

**PERANCANGAN DAN SIMULASI PENERAPAN PENGENDALI PID
PADA SISTEM KENDALI TEGANGAN KELUARAN GENERATOR
PESAWAT N219 DI PT. DIRGANTARA INDONESIA**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

*Diajukan untuk memenuhi persyaratan
Memperoleh gelar Sarjana Teknik*



AZIZUL HAKIM

NIM. 135060300111007

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2018

LEMBAR PENGESAHAN

**PERANCANGAN DAN SIMULASI PENERAPAN PENGENDALI PID
PADA SISTEM KENDALI TEGANGAN KELUARAN GENERATOR
PESAWAT N219 DI PT. DIRGANTARA INDONESIA**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

**Diajukan Untuk Memenuhi Persyaratan
Memperoleh Gelar Sarjana Teknik**



**AZIZUL HAKIM
NIM. 135060300111007**

Skripsi ini telah direvisi dan disetujui oleh dosen pembimbing
Pada tanggal 25 Januari 2018

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Ir. Bambang Siswojo, M.T.
NIP. 1921211198802001

Ir. Purwanto, M.T.
NIP. 195404241986011001

**Mengetahui,
Ketua Jurusan Teknik Elektro**

Ir. Hadi Suyono, S.T., M.T., Ph.D. IPM.
NIP. 19730520 200801 1 013

JUDUL SKRIPSI:
PERANCANGAN DAN SIMULASI PENERAPAN PENGENDALI PID
PADA SISTEM KENDALI TEGANGAN KELUARAN GENERATOR
PESAWAT N219 DI PT. DIRGANTARA INDONESIA

Nama Mahasiswa : AZIZUL HAKIM

NIM : 135060300111007

Program Studi : TEKNIK ELEKTRO

Konsentrasi : TEKNIK KONTROL

Komisi Pembimbing

:

Ketua : Dr. Ir. Bambang Siswojo, M.T.

Anggota : Ir. Purwanto, MT

Tim Dosen Penguji :

Dosen Penguji 1 : Goegoes Dwi Nusantoro, ST., MT.

Dosen Penguji 2 : Dr. Ir. Erni Yudaningtyas, MT.

Dosen Penguji 3 : Ir. Dipl. -Ing M. Rusli

Tanggal Ujian : 8 Januari 2018

SK Penguji : No.SK 18/UN.10.F07/SK/2018

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya dan berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang diteliti dan diulas di dalam Naskah Skripsi ini adalah asli dari pemikiran saya. Tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu Perguruan Tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur

jiplakan, saya bersedia Skripsi dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, pasal 25 ayat 2 dan pasal 70).

Malang, Januari 2018

Mahasiswa,

AZIZUL HAKIM

NIM. 135060300111007

*Teriring Ucapan Terima Kasih kepada:
Ayahanda dan Ibunda Tercinta*

RINGKASAN

Azizul Hakim, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Desember 2017, Perancangan dan Simulasi Penerapan Pengendali PID Pada Sistem Kendali Tegangan Keluaran Generator Pesawat N219 Di PT. Dirgantara Indonesia, Dosen Pembimbing: Bambang Siswojo dan Purwanto.

Generator DC merupakan komponen penting dalam sistem tenaga listrik pada pesawat terutama pesawat N219. Sebelum komponen dipasangkan ke badan pesawat terlebih dahulu komponen diuji kestabilannya termasuk generator pada pesawat N219 yang dilakukan pengujian laboratorium sistem elektrik di PT. Dirgantara Indonesia. Generator DC memiliki *error steady state* 3,5174%, *maximum overshoot* 12,5% dan *settling time* 5 detik. Pengontrolan dengan struktur PID dilakukan agar tegangan *output* generator DC memiliki *error steady state* kurang dari 3,5174%, *settling time* kurang dari 5 detik dan *overshoot* kurang dari 12,5%. Penalaan struktur PID dilakukan oleh metode Penalaan Kontroler menggunakan penala PID – pidTuner. Untuk menentukan jenis kontroler dan konstanta penguatan pada fungsi alih $C(s)$ dapat diperoleh secara cepat, mudah dan akurat. Hal ini bertujuan agar sistem lebih tahan terhadap perubahan gangguan. Dan membuat respon sistem agar menyerupai perilaku respon model referensi.

Kata Kunci: Generator DC, Tegangan *Output*, Struktur PID, PID - pidTuner.

SUMMARY

Azizul Hakim, Department of Electrical Engineering, Faculty of Engineering Universitas Brawijaya, December 2017, Design and Simulation Application of PID Controller on N219 Aircraft Output Voltage Generator Control System In PT. Dirgantara Indonesia, Academic Supervisor: Bambang Siswojo and Purwanto.

The DC generator is an important component in power systems on aircraft, especially N219 aircraft. Before the components are attached to the fuselage, the components are tested for stability including the generator on the N219 aircraft that is done by the laboratory testing of the electrical system at PT. Dirgantara Indonesia. DC generator has steady state error 3,5174%, maximum overshoot 12,5% and settling time 5 second. Control with PID structure is done so that the DC generator output voltage has steady state error less than 3,5174%, settling time less than 5 second and overshoot less than 12,5%. PID structure tuning is done by the Controller Tuning method using the PID-pidTuner tuner. To determine the type of controller and the reinforcing constant on the transfer function $C(s)$ can be obtained quickly, easily and accurately. It aims to make the system more resistant to disturbance changes. And make the system response to resemble the reference model response behavior.

Keywords: DC Generator, Output Voltage, PID Structure, PID - pidTuner.

PENGANTAR

Segala puji syukur bagi Allah SWT berkat segala limpahan rahmat-Nya, penulisan laporan Skripsi berjudul “Perancangan dan Simulasi Penerapan Pengendali PID Pada Sistem Kendali Tegangan Keluaran Generator Pesawat N219 Di PT. Dirgantara Indonesia” dapat diselesaikan dengan baik. Tak lepas shalawat serta salam tercurahkan kepada junjungan kita Nabi Muhammad SAW yang telah menjadi suri tauladan bagi yang mengharapkan rahmat dan hidayah-Nya.

Tak lupa penulis juga mengucapkan banyak terima kasih kepada pihak yang membantu terselesainya laporan skripsi ini.

1. Allah SWT yang telah memberikan kelancaran, kemudahan, ilham dan hidayahnya.
2. Budjang Husein dan Musurkinah Aziz selaku orang tua penulis yang segenap hati mendukung dan mendoakan terselesainya skripsi ini secara moril dan materil.
3. Keluarga tercinta yang segenap hati mendukung dan mendoakan terselesainya skripsi ini.
4. Mely Suriyanti yang telah menjadi adik, teman, sahabat yang selalu memberikan semangat dan doa kepada penulis.
5. Bapak Abdul Kadir, S.Si. selaku atasan Pembimbing Supervisor AET Ground Test System FT-3000 yang sudah berkenan memberikan kesempatan untuk bisa melakukan Kuliah Kerja Nyata - Praktik (KKN-P) dan pelaksanaan penyusunan skripsi.
6. Sdra. Ario Susilo, S.T. selaku pembimbing lapangan yang telah meluangkan waktu untuk memberi pelajaran, masukan dan pengalaman selama pelaksanaan Kuliah Kerja Nyata - Praktik (KKN-P) dan penyusunan skripsi
7. Ir. Hadi Suyono. S.T., M.T., Ph.D. IPM. selaku Ketua Jurusan Teknik Elektro Universitas Brawijaya.
8. Ir. Nurussa'adah, MT. selaku Sekretaris Jurusan Teknik Elektro Universitas Brawijaya.
9. Ali Mustofa, S.T., M.T. selaku Ketua Program Studi Jurusan Teknik Elektro Universitas Brawijaya.
10. Ir. Purwanto, M.T. selaku Ketua Kelompok Dosen Keahlian Teknik Kontrol Jurusan Teknik Elektro Universitas Brawijaya.

11. Dr. Ir. Bambang Siswojo, M.T. dan Ir. Purwanto, M.T. selaku Dosen Pembimbing I dan Dosen Pembimbing II yang telah memberikan bimbingan sehingga skripsi ini dapat terelesaikan.
12. Teman-teman Spectrum 2013 yang selalu memberikan dorongan semangat kepada penulis.
13. Teman-teman Teknik Kontrol 2013 yang selalu memberikan dorongan semangat kepada penulis.
14. Teman-teman keluarga besar IKAPEMA KEPRI–MALANG yang selalu memberikan dorongan semangat dan doa kepada penulis.
15. Teman-teman keluarga ASPURA-ASPURI (Rumah Kita) yang selalu memberikan dorongan semangat dan doa kepada penulis.
16. Teman-teman keluarga ASPURA(Home) yang telah menjadi keluarga yang luar biasa selama penulis berada di Malang serta selalu memberikan dorongan semangat dan doa kepada penulis.
17. Semua pihak yang telah memberikan bantuan serta dukungan baik secara langsung maupun tidak langsung atas penyusunan skripsi ini.

Dalam penyusunan skripsi ini, penulis menyadari bahwa skripsi ini belumlah sempurna, karena keterbatasan ilmu dan kendala–kendala lain yang terjadi selama pengerjaan skripsi ini. Oleh karena itu, penulis berharap kritik dan saran untuk penyempurnaan tulisan di masa yang akan datang. Semoga tulisan ini dapat bermanfaat dan dapat digunakan untuk pengembangan lebih lanjut.

Malang, 8 Desember 2017

Penulis

DAFTAR ISI

| | |
|--|-----|
| PENGANTAR..... | i |
| DAFTAR ISI | iii |
| DAFTAR TABEL | vii |
| DAFTAR GAMBAR..... | ix |
| BAB I..... | 1 |
| PENDAHULUAN | 1 |
| 1.1 Latar Belakang..... | 1 |
| 1.2 Rumusan Masalah..... | 2 |
| 1.3 Batasan Masalah | 3 |
| 1.4 Tujuan | 3 |
| 1.5 Manfaat | 3 |
| BAB II | 5 |
| TINJAUAN PUSTAKA..... | 5 |
| 2.1 Generator Direct Current (DC)..... | 5 |
| 2.2 Kontroler..... | 6 |
| 2.2.1 Kontroler Proporsional | 7 |
| 2.2.2 Kontroler Integral | 8 |
| 2.2.3 Kontroler Diferensial..... | 9 |
| 2.2.4 Kontroler Proporsional Integral Diferensial (PID)..... | 9 |
| 2.3 Desain Kontroler menggunakan pidTuner..... | 10 |
| 2.4 Penalaan Kontroler menggunakan penala PID – pidTune..... | 11 |
| 2.5 Fungsi Alih | 15 |
| 2.6 Analisis Respon Transien | 16 |

| | |
|---|----|
| BAB III..... | 21 |
| METODE PENELITIAN | 21 |
| 3.1 Penyusunan Model Sistem | 21 |
| 3.2 Spesifikasi Desain..... | 22 |
| 3.3 Pemodelan Matematis Sistem..... | 22 |
| 3.3.1 Fungsi Alih dan Blok Diagram Generator DC..... | 23 |
| 3.3.2 Fungsi Alih dan Blok Diagram Sensor Tegangan..... | 25 |
| 3.4 Perancangan Diagram Sistem | 25 |
| 3.5 Perancangan Algoritma | 26 |
| 3.5.1 Desain Menggunakan Matlab..... | 26 |
| 3.5.2 Penalaan Kontroler menggunakan penala PID - pidTuner..... | 28 |
| 3.5.3 Penentuan Model Referensi | 31 |
| 3.5.4 Penetapan Parameter Kontroler..... | 37 |
| BAB IV | 39 |
| PENGUJIAN DAN SIMULASI SISTEM | 39 |
| 4.1 Identifikasi Karakteristik | 39 |
| 4.2 Simulasi Sistem dengan Berbagai Beban | 42 |
| 4.2.1 Simulasi Sistem Dengan Beban 0 A | 42 |
| 4.2.2 Simulasi Sistem Dengan Beban 25 A | 43 |
| 4.2.3 Simulasi Sistem Dengan Beban 50 A | 44 |
| 4.2.4 Simulasi Sistem Dengan Beban 100 A | 45 |
| 4.2.5 Simulasi Sistem Dengan Beban 200 A | 46 |
| BAB V..... | 47 |
| KESIMPULAN DAN SARAN..... | 47 |
| 5.1 Kesimpulan..... | 47 |
| 5.2 Saran..... | 48 |
| DAFTAR PUSTAKA | 49 |

| | |
|----------------|----|
| LAMPIRAN | 51 |
|----------------|----|

DAFTAR TABEL

| | |
|--|----|
| Tabel 2 . 1 Tipe Kontroler | 12 |
| Table 3 . 1 Nilai Parameter Sistem | 22 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2 . 1 Kontruksi Generator | 5 |
| Gambar 2 . 2 Diagram Blok Kontroler Proposional..... | 8 |
| Gambar 2 . 3 Diagram Blok Kontroler Integral | 8 |
| Gambar 2 . 4 Diagram Blok Kontroler Diferensial | 9 |
| Gambar 2 . 5 Diagram Blok Kontroler PID | 10 |
| Gambar 2 . 6 Konfigurasi Kontroler PID 1-DOF secara diagram..... | 13 |
| Gambar 2 . 7 Konfigurasi Kontroler PID 1-DOF secara diagram blok..... | 13 |
| Gambar 2 . 8 Jendela Tampilan menu pidTuner | 14 |
| Gambar 2 . 9 Jendela tampilan Show Parameter | 15 |
| Gambar 2 . 10 Respon Orde Satu Terhadap Masukan Unit Step | 18 |
| Gambar 2 . 11 Respon Orde Dua Terhadap Masukan Unit Step..... | 19 |
| Gambar 3 . 1 Blok Diagram Untai Terbuka | 21 |
| Gambar 3 . 2 Blok Diagram Untai Tertutup..... | 21 |
| Gambar 3 . 3 Blok Sistem Generator..... | 23 |
| Gambar 3 . 4 Blok Subsistem Generator | 23 |
| Gambar 3 . 5 Fungsi Alih Subsistem Blok Generator DC..... | 25 |
| Gambar 3 . 6 Blok Diagram Sistem Kendali | 25 |
| Gambar 3 . 7 Respon Fungsi Alih Plant | 28 |
| Gambar 3 . 8 Jendela Tampilan Menu pidTuner Sebelum Ditala | 29 |
| Gambar 3 . 9 Jendela Tampilan Menu pidTuner Setelah Ditala | 30 |
| Gambar 3 . 10 Jendela Tampilan Show Parameter Setelah Ditala | 30 |
| Gambar 3 . 11 Model Referensi..... | 31 |
| Gambar 3 . 12 Gambar 3 . 12 Blok Diagram Kontroler PID Paralel..... | 32 |
| Gambar 3 . 13 Jendela Penalaan..... | 35 |
| Gambar 3 . 14 Jendela Show Parameter | 36 |
| Gambar 4 . 1 Kurva Karakteristik Plant | 40 |
| Gambar 4 . 2 Karakteristik Sistem Keseluruhan | 41 |
| Gambar 4 . 3 Respon Sistem Dengan Beban 0 A..... | 42 |
| Gambar 4 . 4 Respon Sistem Dengan Beban 25 A..... | 43 |
| Gambar 4 . 5 Respon Sistem Dengan Beban 50 A..... | 44 |
| Gambar 4 . 6 Respon Sistem Dengan Beban 100 A..... | 45 |

Gambar 4 . 7 Respon Sistem Dengan Beban 200 A..... 46

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pesawat merupakan sarana transportasi yang memiliki arti penting bagi pembangunan ekonomi dan pertahanan, mengingat bahwa Indonesia adalah sebuah Negara kepulauan dengan kondisi geografis yang sulit untuk diakses tanpa sarana transportasi yang memadai. Dari kondisi tersebut muncul pemikiran bahwa sebagai sebuah negara kepulauan Indonesia berada dalam posisi untuk memiliki industri maritim dan penerbangan. Hal ini yang mendorong lahirnya industri pesawat terbang di Indonesia.

Dalam perkembangan zaman yang semakin modern, pesawat terbang banyak digunakan sebagai sarana transportasi baik antar kota, pulau, negara maupun benua. Dan Indonesia mampu menciptakan pesawat hasil buah tangan anak bangsa sendiri yaitu pesawat N219 yang menggunakan 2(dua) engine turboprop. Meskipun begitu disebut buatan buah tangan anak bangsa namun komponen-komponen yang digunakan tetap mengimpor dari negara lain sama halnya generator pada pesawat dan lain sebagainya.

Generator adalah sebuah perangkat mesin listrik yang mengubah energi mekanis menjadi energi listrik. Prinsip generator dengan sederhana dikatakan bahwa tegangan diinduksikan pada konduktor apabila konduktor digerakkan pada medan magnet sehingga memotong garis gaya. Generator digerakkan oleh beberapa jenis mekanis seperti uap, turbin air, mesin bensin, atau motor listrik (Petruzella, 2001: 312).

Penelitian ini akan melakukan penerapan rangkaian kendali menggunakan pengendali *Proportional Integral dan Derivatif (PID)* untuk mengontrol tegangan keluaran generator DC. Metode penalaan yang digunakan adalah penalaan controller menggunakan penala PID – *pidtune* yang disimulasikan menggunakan program MATLAB. Dengan harapan pengendalian ini tetap sesuai dengan spesifikasi desain yang ditetapkan oleh perusahaan yang bersangkutan.

Penggunaan pengendali PID ini diharapkan dapat mengurangi *overshoot, error steady state*, mempercepat *settling time* dan dapat tangguh ketika diberikan beban dengan waktu

perbaikan yang cepat dan akurat. Karena kontroler PID merupakan kontroler yang dapat mengurangi *error steady state*, mempercepat respon transien, mengurangi *overshoot* (Xiong, 2007: 2616).

Matlab telah melengkapi sistem penalaan berbasis antarmuka pengguna secara grafis (*GUI, Graphical user interface*) sehingga desainer sistem kontrol PID dapat melakukan pendesainan secara cepat, mudah dan tepat (Siswojo, 2017: 49).

Dengan menggunakan software MATLAB dan bisa memonitoring performansi maupun responnya. Lalu kita juga dapat mengendalikan besar nilai tegangan keluaran yang dihasilkan dari generator. fungsi dari simulasi ini dapat memperlihatkan atau memonitoring besar nilai tegangan keluaran yang dihasilkan dari generator dan dapat dikendalikan sesuai aturan yang berlaku atau yang dibutuhkan pada pesawat N219.

Besar nilai tegangan keluaran yang tidak sesuai akan dimanipulasi atau dikendalikan sedemikian rupa agar sesuai aturan yang berlaku atau yang dibutuhkan pada pesawat. Hal ini dapat dilakukan dengan menggunakan suatu sistem kendali yang menggunakan suatu kontroler yang sesuai demi mendapatkan performansi dan respon yang dibutuhkan.

Laporan skripsi ini menggunakan kontroler PIDF dalam pengaturan besar nilai tegangan keluaran yang akan didistribusikan, dengan harapan dapat mengatur besar nilai tegangan keluaran sesuai dengan yang diinginkan. Saat ini teknologi kontrol PIDF dikenal sebagai kontroler berumpan balik yang paling sering digunakan dalam dunia teknologi. Karena terbukti dapat memberikan performa kontrol yang baik meskipun mempunyai algoritma sederhana tetapi mudah dipahami.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang di atas, dapat disusun rumusan masalah sebagai berikut:

1. Bagaimana respon penalaan controller menggunakan penala PIDF – *pidtune*?
2. Bagaimana merancang sistem kontrol tegangan keluaran pada generator DC pesawat dengan pengendali PIDF?
3. Bagaimana performansi sistem dengan diterapkannya sistem kontrol tegangan keluaran pada generator pesawat N219 dengan pengendali PIDF pada simulasi?
4. Bagaimana performansi sistem dengan diterapkannya pengendali PIDF dan pengaruh gangguan pada sistem?

1.3 Batasan Masalah

Mengacu pada permasalahan pada skripsi ini, maka akan dibatasi pada:

1. Dititikberatkan pada penerapan sistem kontrol tegangan keluaran pada pesawat untuk menjaga besar tegangan keluaran pada nilai tertentu di PT. Dirgantara Indonesia.
2. Fungsi alih dan parameter didapatkan melalui jurnal yang terkait dan sesuai dengan spesifikasi desain pada dokumen-dokumen yang terkait.
3. Besar tegangan keluaran mengacu pada data-data literatur yang membahas obyek penulisan skripsi ini.
4. Menggunakan pengendali PIDF untuk menjaga kestabilan nilai tegangan keluaran. Sistem disimulasikan dengan menggunakan perangkat lunak MATLAB R2015a.
5. Simulasi berlaku pada saat uji laboratorium PT. Dirgantara Indonesia.
6. Kecepatan *engine* konstan pada 7200 Rpm.
7. Gangguan yang diberikan berupa manipulasi beban yang sebenarnya menggunakan data yang menunjang penelitian.
8. Berpicu pada standar yang telah ditetapkan dalam MIL-STD-704F.

1.4 Tujuan

Tujuan yang ingin dicapai dalam skripsi ini adalah untuk menerapkan sistem kontrol tegangan keluaran ini agar besar tegangan keluaran yang di hasilkan yang akan didistribusikan pada pesawat konstan sesuai ketentuan yang di pakai atau berlaku pesawat N219 .

1.5 Manfaat

Manfaat skripsi ini adalah dapat dipahaminya penerapan sistem kontrol tegangan keluaran serta performansinya dengan menggunakan pengendali PIDF.

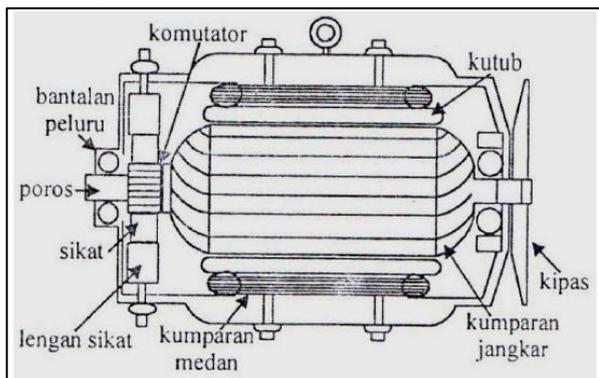
BAB II

TINJAUAN PUSTAKA

2.1 Generator Direct Current (DC)

Generator *Direct Current* (DC) merupakan sebuah piranti elektronik yang mengubah energi mekanis menjadi energi listrik. Energi mekanis di pergunakan untuk memutar kumparan kawat penghantar di dalam medan magnet. Energi yang dipergunakan bisa berbentuk uap, turbin air, mesin bensin, atau motor listrik.

Generator DC memiliki konstruksi yang terdiri atas dua bagian yaitu bagian yang berputar (rotor) dan bagian yang diam (stator). Contoh dari stator adalah rangka, komponen magnet dan komponen sikat. Sedangkan yang termasuk rotor adalah jangkar, kumparan jangkar dan komutator (Ardian, 2010: 6). Secara umum, konstruksi generator DC dapat dilihat pada Gambar 2.1.



Gambar 2. 1 Kontruksi Generator

Sumber : Niko

Prinsip kerja dari suatu generator DC adalah berdasarkan hukum Faraday mengenai induksi elektromagnetik. Hukum Faraday menyebutkan “Apabila suatu konduktor digerakkan dan memotong garis-garis gaya magnetik (fluksi), maka pada konduktor tersebut akan timbul suatu gaya gerak listrik (g.g.l.)” (Soemarwanto, 2010: 79).

Jika kumparan kawat dihubungkan dengan rangkaian listrik tertutup, maka akan timbul pula arus listrik yang mengalir pada rangkaian. Berdasarkan Hukum Induksi Faraday/ Hukum Faraday besarnya g.g.l. yang diinduksikan satu belitan adalah:

$$e = -N \frac{d\phi}{dt} \times 10^{-8} \text{ Volt } (\phi \text{ dalam Maxwell}) \dots\dots\dots(2-1)$$

Untuk N belitan maka:

$$e = -N \frac{d\phi}{dt} \times 10^{-8} \text{ Volt} \dots\dots\dots (2-2)$$

Apabila kecepatan putaran = $\frac{n \text{ putaran}}{\text{menit}}$, maka untuk 1 putaran = $\frac{1}{n}$ menit = $\frac{60}{n}$ detik, konduktor akan memotong P x ϕ Maxwell, sehingga:

$$e_{rata-rata} = \frac{p \cdot \phi}{60/n} \times 10^{-8} \text{ Volt} = \frac{p \cdot n \cdot \phi}{60} \times 10^{-8} \dots\dots\dots (2-3)$$

Karena ada $\frac{z}{a}$ konduktor pada setiap cabang jangkar paralel, maka:

$$E = \frac{z}{a} \frac{p \cdot n \cdot \phi}{60} \times 10^{-8} \text{ Volt} \dots\dots\dots (2-4)$$

$$E = \frac{p}{a} \frac{n}{60} \cdot z \cdot \phi \times 10^{-8} \text{ Volt} \dots\dots\dots (2-5)$$

Dengan:

P = Jumlah kutub.

a = Jumlah cabang jangkar paralel.

n = Jumlah putaran (putaran/menit).

z = Jumlah konduktor di jangkar.

ϕ = Fluksi tiap kutub (Maxwell).

Di dalam generator DC besar p, a, dan z adalah tertentu/ tetap/ konstan. Maka dari persamaan 2-5, apabila $\frac{p}{a} \frac{n}{60} \cdot z \cdot \phi \times 10^{-8} C$ konstan, maka:

$$E = C \cdot n \cdot \phi \text{ Volt} \dots\dots\dots (2-6)$$

Dengan:

E = Tegangan induksi.

C = Konstanta.

n = Jumlah putaran (putaran/menit).

ϕ = Fluksi tiap kutub (Maxwell).

2.2 Kontroler

Sistem pengendalian dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengendalian adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolak ukur kualitas kestabilan sistem dengan menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika dikenai gangguan (Ogata K.,1997). Selain itu analisis juga dilakukan untuk mengetahui bagaimana

kecepatan sistem dalam merespons masukan, dan bagaimana peredaman terhadap adanya lonjakan (*over shoot*).

Suatu sistem dikatakan stabil jika diberi gangguan maka sistem tersebut akan kembali ke keadaan *steady state* di mana *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem dikatakan tidak stabil jika *outputnya* berosilasi terus menerus ketika dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi maka *output* sistem ketika diberi suatu masukan, tidak dapat mengikuti masukan secara serentak, tapi menunjukkan respons transien berupa suatu osilasi teredam sebelum mencapai *steady state*.

Dalam sistem pengendalian terdapat 2 macam *loop*:

1. Pengendalian dengan *loop* terbuka

Sistem kontrol *loop* terbuka adalah sistem kontrol yang keluarannya tidak berpengaruh pada aksi pengontrolan. Jadi pada sistem kontrol *loop* terbuka, keluaran tidak diukur atau diumpan balik untuk dibandingkan dengan masukan.

2. Pengendalian dengan *loop* tertutup

Sistem kontrol *loop* tertutup adalah sistem kontrol yang keluarannya mempunyai pengaruh langsung pada aksi pengontrolan. Disebut juga sistem kontrol yang menggunakan umpan balik untuk memperkecil kesalahan sistem.

2.2.1 Kontroler Proporsional

Kontroler proporsional memiliki keluaran yang sebanding dengan sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya). Untuk kontroler dengan aksi kontrol proporsional, hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ adalah:

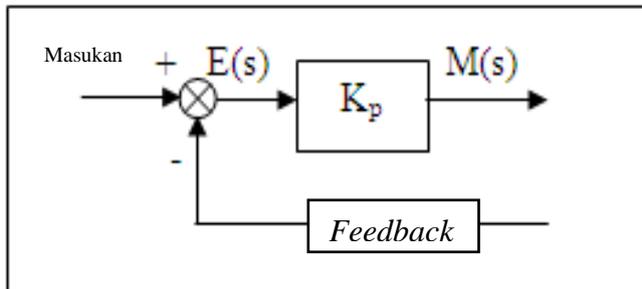
$$m(t) = K_p \cdot e(t) \dots\dots\dots(2-7)$$

atau, dalam besaran transformasi Laplace,

$$\frac{M(s)}{E(s)} = K_p \dots\dots\dots(2-8)$$

Di mana K_p adalah kepekaan proporsional atau penguatan.

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggerak, kontroler proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata K.,1997). Diagram blok kontroler proporsional ditunjukkan dalam Gambar 2.2.



Gambar 2 . 2 Diagram Blok Kontroler Proporsional

Sumber : Ogata K, 1997

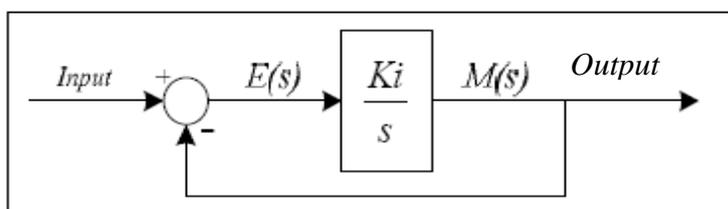
2.2.2 Kontroler Integral

Kontroler integral berfungsi mengurangi kesalahan keadaan mantap pada kontroler proporsional sebelumnya. Pada kontroler dengan aksi integral, harga keluaran kontroler $m(t)$ diubah dengan laju yang sebanding dengan sinyal kesalahan penggerak $e(t)$.

Jadi,

$$\frac{dm(t)}{dt} = Ki \cdot e(t) \dots\dots\dots (2-9)$$

Jika harga $e(t)$ diduakalikan, maka harga $m(t)$ berubah dengan laju perubahan menjadi dua kali semula. Jika kesalahan penggerak nol, maka harga $m(t)$ tetap stasioner. Aksi kontrol integral seringkali disebut kontrol *reset* (Ogata K.,1997). Gambar 2.3 menunjukkan diagram blok kontroler integral.



Gambar 2 . 3 Diagram Blok Kontroler Integral

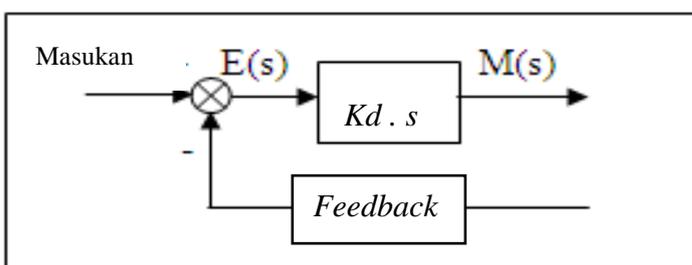
Sumber : Ogata K, 1997

2.2.3 Kontroler Diferensial

Kontroler ini digunakan untuk memperbaiki atau mempercepat respons transien sebuah sistem kontrol dengan cara memperbesar *phase lead* terhadap penguatan kontrol dan mengurangi *phase lag* penguatan tersebut (Ogata K.,1997). Kontroler diferensial tidak dapat mengeluarkan *output* bila tidak ada perubahan masukan, selain itu kontroler differensial tidak dapat digunakan untuk proses yang mengandung *noise*. Hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ adalah:

$$\frac{M(s)}{E(s)} = Kd . s \dots\dots\dots(2-10)$$

Gambar 2.4 menunjukkan diagram blok kontroler diferensial.



Gambar 2 . 4 Diagram Blok Kontroler Diferensial

Sumber : Ogata K, 1997

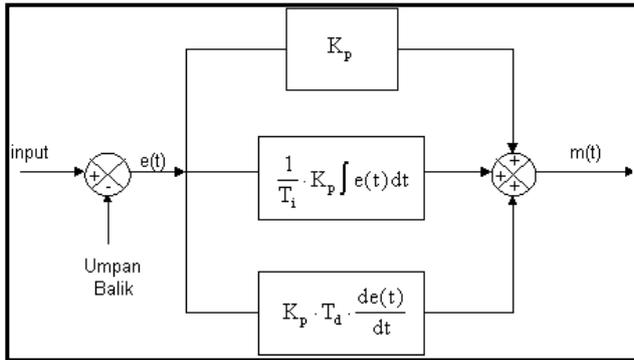
2.2.4 Kontroler Proporsional Integral Diferensial (PID)

Gabungan aksi kontrol proporsional, integral, dan diferensial mempunyai keunggulan dapat saling menutupi kekurangan dan kelebihan dari masing-masing kontroler. Elemen-elemen P, I, dan D masing-masing secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem, menghilangkan *offset* dan menghasilkan perubahan awal yang besar. Persamaan kontroler PID ini dapat dinyatakan sebagai berikut:

$$m(t) = Kp . e(t) + \frac{Kp}{Ti} . e(t)dt + Kp . Td \frac{de(t)}{dt} \dots\dots\dots(2-11)$$

Dalam transformasi Laplace dinyatakan sebagai berikut:

$$\frac{M(s)}{E(s)} = Kp (1 + \frac{1}{Ti . s} + Td . s) \dots\dots\dots(2-12)$$



Gambar 2 . 5 Diagram Blok Kontroler PID

Sumber : Ogata K, 1997

2.3 Desain Kontroler menggunakan pidTuner

Ketika fungsi alih plant semakin kompleks, proses desain sistem kontrol secara manual akan menjadi semakin rumit. Matlab menyediakan aplikasi yang memiliki kemampuan melakukan penalaan kontroler PID berdasarkan fungsi alih plantnya secara otomatis. Aplikasi fungsi tersebut adalah *pidtune* yang digunakan bersama fungsi *pidTuner*. Fungsi *pidtune* untuk memprediksi jenis kontroler untuk fungsi alih plant dengan jenis kontroler kombinasi dari P,I dan D. jika variabel P mendefinisikan fungsi alih plant, maka untuk memprediksi jenis kontroler PID harus diberikan perintah:

```
1 C=pidtune(P, 'pid')
```

Variabel C adalah jenis kontroler yang diprediksi berikut konstantanya yaitu K_p dan K_i . Jika diinginkan untuk memprediksi jenis kontroler yang lainnya misalnya PID, maka argumen harus dimasukkan ke parameter kedua. Fasilitas ini dapat juga digunakan untuk mendesain sistem kontrol secara manual. Untuk memprediksi parameter awal kontroler guna mendapatkan kestabilan sistem.

Selanjutnya desain dilanjutkan dengan penalaan kontroler dengan mengeksekusi fungsi *pidTuner* dengan perintah;

```
1 pidTuner(P, C)
```

Parameter pertama adalah fungsi alih plant dengan variabel P, parameter kedua adalah kontroler kontroler hasil prediksi. Dengan mengeksekusi fungsi tersebut. Penalaan dapat dilakukan dengan sangat mudah dan cepat untuk mendapatkan respon yang diinginkan.

2.4 Penalaan Kontroler menggunakan penala PID – pidTune

Untuk menentukan jenis kontroler dan konstanta penguatan pada fungsi alih $C(s)$ dapat diperoleh secara cepat, mudah dan akurat, dengan menggunakan aplikasi-aplikasi yang tersedia di Matlab. (Siswojo,2017: 45).

pidTune – fungsi digunakan untuk menala kontroler dari fungsi alih untai maju dengan umpan balik satu (*unity feedback*).

Berdasarkan model plant berupa fungsi alih, pidtune secara otomatis dapat melakukan penalaan masing-masing penguatan kontroler PID untuk menyeimbangkan unjuk kerja (respon waktu) dan ketegaran (margin kestabilan). Dapat dipilih berbagai variasi dari konfigurasi kontroler PID dan dapat memberikan spesifikasi respon waktu dan margin fasa yang diinginkan. Perlu dicatat bahwa menaikkan unjuk kerja secara tipikal akan mengurangi ketegaran dan sebaliknya.

c = pidtune(G, TYPE) – menala kontroler PID untuk jenis SISO (*single masukan single output*) dari plant G. Dapat dispesifikasikan jenis sistem linier untuk G. Variabel c adalah fungsi alih controller hasil penalaan. Argumen string TYPE untuk memberikan spesifikasi jenis kontroler seperti yang di perlihatkan dalam tabel

pidtune – mengembalikan sebuah objek c PID/PID2 dengan waktu sampling yang sama sebagai G. Jika G adalah larik dari model LTI (*linier time-invariant*), **pidtune** mendesain sebuah kontroler untuk masing-masing model plant dan mengembalikannya dalam bentuk objek larik dari PID/PID2.

C = pidtune(G, C0) – memaksa C supaya cocok dengan struktur dari objek C0 dan PID, PID2, PIDSTD atau PIDSTD2. Hasil C akan berjenis sama, bentuk, dan formula integrator/derivatifnya C0. Sebagai contoh, untuk menala sebuah waktu diskrit kontrolerPI dalam bentuk standar dengan waktu sampling 0,1 dan formula trapezoidal.

C0 = pidstd(1, 1, 'Ts', 0.1, IFormula', 'T') dan **C = pidtune(C,C0,WC)** – menspesifikasikan sebuah nilai target WC (dalam *rad/TimeUnit* relatif terhadap *time unit* dari G) untuk frekuensi crossover penguatan 0dB dari respon untai terbuka $L = L \cdot G$. secara tipikal, WC kurang lebih men-set lebar pita kontrol untuk meningkatkan respon kecepatan dan mengurangi WC untuk memperbaiki stabilitas. Ketika dihilangkan, WC akan dipilih secara otomatis berdasarkan dinamika plant.

$C = \text{pidtune}(G, \dots, \text{OPTIONS})$ – menspesifikasikan pilihan penalaan tambahan seperti margin fasa yang ditargetkan. Penggunaan perintah `PIDTUNESOPTIONS` mengatur pilihan dari singkatan `OPTIONS`.

$[C, \text{INFO}] = \text{pidtune}(G, \dots)$ – juga mengembalikan sebuah stuktur `INFO` dengan informasi tentang stabilitas untai tertutup, pemilihan penguatan frekuensi crossover, dan margin fasa sebenarnya.

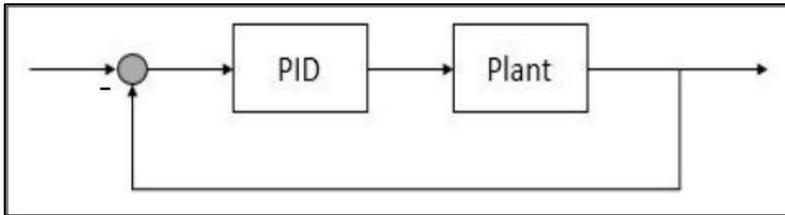
Tabel 2. 1 Tipe Kontroler

| TYPE | Fungsi alih |
|---------|---|
| 'P' | Kontroler P saja |
| 'I' | Kontroler I saja |
| 'PI' | Kontroler PI |
| 'PD' | Kontroler PD |
| 'PID' | Kontroler PID |
| 'PDF' | Kontroler PD dengan filter orde pertama derivative |
| 'PIDF' | Kontroler PID dengan filter orde pertama derivative |
| 'PI2' | Kontroler 2-dof PI |
| 'PD2' | Kontroler 2-dof PD |
| 'PDF2' | Kontroler 2-dof PD dengan filter orde pertama derivative |
| 'PID2' | Kontroler 2-dof PID |
| 'PIDF2' | Kontroler 2-dof PID dengan filter orde pertama derivative |
| 'I-PD' | Kontroler 2-DOF PID dengan $b = 0, c = 0$ |
| 'I-PDF' | Kontroler 2-DOF PID dengan filter orde pertama derivatif dan $b = 0, c = 0$ |
| 'ID-F' | Kontroler 2-DOF PID dengan $b = 0, c = 1$ |
| 'IDF-P' | Kontroler 2-DOF PID dengan filter orde pertama derivatif dan $b = 0, c = 1$ |
| 'PI-D' | Kontroler 2-DOF PID dengan $b = 1, c = 1$ |
| 'PI-DF' | Kontroler 2-DOF PID dengan filter orde pertama derivatif dan $b = 1, c = 1$ |

Matlab telah melengkapi sistem penalaan berbasis antarmuka pengguna secara grafis (*GUI, Graphical user interface*) sehingga desainer sistem kontrol PID dapat melakukan pendesainan secara cepat, mudah dan tepat. (Siswojo,2017: 45).

pidTuner – mengesekusi aplikasi interaktif berbasis GUI, untuk mendesain kontroler PID secara grafis.

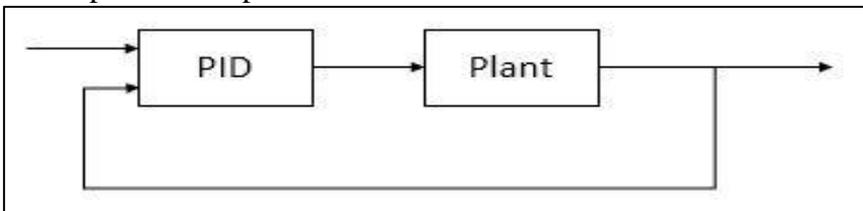
pidTuner – berfungsi untuk mendesain kontroler PID tipe 1-DOF atau 2-DOF. Penalaan kontroler PID tipe 1-DOF, konfigurasi sistem kontrol yang digunakan oleh penala PID diperlihatkan pada Gambar 2.6.



Gambar 2 . 6 Konfigurasi Kontroler PID 1-DOF secara diagram

Sumber : siswojo,2017: 45

Sedangkan kontroler 2-DOF, konfigurasi sistem kontrol yang digunakan oleh penala PID diperlihatkan pada Gambar 2.7.



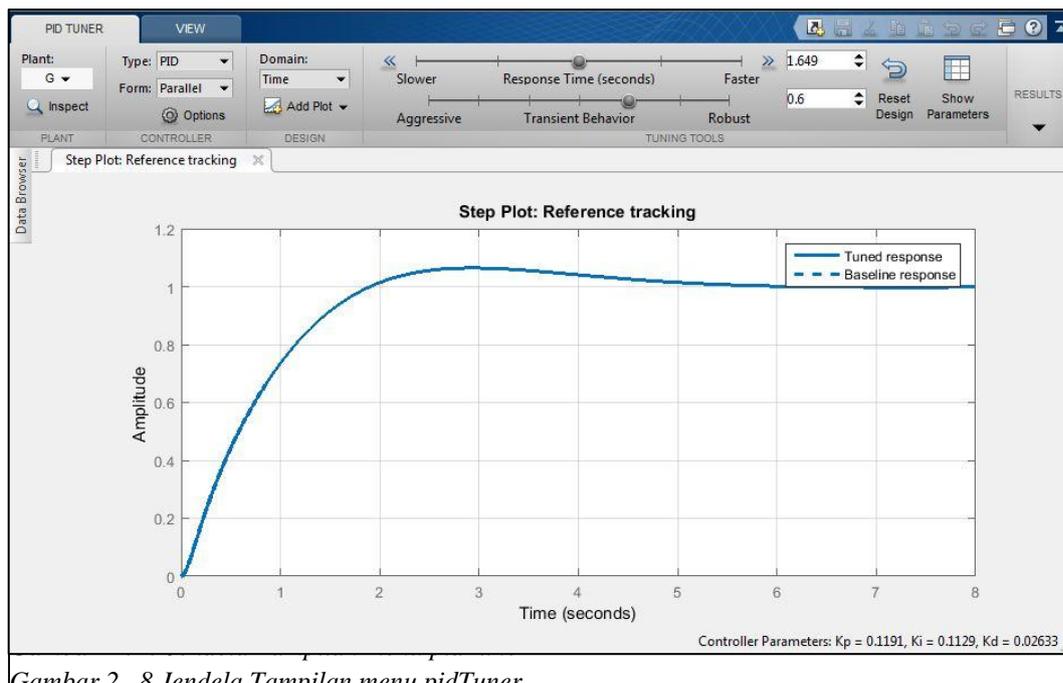
Gambar 2 . 7 Konfigurasi Kontroler PID 2-DOF secara diagram blok

Sumber : siswojo,2017: 45

pidTuner(SYS, TYPE) – mendesain sebuah kontroler PID untuk plant SYS. SYS adalah sistem SISO LTI seperti fungsi TF, ZPK, SS, FRD, atau sebuah model linier dihasilkan oleh aplikasi IDTF, IDFRD, IDGREY, IDPOLY, IDPROC dan IDSS. TYPE mendefinisikan jenis kontroler, dan dapat menggunakan salah satu dari yang diperlihatkan dalam tabel 2.1 Untuk sistem diskrit SYS, kontroler PID memiliki waktu sampling yang sama dengan SYS.

pidTuner(SYS, C) – mengambil sistem LTI sebagai kontroler panduan, selanjutnya digunakan sebagai pembanding unjuk kerja setelah dilakukan penalaan. Jika C adalah sebagai obyek PID, PIDSTD, PID2 atau PIDSTD2. Desain kontroler adalah jenis yang sama, bentuk dan metode diskritisasi sebagai C. C dapat juga sebagai sistem SS, TF, atau ZPK.

Ketika SYS adalah (1) sistem FRD atau (2) sistem SS yang memiliki waktu tunda (*delay*) secara internal dan tidak dapat diubah kedalam sistem ZPK, penala PID mengasumsikan bahwa plant tidak memiliki pole yang tidak stabil, jika terdapat pole yang tidak stabil, harus membuka dialog impor sistem linier sesudah penalaan PID diaktifkan dan mengimpor SYS dengan jumlah pole tidak stabil yang dispesifikasikan dalam dialognya. Jendela tampilan menu pidTuner dapat dilihat pada Gambar 2.8



Gambar 2.8 Jendela Tampilan menu pidTuner

Plant – menu daftar, dengan menu ini dapat memilih plant yaitu (1) vairabel yang telah teridefinisi, (2) membuat baru yaitu mengimpor atau mengidentifikasi plant baru.

1. **Type** – menu daftar untuk pemilihan jenis kontroler yaitu konfigurasi kontroler seperti yang terlihat pada table.
2. **Form** – menu daftar untuk pemilihan model kontroler standar atau parallel
3. **Domain** – menu daftar untuk pemilihan domain waktu dan frekuensi.
4. **Response Time** – menu geser untuk menala waktu respon sistem kontrol, ketika ditala grafik respon akan berubah secara realtime, dapat juga secara langsung memasukkan nilainya.
5. **Transient Behavior** – menu geser, untuk menala perilaku transiennya, ketika ditala grafik respon akan berubah secara realtime, dapat juga secara langsung memasukkan nilainya.
6. **Reset Design** – menu klik untuk mengembalikan penalaan seperti semula.

7. **Show Parameter** – menu klik yang menampilkan parameter kontroler, unjuk kerja dan ketegaran (*robustness*). Dapat dilihat pada gambar 2.9.

Export – menu yang menyediakan untuk (1) mengekspor kontroler atau plant ke ruang kerja (*workspace*) Matlab. Selanjutnya fungsi alih dapat digunakan di konsol Matlab, (2) menyimpan sebagai acuan penalaan.

| Controller Parameters | | |
|-----------------------|---------|----------|
| | Tuned | Baseline |
| Kp | 0.67148 | 0.11909 |
| Ti | 1.1187 | 0.1129 |
| Td | 0.2078 | 0.026335 |
| N | | |

| Performance and Robustness | | |
|----------------------------|---------------------|-----------------------|
| | Tuned | Baseline |
| Rise time | 0.229 seconds | 1.31 seconds |
| Settling time | 1.65 seconds | 4.74 seconds |
| Overshoot | 2.97 % | 6.43 % |
| Peak | 1.03 | 1.06 |
| Gain margin | Inf dB @ Inf rad/s | Inf dB @ Inf rad/s |
| Phase margin | 73 deg @ 6.37 rad/s | 73.8 deg @ 1.21 rad/s |
| Closed-loop stability | Stable | Stable |

Gambar 2. 9 Jendela tampilan Show Parameter

2.5 Fungsi Alih

Dalam teori kontrol, fungsi yang disebut “fungsi alih” seringkali digunakan untuk mencirikan hubungan masukan-keluaran dari sistem linier parameter konstan. Konsep fungsi alih hanya digunakan pada sistem linier parameter konstan, walaupun dapat diperluas untuk suatu sistem kontrol nonlinier.

Fungsi alih sistem linier parameter konstan didefinisikan sebagai bandingan dari transformasi Laplace keluaran (fungsi respon) dan transformasi Laplace masukan (fungsi penggerak), dengan anggapan bahwa semua syarat awal adalah nol.

Tinjau sistem linier parameter konstan yang didefinisikan persamaan differensial berikut:

$$a_0y^{(n)} + a_1y^{(n-1)} + \dots + a_{n-1}\dot{y} + a_ny = b_0x^{(m)} + b_1x^{(m-1)} + \dots + b_{m-1}\dot{x} + b_mx \quad (n \geq m) \dots\dots\dots(2-13)$$

di mana y adalah keluaran sistem dan x adalah masukan. Fungsi alih dari sistem ini diperoleh dengan mencari transformasi Laplace dari kedua ruas persamaan (2-13), dengan menganggap bahwa semua syarat awal adalah nol

$$\text{Fungsi alih} = G(s) = \frac{Y(s)}{X(s)} = \frac{b_0s^m + b_1s^{m-1} + \dots + b_{m-1}s + b_m}{a_0s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n} \dots\dots\dots (2-14)$$

Fungsi alih adalah suatu ekspresi yang merealisasikan keluaran dan masukan suatu sistem linier parameter konstan dalam bentuk parameter sistem dan merupakan sifat dari sistem itu sendiri, tidak bergantung pada fungsi masukan dengan keluaran, meskipun demikian, fungsi alih tidak memberikan informasi mengenai struktur fisik dari sistem.

Dengan menggunakan konsep ini, kita dapat menyatakan dinamika sistem dengan beberapa persamaan aljabar dalam s . pangkat tertinggi s penyebut fungsi alih sama dengan orde suku turunan tertinggi dari keluaran. Jika pangkat tertinggi dari s tersebut adalah n , maka sistem tersebut disebut sistem orde ke n (Ogata, 1997).

2.6 Analisis Respon Transien

Langkah pertama dalam menganalisis sistem kontrol adalah menurunkan model matematik sistem. Setelah model diperoleh, kita dapat menggunakan berbagai metoda untuk analisis performansi sistem(system performance).

Dalam praktek, sinyal masukan sistem kontrol tidak dapat diketahui sebelumnya, tetapi mempunyai sifat acak, sehingga masukan sesaat tidak dapat dinyatakan secara analitis. Hanya pada beberapa kasus khusus sinyal masukan dapat diketahui terlebih dahulu sehingga dapat dinyatakan secara analitis atau dengan kurva, seperti kasus kontrol otomatis pada perkakas potong.

Dalam menganalisis dan mendisain sistem kontrol, kita harus mempunyai suatu dasar perbandingan performansi berbagai sistem kontrol. Dasar ini dapat disusun dengan menetapkan sinyal-sinyal uji tertentu dan membandingkan respon berbagai sistem terhadap sinyal-sinyal masukan ini.

Beberapa kriteria disain didasarkan pada sinyal-sinyal semacam itu atau pada respon sistem terhadap perubahan syarat awal (tanpa suatu sinyal uji). Penggunaan sinyal uji dapat dibenarkan karena ada suatu korelasi antara karakteristik sistem terhadap sinyal masukan uji tertentu dan kemampuan sistem untuk mengikuti sinyal masukan yang sebenarnya.

Sinyal uji yang khas, sinyal masukan sinyal masukan uji (test input signals) yang biasa digunakan adalah fungsi tangga, fungsi “ramp”, fungsi percepatan, fungsi impuls, fungsi

sinusoida dan sebagainya. Dengan sinyal uji ini dapat dilakukan analisis matematik dan eksperimental sistem kontrol secara mudah karena sinyal-sinyal ini merupakan fungsi waktu yang sangat sederhana.

Jenis sinyal masukan yang akan digunakan untuk menganalisis karakteristik sistem diantara sinyal-sinyal masukan khas ini, dapat ditentukan dari bentuk masukan yang paling sering akan diberikan ke sistem pada operasi normal. Jika masukan sistem kontrol merupakan fungsi waktu yang berangsur-angsur berubah, maka fungsi waktu “ramp” mungkin merupakan sinyal uji yang baik. Demikian pula, jika sistem dikenai gangguan secara tiba-tiba, maka fungsi waktu tangga (step) mungkin merupakan sinyal uji yang baik mungkin adalah sinyal impuls. Setelah sistem kontrol didisain berdasarkan sinyal uji, performansi sistem dalam memberikan respon terhadap masukan yang sebenarnya biasanya memuaskan. Penggunaan sinyal uji memungkinkan kita untuk membandingkan performansi semua sistem dengan basis yang sama.

Respon transien dan respon steady state, Respon waktu sistem kontrol terdiri dari dua bagian: respon transien dan respon keadaan mantap(steady state). Yang dimaksud dengan respon transien adalah respon sistem yang berlangsung dari keadaan awal sampai keadaan akhir. Untuk respon keadaan mantap, kita maksudkan sebagai perilaku keluaran sistem jika τ mendekati tak terhingga.(ogata,1985:222).

Karakteristik Respon Sistem Orde Satu

Fungsi alih sistem orde satu dinyatakan sebagai berikut:

$$\frac{C(s)}{R(s)} = \frac{K}{\tau s + 1} \dots\dots\dots(2-15)$$

Dimana :

K = Gain Overall

τ = Konstanta Waktu

Untuk masukan sinyal *unit step*, , transformasi Laplace dari sinyal masukan . Maka, respon keluaran sistem orde satu dengan memasukkan sinyal *step* dalam kawasan *s* adalah:

$$C(s) = \frac{K/\tau}{s(s+1/\tau)} = \frac{A}{s} + \frac{B}{(s+1/\tau)} \dots\dots\dots(2-16)$$

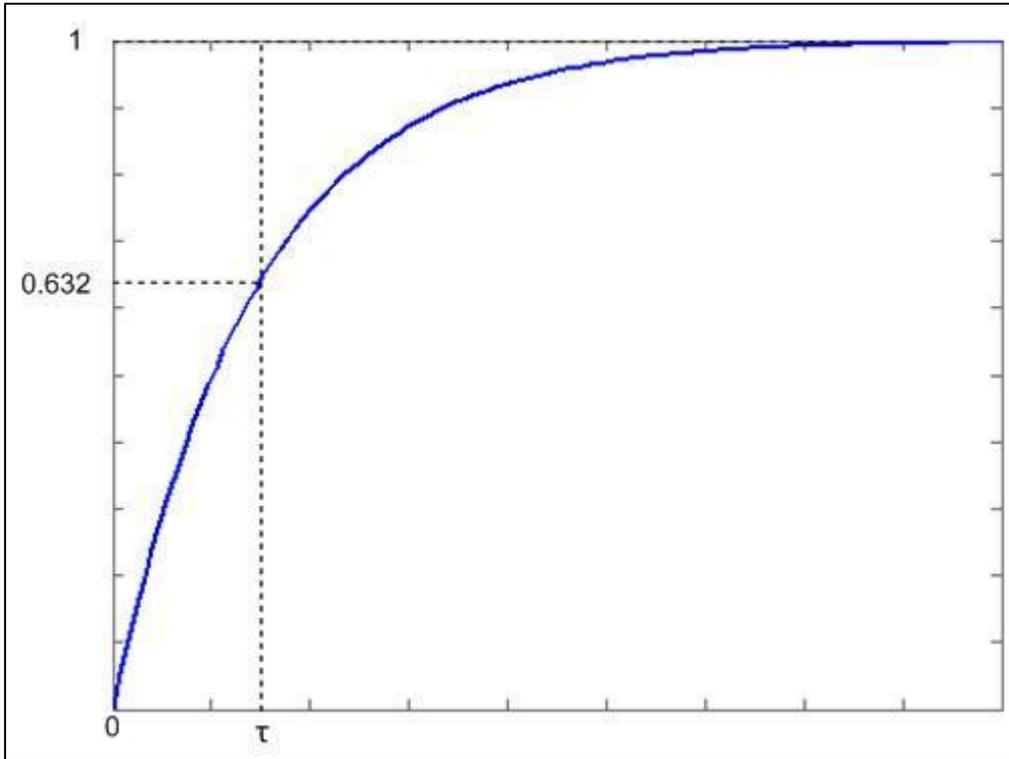
$$A = \frac{K/\tau}{s(s+1/\tau)} \Big|_{s=0} = K \dots\dots\dots(2-17)$$

$$A = \frac{K/\tau}{s} \Big|_{s=1/\tau} = -K \dots\dots\dots(2-18)$$

Dengan menggunakan inversi transformasi Laplace diperoleh respon dalam kawasan waktu yang dinyatakan dalam persamaan berikut:

$$c(t) = K \left(1 - e^{-\frac{t}{\tau}} \right), t \geq 0 \dots\dots\dots (2-19)$$

Kurva respon orde satu untuk masukan sinyal unit *step* ditunjukkan pada Gambar 2.10.



Gambar 2. 10 Respon Orde Satu Terhadap Masukan Unit Step

Sumber : Ogata :1997

Ketika diberi masukan *unit step*, keluaran sistem $c(t)$ mula-mula adalah nol dan terus naik hingga mencapai nilai K. salah satu karakteristik sistem orde satu adalah ketika nilai $t = \tau$, yaitu ketika nilai keluaran mencapai 63,2% dari nilai akhirnya.

Karakteristik Respon Sistem Orde Dua

Persamaan umum sistem orde dua dinyatakan oleh persamaan berikut

$$\frac{C(s)}{R(s)} = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \dots\dots\dots (2-20)$$

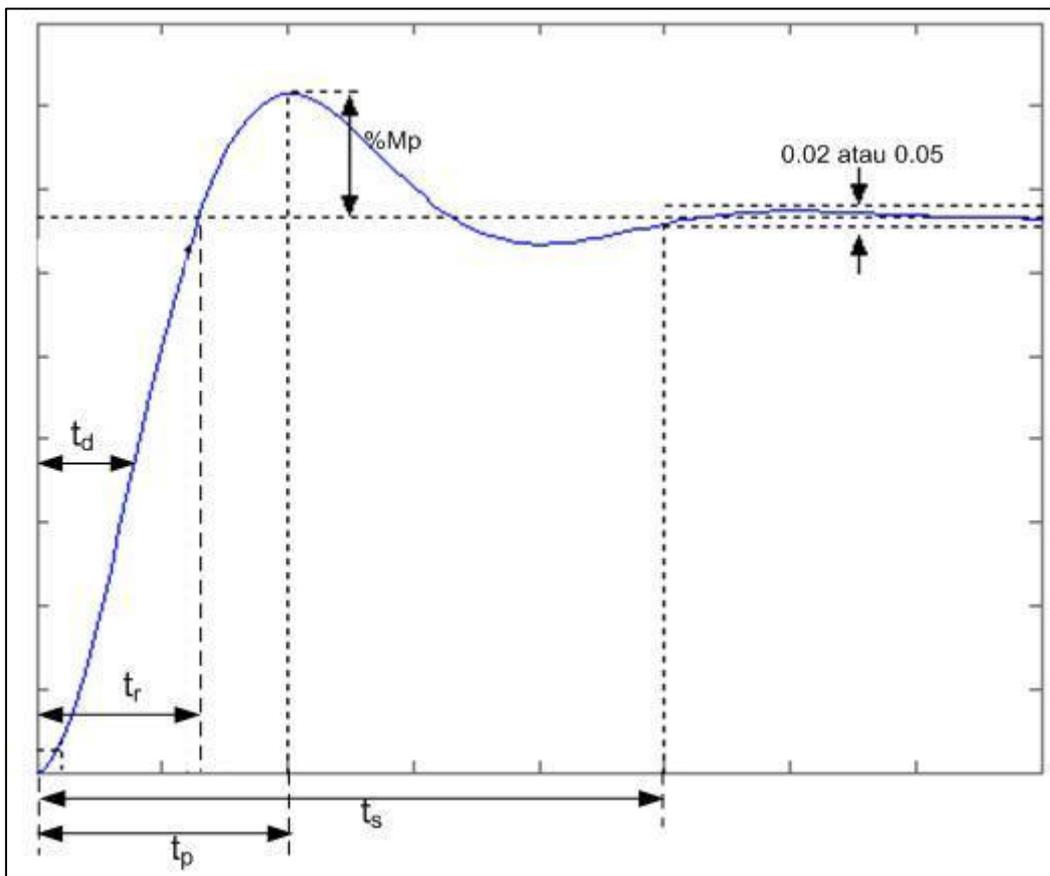
Dimana:

K = Gain

ζ = Ratio Redaman

ω_n = Frekuensi Alami

Bentuk umum kurva respon orde dua untuk masukan sinyal *unit step* ditunjukkan pada Gambar 2.11



Gambar 2 . 11 Respon Orde Dua Terhadap Masukan Unit Step

Sumber : Ogata : 1997

Dari grafik di atas diketahui karakteristik keluaran sistem orde dua terhadap masukan *unit step*, yaitu:

1. Waktu tunda (*delay time*), τ_d

Ukuran waktu yang menyatakan faktor keterlambatan respon output terhadap input, diukur mulai $t = 0$ s/d respon mencapai 50% dari respon *steady state*. Persamaan berikut menyatakan besarnya waktu tunda dari respon orde dua.

$$\tau_d = \frac{0,742}{\zeta\omega_n} \dots\dots\dots(2-21)$$

2. Waktu naik (*rise time*), τ_r

Waktu naik adalah ukuran waktu yang di ukur mulai dari respon $t=0$ sampai dengan respon memotong sumbu *steady state* yang pertama. Besarnya nilai waktu naik dinyatakan pada persamaan berikut:

$$\tau_r = \frac{1}{\omega_n \sqrt{1-\zeta^n}} \left(\pi \tan^{-1} \frac{\sqrt{1-\zeta^n}}{\zeta} \right) \dots\dots\dots (2-22)$$

3. Waktu puncak (*peak time*), τ_p

Waktu puncak adalah waktu yang diperlukan respon mulai dari $t=0$ hingga mencapai puncak pertama *overshoot*. Waktu puncak dinyatakan pada persamaan berikut:

$$\tau_p = \frac{1}{\omega_n \sqrt{1-\zeta^2}} \dots\dots\dots (2-23)$$

4. *Overshoot* maksimum, M_p

Nilai relatif yang menyatakan perbandingan antara nilai maksimum respon (*overshoot*) yang melampaui nilai *steady state* dibanding dengan nilai *steady state*. Besarnya persen overshoot dinyatakan dalam persamaan berikut:

5. Waktu tunak (*settling time*), τ_s

Waktu tunak adalah ukuran waktu yang menyatakan respon telah masuk $\pm 5\%$, atau $\pm 2\%$, atau $\pm 0.5\%$ dari keadaan *steady state*, dinyatakan dalam persamaan berikut:

$$\tau_s(\pm 5\%) = \frac{3}{\zeta \omega_n} \quad \tau_s(\pm 2\%) = \frac{4}{\zeta \omega_n} \quad \tau_s(\pm 0,5\%) = \frac{3}{\zeta \omega_n} \quad \dots\dots\dots (2-24)$$

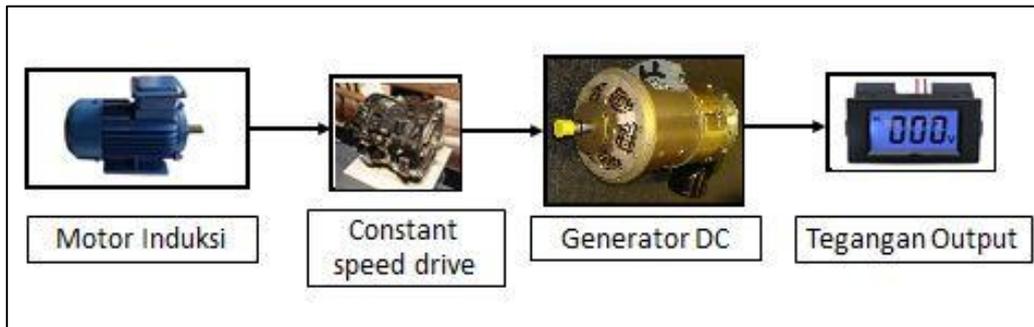
BAB III

METODE PENELITIAN

Metode penelitian pada dasarnya merupakan cara ilmiah untuk mendapatkan informasi dengan tujuan dan manfaat tertentu. Dalam menyelesaikan rumusan masalah dan merealisasikan tujuan penelitian yang terdapat di bab pendahuluan maka diperlukan langkah-langkah untuk menyelesaikan masalah tersebut. Metode penelitian pada skripsi ini meliputi:

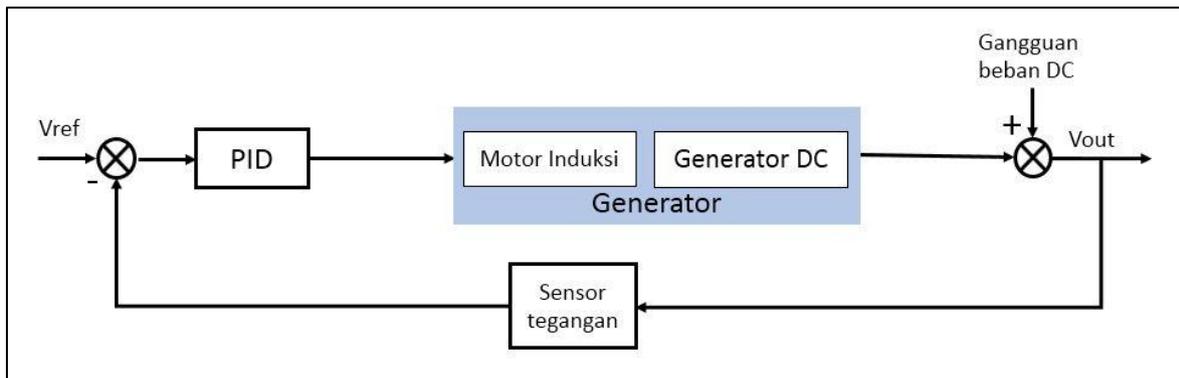
3.1 Penyusunan Model Sistem

Untuk blok diagram sistem untai terbuka dapat dilihat pada Gambar 3.1.



Gambar 3 . 1 Blok Diagram Untai Terbuka

Sedangkan untuk blok diagram sistem untai tertutup dapat dilihat pada Gambar 3.2.



Gambar 3 . 2 Blok Diagram Untai Tertutup

Langkah-langkah yang dilakukan untuk mendapatkan model sistem adalah sebagai berikut:

1. Menentukan sistem yang akan diteliti dan komponen-komponen yang menyusun sistem tersebut sesuai data yang didapatkan.

2. Setelah diketahui komponen-komponen penyusun sistem beserta parameter-parameternya, maka sistem dapat dimodelkan menjadi sebuah persamaan matematis yang sesuai. Dengan memasukkan data-data sistem pada persamaan matematis tersebut, dan dengan transformasi Laplace, maka didapatkan fungsi alih sistem.

3.2 Spesifikasi Desain

Spesifikasi desain yang diinginkan pada perancangan dan simulasi sistem yaitu:

1. *Error steady state* $< 3,5174\%$, karena sistem yang baik memiliki respon dengan batas nilai akhir $3,5174\%$ dari *setpoint*.
2. *Maximum overshoot* $< 12.5\%$, karena jika melebihi batas *overshoot* pada nilai tegangan bisa membahayakan sistem kelistrikan pesawat secara keseluruhan.
3. *Settling time* < 7 detik, karena *settling time* diharapkan lebih cepat daripada sebelum diberi kontroler yaitu < 7 detik.
4. *Recovery time* < 10 detik, karena *Recovery time* diharapkan lebih cepat agar sistem dapat kembali dengan semula.

Table 3. 1 Nilai Parameter Sistem

| Parameter | Nilai | Minimum | Maksimum |
|-----------|---------|---------|----------|
| K_A | 20.0000 | 10.0000 | 400.0000 |
| T_A | 0.0600 | 0.0200 | 0.1000 |
| K_E | 1.0000 | - | - |
| T_E | 0.2000 | - | - |
| K_G | 0.8000 | 0.7000 | 1.0000 |
| T_G | 1.5000 | 1.0000 | 2.0000 |
| K_R | 1.0000 | - | - |
| T_R | 0.0250 | - | - |

3.3 Pemodelan Matematis Sistem

Pemodelan (*modelling*) merupakan deskripsi dari prediksi kelakuan sistem atau komponen. Model matematis sistem diperlukan untuk simulasi yang bertujuan untuk mendapatkan respon dinamik sistem.

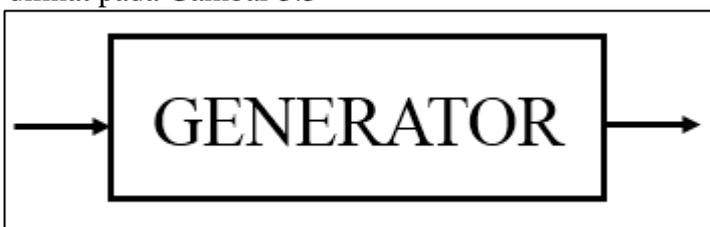
Jika model matematis sistem telah diketahui, maka grafik tanggapan sistem dapat diketahui pula dengan jalan melakukan simulasi terhadap model sistem tersebut. Kemudian langkah berikutnya adalah proses validasi. Pada proses ini data hasil simulasi tersebut

dibandingkan dengan data yang berasal dari proses yang sebenarnya. Jika model tersebut telah cocok (*valid*), proses dapat dilanjutkan ke arah desain dan *prototype*.

Proses penurunan model matematis sistem, terdapat tiga macam cara pendekatan yang dapat dipergunakan. Yang pertama adalah menyajikan model matematis sistem dengan menggunakan persamaan diferensial. Yang kedua adalah dengan menggunakan pendekatan fungsi alih (*transfer function*). Yang ketiga adalah pendekatan ruang keadaan (*state space*). Dalam masalah kendali PID, akan lebih menguntungkan jika menggunakan pendekatan fungsi alih.

3.3.1 Fungsi Alih dan Blok Diagram Generator DC

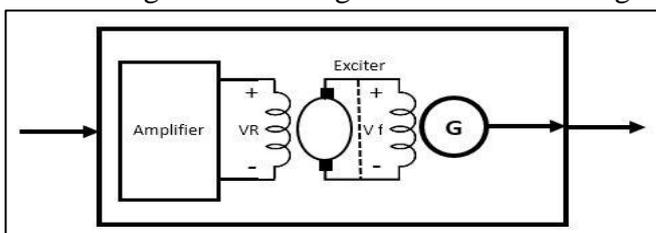
Sistem pasokan listrik arus searah sebagai penguatan pada generator listrik atau sebagai pembangkit medan magnet, sehingga suatu generator dapat menghasilkan energi listrik dengan besar tegangan keluaran generator bergantung pada besarnya arus eksitasinya. Hal ini dikenal sebagai sistem eksitasi (Graham, R: 1999). Diagram blok sistem generator dapat dilihat pada Gambar 3.3



Gambar 3 . 3 Blok Sistem Generator

Sumber Gunadin, 2016

Sedangkan untuk diagram blok Subsistem generator dapat dilihat pada Gambar 3.4.



Gambar 3 . 4 Blok Subsistem Generator

Sumber Gunadin, 2016

Kekokohan tanggapan tegangan sistem eksitasi generator merupakan kemampuan dari sistem eksitasi generator untuk meredam derau pada frekuensi tinggi, mempunyai tanggapan yang cepat terhadap masukan tertentu dan mampu menghilangkan gangguan pada saat beroperasi. (Skogestad, S. dan Postlethwaite: 1996).

Analisa kekokohan ini akan diamati tingkat kekokohan tanggapan tegangan sistem eksitasi generator terhadap perubahan parameter dari komponen – komponen sistem eksitasi. Adapun perubahan parameter yang diamati adalah perubahan konstanta penguatan generator

terhadap tanggapan tegangan sistem eksitasi. Untuk perubahan parameter komponen sistem eksitasi yang lain tidak dilakukan.

Pemodelan matematis sistem eksitasi generator yang dibahas meliputi pemodelan amplifier, pemodelan eksiter dan pemodelan generator. Tipe sistem eksitasi generator yang digunakan adalah sistem eksitasi generator tipe arus searah dengan umpan balik satu dengan bentuk diagram blok. Selain itu pemodelan matematis sistem eksitasi generator ini dilakukan dengan menggunakan persamaan linear diferensial dan transformasi Laplace. Hasil pemodelan masing – masing komponen ini berupa fungsi alih orde satu Fungsi alih dari masing-masing komponen ini kemudian digabungkan dan diperoleh fungsi alih lingkaran terbuka dan fungsi alih lingkaran tertutup dari sistem eksitasi generator.

Bagian ini membahas tentang pemodelan matematis komponen – komponen sistem eksitasi generator yang meliputi model amplifier, model eksiter dan model generator. Untuk model amplifier dinyatakan dalam bentuk persamaan(3-1).

$$\frac{V_R(s)}{V_E(s)} = \frac{K_A}{1+\tau_A s} \dots\dots\dots (3-1)$$

Nilai konstanta penguatan dari amplifier memiliki rentang nilai dari 10.0000 sampai 400.000 sedangkan nilai konstanta waktu amplifier memiliki rentang nilai dari 0.0200 detik sampai 0.1000 detik. Untuk model eksiter dinyatakan dalam bentuk persamaan(3-2).

$$\frac{V_F(s)}{V_R(s)} = \frac{K_E}{1+\tau_E s} \dots\dots\dots (3-2)$$

Untuk konstanta penguatan eksiter dan konstanta waktu eksiter ini mempunyai nilai yang kecil. Untuk model generator dinyatakan dalam bentuk persamaan(3-3).

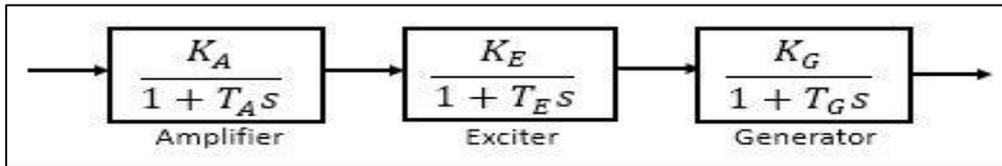
$$\frac{V_T(s)}{V_F(s)} = \frac{K_G}{1+\tau_G s} \dots\dots\dots (3-3)$$

dimana nilai konstanta penguatan generator memiliki rentang nilai dari 0.7000 sampai 1.0000 sedangkan untuk konstanta waktu generator memiliki rentang nilai dari 1.0000 detik sampai 2.0000 detik pada keadaan beban nol sampai keadaan beban penuh. Komponen – komponen sistem eksitasi generator ini kemudian digabungkan dan terbentuk diagram blok dari sistem eksitasi generator yang diperlihatkan pada Gambar 3.5. Berdasarkan diagram blok pada Gambar 3.5. Kemudian diperoleh fungsi alih lingkaran terbuka dan fungsi alih lingkaran tertutup dari sistem eksitasi generator. Untuk fungsi alih lingkaran terbuka diperlihatkan pada persamaan(3-4).

Untuk fungsi alih terbuka dinyatakan dengan persamaan (3-4).

$$\frac{V_T(s)}{V_{ref}(s)} = \frac{K_A K_E K_G}{(1+\tau_A s)(1+\tau_E s)(1+\tau_G s)} \dots\dots\dots(3-4)$$

$$\frac{V_T(s)}{V_{ref}(s)} = \frac{16,0000}{0,0180s^3+0,4020s^2+1,7600s+1,0000} \dots\dots\dots(3-5)$$



Gambar 3 . 5 Fungsi Alih Subsistem Blok Generator DC

3.3.2 Fungsi Alih dan Blok Diagram Sensor Tegangan

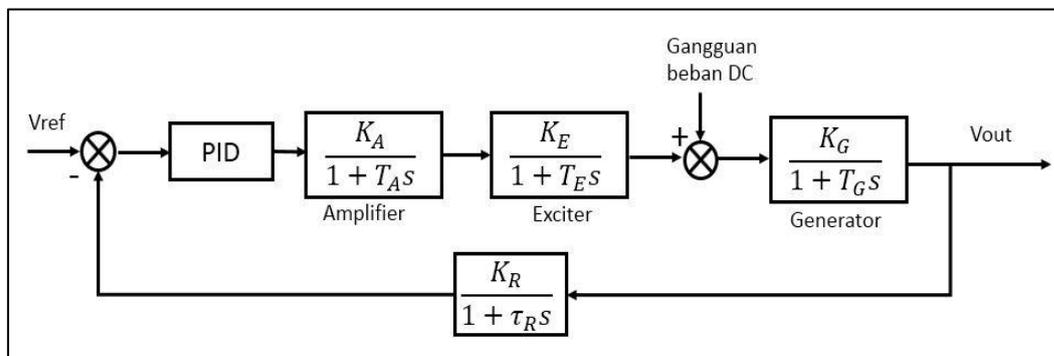
Sensor terdiri atas transformator tegangan (PT) dan sebuah penyearah. Sensor dapat direpresentasikan sebagai KR dengan konstanta waktu TR dan fungsi transfernya sebagai berikut:

$$\frac{V_S(s)}{V_t(s)} = \frac{K_R}{1+\tau_R s} \dots\dots\dots(3-6)$$

Hasil penyearahan lalu dibandingkan dengan tegangan referensi (Vref) apabila terjadi perbedaan maka kontroler akan memerintahkan amplifier untuk menaikkan atau menurunkan arus penguatan generator DC sehingga tegangan keluar dari generator tersebut juga berubah. Jika tegangan keluar generator DC berubah maka arus penguatan generator sinkron juga berubah, akibatnya tegangan keluaran generator kembali stabil.

3.4 Perancangan Diagram Sistem

Perancangan diagram sistem menjelaskan sistem yang direncanakan secara garis besar dan diharapkan sistem dapat menunjukkan hasil yang sesuai karakteristik komponen asli dan sesuai dengan aturan yang digunakan. Perancangan diagram proses sistem dan diagram blok sistem dapat dilihat pada gambar 3.6.



Gambar 3 . 6 Blok Diagram Sistem Kendali

3.5 Perancangan Algoritma

Terdapat tiga tahapan pada proses perancangan algoritma, yaitu:

1. Desain Menggunakan Matlab
2. Penalaan Kontroler menggunakan penala PIDF - pidTuner
3. Penentuan model referensi
4. Penetapan parameter kontroler

Adapun penjelasan ketiga tahapan tersebut adalah sebagai berikut:

3.5.1 Desain Menggunakan Matlab

A. Pendahuluan

Fungsi alih untai tertutup sebuah sistem kontrol dengan umpan balik negatif secara matematik dapat dituliskan sebagai:

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s) \cdot H(s)}$$

$P(s)$ adalah fungsi alih dari plant, $G(s)$ adalah fungsi alih untai maju (*feed forward*) pada umumnya adalah perkalian dari fungsi alih kontroler, aktuator dan plant. Sedangkan $H(s)$ adalah fungsi alih dari umpan balik yaitu sensor. Agar sistem kontrol mempunyai respon keluaran yang stabil, secara teori mempersyaratkan $1 + G(s) \cdot H(s) = 0$. Semakin rumit fungsi alih plant maka semakin rumit pula proses desainnya, namun dengan tersedianya perangkat lunak analisis sistem kontrol yaitu matlab – proses desain sistem kontrol akan menjadi lebih cepat, mudah dan akurat. Pada sesi ini hanya menjelaskan tentang proses desain kontrol analog dengan menggunakan fungsi-fungsi yang tersedia di Matlab dengan menggunakan fungsi alih plant yang digunakan dalam pembahasan skripsi ini sehingga dapat melakukan analisa secara cepat.

B. Desain Sistem Kontrol Secara Manual

Mendesain sistem kontrol secara manual menggunakan Matlab adalah cara mendesain dengan menggunakan fungsi-fungsi secara terpisah seperti yang di jelaskan pada sub bab sebelumnya. Langkah-langkah mengikuti metode desain yang lazim dalam sistem kontrol, fungsi-fungsi yang ada di Matlab untuk membantu menganalisis secara cepat. Fungsi-fungsi dieksekusi dengan urutan tergantung kebutuhan dan situasi dalam langkah-

langkah proses desain sistem kontrolnya. Langkah-langkah umumnya mendesain sistem kontrol adalah:

1. Mendefinisikan fungsi alih plant

Fungsi alih plant yang digunakan sebagai berikut:

$$G(s) = \frac{16,0000}{0,0180s^3 + 0,4020s^2 + 1,7600s + 1,0000}$$

Jika konstanta polinomial fungsi alih telah diketahui pada penyebut dan pembilang, maka pendefinisian secara langsung dapat memasukkan konstanta tersebut kedalam sebuah variabel dengan cara:

```
G = tf(16.0000,[0.0180 0.4020 1.7600 1.0000])
```

```
G =
```

```
16
```

```
-----
```

```
0.018 s^3 + 0.402 s^2 + 1.76 s + 1
```

```
Continuous-time transfer function.
```

Melalui cara ini pendefinisian variabel polinomial s tidak perlu dilakukan.

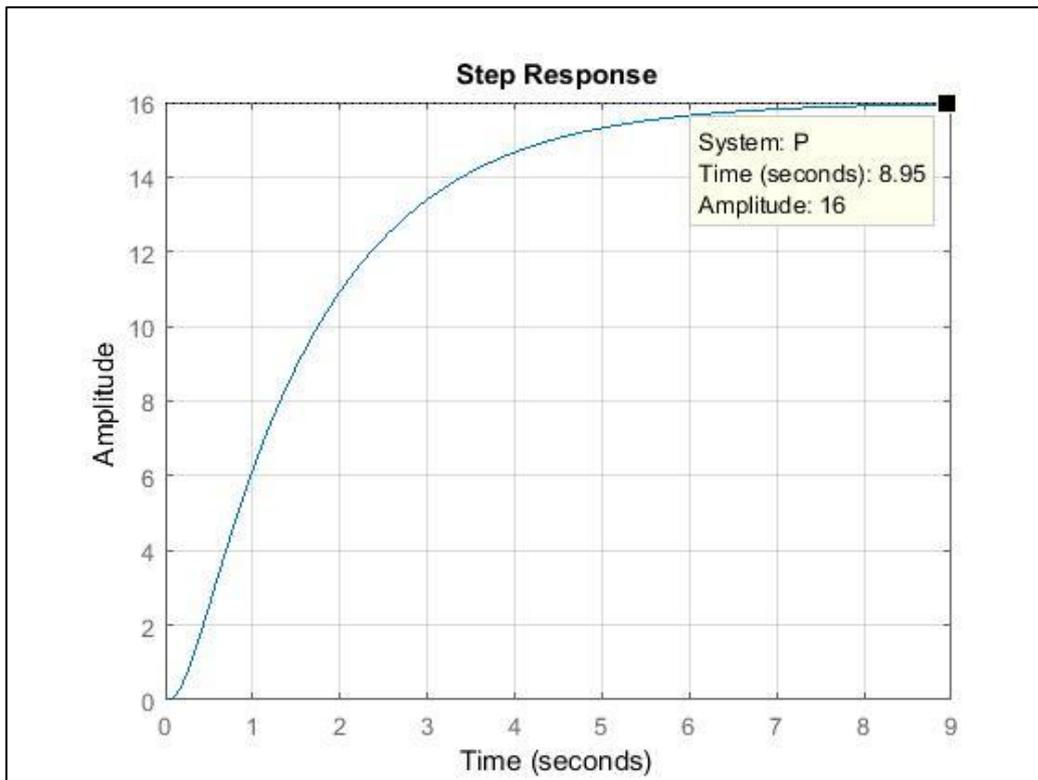
2. Melihat Karakteristik plant

Sebelum mendesain sistem kontrol, pengujian karakteristik plant sangat diperlukan untuk memutuskan jenis kontroler yang diperlukan. Untuk melihat karakteristik fungsi alih plant diantaranya melakukan pengujian respon keluarannya menggunakan fungsi step melalui *stepplot*.

```
stepplot(G)
```

```
grid on
```

pada baris 1 menguji keluaran menggunakan sinyal uji step, baris 2 memasang grid pada grafik. Hasil pengujian karakteristik fungsi alih plant diperlihatkan pada Gambar 3.7.



Gambar 3. 7 Respon Fungsi Alih Plant

Gambar diatas karakteristik dari fungsi alih: $\frac{16,0000}{0,0180s^3+0,4020s^2+1,7600s+1,0000}$. respon stabil pada amplitudo 16 dengan waktu 8,95

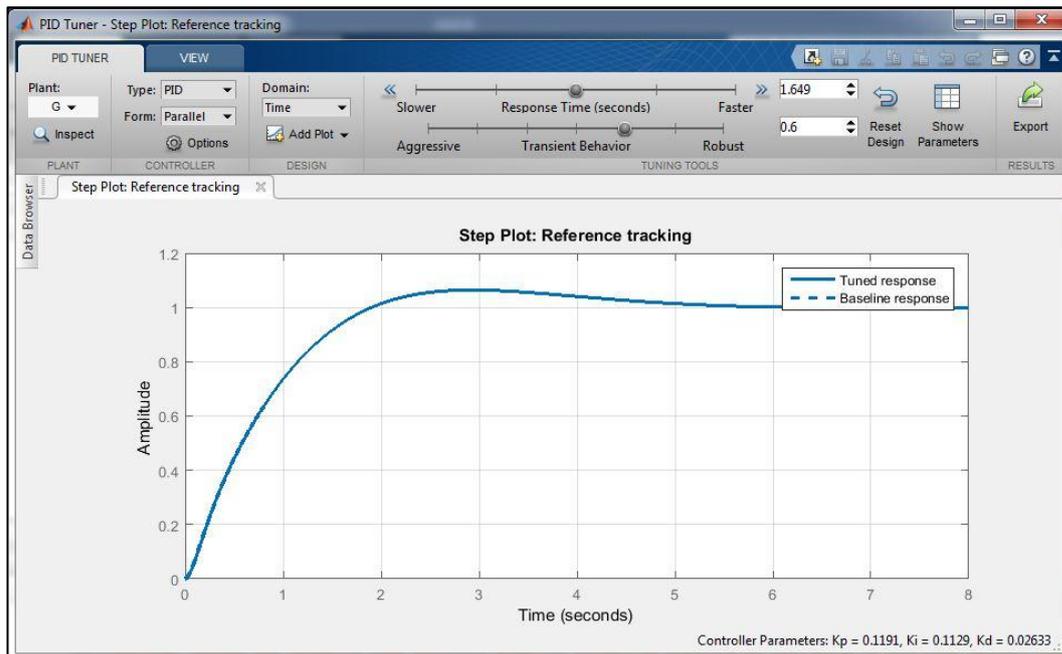
3.5.2 Penalaan Kontroler menggunakan penala PIDF - pidTuner

Penalaan kontroler menggunakan penala PIDF – pidTuner adalah metode penalaan untuk menentukan jenis kontroler dan konstanta penguatan dari fungsi alih $C(s)$ dapat di peroleh secara cepat, mudah dan akurat dengan menggunakan aplikasi-aplikasi yang tersedia di matlab.

Plant diketahui memiliki fungsi alih

$$\frac{V_T(s)}{V_{ref}(s)} = \frac{16,0000}{0,0180s^3+0,4020s^2+1,7600s+1,0000} \dots\dots\dots(3-6)$$

Jika didesain menggunakan kontroler PIDF maka penalaannya akan di tampilkan pada jendela tampilan menu pidTuner pada Gambar 3.8.

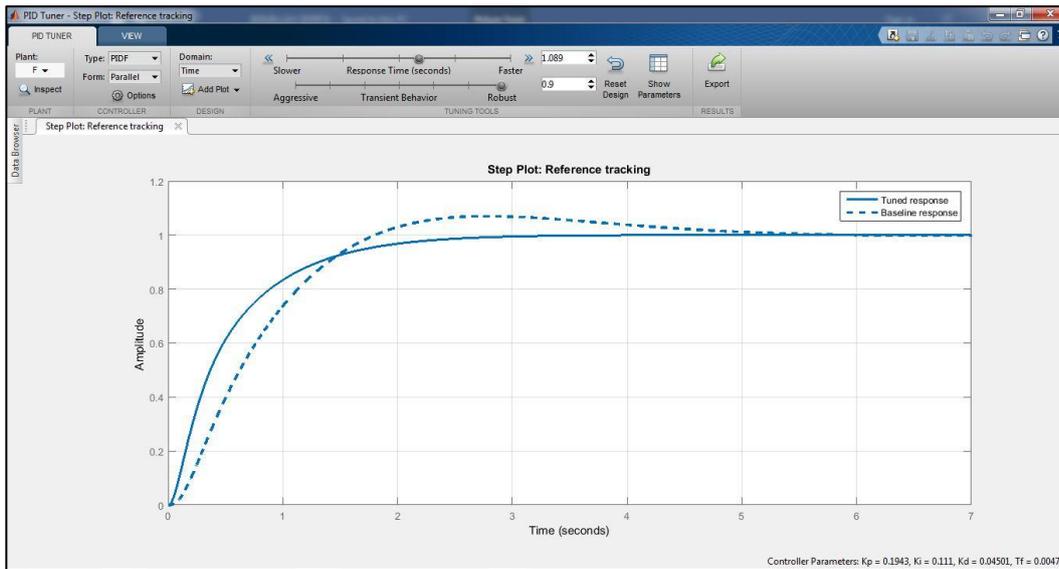


Gambar 3 . 8 Jendela Tampilan Menu pidTuner Sebelum Ditala

Untuk melihat parameter acuan dan hasil penalaan dilakukan dengan cara klik menu tombol *show parameter*, hasilnya diperlihatkan pada Gambar 3.10. Menu ini digunakan untuk melihat parameter secara realtime, karena setiap perubahan penalaan data parameter akan berubah secara otomatis.

Dalam penelitian ini desain dibutuhkan agar respon dipercepat sesuai spesifikasi desain yang telah ditetapkan pada sub bab yang sebelumnya. Dengan cara menggeser menu geser *Response Time* untuk menala waktu respon sistem kontrol dan menu geser *Transient Behavior* untuk menala perilaku transiennya.

Berikut tampilan jendela tampilan penalaan pidTuner yang telah dimodifikasi sesuai spesifikasi desain yang ditetapkan. Dapat dilihat pada Gambar 3.9



Gambar 3. 9 Jendela Tampilan Menu pidTuner Setelah Ditala

Selanjutnya dengan melihat di menu *show parameter* pada Gambar 3.10 kita akan melihat apakah penalaan sistem kita sudah sesuai dengan spesifikasi desain yang telah ditentukan sebelumnya. Ketika sudah memenuhi atau sesuai maka penelitian akan lanjut ke tahap berikutnya.

| Controller Parameters | | |
|-----------------------|-----------|----------|
| | Tuned | Baseline |
| Kp | 0.19425 | 0.11169 |
| Ki | 0.11098 | 0.10941 |
| Kd | 0.045014 | 0.016024 |
| Tf | 0.0047665 | 0.078599 |

| Performance and Robustness | | |
|----------------------------|----------------------|---------------------|
| | Tuned | Baseline |
| Rise time | 1.22 seconds | 1.2 seconds |
| Settling time | 2.27 seconds | 4.62 seconds |
| Overshoot | 0 % | 7.02 % |
| Peak | 1 | 1.07 |
| Gain margin | 40.1 dB @ 61.8 rad/s | 28 dB @ 15.3 rad/s |
| Phase margin | 90 deg @ 1.84 rad/s | 69 deg @ 1.21 rad/s |
| Closed-loop stability | Stable | Stable |

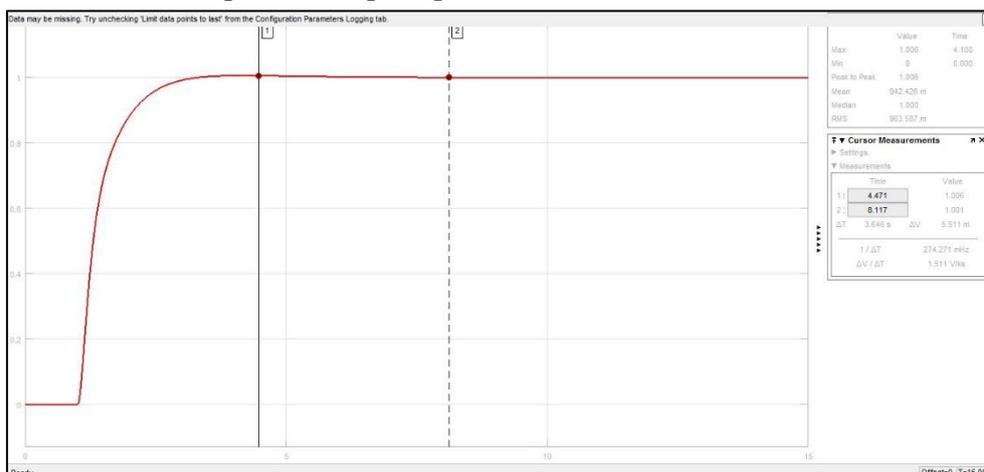
Gambar 3. 10 Jendela Tampilan Show Parameter Setelah Ditala

3.5.3 Penentuan Model Referensi

Penentuan model referensi dilakukan dengan cara *try and error* sehingga didapatkan respon yang stabil yang mengikuti karakteristik dari *plant*. Model referensi yang akan ditentukan tidak memiliki *error steady state*, tidak terdapat osilasi dan memiliki *settling time* yang cepat. Penentuan model referensi dapat ditentukan dengan cara melihat hasil respon saat simulasi melalui aplikasi matlab dan pada saat implementasi sistem. Fungsi alih dari model referensi adalah seperti pada persamaan (3-6)

$$\frac{V_T(s)}{V_{ref}(s)} = \frac{16,0000}{0,0180s^3 + 0,4020s^2 + 1,7600s + 1,0000} \dots\dots\dots(3-6)$$

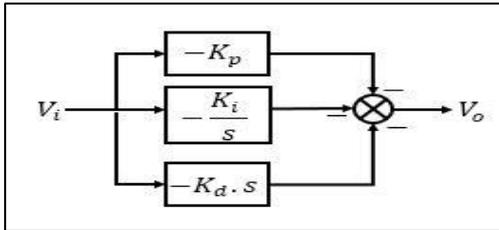
Dengan memberikan masukan unit step dari program matlab, maka dapat diketahui bahwa respon model referensi memiliki nilai time constant sebesar 1,511 detik dan nilai settling time sebesar 3.022 detik. Respon transien dari model referensi yang telah diberi masukan unit step adalah seperti pada Gambar 3.11.



Gambar 3 . 11 Model Referensi

Kontroler PID Paralel

Disebut sebagai kontroler PID paralel, karena masing-masing kontroler P,I dan D dihubungkan secara paralel dimana semua sinyal keluarannya dijumlah dengan menggunakan penjumlah membalikkan, sedangkan masukannya dihubungkan secara paralel. Secara blok diagram kontroler PID diperlihatkan pada Gambar 3.12.



Gambar 3. 12 Gambar 3. 12 Blok Diagram Kontroler PID Paralel

Sumber : Siswojo, 2017 : 129

Dari Gambar 3.12, fungsi alih jika disederhanakan ditambah dengan fungsi alih filter akan menjadi:

$$\frac{v_o(s)}{v_i(s)} = - \left(-K_p - \frac{K_i}{s} - K_d \cdot \frac{s}{T_f * s + 1} \right) \dots \dots \dots (3-7)$$

Eksperimen kontroler PID paralel

Tentang kontroler PID paralel telah di jelaskan sebelumnya. Eksperimen desain kontroler PID menggunakan fungsi alih sistem untai tertutup dalam persamaan yaitu

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s) \cdot H(s)} \dots \dots \dots (3-8)$$

Dengan:

$$G(s) = \frac{16,0000}{0,0180s^3 + 0,4020s^2 + 1,7600s + 1.0000}$$

$$H(s) = \frac{1}{0,025 + 1}$$

$$\frac{C(s)}{R(s)} = \frac{0,0072 s^4 + 0,7368 s^3 + 13,57 s^2 + 56,72 s + 32}{8,1e - 06 s^7 + 0,0006858 s^6 + 0,0201 s^5 + 0,2612 s^4 + 1,549 s^3 + 3,99 s^2 + 3,545 s + 1}$$

1. Memasukkan fungsi alih sistem

fungsi alih plant sebelum didesain kontrolernya harus didefinisikan sebagai fungsi s dengan memasukkan sebuah variabel G, dan fungsi alih umpan balik juga didefinisikan dengan variabel H dengan cara berikut

$$G = \text{tf}(16.0000,[0.0180 0.4020 1.7600 1.0000]);$$

G =

16

$$0.018 s^3 + 0.402 s^2 + 1.76 s + 1$$

Continuous-time transfer function.

Dan

$$H = \text{tf}(1,[0.025 \ 1]);$$

H =

1

$$0.025 s + 1$$

Continuous-time transfer function.

Lalu fungsi alih sistem $\frac{C(s)}{R(s)}$ sebelum didesain kontrolernya harus didefinisikan sebagai fungsi s dengan memasukkan kesebuah variabel F, dengan cara berikut

$$F = \text{tf}(G/1+H*G);$$

F =

$$0.0072 s^4 + 0.7368 s^3 + 13.57 s^2 + 56.72 s + 32$$

$$8.1e-06 s^7 + 0.0006858 s^6 + 0.0201 s^5 + 0.2612 s^4 + 1.549 s^3 + 3.99 s^2 + 3.545 s + 1$$

Continuous-time transfer function.

Nama variabel untuk menyimpan fungsi alih sistem adalah F jika lebih suka menggunakan persamaan matematik polinomial s, harus mendefinisikan dengan memasukkan perintahs = *tf('s')* terlebih dahulu.

2. Inisialisasi kontroler PIDF

Awalnya kontroler PI belum didefinisikan, untuk memprediksi kontroler PIDF yang cocok dengan lant tersebut, harus didefinisikan terlebih dahulu dengan memberikan perintah *pidtune*. Cara mengawali prediksi kontroler PIDF:

```
C=pidtune(F,'pidf')
```

C =

$$K_p + K_i * \frac{1}{s} + K_d * \frac{s}{T_f * s + 1}$$

with $K_p = 0.112$, $K_i = 0.109$, $K_d = 0.016$, $T_f = 0.0786$

Continuous-time PIDF controller in parallel form.

Parameter awal adalah fungsi alih sistem dalam hal ini adalah variabel F sebagai fungsi alih sistem yang telah didefinisikan pada langkah 1, parameter kedua adalah pemilihan jenis kontroler PIDF. Sedangkan hasil prediksi kontroler PIDF diarahkan sebagai variabel C. sebagai awalan kontroler PIDF ditala menjadi $K_p = 0.0566$, $K_i = 0.0523$, $K_d = 0.0129$, dan $T_f = 0.0786$. selanjutnya konfigurasi kontroler PID ini akan digunakan sebagai referensi untuk melakukan proses penalaan guna memperbaiki respon sesuai kebutuhan.

3. Mengeksekusi pidTuner untuk menala PIDF

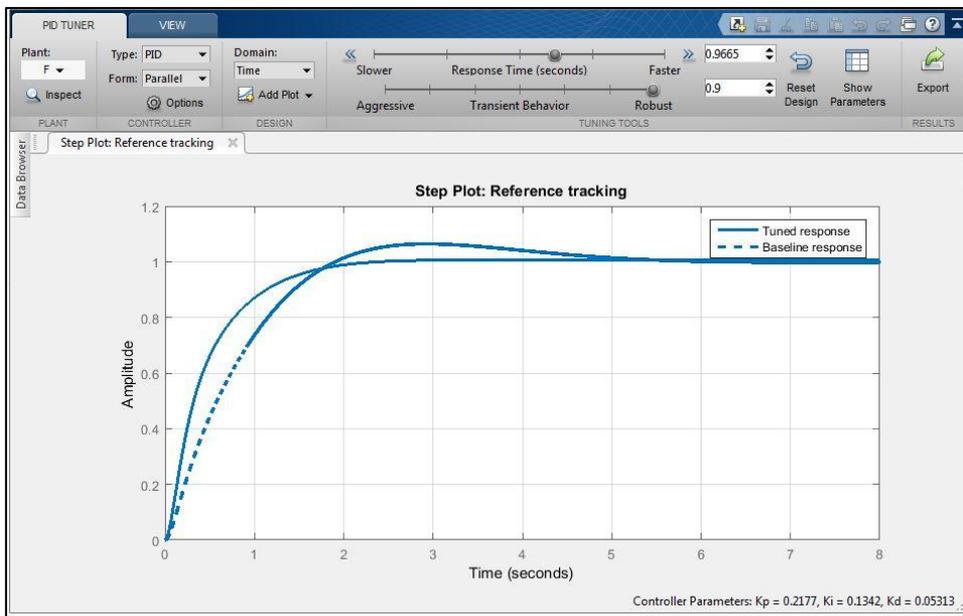
Kontroler PID ini dapat ditala ulang dengan menggunakan perintah perintah *pidTuner*, dengan memasukkan perintah sebagai berikut:

```
1 C=pidtune(P, 'pid')
```

Setelah perintah *pidTuner* diberikan, akan muncul tampilan menu grafis aplikasi untuk melakukan penalaan kontroler PIDF seperti pada Gambar 3.13 .respon diperlihatkan berdasarkan prediksi awal kontrolernya, untuk selanjutnya rson tersebut dapat diperbaiki dengan sangat mudah dan cepat dengan menggunakan menu slider. Menu slider yang disediakan berfungsi untuk melakukan penalaan perbaikan respon kontroler PID. Dua setelan respon yang disediakan yaitu:

- (a) **Setelan perilaku waktu** (*response time*): digunakan untuk memperbaiki respon waktu agar menjadi lebih lambat (*slower*) atau lebih cepat (*faster*) dengan menggeser slider atau dengan cara memasukkan nilainya melalui isian.
- (b) **Setelan perilaku transien** (*response behavior*): digunakan untuk memperbaiki perilaku transiennya menjadi lebih responsif (*aggressive*) atau lebih tegar (*robust*) dengan menggeser-geser slider atau dengan cara memasukkan nilainya melalui isian

sebagai contoh pernalaannya dengan menentukan nilai perilaku transien dibuat 0,9 dan nilai respon waktu 0,9665. setelah dilakukan penalaan responnya berubah, respon awal diperlihatkan sebagai grafik yang terputus dapat dilihat pada Gambar 3.13.



Gambar 3 . 13 Jendela Penalaan

4. Melihat parameter penalaan

Untuk melihat parameter kontroler respon referensi dan respon setelah dilakukan penalaan klik tombol menu *Show Parameters*, hasilnya diperlihatkan dalam gambar (). Ketika dilakukan penalaan grafik respon, parameter-parameter ini juga berubah secara realtime untuk dapat memastikan parameter-parameter kontrolnya. Parameter-parameter kontroler yang ditampilkan:

- (a) $K_p = 0,11169$ menjadi $K_p = 0,19425$
- (b) $K_i = 0,10941$ menjadi $K_i = 0,11098$
- (c) $K_d = 0,016024$ menjadi $K_d = 0,045014$
- (d) $T_f = 0,078599$ menjadi $T_f = 0,0047665$

Sedangkan parameter untuk kerja dan ketegaran yang di tampilkan:

- (a) *Rise time*=1,2 detik menjadi *Rise time*=1,22 detik
- (b) *Settling time*= 4,62 detik menjadi *settling time*= 2,27 detik
- (c) *Overshoot* = 7,02% menjadi *Overshoot* = 0%
- (d) *Peak* = 1,07 menjadi *Peak* menjadi *Peak* = 1
- (e) *Gain margin*=28 dB @ 15,3 rad/s menjadi *Gain margin* = 40,1 dB @ 61,8 rad/s
- (f) *Phase margin* =69 deg @1,21 Rad/s menjadi *Phase margin* = 90 deg @1,84rad/s

(g) *Closed – loop stability = Stable* menjadi *Closed – loop stability = Stable*

5. Mengekpor parameter kontroler

Parameter hasil penalaan kontroler dapat diekspor ke konsol untuk keperluan pemrosesan lebih lanjut jika diperlukan. Untuk mengekspor parameter kontroler yaitu variabel C cukup klik tombol menu *Export*, selanjutnya pilih variabel yang akan diekspor. Ketika kembali ke konsol fungsi alih kontroler dapat ditampilkan sebagai berikut. Dan dapat dilihat pada Gambar 3.14.

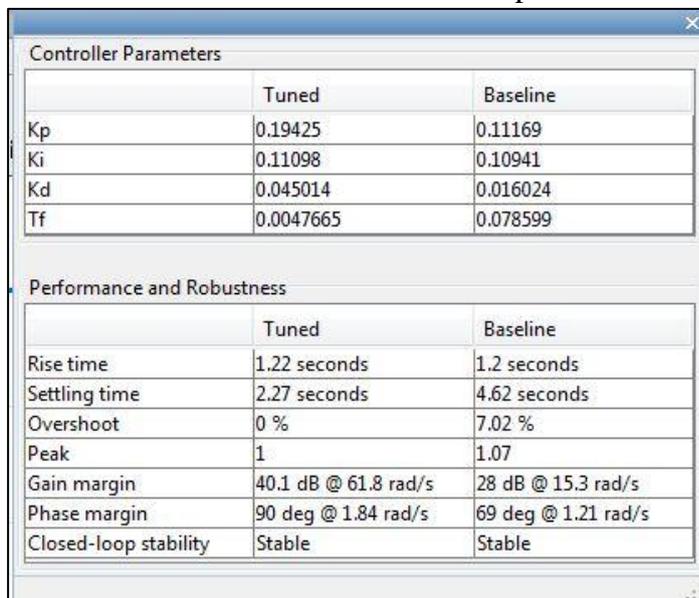
>> C

C =

$$K_p + K_i * \frac{1}{s} + K_d * \frac{s}{T_f * s + 1}$$

with $K_p = 0.112$, $K_i = 0.109$, $K_d = 0.016$, $T_f = 0.0786$

Continuous-time PIDF controller in parallel form.



| Controller Parameters | | |
|-----------------------|-----------|----------|
| | Tuned | Baseline |
| Kp | 0.19425 | 0.11169 |
| Ki | 0.11098 | 0.10941 |
| Kd | 0.045014 | 0.016024 |
| Tf | 0.0047665 | 0.078599 |

| Performance and Robustness | | |
|----------------------------|----------------------|---------------------|
| | Tuned | Baseline |
| Rise time | 1.22 seconds | 1.2 seconds |
| Settling time | 2.27 seconds | 4.62 seconds |
| Overshoot | 0 % | 7.02 % |
| Peak | 1 | 1.07 |
| Gain margin | 40.1 dB @ 61.8 rad/s | 28 dB @ 15.3 rad/s |
| Phase margin | 90 deg @ 1.84 rad/s | 69 deg @ 1.21 rad/s |
| Closed-loop stability | Stable | Stable |

Gambar 3. 14 Jendela Show Parameter

Dengan demikian dari hasil proses penalaan, fungsi alih kontroler PID menjadi:

$$C(s) = 0,11169 + 0,10941 \cdot \frac{1}{s} + 0,016024 \cdot \frac{s}{0,078599s + 1}$$

Menjadi

$$C(s) = 0,19425 + 0,11098 \cdot \frac{1}{s} + 0,045014 \cdot \frac{s}{0,0047665s + 1}$$

3.5.4 Penetapan Parameter Kontroler

Penetapan parameter kontroler pada penelitian ini dapat dilihat pada hasil penalaan menggunakan pidTuner yang dapat dilihat pada menu *Show Parameter* pada gambar 3.13 Dan didapatkan nilai parameter kontrolernya sebagai berikut:

$$K_p = 0,19425$$

$$K_i = 0,11098$$

$$K_d = 0,045014$$

$$T_f = 0,0047665$$

Kemudian nilai parameter yang didapat dimasukkan ke dalam parameter blok kontroler PID yang terdapat pada blok diagram sistem yang akan diteliti.

BAB IV

PENGUJIAN DAN SIMULASI SISTEM

Pada bab ini akan dibahas mengenai identifikasi karakteristik, pengujian dan simulasi dari tiap blok sistem secara keseluruhan. Identifikasi karakteristik dilakukan untuk mengamati bagaimana karakteristik dari tiap blok sistem secara keseluruhan. Simulasi ini dilakukan untuk mengamati respon sistem yang terjadi. Identifikasi dan Simulasi dilakukan dengan program Matlab R2015a

4.1 Identifikasi Karakteristik

Identifikasi Karakteristik setiap blok dilakukan untuk mempermudah analisis sistem. Identifikasi Karakteristik dibagi menjadi beberapa bagian, yaitu:

- a. Identifikasi Karakteristik *plant*
- b. Identifikasi Karakteristik sistem keseluruhan

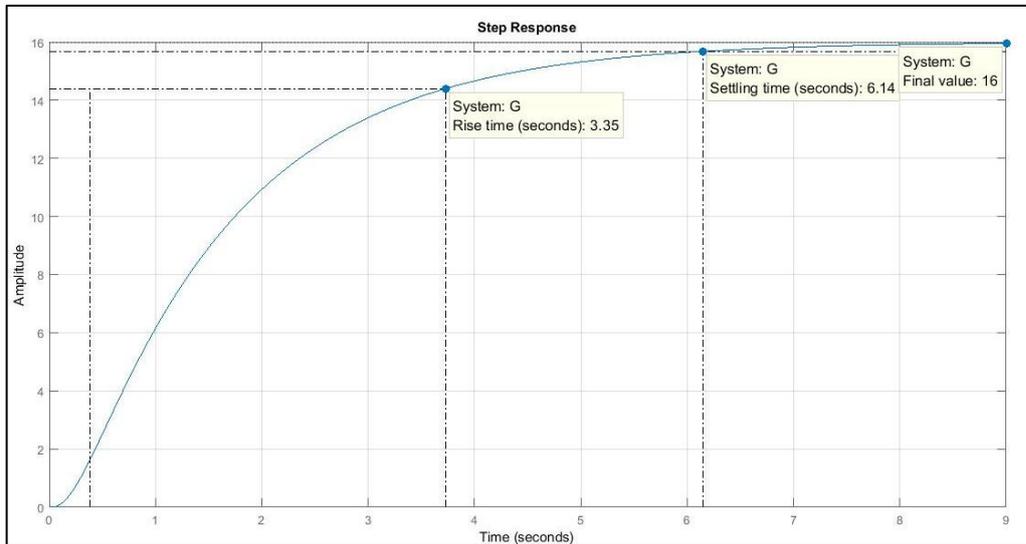
Salah satu teknik untuk mengetahui karakteristik sistem adalah dengan memberinya sinyal uji sebagai input dan mengamati output-nya atau respon alih-nya (transient). Salah satu sinyal uji adalah sinyal unit step $u(t)$ dan respon alih-nya dikenal dengan respon step (step response). Sinyal unit step $u(t)$ memiliki nilai satu untuk $t \geq 0$ dan nol untuk $t < 0$.

a) Identifikasi karakteristik *plant*

Dari hasil identifikasi, didapatkan fungsi alih *plant* seperti pada persamaan(4-1)

$$\frac{V_T(s)}{V_{ref}(s)} = \frac{16,0000}{0,0180s^3 + 0,4020s^2 + 1,7600s + 1,0000} \dots\dots\dots (4-1)$$

Dari fungsi alih pada yang telah diperoleh, dapat dilihat melalui fungsi stepinfo(Tf)



Gambar 4 . 1 Kurva Karakteristik Plant

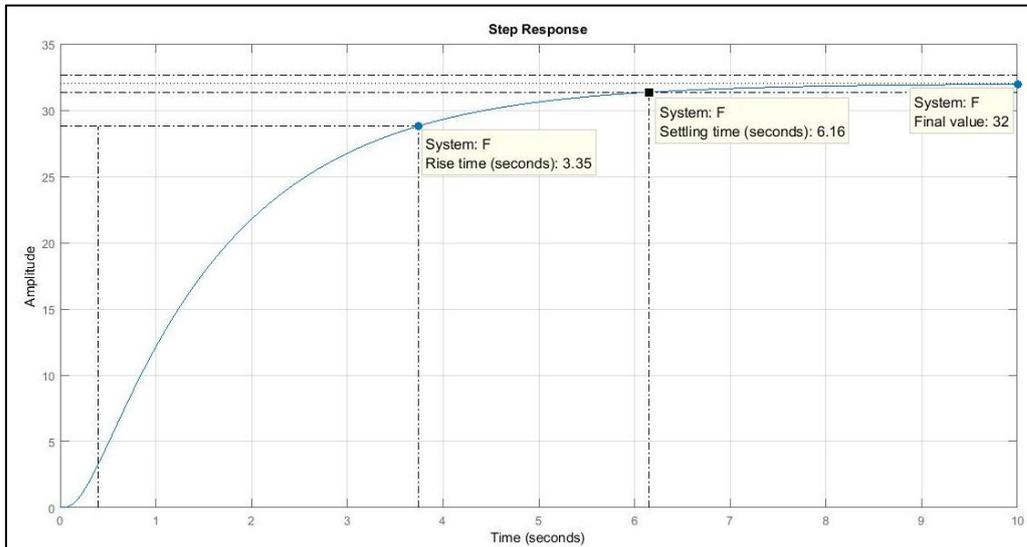
bahwa *plant* memiliki:

- RiseTime: 3,3496
- SettlingTime: 6,1439
- SettlingMin: 14,4102
- SettlingMax: 15,9773
- Overshoot: 0
- Undershoot: 0
- Peak: 15,9773
- PeakTime: 10,1130

b) Identifikasi karakteristik sistem keseluruhan

Dari hasil identifikasi, didapatkan fungsi sistem keseluruhan seperti pada persamaan (4-2).

$$\frac{V_S(s)}{V_t(s)} = \frac{0,0072 s^4 + 0,7368 s^3 + 13,57 s^2 + 56,72 s + 32}{8,1e-06 s^7 + 0,0006858 s^6 + 0,0201 s^5 + 0,2612 s^4 + 1,549 s^3 + 3,99 s^2 + 3,545 s + 1} \quad (4-2)$$



Gambar 4 . 2 Karakteristik Sistem Keseluruhan

Dari fungsi alih pada yang telah diperoleh, dapat dilihat melalui fungsi stepinfo(Tf) bahwa sistem keseluruhan memiliki:

- RiseTime: 3,3499
- SettlingTime: 6,1566
- SettlingMin: 28,8034
- SettlingMax: 31,9814
- Overshoot: 0
- Undershoot: 0
- Peak: 31,9814
- PeakTime: 11,4646

4.2 Simulasi Sistem dengan Berbagai Beban

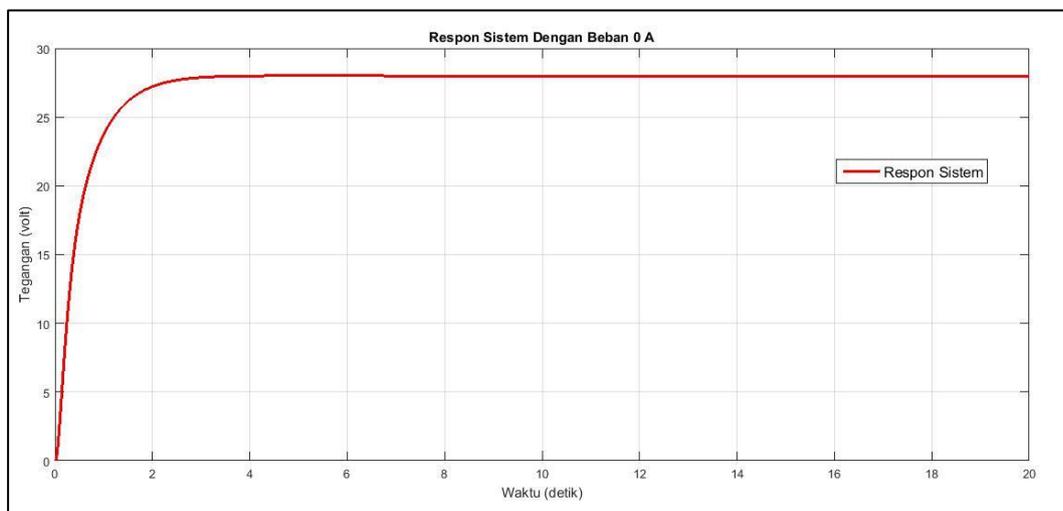
Simulasi dilakukan dengan memberikan masukan berupa tegangan referensi 28 VDC pada blok diagram simulink yang ditunjukkan lampiran. Kemudian diberikan gangguan berupa beban. Adapun simulasi yang dilakukan meliputi berbagai nilai beban sebagai berikut:

1. Nilai beban 0 A
2. Nilai beban 25 A
3. Nilai beban 50 A
4. Nilai beban 100 A
5. Nilai beban 200 A

Simulasi ini dilakukan sesuai dengan spesifikasi desain yang telah ditentukan dengan kontroler di tala menggunakan pidTuner. Simulasi dilakukan dengan nilai $K_p = 0,2177$, $K_i = 0,1342$, dan $K_d = 0,05313$.

4.2.1 Simulasi Sistem Dengan Beban 0 A

Simulasi dilakukan dengan memberikan masukan berupa tegangan referensi 28 VDC pada blok diagram simulink yang ditunjukkan lampiran. Kemudian diberikan gangguan berupa beban 0 A

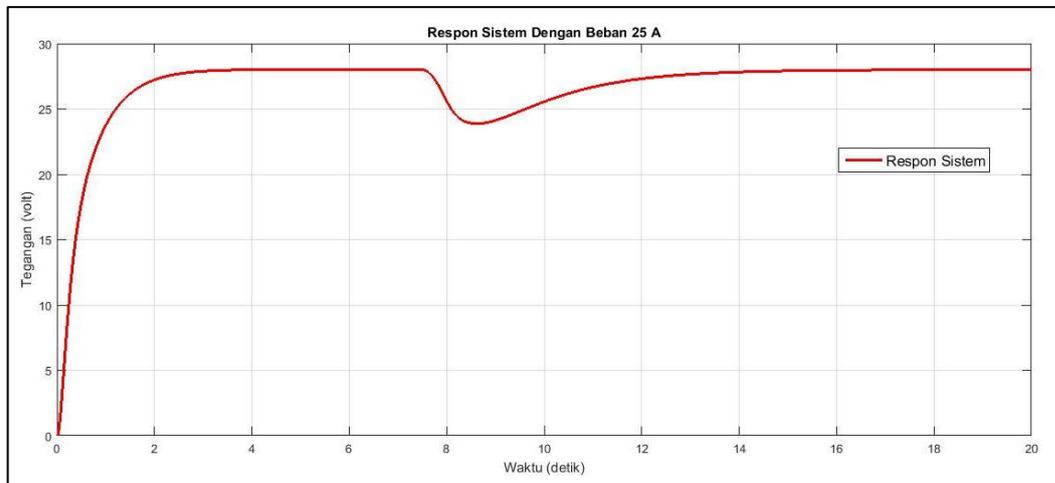


Gambar 4. 3 Respon Sistem Dengan Beban 0 A

Dari Gambar 4.3 dapat dilihat bahwa dengan masukan 28 Volt dengan beban 0 A, respon sistem steady state dengan waktu 4 detik dan tidak ada overshoot atau kesalahan pada kurva.

4.2.2 Simulasi Sistem Dengan Beban 25 A

Simulasi dilakukan dengan memberikan masukan berupa tegangan referensi 28 VDC pada blok diagram simulink yang ditunjukkan lampiran. Kemudian diberikan gangguan berupa beban 25 A

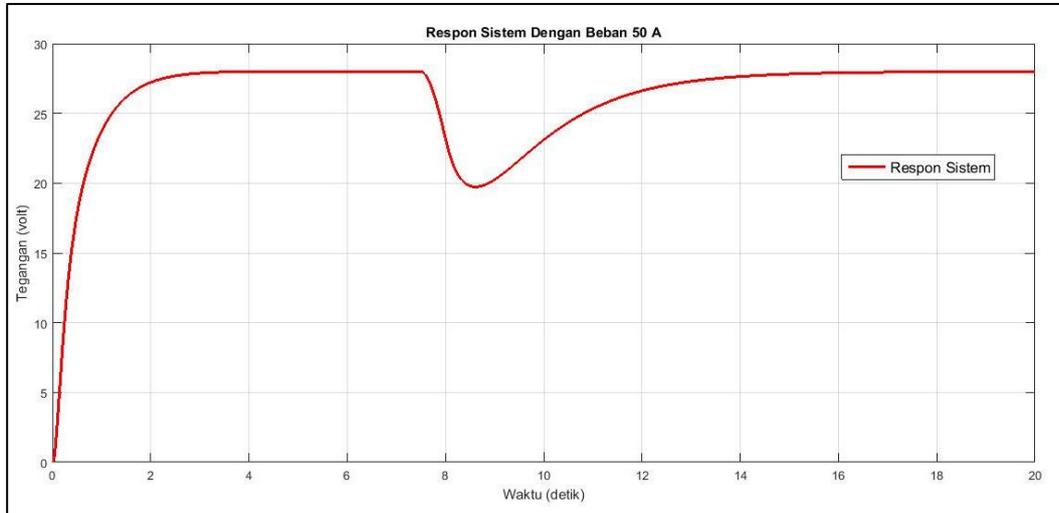


Gambar 4. 4 Respon Sistem Dengan Beban 25 A

Dari Gambar 4.4 dapat dilihat bahwa dengan masukan 28 Volt dengan beban 25 A, respon sistem steady state dengan waktu 3 detik dan tidak ada overshoot atau kesalahan pada keadaan normal yang terdapat pada kurva. Namun ketika pemberian beban sebesar 25 A kurva bergerak turun ke posisi angka 24 Volt selama 1 detik dari posisi awal 28 Volt, lalu dalam waktu 5 detik pengendali dapat memperbaiki kesalahan dan mengembalikan respon pada posisi awal di 28 Volt

4.2.3 Simulasi Sistem Dengan Beban 50 A

Simulasi dilakukan dengan memberikan masukan berupa tegangan referensi 28 VDC pada blok diagram simulink yang ditunjukkan lampiran. Kemudian diberikan gangguan berupa beban 50 A.

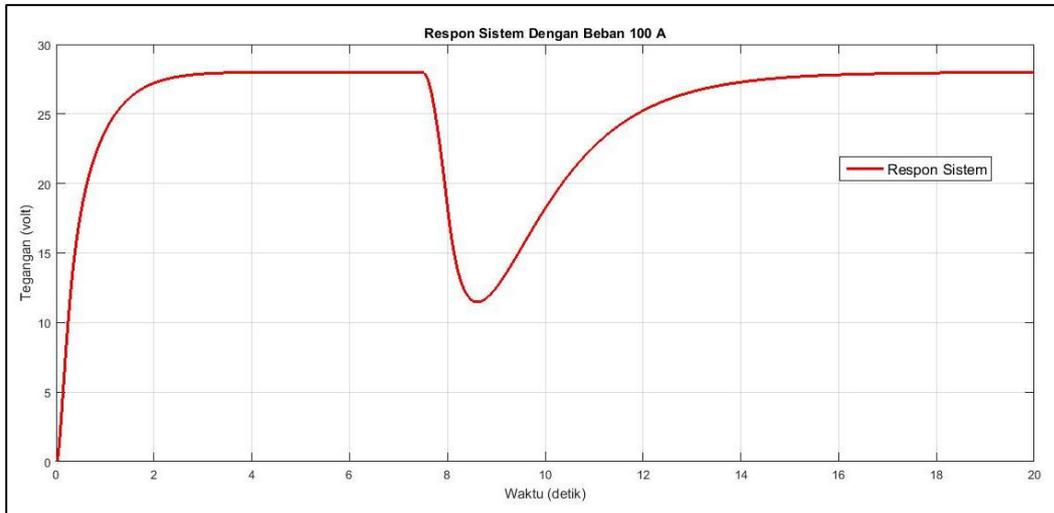


Gambar 4. 5 Respon Sistem Dengan Beban 50 A

Dari Gambar 4.5 dapat dilihat bahwa dengan masukan 28 Volt dengan beban 50 A, respon sistem steady state dengan waktu 4 detik dan tidak ada overshoot atau kesalahan pada keadaan normal yang terdapat pada kurva. Namun ketika pemberian beban sebesar 50 A kurva bergerak turun ke posisi angka 19 Volt selama 1 detik dari posisi awal 28 Volt, lalu dalam waktu 6 detik pengendali dapat memperbaiki kesalahan dan mengembalikan respon pada posisi awal di 28 Volt

4.2.4 Simulasi Sistem Dengan Beban 100 A

Simulasi dilakukan dengan memberikan masukan berupa tegangan referensi 28 VDC pada blok diagram simulink yang ditunjukkan lampiran. Kemudian diberikan gangguan berupa beban 100 A

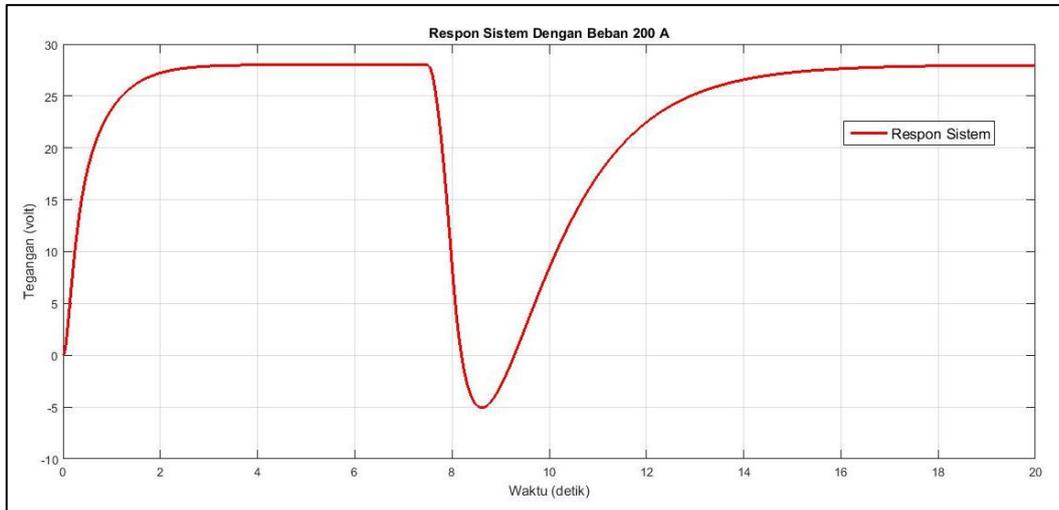


Gambar 4. 6 Respon Sistem Dengan Beban 100 A

Dari Gambar 4.6 dapat dilihat bahwa dengan masukan 28 Volt dengan beban 100 A, respon sistem steady state dengan waktu 4 detik dan tidak ada overshoot atau kesalahan pada keadaan normal yang terdapat pada kurva. Namun ketika pemberian beban sebesar 100 A kurva bergerak turun ke posisi angka 13 Volt selama 1 detik dari posisi awal 28 Volt, lalu dalam waktu 7 detik pengendali dapat memperbaiki kesalahan dan mengembalikan respon pada posisi awal di 28 Volt

4.2.5 Simulasi Sistem Dengan Beban 200 A

Simulasi dilakukan dengan memberikan masukan berupa tegangan referensi 28 VDC pada blok diagram simulink yang ditunjukkan lampiran. Kemudian diberikan gangguan berupa beban 200 A



Gambar 4. 7 Respon Sistem Dengan Beban 200 A

Dari Gambar 4.7 dapat dilihat bahwa dengan masukan 28 Volt dengan beban 200 A, respon sistem steady state dengan waktu 4 detik dan tidak ada overshoot atau kesalahan pada keadaan normal yang terdapat pada kurva. Namun ketika pemberian beban sebesar 200 A kurva bergerak turun ke posisi angka -5 Volt selama 1 detik dari posisi awal 28 Volt, lalu dalam waktu 8 detik pengendali dapat memperbaiki kesalahan dan mengembalikan respon pada posisi awal di 28 Volt.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang diperoleh dari penelitian ini adalah sebagai berikut:

1. Penerapan metode penalaan kontroler menggunakan penala PIDF – pidTuner dapat mempermudah untuk mendapatkan nilai parameter dari pengendali(kontroler) secara cepat, mudah dan akurat. Respon yang dihasilkan sesuai dengan spesifikasi yang diinginkan dan didapatkan dengan cepat, mudah dan akurat.

Dengan metode penalaan kontroler menggunakan penala PIDF – pidTuner, diperoleh parameter-parameter PIDF sebagai berikut:

$$K_p = 0,19425$$

$$K_i = 0,11098$$

$$K_d = 0,045014$$

$$T_f = 0,0047665$$

2. Perancangan sistem kontrol tegangan keluaran pada generator DC pesawat dengan pengendali PIDF menggunakan blok sistem untai tertutup (*close loop*) dengan umpan balik yang diimplementasikan sebagai sensor tegangan dan umpan balik sebagai pembanding.
3. performansi sistem dengan diterapkannya sistem kontrol tegangan keluaran pada generator pesawat N219 dengan pengendali PIDF pada simulasi sangat sesuai seperti yang diharapkan dan telah memenuhi syarat yang ditetapkan pada spesifikasi desain dengan *settling time* 3 detik, tidak ada *error steady state* dan tidak ada *overshoot*.
4. Performansi sistem ketika diterapkannya pengendali PIDF dan pengaruh gangguan pada sistem adalah baik dan sesuai yang diinginkan. Karena dengan diterapkannya pengendali PIDF dapat mempercepat *settling time* menjadi 4 detik dan menghilangkan *error steady state* dan tidak ada *overshoot*. Pada saat sistem diberikan beban terjadi *undervoltage* namun pengendali mampu memperbaiki nilai error yang terjadi sehingga tidak terdapat nilai error lagi pengendali juga mampu memperbaiki nilai error yang terjadi dengan waktu berbeda-beda pada setiap nilai beban yang diberikan dengan hubungan semakin besar nilai beban yang diberikan maka semakin lama waktu perbaikikan oleh pengendali.

5.2 Saran

1. Perlu dipelajari metode penalaan kontroler menggunakan penala PIDF – pidTuner metode penalaan lainnya untuk didapatkan respon yang lebih baik.
2. Diterapkannya sistem Penerapan metode penalaan kontroler menggunakan penala PIDF – pidTuner agar dapat mempermudah untuk mendapatkan nilai parameter dari pengendali(kontroler) secara cepat, mudah dan akurat.
3. Perlu penelitian untuk penentuan fungsi alih dan parameter yang benar-benar sesuai dengan sistem yang ada pada perusahaan yang bersangkutan.

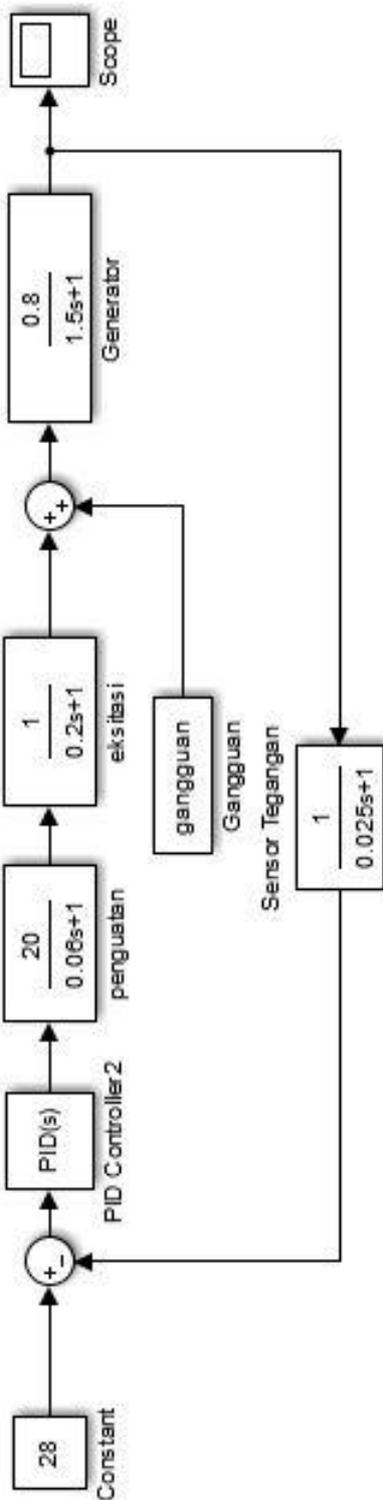
Perlu penelitian lebih lanjut untuk sistem kendali tegangan keluarangenerator pesawat N219 agar respon yang didapatkan lebih sempurna dan dapat diimplementasikan secara real.

DAFTAR PUSTAKA

- Graham, R., 1999. *Power Sistem Oscillations*. Massachusetts: Kluwer Academic Publisher.
- Gunadin, I. C. 2016. *Analisis Penerapan PID Kontroller Pada AVR (Automatic Voltage Regulator)*. *Jurnal*. Makassar: Universitas Hasanuddin.
- Kustija, J. 2014. *Listrik Pesawat*. *Jurnal*. Bandung. Universitas Pendidikan Bandung.
- Laksono, H. D. 2013. *Perilaku Tegangan Sistem Eksitasi Generator Dengan Metoda Penempatan Kutub Dalam Domain Waktu*. *Jurnal*. Padang. Universitas Andalas.
- Laksono, H. D. 2013. *Analisa Performansi Tanggapan Tegangan Sistem Eksitasi Generator Terhadap Perubahan Parameter*. *Jurnal*. Padang. Universitas Andalas.
- Niko, A. S. 2010. *Analisis Perbandingan Karakteristik Berbeban Generator Arus Searah Penguatan Bebas dengan Generator Arus Searah Penguat Shunt*. *Skripsi*. Tidak Dipublikasikan. Medan: Universitas Sumatera Utara.
- Ogata, K. 1997. *Teknik Kontrol Automatik (Sistem Pengaturan)*. Jakarta: Erlangga.
- Petruzella, Frank D. 2001. *Elektronik Industri*. Yogyakarta: Andi Yogyakarta.
- Siswojo, B. 2017. *Elektronika Kontrol*. Malang: UB Press
- Skogestad, S. & Postlethwaite, I., 1996. *Multivariable Feedback Kontrol Analysis and Design*. New York: McGraw Hill.

LAMPIRAN

PROGRAM SIMULINK



LISTING PROGRAM MATLAB

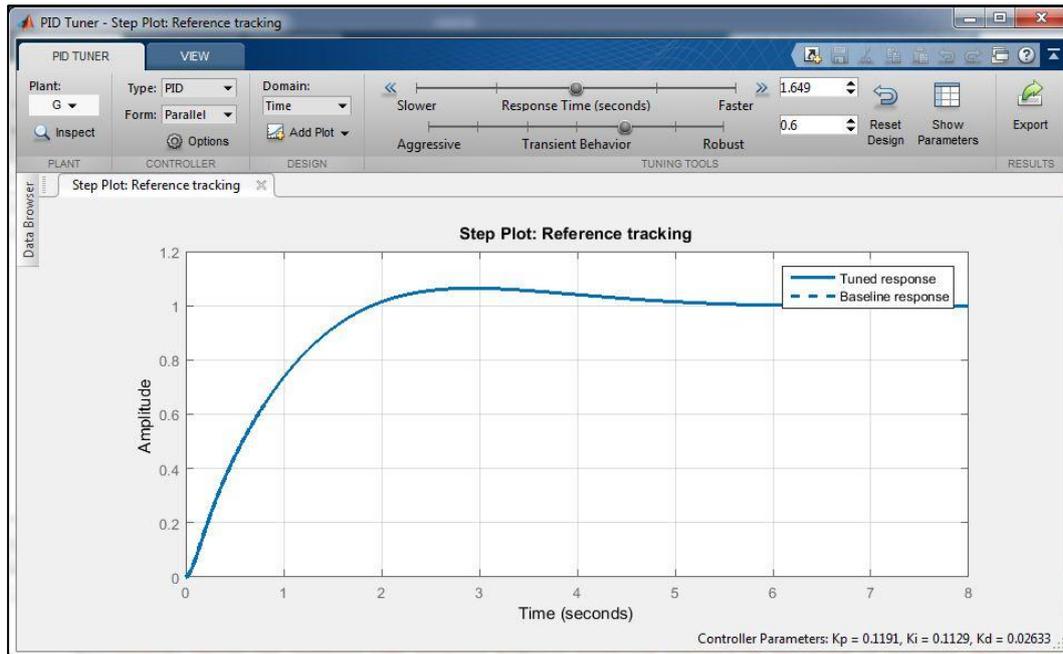
```

1 G = tf(16.0000,[0.0180 0.4020 1.7600 1.0000]); % plant model
2 H = tf(1,[0.025 1]); % feedback model
3 F = tf(G,[1 H*G]); % plant model with feedback model
4 C = pidtune(F,'PID');
5 pidTuner(F,C)

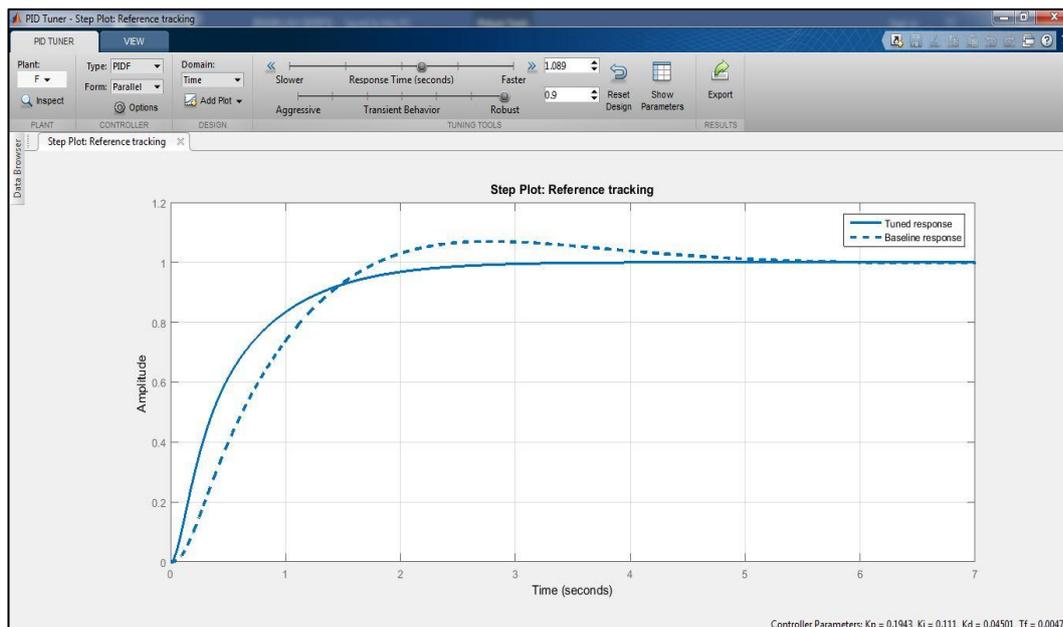
```

Penalaan dengan pidtuner

Sebelum penalaan



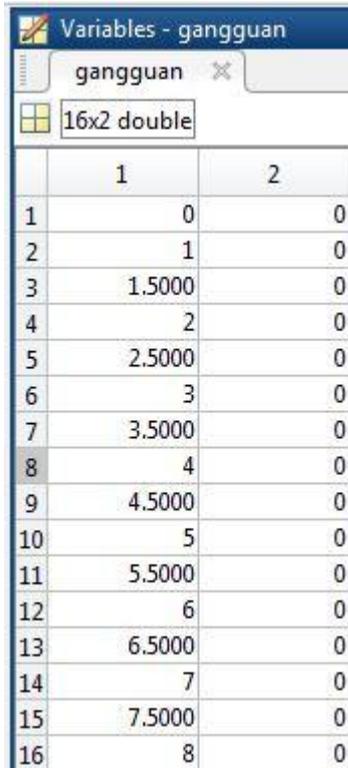
Sesudah penalaan



LAIN LAIN

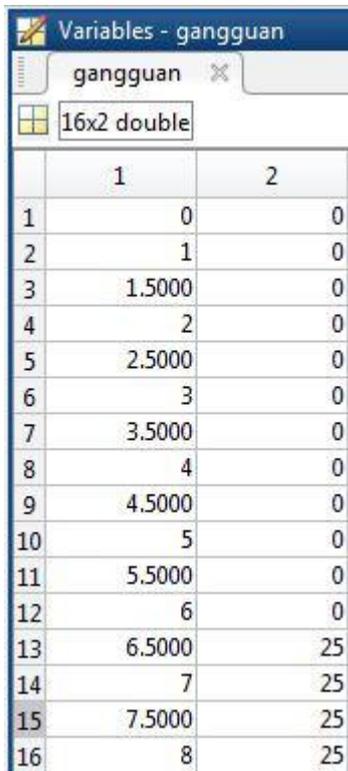
Tabel variabel beban

Beban 0 A



| | 1 | 2 |
|----|--------|---|
| 1 | 0 | 0 |
| 2 | 1 | 0 |
| 3 | 1.5000 | 0 |
| 4 | 2 | 0 |
| 5 | 2.5000 | 0 |
| 6 | 3 | 0 |
| 7 | 3.5000 | 0 |
| 8 | 4 | 0 |
| 9 | 4.5000 | 0 |
| 10 | 5 | 0 |
| 11 | 5.5000 | 0 |
| 12 | 6 | 0 |
| 13 | 6.5000 | 0 |
| 14 | 7 | 0 |
| 15 | 7.5000 | 0 |
| 16 | 8 | 0 |

Beban 25 A



| | 1 | 2 |
|----|--------|----|
| 1 | 0 | 0 |
| 2 | 1 | 0 |
| 3 | 1.5000 | 0 |
| 4 | 2 | 0 |
| 5 | 2.5000 | 0 |
| 6 | 3 | 0 |
| 7 | 3.5000 | 0 |
| 8 | 4 | 0 |
| 9 | 4.5000 | 0 |
| 10 | 5 | 0 |
| 11 | 5.5000 | 0 |
| 12 | 6 | 0 |
| 13 | 6.5000 | 25 |
| 14 | 7 | 25 |
| 15 | 7.5000 | 25 |
| 16 | 8 | 25 |

Beban 50 A

| | 1 | 2 |
|----|--------|-----|
| 1 | 0 | 0 |
| 2 | 1 | 0 |
| 3 | 1.5000 | 0 |
| 4 | 2 | 0 |
| 5 | 2.5000 | 0 |
| 6 | 3 | 0 |
| 7 | 3.5000 | 0 |
| 8 | 4 | 0 |
| 9 | 4.5000 | 0 |
| 10 | 5 | 0 |
| 11 | 5.5000 | 0 |
| 12 | 6 | 0 |
| 13 | 6.5000 | 100 |
| 14 | 7 | 100 |
| 15 | 7.5000 | 100 |
| 16 | 8 | 100 |

Beban 100 A

| | 1 | 2 |
|----|--------|----|
| 1 | 0 | 0 |
| 2 | 1 | 0 |
| 3 | 1.5000 | 0 |
| 4 | 2 | 0 |
| 5 | 2.5000 | 0 |
| 6 | 3 | 0 |
| 7 | 3.5000 | 0 |
| 8 | 4 | 0 |
| 9 | 4.5000 | 0 |
| 10 | 5 | 0 |
| 11 | 5.5000 | 0 |
| 12 | 6 | 0 |
| 13 | 6.5000 | 50 |
| 14 | 7 | 50 |
| 15 | 7.5000 | 50 |
| 16 | 8 | 50 |

Beban 200 A

| | 1 | 2 |
|----|--------|-----|
| 1 | 0 | 0 |
| 2 | 1 | 0 |
| 3 | 1.5000 | 0 |
| 4 | 2 | 0 |
| 5 | 2.5000 | 0 |
| 6 | 3 | 0 |
| 7 | 3.5000 | 0 |
| 8 | 4 | 0 |
| 9 | 4.5000 | 0 |
| 10 | 5 | 0 |
| 11 | 5.5000 | 0 |
| 12 | 6 | 0 |
| 13 | 6.5000 | 200 |
| 14 | 7 | 200 |
| 15 | 7.5000 | 200 |
| 16 | 8 | 200 |
| 17 | | |

Tabel parameter fungsi alih yang digunakan

| Parameter | nilai |
|-----------|---------|
| K_A | 20,0000 |
| τ_A | 0,0600 |
| K_E | 1,0000 |
| τ_E | 0,2000 |
| K_G | 0,8000 |
| τ_G | 1,5000 |

- Dalam figur 1 garis grafik

```

y=A(:,2);
x=A(:,1);
plot(x,y)
grid
ylabel('Tegangan (volt)')
xlabel('Waktu (detik)')
title('')

```

- ❖ Gambar grafik pada scope Matlab



- ❖ Gambar grafik pada figure Matlab

