Dispenser Air Minum dengan Pengaturan Suhu Menggunakan Metode PID

Raras Azra Farhani[1], Septriandi Wirayoga[2]

[1,2] Prodi Teknik Elektronika Kampus Kediri, Politeknik Negeri Malang, Indonesia
Email: [1]rarasazra@gmail.com, [2]yoga.septriandi@polinema.ac.id

Nomor Handphone : 081615245819

Abstrak

Dispenser air minum yang tersedia di pasaran umumnya hanya menyediakan pilihan suhu panas, normal, dan dingin sehingga pengguna tidak dapat mengatur suhu air sesuai kebutuhannya. Selain itu, proses pengambilan air masih dilakukan secara manual. Penelitian ini bertujuan merancang dispenser air minum yang mampu mengatur suhu sesuai *set point* dan volume yang ditentukan pengguna. Untuk memperoleh pengendalian suhu yang stabil, diterapkan metode PID *Gain Scheduling* yang menyesuaikan parameter Kp, Ki, dan Kd pada setiap rentang suhu. Hasil pengujian menunjukkan bahwa sistem yang dirancang mampu mengatur suhu air sesuai *set point* serta mengeluarkan air berdasarkan volume yang dipilih pengguna. Penerapan PID *Gain Scheduling* menghasilkan respon pengendalian suhu yang lebih baik dibandingkan sistem tanpa PID, ditunjukkan dengan penurunan rata-rata *overshoot* pada sistem pemanas dari 4,07 °C menjadi 0,71 °C dan pada sistem pendingin dari 0,99 °C menjadi 0,16 °C. Selain itu, *steady-state error* sistem pendingin menurun dari 0,35 °C menjadi 0,07 °C, sedangkan sensor *water flow* yang telah dikalibrasi memiliki tingkat akurasi sebesar 98.03% untuk sensor *water flow* pada air panas dan 97,06% untuk sensor *water flow* pada air dingin. Berdasarkan hasil tersebut, metode PID *Gain Scheduling* mampu meningkatkan kestabilan pengendalian suhu sehingga dispenser dapat menghasilkan suhu air yang lebih mendekati *set point* dibandingkan sistem tanpa PID. Selain itu, sistem juga mampu mengeluarkan volume air yang mendekati volume yang diinginkan pengguna.

Kata kunci: dispenser, pengendalian suhu, PID *gain scheduling*.

Abstract

Commercially available drinking water dispensers generally only offer hot, normal, and cold temperature options, preventing users from adjusting the water temperature to their needs. Furthermore, the water draw process is still manual. This research aims to design a drinking water dispenser capable of regulating the temperature according to the user's set point and volume. To achieve stable temperature control, a PID Gain Scheduling method was applied, adjusting the Kp, Ki, and Kd parameters for each temperature range. Test results showed that the designed system was able to regulate the water temperature according to the set point and dispense water based on the user's selected volume. The application of PID Gain Scheduling resulted in a better temperature control response than a system without PID, as evidenced by a decrease in the average overshoot in the heating system from 4.07°C to 0.71°C and in the cooling system from 0.99°C to 0.16°C. In addition, the cooling system's steady-state error decreased from 0.35°C to 0.07°C, while the calibrated water flow sensor had an accuracy level of 98.03% for the water flow sensor on hot water and 97.06% for the water flow sensor on cold water. Based on these results, the PID Gain Scheduling method can improve temperature control stability, allowing the dispenser to produce water temperatures closer to the set point than a system without PID. In addition, the system is also capable of dispensing water volumes close to the volume desired by the user.

Keywords: *dispenser, PID gain scheduling, temperature control.*

This work is an open access article and licensed under a Creative Commons Attribution-NonCommercial ShareAlike 4.0 International (CC BY-NC-SA 4.0)

1. PENDAHULUAN

Sebagai kebutuhan pokok manusia, air minum memiliki peran besar dalam menjaga kesehatan dan mendukung aktivitas sehari-hari. Salah satu alat rumah tangga yang banyak dipakai untuk memenuhi kebutuhan tersebut adalah dispenser, karena mampu menyediakan air minum lebih praktis. Dengan adanya sistem pemanas dan pendingin pada dispenser, pengguna tak perlu lagi repot memanaskan maupun mendinginkan air secara manual. Meski demikian, dispenser yang beredar di pasaran umumnya

masih terbatas pada pilihan suhu yang sudah ditetapkan pabrik, yaitu panas, normal, dan dingin, sehingga pengguna belum bisa menentukan sendiri seberapa panas atau dingin suhu air yang diinginkan [1].

Kebutuhan suhu air setiap orang berbeda-beda tergantung keperluannya. Contohnya, menyeduh teh, kopi, susu, atau minuman instan membutuhkan suhu air tertentu agar cita rasa dan kualitas sajiannya maksimal. Karena itu, dispenser yang fleksibel dalam mengatur suhu air sesuai kebutuhan pengguna menjadi penting untuk dikembangkan. Selain aspek suhu, kepraktisan saat mengambil air juga patut diperhatikan. Untuk itu, dispenser pada penelitian ini dirancang agar dapat mengeluarkan air secara otomatis sesuai volume yang telah dipilih, sehingga pengguna tak perlu lagi membuka keran secara manual dan proses pengambilan air menjadi lebih praktis.

Supaya dispenser dapat menghasilkan suhu sesuai pengaturan pengguna, dibutuhkan sistem pengendalian yang mampu menjaga suhu air tetap berada di sekitar *set point*. Jika hanya mengandalkan kontrol *ON-OFF*, elemen pemanas maupun sistem pendingin akan menyala penuh saat suhu berada di bawah atau di atas *set point*, lalu berhenti begitu *set point* tercapai. Akibatnya, suhu air cenderung berosilasi terus-menerus di sekitar *set point* sehingga kestabilannya sulit dijaga. Mengacu pada persoalan tersebut, penelitian ini merancang dispenser air minum yang memberi keleluasaan bagi pengguna untuk menentukan sendiri suhu air sesuai kebutuhan, pada rentang suhu yang telah ditetapkan. Sistem juga dilengkapi kemampuan mengeluarkan air secara otomatis mengikuti volume yang dipilih.

Untuk mencapai dan mempertahankan suhu air yang diinginkan, digunakan metode Proportional Integral Derivative (PID), yang bekerja dengan membandingkan suhu aktual hasil pembacaan sensor terhadap suhu *set point* yang ditentukan. Selisih antara keduanya disebut *error*, dan dari nilai *error* inilah pengendali PID menghitung besar sinyal PWM yang dikirim ke elemen pemanas maupun sistem pendingin. Saat suhu masih jauh dari *set point*, PWM akan diperbesar agar suhu bergerak mendekati *set point*. Sebaliknya, ketika suhu sudah mendekati *set point*, nilai PWM diturunkan secara bertahap sehingga perubahan suhu berlangsung lebih halus dan potensi *overshoot* dapat ditekan.

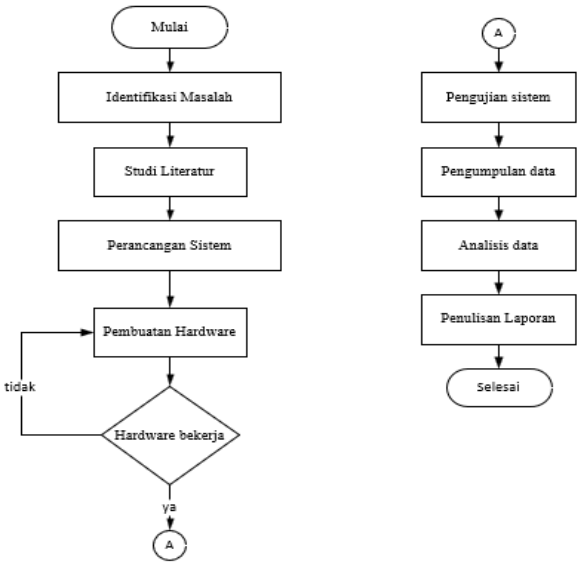
Walaupun pengendalian suhu dengan PID lebih unggul dibandingkan kontrol *ON-OFF*, penggunaan parameter Kp, Ki, dan Kd yang bernilai tetap belum tentu menghasilkan performa optimal di seluruh rentang suhu. Penyebabnya, karakteristik sistem pemanas maupun sistem pendingin berubah-ubah pada setiap rentang suhu. Misalnya, saat proses pemanasan, energi panas yang masih tersimpan pada elemen pemanas dan air dapat membuat suhu terus naik meski daya pemanas sudah dikurangi. Kondisi semacam ini berisiko menimbulkan *overshoot* apabila hanya satu set parameter PID yang dipakai untuk seluruh rentang suhu. Karena itulah, penelitian ini menerapkan metode PID *Gain Scheduling*, yaitu penyesuaian parameter Kp, Ki, dan Kd berdasarkan rentang suhu tertentu. Dengan menyesuaikan parameter pengendali terhadap perubahan karakteristik sistem, penerapan PID *Gain Scheduling* diharapkan mampu meningkatkan kinerja pengendalian suhu dispenser sehingga suhu air dapat dicapai dan dipertahankan lebih dekat dengan *set point*.

2. METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini adalah metode *Research and Development* atau riset dan pengembangan. Tujuan dari penelitian pengembangan adalah memperluas atau memperdalam pengetahuan yang telah ada sebelumnya. Penelitian ini dirancang untuk mengembangkan dispenser yang memungkinkan penyesuaian suhu air sesuai dengan preferensi pengguna dengan menggunakan metode PID. Untuk mencapai dan mempertahankan suhu tersebut, sistem akan mengatur besar kecilnya daya yang diberikan ke elemen pemanas dan pendingin.

2.1 Tahapan Penelitian

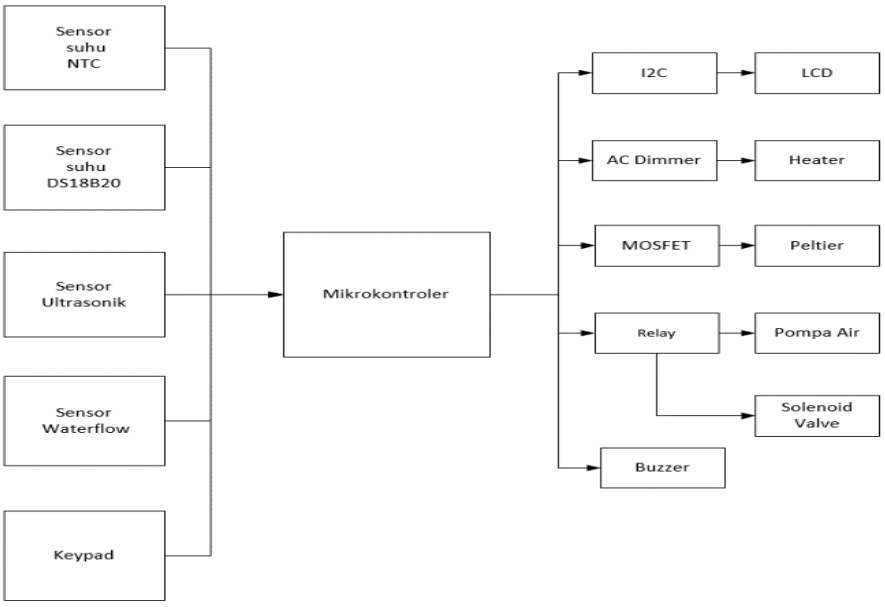
Tahapan penelitian yang diterapkan dalam proyek ini mengikuti alur yang terstruktur, dimulai dari identifikasi masalah hingga penyusunan laporan. Hal ini memungkinkan untuk mencapai tujuan penelitian secara efektif dan efisien.



Gambar 1. Tahapan Penelitian

Tahapan penelitian dimulai dengan identifikasi masalah untuk menentukan permasalahan yang akan diteliti. Selanjutnya dilakukan observasi dan studi literatur untuk memperoleh informasi dari sumber ilmiah yang relevan. Data yang diperoleh digunakan dalam perancangan sistem, meliputi alur kerja, skema komponen, perangkat keras, dan integrasi sistem. Setelah itu dilakukan pengujian sistem untuk memastikan seluruh komponen bekerja sesuai perancangan dan melakukan perbaikan apabila ditemukan kesalahan. Hasil pengujian kemudian digunakan untuk pengumpulan dan analisis data sebagai dasar evaluasi kinerja sistem. Tahap terakhir adalah penulisan laporan berdasarkan hasil penelitian yang telah diperoleh.

2.2 Diagram Blok Sistem



Gambar 2. Diagram Blok Sistem

Diagram blok sistem terdiri dari dua mikrokontroler, yaitu Arduino Mega 2560 dan ESP32. Arduino Mega 2560 berfungsi sebagai pengendali utama yang mengatur seluruh proses kerja sistem, mulai dari pembacaan sensor, pengolahan data, perhitungan kendali PID *Gain Scheduling*, hingga pengendalian aktuator. Sementara itu, ESP32 berfungsi sebagai pengendali sistem pemanas dengan

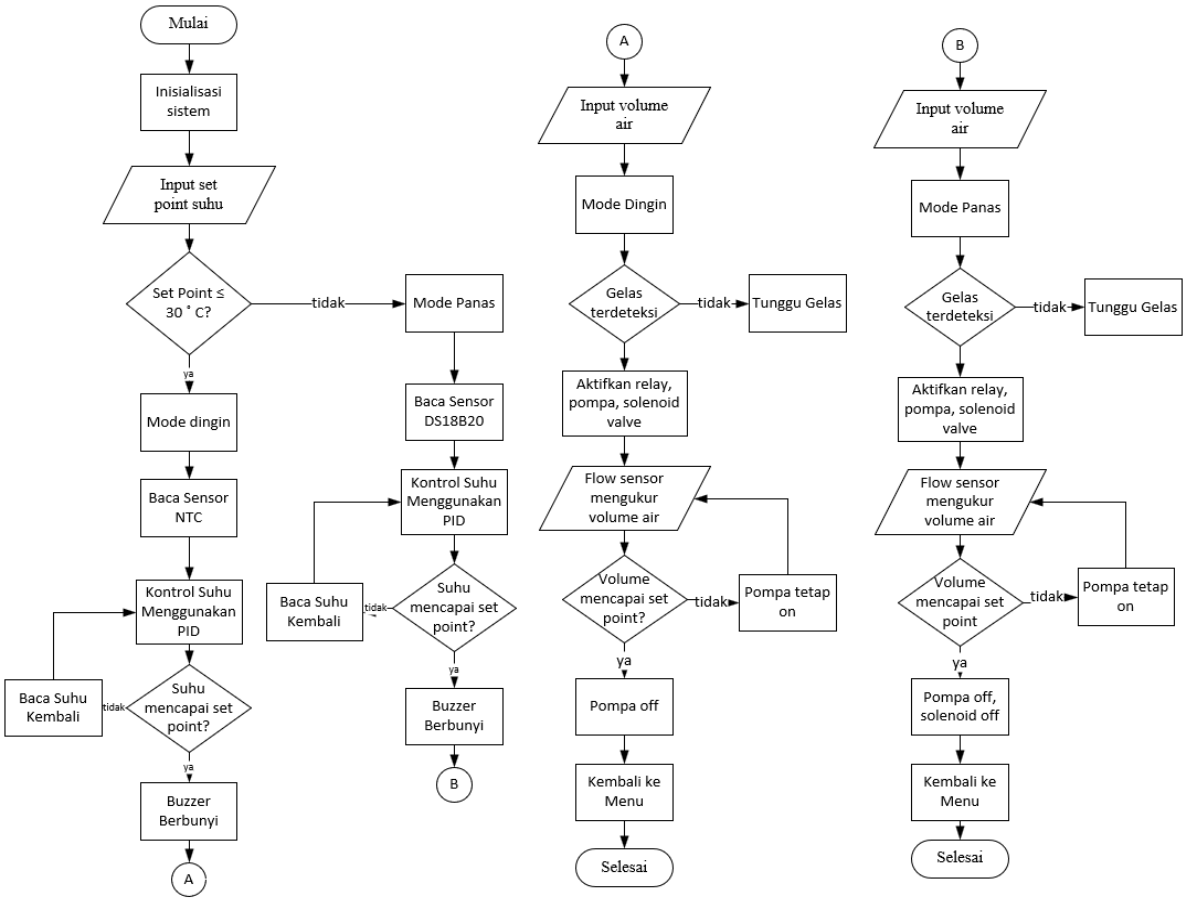
menerima data *set point* dan nilai PWM dari Arduino Mega 2560. Data tersebut digunakan untuk mengatur kerja *heater* melalui AC Dimmer.

Pada sistem pengukuran suhu, sensor DS18B20 digunakan untuk mengukur suhu air pada tabung pemanas sebagai umpan balik dalam proses pengendalian suhu air panas. Sementara itu, sensor NTC digunakan untuk mengukur suhu air pada tabung pendingin sebagai umpan balik dalam proses pengendalian suhu air dingin. Selain sensor suhu, sensor ultrasonik HC-SR04 digunakan untuk mendeteksi keberadaan gelas pada area pengisian. Proses pengisian air hanya dapat dilakukan apabila gelas terdeteksi pada jarak yang telah ditentukan. Sensor *water flow* digunakan untuk mengukur debit aliran air yang keluar dari pompa sehingga volume air yang dikeluarkan dapat disesuaikan dengan volume yang dipilih oleh pengguna.

Dalam proses pengendalian suhu, AC Dimmer digunakan untuk mengatur besar kecilnya daya listrik yang diberikan kepada elemen pemanas berdasarkan nilai PWM hasil perhitungan PID *Gain Scheduling*. Selanjutnya, *heater* berfungsi untuk memanaskan air pada tabung pemanas menggunakan energi listrik. Pada sistem pendingin, MOSFET digunakan untuk mengatur daya listrik yang diberikan kepada Peltier berdasarkan nilai PWM hasil perhitungan PID *Gain Scheduling*. Peltier berfungsi sebagai pendingin termoelektrik yang digunakan untuk mendinginkan air pada tabung pendingin hingga mencapai nilai *set point* yang telah ditentukan pengguna.

Selain sistem pengaturan suhu, relay digunakan sebagai sakelar elektronik untuk mengendalikan pompa DC dan solenoid *valve* sesuai perintah dari mikrokontroler. Pompa DC berfungsi untuk mengalirkan air dari tabung menuju keran selama proses pengisian berlangsung, sedangkan solenoid *valve* berfungsi untuk membuka dan menutup saluran air menuju gelas sehingga aliran air dapat dikendalikan secara otomatis. Buzzer digunakan sebagai indikator suara ketika suhu air telah mencapai *set point* yang ditentukan. Seluruh informasi mengenai kondisi sistem, seperti suhu aktual, *set point*, mode operasi, dan volume air, ditampilkan melalui LCD.

2.3 *Flowchart* Sistem



Gambar 3. *Flowchart* Sistem

Proses kerja sistem dimulai dengan inisialisasi sistem, yaitu mikrokontroler ardiuno mega mengaktifkan seluruh komponen yang digunakan, meliputi esp 32, LCD, keypad, sensor suhu DS18B20, sensor NTC, *heater*, sensor ultrasonik, ac dimmer, mosfet, Peltier, relay, pompa, solenoid *valve*, *flow* sensor, buzzer, serta parameter kontrol PID. Setelah proses inisialisasi selesai, sistem berada pada kondisi siap menerima masukan dari pengguna. Selanjutnya, pengguna memasukkan nilai *set point* suhu yang diinginkan melalui keypad. Apabila rentang nilai *set point* yang diinginkan 17 °C hingga 30 °C, pengguna bisa mode dingin. Namun jika rentang nilai *set point* yang diinginkan 30 °C hingga 85°C, pengguna bisa memilih mode panas.

Sistem menggunakan dua mikrokontroler, yaitu Arduino Mega sebagai pengendali utama dan ESP32 sebagai pengendali AC dimmer. Pada mode panas, Arduino Mega membaca suhu air menggunakan sensor DS18B20. Nilai suhu yang diperoleh dibandingkan dengan nilai *set point*, kemudian diolah menggunakan metode *Proportional Integral Derivative* (PID) untuk menghasilkan nilai PWM. Selama suhu air belum mencapai *set point*, pembacaan suhu dan perhitungan PID dilakukan secara berulang. Nilai PWM yang dihasilkan selanjutnya dikirim dari Arduino Mega ke ESP32 melalui komunikasi serial. Berdasarkan nilai tersebut, ESP32 mengatur besar daya yang diberikan ke *heater* melalui AC dimmer. Ketika suhu air telah mencapai *set point*, buzzer akan berbunyi sebagai tanda bahwa air panas sudah siap.

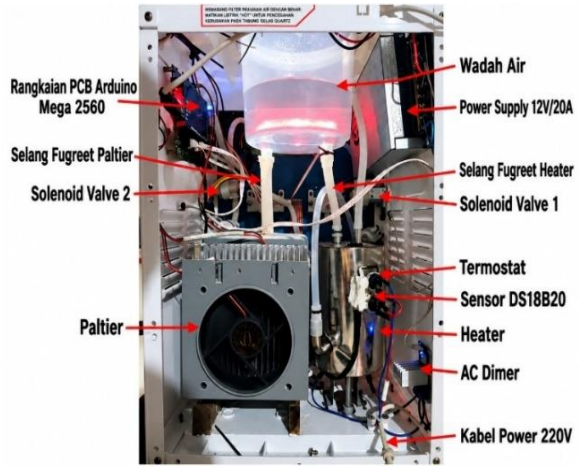
Pada mode dingin, mikrokontroler membaca suhu air menggunakan sensor NTC. Nilai suhu yang terbaca kemudian diproses menggunakan metode *Proportional Integral Derivative* (PID) untuk menghasilkan nilai keluaran berupa nilai PWM untuk mengendalikan daya pada Peltier. Selama suhu air belum mencapai nilai *set point*, sensor NTC akan terus membaca suhu dan proses kontrol PID dilakukan secara berulang sehingga daya pendinginan dapat disesuaikan dengan kondisi suhu air. Ketika suhu telah mencapai nilai *set point*, buzzer akan berbunyi sebagai tanda bahwa air dingin sudah siap.

Setelah buzzer berbunyi, pengguna memasukkan volume air yang diinginkan melalui keypad. Selanjutnya, sistem memeriksa keberadaan gelas menggunakan sensor ultrasonic HC-SR04. Apabila gelas belum terdeteksi, sistem akan tetap menunggu hingga gelas diletakkan pada posisi pengisian. Setelah gelas terdeteksi, sistem akan mengaktifkan relay yang mengendalikan pompa dan solenoid *valve* sesuai dengan mode yang dipilih. Pada mode dingin, relay mengaktifkan pompa dingin dan solenoid *valve* dingin, sedangkan pada mode panas, relay mengaktifkan pompa panas dan solenoid *valve* panas.

Ketika pompa dan solenoid *valve* aktif, air akan mengalir menuju gelas. Selama proses pengisian berlangsung, *flow* sensor mengukur volume air yang telah keluar. Apabila volume air belum mencapai nilai yang ditentukan, pompa dan solenoid *valve* tetap aktif sehingga proses pengisian terus berlangsung. Setelah *volume* air tercapai, pompa dan solenoid *valve* akan mati sehingga aliran air berhenti secara minum. Selanjutnya, sistem kembali ke menu utama dan siap menerima perintah berikutnya.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Perancangan Perangkat Keras (Hardware)



Gambar 4. Dispenser Tampak Dalam



Gambar 5. Dispenser Tampak Luar

Desain mekanik *box* dispenser pada penelitian ini merupakan hasil modifikasi dari dispenser merek Denpoo yang digunakan sebagai rangka utama sistem. Memiliki ukuran panjang 30 cm, lebar 24 cm, dan tinggi 42 cm. *Box* ini berfungsi sebagai tempat seluruh komponen elektronik, sensor, aktuator sehingga alat terlihat lebih rapi serta komponen terlindungi dari gangguan luar. LCD dipasang pada bagian atas panel dengan ukuran panjang 10 cm dan lebar 4 cm dan layar LCD oled dengan ukuran Panjang 7 cm dan lebar 3 cm. Di bawah LCD terdapat keypad berukuran 7 cm × 7 cm. Penempatan keypad di bawah LCD bertujuan agar proses pengaturan dapat dilakukan dengan lebih mudah karena pengguna dapat langsung melihat perubahan data pada layar.

Sensor ultrasonik dipasang pada bagian bawah panel depan. Sensor ini terdiri dari dua transduser yang ditempatkan pada dua lubang berdiameter 1,5 cm. Bagian dalam box digunakan untuk menempatkan komponen utama seperti mikrokontroller, modul relay, driver MOSFET, modul AC dimmer, sensor suhu, pompa air, dan lainnya. Ruang yang tersedia cukup untuk penataan komponen dan jalur kabel sehingga proses perakitan maupun perawatan alat dapat dilakukan dengan lebih mudah.

3.2 Hasil Pengujian Sensor DS18B20

Pengujian sensor DS18B20 dilakukan melalui perbandingan antara data hasil keluaran sensor terhadap alat ukur termometer sebagai referensi. Pengujian ini dilaksanakan sebanyak 10 kali pengambilan sampel untuk mendapatkan variasi data yang variatif. Pengujian ini bertujuan untuk menentukan tingkat akurasi dari sensor DS18B20. Proses pengujian sensor DS18B20 dengan termometer ditunjukkan pada tabel 1 berikut ini.

Tabel 1. Pengujian sensor DS18B20

Pengujian ke-	**Pembacaan Termometer**	**Pembacaan DS18B20**	***Error* (%)**
1	51,6	50,4	2,33
2	35,3	33,9	3,97
3	37,6	36,9	1,86
4	38,1	39,4	3,41
5	40,2	40,9	1,74
6	47,9	44,4	7,31
7	43,3	43,4	0,23
8	38,4	40,4	5,21
9	44,5	45,4	2,02
10	51,5	47,9	6,99
		Rata-rata *error*	3,51

Berdasarkan hasil pengujian yang dilakukan, sensor DS18B20 memiliki nilai error *sebesar* 3,51 % dengan akurasi sebesar 96,49 %.



Gambar 6. Posisi sensor DS18B20

Hasil pembacaan suhu oleh sensor DS18B20 tidak selalu sama dengan suhu air yang diukur menggunakan termometer pada keluaran keran. Perbedaan ini disebabkan oleh posisi pemasangan sensor DS18B20 yang ditempel pada permukaan luar tabung pemanas, sehingga sensor tidak bersentuhan langsung dengan air. Akibatnya, suhu yang terbaca merupakan suhu dinding tabung yang dipengaruhi oleh proses perpindahan panas dari air ke material tabung. Selain itu, material tabung memiliki hambatan termal sehingga perpindahan panas memerlukan waktu. Kondisi tersebut menyebabkan pembacaan DS18B20 mengalami keterlambatan dibandingkan perubahan suhu air di dalam tabung.

Sementara itu, termometer digunakan untuk mengukur suhu air yang keluar dari keran, sehingga nilai yang diperoleh merupakan suhu aktual air setelah mengalir keluar dari sistem. Selama air mengalir, suhu juga dapat mengalami perubahan akibat pelepasan panas ke lingkungan dan kontak dengan saluran air sebelum mencapai termometer. Oleh karena itu, nilai suhu yang ditunjukkan DS18B20 dan termometer dapat berbeda meskipun keduanya digunakan untuk mengukur kondisi yang sama. Untuk mengurangi perbedaan tersebut, dilakukan kalibrasi dengan memberikan nilai *offset* pada pembacaan DS18B20.

3.3 Hasil Pengujian Sensor NTC

Pengujian sensor NTC dilakukan melalui perbandingan antara data hasil keluaran sensor terhadap alat ukur termometer sebagai referensi. Pengujian ini dilaksanakan sebanyak 10 kali pengambilan sampel untuk mendapatkan variasi data yang variatif. Pengujian ini bertujuan untuk menentukan tingkat akurasi dari sensor. Proses pengujian sensor NTC dengan termometer ditunjukkan pada tabel 2 berikut ini.

Tabel 2. Pengujian Sensor NTC

Pengujian ke-	Pembacaan Termometer	Pembacaan NTC	Error (%)
1	23,2	23,1	0,43
2	23,5	24,0	2,13
3	24,3	25,2	3,70
4	24,8	25,6	3,23
5	29,3	30,1	2,73
6	22,5	22,8	1,33
7	25,1	25,6	1,99
8	23,9	24,3	1,67
9	22,9	22,4	2,18
10	23,5	23,6	0,43
		Rata-rata *error*	1,98%

Berdasarkan hasil pengujian yang dilakukan, sensor NTC memiliki nilai error *sebesar* 1,98 dengan akurasi sebesar 98,02 %.

3.4 Hasil Pengujian Sensor *Water Flow* YF-S401 Air Panas

Pengujian sensor *water flow* dilakukan melalui perbandingan antara data hasil keluaran sensor terhadap gelas ukur sebagai referensi. Pengujian ini dilaksanakan sebanyak 10 kali pengambilan sampel untuk mendapatkan variasi data yang variatif. Pengujian ini bertujuan untuk menentukan tingkat akurasi dari sensor. Proses pengujian sensor *water flow* dengan gelas ukur ditunjukkan pada tabel 3 berikut ini.

Tabel 3. Pengujian Sensor *Water Flow*

Pengujian ke-	Pembacaan *Water flow*	Pembacaan gelas ukur	Error (%)
1.	80	75	6,67%
2.	100	100	0,00%
3.	145	150	3,33%
4.	200	200	0,00%
5.	240	250	4,00%
6.	300	300	0,00%
7.	345	350	1,43%
8.	400	400	0,00%
9.	440	450	2,22%
10.	490	500	2,00%
		Rata-rata *error*	1,97 %

Berdasarkan hasil pengujian yang dilakukan, sensor *water flow* memiliki nilai *error sebesar* 1,97 % dengan akurasi sebesar 98,03 %.

3.5 Hasil Pengujian Sensor *Water Flow* YF-S401 Air Dingin

Pengujian sensor *water flow* dilakukan melalui perbandingan antara data hasil keluaran sensor terhadap gelas ukur sebagai referensi. Pengujian ini dilaksanakan sebanyak 10 kali pengambilan sampel untuk mendapatkan variasi data yang variatif. Pengujian ini bertujuan untuk menentukan tingkat akurasi dari sensor. Proses pengujian sensor *water flow* dengan gelas ukur ditunjukkan pada tabel 4 berikut ini.

Tabel 4. Pengujian Sensor *Water Flow*

Pengujian ke-	Pembacaan *Water flow*	Pembacaan gelas ukur	Error (%)
1.	75	75	0,00%
2.	105	100	5,00%
3.	140	150	6,67%
4.	200	200	0,00%
5.	235	250	6,00%
6.	300	300	0,00%
7.	340	350	2,86%
8.	405	400	1,25%
9.	440	450	2,22%
10.	500	500	0,00%
		Rata-rata *error*	2,40 %

Berdasarkan hasil pengujian yang dilakukan, sensor *water flow* memiliki nilai *error sebesar* 2,40 % dengan akurasi sebesar 97,06 %.

3.6 Hasil Pengujian PID Sistem Pemanas

3.6.1 Hasil Pengujian Sistem Pemanas Sebelum Menggunakan PID *Gain Scheduling*

Tabel 5. Hasil Pengujian Sistem Pemanas Sebelum Menggunakan PID

No	*Set point* (°C)	Suhu Awal (°C)	Suhu Max (°C)	Suhu Akhir (°C)	Waktu Mencapai SP (s)	PWM Rata-rata
1	40	35,5	44,5	40,00	205	14,4
2	48	41,5	52,0	51,65	206	46,0
3	55	50,0	58,5	58,50	185	53,3
4	65	59,5	69,0	68,52	152	53,3
5	72	67,0	76,5	75,98	243	51,0
6	72	68,5	76,0	75,98	182	53,2

Berdasarkan Tabel 5, sistem pemanas sebelum penerapan PID *Gain Scheduling* masih menunjukkan *overshoot* yang cukup besar pada beberapa *set point*. *Overshoot* tertinggi terjadi pada *set point* 40°C dan 72°C, yaitu sebesar 4,5°C. Hal ini menunjukkan bahwa sistem belum mampu menjaga suhu pada nilai target secara stabil dan sistem terlalu agresif sehingga menyebabkan *overshoot*.

3.6.2 Hasil Pengujian Sistem Pemanas Setelah Menggunakan PID *Gain Scheduling*

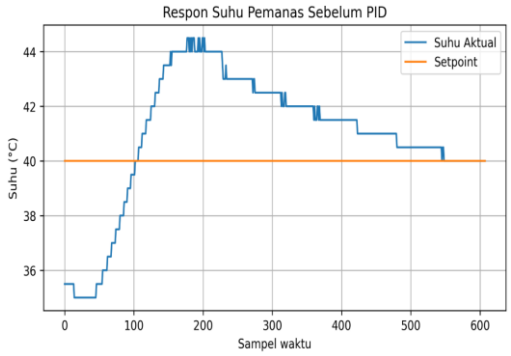
Tabel 6. Hasil Pengujian Sistem Pemanas Setelah PID

No	*Set point* (°C)	Suhu Awal (°C)	Suhu Max (°C)	Suhu Akhir (°C)	Waktu Mencapai SP (s)	PWM Rata-rata
1	70	59	70	70	427	49,6
2	75	64,5	75,5	75	332	53,3
3	39	32	39,5	39	210	42,4
4	50	39,5	51,5	50	191	28,6
5	45	36	46,5	45	258	34,2
6	55	43,5	55	55	393	48,5
7	58	48,5	58,5	58	300	46,6

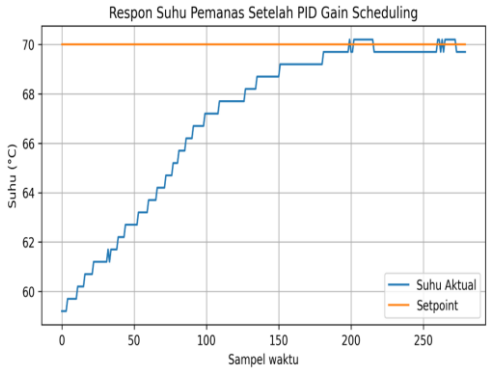
Berdasarkan Tabel 6, penerapan *PID Gain Scheduling* mampu menurunkan nilai *overshoot* pada hampir seluruh pengujian. *Overshoot* terbesar setelah *PID Gain Scheduling* adalah 1,5 °C, yaitu pada *set point* 50°C dan 45°C. Nilai tersebut lebih kecil dibandingkan pengujian sebelum *PID Gain Scheduling* yang menghasilkan *overshoot* hingga 4,5°C.

Pada *set point* 70 °C, suhu maksimum hanya mencapai 70 °C dengan *overshoot* sebesar 0,0 °C. Menunjukkan bahwa suhu aktual akhir sangat dekat dengan *set point*. Pada *set point* 75°C, sistem menghasilkan *overshoot* sebesar 0,5°C. Hal ini menunjukkan bahwa sistem tetap mampu mencapai suhu tinggi dengan kestabilan yang lebih baik.

3.6.3 Analisis Respon Sistem Pemanas Sebelum dan Setelah PID



Gambar 7. Respon Suhu Pemanas Sebelum PID



Gambar 8. Respon Suhu Pemanas Setelah PID

Berdasarkan Gambar 7, respon suhu pemanas sebelum penerapan PID menunjukkan bahwa kenaikan suhu belum dapat dikendalikan secara optimal. Kurva suhu aktual mengalami kenaikan melewati nilai *set point* sebelum akhirnya turun kembali. Hal ini menunjukkan adanya *overshoot* akibat energi panas yang masih tersimpan pada elemen pemanas dan media air. Pada kondisi tanpa pengaturan PID *gain scheduling*, nilai PWM *heater* tidak mempertimbangkan perubahan *error* suhu secara dinamis. Akibatnya, ketika suhu mendekati *set point*, daya pemanas masih terlalu besar sehingga suhu aktual melewati target.

3.6.4 Perbandingan Pengujian Sebelum dan Setelah *PID Gain Scheduling*

Untuk melihat peningkatan performa sistem, dilakukan perbandingan nilai rata-rata antara pengujian sebelum dan setelah penerapan *PID Gain Scheduling*.

Tabel 7. Perbandingan Sistem Pemanas

Parameter (*Set point* 55 ˚C)	**Sebelum *PID Gain Scheduling***	**Setelah *PID Gain Scheduling***	**Perubahan**
Overshoot	3,5 °C	0,0 °C	Turun 3,5 °C atau turun 100%
Rata-rata PWM	53,3	48,5	Turun 4,8 atau turun 9,34%
Waktu mencapai *set point*	185 s	393 s	Lebih lambat 208 s atau naik 52.92 %

Berdasarkan Tabel 7, penerapan *PID Gain Scheduling* mampu menurunkan *overshoot* dari 3,05 °C menjadi 0,0 °C. Penurunan ini menunjukkan bahwa sistem menjadi lebih stabil dan tidak terlalu melewati suhu target. Namun, waktu mencapai *set point* setelah penerapan PID menjadi lebih lama, yaitu dari 185 detik menjadi 393 detik. Kenaikan waktu ini terjadi karena sistem tidak lagi memberikan daya pemanas secara terlalu agresif. Dengan kata lain, sistem menjadi lebih lambat, tetapi lebih stabil.

Kondisi ini masih dapat diterima karena tujuan utama pengaturan suhu pada dispenser adalah menjaga suhu agar mendekati *set point*, bukan sekadar mencapai suhu target dengan cepat lalu melewatinya

3.6.5 Parameter PID *Gain Scheduling* Sistem Pemanas

Parameter PID yang digunakan pada sistem pemanas dibuat berubah berdasarkan rentang *set point* suhu. Tabel berikut menunjukkan nilai parameter PID yang diterapkan.

Tabel 8. Parameter PID Gain Scheduling Sistem Pemanas

Rentang *Set point* Suhu (°C)	**Kp**	**Ki**	**Kd**
30–39	12,0	0,018	6,0
40–49	11,0	0,015	7,5
50–59	10,0	0,012	9,0
60–69	9,5	0,010	10,0
70–79	8,5	0,008	12,0
80–90	7,0	0,005	15,0

Nilai (Kp) diturunkan pada suhu yang lebih tinggi agar respons pemanas tidak terlalu agresif. Nilai (Ki) juga diturunkan untuk mencegah akumulasi integral yang dapat menyebabkan pemanas tetap aktif meskipun suhu sudah mendekati *set point*. Sementara itu, nilai (Kd) dinaikkan pada *set point* tinggi agar sistem memiliki efek peredaman yang lebih besar terhadap perubahan suhu.

3.7 Hasil Pengujian PID Sistem Pendingin

3.7.1 Hasil Pengujian Sistem Pendingin Sebelum PID

Tabel 9. Hasil Pengujian Sistem Pendingin Sebelum PID

No	***Set point* (°C)**	**Suhu Awal (°C)**	**Suhu Min (°C)**	**Suhu Akhir Rata-rata (°C)**	**Waktu Men-pai SP (s)**	**PWM Rata-rata**
1	17	22,30	16,33	16,55	2814	157,70
2	23	27,63	22,30	23.1	1996	110,97
3	19	22,07	18,12	18,34	3928	81,13
4	20	24,25	19,98	20,03	1174	53,94

Pada tabel 9, sistem pendingin sebelum penerapan PID masih menunjukkan respons yang belum konsisten pada beberapa *set point*. *Under shoot* berada pada rentang 0,05–0,88 °C, Waktu mencapai *set point* juga cukup lama, yaitu 1174–3928 detik, dengan waktu terlama terjadi pada *set point* 19 °C. Selain itu, *error* akhir berada pada rentang 0,03–0,66 °C. Hal ini menunjukkan bahwa sistem pendingin belum mampu mencapai dan mempertahankan suhu pada *set point* secara konsisten. Perbedaan waktu pendinginan dan nilai *overshoot* pada setiap pengujian menunjukkan bahwa respons Peltier masih berubah-ubah sesuai kondisi suhu awal dan belum dapat mengatur daya pendinginan berdasarkan perubahan suhu secara otomatis.

3.7.2 Hasil Pengujian Sistem Pendingin Setelah PID

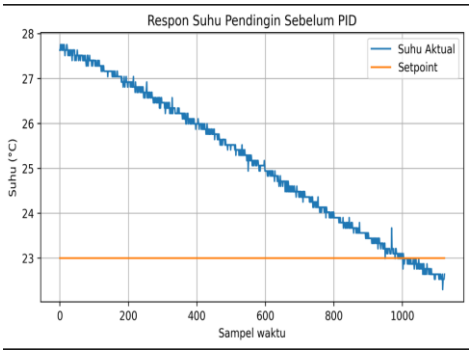
Tabel 10. Hasil Pengujian Sistem Pendingin Setelah PID

No	***Set point* (°C)**	**Suhu Awal (°C)**	**Suhu Min (°C)**	**Suhu Akhir Rata-rata (°C)**	**Waktu Men-pai SP (s)**	**PWM Rata-rata**

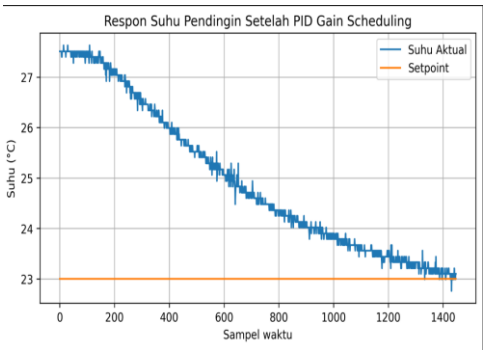
1	27	31,70	26,9	27,02	1838	130,33
2	23	27,51	22,76	23,10	2810	84,53
3	28	30,87	27,99	28,11	2121	50,78
4	25	26,69	24,71	25,06	1677	60,34

Berdasarkan Tabel 10, sistem pendingin setelah penerapan PID menunjukkan respons yang lebih terkontrol pada beberapa *set point*. Nilai *under shoot* berada pada rentang 0,01–0,29 °C, dengan nilai tertinggi pada *set point* 25 °C, yaitu sebesar 0,29 °C. *Error* akhir berada pada rentang 0,02–0,11 °C, sedangkan waktu mencapai *set point* berada pada rentang 1677–2810 detik. Hasil tersebut menunjukkan bahwa penerapan PID mampu mengurangi *under shoot* dan menjaga suhu akhir lebih mendekati *set point*. Meskipun waktu pendinginan masih cukup lama, pengaturan PWM dapat menyesuaikan proses pendinginan sehingga suhu lebih stabil dibandingkan sebelum penerapan PID.

3.7.3 Analisis Respon Pendingin Setelah PID



Gambar 9. Respon Suhu Pendingin Sebelum PID



Gambar 10. Respon Suhu Pendingin Setelah PID

Berdasarkan Gambar 9, setelah penerapan PID peltier, respon suhu menjadi lebih stabil. Kurva suhu aktual lebih mengikuti *set point* dan perubahan PWM peltier menjadi lebih sesuai terhadap besar *error* suhu.

3.7.4 Perbandingan Performa Sistem Pendingin Sebelum dan Setelah PID

Tabel 11. Perbandingan Rata-rata Kinerja Sistem Pendingin

Parameter (*Set point* 23 °C)	**Sebelum *PID Gain Scheduling***	**Setelah *PID Gain Scheduling***	**Perubahan**
Under shoot	0,70 °C	0,1 °C	Turun 0,6 °C atau turun 23,83%.
Rata-rata PWM	110,97	84,53	Naik 26,44 atau naik 23,82 %
waktu mencapai *set point*	1996 s	2810 s	Lebih lama 814 s atau 40,78%.

Error	0,1 °C	0,1 °C	Turun 0,0 °C atau turun 0 %

Berdasarkan hasil pengujian pada tabel 11, sistem pendingin sebelum penerapan PID memiliki *under shoot* suhu lebih besar. Hal ini terlihat dari *under shoot* yang masih cukup besar pada suhu rendah, terutama pada *set point* 20 °C. Setelah diterapkan metode PID, keluaran PWM peltier dapat berubah berdasarkan besar kecilnya *error* suhu. Ketika suhu aktual masih jauh dari *set point*, PID memberikan PWM yang lebih besar untuk mempercepat proses pendinginan. Ketika suhu mendekati *set point*, keluaran PWM dikurangi sehingga mencegah suhu turun terlalu rendah.

Perbedaan nilai rata-rata PWM antara pengujian sebelum dan sesudah penerapan PID *Gain Scheduling* dipengaruhi oleh kondisi awal pengujian yang tidak sama. Pada pengujian sebelum penerapan PID *Gain Scheduling*, suhu awal sebesar 22,30 °C sehingga sistem harus menaikkan suhu sebesar 0,70 °C untuk mencapai *set point* 23 °C. Sementara itu, pada pengujian setelah penerapan PID *Gain Scheduling*, suhu awal sebesar 22,76 °C sehingga kenaikan suhu yang diperlukan hanya 0,24 °C. Perbedaan besar *error* awal tersebut menyebabkan kebutuhan keluaran PWM pada kedua pengujian tidak dapat dibandingkan secara langsung.

Hasil pengujian menunjukkan bahwa penerapan PID mampu mengurangi *under shoot* pendinginan dari 0,70 °C menjadi 0,1°C. Dengan demikian, metode PID pada sistem pendingin mampu meningkatkan kestabilan suhu dan membuat peltier bekerja lebih sesuai dengan kebutuhan pendinginan.

3.7.5 Parameter PID *Gain Scheduling* Sistem Pendingin

Tabel 12. Parameter PID Sistem Pendingin

Rentang *Set point* (°C)	**(Kp)**	**(Ki)**	**(Kd)**
26-30	20,0	0,10	7,0
21–25	23,0	0,015	8,0
16–20	32,0	0,025	12,0

Pada tabel 12, peningkatan nilai Kp dan Ki pada suhu rendah dilakukan karena peltier membutuhkan daya lebih besar untuk mencapai suhu rendah. Pembacaan suhu NTC tidak selalu tepat atau sama dengan nilai *set point* karena nilai suhu diperoleh melalui proses perhitungan dari nilai ADC menggunakan persamaan steinhart dan penambahan nilai *offset* hasil kalibrasi. Oleh karena itu, hasil pembacaan dapat berupa nilai desimal, seperti 27,99 °C atau 28,11 °C**,** yang masih dianggap telah mencapai *set point* karena berada dalam batas toleransi pengukuran.

Rumus untuk pembacaan sensor NTC :

$$T\ (^{\circ}C) = \frac{1}{\frac{1}{T_0} + \frac{1}{B}\ln\left(\frac{R_{NTC}}{R_0}\right)} - 273.15 \qquad (1)$$

Dimana :
T = suhu dalam derajat Celcius (°C)
T_0= suhu referensi, yaitu 298,15 K (25 °C)
T_0 = konstanta Beta thermistor (3950 K)
R = resistansi thermistor pada suhu yang diukur (Ω)
R_0 = resistansi nominal thermistor pada suhu referensi (10 kΩ)
ln = logaritma natural

4. KESIMPULAN

Mengacu pada hasil perancangan dan pengujian yang telah dilakukan, dapat disimpulkan bahwa dispenser air minum yang dirancang mampu memberikan keleluasaan kepada pengguna untuk menentukan suhu air sesuai *set point* yang diinginkan pada rentang suhu yang telah ditetapkan. Selain pengaturan suhu, sistem juga mampu mengeluarkan air secara otomatis berdasarkan volume yang dipilih melalui keypad. Integrasi sensor, aktuator, dan mikrokontroler memungkinkan sistem menjalankan proses pengaturan suhu sekaligus pengeluaran air sesuai pilihan pengguna. Pada proses pengaturan

suhu, metode PID *Gain Scheduling* diterapkan pada sistem pemanas dan pendingin dengan menyesuaikan parameter kendali berdasarkan rentang suhu. Penyesuaian parameter Kp, Ki, dan Kd tersebut memungkinkan pengendali menyesuaikan diri dengan karakteristik sistem pada setiap kondisi operasi, sehingga suhu air dapat mencapai *set point* dan dipertahankan di sekitar nilai tersebut dengan respon yang lebih stabil pada proses pemanasan maupun pendinginan.

Berdasarkan hasil pengujian, penerapan PID *Gain Scheduling* menunjukkan karakteristik respon sistem yang lebih baik dibandingkan sistem sebelum menerapkan PID. Perbaikan tersebut terlihat dari penurunan *overshoot*, *steady-state error* yang lebih kecil, serta suhu yang lebih stabil di sekitar *set point*. Selain itu, keluaran PWM yang berubah secara bertahap membuat proses pengendalian suhu berlangsung lebih halus dibandingkan sistem tanpa PID dan kontrol *ON-OFF*. Dengan demikian, metode PID *Gain Scheduling* mampu meningkatkan kestabilan pengendalian suhu sehingga dispenser dapat menghasilkan suhu air yang lebih sesuai dengan *set point*. Sistem juga mampu mengeluarkan volume air sesuai dengan volume yang dipilih pengguna.

5. SARAN

Berdasarkan hasil perancangan dan pengujian yang telah dilakukan, terdapat beberapa hal yang dapat dikembangkan lebih lanjut pada sistem. Pada sistem pendingin, pengembangan diperlukan agar waktu pencapaian suhu sesuai *set point* dapat dipersingkat. Hal tersebut dapat dilakukan dengan mempertimbangkan penggunaan modul pendingin dengan kapasitas yang lebih besar atau menerapkan metode pendinginan alternatif untuk meningkatkan kinerja sistem. Selain itu, kapasitas tangki air panas maupun air dingin dapat ditingkatkan agar dispenser mampu melayani penggunaan secara berurutan dalam jumlah yang lebih banyak tanpa perlu menunggu proses pemanasan atau pendinginan ulang. Pengembangan selanjutnya juga dapat dilakukan pada sisi aplikasi *smartphone* dengan menambahkan fitur notifikasi ketika suhu telah mencapai *set point* serta pemantauan kondisi dispenser secara lebih menyeluruh, sehingga informasi mengenai kondisi sistem dapat diperoleh pengguna dengan lebih mudah dan praktis.

6. DAFTAR PUSTAKA

[1] D.A.P. Wardhana *et al.*, "Fast Heating Water Dispenser Dengan 5 Pilihan Suhu Air Berbasis Arduino," *Mechonversio: Mechanical Engineering Journal*, vol. 7, no. 1, pp. 15–20, 2024.

[2] F. Fatiatun *et al.*, "Penerapan Termodinamika Heating Dan Colling Pada Dispenser," *Jurnal Penelitian dan Pengabdian Kepada Masyarakat UNSIQ*, vol. 9, no. 2, pp. 146–150.

[3] F. Vauzia, "Pengaplikasian Kendali Suhu Peltier Dengan Menggunakan Kendali PID Berbasis Arduino Uno" *Jurnal Elektro dan Telekomunikasi Terapan*, vol. 9, no. 2, p. 1250, Dec. 2022.

[4] V. F. Rahmadini, A. Ma'arif, and N. S. Abu, "Design of Water Heater Temperature Control System using PID Control," *Control Systems and Optimization Letters*, vol. 1, no. 2, pp. 111–117, Aug. 2023.

[5] A. Anggraini, M. Farsha Ferial, R. Anugrahwaty, "Rancang Bangun Prototype Dispenser Untuk Inovasi Bisnis Minuman Es Teh Berbasis Telegram", *Konferensi Nasional Social dan Engineering Politeknik Negeri Medan*, 2024.

[6] M. Fikrul, M. ' Arif, W. Aribowo, M. Widyartono, and A.C. Hermawan, "Rancang Bangun Kendali PID pada Sistem Penerangan Berbasis Node-red", *Jurnal Teknik Elektro*, vol. 13, no. 1, 2024.

[7] A. Shafi Mukhaitir and I. Setiawan, "Aplikasi Kendali PID Menggunakan Skema Gain Scheduling Untuk Pengendalian Suhu Cairan pada Plant Electric Water Heater," *Transmisi*, vol. 12, no.1, 2010.

[8] J. Sistem and K. Tgd, "Implementasi Pembuka Jembatan Otomatis Menggunakan Teknik Pulse Width Modulation (PWM)", *Jurnal sistem Komputer TGD*, vol. 2, no. 1, pp. 53–59, 2023.

[9] C. Yohana Windra, "Penerapan Mikrokontroller Arduino Mega 2560 sebagai Monitoring pada Pembacaan Arus 3 Phasa di Gardu Induk 150 kV Lubuk Alung," *Jurnal Teknik Elektro ITP*, vol. 10, no. 1, p. 2021.

[10] M. Nizam, H. Yuana, and Z. Wulansari, "Mikrokontroler Esp 32 Sebagai Alat Monitoring Pintu Berbasis Web," *Jurnal Mahasiswa Teknik Informatika*, Vol. 6, No. 2, 2022.

[11] I. Novianto, L. Kurniasari, A. O. Pristisahida, B. K. Prasaja, dan Amanda, "Implementasi Filter Kalman untuk Optimasi Pengukuran Sensor Suhu NTC pada Kompor Listrik Malam Berbasis Fuzzy," *Jurnal Darma Agung*, vol. 30, no. 3, pp. 132–143, 2022.

[12] M. Bagus, R. Huda, and W. D. Kurniawan, "Analisa Sistem Pengendalian Temperatur Menggunakan Sensor DS18B20 Berbasis Mikrokontroler Arduino", *JRM*, Vol. 07, No. 02, 2022.

[13] A. Arfiansyah, P. Wanarti Rusimamto, and L. Anifah, "Penggunaan Sensor *Water flow* YF-S401 Pada Sistem Pencampur Parrfum Otomatis Berbasis Android 160 Penggunaan Sensor *Water flow* YF-S401 Pada Sistem Pencampur Parfum Otomatis Berbasis Android", Jurnal Teknik Elektro, Vol. 13, No. 2, 2024.

[14] M. I. Ralda, L. Ode, A. Barata, N. Endriatno, and K. Kunci, "Dispenser Otomatis Berbasis Arduino Uno", *PISTON: Jurnal Teknologi*, Vol. 07, No. 02, 2022.

[15] A. Aldi, "Rancang Bangun Sistem Kendali Intensitas Cahaya Lampu Ruangan Menggunakan Fuzzy Logic Berbasis Mikrokontroler Arduino Mega," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 1, Jan. 2024.