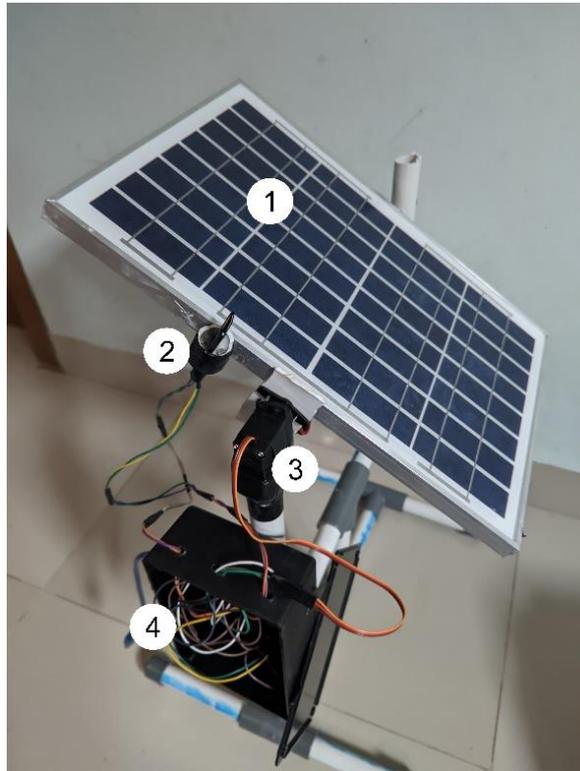


BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Perangkat Keras

Perancangan perangkat keras melibatkan pemilihan komponen yang sesuai berdasarkan spesifikasi. Beberapa komponen tersebut kemudian dirancang membentuk perangkat *single-axis solar tracker*. Berikut adalah hasil perancangan sistem dapat dilihat pada gambar 4.1.



Gambar 4.1 Perangkat Solar Tracker

Keterangan :

1. Pada nomor 1 merupakan panel surya atau *photovoltaic* yang pada sistem ini digunakan untuk menyerap energi yang dihasilkan oleh matahari kemudian dikonversi menjadi energi listrik.

2. Pada nomor 2 merupakan sensor *Light Dependent Resistor* (LDR) yang digunakan untuk mendeteksi intensitas cahaya matahari, dimana pada sistem ini adalah cahaya yang dihasilkan oleh matahari.
3. Pada nomor 3 merupakan motor servo yang digunakan untuk menggerakkan panel surya sehingga sesuai dengan posisi sudut matahari secara *realtime*.
4. Pada nomor 4 merupakan arduino UNO yang digunakan sebagai *microcontroller* untuk menjalankan perintah yang diberikan sehingga *solar tracker* dapat berjalan sesuai dengan logikanya.

4.2. Perancangan Program

Dalam perancangan program perlu memodelkan sistem pelacak surya satu sumbu untuk memahami karakteristik dan dinamika pergerakan pelacak surya. Pemodelan ini melibatkan persamaan-persamaan yang menggambarkan hubungan antara variabel-variabel yang relevan, seperti sudut posisi pelacak surya, intensitas cahaya matahari, dan posisi matahari. Program menggunakan *software* arduino IDE dengan bahasa C yang dapat langsung diupload langsung ke board arduino UNO menggunakan kabel data usb tipe *a to b*. Berikut adalah sub-sub program.

a. *Setup* Variabel

```

14 void setup() {
15     pinMode(9, OUTPUT);
16     pinMode(A0, INPUT);
17     pinMode(A1, INPUT);
18     Serial.begin(9600);

```

Gambar 4.2 Program *setup* variabel

Pada *void setup()* berisi inisialisasi pin pada sensor dan aktuator. Selain itu menetapkan *baudrate* untuk pada serial monitor.

- b. Nilai awal servo dan rumus nilai *error*.

31	MV_1 = 90;
32	error = ldr2 - ldr1;

Gambar 4.3 Program nilai awal sero dan rumus nilai *error*

Nilai awal servo yaitu pada sudut 90° dan rumus mencari nilai *error* didapatkan dari hasil pengurangan kedua nilai ldr.

- c. Implementasi rumus PID

42	currentTime = millis();
43	elapsedTime = (double)(currentTime - previousTime);
44	integral = integral + error * elapsedTime;
45	float derivative = (error - previousError) / elapsedTime;
46	PID = (Kp * error) + (Ki * integral) + (Kd * derivative);

Gambar 4.4 Program rumus PID

Sub program ini hasil dari implementasi rumus PID pada sub bab 2.3.

- d. Implementasi rumus PI

31	currentTime = millis();
32	elapsedTime = (double)(currentTime - previousTime);
33	integral = integral + error * elapsedTime;
34	Pi = (Kp*error)+(Ki*integral);

Gambar 4.5 Program rumus PI

Sub program ini hasil dari implementasi rumus PI pada sub bab 2.3.

- e. *Output*

55	void solar_on(){
56	MV = Pi+MV_1;
57	if(MV>=125){
58	MV = 125;
59	}else if(MV<=55){
60	MV = 55;
61	}else{
62	MV = MV;
63	}
64	motor.write(MV);
65	}

Gambar 4.6 *Output* program

Output dari program adalah hasil perhitungan matematis dari kendali PID dan PI berupa sinyal PWM yang di berikan pada motor servo dengan maksimal pada nilai 125 dan minimal pada nilai 55.

4.3. Tuning Kendali

Tuning dalam konteks kendali merujuk pada proses mengatur parameter-parameter dari kontroler (seperti P, I, dan D pada kontrol PID) agar sistem kontrol dapat berkinerja optimal. Tujuan dari tuning adalah untuk mencapai respon sistem yang diinginkan, dengan waktu respons yang cepat, kestabilan, dan minimal overshoot.

Tuning merupakan proses yang penting karena setiap sistem memiliki karakteristik yang berbeda, dan parameter kendali yang tidak sesuai dapat menghasilkan respon yang tidak diinginkan. Oleh karena itu, tuning melibatkan penyesuaian parameter-parameter kendali untuk mencapai keseimbangan yang tepat antara respon sistem yang cepat dan stabilitas.

Proses tuning dilakukan dengan metode trial and error. Metode *trial and error* adalah pendekatan yang melibatkan percobaan berulang dengan variasi parameter atau strategi untuk mencapai hasil yang diinginkan. Dalam konteks sistem kendali, metode *trial and error* digunakan untuk menentukan nilai optimal dari parameter-parameter kendali (Jayadi *et al.*, 2021).

4.3.1. Tuning Kendali PI

Pada kendali ini terdapat dua parameter yang akan dicari nilainya yaitu konstanta *proportional* (K_p) dan konstanta *integral* (K_i). *percobaan dilakukan menggunakan perangkat lunak Arduino IDE pada fitur serial monitor. Hasil dari*

percobaan berulang-ulang menghasilkan nilai $K_p = 0.5$ dan $K_i = 0.003$. berikut data output yang telah dihasilkan dari nilai K_p dan K_i tersebut.

```

Output Serial Monitor X
Not connected. Select a board and a port to connect automatically.
19:42:45.988 -> Ldr 1 : 35.28 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.054 -> Ldr 1 : 35.28 - Ldr 2 : 35.48 - Output : 101.18 - Error : 0
19:42:46.128 -> Ldr 1 : 35.28 - Ldr 2 : 35.48 - Output : 101.18 - Error : 0
19:42:46.193 -> Ldr 1 : 35.28 - Ldr 2 : 35.48 - Output : 101.18 - Error : 0
19:42:46.258 -> Ldr 1 : 35.28 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.324 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.390 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.455 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.521 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.586 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.652 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.685 -> Ldr 1 : 35.28 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.750 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.815 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.883 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:46.946 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.012 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.078 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.143 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.209 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.274 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.340 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.405 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.471 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.536 -> Ldr 1 : 35.08 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.602 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.667 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.700 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.765 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.832 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0
19:42:47.897 -> Ldr 1 : 34.89 - Ldr 2 : 35.67 - Output : 101.18 - Error : 0

```

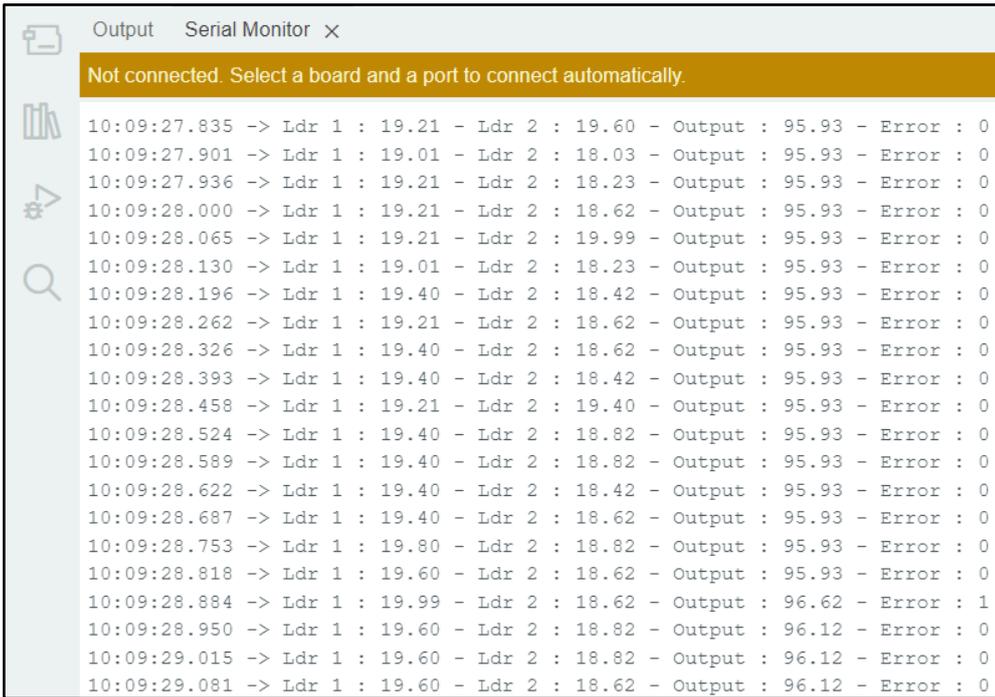
Gambar 4.7 Output tuning kendali PI

Hasil menampilkan nilai waktu, nilai intensitas cahaya pada ldr 1, ldr 2, nilai output dan nilai error. Kestabilan sistem dapat dilihat pada nilai error yang menampilkan nilai : 0. Jika nilai error adalah 0, itu berarti bahwa output dari sistem atau proses yang sedang dikendalikan sudah mencapai nilai yang diinginkan atau *setpoint*. Dalam konteks kontrol, error adalah perbedaan antara *setpoint* dan nilai aktual atau output sistem. Ketika nilai error adalah 0, ini menunjukkan bahwa

sistem telah mencapai keadaan stabil di mana *setpoint* dan *output* sistem adalah sama.

4.3.2. Tuning Kendali PID

Pada kendali ini terdapat dua parameter yang akan dicari nilainya yaitu konstanta *proportional* (K_p), *konstanta integral* (K_i) dan *Konstanta Derivative* (K_d). percobaan dilakukan menggunakan perangkat lunak Arduino IDE pada fitur *serial monitor*. Hasil dari percobaan berulang-ulang menghasilkan nilai $K_p = 0.5$, $K_i = 0.003$ dan $K_d = 0.1$. berikut data *output* yang telah dihasilkan dari nilai K_p , K_i dan K_d tersebut.



The screenshot shows the Serial Monitor window in Arduino IDE. The title bar reads "Output Serial Monitor x". A yellow banner at the top says "Not connected. Select a board and a port to connect automatically." Below this, a list of data points is displayed, each representing a timestamp and the values for Ldr 1, Ldr 2, Output, and Error. The Error column consistently shows 0, indicating a stable system.

Timestamp	Ldr 1	Ldr 2	Output	Error
10:09:27.835	19.21	19.60	95.93	0
10:09:27.901	19.01	18.03	95.93	0
10:09:27.936	19.21	18.23	95.93	0
10:09:28.000	19.21	18.62	95.93	0
10:09:28.065	19.21	19.99	95.93	0
10:09:28.130	19.01	18.23	95.93	0
10:09:28.196	19.40	18.42	95.93	0
10:09:28.262	19.21	18.62	95.93	0
10:09:28.326	19.40	18.62	95.93	0
10:09:28.393	19.40	18.42	95.93	0
10:09:28.458	19.21	19.40	95.93	0
10:09:28.524	19.40	18.82	95.93	0
10:09:28.589	19.40	18.82	95.93	0
10:09:28.622	19.40	18.42	95.93	0
10:09:28.687	19.40	18.62	95.93	0
10:09:28.753	19.80	18.82	95.93	0
10:09:28.818	19.60	18.62	95.93	0
10:09:28.884	19.99	18.62	96.62	1
10:09:28.950	19.60	18.82	96.12	0
10:09:29.015	19.60	18.82	96.12	0
10:09:29.081	19.60	18.62	96.12	0

Gambar 4.8 Output tuning kendali PID

Hasil menampilkan nilai waktu, nilai intensitas cahaya pada ldr 1, ldr 2, nilai *output* dan nilai *error*. Kestabilan sistem dapat dilihat pada nilai *error* yang menampilkan nilai : 0. Jika nilai *error* adalah 0, itu berarti bahwa *output* dari sistem

atau proses yang sedang dikendalikan sudah mencapai nilai yang diinginkan atau *setpoint*. Dalam konteks kontrol, *error* adalah perbedaan antara *setpoint* dan nilai aktual atau output sistem. Ketika nilai error adalah 0, ini menunjukkan bahwa sistem telah mencapai keadaan stabil di mana *setpoint* dan *output* sistem adalah sama.

4.4. Komparasi Respon Sistem

Pada tahap ini dilakukan perbandingan antara metode kendali PI (*Proportional-Integral*) dan PID (*Proportional-Integral-Derivative*) pada sistem pelacak surya satu sumbu. Analisis dilakukan dengan membandingkan kinerja kedua metode kendali berdasarkan respon sistem yang dihasilkan kendali dengan parameter yaitu *Rise Time*, *Settling Time*, *SettlingMin*, *SettlingMax*, *Overshoot*, *Undershoot*, *Peak* dan *PeakTime*.

4.4.1. Respon Sistem Kendali PI

Dalam analisis sistem kendali, respon sistem sangat penting untuk mengevaluasi kinerja sistem dan memastikan bahwa sistem beroperasi sesuai dengan yang diharapkan. Dengan memahami respon sistem, dapat dilakukan identifikasi apakah sistem responsif, stabil, memiliki tingkat osilasi yang dapat diterima, dan mencapai nilai *setpoint* dengan cepat dan akurat.

Diperlukan beberapa nilai *ouput* kendali untuk melihat hasil atau reaksi yang diberikan oleh sistem terhadap perubahan input atau stimulus yang diterima. Proses dilakukan secara langsung dengan memposisikan servo pada sudut 90° dari *setpoint* sebagai titik sudut 0° kemudian sistem akan diaktifkan sehingga *solar tracker* bergerak menuju *setpoint*. dalam proses tersebut menghasilkan 15 data dalam 1

detik, digunakan 75 data *output* untuk menentukan respon sistem sehingga diperlukan waktu 6 detik dalam proses pengambilan data. Berikut adalah sedikit sampel data yang akan digunakan dalam mencari respon sistem dapat dilihat pada tabel 4.1.

Tabel 4.1 Sampel data uji respon PI

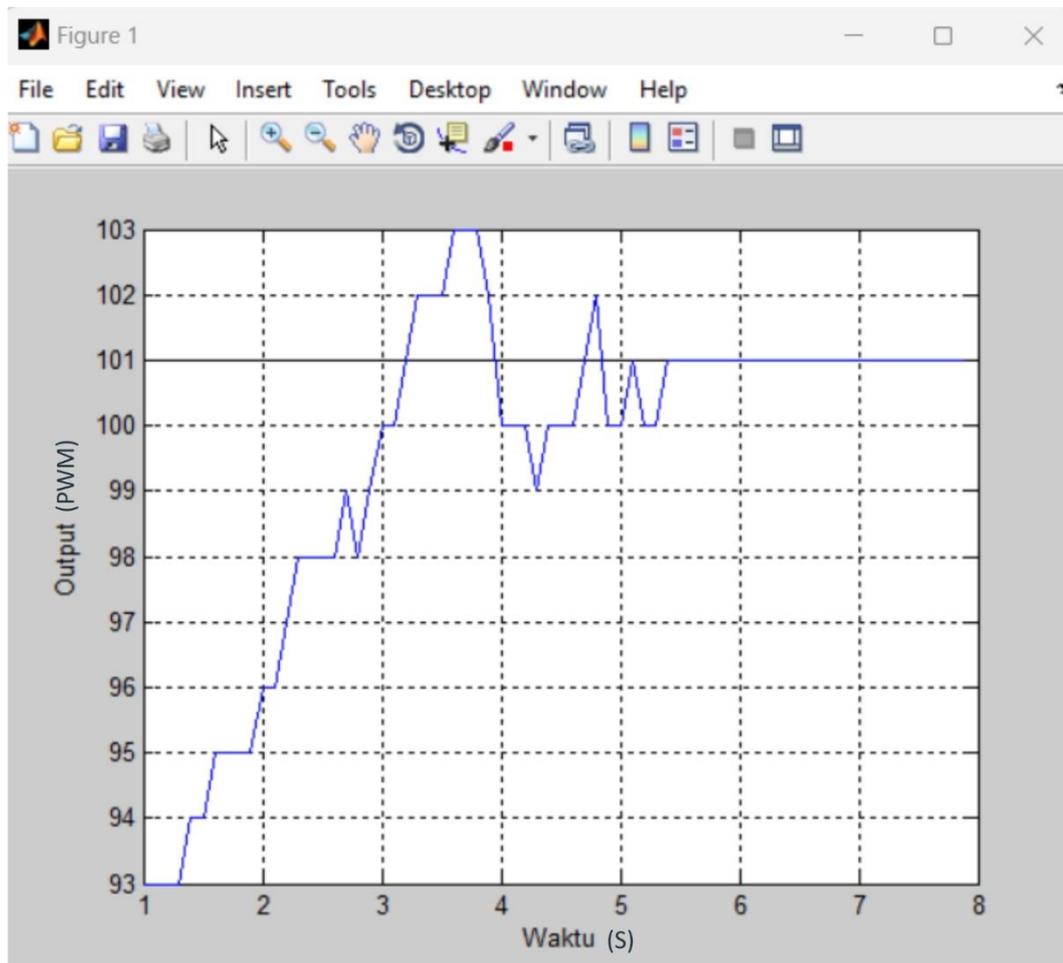
Waktu	Nilai LDR 1	Nilai LDR 2	Output	Error
10:09:28 AM	13.92	7.06	93.00	6
10:09:28 AM	13.92	7.06	93.02	6
10:09:28 PM	13.92	7.25	93.05	6
10:09:28 PM	13.92	7.25	93.25	6
10:09:28 PM	13.92	7.25	94.21	6
10:09:28 PM	13.92	7.06	93.72	6
10:09:28 PM	13.92	7.06	94.71	6
10:09:28 PM	13.92	7.06	95.18	6
10:09:28 PM	11.37	5.88	95.10	5
10:09:28 PM	11.56	6.66	94.92	4
10:09:28 PM	12.15	6.08	96.41	6
10:09:28 PM	11.76	6.86	95.73	4
10:09:28 PM	12.15	6.27	96.64	5
10:09:28 PM	14.11	7.45	97.62	6
10:09:28 PM	12.54	6.86	97.53	5
10:09:29 AM	12.74	7.06	97.93	5
10:09:29 AM	12.94	7.64	98.34	5
10:09:29 AM	13.33	7.84	98.74	5
10:09:29 AM	13.52	10.39	97.99	3
10:09:29 AM	13.72	8.43	99.41	5
10:09:29 AM	14.11	8.43	99.83	5
10:09:29 AM	13.92	8.82	100.22	5
10:09:29 AM	14.31	8.62	100.64	5
10:09:29 AM	14.9	8.43	101.66	6
10:09:29 AM	14.11	8.62	101.56	5
10:09:29 AM	14.11	8.62	101.56	5
10:09:29 AM	15.09	10.39	102.63	4

Keterangan :

- Waktu (s) : Kolom tersebut mencatat waktu dalam satuan detik ketika data diambil. Waktu menunjukkan kapan pengukuran dilakukan, yang dapat membantu dalam analisis terhadap perubahan yang terjadi dalam data selama interval waktu tertentu.
- Nilai LDR1 dan LDR2 : Kolom tersebut mencatat nilai yang diterima dari sensor cahaya LDR1 dan LDR2. Nilai-nilai ini akan berfluktuasi berdasarkan intensitas cahaya yang terdeteksi oleh sensor-sensor tersebut. Data dari sensor LDR ini dapat memberikan wawasan tentang tingkat cahaya di lingkungan sekitar saat pengukuran.
- Output PWM : Kolom tersebut merekam nilai keluaran dari sinyal PWM. PWM (*Pulse Width Modulation*) adalah metode pengaturan sinyal analog dengan cara mengubah lebar pulsa sinyal digital periodik [21]. Output PWM akan bervariasi sesuai dengan parameter pengendalian yang telah ditetapkan dalam sistem.
- Nilai Error : Kolom tersebut mencatat nilai error antara nilai referensi yang diinginkan dan nilai keluaran aktual dari sistem. Error dihitung dengan mengurangi nilai referensi dari nilai keluaran. Nilai error ini menunjukkan seberapa besar perbedaan antara nilai yang diharapkan dan nilai yang diperoleh dalam sistem kendali.

Nilai-nilai yang telah didapatkan tersebut kemudian dimasukkan kedalam *software* MatLab untuk dicari respon sistem kendali PI. Respon sistem pada analisis ini berupa perubahan grafik yang menggambarkan bagaimana sistem bereaksi terhadap

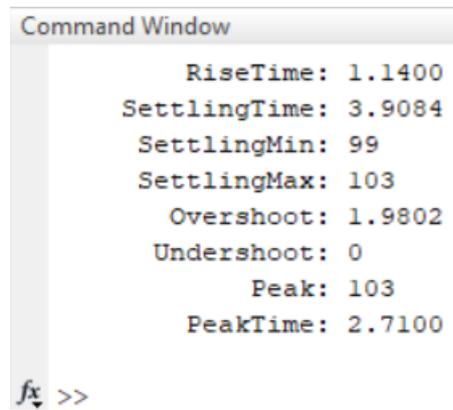
input yang diberikan. Berikut adalah hasil grafik dari respon sistem dapat dilihat pada gambar 4.8.



Gambar 4.9 Grafik Respon Sistem PI

Tahap ini menggunakan Grafik *Transient Response* atau grafik yang menggambarkan bagaimana sistem berperilaku dalam periode transisi setelah terjadi perubahan input. Ini termasuk respons awal, *overshoot*, *undershoot*, *settling time*, dan tingkat osilasi. Pada grafik tersebut menampilkan sumbu X sebagai nilai *output* dengan satuan PWM dan sumbu Y sebagai satuan waktu detik. Nilai gangguan yang diberikan pada sistem memberikan nilai awal pada titik 93 kemudian setelah gangguan dihilangkan nilai terus naik menuju *setpoint* yaitu 101

dan stabil pada nilai tersebut. Berikut adalah keterangan dari grafik dapat dilihat pada gambar 4.9 sebagai berikut.



Gambar 4.10 Keterangan Respon Sistem PI

Sistem membutuhkan waktu 1.1400 detik untuk bergerak dari nilai 93 menuju 101 kemudian bergerak melewati *setpoint* sehingga menghasilkan *overshoot* 1.9802 dan kembali menuju *setpoint*. Setelah nilai mulai stabil pada *setpoint* terdapat sedikit osilasi terjadi sehingga menghasilkan *SettlingMin* 99 dan *SettlingMax* 103. *Peak* pada uji ini yaitu 103 dengan waktu tempuh dari nilai awal menuju *Peak* yaitu 2.7100 detik. *SettlingTime* atau waktu tempuh dari nilai awal menuju stabil yaitu 3.9084 detik.

4.2.3. Respon Sistem Kendali PID

Berikut adalah sedikit sampel data yang akan digunakan dalam mencari respon sistem dapat dilihat pada tabel 4.2.

Tabel 4.2 Sampel data uji respon PID

Waktu	Nilai Ldr 1	Nilai Ldr 2	Output	Error
10:30:28 AM	7.84	10.58	90.66	-2
10:30:28 AM	8.04	11.17	88.01	-3
10:30:28 AM	8.04	10.98	88.15	-2

Waktu	Nilai LDR 1	Nilai LDR 2	Output	Error
10:30:28 AM	6.66	9.02	87.77	-2
10:30:28 AM	6.27	9.6	86.72	-3
10:30:28 AM	5.88	10	85.48	-4
10:30:28 AM	6.08	11.17	84.05	-5
10:30:28 AM	5.68	11.96	82.41	-6
10:30:28 AM	9.02	10.78	84.74	-1
10:30:28 AM	9.41	10.98	84.54	-1
10:30:28 AM	9.41	10.98	84.54	-1
10:30:28 AM	10.98	12.15	87.16	-1
10:30:28 AM	13.92	10.98	89.07	2
10:30:28 AM	14.7	11.17	90.13	3
10:30:28 AM	15.29	12.15	90.69	3
10:30:29 AM	15.68	12.74	90.56	2
10:30:29 AM	16.27	13.52	90.94	2
10:30:29 AM	16.66	13.33	92.00	3
10:30:29 AM	16.86	15.09	91.19	1
10:30:29 AM	16.66	12.15	93.44	4
10:30:29 AM	17.05	11.17	94.88	5
10:30:29 AM	17.25	14.31	93.75	2
10:30:29 AM	17.64	15.09	94.13	2
10:30:29 AM	17.84	15.88	93.81	1
10:30:29 AM	18.03	16.66	94.00	1
10:30:29 AM	18.03	17.05	93.50	0
10:30:29 AM	18.23	17.05	94.19	1
10:30:29 AM	16.66	12.15	93.44	4
10:30:29 AM	17.05	11.17	94.88	5
10:30:29 AM	17.25	14.31	93.75	2

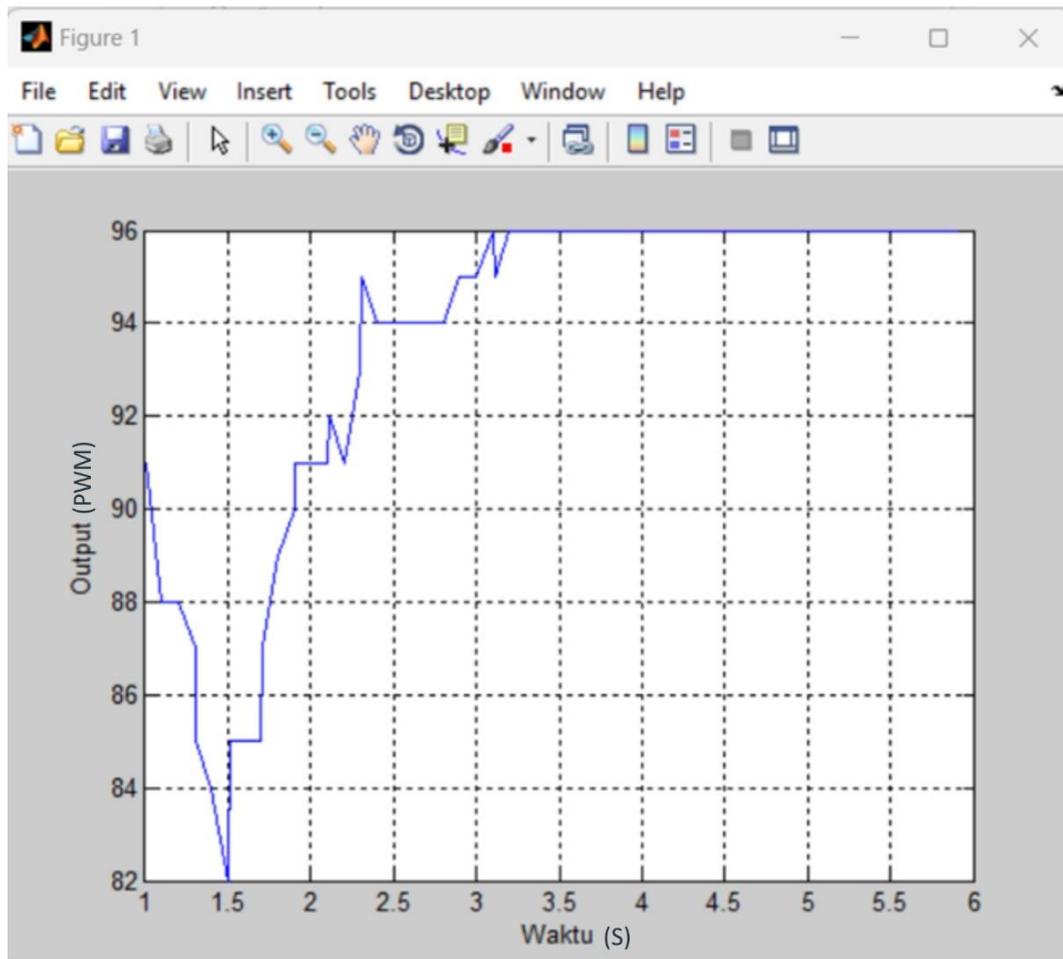
Keterangan :

- Waktu (s) : Kolom tersebut mencatat waktu dalam satuan detik ketika data diambil. Waktu menunjukkan kapan pengukuran dilakukan, yang dapat membantu dalam analisis terhadap perubahan yang terjadi dalam data selama interval waktu tertentu.
- Nilai LDR1 dan LDR2 : Kolom tersebut mencatat nilai yang diterima dari sensor cahaya LDR1 dan LDR2. Nilai-nilai ini akan berfluktuasi

berdasarkan intensitas cahaya yang terdeteksi oleh sensor-sensor tersebut. Data dari sensor LDR ini dapat memberikan wawasan tentang tingkat cahaya di lingkungan sekitar saat pengukuran.

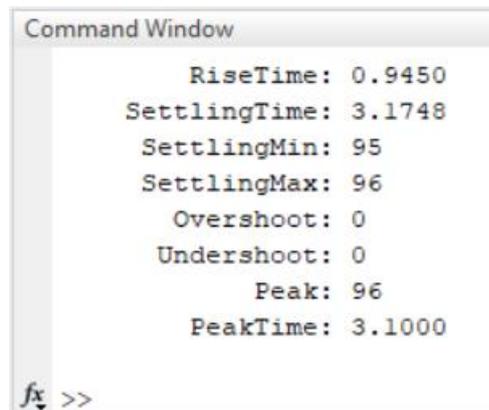
- Output PWM : Kolom tersebut merekam nilai keluaran dari sinyal PWM. PWM (*Pulse Width Modulation*) adalah metode pengaturan sinyal analog dengan cara mengubah lebar pulsa sinyal digital periodik [21]. Output PWM akan bervariasi sesuai dengan parameter pengendalian yang telah ditetapkan dalam sistem.
- Nilai Error : Kolom tersebut mencatat nilai error antara nilai referensi yang diinginkan dan nilai keluaran aktual dari sistem. Error dihitung dengan mengurangi nilai referensi dari nilai keluaran. Nilai error ini menunjukkan seberapa besar perbedaan antara nilai yang diharapkan dan nilai yang diperoleh dalam sistem kendali.

Nilai-nilai yang telah didapatkan tersebut kemudian dimasukkan kedalam *software* MatLab untuk dicari respon sistem kendali PI. Respon sistem pada analisis ini berupa perubahan grafik yang menggambarkan bagaimana sistem bereaksi terhadap *input* yang diberikan. Berikut adalah hasil grafik dari respon sistem dapat dilihat pada gambar 4.11.



Gambar 4.11 Grafik respon PID

Tahap ini menggunakan Grafik *Transient Response* atau grafik yang menggambarkan bagaimana sistem berperilaku dalam periode transisi setelah terjadi perubahan input. Ini termasuk respons awal, *overshoot*, *undershoot*, *settling time*, dan tingkat osilasi. Pada grafik tersebut menampilkan sumbu X sebagai nilai *output* dengan satuan PWM dan sumbu Y sebagai waktu dengan satuan detik. Nilai gangguan yang diberikan pada sistem memberikan nilai awal pada titik 82 kemudian setelah gangguan dihilangkan nilai terus naik menuju *setpoint* yaitu 96 dan stabil pada nilai tersebut. Berikut adalah keterangan dari grafik dapat dilihat pada gambar 4.12 sebagai berikut.



Gambar 4.12 Keterangan respon PID

Sistem membutuhkan waktu 0.9450 detik untuk bergerak dari nilai 85 menuju 96 kemudian bergerak menuju *setpoint* dan stabil pada *setpoint* tanpa terjadi osilasi sehingga menghasilkan *overshoot* dan *undershoot* 0. Sistem menghasilkan nilai *SettlingMin* 95 dan *SettlingMax* 103. *Peak* pada uji ini yaitu 96 dengan waktu tempuh dari nilai awal menuju *Peak* yaitu 3.1000 detik. *SettlingTime* atau waktu tempuh dari nilai awal menuju stabil yaitu 3.1000 detik.

4.2.4. Hasil Komparasi Respon Sistem

Berdasarkan hasil uji respon sistem kendali pada sistem *single-axis solar tracker* dengan menggunakan kendali PID dan kendali PI, dapat diidentifikasi beberapa hal sebagai berikut:

Tabel 4.3 Hasil Komparasi Respon Sistem

Parameter	PID	PI
RiseTime	0.9450	1.1400
SettlingTime	3.1748	3.9084
SettlingMin	95	99
SettlingMax	96	103

Parameter	PID	PI
OverShoot	0	1.9802
UnderShoot	0	0
Peak	96	103
PeakTime	3.1000	2.7100

1. Kendali PI :

- Sistem dengan kendali PI membutuhkan waktu sekitar 1.1400 detik untuk bergerak dari nilai 93 menuju 101.
- Setelah mencapai nilai 101, sistem mengalami overshoot sebesar 1.9802 sebelum kembali menuju setpoint dan mulai stabil.
- Setelah mencapai setpoint, terdapat sedikit osilasi yang menyebabkan SettlingMin sebesar 99 dan SettlingMax sebesar 103.
- Peak pada uji ini terjadi pada nilai 103, dengan waktu tempuh dari nilai awal menuju Peak adalah 2.7100 detik.

2. Kendali PID :

- Sistem dengan kendali PID membutuhkan waktu sekitar 0.9450 detik untuk bergerak dari nilai 85 menuju 96.
- Setelah mencapai nilai 96, sistem bergerak menuju setpoint dan stabil pada setpoint tanpa terjadi osilasi, sehingga menghasilkan overshoot dan undershoot 0.
- SettlingMin pada uji ini adalah 95, dan SettlingMax adalah 103.
- Peak pada uji ini terjadi pada nilai 96, dengan waktu tempuh dari nilai awal menuju Peak adalah 3.1000 detik.

- Waktu tempuh dari nilai awal menuju stabil (*SettlingTime*) adalah 3.1000 detik.
- dari nilai awal menuju stabil (*SettlingTime*) adalah 3.9084 detik.

Hasil uji respon sistem kendali dengan menggunakan kendali PID dan kendali PI, kedua metode kendali tersebut menunjukkan kinerja yang memadai dalam mencapai *setpoint* dan mempertahankan kestabilan sistem. Meskipun terdapat perbedaan dalam waktu tempuh menuju *setpoint* dan nilai *overshoot* pada kedua metode, keduanya berhasil mencapai *SettlingMin* dan *SettlingMax* yang sesuai. Namun, metode kendali PID menunjukkan waktu tempuh menuju Peak yang lebih lama dibandingkan dengan kendali PI, yang dapat mengindikasikan kecepatan respons yang lebih rendah. Sebaliknya, kendali PI menunjukkan *overshoot* yang lebih besar dan sedikit osilasi setelah mencapai *setpoint*, menunjukkan sedikit ketidakstabilan dalam jangka waktu tertentu.

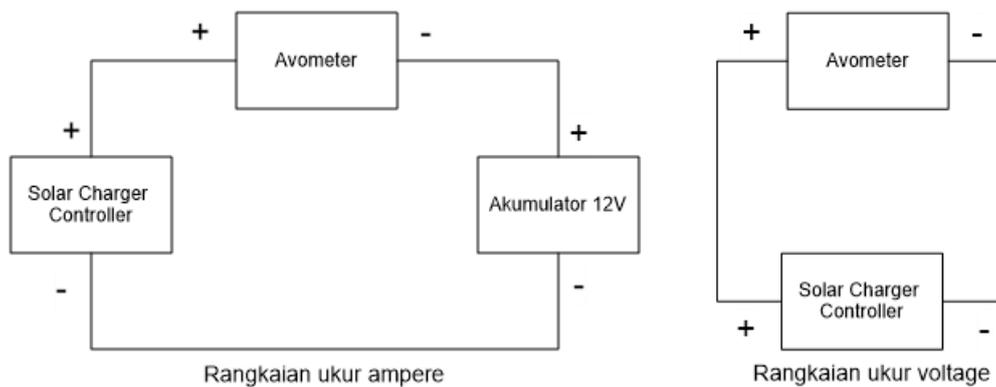
Oleh karena itu, pemilihan metode kendali antara PID dan PI dapat dipertimbangkan berdasarkan prioritas tertentu, seperti kecepatan respons, tingkat *overshoot*, atau kestabilan sistem. Dalam hal ini, keduanya telah menunjukkan hasil yang memadai.

4.4. Komparasi Daya

Pada tahap ini bertujuan untuk melakukan komparasi daya yang dihasilkan panel surya dengan tujuan menganalisis efisiensi, kinerja, dan keandalan dari kedua sistem. Komparasi daya dilakukan selama 2 hari dengan pengukuran dan pengamatan langsung, dimana pada hari pertama mengamati dan mengukur daya *open circuit* yang masuk kedalam kedua panel surya dengan parameter *voltage*,

ampere dan *watt* kemudian pada hari kedua secara *close circuit* dengan melakukan *charging* pada akumulator 12V7Ah. Pada proses *charging*, akan diukur dan dibandingkan berapa daya yang dihasilkan oleh solar panel 10WP dengan parameter *voltage*, *ampere* dan *watt* selama 6 jam kemudian menghitung total dari *watt*. Adapun nilai *watt* diambil dari perhitungan matematis dengan menggunakan $Watt = Ampere \times Voltage$

Sedangkan nilai *ampere* diambil dari hasil pengukuran langsung menggunakan *avometer* dan *voltage* menggunakan *solar charge controller* dengan rangkaian sebagai berikut.



Gambar 4.13 Rangkaian ukur *ampere* dan *voltage*

Komparasi dimulai pada tanggal 26 juli 2023 – 27 juli 2023. Sebelum melakukan pengisian akumulator, maka terlebih dahulu akumulator dikosongkan atau *battery discharger*. Berikut adalah hasil data yang telah dilakukan pengukuran.

1. 26 Juli 2023

Tabel 4.4 Data daya *open circuit* 25 juli 2023

Waktu	PID			PI		
	<i>Watt</i>	<i>Voltage</i>	<i>Ampere</i>	<i>Watt</i>	<i>Voltage</i>	<i>Ampere</i>
9.00	6.77	17.35	0.39	6.77	17.35	0.39
9.30	8.14	18.49	0.44	8.14	18.49	0.44

Waktu	PID			PI		
	Watt	Voltage	Ampere	Watt	Voltage	Ampere
10.00	9.91	18.69	0.53	9.91	18.69	0.53
10.30	11.19	18.97	0.59	11.19	18.97	0.59
11.00	13.64	19.21	0.71	13.64	19.21	0.71
11.30	15.10	19.11	0.79	15.10	19.11	0.79
12.00	13.98	18.89	0.74	13.98	18.89	0.74
12.30	12.89	18.16	0.71	12.89	18.16	0.71
13.00	12.59	18.52	0.68	12.59	18.52	0.68
13.30	10.69	18.43	0.58	10.69	18.43	0.58
14.00	7.80	18.57	0.42	7.80	18.57	0.42
<i>Average</i>	11.15	18.58	0.60	11.15	18.58	0.60

Pada hari pertama pengukuran daya *open circuit*, hasil pengukuran menunjukkan bahwa rata-rata nilai *voltage* yang dihasilkan oleh kedua solar tracker adalah sebesar 18.58V. Selain itu, *ampere* yang diukur pada kedua solar tracker memiliki rata-rata nilai sebesar 0.60A. Selanjutnya, daya (*watt*) yang dihasilkan oleh kedua *solar tracker* memiliki rata-rata nilai sebesar 11.15W. Data ini mencerminkan performa dan efisiensi sistem pelacak surya pada hari pertama uji, dengan rata-rata nilai *voltage*, *ampere*, dan *watt* yang stabil dan konsisten pada kedua *solar tracker*.

2. 26 Juli 2023

Tabel 4.5 Data pengujian daya 26 juli 2023

Waktu	PID			PI		
	Watt	Voltage	Ampere	Watt	Voltage	Ampere
9.00	3.84	12.80	0.30	3.84	12.80	0.30
9.30	4.00	12.89	0.31	4.00	12.89	0.31
10.00	4.55	13.00	0.35	4.55	13.00	0.35
10.30	5.20	13.00	0.40	5.20	13.00	0.40
11.00	5.20	13.00	0.40	5.20	13.00	0.40
11.30	5.20	13.00	0.40	5.20	13.00	0.40
12.00	5.20	13.00	0.40	5.20	13.00	0.40

Waktu	PID			PI		
	Watt	Voltage	Ampere	Watt	Voltage	Ampere
12.30	5.20	13.00	0.40	5.20	13.00	0.40
13.00	5.20	13.00	0.40	5.20	13.00	0.40
13.30	5.20	13.00	0.40	5.20	13.00	0.40
14.00	4.37	12.85	0.34	4.37	12.85	0.34
14.30	3.86	12.45	0.31	3.86	12.45	0.31
15.00	3.74	12.45	0.30	3.74	12.45	0.30
<i>Average</i>	4.67	12.88	0.36	4.67	12.88	0.36

Pada hari kedua, hasil pengukuran menunjukkan bahwa rata-rata nilai *voltage* yang diinputkan ke akumulator adalah sebesar 12.88V. Selain itu, *ampere* yang dihasilkan pada kedua *solar tracker* memiliki rata-rata nilai sebesar 0.36A, dan daya (*watt*) yang dihasilkan oleh kedua *solar tracker* memiliki rata-rata nilai sebesar 12.35W. Selanjutnya hasil dari pengukuran selama 6 jam dilakukan perhitungan total daya yang dihasilkan dengan rumus energi listrik sebagai berikut.

$$Total\ Daya = Voltage \times Ampere \times Time$$

Diketahui rata-rata nilai *voltage* 12.96V, *ampere* 0.37A dan *time* 6 jam. ketika dimasukan kedalam rumus, maka sebagai berikut.

$$27.8208 = 12.88 \times 0.36 \times 6$$

Sehingga kedua panel surya menghasilkan total daya pada akumulator 12V sebesar 27.8208W selama 6 jam. Data ini mencerminkan performa dan efisiensi sistem pelacak surya pada hari pertama uji, dengan rata-rata nilai *voltage*, *ampere*, dan *watt* yang stabil dan konsisten pada kedua tracker.

4.4.1. Hasil Komparasi Daya

Berdasarkan pengujian yang telah dilakukan berikut adalah beberapa kesimpulan hasil komparasi yang dapat diambil:

1. *Power (Watt)*: Hasil komparasi menunjukkan bahwa kedua *solar tracker* dengan kendali PI dan PID menghasilkan daya *watt* yang sama pada nilai *setpoint* yang ditetapkan. Artinya, keduanya mampu mencapai daya *watt* yang diinginkan pada tingkat kinerja yang setara.
2. *Current (Ampere)*: Hasil komparasi menunjukkan bahwa kedua *solar tracker* menunjukkan arus yang sama ketika berada pada nilai *setpoint* yang ditentukan. Hal ini menandakan bahwa kedua metode kendali PI dan PID memiliki kemampuan yang setara dalam mengatur arus pada tingkat yang diinginkan.
3. *Voltage (Volt)*: Hasil komparasi menunjukkan bahwa kedua *solar tracker* mencapai *voltage* yang sama saat mencapai nilai *setpoint*. Ini menandakan bahwa keduanya memiliki kemampuan yang setara dalam mempertahankan *voltage* pada tingkat yang diinginkan.
4. Berdasarkan hasil pengamatan pada kedua *solar tracker* pada proses pengukuran tidak ada perbedaan perubahan nilai V dan A yang terjadi secara signifikan seiring perubahan sudut matahari.

sehingga dari hasil komparasi tersebut, dapat disimpulkan bahwa kedua metode kendali PI dan PID memberikan hasil daya *watt*, *ampere*, *voltage*, dan waktu durasi pengisian yang setara dalam sistem *single-axis solar tracker* pada nilai *setpoint* yang ditentukan. Hal ini menunjukkan bahwa keduanya memiliki kinerja yang sebanding dalam mengoptimalkan daya dan efisiensi sistem *solar tracker*.