

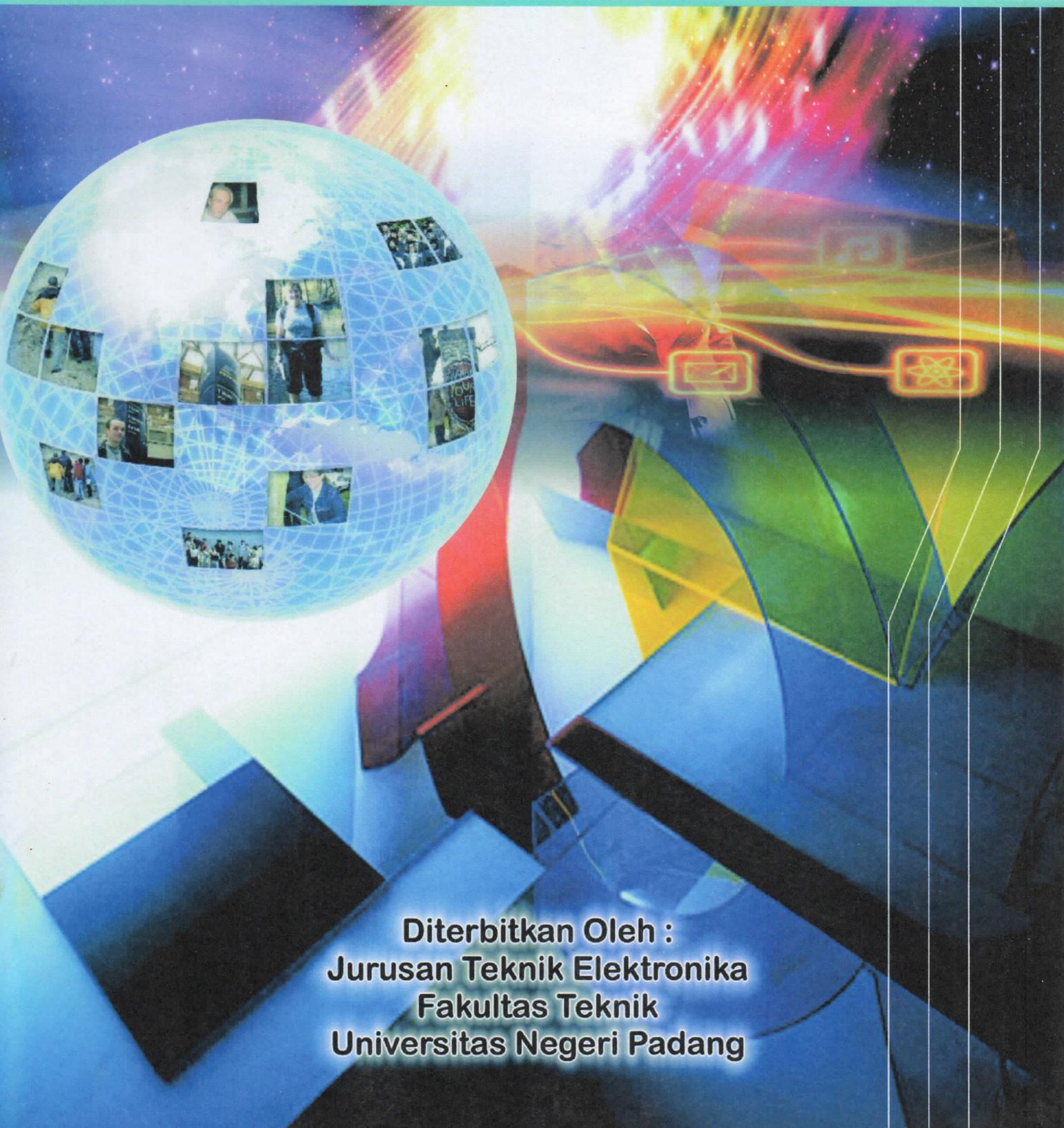
Vol. 4

No. 1

September 2011

ISSN
2086 - 4981

Jurnal Teknologi Informasi dan Pendidikan



Diterbitkan Oleh :
Jurusan Teknik Elektronika
Fakultas Teknik
Universitas Negeri Padang

Jurnal Teknologi Informasi dan Pendidikan

Susunan Redaksi Jurnal

Penasehat

Dekan Fakultas Teknik UNP

Penanggung Jawab

Ketua Jurusan Teknik Elektronika FT UNP

Pimpinan Umum/Pimpinan Redaksi

Dony Novaliendry, S.Kom., M.Kom

Mitra Bestari

Lukito, Ph.D

Retantyo Wardoyo, Drs., M.Sc., Ph.D

Sri Hartati, Dra. Ph.d

Vid Adrison, Ph.d

Sekretaris

Titi Sriwahyuni, S.Pd., MT

Bendahara

Nurindah Dwiyani, S.Pd., MT

Dewan Redaksi

Kasman Rukun, M.Pd., Prof

Zulhendra, Drs., M.Kom

Elfi Tasrif, Drs., M.T

Denny Kurniadi, Drs., M.Kom

Putra Jaya, Drs., M.T

Efrizon, Drs. M.T

M. Adri., S.Pd., M.T

Edidas, Drs. M.T

Redaksi Pelaksana

Thamrin, S.Pd

Ahmaddul Hadi, S.Pd

Yasdinul Huda, S.Pd., M.T

Alamat Redaksi

Fakultas Teknik Universitas Negeri Padang

Jurusan Elektronika

Jl. Prof. Dr. Hamka

Padang 25131

DAFTAR ISI

Perancangan Pengendali Kecepatan Motor Arus Searah (DC) Dengan Pendekatan Tanggapan Frekuensi Dengan Menggunakan MATLAB <i>Heru Dibyo Laksono</i>	1-14
Penggunaan Program AUTOCAD 2010 Dalam Pembelajaran Gambar Listrik <i>Fivia Eliza</i> <i>Oriza Candra</i>	15-26
Optimalisasi Produk Mobil Dengan Sistem Pendukung Keputusan Berbasis Fuzzy <i>Dedy Irfan</i>	27-44
Implementasi Perancangan Sistem Informasi Ekspedisi Paket Pada PT. POS Indonesia <i>Titi Sriwahyuni</i>	45-64
Pemanfaatan Animasi Multimedia Pada Mata Kuliah Kimia Teknik Untuk Peningkatan Pemahaman Mahasiswa Terhadap Konsep Ikatan Kimia <i>Irma Yulia Basri</i> <i>Muhammad Adri</i>	65-77
Pembuatan Aplikasi Pembelajaran Berhitung Bagi Anak Playgroup Menggunakan Metode Menghitung Gambar Dengan Menerapkan Bahasa Pemrograman Macromedia Flash 8 <i>Rini Sovia</i>	78-89
Perancangan Dan Penalaan Pengendali <i>Proportional Integral Derivatif</i> Menggunakan Simulink <i>Hastuti</i>	90-97
Peta Lokal Wilayah Gempa Untuk Daerah Padang <i>Yoszi Mingsi Anaperta</i>	98-112
Simulasi Algoritma Backpropagation Pada Jaringan Syaraf Tiruan Untuk Pengendalian Kecepatan Motor DC <i>Irma Husnaini</i>	113-121
Simulasi Kombinasi Space Time Block Code Dengan Teknik MC CDMA Pada Sistem MIMO Menggunakan MATLAB 7.0 <i>Delsina Faiza</i>	122-136

Sistem Informasi Geografis Jaringan Distribusi Listrik Berbasis WEB (Studi Kasus: Wilayah Kentungan) 137-156

Oktaria

Aplikasi Software Bantu Heap Short 157-176

Dony Novaliendryr

**PERANCANGAN DAN PENALAAN
PENGENDALI *PROPORTIONAL INTEGRAL DERIVATIF* MENGGUNAKAN
SIMULINK**

Hastuti¹

ABSTRACT

This paper describes how to design and to adjust parameters of the PID Controller in order to achieve the best performance. The design of PID Controller use a simulink model. This simulation model is developed to support students to understand easily the close loop response of control system with a variety of input. The model is also very simple. Therefore, the students do not use their time for complex procedures, mathematical equations, and script file, but they will have much time to analyse the close loop response of control system.

Keywords : *PID Controller, The Simulink Model, Close Loop Response of Control System*

INTISARI

Artikel ini memaparkan perancangan dan penalaan konstanta-konstanta pengendali PID sehingga menghasilkan tanggapan atau performansi yang diinginkan. Perancangan pengendali yang dibahas pada artikel ini menggunakan model simulink. Model simulasi simulink yang dirancang memudahkan mahasiswa untuk mengamati tanggapan lup tertutup sistem kendali dengan berbagai input. Model dirancang sesederhana mungkin untuk menghindari prosedur yang kompleks, perhitungan matematika yang rumit, dan pembuatan script file yang panjang, sehingga mahasiswa lebih mempunyai banyak waktu untuk menganalisa perilaku (tanggapan sistem lup tertutup) sistem kendali tersebut.

Kata Kunci : Pengendali PID, Model Simulink, Tanggapan Sistem Lup Tertutup

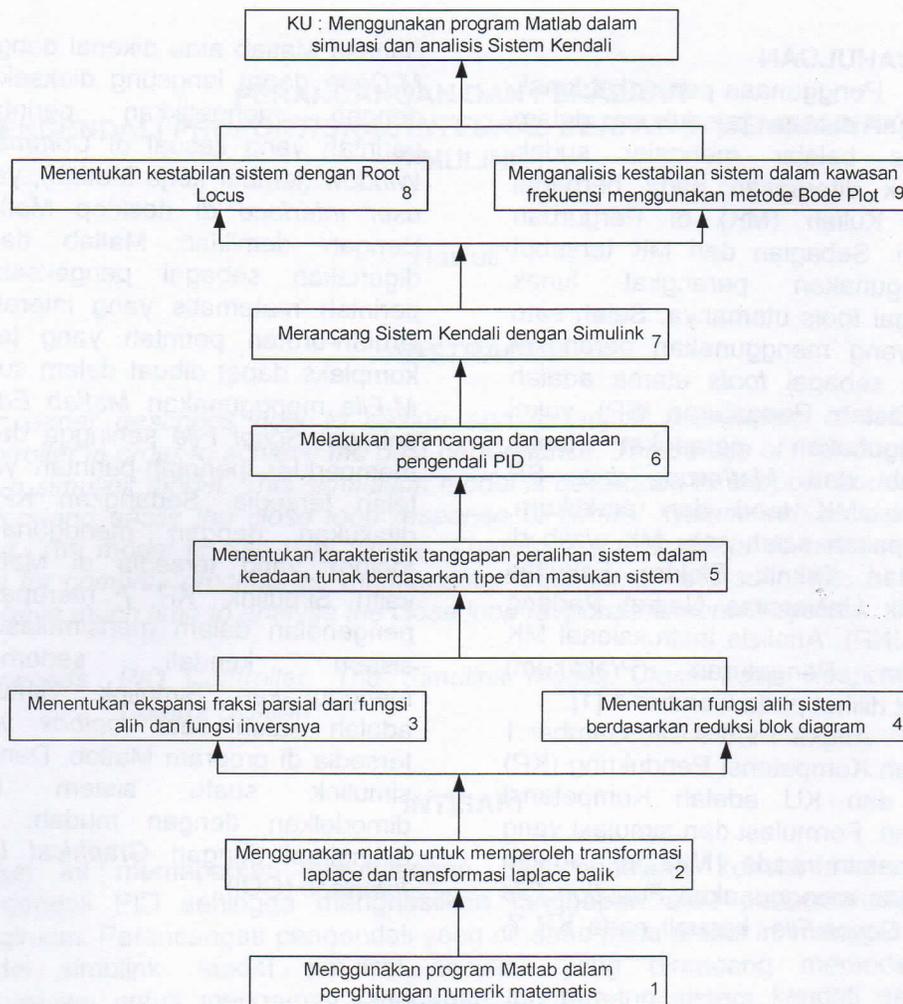
¹ Dosen Teknik Elektro Fakultas Teknik Universitas Negeri Padang

PENDAHULUAN

Penggunaan perangkat lunak sebagai sarana pendukung dalam proses belajar mengajar sudah banyak diterapkan pada berbagai Mata Kuliah (MK) di Perguruan Tinggi. Sebagian dari MK tersebut menggunakan perangkat lunak sebagai *tools* utamanya. Salah satu MK yang menggunakan perangkat lunak sebagai *tools* utama adalah MK Sistem Pengaturan (SP), yakni menggunakan perangkat lunak Matlab dari *Mathwork, Inc.* SP adalah MK teori dan praktikum, merupakan salah satu MK wajib di Jurusan Teknik Elektro Fakultas Teknik Universitas Negeri Padang (FT-UNP). Analisis instruksional MK Sistem Pengaturan (Praktikum) dapat dilihat pada Gambar 1 [1].

Angka 1 s/d 9 dari Gambar 1 adalah Kompetensi Pendukung (KP) SP, dan KU adalah Kompetensi Utama. Formulasi dan simulasi yang digunakan pada MK ini adalah dengan menggunakan *Function File* dan *Script File*, kecuali pada KP 7.

Bahasa Matlab atau dikenal dengan *M-Code* dapat langsung dieksekusi dengan menetikkan perintah-perintah yang sesuai di *Command Window* (lembar kerja Matlab), yaitu *user interface* di desktop Matlab. Dengan demikian Matlab dapat digunakan sebagai pengeksekusi perintah matematis yang interaktif. Urutan-urutan perintah yang lebih kompleks dapat dibuat dalam suatu *M-File* menggunakan *Matlab Editor* sebagai *Script File* sehingga dapat memperluas perintah-perintah yang telah tersedia. Sedangkan KP 7 dilakukan dengan menggunakan *toolbox* yang tersedia di Matlab, yaitu Simulink. KP 7 merupakan pengenalan dalam mensimulasikan sistem kendali sederhana menggunakan Simulink. Simulink adalah salah satu *toolbox* yang tersedia di program Matlab. Dengan simulink suatu sistem bisa dimodelkan dengan mudah, dan dilengkapi dengan *Graphical User Interface* (GUI).

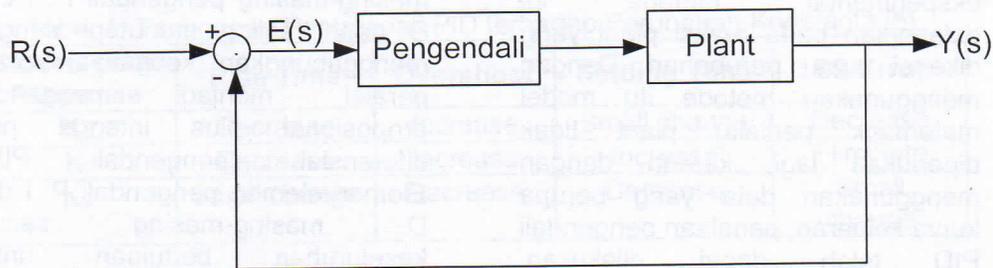


Gambar 1. Analisis Instruksional MK SP (Praktikum)

Kendala yang ditemui dalam pencapaian KP-KP pada MK SP ini adalah pada pencapaian KP 6, yaitu melakukan perancangan dan penalaan pengendali PID (*Proportional Integral Derivatif*). Perancangan sebuah sistem kendali diawali dengan pembuatan blok diagram sistem (Gambar 2). Blok-blok diagram tersebut diformulasikan sebagai fungsi alih atau persamaan keadaan, kemudian dianalisis dengan input dan aksi pengontrolan yang berbeda. Untuk menggambarkan tanggapan sistem kendali dilakukan prosedur berikut [2] :

1. Mendapatkan fungsi alih sistem dalam persamaan transformasi laplace (domain s).

2. Menentukan jenis aksi pengontrolan beserta dengan konstantanya.
3. Menggabungkan fungsi alih yang sudah didapatkan dengan jenis aksi pengontrolan.
4. Menentukan sinyal input yang akan dimasukkan (biasanya fungsi *step*, fungsi *ramp* dan *pulse*) dan gabungkan ke dalam fungsi alih yang baru.
5. Melakukan perhitungan invers transformasi laplace untuk mendapatkan fungsi dalam domain t.
6. Menggambar tanggapan berdasarkan fungsi dalam domain t (fungsi waktu).



Gambar 2. Blok Diagram Sistem Kendali dengan Umpan Balik Unity

Prosedur di atas dilakukan dengan teliti dan berulang kali, sampai diperoleh tanggapan sistem yang diinginkan. Tambahan lagi, untuk mendapatkan tanggapan sistem yang diinginkan maka prosedur tersebut juga dilakukan untuk kombinasi input dan aksi-aksi pengontrolan yang berbeda. Oleh karenanya, proses perhitungan yang dilakukan tidaklah mudah dan butuh waktu yang lama. Untuk itu, dalam artikel ini penulis mengembangkan suatu media pembelajaran yang lebih interaktif, serta menghindari proses perhitungan yang lama dan membosankan, sehingga mahasiswa bisa memahami dengan mudah materi perancangan sistem kendali dan penentuan parameter-parameter PID yang biasanya dilakukan dengan metoda *trial and error*. Media pembelajaran yang akan dibuat adalah suatu model sistem dengan Simulink, yaitu sistem *closed-loop* dengan pengendali PID sehingga dengan input apa saja bisa langsung dilihat tanggapan sistem atau performansi keluarannya berupa grafik.

PENDEKATAN PEMECAHAN MASALAH

Penalaan Pengendali PID

Keberadaan pengendali dalam sebuah sistem kendali mempunyai kontribusi yang besar terhadap perilaku sistem. Pada prinsipnya hal itu disebabkan oleh tidak dapat diubahnya komponen penyusun sistem tersebut.

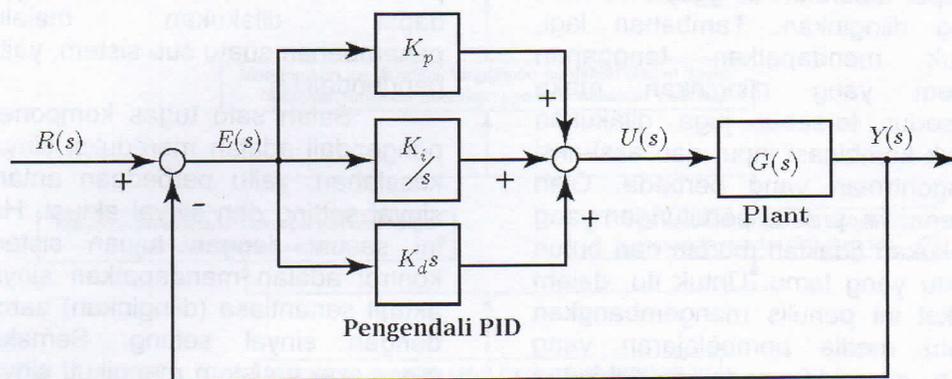
Karakteristik *plant* harus diterima sebagaimana adanya, sehingga perubahan perilaku sistem hanya dapat dilakukan melalui penambahan suatu sub sistem, yaitu pengendali [3].

Salah satu tugas komponen pengendali adalah mereduksi sinyal kesalahan, yaitu perbedaan antara sinyal setting dan sinyal aktual. Hal ini sesuai dengan tujuan sistem kontrol adalah mendapatkan sinyal aktual senantiasa (diinginkan) sama dengan sinyal setting. Semakin cepat reaksi sistem mengikuti sinyal aktual dan semakin kecil kesalahan yang terjadi, semakin baiklah kinerja sistem kontrol yang diterapkan. Apabila perbedaan antara nilai setting dengan nilai keluaran relatif besar, maka pengendali yang baik seharusnya mampu mengamati perbedaan ini untuk segera menghasilkan sinyal keluaran untuk mempengaruhi *plant*. Dengan demikian sistem secara cepat mengubah keluaran *plant* sampai diperoleh selisih antara setting dengan besaran yang diatur sekecil mungkin.

Penalaan parameter pengendali PID selalu didasari atas tinjauan terhadap karakteristik yang diatur (*plant*). Dengan demikian betapapun rumitnya suatu *plant*, perilaku *plant* tersebut harus diketahui terlebih dahulu sebelum penalaan parameter PID itu dilakukan. Karena penyusunan model matematik *plant* tidak mudah, maka dikembangkan suatu metode

eksperimental. Metode ini didasarkan pada reaksi plant yang dikenai suatu perubahan. Dengan menggunakan metode itu model matematik perilaku plant tidak diperlukan lagi, karena dengan menggunakan data yang berupa kurva keluaran, penalaan pengendali PID telah dapat dilakukan. Pendekatan eksperimental penalaan pengendali PID yang dipaparkan pada artikel ini menggunakan metode Ziegler-Nichols. Setiap kekurangan dan kelebihan dari

masing-masing pengendali P, I dan D dapat saling menutupi dengan menggabungkan ketiganya secara paralel menjadi pengendali proposional plus integral plus diferensial (pengendali PID). Elemen-elemen pengendali P, I dan D masing-masing secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem, menghilangkan *offset* dan menghasilkan perubahan awal yang besar (Gambar 3).



Gambar 3. Blok Diagram Sistem dengan Pengendali PID [4]

Keluaran pengendali PID merupakan penjumlahan dari keluaran

pengendali proporsional, keluaran pengendali integral dan derivatif.

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \dots\dots\dots(1)$$

Fungsi alih pengendali PID [3] :

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s$$

Dengan :

- K_p = konstanta proporsional (*adjustable*)
- T_d = waktu derivatif (*adjustable*)
- T_i = waktu integral (*adjustable*)
- K_i = konstanta integral (*adjustable*)
- K_d = konstanta derivatif (*adjustable*)

Setiap parameter PID, memiliki kelebihan dan kekurangan

masing-masing, hal ini dapat dilihat pada Tabel 1.

Tabel 1. Tanggapan Pengendali PID terhadap Perubahan Konstanta [6]

Closed-Loop Response	Rise Time	Overshoot	Settling Time	SS Error
K_p	Decrease	Increase	Small change	Decrease
K_i	Decrease	Increase	Increase	Eliminate
K_d	Small change	Decrease	Decrease	Small change

Parameter-parameter tersebut tidak bersifat independen, sehingga jika salah satu nilai konstantanya diubah, maka mungkin sistem tidak akan bereaksi seperti yang diinginkan. Tabel 1 hanya dipergunakan sebagai pedoman jika akan melakukan perubahan konstanta.

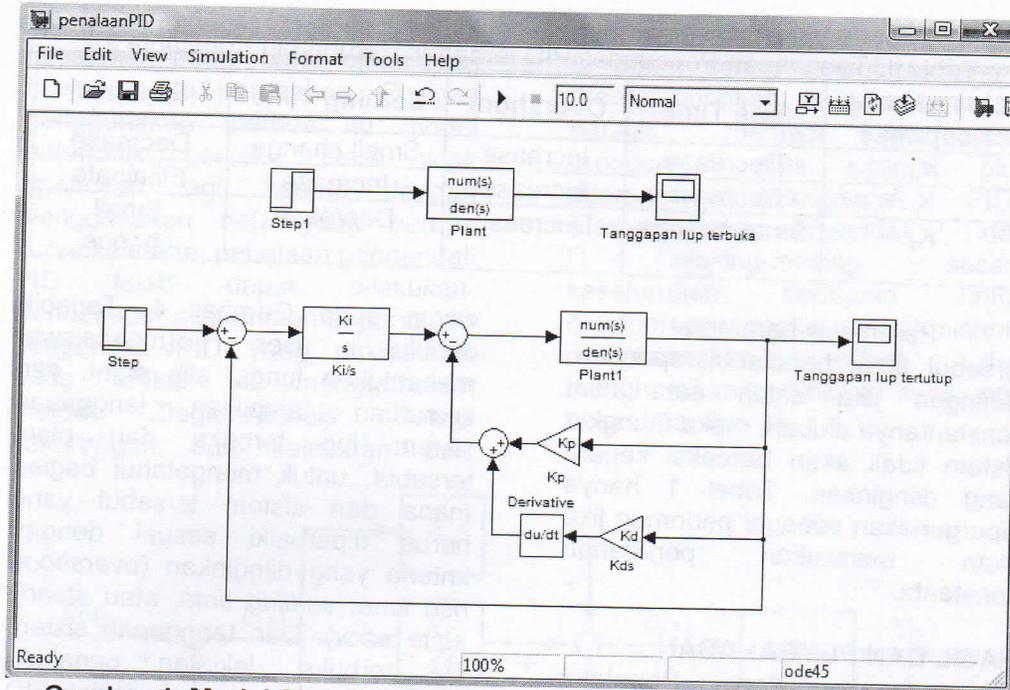
HASIL DAN PEMBAHASAN

Perancangan dan Penalaan Pengendali PID Dengan SIMULINK

Simulink adalah perangkat lunak yang dapat digunakan untuk pemodelan, simulasi, dan analisa sistem dinamik, baik linier maupun nonlinier. Pada pemodelan sistem kendali, simulink dilengkapi dengan *Graphical User Interface* (GUI) untuk membangun model sebagai blok diagram dengan sangat mudah, yaitu dengan pengoperasian mouse secara *click-and-drag* saja.

Model simulink perancangan dan penalaan pengendali PID dapat

dilihat pada Gambar 4. Terlebih dahulu user (mahasiswa) menentukan fungsi alih plant, dan kemudian tampilkan tanggapan sistem lup terbuka dari plant tersebut, untuk mengetahui bagian mana dari sistem tersebut yang harus diperbaiki sesuai dengan kriteria yang diinginkan (*overshoot*, *rise time*, *settling time*, atau *steady state error*). Dari tanggapan sistem lup terbuka, lakukan penalaan konstanta-konstanta pengendali PID dengan berpedoman pada Tabel 1. Penalaan konstanta K_p untuk memperbaiki *rise time*, penalaan konstanta K_d untuk memperbaiki *overshoot*, dan penalaan konstanta K_i untuk menghilangkan *steady state error*. Setelah penentuan konstanta-konstanta selesai, simulasikan model tersebut. Sehingga, kombinasi dari konstanta-konstanta ini akan menghasilkan tanggapan sistem lup tertutup yang diinginkan.

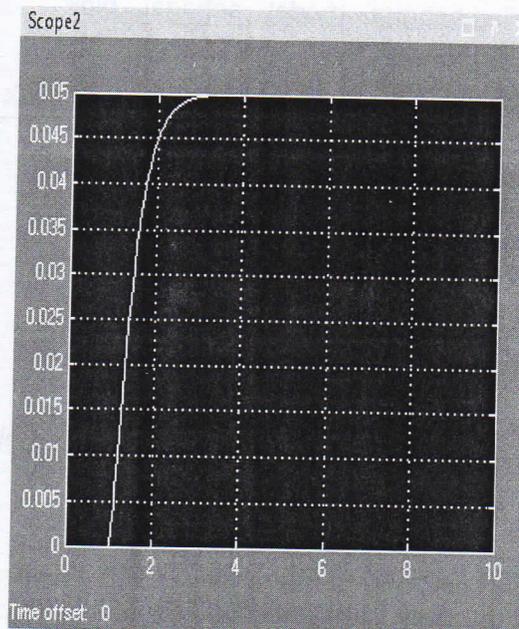


Gambar 4. Model Simulink Perancangan dan Penalaan Pengendali PID

Berikut akan ditampilkan contoh perancangan dan penalaan pengendali PID menggunakan model simulink pada Gambar 4. Diketahui model matematis sistem kendali yang dinyatakan dengan fungsi alih $G(s) = \frac{1}{s^2 - 10s + 20}$. Rancanglah pengendali PID spada sistem kendali tersebut, sehingga diperoleh tanggapan step sistem dengan kriteria berikut [1] :

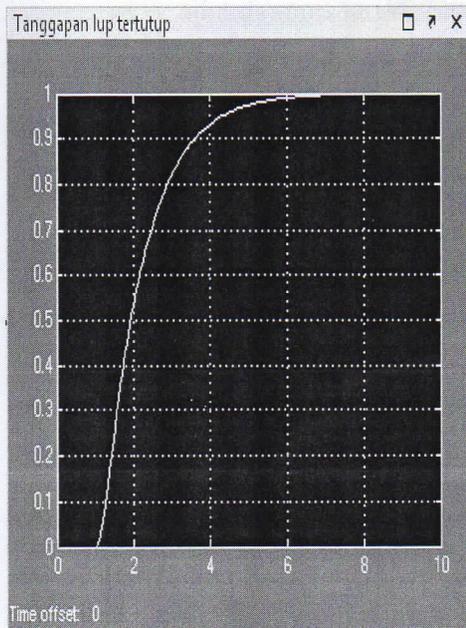
1. Memiliki *rise time* yang cepat
2. *Overshoot* sekecil mungkin
3. Tidak memiliki steady state error.

Perancangan ini dapat dengan mudah dilakukan oleh *user* (mahasiswa) tanpa melakukan perhitungan-perhitungan yang rumit yakni menggunakan prosedur yang telah dijelaskan pada bagian pendahuluan. *User* (mahasiswa) cukup mengetikkan fungsi alih alih sistem yakni : *num* dan *den* pada *command window*. Simulasikan model simulink, dan tampilkan tanggapan sistem lup terbuka (Gambar 5).



Gambar 5. Tanggapan Sistem Lup Terbuka

Kemudian tentukan dan ketikkan nilai-nilai K_p , K_i , dan K_d pada *command window*, dan simulasikan model tersebut kembali, dan tampilkan tanggapan sistem lup tertutup (Gambar 6).



Gambar 6. Tanggapan Sistem Lup Tertutup

Jika tanggapan sistem lup tertutup belum sesuai dengan kriteria yang diinginkan, tentukan lagi konstanta-konstanta pengendali PID tersebut, dan tampilkan kembali tanggapan sistem lup tertutupnya. Setelah sesuai dengan kriteria yang diinginkan, maka proses penalaan pengendali PID selesai. Dengan demikian, saat proses pembelajaran berlangsung mahasiswa tidak menghabiskan waktu untuk melakukan perhitungan matematis yang membutuhkan waktu lama dan pembuatan *script file*, tetapi waktu yang ada lebih banyak digunakan untuk menganalisis perilaku sistem kendali tersebut.

KESIMPULAN

Perancangan dan penalaan pengendali PID dengan

menggunakan model simulink (Gambar 4) dapat dengan mudah dilakukan tanpa melakukan perhitungan atau prosedur yang rumit dan waktu yang lama, walaupun penentuan konstanta-konstantanya masih *trial and error*. Perancangan dan penalaan dilakukan dengan waktu yang relatif singkat karena *user* (mahasiswa) dapat menampilkan tanggapan sistem saat itu juga, tanpa perlu membuat *script file* ataupun *listing program* pada *command window* atau M-File. Selain itu juga, tanggapan sistem dapat ditampilkan untuk input yang berbeda-beda, dengan mengganti input sebelumnya dengan input yang diinginkan secara *click and drag* saja. Tambahan lagi, *plant* dapat juga dinyatakan dalam persamaan ruang keadaan, dengan mengganti blok diagram *plant* tersebut dengan blok diagram persamaan keadaan.

DAFTAR PUSTAKA

- [1] Hastuti (2008). *Rekonstruksi Mata Kuliah Sistem Kendali II*, Jurusan Teknik Elektro FT-UNP, Padang.
- [2] Ogata, Katsuhiko. 1997. *Modern Control Engineering*. 3rd ed, Prentice Hall International.
- [3] Husnaini, Irma dan Hastuti (2009). *Modul Praktikum Sistem Pengaturan*, Jurusan Teknik Elektro FT-UNP, Padang.
- [4] Paz, A. R. 2001. *The Design of PID Controller*. Klipsch School of Electrical and Computer Engineering.
- [5] Ferdinando, Fany. 2009. *Desain PID Controller dengan Software Matlab*, Jurusan Teknik Elektro Universitas Kristen Petra, Surabaya.