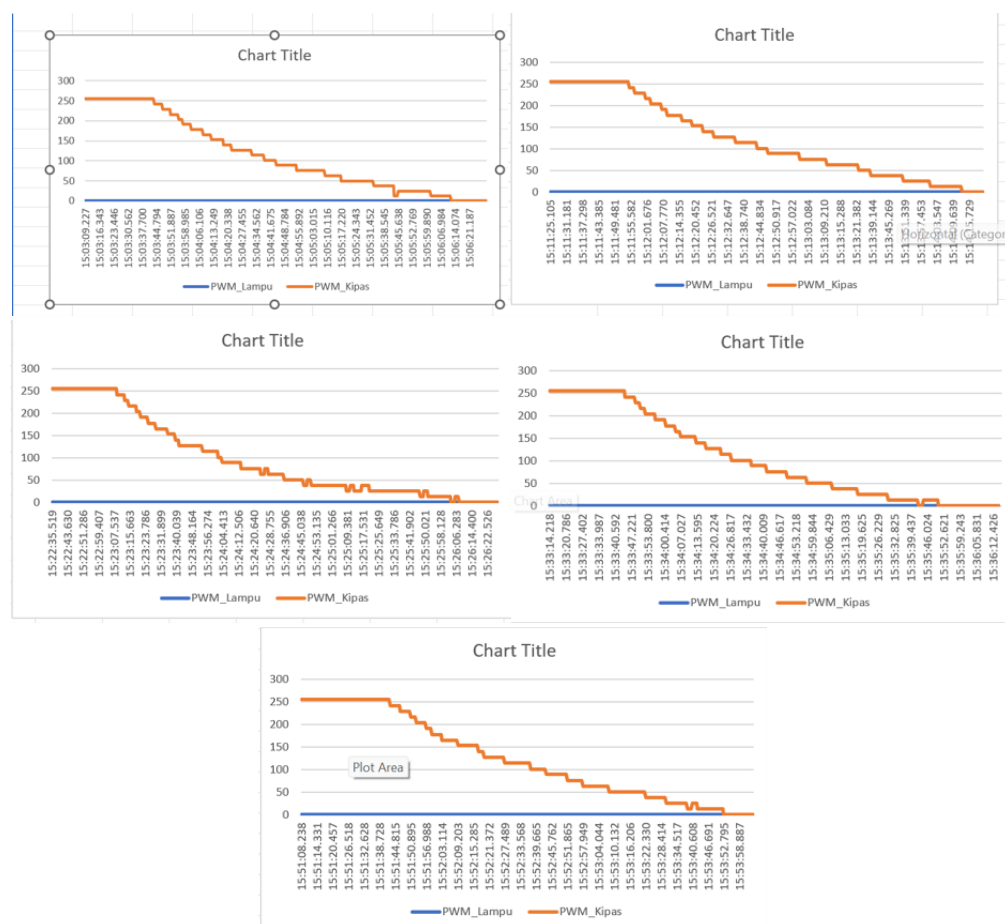
Percobaan	Suhu awal (°C)	Setpoint (°C)	Suhu akhir (°C)	Waktu respons
1	24	26	26	4 menit 30 detik
2	24	26	26	3 menit
3	24	26	26	2 menit 40 detik
4	24	26	26	2 menit 46 detik
5	24	26	26	2 menit 35 detik

Pengujian pemanasan dilakukan sebanyak lima kali dari suhu awal 24 °C menuju setpoint 26 °C. Waktu respons yang diperoleh berturut-turut adalah 4 menit 30 detik, 3 menit, 2 menit 40 detik, 2 menit 46 detik, dan 2 menit 35 detik. Berdasarkan hasil tersebut, rata-rata waktu yang dibutuhkan sistem untuk menaikkan suhu sebesar 2 °C adalah sekitar 3 menit 6 detik.

Waktu respons tercepat diperoleh pada percobaan kelima sebesar 2 menit 35 detik, sedangkan waktu terlama terjadi pada percobaan pertama sebesar 4 menit 30 detik. Selisih antara waktu tercepat dan terlama mencapai 1 menit 55 detik. Percobaan kedua sampai kelima memiliki waktu respons yang lebih berdekatan, yaitu antara 2 menit 35 detik sampai 3 menit, sedangkan percobaan pertama membutuhkan waktu yang lebih lama dibandingkan percobaan lainnya.

Perbedaan tersebut menunjukkan bahwa meskipun suhu awal dan setpoint dibuat sama, kecepatan perubahan suhu pada setiap percobaan tidak sepenuhnya identik. Kondisi termal ruang pengujian pada saat pengujian dimulai dapat memengaruhi proses perpindahan panas. Selain itu, kondisi lingkungan dan panas yang tersimpan pada ruang pengujian juga dapat menyebabkan waktu yang dibutuhkan untuk mencapai setpoint menjadi berbeda.

Secara keseluruhan, kelima percobaan menunjukkan bahwa sistem mampu menaikkan suhu dari 24 °C menuju 26 °C. Berdasarkan rata-rata hasil pengujian, sistem membutuhkan waktu sekitar 3 menit 6 detik untuk meningkatkan suhu sebesar 2 °C. Hal ini menunjukkan bahwa lampu pemanas yang dikendalikan oleh sistem dapat memberikan respons pemanasan hingga suhu mencapai nilai yang telah ditentukan.



Gambar 5. Grafik Pengujian Pemanasan

Keterangan: sumbu x = waktu; sumbu y = nilai PWM aktuator.

Respons di Sekitar Setpoint

Ketika suhu mendekati 26 °C, keluaran lampu maupun kipas menurun sehingga perubahan suhu menjadi lebih terkendali. Pada salah satu pengujian pendinginan, suhu mencapai 26 °C dan bertahan pada nilai tersebut hingga akhir pengamatan tanpa turun di bawah setpoint. Kondisi ini menunjukkan tidak adanya undershoot yang terlihat pada percobaan tersebut.

Penilaian overshoot, undershoot, settling time, dan steady-state error untuk seluruh percobaan memerlukan data deret waktu yang lengkap. Dokumen sumber menjelaskan bahwa parameter tersebut diamati melalui grafik, tetapi belum menyediakan rekap numerik untuk setiap percobaan (Bolton, 2021). Oleh karena itu, kesimpulan kuantitatif mengenai besar overshoot atau settling time tidak boleh dibuat hanya berdasarkan suhu akhir. Temuan yang dapat dipertahankan dari data yang tersedia adalah bahwa seluruh sepuluh pengujian mencapai 26 °C dan sistem dapat mempertahankan suhu di sekitar setpoint selama pengamatan.

Implikasi Kinerja Sistem

Hasil pengujian memperlihatkan fungsi kontrol dua arah: lampu memberikan aksi ketika suhu berada di bawah setpoint dan kipas memberikan aksi ketika suhu

berada di atas setpoint. Pengaturan keluaran yang berkurang saat error mengecil merupakan karakteristik yang dibutuhkan agar aktuator tidak terus bekerja pada keluaran maksimum ketika suhu telah mendekati target.

Keterbatasan utama prototipe terletak pada sensor DHT11 dan kelengkapan pencatatan data. Resolusi dan respons sensor membatasi kemampuan untuk mengevaluasi perubahan suhu yang sangat kecil. Selain itu, tanpa pencatatan otomatis terhadap waktu, suhu, error, dan PWM pada setiap interval, parameter respons PID tidak dapat dihitung secara konsisten untuk seluruh percobaan. Pengembangan data logging dan pengujian beberapa kombinasi tuning PID diperlukan agar performa pengendali dapat dibandingkan secara kuantitatif.

Perbandingan dengan Penelitian Terdahulu

Penelitian ini memiliki persamaan dengan Megido & Ariyanto (2016) dan Noviyani & Sumariyah (2019) yang menggunakan PID berbasis Arduino untuk pengendalian suhu. Perbedaannya, penelitian ini menggunakan ruang akrilik dengan aktuator pemanas lampu dan pendingin kipas DC sehingga pengujian mencakup dua arah. Akbar et al. (2024) juga menerapkan PID pada Arduino Uno untuk optimasi suhu udara; penelitian ini melengkapi kajian tersebut pada skala prototipe ruang akrilik. Tyas & Sumiharto (2013) menekankan studi kendali suhu ruang; penelitian ini menunjukkan respons serupa, tetapi masih terbatas pada sensor DHT11 dan pencatatan data manual. Dengan demikian, kontribusi penelitian ini terletak pada implementasi PID dua arah yang sederhana dan murah untuk ruang uji prototipe.

KESIMPULAN

Berdasarkan penelitian dengan metode Research and Development (R&D), telah dihasilkan sebuah sistem kontrol suhu berbasis Arduino Uno dengan metode PID yang dapat digunakan untuk mengatur suhu pada ruang pengujian. Secara ringkas, hasil penelitian menunjukkan tiga hal utama: (1) sistem berhasil dirancang dengan sensor DHT11, lampu pemanas, kipas DC 12 V, dan driver MOSFET; (2) sistem mampu mengarahkan suhu menuju setpoint 26 °C baik dari kondisi di bawah maupun di atas setpoint; dan (3) rata-rata waktu respons pemanasan adalah sekitar 3 menit 6 detik, sedangkan rata-rata waktu respons pendinginan adalah sekitar 4 menit 10 detik. Kontribusi penelitian ini adalah tersedianya prototipe sistem kontrol suhu dua arah berbasis Arduino Uno yang ekonomis dan dapat menjadi dasar pengembangan sistem kontrol termal pada skala lebih luas.

REKOMENDASI

Untuk penelitian selanjutnya, disarankan agar penggunaan sensor DHT11 digantikan atau dilengkapi dengan sensor suhu yang lebih akurat dan responsif, seperti DS18B20 atau thermocouple, agar perubahan suhu kecil dapat terdeteksi secara lebih presisi. Selain itu, sistem perlu dilengkapi dengan data logging otomatis untuk merekam waktu, suhu, error, dan sinyal PWM secara berkala sehingga parameter overshoot, undershoot, settling time, dan steady-state error dapat

dianalisis secara kuantitatif. Pengujian juga sebaiknya dilakukan dengan variasi nilai tuning PID, variasi suhu awal yang lebih luas, serta jumlah percobaan yang lebih banyak, termasuk pada ruang dengan ukuran dan kondisi lingkungan yang berbeda, agar karakteristik sistem dapat diketahui secara lebih komprehensif. Dengan langkah-langkah tersebut, prototipe yang dikembangkan tidak hanya terbatas pada skala laboratorium, tetapi juga berpotensi diadaptasi untuk kebutuhan pengendalian suhu ruangan yang lebih luas.

REFERENSI

- Akbar, A. H., Ma'arif, A., Rekik, C., Abougarair, A. J., & Mekonnen, A. M. (2024). Implementing PID control on Arduino Uno for air temperature optimization. *Buletin Ilmiah Sarjana Teknik Elektro*, 6(1), 1–13. <https://doi.org/10.12928/biste.v6i1.9725>
- Arduino. (2024). Arduino UNO R3 datasheet. Arduino.
- Åström, K. J., & Murray, R. M. (2021). *Feedback systems: An introduction for scientists and engineers* (2nd ed.). Princeton University Press.
- Baskoro, F., Rohman, M., Arya S. F., D., & Putra, A. (2024). Temperature control of focused beam lamp: PID-controlled pulse width modulation using MATLAB Simulink and Arduino. *JE-Unisla*, 9(2), 97–109. <https://doi.org/10.30736/jesunisla.v9i2.1223>
- D-Robotics. (2010). DHT11 humidity & temperature sensor [Datasheet]. D-Robotics.
- Hadi, Y., Azaria, T., Purnomo, Putrianto, N. K., Oktiarso, T., Ekawati, Y., & Noya, S. (2020). Analisis kenyamanan termal ruang kuliah. *Metris: Jurnal Sains dan Teknologi*, 21(1), 13–26. <https://doi.org/10.25170/metris.v21i01.2428>
- Kuo, B. C. (1995). *Automatic control systems* (7th ed.). Prentice Hall.
- 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(4), 21–28. <https://doi.org/10.14710/gt.v18i4.21912>
- Nise, N. S. (2020). *Control systems engineering* (8th ed.). Wiley.
- Noviyani, S., & Sumariyah, S. (2019). Rancang bangun sistem pengendali temperatur dengan proporsional, integral dan diferensial (PID) berbasis mikrokontroler Arduino Uno. *Berkala Fisika*, 22(2), 62–67.
- Ogata, K. (2010). *Modern control engineering* (5th ed.). Pearson.
- Tyas, D. A., & Sumiharto, R. (2013). Purwarupta sistem kendali PID: Studi kasus kendali suhu ruang. *IJES (Indonesian Journal of Electronics and Instrumentation Systems)*, 3(1), 95–104. <https://doi.org/10.22146/ijes.3875>
- Ziegler, J. G., & Nichols, N. B. (1942). Optimum settings for automatic controllers. *Transactions of the ASME*, 64(11), 759–768.
- Cohen, G. H., & Coon, G. A. (1953). Theoretical consideration of retarded control. *Transactions of the ASME*, 75, 827–834.
- Åström, K. J., & Hägglund, T. (2006). *Advanced PID control*. ISA.
- Visioli, A. (2006). *Practical PID control*. Springer.
- Franklin, G. F., Powell, J. D., & Emami-Naeini, A. (2019). *Feedback control of dynamic systems* (8th ed.). Pearson.
- Bolton, W. (2021). *Mechatronics: Electronic control systems in mechanical and electrical engineering* (7th ed.). Pearson.

- Banzy, M., & Shiloh, M. (2022). Getting started with Arduino (4th ed.). Make: Community.
- Ibrahim, D. (2019). Microcontroller projects in C for the 8051. Newnes.