



UNIVERSITAS
Dinamika

**Perancangan Modul Pembelajaran Sistem Kontrol Menggunakan
PID Berdasarkan Sensor Suhu di Laboratorium FTI Universitas
Dinamika**



KERJA PRAKTIK

Program Studi

UNIVERSITAS
Dinamika

S1 Teknik Komputer

Oleh:

Albert

22410200005

FAKULTAS TEKNOLOGI DAN INFORMATIKA

UNIVERSITAS DINAMIKA

2025

**Perancangan Modul Pembelajaran Sistem Kontrol Menggunakan PID
Berdasarkan Sensor Suhu di Laboratorium FTI
Universitas Dinamika**

Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Strata Satu (S1)
Disusun Oleh :



Nama : ALBERT
Nim : 22410200005
Program : S1 (Strata Satu)
Jurusan : Teknik Komputer

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA
2025**

MOTO



"Jika kamu bisa membayangkannya, kamu bisa mencapainya."

UNIVERSITAS
Dinamika

LEMBAR PENGESAHAN
Perancangan Modul Pembelajaran Sistem Kontrol Menggunakan PID
Berdasarkan Sensor Suhu di Laboratorium FTI
Universitas Dinamika

Laporan Kerja Praktik oleh
ALBERT
NIM. 22414200005
Telah diperiksa, diuji, dan disetujui

Surabaya, 8 Juli 2025

Disetujui:

 Pembimbing

Pauladie Susanto, S.Kom., M.T.
NIDN. 0729047501

Penyelia
 
Teguh Sutanto, M.Kom.
NIK. 000290

Mengetahui,
Ketua Program Studi Teknik Komputer
Fakultas Teknologi dan Informatika
 **UNIVERSITAS**
Dinamika 
Pauladie Susanto, S.Kom., M.T.
NIDN. 0729047501

PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa **Universitas Dinamika**, Saya :

Nama : **Albert**
NIM : **22410200005**
Program Studi : **S1 Teknik Komputer**
Fakultas : **Fakultas Teknologi dan Informatika**
Jenis Karya : **Laporan Kerja Praktik**
Judul Karya : **PERANCANGAN MODUL PEMBELAJARAN
SISTEM KONTROL MENGGUNAKAN PID
BERDASARKAN SENSOR SUHU DI
LABORATORIUM FTI UNIVERSITAS DINAMIKA**

Menyatakan dengan sesungguhnya bahwa :

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, Saya menyetujui memberikan kepada **Universitas Dinamika** Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalty Free Right*) atas seluruh isi/sebagian karya ilmiah Saya tersebut diatas untuk disimpan, dialihmediakan, dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama Saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut diatas adalah hasil karya asli Saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya, atau pendapat orang lain yang ada dalam karya ilmiah ini semata-mata hanya sebagai rujukan yang dicantumkan dalam Daftar Pustaka Saya.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiasi pada karya ilmiah ini, maka Saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada Saya.

Demikian surat pernyataan ini Saya buat dengan sebenar-benarnya.

Surabaya, 7 Juli 2025



Albert
NIM : 22410200005

ABSTRAK

Kerja praktek ini bertujuan untuk merancang dan mengembangkan modul pembelajaran sistem kontrol PID yang bersifat praktis dan interaktif, sebagai media bantu pembelajaran mahasiswa di Laboratorium FTI Universitas Dinamika. Modul ini dirancang agar mahasiswa dapat memahami penerapan prinsip dasar kontrol PID (Proportional, Integral, Derivative) secara langsung melalui pengaturan parameter K_p , K_i , dan K_d untuk mengontrol kecepatan motor DC berdasarkan perhitungan nilai RPM (Rotasi per Menit). Sistem ini menggunakan sensor suhu LM35 sebagai acuan kendali, dengan batas atas dan bawah suhu yang telah ditentukan sebelumnya. Nilai Set Point (kecepatan motor) juga telah diatur agar tidak dapat diubah oleh pengguna, sehingga fokus pembelajaran tertuju pada bagaimana perubahan parameter PID memengaruhi sistem. Input parameter PID dilakukan melalui Keypad 4x4, sedangkan nilai PID dan hasil RPM ditampilkan pada LCD 16x2. Motor DC akan berputar sesuai dengan hasil pengendalian sistem. Perangkat yang digunakan adalah Arduino IDE dengan Arduino UNO sebagai mikrokontroler utama. Dengan adanya modul ini, diharapkan mahasiswa tidak hanya memahami konsep teoritis PID, tetapi juga mampu mengamati respons sistem secara nyata dan mengembangkan keterampilan praktis dalam merancang sistem kontrol berbasis mikrokontroler.

Kata Kunci : Sistem Kontrol PID, Arduino UNO, Sensor LM35, Motor DC, RPM, Modul Pembelajaran.



UNIVERSITAS
Dinamika

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa atas segala rahmat yang telah diberikan-Nya, sehingga penulis dapat menyelesaikan laporan kerja praktek ini sebagai salah satu persyaratan untuk menyelesaikan program sarjana pada program studi S1 Teknik Komputer Universitas Dinamika Surabaya.

Laporan kerja praktik di Laboratorium IOT Univesitas Dinamika Surabaya, disusun berdasarkan hasil kerja yang telah dilaksanakan di bulan Februari sampai bulan maret. Dan hasil dari pelaksanaan kerja praktik ini, bertujuan untuk mengembangkan dan membuat Modul Pembelajaran Sistem Kontrol Menggunakan PID Berdasarkan Sensor Suhu.

Dalam penyusunan laporan ini, penulis menyadari masih banyak kekurangan baik dari segi susunan serta cara penulisan laporan ini, oleh karena itu harapan penulis agar bisa mendapatkan saran dan kritik yang sifatnya membangun demi kesempurnaan laporan ini. Semoga laporan ini bisa bermanfaat bagi para pembaca pada umumnya dan juga bermanfaat bagi penyusun.

Surabaya, 7 Juli 2025

Albert

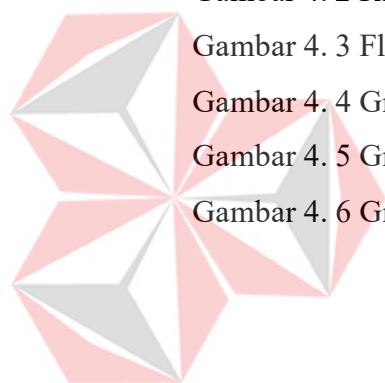
DAFTAR ISI

ABSTRAK	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xi
BAB I	1
PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Manfaat	3
BAB II	4
GAMBARAN UMUM PERUSAHAAN	4
2.1 Sejarah Singkat Perusahaan	4
2.1.1. TUJUAN	6
2.1.2. Struktur Organisasi.....	6
2.1.3. Informasi Kontak.....	7
2.2 Program Studi S1 Teknik Komputer.....	7
2.2.1. VISI MISI	8
2.2.2. TUJUAN	8
2.2.3. PROFESI LULUSAN	9
BAB III.....	10
LANDASAN TEORI	10
3.1 Sistem Kontrol.....	10
3.1.1. Elemen Penting dalam Sistem Kontrol	11
3.1.2. Representasi dan Analisis Sistem Kontrol.....	12
3.1.3. Penerapan Sistem Kontrol dalam Kehidupan Nyata.....	12
3.1.4. Perkembangan dalam Sistem Kontrol.....	13
3.2 Pengendali PID.....	13
3.2.1. Aplikasikan Kontroller PID dalam Pengendalian Proses	15

3.2.2.	Keuntungan Kontoller PID	15
3.2.3.	Tantangan dalam Penggunaan Kontroller PID	16
3.3	Arduino UNO	17
3.4	Sensor LM35	18
3.5	Keypad 4x4.....	20
3.6	LCD (Liquid Crystal Display) 16X2	22
3.7	Motor DC JGA25-370	24
3.8	Driver Motor Ardumoto.....	26
BAB IV		28
PEMBAHASAN		28
4.1	Prinsip Kerja	28
4.2	Diagram Blok	29
4.3	Perancangan Sistem	30
4.3.1	Perancangan Hardware	31
4.3.2	Perancangan software	33
4.4	Implementasi dan Pengujian.....	34
4.5	Karakteristik Motor Yang Digunakan	35
4.5.1	Uji Respons Dinamis Tanpa Beban	36
4.5.2	Analisis Grafik dan Parameter Sistem	37
4.5.3	Pemodelan Sistem Orde Satu	37
4.5.4	Fungsi Alih (Transfer Function)	38
4.6	Pengujian Transef Function Pada Motor	38
4.7	Analisa dan Pembahasan Pada Arduino	40
4.7.1	Metode Pengujian dan Sistem Kendali	40
4.7.2	Analisa Grafik Respon Sistem	44
BAB V		46
KESIMPULAN DAN SARAN		46
5.1	Kesimpulan.....	46
5.2	Saran	47
DAFTAR PUSTAKA		48

DAFTAR GAMBAR

Gambar 2. 1 Universitas Dinamika	4
Gambar 2. 2 Struktur Organisasi	6
Gambar 3. 1 Rumus PID	14
Gambar 3. 2 Arduino UNO	17
Gambar 3. 3 Sensor LM35	18
Gambar 3. 4 Keypad 4x4	20
Gambar 3. 5 LCD 16X2	22
Gambar 3. 6 Motor DC JGA25-370	24
Gambar 3. 7 Driver Motor Ardumoto	26
Gambar 4. 1 Diagram Blok	29
Gambar 4. 2 Rangkaian Alat	31
Gambar 4. 3 Flowchart Cara Kerja Alat	33
Gambar 4. 4 Grafik Tanggapan Waktu Motor DC JGA25-370 Tanpa Beban	37
Gambar 4. 5 Grafik Motor Dengan Transfer Function	39
Gambar 4. 6 Grafik PID Perubahan Set Point Berdasarkan Suhu	45



UNIVERSITAS
Dinamika

DAFTAR TABEL

Tabel 4. 1 Tabel Alamat PIN	32
Tabel 4. 2 Karakteristik Motor	36



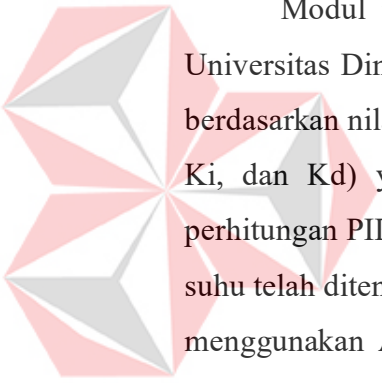
UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem kontrol PID (Proportional, Integral, Derivative) merupakan salah satu metode pengendalian yang banyak digunakan dalam dunia teknik, terutama untuk mengatur suhu, tekanan, dan kecepatan motor. Namun dalam proses pembelajaran ini, mahasiswa sering kali hanya memahami konsep PID secara teori tanpa adanya praktik langsung yang mendukung. Hal ini membuat pemahaman menjadi kurang maksimal. Untuk itu, dibutuhkan sebuah modul pembelajaran yang sederhana dan mudah digunakan agar mahasiswa dapat memahami penerapan PID secara nyata.



Modul ini dirancang sebagai media pembelajaran di Laboratorium IoT Universitas Dinamika. Modul ini berfungsi untuk mengatur kecepatan motor DC berdasarkan nilai suhu yang terbaca dari sensor LM35, dengan parameter PID (K_p , K_i , dan K_d) yang dapat diatur melalui Keypad 4x4. Nilai RPM dan hasil perhitungan PID ditampilkan pada LCD 16x2, sedangkan nilai set point dan batas suhu telah ditentukan agar pengguna hanya fokus pada pengaturan PID. Sistem ini menggunakan Arduino UNO sebagai mikrokontroler dan Arduino IDE sebagai software pemrogramannya.

Dengan adanya modul ini, mahasiswa diharapkan dapat lebih memahami prinsip kerja sistem kontrol PID secara langsung. Modul ini juga memberikan pengalaman praktik dalam merancang sistem kontrol sederhana yang berbasis mikrokontroler, sehingga dapat mendukung proses pembelajaran yang lebih efektif di Laboratorium IoT Universitas Dinamika.

1.2 Rumusan Masalah

Dari latar belakang, maka dapat dirinci perumusan masalah sebagai berikut:

1. Bagaimana merancang modul pembelajaran yang dapat membantu mahasiswa memahami konsep dasar sistem kontrol PID secara praktis dan interaktif?
2. Bagaimana pengguna dapat mengatur parameter PID (K_p , K_i , K_d) untuk mempengaruhi kecepatan motor berdasarkan suhu lingkungan?
3. Bagaimana menampilkan hasil perhitungan PID dan nilai kecepatan motor (RPM) secara jelas dan informatif menggunakan LCD 16x2?
4. Bagaimana Respon sistem saat terjadi perubahan setpoint yang di akibatkan oleh sensor LM35?

1.3 Batasan Masalah

Batasan masalah pada pelaksanaan kerja praktik ini adalah sebagai berikut:

1. Modul hanya mengatur kecepatan motor DC menggunakan sistem kontrol PID berbasis suhu dari sensor LM35.
2. Parameter PID yang digunakan terbatas pada tiga komponen utama, yaitu K_p , K_i , dan K_d , yang dapat diatur melalui keypad 4x4.
3. Nilai Set Point kecepatan motor ditetapkan secara tetap, yaitu batas atas 200 RPM saat suhu di atas 30°C, dan tidak dapat diubah oleh pengguna.
4. Penyesuaian kecepatan motor (RPM) hanya didasarkan pada suhu yang terdeteksi oleh sensor LM35 dan nilai parameter PID yang dimasukkan.

1.4 Tujuan

Tujuan Kerja praktik di Laboratorium IOT Universitas Dinamika Surabaya adalah sebagai berikut:

1. Merancang dan mengembangkan modul pembelajaran sistem kontrol PID yang bersifat praktis dan interaktif untuk membantu mahasiswa memahami konsep dasar kendali otomatis.
2. Mengimplementasikan sistem kontrol kecepatan motor DC dengan pengaturan parameter PID (K_p , K_i , K_d) oleh pengguna berdasarkan input suhu dari sensor LM35.
3. Menampilkan informasi hasil perhitungan PID dan nilai RPM motor secara real-time dan informatif menggunakan LCD 16x2 sebagai alat bantu visual dalam proses pembelajaran.

1.5 Manfaat

Modul pembelajaran sistem kontrol PID yang dirancang di Laboratorium FTI Universitas Dinamika memiliki berbagai manfaat, baik bagi pengguna maupun lingkungan laboratorium itu sendiri, antara lain:

1. Dengan modul ini, mahasiswa dapat lebih mudah memahami konsep dasar dan aplikasi sistem kontrol PID secara praktis. Mereka akan belajar bagaimana cara mengatur parameter PID (K_p , K_i , K_d) dan melihat dampaknya terhadap pengendalian motor DC, yang dapat memperdalam pengetahuan mereka dalam bidang kontrol otomatis.
2. Modul ini berfungsi sebagai alat pendidikan yang sangat berguna di Laboratorium FTI Universitas Dinamika. Mahasiswa dapat memanfaatkan modul ini untuk eksperimen langsung dalam pengaturan suhu dan kecepatan motor, yang akan mendukung pembelajaran dalam mata kuliah terkait otomasi dan sistem kontrol.
3. Melalui penggunaan perangkat keras seperti Arduino UNO, keypad 4x4, dan sensor LM35, mahasiswa akan memperoleh keterampilan praktis dalam merancang dan mengimplementasikan sistem kontrol menggunakan algoritma PID.

BAB II

GAMBARAN UMUM PERUSAHAAN

2.1 Sejarah Singkat Perusahaan



Gambar 2. 1 Universitas Dinamika

Pembangunan teknologi dan informasi menjadi hal penting dalam pembangunan dan pengembangan nasional. 2 hal tersebut juga harus diiringi dengan pengetahuan di bidang ekonomi dan bisnis untuk bisa bersaing di era yang terus berkembang. Tidak lupa kebudayaan dan seni harus tetap di pertahankan agar identitas bangsa tidak musnah. Melalui kemajuan teknologi informasi dan ekonomi Negara bisa berkembang dan menjawab seluruh tantangan. Melalui 4 hal utama, kritis dalam menyelesaikan suatu permasalahan, kreatif dalam menciptakan inovasi, berkolaborasi dengan seluruh pihak, serta membangun komunikasi seluas-luasnya untuk terciptanya suatu hubungan yang baik.

Pada tanggal 30 April 1983, Yayasan Putra Bhakti mendirikan Akademi Komputer dan Informatika Surabaya (AKIS), yang fokus pada pendidikan teknologi informasi. Tahun 1984, AKIS mendapatkan izin operasional untuk program Diploma III Manajemen Informatika. Pada tanggal 20 Maret 1986,

AKIS berubah menjadi Sekolah Tinggi Manajemen Informatika dan Teknik Komputer Surabaya (STIKOM Surabaya). Pada tahun 1987, STIKOM Surabaya mendirikan kampus pertama di Jalan Kutisari 66 Surabaya.

Kemudian pada September 1997, STIKOM Surabaya membangun gedung baru di Jalan Raya Kedung Baruk 98. Dan di tahun 2012, STIKOM berubah nama menjadi Sekolah Tinggi Manajemen Informatika dan Teknik Komputer Surabaya (STMIK STIKOM Surabaya). Pada tanggal 4 September 2014, STMIK STIKOM Surabaya berubah menjadi Institut Bisnis dan Informatika STIKOM Surabaya dengan 2 Fakultas dan 9 program studi.

Pada tanggal 29 Juli 2019 Melalui Surat Keputusan Riset Dikti, Institut Bisnis dan Informatika STIKOM Surabaya resmi berubah menjadi Universitas Dinamika yang memiliki 2 fakultas dengan 9 program studi, yakni Fakultas Teknologi dan Informatika, Prodi S1 Teknik Komputer, Prodi S1 Desain Komunikasi Visual, Prodi S1 Desain Produk, Prodi D4 Produksi Film dan Televisi, dan Prodi D3 Sistem Informasi. Serta Fakultas Ekonomi dan Bisnis (FEB) dengan Prodi S1 Manajemen, Prodi S1 Akuntansi, dan Prodi S1 Administrasi Perkantoran.

Pada 31 Mei 2021 Melalui Surat Keputusan Rektor, Universitas Dinamika melakukan perubahan struktur organisasi dengan membentuk fakultas baru, yakni Fakultas Desain dan Industri Kreatif (FDIK) dengan 3 program studi, yaitu Prodi S1 Desain Produk, Prodi S1 Desain Komunikasi Visual, dan D4 Produksi Film dan Televisi yang sebelumnya berada dibawah naungan Fakultas Teknologi dan Informatika (FTI).

Universitas Dinamika telah mengalami perkembangan yang signifikan dari pendiriannya sebagai AKIS hingga menjadi universitas yang komprehensif dengan berbagai fakultas dan program studi yang mendukung pengembangan teknologi informasi, ekonomi, bisnis, dan industri kreatif di Surabaya dan sekitarnya.

2.1.1. Visi dan Misi

Visi

Menjadi *smart entrepreneurial university* berskala global yang produktif dalam berinovasi.

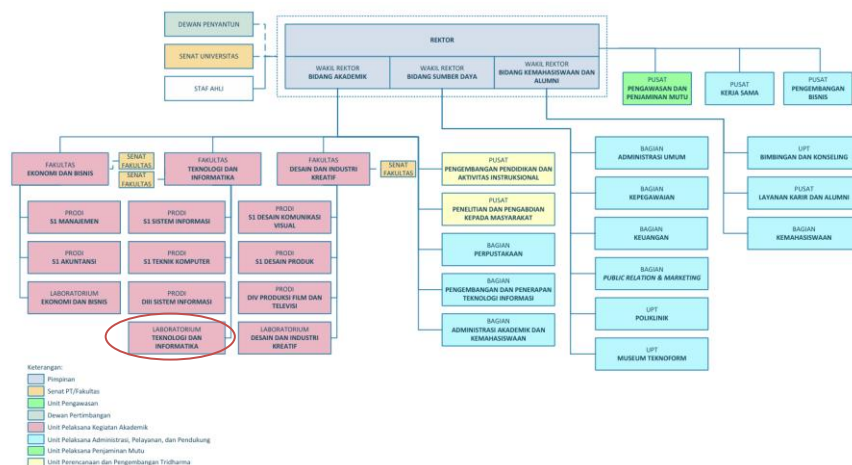
Misi

1. Menyelenggarakan dan mengembangkan pendidikan berbasis teknologi informasi yang bermutu dan berdaya saing global.
2. Melaksanakan penelitian yang berfokus pada pengembangan inovasi untuk mewujudkan entrepreneurial university.
3. Melakukan pengabdian untuk menyebar luaskan ipteks dan hasil inovasi bagi kesejahteraan masyarakat.
4. Melaksanakan kemitraan berskala global.
5. Mengembangkan bisnis dan kewirausahaan secara otonom yang akuntabel dan transparan.

2.1.1. TUJUAN

1. Menyelenggarakan pendidikan yang berkualitas, inovatif, dan futuristik.
 2. Menciptakan SDM berdaya saing global dan berjiwa *entrepreneur*.
 3. Menghasilkan penelitian berkualitas dan berskala global.
 4. Menghasilkan inovasi yang bernilai jual dan bermanfaat bagi masyarakat.
 5. Melaksanakan diseminasi ipteks dan/atau hasil inovasi untuk meningkatkan kesejahteraan masyarakat.
 6. Mewujudkan kemitraan berskala global.
- Menjamin keberlanjutan Perguruan Tinggi.

2.1.2. Struktur Organisasi



Gambar 2. 2 Struktur Organisasi

Lingkaran merah pada gambar tersebut menunjukkan Laboratorium

Teknologi dan Informatika, yang berada di bawah Fakultas Teknologi dan Informatika, Universitas Dinamika. Di laboratorium inilah kamu melaksanakan kerja praktik untuk merancang modul pembelajaran sistem kontrol berbasis PID menggunakan sensor suhu, dengan tujuan mendukung proses pembelajaran mahasiswa secara praktis dan interaktif.

2.1.3. Informasi Kontak

Tempat : Jl. Raya Kedung Baruk 98 Surabaya

Email : official@dinamika.ac.id

Website : dinamika.ac.id

No telfon : +62(31)8721731

Sosial Media

Facebook : Universitas Dinamika

Youtube : Universitas Dinamika

Instagram : @universitasdinamika

2.2 Program Studi S1 Teknik Komputer

Program Studi S1 Teknik Komputer mulai beroperasi sejak tahun 1991. Tujuan dari program ini adalah menghasilkan lulusan yang memiliki kompetensi di bidang sistem komputer dan otomasi industri. Mahasiswa dibekali dengan berbagai mata kuliah pendukung, seperti Mikrokontroler, Programmable Logic Controller (PLC), Sistem Digital, Jaringan Komputer, serta Robotika.

Lulusan dari program ini diharapkan mampu melakukan analisis, perancangan, dan pengembangan sistem otomasi industri yang berbasis komputer. Karier yang dapat dijalani antara lain sebagai Computer System Engineer, Embedded System Developer, dan Automation Engineer.

Program ini juga memberikan penekanan pada keseimbangan antara pemahaman teoritis dan keterampilan praktis melalui kegiatan seperti praktikum, penelitian, dan proyek akhir. Mahasiswa juga dibekali kemampuan dalam pemrograman tingkat lanjut, pengolahan data, serta pengembangan sistem berbasis kecerdasan buatan (AI). Kurikulum yang disusun selalu diperbarui agar selaras dengan perkembangan kebutuhan industri, sehingga lulusan S1 Teknik Komputer Universitas Dinamika mampu bersaing di dunia kerja maupun melanjutkan pendidikan ke jenjang lebih tinggi.

Selain itu, mahasiswa diberikan peluang untuk mengikuti sertifikasi industri seperti Cisco Certified Network Associate (CCNA) dan sertifikasi di bidang embedded system programming, guna meningkatkan daya saing di dunia profesional. Melalui kegiatan kerja praktik, mahasiswa dapat mengembangkan keterampilan sesuai kebutuhan industri dan memperoleh pengalaman nyata sebelum memasuki dunia kerja secara langsung.

2.2.1. VISI MISI

Visi:

Mengembangkan keilmuan di bidang IoT yang didukung oleh kecerdasan artifial, dan diintegrasikan dengan konsep *technopreneurship*, sehingga mampu menciptakan inovasi yang bermanfaat bagi masyarakat dan industri berskala global.

Misi:

1. Mengembangkan pendidikan dan pengajaran di bidang Teknik Komputer yang bermutu, berwawasan global, dan mengarah pada *technopreneurship*.
2. Melaksanakan penelitian di bidang Teknik Komputer yang inovatif dan solutif bagi masyarakat dan industri berskala global.
3. Melaksanakan pengabdian atau penerapan hasil inovasi di bidang Teknik Komputer yang bermanfaat bagi masyarakat dan industri.

2.2.2. TUJUAN

Tujuan Program Studi S1 Teknik Komputer adalah sebagai berikut :

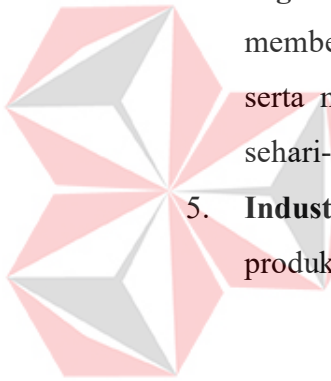
1. Lulusan memiliki kemampuan menganalisis permasalahan sistem komputer khususnya pada aspek perangkat keras untuk menghasilkan solusi bagi organisasi.
2. Lulusan memiliki kemampuan menganalisis perangkat lunak (meliputi pemrograman antarmuka, pemrograman *real-time*) dan perangkat keras (meliputi pemantauan, pengendalian) sistem komputer sebagai solusi bagi permasalahan organisasi. Lulusan memiliki kemampuan menganalisis dan merancang sistem komputer dengan menerapkan sistem tertanam, *Internet of Things*(IoT), kecerdasan artifisial, dan atau jaringan komputer untuk menghasilkan solusi bagi organisasi.
3. Lulusan yang memiliki kemampuan dalam merumuskan keputusan yang tepat berdasarkan analisis informasi dan data, beretika, dan bertanggung jawab pada

pekerjaan dalam lingkup tugasnya.

2.2.3. PROFESI LULUSAN

Profesi Lulusan Program Studi Teknik Komputer

1. **IoT Engineer** : Menyediakan produk dan atau solusi IoT sesuai dengan kebutuhan pengguna.
2. **Artificial Intelligent Engineer** : Membangun solusi berbasis kecerdasan artifisial (Artificial Intelligence/AI)
3. **Network Designer** : Melaksanakan penyediaan desain instalasi jaringan dan infrastruktur meliputi kegiatan pemetaan kebutuhan, monitoring dan pengawasan dampak design pembangunan dan pengembangan instalasi jaringan dan infrastruktur yang dibutuhkan oleh user sejalan dengan rencana dan pengembangan organisasi.
4. **Digital Computer Technology Advisor** : Memecahkan masalah teknis, memberikan saran tentang perangkat keras dan perangkat lunak yang tepat, serta mengoptimalkan penggunaan teknologi dalam bisnis atau kehidupan sehari-hari.
5. **Industrial Automation** : Meningkatkan efisiensi dan produktivitas sistem produksi di industri.



UNIVERSITAS
Dinamika

BAB III

LANDASAN TEORI

3.1 Sistem Kontrol

Sistem kontrol merupakan sekumpulan komponen atau perangkat yang bekerja secara terorganisir untuk mengatur dan mengendalikan perilaku suatu sistem agar menghasilkan keluaran (output) yang sesuai dengan tujuan yang telah ditentukan. Sistem ini umumnya melibatkan hubungan sebab-akibat antara masukan (input) dan keluaran (output) yang diatur oleh suatu mekanisme pengontrol. Dalam praktiknya, sistem kontrol sangat luas digunakan di berbagai bidang teknik seperti otomasi industri, robotika, kendaraan, sistem kelistrikan, hingga peralatan rumah tangga (Suhendar, 2021).

Secara umum, sistem kontrol diklasifikasikan menjadi dua jenis utama, yaitu sistem kontrol terbuka (open-loop) dan sistem kontrol tertutup (closed-loop). Sistem kontrol terbuka bekerja tanpa memantau keluaran; artinya, kontrol diberikan berdasarkan input yang telah ditentukan sebelumnya, tanpa memperhatikan apakah output sesuai atau tidak. Sistem ini lebih sederhana, tetapi kurang akurat ketika menghadapi gangguan dari lingkungan atau variasi beban. Sebaliknya, sistem kontrol tertutup menggunakan mekanisme umpan balik (feedback) untuk mengamati hasil output dan secara otomatis menyesuaikan input agar sistem tetap berjalan sesuai dengan yang diinginkan (Nise, 2011).

Dalam sistem kontrol tertutup, terdapat komponen utama seperti sensor, aktuator, pengendali (kontroller), dan elemen proses. Sensor digunakan untuk mengukur output aktual dari sistem dan memberikan informasi kepada pengendali. Kontroller kemudian membandingkan nilai aktual dengan nilai referensi (setpoint) dan menghitung besarnya koreksi yang perlu diberikan agar error dapat dikurangi atau dieliminasi. Koreksi ini kemudian dikirimkan ke aktuator yang akan mempengaruhi sistem proses secara langsung (Ogata, 2010).

Salah satu tujuan penting dalam merancang sistem kontrol adalah untuk menjamin stabilitas, kecepatan respons, akurasi, dan ketahanan terhadap gangguan. Oleh karena itu, dalam implementasinya, perancangan sistem kontrol harus mempertimbangkan karakteristik dinamis dari sistem yang dikendalikan serta

parameter-parameter yang dapat mempengaruhi kinerjanya, seperti waktu naik (rise time), waktu tunda (delay), overshoot, dan kesalahan keadaan tunak (steady-state error) (Suhendar, 2021).

Penerapan sistem kontrol dapat ditemukan dalam berbagai aplikasi nyata, seperti sistem pendingin ruangan (AC), mesin cuci otomatis, kendaraan otonom, sistem kontrol suhu di laboratorium, dan masih banyak lagi. Dalam konteks pendidikan dan penelitian, sistem kontrol juga dijadikan sebagai dasar pemahaman bagi mahasiswa dalam mempelajari interaksi antara perangkat keras dan perangkat lunak dalam sistem tertanam (embedded system).

3.1.1. Elemen Penting dalam Sistem Kontrol

Sistem kontrol merupakan suatu rangkaian komponen yang saling bekerja sama untuk mengatur atau mengendalikan parameter dari suatu sistem agar sesuai dengan nilai yang diinginkan (set point). Elemen-elemen penting dalam sistem kontrol terdiri dari beberapa komponen utama, yaitu:

1. **Plant (tanaman):** Merupakan bagian dari sistem yang dikendalikan. Misalnya, dalam sistem pengendalian suhu, plant-nya adalah ruangan atau perangkat pemanas itu sendiri.
2. **Sensor:** Berfungsi untuk mengukur keluaran sistem (output) dan mengubahnya menjadi sinyal yang dapat diolah oleh pengendali.
3. **Controller (pengendali):** Berfungsi membandingkan nilai keluaran yang diukur dengan set point, lalu menghasilkan sinyal kendali berdasarkan selisih (error) tersebut.
4. **Actuator (aktuator):** Bertugas mengubah sinyal kendali menjadi aksi nyata yang mempengaruhi plant.
5. **Set point:** Nilai yang diharapkan atau dituju oleh sistem.

Setiap komponen memiliki peran penting untuk memastikan sistem kontrol bekerja secara optimal. Dalam sistem kontrol tertutup, elemen-elemen ini bekerja dalam siklus umpan balik yang memungkinkan sistem untuk beradaptasi terhadap gangguan dan mempertahankan kestabilan (Ogata, 2020).

3.1.2. Representasi dan Analisis Sistem Kontrol

Representasi sistem kontrol diperlukan untuk menganalisis perilaku dinamis sistem dan merancang pengendali yang sesuai. Beberapa bentuk representasi umum yang digunakan dalam sistem kontrol antara lain:

1. Diagram blok: Menunjukkan hubungan antar elemen dalam sistem kontrol secara grafis.
2. Model matematis: Biasanya menggunakan persamaan diferensial untuk menggambarkan dinamika sistem.
3. Fungsi alih (Transfer Function): Merupakan representasi dalam domain Laplace yang menggambarkan hubungan antara input dan output sistem.
4. State Space: Representasi dalam bentuk vektor status yang digunakan untuk sistem multivariat dan sistem orde tinggi.

Analisis dilakukan untuk mengevaluasi stabilitas, respon waktu, dan karakteristik frekuensi dari sistem. Beberapa teknik analisis yang umum digunakan adalah analisis akar-pohon (root locus), diagram Bode, dan diagram Nyquist (Nise, 2020). Analisis ini membantu dalam menentukan parameter kontrol yang optimal agar sistem bekerja sesuai harapan.

3.1.3. Penerapan Sistem Kontrol dalam Kehidupan Nyata

Sistem kontrol memiliki penerapan yang sangat luas dalam berbagai bidang kehidupan sehari-hari, mulai dari industri, transportasi, rumah tangga, hingga peralatan medis. Contohnya meliputi:

1. Sistem kontrol suhu: Digunakan dalam AC, pemanas ruangan, dan inkubator.
2. Kendali kecepatan motor DC: Umum pada robotika, kendaraan listrik, dan peralatan otomasi.
3. Sistem kontrol lalu lintas: Untuk mengatur lampu lalu lintas dan aliran kendaraan di jalan raya.
4. Pengendalian proses industri: Seperti dalam sistem pemurnian minyak, pengolahan air, dan produksi makanan.

Penerapan sistem kontrol memungkinkan efisiensi energi, peningkatan keamanan, serta penghematan biaya operasional dalam jangka panjang. Perkembangan teknologi digital dan mikrokontroler juga mempermudah implementasi sistem kontrol otomatis dengan akurasi tinggi (Astrom & Murray, 2016).

3.1.4. Perkembangan dalam Sistem Kontrol

Sistem kontrol telah mengalami perkembangan pesat seiring kemajuan teknologi digital, komputasi, dan kecerdasan buatan. Awalnya, sistem kontrol dikembangkan secara analog dan manual, namun kini telah bertransformasi menjadi sistem digital dan adaptif.

1. Beberapa perkembangan signifikan meliputi:
2. Kontrol digital: Menggunakan komputer atau mikrokontroler sebagai pengendali utama.
3. Kontrol adaptif: Sistem dapat menyesuaikan parameternya secara otomatis berdasarkan perubahan lingkungan.
4. Kontrol berbasis AI dan machine learning: Digunakan untuk sistem yang kompleks dan tidak linier.
5. Internet of Things (IoT): Memungkinkan sistem kontrol terhubung dengan jaringan untuk monitoring dan pengendalian jarak jauh.

Perkembangan ini meningkatkan fleksibilitas dan efisiensi dari sistem kontrol modern, serta membuka peluang aplikasi baru di berbagai sektor (Karray et al., 2021). Sistem kontrol kini tidak hanya sekadar alat otomatisasi, tetapi juga bagian dari sistem cerdas dalam era industri 4.0.

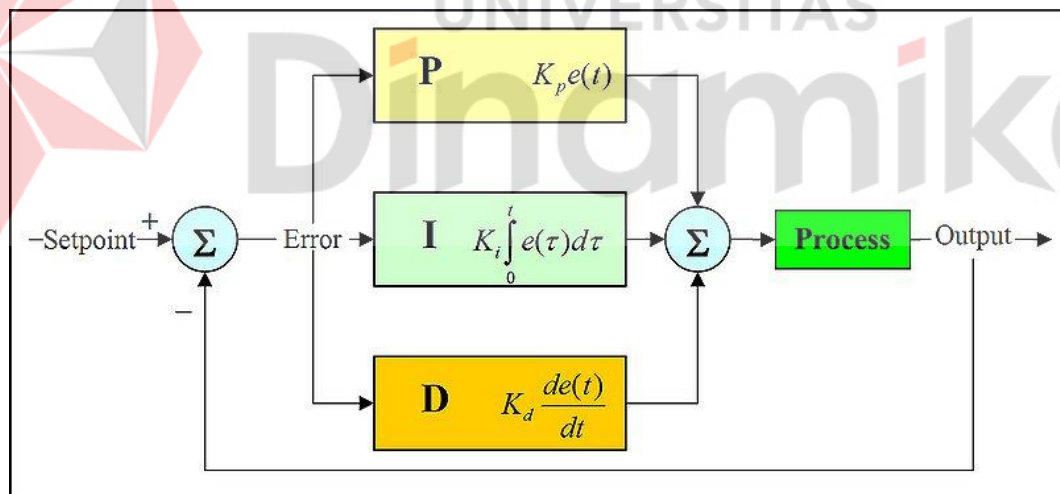
3.2 Pengendali PID

Pengendali PID (Proportional-Integral-Derivative) merupakan salah satu jenis sistem pengendalian umpan balik yang paling umum digunakan dalam dunia otomasi dan kontrol teknik karena kesederhanaannya serta kemampuannya dalam memberikan respons yang stabil dan akurat. PID digunakan untuk mengatur variabel proses tertentu agar tetap berada pada nilai yang diinginkan (setpoint),

meskipun terjadi gangguan atau perubahan dalam sistem (Nise, 2011).

Pengendali PID bekerja dengan menghitung error, yaitu selisih antara nilai setpoint dan nilai aktual (output) dari sistem. Nilai error ini kemudian diolah menggunakan tiga komponen utama: proportional (P), integral (I), dan derivative (D). Komponen proportional bertugas memberikan aksi kontrol berdasarkan besarnya error saat ini. Semakin besar error, semakin besar aksi kontrol yang diberikan. Namun, kendali proporsional saja tidak cukup untuk menghilangkan error secara menyeluruh (Ogata, 2010).

Komponen integral berfungsi untuk mengakumulasi error dari waktu ke waktu. Ini berguna untuk mengeliminasi kesalahan keadaan tunak (steady-state error), yaitu kondisi ketika output tidak pernah mencapai nilai setpoint secara presisi walau error sudah kecil. Sedangkan komponen derivative merespons terhadap laju perubahan error. Ia memperkirakan arah dan kecepatan perubahan error, sehingga dapat mengurangi overshoot dan meningkatkan stabilitas sistem (Kuo & Golnaraghi, 2003).



Gambar 3. 1 Rumus PID

Keunggulan utama dari pengendali PID adalah kemampuannya untuk menyesuaikan terhadap beragam kondisi sistem, baik sistem yang sederhana maupun kompleks. Dengan penyesuaian nilai parameter K_p , K_i dan K_d yang tepat, PID dapat menghasilkan performa kendali yang cepat, stabil, dan akurat. Namun, tuning parameter PID memerlukan pendekatan yang tepat, baik secara manual (trial and error), metode Ziegler-Nichols, atau teknik optimasi berbasis

algoritma (Astrom & Hägglund, 2006).

Penggunaan PID tidak hanya terbatas pada sistem suhu, tetapi juga digunakan luas dalam pengaturan kecepatan motor, kontrol tekanan, ketinggian air, hingga sistem kendali posisi dalam robotika. Di bidang pendidikan dan penelitian, PID juga menjadi titik awal untuk memahami prinsip-prinsip dasar pengendalian otomatis dan desain sistem dinamis (Kuo & Golnaraghi, 2003).

3.2.1. Aplikasikan Kontroller PID dalam Pengendalian Proses

Pengendali Proportional-Integral-Derivative (PID) merupakan jenis kontroler yang paling umum digunakan dalam sistem kontrol industri karena kemampuannya mengatur variabel proses secara otomatis dan efisien. Kontroler ini bekerja dengan menghitung error atau selisih antara nilai aktual dan nilai yang diharapkan, kemudian memberikan sinyal kontrol berdasarkan perhitungan tiga komponen utama, yaitu proporsional (K_p), integral (K_i), dan derivatif (K_d). Dalam aplikasinya, PID banyak digunakan untuk mengendalikan suhu, tekanan, aliran, dan kecepatan motor karena kesederhanaannya serta kemampuannya dalam menjaga kestabilan sistem (Ogata, 2020).

Penggunaan PID dalam sistem berbasis mikrokontroler seperti Arduino juga telah banyak diterapkan untuk mendukung otomatisasi di bidang akademik maupun industri. Pada sistem pengatur suhu, misalnya, nilai suhu yang dibaca oleh sensor akan dibandingkan dengan suhu target (setpoint), kemudian PID menentukan seberapa besar daya yang perlu diberikan ke elemen pemanas atau pendingin untuk menjaga kestabilan suhu (Åström & Murray, 2016). Fleksibilitas ini menjadikan PID sebagai fondasi dalam banyak sistem kontrol modern karena dapat diimplementasikan pada sistem analog maupun digital secara efisien.

3.2.2. Keuntungan Kontroller PID

Salah satu keuntungan utama dari kontroler PID adalah kemampuannya menghasilkan kestabilan sistem yang tinggi meskipun struktur algoritmanya cukup sederhana. PID dapat diterapkan pada berbagai jenis sistem tanpa memerlukan model matematis yang kompleks, sehingga cocok untuk sistem dinamis yang sulit dimodelkan secara eksak (Nise, 2020). Hal ini menjadikan PID sebagai solusi praktis dalam sistem yang hanya memerlukan kestabilan dan presisi dasar, seperti dalam proses manufaktur, pengendalian robotik, hingga otomasi rumah tangga.

Selain itu, kemampuan PID dalam mengurangi error steady-state (melalui komponen integral) dan mengantisipasi perubahan mendadak (melalui komponen derivatif) memungkinkan sistem mencapai respons yang cepat dan minim overshoot. Tuning parameter yang tepat dapat mempercepat waktu respon sistem dan memperkecil fluktuasi (Zhou et al., 2019). Tidak hanya itu, karena bersifat modular, kontroler PID dapat dengan mudah dikombinasikan dengan teknik kontrol lainnya, seperti fuzzy logic atau neural network, untuk menghasilkan sistem kendali hibrida yang lebih cerdas dan adaptif (Karray et al., 2021).

3.2.3. Tantangan dalam Penggunaan Kontroller PID

Walaupun kontroler PID menawarkan banyak keuntungan, penggunaannya juga dihadapkan pada sejumlah tantangan teknis. Tantangan utama dalam penggunaan PID adalah proses tuning parameter K_p , K_i , dan K_d yang tidak selalu mudah dilakukan, terutama pada sistem yang tidak linier atau memiliki karakteristik yang berubah seiring waktu. Kesalahan dalam tuning dapat menyebabkan sistem menjadi terlalu lambat, mengalami overshoot berlebihan, atau bahkan menjadi tidak stabil (Khosrowjerdi & Chen, 2020).

Selain itu, komponen derivatif dalam PID sangat sensitif terhadap noise, karena sifatnya yang merespons perubahan sinyal secara cepat. Dalam sistem nyata, noise dari sensor atau gangguan eksternal bisa memicu sinyal derivatif untuk memberikan respons yang salah, sehingga diperlukan tambahan filter atau peredam untuk menghindari osilasi berlebih. Tantangan lainnya adalah keterbatasan PID dalam menangani proses dengan keterlambatan waktu (time-delay) atau sistem multivariat yang saling memengaruhi. Untuk mengatasi hal ini, para peneliti dan insinyur mulai mengembangkan variasi PID yang lebih adaptif atau menggabungkannya dengan algoritma kecerdasan buatan agar sistem tetap responsif dalam berbagai kondisi (Zhou et al., 2019; Karray et al., 2021).

3.3 Arduino UNO



Gambar 3. 2 Arduino UNO

Arduino UNO merupakan salah satu papan mikrokontroler yang paling populer dan banyak digunakan dalam bidang elektronika, otomasi, dan pendidikan karena kemudahan penggunaannya serta fleksibilitasnya dalam berbagai aplikasi. Papan ini dirancang berbasis mikrokontroler ATmega328P yang memiliki kemampuan pemrosesan cukup untuk mendukung proyek-proyek berbasis sistem tertanam atau sistem kendali otomatis (Banzi & Shiloh, 2014).

Arduino UNO memiliki 14 pin input/output digital (di mana 6 di antaranya dapat digunakan sebagai output PWM), 6 input analog, serta konektivitas USB yang memudahkan pemrograman dari komputer. Selain itu, papan ini mendukung komunikasi serial, I2C, dan SPI, yang memungkinkan komunikasi dengan berbagai perangkat eksternal seperti sensor, motor, layar LCD, dan modul lainnya (Monk, 2016). Ketersediaan voltage regulator, oscillator internal, serta antarmuka komunikasi membuat papan ini dapat berdiri sendiri sebagai otak dari suatu sistem otomasi, tanpa memerlukan rangkaian tambahan yang rumit.

Salah satu alasan Arduino UNO menjadi sangat populer adalah karena keberadaan Arduino IDE (Integrated Development Environment), yaitu perangkat lunak lintas platform yang sederhana namun powerful untuk menulis, mengunggah, dan memonitor program dalam bahasa pemrograman C/C++ yang sudah dimodifikasi untuk lebih mudah dipahami. Lingkungan pengembangan ini dilengkapi dengan pustaka (library) yang luas serta komunitas daring yang aktif, sehingga pengguna dapat dengan cepat mencari referensi dan solusi (Margolis,

2011).

Keunggulan lainnya adalah sifatnya sebagai perangkat open-source, baik dari sisi perangkat keras maupun perangkat lunak. Artinya, siapa pun dapat memodifikasi, mengembangkan, dan mendistribusikan ulang desain Arduino. Hal ini mendorong terciptanya banyak versi turunan dan modul-modul pendukung yang kompatibel, sehingga memperluas aplikasi Arduino dalam berbagai bidang seperti sistem kontrol, Internet of Things (IoT), robotika, hingga sistem monitoring berbasis sensor (Banzi & Shiloh, 2014).

Dalam konteks sistem kontrol berbasis PID, Arduino UNO sangat ideal digunakan sebagai pengendali pusat karena mampu membaca input analog dari sensor seperti LM35, melakukan perhitungan logika PID, dan mengendalikan aktuator seperti motor DC secara presisi. Dengan kemampuan pemrosesan real-time dan latensi yang rendah, Arduino UNO memungkinkan pengaturan sistem kontrol tertutup (closed-loop) yang responsif dan stabil (Monk, 2016).

3.4 Sensor LM35



Gambar 3. 3 Sensor LM35

Sensor LM35 merupakan salah satu sensor suhu analog yang umum digunakan dalam berbagai aplikasi elektronik dan sistem kontrol karena akurasi tinggi, harga terjangkau, dan kemudahan integrasi dengan mikrokontroler seperti Arduino. Sensor ini dikembangkan oleh Texas Instruments dan dirancang untuk menghasilkan output tegangan linear yang proporsional terhadap suhu dalam satuan derajat Celcius, tanpa memerlukan kalibrasi eksternal yang kompleks (Texas Instruments, 2000).

LM35 bekerja berdasarkan prinsip termal semikonduktor di mana

perubahan suhu akan memengaruhi tegangan keluaran sensor. Output dari LM35 sebesar 10 mV untuk setiap derajat Celcius. Misalnya, pada suhu 25 °C, sensor akan memberikan tegangan output sebesar 250 mV. Rentang pengukuran suhu yang dapat dideteksi oleh sensor ini umumnya berkisar antara -55 °C hingga +150 °C, dengan akurasi sekitar $\pm 0,5$ °C pada suhu kamar, menjadikannya pilihan yang andal untuk sistem pemantauan suhu presisi rendah hingga menengah (Khandpur, 2004).

Salah satu keunggulan LM35 dibandingkan sensor suhu lainnya seperti termistor adalah kestabilan output yang tidak terpengaruh oleh perubahan resistansi, karena LM35 menghasilkan tegangan secara langsung. Sensor ini juga memiliki konsumsi daya yang rendah, yaitu sekitar 60 μ A, sehingga cocok untuk sistem dengan kebutuhan energi rendah seperti perangkat portabel atau sistem IoT berbasis baterai (Boylestad & Nashelsky, 2012).

Dalam praktiknya, LM35 dihubungkan ke pin analog pada mikrokontroler (misalnya pin A0 pada Arduino UNO). Nilai analog yang terbaca kemudian dikonversi menjadi suhu melalui perhitungan matematis. Karena ADC (Analog to Digital Converter) pada Arduino memiliki resolusi 10-bit, nilai input analog dari LM35 dapat dikonversi menjadi tegangan, dan selanjutnya ke suhu dalam Celcius. Misalnya, jika tegangan referensi adalah 5 volt, maka tiap unit ADC mewakili 4,88 mV, sehingga hasil pembacaan perlu dikalikan dengan faktor tertentu untuk mendapatkan hasil dalam °C (Monk, 2016).

LM35 juga tersedia dalam beberapa bentuk kemasan seperti TO-92 (mirip transistor), SO-8, dan TO-220, yang dapat disesuaikan dengan kebutuhan penggunaan di breadboard, PCB, atau sistem pendingin. Sensor ini banyak digunakan dalam sistem pendingin otomatis, pengendali suhu ruangan, inkubator, serta aplikasi yang membutuhkan pemantauan suhu lingkungan atau benda tertentu secara akurat.

Dengan kestabilan, akurasi, dan antarmuka yang sederhana, LM35 menjadi salah satu sensor suhu pilihan utama dalam proyek-proyek pengendalian suhu berbasis PID, di mana informasi suhu menjadi input penting dalam menentukan respons aktuator seperti kipas atau motor untuk menjaga suhu dalam batas yang diinginkan (Texas Instruments, 2000).

3.5 Keypad 4x4



Gambar 3. 4 Keypad 4x4

Keypad 4x4 adalah sebuah perangkat input berbentuk matriks tombol yang terdiri dari 16 tombol yang tersusun dalam 4 baris dan 4 kolom. Komponen ini umum digunakan dalam sistem elektronik interaktif seperti kalkulator, sistem keamanan, antarmuka menu perangkat, dan sistem kendali berbasis mikrokontroler. Keypad ini memberikan solusi praktis dan ekonomis untuk memasukkan data secara manual ke dalam sistem digital (Monk, 2013).

Secara teknis, keypad 4x4 bekerja dengan prinsip penggabungan baris dan kolom. Ketika sebuah tombol ditekan, sambungan antara satu baris dan satu kolom akan terjadi, menciptakan titik hubung yang dapat dideteksi oleh mikrokontroler. Sistem kemudian membaca posisi tombol yang ditekan berdasarkan kombinasi koordinat baris dan kolom tersebut. Untuk mengidentifikasi tombol yang aktif, mikrokontroler mengaktifkan satu per satu jalur baris secara bergantian sambil membaca sinyal dari jalur kolom. Proses ini dikenal sebagai scanning matrix, yang sangat efisien dan memungkinkan deteksi hingga 16 tombol hanya dengan menggunakan delapan pin I/O (Alciatore & Histan, 2011).

Keypad 4x4 biasanya terbuat dari lapisan plastik fleksibel dengan sirkuit internal yang tercetak menggunakan konduktor karbon atau logam tipis. Material ini memungkinkan tombol untuk ditekan berkali-kali tanpa cepat aus. Selain itu, keypad ini juga tersedia dalam bentuk membran fleksibel maupun versi PCB keras, tergantung pada kebutuhan aplikasi dan lingkungan penggunaannya.

Penggunaan keypad pada sistem mikrokontroler seperti Arduino sangat populer karena kemudahannya dalam pemrograman serta dukungan pustaka (library) seperti Keypad.h yang menyederhanakan proses pemindaian tombol dan interpretasi input. Dengan library ini, pengguna cukup mendefinisikan susunan tombol dan pin yang digunakan, lalu fungsi pemrosesan input dapat dijalankan secara otomatis tanpa harus memprogram logika pemindaian secara manual (Hughes, 2020).

Dalam konteks sistem kendali berbasis PID atau sistem otomasi lainnya, keypad 4x4 dapat digunakan untuk memasukkan nilai parameter seperti konstanta PID (KP, KI, KD), setpoint, atau bahkan sebagai tombol kontrol untuk memulai dan menghentikan sistem. Kelebihan ini menjadikan keypad sebagai antarmuka manusia-mesin (HMI) yang penting untuk interaksi langsung antara pengguna dan perangkat.

Dari sisi konektivitas, untuk menghubungkan keypad 4x4 ke mikrokontroler, delapan pin digital diperlukan, yaitu empat untuk baris dan empat untuk kolom. Namun, untuk menghemat pin pada proyek skala besar, solusi seperti penggunaan multiplexer atau I2C expander juga bisa diterapkan. Selain itu, keypad dapat diintegrasikan dengan sistem tampilan seperti LCD untuk menciptakan antarmuka pengguna yang lebih intuitif dan informatif.

Dengan fleksibilitas, kemudahan integrasi, serta ketersediaan luas di pasaran, keypad 4x4 tetap menjadi pilihan ideal dalam pengembangan sistem kontrol dan perangkat elektronik yang membutuhkan masukan numerik atau perintah sederhana secara langsung dari pengguna (Monk, 2013).

3.6 LCD (Liquid Crystal Display) 16X2



Gambar 3. 5 LCD 16X2

LCD 16x2 adalah jenis modul tampilan karakter yang terdiri atas dua baris (rows), masing-masing dengan enam belas kolom (columns), sehingga dapat menampilkan hingga 32 karakter dalam satu waktu. Modul ini termasuk dalam kategori LCD (Liquid Crystal Display) alfanumerik, yang secara luas digunakan dalam berbagai aplikasi mikrokontroler karena konsumsi daya yang rendah, bentuk yang ringkas, dan kemudahan dalam pemrograman. LCD 16x2 banyak digunakan dalam proyek-proyek elektronik, sistem kontrol, alat ukur digital, serta perangkat dengan antarmuka pengguna sederhana (Mazidi, Naimi, & Naimi, 2016).

Secara fisik, LCD ini dilengkapi dengan 16 pin, walaupun beberapa versi juga dilengkapi dengan antarmuka I2C (Inter-Integrated Circuit) untuk menghemat penggunaan pin input/output (I/O) pada mikrokontroler. Dalam versi standar paralel, LCD ini memerlukan setidaknya enam pin data dan kontrol untuk komunikasi, yaitu RS (Register Select), E (Enable), dan empat pin data (D4-D7) dalam mode 4-bit. Jika menggunakan mode 8-bit, maka akan memerlukan delapan pin data (D0-D7) yang tentunya akan menyita lebih banyak jalur I/O dari mikrokontroler. Oleh karena itu, mode 4-bit dan antarmuka I2C lebih sering digunakan untuk efisiensi (Simon & Monk, 2012).

Prinsip kerja dari LCD 16x2 melibatkan penggunaan kristal cair yang bereaksi terhadap sinyal listrik untuk memodulasi cahaya. Ketika sinyal listrik diberikan ke salah satu segmen, molekul cair di dalam kristal akan berubah

orientasi, sehingga mengubah tingkat transmisi cahaya. Bagian belakang LCD biasanya dilengkapi dengan backlight LED yang memungkinkan tampilan tetap terlihat jelas dalam berbagai kondisi pencahayaan. Sementara itu, lapisan polarisator dan elektroda transparan membantu menciptakan kontras antara karakter dan latar belakang (Displaytech, 2019).

Karakter yang ditampilkan pada LCD 16x2 berasal dari memori karakter internal, yaitu Character Generator ROM (CGROM) yang menyimpan pola piksel dari karakter ASCII standar. Modul ini juga menyediakan Character Generator RAM (CGRAM) yang memungkinkan pengguna untuk mendefinisikan karakter kustom sesuai kebutuhan aplikasi. Hal ini sangat berguna dalam sistem kontrol atau antarmuka pengguna yang memerlukan simbol khusus (Ibrahim, 2011).

LCD 16x2 dikendalikan menggunakan instruksi yang dikirim secara serial atau paralel melalui pin data, seperti perintah untuk mengatur posisi kursor, membersihkan layar, menampilkan karakter, dan menggulir tampilan. Dengan dukungan pustaka pemrograman seperti LiquidCrystal.h (untuk koneksi paralel) atau LiquidCrystal_I2C.h (untuk koneksi I2C) pada platform seperti Arduino, pengguna dapat dengan mudah mengintegrasikan LCD ke dalam sistem kendali atau tampilan antarmuka (Hughes, 2020).

Dalam konteks sistem kontrol berbasis mikrokontroler dan PID, LCD 16x2 berfungsi sebagai antarmuka visual untuk menampilkan informasi penting seperti nilai suhu aktual, setpoint, status kendali, serta nilai parameter PID (K_p , K_i , dan K_d). Keberadaan tampilan ini sangat membantu pengguna dalam memantau dan menyesuaikan sistem secara langsung tanpa memerlukan perangkat tambahan seperti komputer.

Dengan kemudahan integrasi, kompatibilitas yang luas dengan mikrokontroler seperti Arduino, serta dukungan ekosistem pustaka yang kaya, LCD 16x2 tetap menjadi pilihan utama dalam berbagai proyek elektronika dan sistem kontrol skala kecil hingga menengah (Mazidi et al., 2016).

3.7 Motor DC JGA25-370



Gambar 3. 6 Motor DC JGA25-370

Motor DC JGA25-370 adalah jenis motor listrik arus searah yang dilengkapi dengan gearbox (gear reduction) bawaan, sehingga dapat menghasilkan torsi yang lebih besar dengan kecepatan putaran yang lebih rendah. Motor ini dirancang khusus untuk aplikasi yang membutuhkan presisi pergerakan dan tenaga mekanik tinggi pada kecepatan rendah, seperti pada sistem kontrol otomatis, robotika, lengan robot, penggerak roda, serta aplikasi mekatronika lainnya (Pololu, 2023).

Secara struktural, motor ini terdiri atas dua bagian utama, yaitu motor inti berjenis brushed DC dan unit gearhead berbentuk silinder yang terpasang pada poros output-nya. Motor DC menghasilkan putaran melalui interaksi medan magnet antara kumparan armature dan magnet permanen. Arus yang mengalir melalui sikat (brush) menuju komutator akan mengaktifkan medan magnet yang membuat rotor berputar (Hughes & Drury, 2013). Dalam hal ini, gearbox memainkan peran penting untuk menurunkan kecepatan putaran dan sekaligus meningkatkan torsi yang dihasilkan oleh motor. Rasio gearbox yang umum pada JGA25-370 bervariasi, misalnya 1:30, 1:50, hingga 1:150, tergantung pada kebutuhan aplikasi.

Motor JGA25-370 umumnya bekerja pada tegangan antara 6 hingga 12 volt DC, dengan arus kerja normal berkisar antara 100 hingga 400 mA, namun dapat meningkat saat motor dalam kondisi beban maksimum (stall current). Torsi yang dihasilkan bisa mencapai beberapa kilogram sentimeter (kg.cm), tergantung pada rasio gear yang digunakan. Dalam sistem kontrol seperti pengendali PID berbasis

Arduino, motor ini dapat digunakan sebagai aktuator yang menggerakkan sistem fisik, seperti kipas pendingin atau katup pengatur suhu (Bulan & Setiawan, 2021).

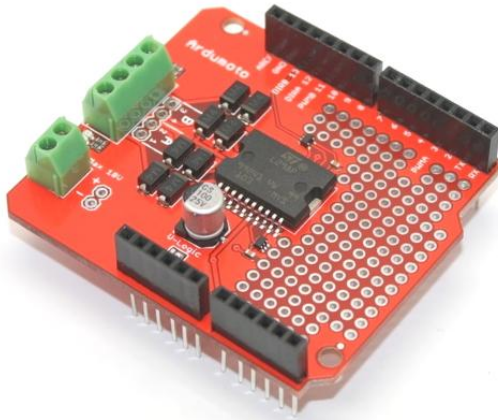
Keunggulan utama dari JGA25-370 dibandingkan motor DC biasa tanpa gearbox adalah kemampuannya dalam mempertahankan kekuatan torsi pada kecepatan rendah, yang sangat dibutuhkan dalam aplikasi kendali presisi. Misalnya, dalam sistem pengaturan suhu berbasis sensor dan motor, motor ini dapat digunakan untuk mengontrol kecepatan kipas secara akurat sesuai dengan sinyal keluaran dari kendali PID. Selain itu, motor ini juga memiliki daya tahan yang tinggi, umur pakai yang panjang, serta kemampuan beroperasi secara stabil dalam waktu lama (Dong, 2020).

Dalam implementasinya, motor ini biasanya dikendalikan menggunakan driver motor seperti Ardumoto, L298N, atau driver berbasis H-Bridge lainnya. Driver tersebut berfungsi mengatur arah dan kecepatan putaran motor berdasarkan sinyal PWM (Pulse Width Modulation) dari mikrokontroler. Dengan kontrol yang tepat, JGA25-370 mampu memberikan respons yang cepat dan akurat terhadap perubahan nilai setpoint atau gangguan dalam sistem kendali (Pololu, 2023).

Motor ini juga kompatibel dengan encoder tambahan, yang memungkinkan pembacaan posisi atau kecepatan secara lebih presisi, menjadikannya ideal untuk aplikasi closed-loop control. Encoder dapat digunakan untuk memberi umpan balik ke sistem PID agar pengendalian menjadi lebih adaptif dan stabil, terutama dalam sistem kendali suhu berbasis Arduino yang membutuhkan akurasi dalam pengaturan aktuator.

Secara keseluruhan, Motor DC JGA25-370 merupakan komponen vital dalam berbagai proyek teknik dan kontrol otomatis karena kemampuannya menghasilkan torsi besar, fleksibilitas dalam pemakaian, serta kemudahan dalam integrasi dengan sistem mikrokontroler.

3.8 Driver Motor Ardumoto



Gambar 3. 7 Driver Motor Ardumoto

Driver motor Ardumoto merupakan modul pengendali motor DC berbasis IC L298P yang didesain khusus untuk bekerja secara langsung dengan papan mikrokontroler seperti Arduino UNO. Modul ini memiliki bentuk fisik dan tata letak pin yang kompatibel dengan header Arduino, sehingga dapat dipasangkan langsung tanpa kabel tambahan, menjadikannya pilihan yang efisien dan praktis untuk berbagai proyek kendali motor (SparkFun, 2020).

Ardumoto dapat mengendalikan dua buah motor DC secara independen dalam mode arah maju dan mundur (bidirectional) menggunakan prinsip pengendalian H-Bridge. Setiap saluran motor pada modul ini mampu menangani arus hingga sekitar 2 ampere dalam jangka pendek, serta mampu bekerja dengan tegangan input motor hingga 12 volt DC. Selain itu, pengguna dapat mengatur kecepatan putaran motor dengan memberikan sinyal PWM (Pulse Width Modulation) dari mikrokontroler ke pin kendali pada modul ini (Monk, 2013).

IC L298P yang menjadi inti dari Ardumoto bekerja sebagai pengendali daya (power driver) yang mampu memperkuat sinyal logika dari mikrokontroler agar dapat menggerakkan beban motor. Sinyal PWM yang dikirimkan ke input enable akan menentukan seberapa besar duty cycle, sehingga mempengaruhi kecepatan motor. Sementara itu, dua input digital lainnya digunakan untuk menentukan arah putaran motor, dengan konfigurasi logika tertentu (Kurniawan, 2020). Dengan

demikian, pengguna memiliki kontrol penuh terhadap arah dan kecepatan motor hanya dengan beberapa pin digital dari Arduino.

Modul Ardumoto juga dilengkapi dengan indikator LED yang menunjukkan status dari masing-masing motor saat aktif, serta terminal screw atau header yang memudahkan koneksi ke motor DC. Kemudahan integrasi ini menjadikan Ardumoto sangat cocok digunakan pada sistem kontrol berbasis Arduino seperti robot bergerak, sistem konveyor, atau kipas pendingin otomatis yang dikendalikan dengan logika tertentu, misalnya berdasarkan sensor suhu (SparkFun, 2020).

Dari sisi implementasi dalam sistem kendali PID, Ardumoto berperan sebagai aktuator yang menerima sinyal hasil pengolahan dari kendali PID untuk mengatur kecepatan dan arah motor sesuai kebutuhan sistem. Misalnya, pada sistem pengatur suhu berbasis PID, perubahan suhu yang terdeteksi oleh sensor akan menghasilkan keluaran PID yang kemudian diterjemahkan menjadi sinyal PWM. Sinyal ini selanjutnya dikirimkan ke Ardumoto untuk mengatur kecepatan motor kipas secara otomatis, menjaga suhu pada titik setpoint yang diinginkan (Nugroho & Prasetyo, 2022).

Keunggulan Ardumoto dibandingkan dengan driver motor lainnya terletak pada desainnya yang ringkas, pemasangan yang sederhana, serta kemampuannya untuk bekerja dengan berbagai jenis motor DC kecil hingga menengah. Selain itu, modul ini juga relatif hemat energi, dan memiliki efisiensi daya yang cukup baik untuk penggunaan di bidang edukasi, prototipe industri, serta sistem kendali mekatronika berbasis mikrokontroler.

Secara keseluruhan, driver motor Ardumoto memainkan peran penting dalam perancangan sistem pengendalian otomatis, terutama pada sistem yang melibatkan motor sebagai komponen aktuator utama. Dengan dukungan pengendalian PWM dan kontrol arah, modul ini menjadi pilihan yang andal untuk proyek-proyek teknik berbasis Arduino.

BAB IV

PEMBAHASAN

4.1 Prinsip Kerja

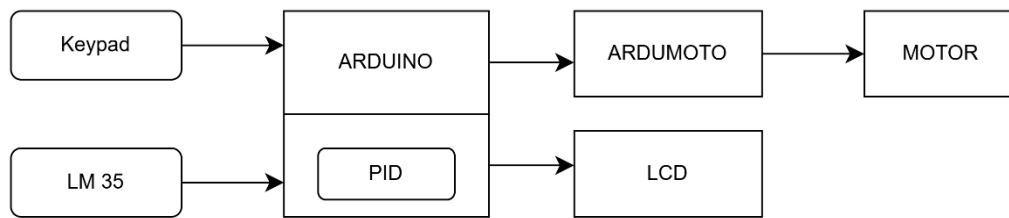
Dalam proses pelaksanaan kerja praktik yang dilaksanakan di Laboratorium Internet of Things (IoT) Program Studi S1 Teknik Komputer Universitas Dinamika Surabaya, mahasiswa diwajibkan menyelesaikan total waktu kerja praktik selama 160 jam. Kegiatan ini merupakan bagian dari kurikulum pendidikan yang bertujuan untuk menjembatani antara teori yang diperoleh selama perkuliahan dengan praktik langsung di lapangan, sehingga mahasiswa memiliki kesempatan untuk mengaplikasikan pengetahuan dan keterampilan yang telah dipelajari dalam suasana kerja nyata.

Pelaksanaan kerja praktik di laboratorium ini memberikan simulasi lingkungan kerja yang menyerupai kondisi di industri maupun instansi profesional, dengan memanfaatkan berbagai perangkat dan teknologi berbasis Internet of Things (IoT). Dengan demikian, mahasiswa tidak hanya dituntut untuk menyelesaikan tugas teknis, tetapi juga diajak untuk mengembangkan kemampuan problem solving, kerja sama tim, manajemen waktu, serta komunikasi profesional semua kemampuan yang sangat dibutuhkan di dunia kerja.

Lebih dari sekadar memenuhi kewajiban akademik, kegiatan kerja praktik ini memberikan manfaat yang signifikan bagi mahasiswa, antara lain sebagai sarana untuk menggali pengalaman, memperluas wawasan, serta memahami dinamika dan tantangan yang dihadapi dalam dunia kerja yang sesungguhnya. Mahasiswa juga memiliki kesempatan untuk melakukan pengumpulan data, melakukan analisis, serta memperoleh informasi dan pengetahuan baru yang tidak didapatkan sepenuhnya di bangku kuliah.

Dengan adanya kegiatan kerja praktik ini, diharapkan mahasiswa dapat mempersiapkan diri secara lebih matang dalam menghadapi persaingan kerja di masa depan, serta mampu beradaptasi dengan perkembangan teknologi dan kebutuhan industri yang terus berkembang. Selain itu, pengalaman ini juga menjadi bekal penting dalam membentuk sikap profesionalisme, tanggung jawab, dan etika kerja yang baik sebagai calon lulusan Teknik Komputer yang kompeten.

4.2 Diagram Blok



Gambar 4. 1 Diagram Blok

Diagram blok di atas menggambarkan alur kerja dari sistem pengendalian suhu menggunakan Arduino yang dikombinasikan dengan metode PID (Proportional-Integral-Derivative). Sistem ini dirancang untuk mengontrol kecepatan motor berdasarkan suhu yang terbaca, dengan masukan nilai set point dari pengguna melalui keypad. Berikut ini penjelasan masing-masing blok:

1. Keypad

Keypad berfungsi sebagai alat input utama yang digunakan oleh pengguna untuk memasukkan nilai set point suhu yang diinginkan. Nilai set point ini akan menjadi referensi bagi sistem untuk menjaga suhu aktual tetap berada pada nilai yang telah ditentukan.

2. Sensor Suhu LM35

Sensor LM35 berperan sebagai alat pendeteksi suhu lingkungan secara real-time. Sensor ini akan mengubah suhu menjadi sinyal analog yang dibaca oleh Arduino. Hasil pembacaan suhu ini kemudian digunakan sebagai umpan balik (feedback) dalam sistem kontrol PID untuk dibandingkan dengan nilai set point.

3. Arduino

Arduino UNO merupakan otak utama sistem. Arduino membaca masukan dari keypad dan sensor LM35, kemudian memproses data tersebut menggunakan algoritma PID. Arduino akan menghitung error antara suhu aktual dan suhu target, lalu menghasilkan sinyal kendali untuk mengatur kecepatan motor melalui driver motor Ardumoto. Selain itu, Arduino juga menampilkan data suhu dan status sistem melalui LCD.

4. Modul PID

Blok PID merupakan bagian dari program pada Arduino yang

mengimplementasikan kontroler PID. Algoritma ini menghitung tiga komponen utama:

- Proportional (K_p): Mengoreksi error berdasarkan besarnya selisih antara suhu aktual dan suhu target.
- Integral (K_i): Mengatasi error akumulatif yang terjadi secara terus-menerus.
- Derivative (K_d): Mengantisipasi perubahan suhu yang mendadak dan memperhalus respon sistem.

Hasil kalkulasi PID ini menentukan seberapa besar sinyal kendali yang harus dikirim ke aktuator (motor) untuk menyesuaikan dengan suhu yang diinginkan.

5. LCD (Liquid Crystal Display)

LCD digunakan untuk menampilkan informasi penting seperti:

- Suhu aktual yang terbaca dari sensor LM35
- Nilai set point yang dimasukkan dari keypad
- Status kendali PID (misalnya output kendali)

LCD berfungsi sebagai interface pengguna, agar pengguna dapat memantau sistem secara langsung.

6. Driver Motor Ardumoto

Ardumoto adalah modul driver motor yang menerima sinyal PWM dari Arduino dan meneruskannya ke motor DC. Ardumoto bertindak sebagai perantara antara mikrokontroler dan motor, dengan fungsi mengatur arah putaran dan kecepatan motor sesuai sinyal kendali dari PID.

7. Motor

Motor dalam sistem ini dikendalikan berdasarkan hasil kalkulasi PID. Kecepatan motor dapat diatur secara proporsional terhadap suhu yang terdeteksi, misalnya untuk sistem ventilasi atau pendinginan otomatis. Motor akan mempercepat atau memperlambat gerakannya untuk menjaga suhu tetap berada di sekitar set point.

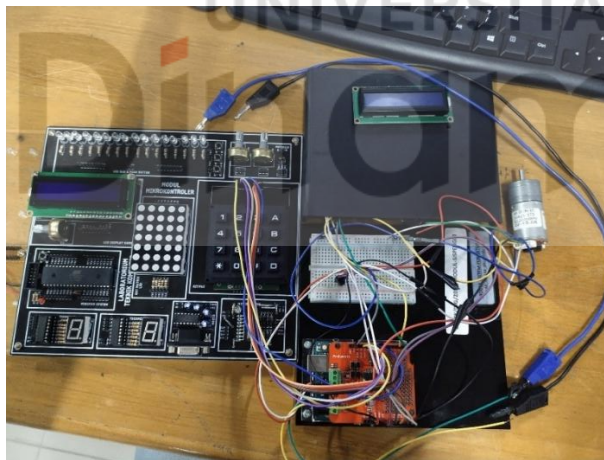
4.3 Perancangan Sistem

Pada Bagian ini membahas proses pengembangan sistem dari dua sisi utama, yaitu perangkat keras (hardware) dan perangkat lunak (software). Tujuan

dari perancangan ini adalah untuk menjamin integrasi yang efektif antara komponen elektronik dan logika pemrograman dalam sistem pengaturan kecepatan motor berbasis suhu dengan menggunakan kendali PID.

Perancangan sistem menjadi tahap yang sangat penting dalam pengembangan modul pembelajaran kontrol PID, karena merupakan dasar utama dalam membangun arsitektur sistem kendali tertutup berbasis mikrokontroler. Sistem ini dirancang agar dapat berfungsi secara mandiri dan mudah digunakan melalui antarmuka input dan output yang sederhana. Selain itu, sistem ini juga dirancang agar dapat diintegrasikan ke dalam kegiatan praktik di laboratorium pendidikan tinggi. Secara umum, proses perancangan sistem ini dibagi menjadi dua bagian utama, yaitu perancangan perangkat keras yang mencakup integrasi komponen fisik, serta perancangan perangkat lunak yang mengatur logika pengendalian dan komunikasi antar komponen. Kedua bagian ini saling melengkapi untuk memastikan sistem dapat bekerja secara optimal, responsif, dan real-time.

4.3.1 Perancangan Hardware



Gambar 4. 2 Rangkaian Alat

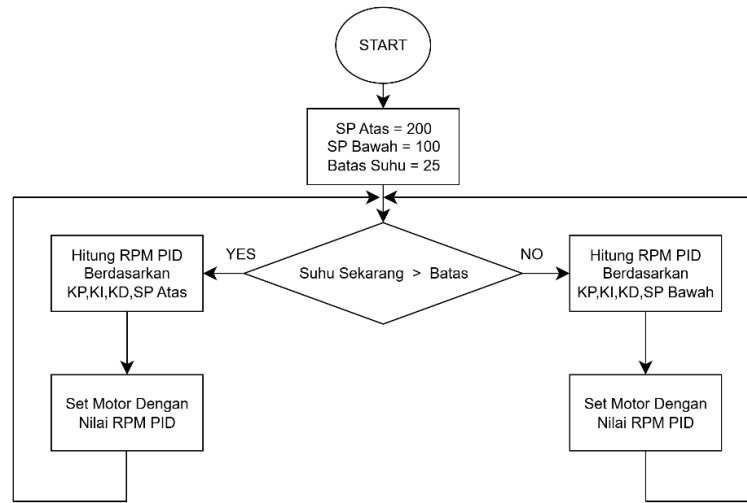
Gambar 4.2 menunjukkan rancangan perangkat keras sistem kontrol suhu berbasis PID yang terdiri dari beberapa komponen inti. Arduino UNO berfungsi sebagai pusat kendali sistem yang memproses data masukan dan menghasilkan sinyal keluaran. Sensor suhu LM35 digunakan untuk mendeteksi suhu lingkungan secara terus-menerus dan mengirimkan data suhu ke mikrokontroler. Keypad 4x4 berperan sebagai alat input untuk memasukkan nilai parameter PID seperti K_p , K_i , dan K_d . Tampilan LCD 16x2 digunakan untuk menampilkan informasi penting

seperti suhu, parameter PID, dan kecepatan putar motor (RPM). Untuk mengendalikan motor DC JGA25-370, digunakan driver motor Ardumoto yang menerima sinyal PWM dari Arduino dan mengatur kecepatan putaran motor berdasarkan hasil perhitungan PID.

Tabel 4. 1 Tabel Alamat PIN

Komponen	Pin Arduino	Deskripsi
Keypad 4x4	RowPins {8, A3, A2, A1}	Pin baris untuk membaca input tombol
	ColPins {12, 11, 10, 9}	Pin kolom untuk membaca input tombol
LCD I2C 16x2	VCC (5V)	Tegangan untuk menghidupkan modul LCD
	GND (GND)	Ground (negatif)
	SCL (A5)	Serial Clock Line untuk komunikasi I2C
	SDA (A4)	Serial Data Line untuk komunikasi I2C
Sensor LM35	A0	Input analog untuk membaca nilai suhu dari sensor
Encoder Motor	encoderPin (2)	Input interrupt untuk membaca pulsa dari encoder motor
Motor Driver	motorPin (3)	Output PWM untuk mengatur kecepatan motor DC

4.3.2 Perancangan software



Gambar 4. 3 Flowchart Cara Kerja Alat

Diagram alir pada Gambar 4.3 menjelaskan logika kerja sistem pengaturan kecepatan motor berbasis kontroler PID, dengan referensi suhu dari sensor LM35 sebagai pemicu perubahan setpoint. Sistem diawali dengan inisialisasi nilai Set Point (SP) atas sebesar 200 RPM, SP bawah sebesar 100 RPM, dan batas suhu sebesar 25°C. Nilai batas suhu ini digunakan sebagai kriteria logika dalam pengambilan keputusan sistem.

Setelah inisialisasi, sistem akan membaca suhu lingkungan secara berkala menggunakan sensor LM35. Suhu yang terbaca tidak digunakan secara langsung dalam perhitungan PID, melainkan hanya berfungsi sebagai **pemicu perubahan setpoint**. Artinya, suhu berperan sebagai *pemindah kondisi* sistem.

Jika suhu saat ini lebih besar dari nilai batas (misalnya suhu > 25°C), maka sistem akan menetapkan SP atas (200 RPM) sebagai target kecepatan motor. Sebaliknya, jika suhu sama dengan atau lebih kecil dari batas, maka sistem menggunakan SP bawah (100 RPM). Setelah penentuan setpoint, sistem akan menjalankan algoritma PID untuk menghitung nilai keluaran (output) berdasarkan selisih antara setpoint RPM dan RPM aktual dari motor yang dihitung melalui encoder.

Hasil perhitungan PID ini kemudian digunakan untuk mengatur sinyal PWM yang diberikan ke motor, sehingga motor dapat menyesuaikan kecepatan

sesuai dengan target RPM. Proses ini terus berjalan secara real-time, dengan PID yang aktif menyesuaikan kecepatan berdasarkan error RPM, sementara suhu hanya mempengaruhi nilai target (setpoint) secara tidak langsung.

Pendekatan ini efektif untuk kasus di mana suhu lingkungan memengaruhi kebutuhan kecepatan motor, tanpa melibatkan suhu secara langsung dalam kontrol PID. Dengan memisahkan logika penentuan setpoint dan pengendalian PID, sistem menjadi lebih modular, stabil, dan mudah disesuaikan.

4.4 Implementasi dan Pengujian

Tahap implementasi dan pengujian merupakan proses lanjutan setelah seluruh komponen sistem dirancang dan dikembangkan, baik dari sisi perangkat keras (hardware) maupun perangkat lunak (software). Sistem ini menggabungkan sejumlah perangkat utama, seperti Arduino UNO sebagai otak pengendali, sensor suhu LM35 sebagai input suhu lingkungan, keypad 4x4 sebagai alat input nilai konstanta PID (KP, KI, KD), LCD I2C sebagai media informasi, serta motor DC sebagai aktuator yang dikendalikan menggunakan sinyal PWM dari Arduino. Implementasi dilakukan dengan menyusun dan merangkai seluruh perangkat keras berdasarkan rancangan pin yang telah ditentukan sebelumnya, lalu memprogram Arduino dengan kode yang sesuai untuk memastikan seluruh fungsi berjalan dengan sinkron.

Proses implementasi ini tidak hanya melibatkan pemasangan komponen secara fisik, tetapi juga penyesuaian pada bagian logika kontrol agar mampu beroperasi secara adaptif. Sistem akan memproses data suhu yang diterima dari sensor LM35, kemudian menghitung respon kendali melalui algoritma PID untuk menentukan kecepatan motor. LCD I2C digunakan untuk menampilkan informasi penting seperti suhu dan RPM motor secara real-time. Sedangkan keypad berfungsi sebagai alat input utama bagi pengguna untuk memasukkan parameter PID dan mengaktifkan atau menonaktifkan sistem.

Setelah proses implementasi selesai, dilakukan pengujian untuk mengevaluasi fungsionalitas dan performa sistem. Pengujian ini bertujuan untuk mengetahui apakah sistem mampu merespons perubahan suhu dengan mengatur kecepatan motor sesuai dengan nilai setpoint yang diinginkan. Hasil dari pengujian digunakan untuk menilai akurasi kontrol PID dan stabilitas sistem.

Tahapan Pengujian:

1. Pengujian Berdasarkan Variasi Suhu

Sistem diuji dengan kondisi suhu rendah ($< 25^{\circ}\text{C}$) dan suhu tinggi ($> 25^{\circ}\text{C}$). Pada suhu rendah, motor diatur pada kecepatan sekitar 100 RPM, sedangkan pada suhu tinggi motor diarahkan mencapai 200 RPM.

2. Pemantauan Output Sistem

Output seperti nilai suhu dan RPM motor ditampilkan secara real-time di LCD dan juga dikirim ke Serial Monitor untuk keperluan dokumentasi dan observasi.

3. Analisis Respons PID

Nilai error, integral, dan turunan dari PID dianalisis untuk melihat seberapa baik sistem dalam mencapai nilai setpoint yang ditentukan.

4. Evaluasi Stabilitas dan Akurasi

Sistem diuji terus menerus untuk mengetahui apakah kendali PID mampu mempertahankan kecepatan motor secara stabil dalam waktu tertentu dan dalam kondisi suhu yang berubah-ubah.

5. Penyesuaian Konstanta PID

Berdasarkan hasil pengujian, dilakukan pengaturan ulang terhadap nilai KP, KI, dan KD untuk mendapatkan respons sistem yang lebih optimal.

Secara keseluruhan, tahap implementasi dan pengujian menunjukkan bahwa sistem kontrol berbasis PID ini dapat bekerja secara efektif dalam menyesuaikan kecepatan motor terhadap perubahan suhu lingkungan. Namun, perlu dilakukan kalibrasi berulang pada nilai konstanta PID untuk mencapai performa sistem yang lebih stabil dan akurat. Dokumentasi proses ini menjadi landasan evaluasi dan pengembangan sistem di masa mendatang.

4.5 Karakteristik Motor Yang Digunakan

Motor yang digunakan dalam proyek ini adalah motor DC tipe JGA25-370, yaitu jenis motor DC dengan gearbox yang dirancang untuk menghasilkan torsi tinggi pada putaran rendah. Motor ini cocok untuk berbagai aplikasi sistem kendali karena memiliki karakteristik mekanik yang stabil serta mudah diintegrasikan dengan mikrokontroler dan sensor umpan balik. Beberapa spesifikasi umum dari motor JGA25-370 ditunjukkan pada Tabel berikut:

Tabel 4. 2 Karakteristik Motor

Spesifikasi	Nilai
Tegangan Operasional	6V – 12V DC
Kecepatan Tanpa Beban (12V)	$\pm 100 - 1000$ RPM (tergantung rasio gearbox)
Torsi Maksimum	Hingga ± 10 kg·cm
Rasio Gearbox	1:10 hingga 1:150
Diameter Motor	25 mm
Panjang Motor + Gearbox	± 65 mm
Diameter Poros	6 mm

Motor ini juga dilengkapi dengan rotary encoder yang terpasang di bagian belakang, dengan resolusi 360 pulsa per putaran. Encoder ini berfungsi sebagai sensor umpan balik untuk mengukur kecepatan rotasi aktual motor secara real-time, sehingga sangat mendukung implementasi sistem kontrol berbasis PID.

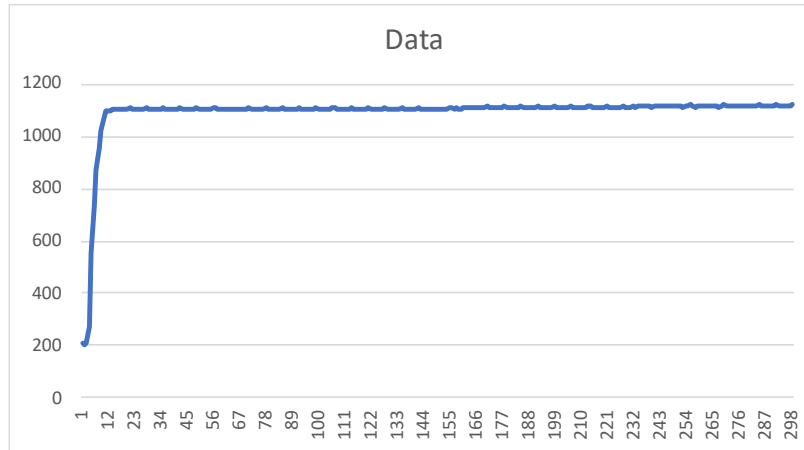
4.5.1 Uji Respons Dinamis Tanpa Beban

Sebelum implementasi sistem kendali PID, dilakukan pengujian awal untuk memperoleh karakteristik respons waktu alami motor tanpa beban. Pengujian ini bertujuan untuk mengidentifikasi tipe sistem berdasarkan tanggapan motor terhadap masukan konstan (PWM duty cycle tetap).

Metode:

- Pengambilan data dilakukan selama 75 detik dengan interval 0,25 detik, sehingga diperoleh 300 data sampel.
- Data diambil dari output kecepatan motor dalam satuan pulsa per interval waktu menggunakan encoder.
- Pengolahan dan visualisasi data dilakukan dengan perangkat lunak Microsoft Excel.

Gambar dibawah ini memperlihatkan kurva tanggapan waktu motor berdasarkan hasil pengukuran pengambilan data di ambil menggunakan software matlab yang di mana codenya bisa di lihat di Lampiran 7 Code Pengambilan Data Motor Tanpa Beban:



Gambar 4. 4 Grafik Tanggapan Waktu Motor DC JGA25-370 Tanpa Beban

4.5.2 Analisis Grafik dan Parameter Sistem

Dari hasil grafik diperoleh beberapa parameter penting dari sistem motor,

yaitu:

- Waktu tunak (settling time, 2%): 2.5 detik
- Steady-State Error (SSE): 0.43%
- Overshoot: 1.22%
- Waktu naik (rising time, 10%–90%): 0 – 2.5 detik
- Nilai steady-state (K): 1111

Nilai-nilai ini menunjukkan bahwa motor memiliki respons yang cepat, stabil, dan minim osilasi, sesuai dengan karakteristik sistem orde satu.

4.5.3 Pemodelan Sistem Orde Satu

Respons sistem pertama kali dimodelkan sebagai sistem orde satu berdasarkan bentuk kurva eksponensial:

$$c(t) = 1 - e^{-\frac{t}{T}}$$

Diketahui:

- Nilai steady-state: $K=1111$
- Waktu tunak 98% ($4T$): 2.5 detik, $T = 0.625$

Maka:

$$c(t) = 1111(1 - e^{-\frac{t}{0.625}})$$

$$c(t) = 1111(1 - e^{-1.6t})$$

Atau :

$$c(t) = 1111 - 1111e^{-1.6t}$$

4.5.4 Fungsi Alih (Transfer Function)

Transformasi Laplace dari model di atas menghasilkan fungsi alih sistem motor:

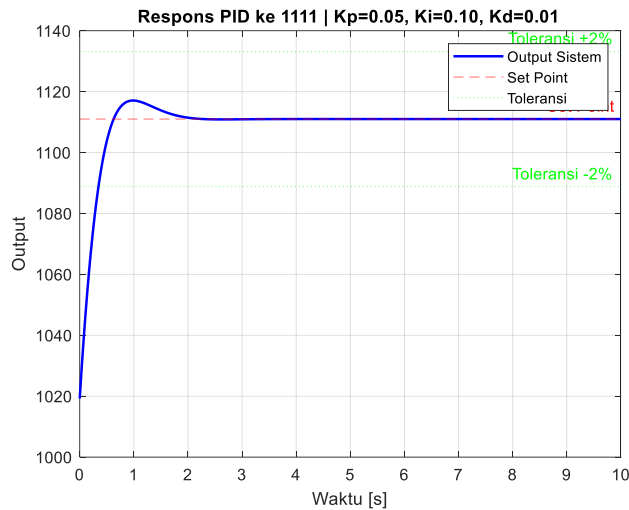
$$C(s) = \frac{1111}{s+1.6}$$

Model ini menggambarkan bahwa sistem memiliki satu kutub real negatif di -1.6, tanpa nol, sehingga menunjukkan karakteristik stabil dan respons eksponensial klasik. Fungsi alih ini akan digunakan dalam perancangan simulasi dan tuning parameter PID pada bab selanjutnya.

4.6 Pengujian Transef Function Pada Motor

Setelah diperoleh fungsi alih dari hasil identifikasi motor DC JGA25-370, dilakukan pengujian lebih lanjut terhadap model matematis tersebut menggunakan simulasi berbasis perangkat lunak MATLAB. Tujuan dari pengujian ini adalah untuk melihat bagaimana respons sistem terhadap kontrol PID dan memastikan bahwa model transfer function yang dibuat benar-benar sesuai dengan karakteristik motor yang sebenarnya.

Berikut Gambar yang menunjukkan respons sistem terhadap kendali PID yang diterapkan dengan matlab dan source codenya dapat di lihat di lampiran 7 Source Code Percobaan Transfer Function.



Gambar 4. 5 Grafik Motor Dengan Transfer Function

Fungsi alih yang digunakan dalam simulasi ini adalah:

$$C(s) = \frac{1111}{s+1.6}$$

Simulasi dilakukan dengan memberi masukan step ke sistem, menggunakan parameter PID sebagai berikut:

- $K_p = 0.05$
- $K_i = 0.10$
- $K_d = 0.01$

Hasil simulasi ditampilkan dalam Gambar 4.5, yang menunjukkan kurva output sistem terhadap waktu. Dari grafik tersebut terlihat bahwa sistem dapat mencapai nilai set point sebesar 1111 dengan cepat dan stabil.

Berdasarkan hasil simulasi, didapatkan beberapa parameter penting, yaitu:

- Steady-state error = 0.0000 (0%)
- Nilai akhir (output akhir) = 1111.0000
- Settling time ($\pm 2\%$) = 0.3500 detik
- Overshoot = 0.55% (output maksimum = 1117.0819)
- Rise time (10%–90%) = 0.0000 detik

Dari data tersebut dapat dilihat bahwa sistem memiliki waktu tunak yang cukup cepat dan overshoot yang sangat kecil, yaitu hanya sekitar 0.55%. Hal ini menunjukkan bahwa parameter PID yang digunakan sudah cukup baik dalam

mengendalikan sistem agar cepat mencapai nilai target tanpa fluktuasi besar.

Kesimpulannya, simulasi menggunakan MATLAB ini menunjukkan bahwa fungsi alih motor yang sudah diperoleh bisa digunakan sebagai dasar untuk tuning PID secara simulatif. Dengan begitu, pemilihan nilai Kp, Ki, dan Kd bisa diuji dulu melalui simulasi sebelum diterapkan langsung pada perangkat keras, sehingga proses tuning menjadi lebih efisien dan aman.

4.7 Analisa dan Pembahasan Pada Arduino

4.7.1 Metode Pengujian dan Sistem Kendali

Pengujian dilakukan secara langsung menggunakan sistem berbasis mikrokontroler Arduino untuk mengendalikan kecepatan motor DC JGA25-370 dengan pendekatan algoritma PID. Sistem dilengkapi dengan sensor suhu LM35, LCD I2C 16x2, keypad 4x4 untuk input nilai PID, dan encoder sebagai pembaca kecepatan aktual (RPM). Program dibuat sedemikian rupa sehingga suhu dari sensor LM35 menjadi penentu nilai set point kecepatan motor.

Logika sistem adalah sebagai berikut:

- Jika suhu $\leq 25^{\circ}\text{C}$, maka set point RPM = 100
- Jika suhu $> 25^{\circ}\text{C}$, maka set point RPM = 200

Namun, suhu hanya memengaruhi pemilihan nilai set point, sedangkan kendali kecepatan motor tetap dilakukan oleh algoritma PID.

- Kode program Arduino memuat fungsi utama seperti:
- Pembacaan suhu dengan filter sederhana (low-pass filter)
- Perhitungan PID secara berkala setiap 50 ms
- Perhitungan RPM per detik
- Tampilan suhu dan RPM secara real-time di LCD
- Pengaturan nilai PID melalui keypad

Berikut merupakan kode sumber implementasi pengendalian motor tersebut.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```

// === Konfigurasi pin dan sistem ===
const int motorPin = 3;
const int lm35Pin = A0;
const int encoderPin = 2;
volatile int countEncoder = 0;
const int encoderPPR = 360;

float KP = 1.0, KI = 0.0, KD = 0.0;
float setPoint = 100;
float highSetPoint = 200;

float rpm = 0.0;
float previousError = 0;
float integral = 0;
float output = 0;

unsigned long lastPIDTime = 0;
unsigned long lastDisplayTime = 0;
unsigned long lastRPMTIME = 0;
const float dt = 0.05;
bool systemStarted = false;
String inputStr = "";
char mode = ' ';
bool KP_flag = false, KI_flag = false, KD_flag = false;
float avgTemperature = 0;

// === Keypad setup ===
const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  { '1', '2', '3', 'A' },
  { '4', '5', '6', 'B' },
  { '7', '8', '9', 'C' },
  { '*', '0', '#', 'D' }
};
byte rowPins[ROWS] = { 8, A3, A2, A1 };
byte colPins[COLS] = { 12, 11, 10, 9 };
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins,
ROWS, COLS);

void countEncoderPulse() {
  countEncoder++;
}

float readLM35() {
  int val = analogRead(lm35Pin);
  float voltage = val * (5.0 / 1023.0);
  return voltage * 100.0;
}

float readTemperatureFiltered() {
  float temp = readLM35();
  avgTemperature = 0.8 * avgTemperature + 0.2 * temp;
  return avgTemperature;
}

```

```

}

void setup() {
    Serial.begin(9600);
    lcd.init();
    lcd.backlight();
    lcd.setCursor(0, 0); lcd.print("Inisialisasi...");

    pinMode(encoderPin, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(encoderPin),
countEncoderPulse, RISING);
    pinMode(motorPin, OUTPUT);
    analogWrite(motorPin, 0);

    avgTemperature = readLM35();
}

void loop() {
    unsigned long currentMillis = millis();
    if (systemStarted) {
        float temperature = readTemperatureFiltered();
        setPoint = (temperature > 30) ? highSetPoint : 100;
        if (currentMillis - lastPIDTime >= 50) {
            lastPIDTime = currentMillis;
            updateMotorSpeed();
        }
        if (currentMillis - lastRPMTIME >= 1000) {
            lastRPMTIME = currentMillis;
            noInterrupts();
            int pulses = countEncoder;
            countEncoder = 0;
            interrupts();
            rpm = (pulses / (float)encoderPPR) * 60.0;
            Serial.print("RPM: "); Serial.println(rpm);
        }
        if (currentMillis - lastDisplayTime >= 500) {
            lastDisplayTime = currentMillis;
            // Tampilkan Temp dan RPM
            lcd.setCursor(0, 0); lcd.print("Temp:          C"); //
label
            lcd.setCursor(6, 0); lcd.print(temperature, 1);
            lcd.print(" "); // nilai suhu
            lcd.setCursor(0, 1); lcd.print("RPM:          "); //
label
            lcd.setCursor(5, 1); lcd.print(rpm, 0); lcd.print("
"); // nilai RPM
        }
    }

    char key = keypad.getKey();
    if (key) handleKeyPress(key);
}

```

```

// === PID dan Keypad ===
void updateMotorSpeed() {
    float error = setPoint - rpm;
    integral += error * dt;
    integral = constrain(integral, -100, 100);
    float derivative = (error - previousError) / dt;
    output = (KP * error) + (KI * integral) + (KD *
derivative);
    output = constrain(output, 0, 255);
    analogWrite(motorPin, output);
    previousError = error;
    Serial.print("Error: "); Serial.print(error);
    Serial.print(" | Output: "); Serial.println(output);
}

void handleKeyPress(char key) {
    if (key == 'A' || key == 'B' || key == 'C') {
        mode = key;
        inputStr = "";
        lcd.clear();
        lcd.setCursor(0, 0); lcd.print((mode == 'A') ? "Set KP:"
: (mode == 'B') ? "Set KI:" : "Set KD:");
        lcd.setCursor(0, 1);
    } else if ((key >= '0' && key <= '9') || key == '*') {
        inputStr += (key == '*') ? '.' : key;
        lcd.print(key);
    } else if (key == '#') {
        if (!systemStarted) {
            if (!KP_flag || !KI_flag || !KD_flag) {
                processInput();
            } else {
                systemStarted = true;
                lcd.clear();
                lcd.print("Sistem Dimulai");
                Serial.println("=== Sistem Dimulai ===");
                delay(500);
            }
        } else {
            systemStarted = false;
            analogWrite(motorPin, 0);
            resetPID();
            lcd.clear();
            lcd.print("Sistem Dimatikan");
            Serial.println("=== Sistem Dimatikan ===");
        }
    }
}

void processInput() {
    if (mode != ' ') {
        if (mode == 'A') { KP = inputStr.toFloat(); KP_flag =
true; }
        if (mode == 'B') { KI = inputStr.toFloat(); KI_flag =
true; }
    }
}

```

```

        if (mode == 'C') { KD = inputStr.toFloat(); KD_flag =
true; }
        lcd.clear();
        lcd.print("KP:"); lcd.print(KP); lcd.print("  KI:");
lcd.print(KI); lcd.setCursor(0, 1); lcd.print("KD:");
lcd.print(KD);
        Serial.println("=== Parameter PID ===");
        Serial.print("KP:"); Serial.println(KP);
Serial.print("KI: "); Serial.println(KI); Serial.print("KD:
"); Serial.println(KD);
        delay(1000);
        lcd.clear();
        lcd.print("Tekan # untuk mulai");
        mode = ' ';
    }
}

void resetPID() {
    integral = 0;
    previousError = 0;
    output = 0;
    KP_flag = KI_flag = KD_flag = false;
    lcd.clear();
    lcd.print("PID Reset");
    Serial.println(">>> PID Reset <<<");
    delay(1000);
}

```

4.7.2 Analisa Grafik Respon Sistem

Grafik pada Gambar 4.6 menampilkan respon sistem terhadap perubahan nilai set point kecepatan motor yang dikendalikan secara otomatis berdasarkan suhu yang dibaca oleh sensor LM35. Pada pengujian ini, data diambil sebanyak 200 sampel dalam rentang waktu tertentu, dengan menggunakan kombinasi parameter PID yaitu $K_p = 0.05$, $K_i = 0.10$, dan $K_d = 0.01$.

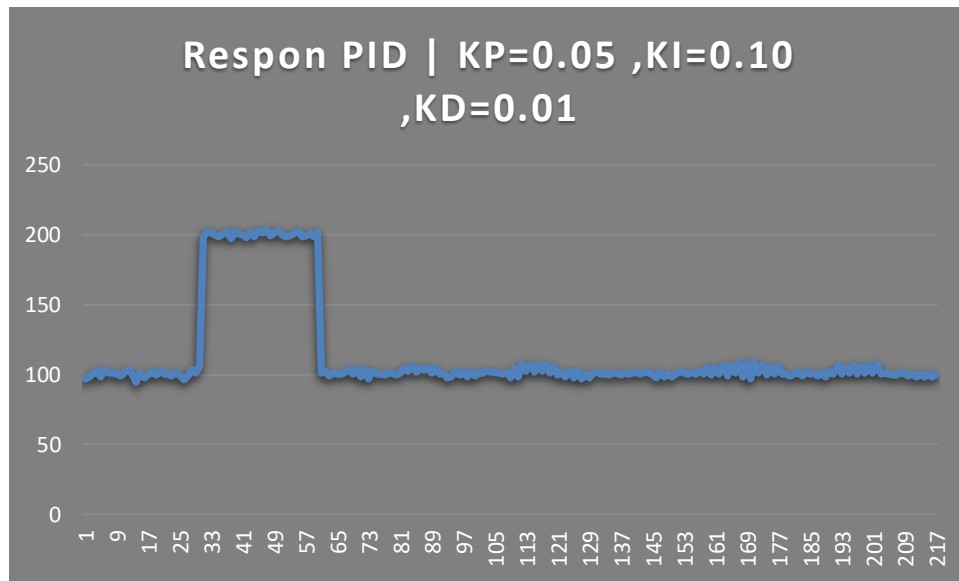
Sesuai dengan rancangan sistem, nilai set point kecepatan motor ditentukan berdasarkan suhu lingkungan:

- Jika suhu $\leq 25^\circ\text{C}$, maka set point = 100 RPM,
- Jika suhu $> 25^\circ\text{C}$, maka set point = 200 RPM.

Untuk tujuan pengujian ini, pengaturan suhu dibuat seolah-olah berubah dalam tiga tahapan waktu:

- detik pertama: set point diatur pada 100 RPM,
- detik berikutnya: set point naik ke 200 RPM,

- dan 3 detik terakhir: set point kembali ke 100 RPM.



Gambar 4. 6 Grafik PID Perubahan Set Point Berdasarkan Suhu

Data tersebut direkam dan divisualisasikan dalam grafik dengan sumbu-x sebagai indeks data (1 hingga 200), dan sumbu-y menunjukkan nilai kecepatan motor aktual (dalam RPM). Terlihat bahwa sistem merespons perubahan set point dengan cukup baik:

- Pada awal grafik (data ke-1 hingga sekitar ke-60), kecepatan sistem berada pada kisaran 100 RPM dengan sedikit fluktuasi akibat noise alami atau respon awal PID.
- Setelah titik tersebut, sistem mengikuti perubahan set point menjadi 200 RPM, yang tampak jelas pada bagian tengah grafik. Respons terlihat cepat dan relatif stabil dengan toleransi $\pm 2\%$.
- Di bagian akhir grafik, ketika suhu menurun kembali (atau disimulasikan menurun), sistem kembali menurunkan kecepatan menuju 100 RPM, yang menunjukkan sistem mampu menyesuaikan dengan perubahan suhu secara dinamis.

Respon sistem yang ditampilkan menunjukkan bahwa parameter PID yang digunakan mampu menjaga kestabilan dan ketepatan kecepatan motor dalam menghadapi perubahan suhu. Tidak tampak adanya overshoot berlebihan atau osilasi yang signifikan. Ini membuktikan bahwa PID controller telah bekerja secara efektif pada skenario perubahan suhu yang sederhana.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem kontrol PID untuk kecepatan motor DC berbasis suhu lingkungan, maka dapat disimpulkan hal-hal berikut:

1. Perancangan modul pembelajaran sistem kontrol PID berhasil dilakukan dengan menggunakan kombinasi perangkat keras seperti Arduino UNO, sensor suhu LM35, motor DC JGA25-370, driver motor Arduimoto, keypad 4x4, dan LCD 16x2. Modul ini dirancang agar dapat digunakan sebagai alat bantu pembelajaran interaktif, di mana mahasiswa dapat langsung mengamati respon sistem terhadap perubahan parameter kontrol PID secara nyata. Dengan tampilan output visual dan pengaturan input yang fleksibel, mahasiswa dapat memahami prinsip dasar PID secara praktis.
2. Pengaturan parameter PID (K_p , K_i , K_d) dapat dilakukan langsung melalui keypad 4x4 yang terhubung ke mikrokontroler. Pengguna diberikan kemudahan untuk mengubah nilai-nilai parameter tersebut sebelum sistem dijalankan. Pengaruh perubahan parameter terhadap kecepatan motor dapat diamati melalui respon sistem terhadap suhu lingkungan yang dibaca oleh sensor LM35. Sistem akan menaikkan atau menurunkan kecepatan motor berdasarkan nilai suhu, sehingga pengguna dapat melihat secara langsung bagaimana tuning PID memengaruhi kestabilan, overshoot, dan waktu tunda sistem.
3. Nilai kecepatan motor dan proses kontrol PID ditampilkan secara informatif pada LCD 16x2. LCD menampilkan dua informasi utama: suhu aktual dari sensor LM35 dan kecepatan motor dalam satuan RPM. Informasi ini diperbarui secara real-time, sehingga pengguna dapat dengan mudah memantau hasil proses kontrol dan mengetahui apakah sistem telah mencapai set point atau masih mengalami error.
4. Respon sistem terhadap perubahan set point dari sensor suhu LM35 menunjukkan performa yang baik dan stabil. Set point otomatis berubah

menjadi 100 RPM saat suhu $\leq 25^{\circ}\text{C}$ dan 200 RPM saat suhu $> 25^{\circ}\text{C}$. Dari 200 data pengujian, sistem mampu merespons perubahan dengan cepat dan stabil, terutama setelah parameter PID diatur optimal. Pengujian langsung menunjukkan error sebesar 2,47, kemungkinan akibat noise sensor atau gangguan fisik lainnya, sedangkan simulasi MATLAB menghasilkan SSE 0% karena dilakukan dalam kondisi ideal. Perbandingan ini menunjukkan bahwa sistem bekerja cukup efektif meskipun kondisi nyata tidak seideal simulasi.

Dengan demikian, sistem yang dirancang tidak hanya memenuhi aspek teknis dalam pengendalian motor DC menggunakan PID, tetapi juga menjawab kebutuhan edukatif sebagai alat bantu pembelajaran yang mudah dipahami dan diaplikasikan oleh mahasiswa.

5.2 Saran

Untuk pengembangan di masa mendatang, sistem kontrol suhu berbasis PID ini dapat ditingkatkan dengan beberapa pendekatan inovatif. Pertama, pengembangan fitur visualisasi data secara langsung melalui layar grafis kecil (misalnya OLED) atau antarmuka berbasis aplikasi seluler akan meningkatkan pengalaman interaktif pengguna. Hal ini dapat memudahkan pemantauan perubahan suhu dan respons sistem secara lebih intuitif dibandingkan hanya melalui angka di LCD.

Kedua, sistem ini juga dapat diperluas dengan kemampuan komunikasi dua arah melalui Bluetooth atau Wi-Fi, sehingga mahasiswa dapat mengontrol dan memonitor sistem dari jarak jauh menggunakan perangkat seluler atau laptop tanpa koneksi kabel. Integrasi ini tidak hanya menambah fleksibilitas, tetapi juga mempersiapkan mahasiswa menghadapi era Internet of Things (IoT) dalam sistem kendali.

Dari aspek keamanan, disarankan agar sistem dilengkapi dengan proteksi terhadap suhu berlebih, tegangan tidak stabil, atau gangguan lainnya, untuk menjaga keandalan perangkat saat digunakan secara terus-menerus dalam praktikum. Terakhir, pelatihan penggunaan modul kepada dosen dan mahasiswa perlu dilakukan secara berkala agar pemanfaatan modul ini maksimal, baik dalam kegiatan laboratorium, tugas akhir, maupun proyek riset mahasiswa.

DAFTAR PUSTAKA

- Alciatore, D. G., & Hiestand, M. B. (2011). Introduction to mechatronics and measurement systems (4th ed.). McGraw-Hill Education.
- Åström, K. J., & Hägglund, T. (2006). Advanced PID control. ISA – The Instrumentation, Systems, and Automation Society.
- Åström, K. J., & Murray, R. M. (2016). Feedback systems: An introduction for scientists and engineers (2nd ed.). Princeton University Press.
- Banzy, M., & Shiloh, M. (2014). Getting started with Arduino: The open source electronics prototyping platform (3rd ed.). Maker Media, Inc.
- Bitriá, R., & Palacín, J. (2022). Optimized PID controller for a DC motor using temperature feedback: Real-time implementation on an embedded system. *Sensors*, 22(20), 7817. <https://doi.org/10.3390/s22207817>
- Boylestad, R. L., & Nashelsky, L. (2012). Electronic devices and circuit theory (10th ed.). Pearson.
- Bulan, R., & Setiawan, A. (2021). Pengendalian kecepatan motor DC berbasis PID menggunakan Arduino UNO. *Jurnal Teknologi dan Sistem Komputer*, 9(1), 45–52. <https://doi.org/10.14710/jtsiskom.9.1.45-52>
- Displaytech. (2019). LCD fundamentals. <https://www.displaytech-us.com>
- Dong, L. (2020). Fundamentals of electromechanical energy conversion. Springer.
- Hughes, A., & Drury, B. (2013). Electric motors and drives: Fundamentals, types and applications (4th ed.). Newnes.
- Hughes, T. (2020). Arduino programming in 24 hours, Sams teach yourself (2nd ed.). Pearson Education.
- Ibrahim, D. (2011). Designing embedded systems with PIC microcontrollers: Principles and applications. Newnes.
- Karray, F., De Silva, C. W., & Langlois, D. (2021). Soft computing and intelligent systems design: Theory, tools and applications (3rd ed.). Pearson.
- Khandpur, R. S. (2004). Handbook of analytical instruments (2nd ed.). McGraw-Hill Education.
- Khosrowjerdi, M. J., & Chen, C.-H. (2020). Practical aspects of derivative action in PID control: Design and implementation issues. *ISA Transactions*, 100, 315–327.

<https://doi.org/10.1016/j.isatra.2019.11.001>

Kuo, B. C., & Golnaraghi, F. (2003). Automatic control systems (8th ed.). John Wiley & Sons.

Kurniawan, A. (2020). Belajar sendiri mikrokontroler Arduino dan ESP8266. Informatika.

Ma'arif, S., & Çakan, H. (2021). PID controller tuning for DC motor using trial-and-error method and Arduino implementation. *Journal of Robotics and Control (JRC)*, 2(4), 344–350. <https://doi.org/10.18196/jrc.v2i4.11125>

Margolis, M. (2011). *Arduino cookbook: Recipes to begin, expand, and enhance your projects* (2nd ed.). O'Reilly Media.

Mazidi, M. A., Naimi, S., & Naimi, S. (2016). *AVR microcontroller and embedded systems: Using assembly and C* (1st ed.). Pearson Education.

Monk, S. (2013). *Programming Arduino: Getting started with sketches* (2nd ed.). McGraw-Hill Education.

Monk, S. (2016). *Programming Arduino: Getting started with sketches* (2nd ed.). McGraw-Hill Education.

Monk, S., & Simon, T. (2012). *Programming Arduino: Getting started with sketches*. McGraw-Hill Education.

Nise, N. S. (2011). *Control systems engineering* (6th ed.). John Wiley & Sons.

Nise, N. S. (2020). *Control systems engineering* (8th ed.). Wiley.

Nugroho, A., & Prasetyo, H. (2022). Rancang bangun sistem kendali suhu otomatis menggunakan PID berbasis Arduino. *Jurnal Ilmu Komputer dan Informatika*, 8(2), 89–97.

Ogata, K. (2010). *Modern control engineering* (5th ed.). Prentice Hall.

Ogata, K. (2020). *Modern control engineering* (6th ed.). Pearson.

Pololu. (2023). 25D metal gearmotors. <https://www.pololu.com/category/115/25d-metal-garmotors>

SparkFun. (2020). Ardumoto - Motor driver shield. <https://www.sparkfun.com/products/14129>

Suhendar. (2021). *Dasar teknik kendali*. EPrints Repository UNTIRTA. <https://eprints.untirta.ac.id/15543/2/a2%20Dasar%20Teknik%20Kendali.pdf>

Texas Instruments. (2000). LM35 precision centigrade temperature sensors [Datasheet]. <https://www.ti.com/lit/ds/symlink/lm35.pdf>

Zhou, J., Wang, Y., & Sun, X. (2019). PID parameter tuning based on system identification and optimization method for dynamic systems. *Journal of Electrical Engineering and Automation*, 1(1), 32–40. <https://doi.org/10.21926/jeea.1901005>



UNIVERSITAS
Dinamika