

LEMBAR PERSETUJUAN TUGAS AKHIR

PERANCANGAN DAN IMPLEMENTASI PENGATURAN TEMPERATUR
TANGKI AIR MENGGUNAKAN *HEATER* DAN *PELTIER* DENGAN
METODE PID BERBASIS *OPENPLC* ESP32

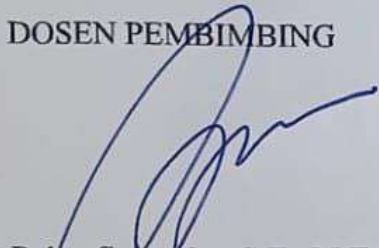
Diajukan oleh :

Muhammad Haikal Ali Akbar Darul Haq

40040322650065

TELAH DISETUJUI DAN DITERIMA DENGAN BAIK OLEH

DOSEN PEMBIMBING



Priyo Sasmoko, S.T., M.Eng.

NIP. 197009161998021001

Tanggal 23 September 2026

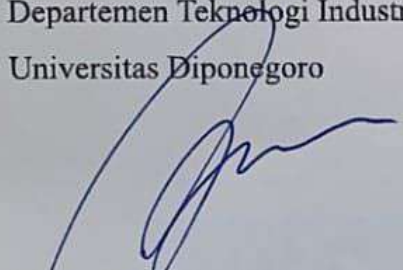
Mengetahui,

Ketua

Program Studi S.Tr Teknologi Rekayasa Otomasi

Departemen Teknologi Industri Sekolah Vokasi

Universitas Diponegoro



Priyo Sasmoko, S.T., M.Eng.

NIP. 197009161998021001

Tanggal 23 September 2026

LEMBAR PENGESAHAN TUGAS AKHIR

PERANCANGAN DAN IMPLEMENTASI PENGATURAN TEMPERATUR
TANGKI AIR MENGGUNAKAN *HEATER* DAN *PELTIER* DENGAN
METODE PID BERBASIS *OPENPLC* ESP32

Diajukan oleh :

Muhammad Haikal Ali Akbar Darul Haq
40040322650065

Telah diujikan dan dinyatakan lulus oleh Tim Penguji

2026

Tim Penguji.

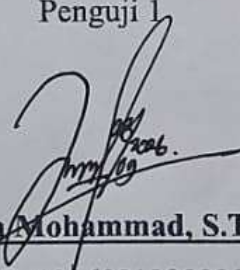
Dosen Pembimbing



Priyo Sasmoko, S.T., M.Eng.

NIP. 197009161998021001

Penguji 1



Luthfansyah Mohammad, S.Tr.T., M.T.

NPPU H.7. 199609132022041001

Penguji 2



Rofiq Cahyo Prayogo, S.T., M.T.

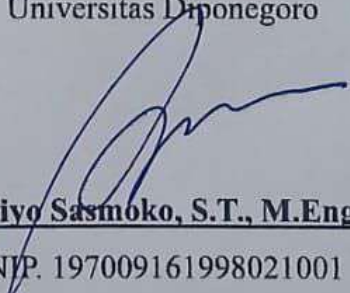
NIP. 199505292024061001

Mengetahui

Ketua Program Studi S.Tr Teknologi Rekayasa Otomasi

Departemen Teknologi Industri Sekolah Vokasi

Universitas Diponegoro



Priyo Sasmoko, S.T., M.Eng.

NIP. 197009161998021001

SURAT PERNYATAAN BEBAS PLAGIAT

Saya yang bertanda tangan di bawah ini:

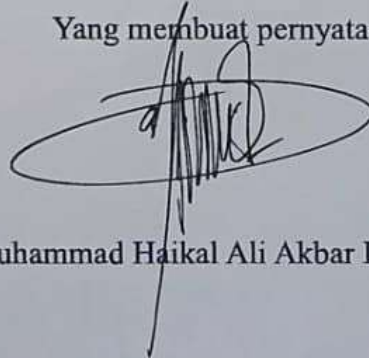
Nama : Muhammad Haikal Ali Akbar Darul Haq
NIM : 40040322650065
Program Studi : Teknologi Rekayasa Otomasi
Judul Tugas Akhir : Perancangan Dan Implementasi Pengaturan
Temperatur Tangki Air Menggunakan *Heater*
Dan *Peltier* Dengan Metode PID Berbasis
Openplc ESP32

Dengan ini menyatakan bahwa dalam tugas akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh derajat keahlian di suatu perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dirujuk dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila dikemudian hari terbukti plagiat dalam tugas akhir ini, maka saya bersedia menerima sanksi sesuai peraturan Mendiknas RI No.17 tahun 2010 dan peraturan perundang-undangan yang berlaku.

Semarang, 23 September 2026

Yang membuat pernyataan,



Muhammad Haikal Ali Akbar Darul Haq

KATA PENGANTAR

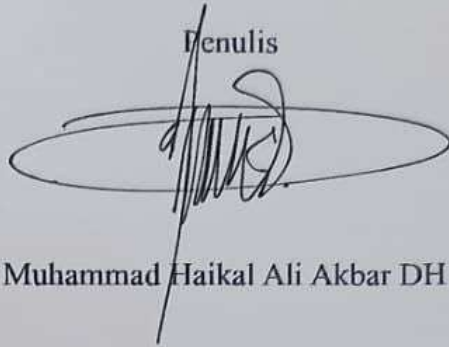
Puji syukur atas kehadiran Allah SWT telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul “Perancangan Dan Implementasi Pengaturan Temperatur Tangki Air Menggunakan *Heater Dan Peltier* Dengan Metode PID Berbasis *OpenPLC* ESP32” Tersusunnya laporan ini tentu tidak lepas dari bantuan, dukungan, dan bimbingan dari berbagai pihak. Penulis menyampaikan terima kasih kepada :

1. Allah SWT yang senantiasa memberikan kesehatan, kelancaran, kemudahan serta kesabaran bagi penulis dalam setiap langkah penyusunan Tugas Akhir sehingga dapat menyelesaikan penelitian ini dengan lancar.
2. Ibu Sri Sukmawati, Ayah Muhammad Nadirudin Darul Haq dan keluarga yang selalu meberikan dukungan doa dari jauh, motivasi dan nasehat kepada penulis untuk menyelesaikan Tugas Akhir ini.
3. Prof Dr. Ir. Budiyo, M. Si, selaku Dekan Sekolah Vokasi Universitas Diponegoro.
4. Priyo Sasmoko, S.T.,M.Eng., selaku Ketua Program Studi Teknologi Rekayasa Otomasi dan selaku dosen pembimbing saya yang telah memberikan arahan dan bimbingan selama proses penyusunan Tugas Akhir ini.
5. Rizal Nadhif Pratista dan Afdhol Arafat Lintang Pasha Pamungkas Setia Nusantara sebagai rekan satu kelompok dalam Tugas Akhir yang telah berjuang bersama sama dalam pengerjaan alat ini samapai berhasil.
6. Arvisha Putri Az-zahra selaku teman dekat dari penulis yang telah menemani, membantu dan senantiasa selalu memberikan dukungan selama penyusunan laporan Tugas Akhir ini sehingga bisa menyelesaikan dengan baik.
7. Teman teman kontrakan Ibu Saparmi dan Teknologi Rekayasa Otomasi Angkatan 2022 ”*SATNAYAKA*” yang sudah memberikan dukungan dan semangat serta perjuanganya dalam menyusun Laporan Tugas Akhir.

Penulis menyadari bahwa laporan Tugas Akhir ini masih kurang sempurna dan banyak kekurangan. Penulis berharap semoga laporan ini dapat memberikan manfaat bagi penulis khususnya dan bagi pembaca serta perkembangan ilmu pengetahuan pada umumnya di teknologo otomasi.

Semarang, 23 September 2026

Penulis

A handwritten signature in black ink, consisting of several loops and a long horizontal stroke, enclosed within a large, hand-drawn oval.

Muhammad Haikal Ali Akbar DH

DAFTAR ISI

LEMBAR PERSETUJUAN TUGAS AKHIR.....	ii
LEMBAR PENGESAHAN TUGAS AKHIR.....	iii
SURAT PERNYATAAN BEBAS PLAGIAT.....	iv
KATA PENGANTAR.....	v
DAFTAR ISI	vii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xvii
DAFTAR SINGKATAN DAN SIMBOL.....	xviii
INTISARI.....	xxii
ABSTRAK	xxiii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	4
1.3 Tujuan Tugas Akhir.....	4
1.4 Manfaat Tugas Akhir.....	5
1.5 Pembatasan Masalah	6
1.6 Sistematika Tugas Akhir	7
BAB II DASAR TEORI.....	8
2.1 Tinjauan Pustaka	8
2.2 Sistem Kontrol	12
2.2.1 Sistem Kontrol Loop Terbuka (<i>Open Loop</i>)	12
2.2.2 Sistem Kontrol Loop Tertutup (<i>Closed Loop</i>).....	13
2.3 Karakteristik Respons Sistem Kendali.....	13
2.3.1 <i>Rise Time</i>	14
2.3.2 <i>Overshoot</i>	14
2.3.3 <i>Settling time</i>	15
2.3.4 <i>Error Terhadap Setpoint</i>	16
2.4 OpenPLC (<i>Open Programmable Logic Controller</i>)	16
2.4.1 <i>OpenPLC Runtime</i>	17
2.4.2 <i>Ladder Diagram (LD)</i>	18

2.4.3	<i>Function Block Diagram (FBD)</i>	19
2.4.4	<i>Structured Text (ST)</i>	20
2.4.5	<i>Sequential Function Chart (SFC)</i>	21
2.5	<i>Function Block Yang Digunakan</i>	22
2.5.1	<i>Function Block PID CTRL</i>	22
2.5.2	<i>Function Block ADD</i>	23
2.5.3	<i>Function Block SUB</i>	24
2.5.4	<i>Function Block MUL</i>	24
2.5.5	<i>Function Block DIV</i>	24
2.5.6	<i>Function Block MOVE</i>	24
2.5.7	<i>Function Block Comparator</i>	25
2.5.8	<i>Function Block REAL_TO_UINT</i>	25
2.5.9	<i>Function Block UINT_TO_REAL</i>	25
2.5.10	<i>Function Block TON (On Delay Timer)</i>	26
2.6	<i>Arduino IDE</i>	26
2.7	<i>Pulse Width Modulation</i>	27
2.8	<i>Komunikasi Pin to pin</i>	28
2.8.1	<i>Komunikasi PWM Antar ESP32</i>	29
2.9	<i>Resistor Capacitor Filter (RC filter)</i>	29
2.9.1	<i>Konstanta Waktu RC</i>	30
2.9.2	<i>Frekuensi Cut-Off</i>	30
2.10	<i>Digital Low Pass Filter (Exponential Moving Average)</i>	31
2.11	<i>Metode PID (Proportional Integral Derivative)</i>	31
2.11.1	<i>Persamaan Umum PID</i>	32
2.11.2	<i>Metode Tuning Ziegler Nichols</i>	33
2.12	<i>Mikrokontroller ESP32</i>	37
2.13	<i>Heater</i>	38
2.14	<i>Persamaan Kebutuhan Daya Heater</i>	39
2.15	<i>Peltier TEC1-12706</i>	40
2.16	<i>PT100</i>	41
2.17	<i>Transmitter RTD PT100</i>	42
2.18	<i>Power Supply</i>	43
2.19	<i>Buck Converter</i>	44
2.20	<i>AC Light Dimmer Module</i>	46

BAB III METODE PENELITIAN.....	48
3.1 Blok Diagram Sistem	48
3.2 <i>Flowchart</i>	53
3.3 <i>Wiring</i> Elektrikal Keseluruhan.....	58
3.4 Gambar 3D Alat	60
3.5 Spesifikasi dan Fitur Alat	61
3.5.1 Spesifikasi	61
3.5.2 Fitur Alat	63
3.6 Cara Pembuatan / Fabrikasi	64
3.6.1 Perancangan Perangkat Lunak	65
3.6.2 Perancangan Perangkat Keras	84
BAB IV PENGUJIAN DAN ANALISIS.....	90
4.1 Pengujian Komunikasi <i>Pin to pin</i> Antar ESP32.....	90
4.1.1 Pengujian Pengiriman Nilai 0 pada <i>OpenPLC</i>	90
4.1.2 Pengujian Pengiriman Nilai 20000 pada <i>OpenPLC</i>	92
4.1.3 Pengujian Pengiriman Nilai 40000 pada <i>OpenPLC</i>	93
4.1.4 Pengujian Pengiriman Nilai 65535 pada <i>OpenPLC</i>	95
4.1.5 Analisis Keseluruhan Komunikasi <i>Pin to pin</i>	96
4.2 Pengujian Pembacaan Sensor PT100	98
4.2.1 Hasil Pembacaan PT100 pada Arduino IDE	99
4.2.2 Hasil Pembacaan Temperatur pada <i>OpenPLC</i>	100
4.2.3 Hasil Pembacaan Temperatur pada HMI	102
4.2.4 Perbandingan Pembacaan Sistem dengan Temperatur Aktual	104
4.3 Pengujian Sistem Pengaturan Daya <i>Heater</i>	105
4.3.1 Pengujian Daya <i>Heater</i> 10%.....	105
4.3.2 Pengujian Daya <i>Heater</i> 20%.....	106
4.3.3 Pengujian Daya <i>Heater</i> 30%.....	108
4.3.4 Pengujian Daya <i>Heater</i> 40%.....	109
4.3.5 Pengujian Daya <i>Heater</i> 50%.....	110
4.3.6 Pengujian Daya <i>Heater</i> 60%.....	111
4.3.7 Pengujian Daya <i>Heater</i> 70%.....	112
4.3.8 Pengujian Daya <i>Heater</i> 80%.....	114
4.3.9 Pengujian Daya <i>Heater</i> 90%.....	115
4.3.10 Pengujian Daya <i>Heater</i> 100%.....	116

4.3.11	Analisis Perbandingan Tegangan dan Arus Keluaran AC <i>Dimmer</i>	118
4.4	Pengujian Sistem Pendingin <i>Peltier</i>	120
4.4.1	Pengujian <i>Peltier</i> Penampungan	121
4.4.2	Pengujian <i>Peltier Final</i>	122
4.4.3	Pengujian Arus <i>Peltier</i> Saat ON	125
4.5	Implementasi Sistem Kontrol	126
4.6	Pengujian Karakteristik Sistem Pemanas	127
4.7	Pengujian Sistem Kontrol <i>Open Loop</i>	127
4.7.1	Pengujian <i>Open Loop</i> PWM 69% 4 Liter	128
4.7.2	Data Hasil Pengujian <i>Open Loop</i> Pemanasan Air 4 Liter	134
4.7.3	Grafik Hasil Percobaan	138
4.7.4	Pengujian <i>Open Loop Setpoint</i> 35°C dengan Input <i>Heater</i> 69%	140
4.7.5	Pengujian <i>Open Loop Setpoint</i> 45°C dengan Input <i>Heater</i> 69%	142
4.7.6	Pengujian <i>Open Loop Setpoint</i> 55°C dengan Input <i>Heater</i> 69%	144
4.7.7	Perbandingan Respons <i>Open Loop</i>	145
4.8	Pengujian Sistem Kontrol <i>Closed Loop</i>	146
4.8.1	Pengujian <i>Closed Loop Setpoint</i> 35 °C	146
4.8.2	Pengujian <i>Closed Loop Setpoint</i> 45 °C	154
4.8.3	Pengujian <i>Closed Loop Setpoint</i> 55 °C	161
4.8.4	Perbandingan Respons <i>Closed Loop</i>	168
4.8.5	Analisis Keseluruhan Pengujian <i>Open Loop</i>	169
4.9	Perbandingan <i>Open loop</i> dan <i>Closed Loop</i>	172
4.9.1	Perbandingan <i>Setpoint</i> 35°C	173
4.9.2	Perbandingan <i>Setpoint</i> 45°C	174
4.9.3	Perbandingan <i>Setpoint</i> 55°C	176
4.9.4	Analisis Perbandingan <i>Open Loop</i> dan <i>Closed Loop</i>	177
4.10	Evaluasi Implementasi dan Keterbatasan <i>OpenPLC</i> ESP32	179
BAB V PENUTUP		182
5.1	Kesimpulan	182
5.2	Saran	183
DAFTAR PUSTAKA		185
LAMPIRAN		188

DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka.....	8
Tabel 2.2 Penalaan Rumus.....	36
Tabel 2.3 Parameter dan Spesifikasi.....	38
Tabel 2.4 Parameter dan Spesifikasi.....	39
Tabel 2.5 Parameter dan Spesifikasi.....	41
Tabel 2.6 Parameter dan Spesifikasi.....	42
Tabel 2.7 Parameter dan Spesifikasi.....	43
Tabel 2.8 Parameter dan Spesifikasi.....	44
Tabel 2.9 Parameter dan Spesifikasi.....	45
Tabel 2.10 Parameter dan Spesifikasi.....	46
Tabel 2.11 Parameter dan Spesifikasi.....	47
Tabel 3.1 Pin Aktuator dan Sensor	60
Tabel 3.2 Spesifikasi Komponen dan Fungsi	61
Tabel 3.3 Komponen Sistem Pengontrolan	85
Tabel 4.1 Hasil Pengujian Komunikasi Pin to pin.....	97
Tabel 4.2 Hasil Pengukuran daya Heater.....	118
Tabel 4.3 Kondisi Pengujian.....	129
Tabel 4.4 Hasil Pengukuran.....	133
Tabel 4.5 Data hasil Pengujian PWM Heater 69%.....	134
Tabel 4.6 Hasil Percobaan Open Loop Setpoint 35°C	141
Tabel 4.7 Hasil Percobaan <i>Open Loop Setpoint</i> 45°C	143
Tabel 4.8 Hasil Percobaan Open Loop Setpoint 55°C.....	145
Tabel 4.9 Perbandingan Respon Open Loop	146
Tabel 4.10 Parameter Respon Closed Loop Setpoint 35 °C	151
Tabel 4.11 Parameter Respon Closed Loop Setpoint 45 °C	158
Tabel 4.12 Parameter Respon Closed Loop Setpoint 55 °C	165
Tabel 4.13 Perbandingan Pengujian Closed Loop.....	169
Tabel 4.14 Tabel Error Akhir Percobaan	170
Tabel 4.15 Perbandingan Akurasi.....	174
Tabel 4.16 Perbandingan Akurasi.....	175
Tabel 4.17 Perbandingan Akurasi.....	177

Tabel 4.18 Analisis Hasil Keseluruhan.....	178
Tabel 4.19 Error dan Akurasi Keseluruhan	178

DAFTAR GAMBAR

Gambar 2.1 Software OpenPLC [9]	17
Gambar 2.2 Ladder Diagram	19
Gambar 2.3 Function Block Diagram [14]	20
Gambar 2.4 Structured Text [15]	21
Gambar 2.5 Sequential Function Chart [17]	22
Gambar 2.6 Pulse Width Modulation [20]	27
Gambar 2.7 Proportional Integral Derivative [23]	32
Gambar 2.8 Mikrokontroller ESP32 [28]	37
Gambar 2.9 Heater [30]	38
Gambar 2.10 Peltier TEC1-12706 [32]	40
Gambar 2.11 Sensor PT100	41
Gambar 2.12 Transmitter RTD	43
Gambar 2.13 Power Supply	44
Gambar 2.14 Buck Converter 5V	45
Gambar 2.15 Buck Converter 12V	45
Gambar 2.16 AC Light Dimmer Module [38]	47
Gambar 3.1 Blok Diagram Sistem	48
Gambar 3.2 Flowchart Keseluruhan Sistem	53
Gambar 3.3 Flowchart Sitem Pengaturan Temperatur	56
Gambar 3.4 Wiring sistem	58
Gambar 3.5 Design 3D Alat	60
Gambar 3.6 Alur pin to pin	66
Gambar 3.7 Codingan perhitungan Arduino IDE	69
Gambar 3.8 Ladder Scalling PT100	70
Gambar 3.9 Function Block Ladder PID CTRL	72
Gambar 3.10 Scalling Pwm Out to Arduino IDE	74
Gambar 3.11 Perhitungan Pwm Masuk Dari OpenPLC	75
Gambar 3.12 Flow Peltier	77
Gambar 3.13 Blok diagram Close loop kendali PID	79
Gambar 3.14 Grafik Percobaan Open Loop	80

Gambar 3.15 Scematic PCB Komponen	86
Gambar 3.16 Layout PCB Jalur Atas Komponen.....	87
Gambar 3.17 Layout PCB Jalur bawah komponen	87
Gambar 3.18 Diameter Tangki Final	88
Gambar 3.19 Diameter Tangki Penampungan.....	88
Gambar 3.20 Panel Box Komponen.....	89
Gambar 4.1 Pembacaan Arus PWM 0 <i>OpenPLC</i>	91
Gambar 4.2 Pembacaan Arduino IDE	91
Gambar 4.3 Pembacaan Arus PWM 20000 <i>OpenPLC</i>	92
Gambar 4.4 Pembacaan Arduino IDE	93
Gambar 4.5 Pembacaan Arus PWM 40000 <i>OpenPLC</i>	94
Gambar 4.6 Pembacaan PWM Arduino IDE.....	94
Gambar 4.7 Pembacaan Arus PWM 65535 <i>OpenPLC</i>	95
Gambar 4.8 Pembacaan PWM Arduino IDE.....	96
Gambar 4.9 Pembacaan Temperatur Arduino IDE	99
Gambar 4.10 Pembacaan Temperatur <i>OpenPLC</i>	100
Gambar 4.11 Pembacaan Temperatur <i>OpenPLC</i>	100
Gambar 4.12 Pembacaan Temperatur <i>OpenPLC</i>	101
Gambar 4.13 Tampilan Temperatur Pada HMI.....	102
Gambar 4.14 Tampilan Temperatur Pada HMI.....	102
Gambar 4.15 Tampilan Temperatur Pada HMI.....	103
Gambar 4.16 Temperatur Aktual Air Pada <i>Thermometer</i>	104
Gambar 4.17 a.) Input <i>OpenPLC</i> b.) Keluar Arduino IDE 10% c.) Pengukuran Tegangan 10% d.) Pengukuran Arus 10%.....	106
Gambar 4.18 a.) Input <i>OpenPLC</i> b.) Keluar Arduino IDE 20% c.) Pengukuran Tegangan 20% d.) Pengukuran Arus 20%.....	107
Gambar 4.19 a.) Input <i>OpenPLC</i> b.) Keluar Arduino IDE 30% c.) Pengukuran Tegangan 30% d.) Pengukuran Arus 30%.....	108
Gambar 4.20 a.) Input <i>OpenPLC</i> b.) Keluar Arduino IDE 40% c.) Pengukuran Tegangan 40% d.) Pengukuran Arus 40%.....	110
Gambar 4.21 a.) Input <i>OpenPLC</i> b.) Keluar Arduino IDE 50% c.) Pengukuran Tegangan 50% d.) Pengukuran Arus 50%.....	111

Gambar 4.22 a.) Input OpenPLC b.) Keluar Arduino IDE 60% c.) Pengukuran Tegangan 60% d.) Pengukuran Arus 60%.....	112
Gambar 4.23 a.) Input OpenPLC b.) Keluar Arduino IDE 70% c.) Pengukuran Tegangan 70% d.) Pengukuran Arus 70%.....	113
Gambar 4.24 a.) Input OpenPLC b.) Keluar Arduino IDE 80% c.) Pengukuran Tegangan 80% d.) Pengukuran Arus 80%.....	114
Gambar 4.25 a.) Input OpenPLC b.) Keluar Arduino IDE 90% c.) Pengukuran Tegangan 90% d.) Pengukuran Arus 90%.....	115
Gambar 4.26 a.) Input OpenPLC b.) Keluar Arduino IDE 100% c.) Pengukuran Tegangan 100% d.) Pengukuran Tegangan 100% e.) Pengukuran Arus 100% f.) Pengukuran Arus 100%.....	117
Gambar 4.27 Grafik Tegangan Percobaan Daya Heater.....	119
Gambar 4.28 Grafik Arus Percobaan Daya Heater.....	120
Gambar 4.29 Kondisi Ladder dan <i>Peltier</i> penampungan OFF.....	121
Gambar 4.30 Kondisi Ladder dan <i>Peltier</i> Penampungan ON.....	121
Gambar 4.31 Kondisi Ladder Validasi <i>setpoint</i>	123
Gambar 4.32 Kondisi Ladder Validasi dan Perhitungan <i>Error</i>	123
Gambar 4.33 Kondisi Ladder Perhitungan <i>Error</i>	123
Gambar 4.34 Kondidi Ladder Perhitungan <i>Error</i>	123
Gambar 4.35 Kondisi Ladder Validasi <i>Peltier</i> Tangki <i>Final</i> ON/OFF.....	124
Gambar 4.36 Pengukuran Arus <i>Peltier</i>	125
Gambar 4.37 Aktual Air Pengujian.....	128
Gambar 4.38 <i>Input</i> PWM <i>OpenPLC</i>	130
Gambar 4.39 Perhitungan penerimaan PWW <i>Heater</i> Arduino IDE.....	131
Gambar 4.40 Tampilan Pembacaan PWM <i>Heater</i> Pada Arduino IDE.....	132
Gambar 4.41 Pembacaan Tegangan <i>Heater</i>	133
Gambar 4.42 Pembacaan Arus <i>Heater</i>	133
Gambar 4. 43 Grafik Open Loop Setpoint 35°C.....	141
Gambar 4. 44 Grafik Open Loop Setpoint 45°C.....	142
Gambar 4. 45 Grafik Open Loop Setpoint 55°C.....	144
Gambar 4.46 Hasil grafik Percobaan <i>Setpoint</i> 35 °C.....	147
Gambar 4.47 Hasil Grafik Keterangan <i>Setpoint</i> 35 °C.....	148

Gambar 4.48 Hasil Grafik <i>Error Setpoint</i> Terhadap Aktual Temperatur.....	149
Gambar 4.49 Grafik Laju PWM percobaan <i>Setpoint</i> 35°C	150
Gambar 4.50 Pembacaan Temperatur Pada Akhir Pengujian di Arduino IDE ...	152
Gambar 4.51 Pengukuran Akatual di Akhir Pengujian.....	153
Gambar 4.52 Hasil Grafik Percobaan 45°C.....	154
Gambar 4.53 Hasil Grafik Keterangan <i>Setpoint</i> 45 °C.....	155
Gambar 4.54 Hasil Grafik <i>Error Setpoint</i> Terhadap Aktual Temperatur.....	156
Gambar 4.55 Grafik Laju PWM percobaan <i>Setpoint</i> 45°C	157
Gambar 4.56 Pembacaan Temperatur Pada Akhir Pengujian di Arduino IDE ...	159
Gambar 4.57 Pengukuran Akatual di Akhir Pengujian.....	160
Gambar 4.58 Hasil grafik Percobaan <i>Setpoint</i> 55 °C	161
Gambar 4.59 Hasil Grafik Keterangan <i>Setpoint</i> 55 °C.....	162
Gambar 4.60 Hasil Grafik <i>Error Setpoint</i> Terhadap Aktual Temperatur.....	163
Gambar 4.61 Grafik Laju PWM percobaan <i>Setpoint</i> 55°C	164
Gambar 4.62 Pembacaan Temperatur Pada Akhir Pengujian di Arduino IDE ...	166
Gambar 4.63 Penngukuran Aktual <i>Thermometer</i>	167
Gambar 4.64 Perbandingan Grafik Hasil Percobaan Closed Loop	168
Gambar 4.65 Grafik Perbandingan Open Loop dengan Closed Loop <i>Setpoint</i> 35°C	173
Gambar 4.66 Grafik Perbandingan Open Loop dengan Closed Loop <i>Setpoint</i> 45°C	175
Gambar 4.67 Grafik Perbandingan Open Loop dengan Closed Loop <i>Setpoint</i> 55°C	176

DAFTAR LAMPIRAN

Lampiran 1 Data Hasil Percobaan Setpoint 35°C	188
Lampiran 2 Data Hasil Percobaan <i>Setpoint</i> 45°C	189
Lampiran 3 Data Hasil Percobaan Setpoint 55°C	192
Lampiran 4 Tampilan HMI setpoin 35°C.....	196
Lampiran 5 Tampilan HMI Setpoint 45°C	196
Lampiran 6 Tampilan HMI Setpoint 55°C	196
Lampiran 7 Alat Keseluruhan.....	197

DAFTAR SINGKATAN DAN SIMBOL

Singkatan	Definisi
AC	<i>Alternating Current</i>
DC	<i>Direct Current</i>
PID	<i>Proportional-Integral-Derivative</i>
PWM	<i>Pulse Width Modulation</i>
PLC	<i>Programmable Logic Controller</i>
LD	<i>Ladder Diagram</i>
FBD	<i>Function Diagram Diagram</i>
ST	<i>Structured Text</i>
SFC	<i>Sequential Function Chart</i>
ADC	<i>Analog-to-Digital Converter</i>
IoT	<i>Internet of Things</i>
RTD	<i>Resistance Temperature Detector</i>
WiFi	<i>Wireless Fidelity</i>
PSU	<i>Power Supply Unit</i>
PV	<i>Process Variable</i>
RC	<i>Resistor-Capacitor</i>
GPIO	<i>General-Purpose Input/Output</i>
EMA	<i>Exponential Moving Average</i>
LPF	<i>Low-Pass Filter</i>
SP	<i>Setpoint</i>
MCB	<i>Miniature Circuit Breaker</i>
PCB	<i>Printed Circuit Board</i>
TRIAC	<i>Triode for Alternating Current</i>
RMS	<i>Root Mean Square</i>

Simbol	Definisi	Satuan
$u(t)$	Sinyal keluaran kendali	—
$e(t)$	<i>Error</i> sistem	°C atau sesuai variabel proses
$r(t)$	Nilai <i>setpoint</i>	°C atau sesuai variabel proses
$y(t)$	Nilai keluaran / process variable	°C atau sesuai variabel proses
K_p	Gain proporsional	—
K_i	Konstanta integral	menit ⁻¹
K_d	Konstanta derivatif	menit
T_i	Waktu integral	menit
T_d	Waktu derivatif	menit
KP	Parameter gain proporsional pada Function Block CTRL_PID	—
TN	Parameter waktu integral pada Function Block CTRL_PID	menit
TV	Parameter waktu derivatif pada Function Block CTRLPID	menit
t_r	<i>Rise Time</i>	menit
t_s	<i>Settling time</i>	menit
T_{SP}	Temperatur <i>setpoint</i>	°C
T_{max}	Temperatur maksimum	°C
T_{akhir}	Temperatur representatif pada bagian akhir pengujian	°C
e_{akhir}	<i>Error</i> akhir terhadap <i>setpoint</i>	°C
T_t	Temperatur pada waktu t	°C
T_0	Temperatur awal	°C

Simbol	Definisi	Satuan
ΔT_{ref}	Perubahan temperatur acuan	°C
T_{10}	Temperatur pada 10% perubahan respons	°C
$T_{63,2}$	Temperatur pada 63,2% perubahan respons	°C
T_{90}	Temperatur pada 90% perubahan respons	°C
t_{10}	Waktu pencapaian titik 10%	menit
$t_{63,2}$	Waktu pencapaian titik 63,2%	menit
t_{90}	Waktu pencapaian titik 90%	menit
τ_p	Konstanta waktu plant	menit
R	Slope respons temperatur	°C/menit
L	<i>Dead time</i>	menit
Δu	Perubahan masukan <i>Heater</i>	%
R_1	Unit reaction rate	°C/(menit·%)
D	Duty cycle PWM	—
T_{on}	Waktu sinyal HIGH	s
T_{total}	Periode PWM	s
V_{in}	Tegangan masukan	V
V_{out}	Tegangan keluaran rata-rata	V
R_{RC}	Resistansi pada rangkaian <i>RC filter</i>	Ω
C	Kapasitansi	F
τ_{RC}	Konstanta waktu <i>RC filter</i>	s
f_c	<i>Frekuensi Cut-Off</i>	Hz
α	Koefisien filter EMA	—
$x[n]$	Data pembacaan terbaru pada EMA	—
$y[n]$	Keluaran EMA saat ini	—
$y[n - 1]$	Keluaran EMA sebelumnya	—

Simbol	Definisi	Satuan
Q	Energi kalor	J
m	Massa air	kg
c	Kalor jenis air	J/(kg·°C)
ΔT	Perubahan temperatur	°C
P	Daya <i>Heater</i> / daya listrik	W
V	Tegangan listrik	V
I	Arus listrik	A
t	Waktu pemanasan	s

INTISARI
PERANCANGAN DAN IMPLEMENTASI PENGATURAN TEMPERATUR
TANGKI AIR MENGGUNAKAN *HEATER* DAN *PELTIER* DENGAN
METODE PID BERBASIS *OPENPLC* ESP32

Muhammad Haikal Ali Akbar Darul Haq

Teknologi Rekayasa Otomasi, Sekolah Vokasi, Universitas Diponegoro

Pengendalian temperatur air memerlukan sistem umpan balik agar temperatur dapat mengikuti *setpoint* dengan penyimpangan yang kecil. Penelitian ini bertujuan merancang, mengimplementasikan, dan mengevaluasi pengaturan temperatur tangki air menggunakan metode *Proportional Integral Derivative* (PID) berbasis OpenPLC pada ESP32. Sistem menggunakan dua ESP32, yaitu ESP32 Arduino IDE untuk membaca sensor PT100 dan antarmuka aktuator, serta ESP32 OpenPLC untuk menjalankan logika PID. Pertukaran data dilakukan melalui komunikasi *pin-to-pin* menggunakan PWM, RC *low-pass filter*, ADC, dan *scaling*. Karakteristik *plant* diperoleh melalui pengujian *open-loop* dengan Heater 69%, kemudian parameter PID ditentukan menggunakan metode Ziegler–Nichols *Process Reaction Curve*. Hasil *tuning* diperoleh $K_P = 40,09$, $T_N = 5,59$ menit, dan $T_V = 1,40$ menit. Pada pengujian *closed-loop* setpoint 35 °C, 45 °C, dan 55 °C, *rise time* masing-masing 19,65; 31,29; dan 47,52 menit dengan *overshoot* 8,57%; 8,89%; dan 7,27%. Dibandingkan *open-loop*, rata-rata *error* 10 menit setelah target tercapai berkurang dari 17,23% menjadi 5,34% atau turun 69,02%. OpenPLC ESP32 dapat menjalankan logika PID, namun kestabilan temperatur masih dipengaruhi komunikasi data, proses *scaling*, dan distribusi temperatur tanpa *mixing*.

Kata Kunci: *OpenPLC*, ESP32, Kontrol PID, Pengaturan Temperatur, *Ziegler Nichols Process-Reaction Curve*, Komunikasi *Pin to pin*.

ABSTRAK

DESIGN AND IMPLEMENTATION OF WATER TANK TEMPERATURE CONTROL USING HEATER AND PELTIER WITH PID METHOD BASED ON OPENPLC ESP32

Muhammad Haikal Ali Akbar Darul Haq

Automation Engineering Technology, Vocational School, Diponegoro University

Water temperature control requires a feedback system so that the temperature can follow the setpoint with minimal deviation. This study aims to design, implement, and evaluate a water tank temperature control system using a Proportional Integral Derivative (PID) method based on OpenPLC on ESP32. The system uses two ESP32 devices: an Arduino IDE-based ESP32 for PT100 sensor reading and actuator interfacing, and an OpenPLC ESP32 for executing PID logic. Data exchange is performed through pin-to-pin communication using PWM, an RC low-pass filter, ADC, and scaling. Plant characteristics are obtained from an open-loop test with a fixed 69% Heater input, and PID parameters are determined using the Ziegler–Nichols Process Reaction Curve method. The tuning results are $K_P = 40.09$, $T_N = 5.59$ min, and $T_V = 1.40$ min. Closed-loop tests at 35 °C, 45 °C, and 55 °C produced rise times of 19.65, 31.29, and 47.52 min with overshoots of 8.57%, 8.89%, and 7.27%. Compared with open-loop operation, the average error 10 min after reaching the target decreased from 17.23% to 5.34%, or by 69.02%. OpenPLC ESP32 can execute PID logic, although temperature stability is still affected by data communication, scaling, and non-uniform temperature distribution without mixing.

Keywords: OpenPLC, ESP32, PID, Temperature Control, Ziegler–Nichols, PT100.

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Perkembangan sistem otomasi industri mendorong penggunaan sistem kontrol yang mampu mengatur *variabel proses* secara otomatis. Salah satu variabel yang banyak dikendalikan dalam proses industri adalah temperatur karena perubahan temperatur dapat memengaruhi kualitas proses maupun hasil produksi. Pengendalian temperatur banyak diterapkan pada industri kimia, makanan dan minuman, energi, serta berbagai proses manufaktur lainnya[1].

Tangki air dapat digunakan sebagai salah satu plant untuk mempelajari sistem pengendalian temperatur karena perubahan temperatur air dapat diamati secara langsung terhadap waktu. Pengaturan temperatur pada tangki umumnya dilakukan dengan mengatur energi panas yang diberikan oleh *heater* berdasarkan temperatur yang dibaca oleh sensor. Dengan adanya sistem pengendalian, temperatur diharapkan dapat bergerak menuju nilai *setpoint* yang telah ditentukan[2].

Salah satu metode yang banyak digunakan dalam pengendalian temperatur adalah *Proportional Integral Derivative* (PID). Pengendali PID bekerja berdasarkan selisih antara nilai *setpoint* dengan nilai aktual sistem melalui aksi proporsional, integral, dan derivatif. Beberapa penelitian menunjukkan bahwa penggunaan PID pada sistem temperatur mampu memperbaiki respons sistem dan mengurangi penyimpangan terhadap nilai yang diinginkan [3]. Penerapan PID pada sistem pemanas air juga telah digunakan untuk mengatur daya *heater* sehingga temperatur dapat diarahkan menuju *setpoint* yang ditentukan [4].

Pada penelitian pengendalian temperatur berbasis mikrokontroler, sensor dan aktuator umumnya dihubungkan langsung dengan mikrokontroler sebagai perangkat yang melakukan pembacaan data sekaligus menjalankan algoritma pengendalian. Penelitian sebelumnya menunjukkan bahwa mikrokontroler dapat digunakan untuk membaca temperatur, menjalankan PID, dan mengendalikan *heater* pada sistem pemanas air[5]. Selain *heater*, penggunaan perangkat pendingin

atau penyetabil temperatur juga dapat digunakan sebagai aktuator tambahan ketika temperatur melewati nilai yang diinginkan [6].

Penggunaan mikrokontroler memberikan kemudahan dalam pengembangan sistem, tetapi pada bidang otomasi industri pengendalian proses lebih banyak dikaitkan dengan *Programmable Logic Controller* (PLC). PLC memungkinkan logika pengendalian disusun secara terstruktur menggunakan bahasa pemrograman yang umum digunakan pada otomasi. Namun, penggunaan PLC industri untuk penelitian dan pembelajaran membutuhkan perangkat dengan biaya yang relatif lebih tinggi. Salah satu alternatif yang dapat digunakan adalah *OpenPLC*, yaitu platform PLC berbasis perangkat lunak yang dapat dijalankan pada beberapa jenis perangkat keras, termasuk mikrokontroler ESP32.

OpenPLC pada ESP32 memberikan kemungkinan untuk menjalankan logika PLC dengan perangkat yang lebih sederhana dan biaya yang lebih rendah. Meskipun demikian, penerapan *OpenPLC* pada sistem kontrol nyata tidak hanya berkaitan dengan pembuatan program, tetapi juga bagaimana sensor, data proses, dan aktuator dapat diintegrasikan dengan perangkat tersebut. Pada penelitian ini *OpenPLC* digunakan untuk menjalankan logika pengendalian PID, sedangkan implementasi sistem juga dievaluasi dari sisi penerimaan data temperatur dan keluaran pengendalian ketika diterapkan langsung pada plant.

Pada implementasinya, sistem menggunakan dua ESP32 dengan fungsi yang berbeda. ESP32 yang diprogram menggunakan Arduino IDE digunakan untuk membaca sensor PT100 dan menangani antarmuka perangkat keras, sedangkan ESP32 yang menjalankan *OpenPLC* digunakan untuk mengolah nilai temperatur dan menjalankan logika PID. Pertukaran data antara kedua ESP32 dilakukan menggunakan komunikasi *pin to pin* berbasis PWM yang dilewatkan melalui *RC low pass filter*, kemudian dibaca menggunakan ADC dan dilakukan proses *scaling*. Arsitektur ini digunakan agar nilai temperatur hasil pembacaan sensor dapat diteruskan menuju *OpenPLC* sebagai *process variable* dan keluaran PID dapat dikirim kembali untuk mengatur *heater*.

Berdasarkan kondisi tersebut, penelitian ini tidak hanya membahas pengaturan temperatur menggunakan PID, tetapi juga melihat bagaimana *OpenPLC* berbasis ESP32 dapat diterapkan pada sistem pengendalian temperatur yang sebenarnya.

Pengujian dilakukan melalui pengujian *open loop* untuk memperoleh karakteristik sistem dan parameter awal PID, kemudian dilanjutkan dengan pengujian *closed loop* untuk melihat respons sistem serta keterbatasan yang muncul pada penerapan *OpenPLC*. *Heater* digunakan sebagai pemanas utama, sedangkan *Peltier* digunakan sebagai aktuator pendukung ketika temperatur melebihi batas yang ditentukan.

Berdasarkan uraian tersebut, penelitian ini dilakukan dengan judul “Perancangan dan Implementasi Pengaturan Temperatur Tangki Air Menggunakan *Heater* dan *Peltier* dengan Metode PID Berbasis *OpenPLC* ESP32.” Penelitian diarahkan untuk merancang dan mengimplementasikan sistem pengaturan temperatur sekaligus mengevaluasi kemampuan *OpenPLC* berbasis ESP32 dalam menjalankan logika PID serta melihat keterbatasannya ketika diterapkan pada sistem nyata.

Penelitian ini juga melanjutkan apa yang sudah diteliti oleh teman saya yaitu Rizal Nadhif Pratista dengan judul “Implementasi *OpenPLC* Berbasis ESP32 Menggunakan Metode PID Pada Sistem Pengendalian Laju Aliran Air “, dan penelitian saya juga dilanjutkan oleh teman saya yaitu Afdhol Arafat Lintang PPSN dengan judul “Rancangan Dan Implementasi *Human Machin Interface* (HMI) Berbasis *Node-Red* Untuk Monitoring Sistem Kontrol Temperatur Dan Laju Aliran Air“.

1.2 Perumusan Masalah

Berdasarkan latar belakang telah dijelaskan sebelumnya, maka rumusan masalah dalam penelitian tugas akhir ini dapat dirumuskan berikut:

1. Bagaimana merancang dan mengimplementasikan sistem pengaturan temperatur air menggunakan metode PID berbasis *OpenPLC* pada mikrokontroler ESP32?
2. Bagaimana mengintegrasikan ESP32 Arduino IDE sebagai perangkat pembacaan sensor dan antarmuka aktuator dengan ESP32 *OpenPLC* sebagai pengolah logika pengendalian?
3. Bagaimana menerapkan komunikasi *pin to pin* berbasis PWM, *RC low pass filter*, ADC, dan *scalling* sebagai jalur pertukaran data antara ESP32 Arduino IDE dan ESP32 *OpenPLC*?
4. Bagaimana memperoleh parameter awal PID berdasarkan karakteristik respons *open loop* sistem pemanas menggunakan metode *Ziegler Nichols Process Reaction Curve*?
5. Bagaimana kinerja *OpenPLC* berbasis ESP32 dalam menjalankan logika PID pada sistem *closed loop* pengaturan temperatur air menggunakan *Heater* dan *Peltier*?
6. Apa saja keterbatasan yang muncul pada implementasi *OpenPLC* berbasis ESP32, khususnya pada pembacaan data sensor, komunikasi antar ESP32, proses *scalling*, dan pengaruhnya terhadap keluaran pengendalian?

1.3 Tujuan Tugas Akhir

Adapun tujuan dari penelitian tugas akhir ini untuk :

1. Merancang dan mengimplementasikan sistem pengaturan temperatur air menggunakan metode PID berbasis *OpenPLC* pada mikrokontroler ESP32.
2. Mengintegrasikan ESP32 Arduino IDE sebagai perangkat pembacaan sensor dan antarmuka aktuator dengan ESP32 *OpenPLC* sebagai pengolah logika pengendalian.
3. Menerapkan komunikasi *pin to pin* berbasis PWM, *RC low pass filter*, ADC, dan *scalling* sebagai jalur pertukaran data antara ESP32 Arduino IDE dan ESP32 *OpenPLC*.

4. Mengidentifikasi karakteristik sistem pemanas melalui pengujian *open loop* dan memperoleh parameter awal PID menggunakan metode *Ziegler Nichols Process Reaction Curve*.
5. Mengevaluasi kinerja *OpenPLC* berbasis ESP32 dalam menjalankan logika PID pada sistem *closed loop* pengaturan temperatur air menggunakan *Heater* dan *Peltier*.
6. Menganalisis keterbatasan implementasi *OpenPLC* berbasis ESP32 yang berkaitan dengan pembacaan data sensor, komunikasi antar ESP32, proses *scalling*, dan pengaruhnya terhadap hasil pengendalian.

1.4 Manfaat Tugas Akhir

Adapun manfaat yang diharapkan dari penelitian ini adalah sebagai berikut:

1. Manfaat Akademis

Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan ilmu pengetahuan khususnya pada bidang sistem kontrol dan otomasi industri, terutama dalam penerapan metode *Proportional Integral Derivative* (PID) pada sistem kontrol temperatur berbasis *OpenPLC* dan mikrokontroler ESP32.

2. Manfaat Praktis

Penelitian ini diharapkan dapat menghasilkan modul praktikum sistem kontrol temperatur air yang dapat digunakan sebagai media pembelajaran di laboratorium untuk membantu mahasiswa memahami konsep sistem kontrol, metode PID, serta implementasi *OpenPLC* dalam sistem otomasi.

3. Manfaat Teknologi

Penelitian ini diharapkan dapat menjadi referensi dalam pengembangan sistem kontrol temperatur berbasis *OpenPLC* menggunakan mikrokontroler ESP32 dan metode PID, yang lebih ekonomis dan fleksibel dibandingkan dengan penggunaan PLC industri konvensional.

4. Manfaat Pendidikan

Penelitian ini diharapkan dapat menghasilkan sebuah modul praktikum sistem monitoring berbasis HMI yang dapat digunakan sebagai media pembelajaran sistem kontrol proses otomasi, sehingga mahasiswa dapat memahami konsep integrasi antara sensor, sistem kontrol, serta sistem monitoring berbasis IoT.

1.5 Pembatasan Masalah

Agar penelitian lebih terarah dan tidak menyimpang dari tujuan yang telah ditetapkan, maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem yang dirancang difokuskan pada pengaturan temperatur air pada tangki sebagai variabel proses utama.
2. Metode kontrol yang digunakan adalah *Proportional Integral Derivative* (PID) dengan parameter awal yang diperoleh menggunakan metode tuning *Ziegler Nichols Process Reaction Curve* berdasarkan respon *Open Loop*.
3. Pengujian temperatur dilakukan sampai maksimum 55 °C.
4. *OpenPLC* dijalankan pada mikrokontroler ESP32 dan digunakan sebagai bagian utama untuk menjalankan logika pengendalian PID.
5. Pembacaan sensor PT100 dilakukan menggunakan ESP32 yang diprogram melalui Arduino IDE, kemudian data dikirim menuju ESP32 *OpenPLC* melalui komunikasi *pin to pin* berbasis PWM.
6. Sinyal PWM pada komunikasi *pin to pin* dilewatkan melalui rangkaian *RC filter* untuk menghasilkan tegangan analog yang selanjutnya dibaca oleh ADC pada sisi penerima.
7. Nilai ADC yang diterima oleh ESP32 *OpenPLC* dilakukan proses *scalling* untuk memperoleh nilai temperatur yang digunakan sebagai *Process Variable* pada pengendali PID.
8. Aktuator utama yang digunakan adalah *Heater* sebagai elemen pemanas, sedangkan modul *Peltier* digunakan sebagai elemen pendukung untuk membantu stabilisasi temperatur.
9. Sistem menggunakan dua tangki, yaitu tangki proses dengan kapasitas maksimum 6 liter dan tangki penampungan.
10. Pada tangki *Final* tidak terdapat *mixing* untuk meratakan temperatur air.
11. *Peltier* pada tangki penampungan digunakan untuk membantu menurunkan temperatur air yang berasal dari tangki proses apabila temperatur air masih tinggi.
12. Pengujian *open loop* digunakan untuk memperoleh karakteristik respon sistem pemanas dan parameter yang digunakan dalam proses tuning PID.
13. Pengujian *closed loop* digunakan untuk mengevaluasi respon sistem setelah parameter PID diterapkan pada *OpenPLC*.

14. Evaluasi *closed loop* difokuskan pada kestabilan pembacaan temperatur, perubahan keluaran PID, dan kemampuan sistem mengikuti *setpoint*. Parameter seperti *Rise Time*, *Overshoot*, *Settling time*, dan *Error* akhir terhadap *setpoint* hanya dianalisis apabila respon sistem memenuhi kondisi yang memungkinkan parameter tersebut ditentukan secara valid.
15. Pengujian dilakukan pada skala prototipe dengan kondisi lingkungan laboratorium.
16. Penelitian difokuskan pada evaluasi kemampuan *OpenPLC* berbasis ESP32 dalam menjalankan logika kontrol PID serta pengaruh kualitas pembacaan data terhadap kinerja sistem pengaturan temperatur.

1.6 Sistematika Tugas Akhir

Dalam penulisan laporan tugas akhir ini, terdapat sistematika penyusunan laporan sebagai berikut :

BAB I PENDAHULUAN

Bagian ini berisi tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, serta sistematika penulisan proposal tugas akhir sesuai topik yang diangkat penulis.

BAB II DASAR TEORI

Bagian ini berisikan teori yang mendukung topik yang di angkat oleh penulis.

BAB III METODE PENELITIAN

Bagian ini berisi tentang metode yang digunakan dalam penelitian penulis.

BAB IV PENGUJIAN DAN ANALISIS

Berisi hasil pengujian dan analisis kinerja sistem.

BAB V PENUTUP

Berisi kesimpulan dan saran dari hasil penelitian.