

ANALISIS PERAN PID TERHADAP KINERJA *SLIDING MODE* *CONTROLLER* (SMC) PADA SISTEM *AUTOMATIC* *VOLTAGE REGULATOR* (AVR)

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana Teknik
Pada Program Studi Teknik Elektro Fakultas Sains Dan Teknologi



Oleh :

SAYYID ROBBI AL HIBBAN
12250510344

PRODI TEKNIK ELEKTRO

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU

PEKANBARU

2025

- Hak Cipta Dilindungi Undang-Undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



LEMBAR PERSETUJUAN

© Hak cipta milik UIN Suska Riau

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

ANALISIS PERAN PID TERHADAP KINERJA *SLIDING MODE* *CONTROLLER (SMC)* PADA SISTEM *AUTOMATIC* *VOLTAGE REGULATOR (AVR)*

TUGAS AKHIR

Oleh:

Sayyid Robbi Al Hibban
12250510344

Telah diperiksa dan disetujui sebagai Laporan Tugas Akhir Program Studi Teknik Elektro
di Pekanbaru, Pada Tanggal 29 Desember 2025

Ketua Program Studi Teknik Elektro

Dr. Hj. Lilliana, S.T., M.Eng
NIP: 19781012 200312 2 004

Pembimbing

Dr. Dian Mursyitah, S.T., M.T.
NIP: 19870906 201503 2 006

State Islamic University of Sultan Syarif Kasim Riau



LEMBAR PENGESAHAN

ANALISIS PERAN PID TERHADAP KINERJA *SLIDING MODE* *CONTROLLER* (SMC) PADA SISTEM *AUTOMATIC* *VOLTAGE REGULATOR* (AVR)

TUGAS AKHIR

Oleh :

SAYYID ROBBI AL HIBBAN
12250510344

Telah dipertahankan di depan Sidang Dewan Penguji
sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik
Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau
di Pekanbaru, pada tanggal 29 Desember 2025

Pekanbaru, 29 Desember 2025

Mengesahkan,

Dekan Fakultas Sains dan Teknologi

Ketua Program Studi Teknik Elektro

Dr. Yuslenita Muda, S.Si., M.Sc.
NIP. 19770103 200710 2 001

Dr. Hj. Liliana, ST., M.Eng
NIP. 19781012 200312 2 004

Dewan Penguji :

Ketua : Dr. Fitri Amillia, S.T., M.T.

Sekretaris : Dr. Dian Mursyitah, S.T., M.T.

Anggota I : Aulia Ullah, S.T., M.Eng.

Anggota II : Hilman Zarory, S.T., M.Eng.

- Hak Cipta Dilindungi Undang-Undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

© Hak cipta milik UIN Suska Riau

LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL

Tugas Akhir yang tidak diterbitkan ini terdaftar dan tersedia di Perpustakaan Universitas Islam Negeri Sultan Syarif Kasim Riau dan terbuka untuk umum dengan ketentuan bahwa hak cipta ada pada penulis. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau ringkasan hanya dapat dilakukan dengan mengikuti kaidah pengutipan yang berlaku.

Penggandaan atau penerbitan sebagian atau seluruh Tugas Akhir ini harus memperoleh izin dari Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau. Perpustakaan yang meminjamkan Tugas Akhir ini untuk anggotanya diharapkan untuk mengisi nama, tanda peminjaman dan tanggal pinjam.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Sayyid Robbi Al Hibban
 NIM : 12250510344
 Tempat/Tgl. Lahir : Pekanbaru, 03 Mei 2004
 Fakultas : Sains dan Teknologi
 Prodi : Teknik Elektro
 Judul Artikel :

ANALISIS PERAN PID TERHADAP KINERJA *SLIDING MODE* *CONTROLLER (SMC) PADA SISTEM AUTOMATIC* *VOLTAGE REGULATOR (AVR)*

Menyatakan dengan sebenar-benarnya bahwa:

1. Penulis Artikel dengan judul sebagaimana tersebut di atas adalah hasil pemikiran dan penelitian saya sendiri.
2. Semua kutipan pada Karya Tulis saya ini sudah disebutkan sumbernya.
3. Oleh karena itu Artikel saya ini sah, saya nyatakan bebas dari plagiasi.
4. Apabila dikemudian hari terbukti terdapat plagiat dalam Artikel saya tersebut, maka saya bersedia menerima sanksi sesuai peraturan perundang-undangan.

Demikian surat pernyataan ini saya buat dengan penuh kesadaran dan tanpa paksaan dari pihak manapun juga.

Pekanbaru, 29 Desember 2025

Yang membuat pernyataan,



Sayyid Robbi Al Hibban
NIM. 12250510344

LEMBAR PERSEMBAHAN



Allhamdulillah Allah rabbil alamin, segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas limpahan rahmat, karunia, dan pertolongan-Nya. Atas izin dan kehendak-Nya, penulis dapat menyelesaikan karya sederhana ini sebagai bagian dari ikhtiar menuntut ilmu dan proses pendewasaan diri. Setiap langkah yang dilalui menjadi pengingat bahwa tidak ada usaha yang sia-sia ketika disertai doa dan kesabaran.

Di balik setiap halaman yang tertulis, tersimpan perjuangan panjang, kelelahan, dan harapan yang terus dijaga. Penulis menyadari sepenuhnya bahwa pencapaian ini bukanlah hasil usaha pribadi semata, melainkan buah dari dukungan, doa, dan ketulusan orang-orang terdekat yang senantiasa hadir dalam diam, menguatkan tanpa pamrih.

Karya ini penulis persembahkan dengan penuh hormat dan cinta kepada Ayah dan Ibu tercinta, Eddy Pariyanto dan Niti. Dari Ayah, penulis belajar arti keteguhan, tanggung jawab, dan kerja keras yang tak banyak diucapkan namun nyata ditunjukkan. Dari Ibu, penulis belajar tentang keikhlasan, kesabaran, dan kekuatan doa yang selalu mengiringi setiap langkah kehidupan.

Nilai-nilai yang Ayah dan Ibu tanamkan menjadi fondasi yang menuntun penulis untuk tetap bertahan, bangkit, dan melangkah hingga titik ini. Setiap keberhasilan kecil yang diraih tidak lepas dari restu dan pengorbanan yang tak terhitung nilainya.

Semoga karya ini menjadi saksi bakti penulis kepada Ayah dan Ibu, sekaligus menjadi amal yang bernilai di sisi Allah SWT. Penulis memohon agar Allah SWT senantiasa melimpahkan kesehatan, keberkahan, dan kebahagiaan kepada Ayah dan Ibu, baik di dunia maupun di akhirat, serta membimbing langkah penulis ke jalan yang diridhai-Nya.

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya. Dia mendapat (pahala) dari (Kebajikan) yang dikerjakannya dan mendapa (siksa) dari (kejahatan) yang diperbuatnya.” (QS. Al Baqarah/2:286)

*“Sebutlah namaNya, Tetap di jalanNya Kelak kau mengingat, Kau akan teringat”
(33x - Perunggu)*

SAYYID ROBBI AL HIBBAN

13 DESEMBER 2025

KATA PENGANTAR



Assalamu 'alaikum wr.wb

Puji syukur penulis panjatkan kepada Allah SWT atas segala rahmat, karunia serta hidayah-Nya, sehingga penulis dapat menyelesaikan Laporan *Tugas Akhir* ini dengan waktu. Atas karunia Allah SWT, Tugas Akhir dengan judul “**Analisis Peran PID terhadap Kinerja Sliding Mode Controller (SMC) pada Sistem Automatic Voltage Regulator (AVR)**” dapat diselesaikan penulis tepat waktu.

Dalam penulisan Tugas Akhir ini, bimbingan dan pengarahan diberikan oleh orang-orang yang memiliki pengetahuan, wawasan, dan pengalaman luar biasa, sehingga penulisan Tugas Akhir ini dapat diselesaikan dengan penuh kesederhanaan. Dalam proses penyelesaian Tugas Akhir ini, penulis menerima banyak bantuan, dorongan, dan motivasi dari berbagai pihak. Oleh karena itu, penghargaan dan ucapan terima kasih yang tak terhingga disampaikan kepada :

1. Kepada Allah SWT, dengan rahmat-Nya dan hidayah-Nya, telah memberikan yang terbaik sehingga penyusunan laporan ini dapat berjalan dengan lancar.
2. Kepada kedua orang tua ayahanda Eddy Pariyanto dan ibunda Niti. Terimakasih telah selalu mendoakan dan semua pengorbanan, kepercayaan, dan nasehat yang diberikan sehingga penulis dapat menyelesaikan Tugas Akhir ini.
3. Kepada ketiga saudara kandung adik, Mu'adz Hilal Abbad, Zaid Ariq Ahmad Dharbi, dan Sayyidina Abdu Khudzaifah. Terimakasih atas segala doa, motivasi, dan semangat nya dalam proses penulisan Tugas Akhir ini.
4. Ibu Prof. Dr. Leny Nofianti MS., S.E., M. Si., Ak selaku Rektor UIN Suska Riau beserta kepada seluruh staf dan jajarannya..
5. Ibu Dr. Yuslenita Muda, M.Sc selaku Dekan Fakultas Sains dan Teknologi UIN Suska Riau beserta kepada seluruh Staf dan jajarannya.
6. Ibu Dr. Liliana, S.T., M.Eng selaku Kepala Program Studi Teknik Elektro Fakultas Sains dan Teknologi Uin Suska Riau.
7. Bapak Aulia Ulah, S.T., M.Eng selaku sekretaris jurusan Teknik Elektro Fakultas Sains dan Teknologi Uin Suska Riau.
8. Ibu Dr. Dian Mursyitah, S.T., M.T selaku dosen pembimbing yang telah banyak



Hak Cipta Dimiliki UIN Suska Riau

9.

10.

11.

12.

meluangkan waktu serta pemikirannya dengan ikhlas dalam memberikan penjelasan dan masukan yang sangat berguna sehingga penulis menjadi lebih mengerti dalam menyelesaikan Laporan Tugas Akhir ini.

Ibu Rika Susanti, S.T., M.Eng. selaku Dosen Pembimbing Akademik selama perkuliahan penulis dari awal semester hingga akhir semester.

Ucapan terima kasih juga saya sampaikan kepada diri sendiri, atas usaha keras dan keteguhan untuk bertahan sejauh ini. Kemampuan dalam mengatur waktu, tenaga, dan pikiran memungkinkan penyelesaian Tugas Akhir ini tepat waktu.

Terima kasih banyak penulis ucapkan kepada Faladina Septiani yang sudah bersedia menemani penulis disaat pengerjaan tugas akhir ini sejak awal tugas akhir ini ditulis hingga terselesaikannya tugas ini.

Teman-Teman seperjuangan dalam Konsentrasi Instrumentasi 2022 serta teman-teman teknik elektro angkatan 2022 lainnya yang juga telah memberikan banyak dukungan kepada penulis dalam menyelesaikan Tugas Akhir ini.

Sebagai manusia biasa penulis menyadari bahwa penulisan Tugas Akhir ini masih terdapat kekurangan dan jauh dari kata sempurna karena keterbatasan ilmu pengetahuan, kemampuan dan pengetahuan yang dimiliki penulis. Semua kekurangan hanya datang dari penulis dan kesempurnaan hanya milik Allah SWT, hal ini yang membuat penulis menyadari bahwa dalam pembuatan Tugas Akhir ini masih jauh dari kesempurnaan.

Wassalamu'alaikum wr:wb

Pekanbaru, 16 Oktober 2025

Penulis

Sayyid Robbi Al Hibban

12250510344

ANALISIS PERAN PID TERHADAP KINERJA *SLIDING MODE CONTROLLER* (SMC) PADA SISTEM *AUTOMATIC VOLTAGE REGULATOR* (AVR)

SAYYID ROBBI AL HIBBAN
NIM. 122505010344

Tanggal Sidang : 29 Desember 2025

Program Studi Teknik Elektro

Fakultas Sains dan Teknologi

Universitas Islam Negeri Sultan Syarif Kasim Riau

Jl. Soebrantas No.155, Pekanbaru

ABSTRAK

Sistem *Automatic Voltage Regulator* (AVR) memiliki peran penting dalam menjaga kestabilan tegangan terminal generator sinkron, terutama saat terjadi perubahan beban dan gangguan eksternal. Pengendali konvensional seperti PID banyak digunakan, namun kurang efektif dalam menghadapi ketidakpastian parameter dan dinamika *plant* yang kompleks. Metode *Sliding Mode Control* (SMC) dikenal memiliki sifat *robust*, akan tetapi SMC *static* tidak mampu menangani sistem orde tinggi seperti AVR orde tiga dan secara teoritis cenderung menimbulkan fenomena *chattering* yang menyebabkan respons sistem kurang halus. Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan pendekatan *Hybrid SMC-PID* yang mengombinasikan ketangguhan SMC dengan kemampuan PID sebagai kompensator dalam mereduksi *chattering* dan memperhalus dinamika transien sistem. Metodologi penelitian meliputi pemodelan matematis AVR orde tiga, perancangan pengendali SMC, serta integrasi PID sebagai pengendali pendamping. Simulasi dilakukan pada berbagai skenario pengujian, meliputi perubahan referensi, gangguan *noise*, gangguan *step disturbance*, dan perubahan beban. Hasil simulasi menunjukkan bahwa pengendali *Hybrid SMC-PID* mampu meningkatkan kinerja regulasi tegangan dengan nilai *overshoot* rendah sekitar 0,48%, waktu tunak berkisar antara 15–17 detik, serta *steady-state error* di bawah 0,2 V pada kondisi perubahan beban, yang secara signifikan lebih kecil dibandingkan pengendali SMC statis. Selain itu, pada pengujian gangguan *noise* pada sinyal keluaran, sistem tetap stabil dengan fluktuasi tegangan yang terkendali di sekitar *setpoint*. Hasil ini membuktikan bahwa pendekatan *Hybrid SMC-PID* efektif dalam mengatasi keterbatasan SMC statis pada sistem AVR orde tiga serta meningkatkan kestabilan, kehalusan respons, dan *robustness* sistem secara keseluruhan.

Kata kunci : *Automatic Voltage Regulator, Sliding Mode Control, PID, Hybrid Control, Robustness, Chattering Reduction*

ANALYSIS OF THE ROLE OF PID ON THE PERFORMANCE OF SLIDING MODE CONTROLLER (SMC) IN AUTOMATIC VOLTAGE REGULATOR (AVR) SYSTEM

SAYYID ROBBI AL HIBBAN
NIM. 122505010344

Date of Final Exam: December 29, 2025

*Department of Electrical Engineering
Faculty of Science and Technology
State Islamic University of Sultan Syarif Kasim Riau
Soebrantas St. No.155 Pekanbaru - Indonesia*

ABSTRACT

The Automatic Voltage Regulator (AVR) system plays a crucial role in maintaining terminal voltage stability of synchronous generators, particularly under load variations and external disturbances. Conventional controllers such as PID are widely applied; however, they are less effective in handling parameter uncertainties and complex plant dynamics. Sliding Mode Control (SMC) is known for its robustness, yet static SMC is unable to effectively handle high-order systems such as third-order AVR systems and theoretically tends to induce chattering, resulting in unsmooth system responses. To address these limitations, this study proposes a Hybrid SMC–PID control approach that combines the robustness of SMC with the capability of PID as a compensator to reduce chattering and smooth transient dynamics. The research methodology includes mathematical modeling of a third-order AVR system, SMC controller design, and PID integration as a compensatory controller. Simulations are conducted under various test scenarios, including reference changes, noise disturbances, step disturbances, and load variations. Simulation results indicate that the proposed Hybrid SMC–PID controller improves voltage regulation performance, achieving a low overshoot of approximately 0.48%, a settling time in the range of 15–17 seconds, and a steady-state error below 0.2 V during load change conditions, which is significantly smaller compared to static SMC. Furthermore, under output noise disturbances, the system remains stable with controlled voltage fluctuations around the setpoint. These results demonstrate that the proposed Hybrid SMC–PID approach effectively overcomes the limitations of static SMC in third-order AVR systems while enhancing system stability, response smoothness, and overall robustness.

Keywords : Automatic Voltage Regulator, Sliding Mode Control, PID, Hybrid Control, Robustness, Chattering Reduction

DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL	iv
LEMBAR PERSEMBAHAN	vi
KATA PENGANTAR	vii
ABSTRAK	ix
ABSTRACT	x
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
DAFTAR RUMUS	xv
DAFTAR LAMBANG	xvi
DAFTAR SINGKATAN	xvii
BAB I PENDAHULUAN	I-1
1.1 Latar Belakang	I-1
1.2 Rumusan Masalah	I-4
1.3 Tujuan Penelitian	I-4
1.4 Batasan Masalah	I-4
1.5 Manfaat Penelitian	I-5
BAB II KAJIAN PUSTAKA	II-1
2.1 Penelitian Terkait	II-1
2.2 Dasar Teori	II-3
2.2.1 <i>Automatic Voltage Regulator (AVR)</i>	II-3
2.2.2 Sistem <i>Exciter</i> pada Generator	II-10



2.2.3	<i>Sliding Mode Control (SMC)</i>	II-13
2.2.4	Kontrol PID (<i>Proportional-Integral-Derrivative</i>)	II-19
2.2.5	Respon Transien Sistem Orde II	II-22
2.2.6	<i>Software</i> MATLAB/Simulink	II-24

BAB III METODOLOGI PENELITIAN III-1

1	Diagram Alur Penelitian	III-1
2	Tahapan Penelitian.....	III-2

BAB IV HASIL DAN ANALISA IV-1

4.1	Analisa Hasil Sistem Tanpa Gangguan.....	IV-1
4.2	Pengujian dan Analisa Sistem Tanpa Pengendali	IV-1
4.3	Pengujian dan Analisa Respons terhadap Pencapaian <i>Setpoint</i>	IV-3
4.4	Komparasi Hasil Pengujian	IV-10
4.5	Pengujian dan Analisa Sistem dengan Gangguan (<i>Disturbance</i>)	IV-12
4.5.1	Analisa Hasil Sistem dengan Gangguan pada Sinyal <i>Input</i> (<i>yr</i>).....	IV-13
4.5.2	Analisa Hasil Sistem dengan Gangguan pada Sinyal Kendali (<i>u_total</i>).....	IV-16
4.5.3	Analisa Hasil Sistem dengan Gangguan pada Sinyal Keluaran (<i>y</i>).....	IV-18
4.6	Analisa Hasil Sistem dengan Gangguan (<i>Noise</i>).....	IV-21
4.7	Analisa Hasil Sistem dengan Perubahan Beban (<i>Load Change</i>).....	IV-24

BAB V KESIMPULAN DAN SARAN V-1

5.1	Kesimpulan.....	V-1
5.2	Saran	V-2

DAFTAR PUSTAKA

LAMPIRAN A

LAMPIRAN B

DAFTAR RIWAYAT HIDUP



DAFTAR GAMBAR

Gambar

Halaman

Gambar 2.1	Blok Diagram AVR pada <i>Exciter</i> Generator	II-3
Gambar 2.2	Rangkaian RC <i>Low-pass filter</i>	II-6
Gambar 2.3	Blok Diagram AVR <i>Close-loop</i>	II-9
Gambar 2.4	Komponen jenis <i>brushless excitation system</i>	II-12
Gambar 2.5	Ilustrasi fase <i>reaching</i> dan <i>sliding process</i>	II-16
Gambar 2.6	Logo <i>Software</i> Matlab	II-24
Gambar 2.7	Logo Matlab dengan Simulink	II-24
Gambar 3.1	Diagram Alur Penelitian	III-2
Gambar 3.2	Rangkaian blok Simulink <i>close-loop</i> sistem AVR	III-4
Gambar 3.3	Rangkaian blok Simulink Kontrol SMC	III-10
Gambar 3.4	Rangkaian blok Simulink gabungan SMC dengan AVR	III-11
Gambar 3.5	Rangkaian Blok Simulink PID dan AVR	III-13
Gambar 3.6	Rangkaian Blok Simulink <i>hybrid</i> SMC-PID pada AVR	III-17
Gambar 4.1	Hasil simulasi AVR tanpa pengendali	IV-2
Gambar 4.2	Hasil simulasi SMC dan AVR	IV-6
Gambar 4.3	Hasil simulasi kontrol <i>hybrid</i> SMC-PID dan AVR	IV-8
Gambar 4.4	Hasil komparasi pengujian AVR, SMC, dan <i>Hybrid</i>	IV-10
Gambar 4.5	Bentuk sinyal gangguan <i>disturbance</i>	IV-13
Gambar 4.6	Hasil simulasi AVR <i>hybrid</i> dengan gangguan <i>step disturbance</i> di sinyal <i>input</i>	IV-14
Gambar 4.7	Hasil simulasi AVR Hybrid dengan gangguan <i>step disturbance</i> di sinyal kendali	IV-16
Gambar 4.8	Hasil simulasi AVR Hybrid dengan gangguan <i>step disturbance</i> di sinyal output	IV-19
Gambar 4.9	Bentuk <i>noise</i>	IV-22
Gambar 4.10	Hasil simulasi AVR <i>hybrid</i> dengan gangguan <i>noise</i>	IV-23
Gambar 4.11	Rangkaian Simulink AVR dengan Komponen Beban	IV-24
Gambar 4.12	Hasil simulasi AVR <i>hybrid</i> dengan gangguan <i>load change</i>	IV-25

DAFTAR TABEL

Tabel	Halaman
Tabel 2.1 Bentuk <i>Transfer Function</i> per Komponen	II-8
Tabel 2.2 Data-data sistem AVR [24]	II-9
Tabel 2.3 hubungan konstanta waktu (τ) dengan karakteristik respon.....	II-23
Tabel 2.4 Blok diagram Simulink	II-26
Tabel 3.1 Algoritma Pengujian Sistem AVR	III-4
Tabel 3.2 Parameter SMC hasil tuning.....	III-8
Tabel 3.3 Algoritma Desain SMC pada Sistem AVR	III-11
Tabel 3.4 Parameter PID hasil tuning.....	III-13
Tabel 3.5 Algoritma Desain PID pada Sistem AVR	III-13
Tabel 3.6 Parameter <i>hybrid</i> SMC–PID hasil <i>tuning</i>	III-16
Tabel 3.7 Algoritma Desain PID pada Sistem AVR	III-17
Tabel 4.1 Hasil pengujian AVR tanpa pengendali	IV-2
Tabel 4.2 Hasil Pengujian kombinasi parameter SMC	IV-4
Tabel 4.3 Hasil pengujian AVR dengan SMC	IV-6
Tabel 4.4 Hasil pengujian AVR dengan kontrol <i>Hybrid</i>	IV-9
Tabel 4.5 Hasil informasi komparasi pengujian AVR, SMC, dan <i>Hybrid</i>	IV-11
Tabel 4.6 Hasil pengujian AVR <i>Hybrid</i> dengan gangguan <i>disturbance</i> di <i>input</i>	IV-15
Tabel 4.7 Hasil pengujian AVR <i>Hybrid</i> dengan gangguan <i>disturbance</i> di sinyal kendali	IV-17
Tabel 4.8 Hasil pengujian AVR <i>Hybrid</i> dengan gangguan <i>disturbance</i> di sinyal <i>output</i>	IV-20
Tabel 4.9 Hasil pengujian AVR <i>hybrid</i> dengan gangguan <i>load change</i>	IV-26

DAFTAR RUMUS

Rumus

Halaman

2.1	Fungsi Alih <i>Amplifier</i>	II-5
2.2	Fungsi Alih <i>Exciter</i>	II-5
2.3	Fungsi Alih Generator.....	II-5
2.4	Loop Tegangan <i>Input</i> V_{in}	II-6
2.5	Loop Tegangan <i>Output</i> V_{out}	II-6
2.8	Transformasi <i>Laplace</i> Rangkaian RC	II-7
2.9	Konstanta Waktu Rangkaian RC	II-7
2.10	Fungsi Alih Konstanta Waktu Rangkaian RC	II-7
2.13	Fungsi Alih Setiap Komponen	II-7
2.14	Fungsi Alih Komponen AVR	II-9
2.15	Nilai Besaran <i>Exciter</i>	II-11
2.16	Persamaan <i>Error</i>	II-13
2.18	Persamaan <i>Sliding Surface</i>	II-14
2.21	Persamaan <i>reaching law</i>	II-16
2.22	Komponen SMC	II-17
2.23	Persamaan Komponen <i>Switching</i>	II-17
2.24	Persamaan Fungsi Lyapunov	II-17
2.29	Persamaan PID	II-21
3.1	Fungsi Alih AVR	III-4
3.6	Persamaan <i>error</i>	III-6
3.8	Persamaan Permukaan Luncur AVR	III-6
3.13	Persamaan Komponen Equivalen SMC	III-7
3.14	Persamaan <i>Lyapunov Reaching Law</i>	III-7
3.15	Persamaan <i>Boundary Layer</i>	III-7
3.16	Persamaan Komponen Natural SMC	III-8
3.18	Persamaan SMC	III-8
3.19	Persamaan PID	III-12
3.20	Persamaan <i>Hybrid</i>	III-16

DAFTAR LAMBANG

- τ (Tau) : Konstanta Waktu
- y : Sinyal Keluaran Sistem
- u : Sinyal Kendali (Kontrol)
- u_{total} : Sinyal Kendali Total pada Sistem *Hybrid*
- u_{eq} : Kontrol *equivalent* SMC
- $u_{natural}$: Kontrol natural SMC
- λ_1, λ_0 : Parameter *Sliding Surface* SMC
- η : *Switching Gain* pada kontrol SMC
- ε : Lebar *boundary layer* untuk fungsi saturasi

© Hak cipta milik UIN Suska Riau

State Islamic University of Sultan Syarif Kasim Riau

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

DAFTAR SINGKATAN

© Hak cipta milik UIN Suska Riau

State Islamic University of Sultan Syarif Kasim Riau

AVR	: <i>Automatic Voltage Regulator</i>
SMC	: <i>Sliding Mode Control</i>
PID	: <i>Proportional-Integral-Derivative</i>
TF	: <i>Transfer Function</i>
V_{ref}	: <i>Voltage Reference</i>
V_s	: Tegangan Sensor (<i>Sensing Voltage</i>)
V_t	: Tegangan Terminal (<i>Terminal Voltage</i>)
K_a	: <i>Gain Amplifier</i>
K_e	: <i>Gain Exciter</i>
K_g	: <i>Gain Generator</i>
RC	: <i>Resistor-Capacitor</i>
DC	: <i>Direct Current</i>
AC	: <i>Alternating Current</i>
K_p	: <i>Konstanta Proporsional</i>
K_i	: <i>Konstanta Integral</i>
K_d	: <i>Konstanta Derivatif</i>
RMSE	: <i>Root Mean Square Error</i>
T_r	: <i>Time rise</i>
T_s	: <i>Time settling</i>
T_d	: <i>Time delay</i>
ESS	: <i>Error Steady State</i>
OS	: <i>Overshoot</i>
Sign	: <i>Sign Function</i>
Sat	: <i>Saturation Function</i>

- Hak cipta ini dilindungi Undang-Undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
- Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

BAB I

PENDAHULUAN

1. Latar Belakang

Sistem pembangkit listrik merupakan elemen kritis dalam memenuhi kebutuhan energi elektrik global, baik untuk konsumsi industri maupun rumah tangga. Generator sinkron mengonversi energi mekanik menjadi energi listrik dalam skala besar dan terkontrol. Fluktuasi tegangan seperti *overload* harus dihindari sehingga kestabilan tegangan pada sistem terjaga. Generator sinkron dilengkapi dengan sistem *exciter*, yang mengatur arus yang mengalir ke kumparan medan (*field winding*) untuk menghasilkan fluks magnetik yang sesuai. Arus *exciter* ini memiliki dampak langsung terhadap tegangan terminal generator, sehingga pengendalian *exciter* menjadi elemen kunci dalam menjaga stabilitas sistem [1]. Tegangan *output* generator secara langsung dipengaruhi oleh besar arus *exciter*. Semakin besar arus *exciter*, semakin besar fluks magnetik yang dihasilkan, sehingga tegangan terminal generator juga meningkat. Sebaliknya, jika arus *exciter* berkurang, fluks magnetik akan menurun, menyebabkan penurunan tegangan *output*.

Berdasarkan Standar PLN (SPLN) No. 1:1978, tegangan yang diperbolehkan pada sistem distribusi adalah maksimum +5% dan minimum -10% dari nilai nominal, di mana tegangan nominal yang ditetapkan PLN adalah 220 V (zona-netral) dan 380 V (zona-zona). Dengan demikian, pada sistem 220 V, rentang operasional yang diizinkan berada antara 198 V hingga 231 V. Ketika tegangan melebihi atau kurang dari batas tersebut, sistem dikategorikan mengalami kondisi *overvoltage* atau *undervoltage*. Kondisi ini tidak hanya melanggar standar kualitas daya, tetapi juga dapat menyebabkan kerusakan pada komponen sistem seperti *amplifier*, sensor umpan balik, serta *rectifier* yang mengakibatkan munculnya panas berlebih dan saturasi pada penguat [2].

Fluktuasi tegangan tersebut dapat muncul akibat perubahan beban, variasi kecepatan rotor, maupun gangguan eksternal yang menyebabkan kestabilan tegangan terminal generator terganggu [3]. Oleh karena itu, diperlukan sistem pengatur yang mampu menjaga tegangan agar tetap berada dalam batas toleransi yang diizinkan untuk mencegah *overexcitation* pada *exciter* dan memastikan kestabilan pasokan daya listrik [2, 4]. Dalam hal ini, *Automatic Voltage Regulator* (AVR) berperan penting sebagai sistem yang mengatur besar arus *exciter* guna mempertahankan tegangan terminal generator agar tetap stabil meskipun terjadi perubahan beban maupun gangguan eksternal [2].

AVR merupakan komponen pada sistem *exciter* generator yang berfungsi mengatur arus *exciter*. AVR bekerja berdasarkan perbedaan antara tegangan referensi dan tegangan aktual generator. Ketika tegangan *output* menurun akibat peningkatan beban, AVR meningkatkan arus *exciter* dengan menaikkan tegangan; sebaliknya, jika tegangan terlalu tinggi, AVR mengurangi arus *exciter* untuk menurunkan tegangan. Dengan kemampuan ini, AVR tidak hanya memastikan tegangan *output* tetap pada nilai *setpoint* yang diinginkan, tetapi juga mengurangi risiko gangguan pada jaringan distribusi serta meningkatkan keandalan sistem pembangkit. Selain itu, AVR berperan penting dalam menjaga kualitas daya listrik dengan meredam fluktuasi tegangan yang dapat mengganggu operasi peralatan listrik seperti motor dan transformator [5-7].

AVR konvensional memiliki beberapa keterbatasan dalam menjaga kestabilan tegangan pada kondisi operasi yang dinamis. Salah satu kelemahan utamanya adalah respons yang kurang cepat terhadap gangguan mendadak seperti perubahan beban atau fluktuasi parameter sistem. Selain itu, AVR konvensional cenderung kurang mampu menangani ketidakpastian sistem yang bersifat nonlinier, sehingga rentan mengalami osilasi tegangan atau penurunan performa regulasi saat terjadi gangguan eksternal [6]. Selain itu, gangguan eksternal seperti gangguan jaringan, fluktuasi frekuensi, serta perubahan parameter sistem akibat suhu, usia komponen, atau variasi kondisi operasi juga sering terjadi dan mempersulit kontrol tegangan secara presisi [8]. Kondisi-kondisi ini menuntut adanya strategi kontrol yang lebih adaptif dan *robust* untuk memastikan respons sistem tetap stabil meskipun dalam kondisi tidak pasti.

Sliding Mode Controller (SMC) merupakan salah satu metode kendali nonlinier yang banyak digunakan karena memiliki keunggulan dalam hal kestabilan dan ketahanan terhadap ketidakpastian parameter sistem. Secara teoretis, SMC dirancang untuk memaksa dinamika sistem menuju suatu permukaan *sliding* tertentu dan mempertahankannya pada permukaan tersebut, sehingga respon sistem menjadi *robust* terhadap variasi model. Namun, dalam literatur klasik SMC, metode ini sering dikaitkan dengan potensi munculnya fenomena *chattering*, yaitu osilasi frekuensi tinggi pada sinyal kendali yang disebabkan oleh penggunaan fungsi pensaklaran diskrit (*sign function*) dalam hukum kendali. Untuk mengatasi hal tersebut, implementasi praktis SMC umumnya menggunakan pendekatan *boundary layer* melalui fungsi saturasi agar sinyal kendali menjadi lebih halus dan kontinu di sekitar permukaan *sliding* [9, 10].

Dalam konteks penerapan pada sistem AVR, karakteristik sistem yang bersifat orde lebih tinggi menyebabkan SMC *static* dengan permukaan *sliding* tetap lebih berperan sebagai pengendali stabilisasi dibandingkan pengendali pelacakan (*tracking controller*) yang menjamin pencapaian *setpoint* secara presisi. Oleh karena itu, meskipun SMC mampu menghasilkan respon yang cepat dan stabil, keberadaan *error* keadaan tunak masih dapat terjadi sebagai konsekuensi dari struktur permukaan *sliding static* yang tidak secara eksplisit memaksa *error* menuju nol. Kondisi ini menunjukkan bahwa keterbatasan SMC *static* bukan terletak pada kestabilan sistem, melainkan pada ketepatan *tracking* terhadap nilai referensi, sehingga diperlukan pengembangan atau penggabungan metode kendali lain untuk meningkatkan akurasi *steady-state* pada sistem AVR [11, 12].

Namin dalam literatur SMC *static*, metode ini sering dikaitkan dengan potensi munculnya fenomena *chattering*, salah satu pendekatan yang dapat digunakan adalah mengintegrasikan kontroler PID pada struktur SMC. Kontrol PID memiliki keunggulan dalam menghasilkan respon awal sistem yang halus, cepat, serta mudah disesuaikan terhadap perubahan kondisi operasi [5, 8], sehingga efektif dalam mempercepat pencapaian nilai referensi dan meredam osilasi awal yang mungkin muncul akibat *switching* pada SMC [8]. Akan tetapi, PID juga memiliki kelemahan, yaitu sensitivitas terhadap perubahan parameter dan ketidakmampuannya menghadapi ketidakpastian sistem secara langsung. Oleh sebab itu, integrasi SMC dan PID menjadi strategi *hybrid control* yang saling melengkapi. Kombinasi kedua kontroler ini diharapkan dapat memanfaatkan keunggulan masing-masing untuk menciptakan sistem kontrol AVR yang cepat, stabil, dan tahan gangguan. Dengan demikian, keberadaan PID diharapkan dapat berperan sebagai *compensator* dalam memperbaiki kinerja SMC, baik dari sisi kecepatan respon maupun kestabilan tegangan pada sistem AVR.

Langkah awal pada penelitian ini adalah merancang sistem AVR berbasis model orde tiga yang merepresentasikan karakteristik dinamis dari sistem *exciter* generator. Selanjutnya dilakukan perancangan kontroler SMC untuk pengendalian tegangan AVR dan analisis terhadap performanya. Hasil analisis tersebut menjadi dasar dalam mengidentifikasi kebutuhan perbaikan kinerja melalui penerapan PID sebagai *compensator*. Berdasarkan uraian tersebut, penelitian ini dilakukan untuk menganalisis peran PID terhadap kinerja kontroler SMC dalam sistem *Automatic Voltage Regulator* (AVR) berbasis model orde tiga. Simulasi dilakukan menggunakan MATLAB/Simulink dengan berbagai variasi beban dan gangguan untuk mengevaluasi perubahan karakteristik sistem terhadap parameter respon transien. Oleh karena itu, penelitian ini diberi judul: “**Analisis Peran PID terhadap Kinerja**

Sliding Mode Controller (SMC) pada Sistem Automatic Voltage Regulator (AVR)”
sebagai kontribusi dalam pengembangan sistem kontrol yang lebih optimal dan *robust* pada sistem kelistrikan.

Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana peran PID dalam meningkatkan kinerja kontroler SMC pada sistem AVR?
2. Bagaimana pengaruh integrasi PID terhadap kestabilan tegangan dan respons transien sistem AVR yang dikendalikan oleh SMC dalam menghadapi gangguan dan variasi beban?

Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Menganalisis peran PID terhadap peningkatan kinerja kontroler SMC dalam pengaturan tegangan AVR.
2. Mengevaluasi kinerja sistem kontrol integrasi PID dengan SMC dalam menjaga kestabilan tegangan *output* terhadap gangguan eksternal dan perubahan beban secara dinamis.

Batasan Masalah

Agar penelitian lebih terarah dan fokus, maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem AVR dimodelkan sebagai sistem orde tiga (*third-order transfer function*), bukan berdasarkan parameter aktual generator di lapangan karena penelitian difokuskan pada analisis performa kontrol.
2. Simulasi dilakukan menggunakan *software* MATLAB/Simulink, tanpa realisasi fisik atau pengujian terhadap perangkat keras.
3. Penelitian ini tidak mempertimbangkan parameter lain seperti daya aktif (P), daya reaktif (Q), faktor daya (*power factor*), atau efisiensi sistem.

4. Analisis kinerja SMC difokuskan pada respon sistem AVR terhadap perubahan referensi dan beban, tanpa membahas pengembangan struktur SMC lanjutan di luar SMC *static*.

Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

1. Memberikan kontribusi ilmiah dalam pengembangan metode kontroler pada sistem nonlinier seperti AVR.
2. Memberikan kontribusi ilmiah dalam memahami peran PID dalam memperbaiki performa kontrol *robust* seperti SMC pada sistem nonlinier AVR.
3. Menjadi referensi akademik dalam pengembangan strategi kontrol *hybrid* berbasis evaluasi kinerja, bukan sekadar desain struktur kontrol..
4. Memberikan solusi alternatif untuk menjaga kestabilan tegangan pada sistem pembangkitan.
5. Menjadi acuan awal untuk pengembangan kontrol cerdas pada sistem AVR di masa depan.

BAB II

KAJIAN PUSTAKA

Penelitian Terkait

Penelitian terkait pengendalian tegangan pada sistem AVR telah berkembang pesat, khususnya dalam penerapan berbagai strategi kontrol untuk meningkatkan kestabilan dan keandalan sistem. Kontroler PID masih menjadi metode yang paling banyak digunakan karena strukturnya sederhana dan kemampuannya memberikan respon cepat pada sistem linier. Namun, ketika diterapkan pada sistem yang bersifat nonlinier atau menghadapi gangguan eksternal, performa PID sering kali menurun. Oleh sebab itu, berbagai penelitian telah mengembangkan metode yang mengombinasikan PID dengan kontrol *robust* seperti SMC untuk memperoleh performa yang lebih adaptif terhadap perubahan kondisi operasi. Beberapa penelitian telah berfokus pada peningkatan kinerja kontroler PID dengan menggunakan metode optimisasi seperti *Particle Swarm Optimization* (PSO) [13], *Water Cycle Algorithm* (WCA) [14], *Neural Network* (NN) [15], dan algoritma lainnya. Pendekatan ini terbukti mampu memperbaiki kecepatan respon dan presisi tegangan pada sistem AVR. Namun, efektivitasnya masih terbatas pada sistem linier dan tidak cukup kuat menghadapi ketidakpastian nonlinier.

Pada penelitian yang berjudul *Intelligent PID Controller for Automatic Voltage Regulation* [16] menggunakan kombinasi PSO, WCA, dan NN untuk menyetel parameter kontroler PID pada sistem AVR. Hasil penelitian menunjukkan bahwa pendekatan ini dapat meningkatkan respons dinamis sistem, terutama dalam mengatasi perubahan beban mendadak. Selain itu, pada penelitian yang berjudul *Water cycle algorithm-based PID controller for AVR* juga melakukan penelitian tentang penggunaan *Water Cycle Algorithm* (WCA) untuk menyetel parameter kontroler PID pada sistem AVR [14]. Metode ini berhasil memperbaiki stabilitas dan presisi tegangan keluaran generator sinkron. Penelitian ini membuktikan bahwa penggunaan algoritma metaheuristik dapat meningkatkan kinerja kontroler PID dalam menghadapi ketidakpastian sistem.

Di sisi lain, berbagai penelitian mengusulkan penggunaan SMC sebagai alternatif pengendalian AVR karena ketangguhannya terhadap gangguan dan ketidakpastian sistem. Namun, kelemahan utama SMC adalah munculnya fenomena *chattering* akibat switching yang cepat. Beberapa penelitian seperti *Robust control for automatic voltage regulator system based on learning sliding mode control* [17] dan *Chattering-eliminated adaptive*

sliding-mode control: an experimental comparison study [18] berupaya mengurangi *chattering* melalui fungsi nonlinier tanh dan adaptasi parameter secara real-time. Hasilnya menunjukkan peningkatan kestabilan tegangan, meskipun kehalusan sinyal kontrol masih terbatas.

Sejumlah penelitian telah menggabungkan kontroler PID dengan SMC untuk memanfaatkan keunggulan masing-masing. Misalnya, penelitian *Adaptive Super-Twisting Sliding Mode Control of Permanent Magnet Synchronous Motor* [19] menunjukkan bahwa penambahan PID mampu memperbaiki kehalusan sinyal kontrol dan mempercepat respon sistem. Dalam konteks tersebut, PID tidak hanya berfungsi sebagai pengendali tambahan, tetapi juga sebagai *compensator* yang berperan memperbaiki karakteristik dinamis SMC. Dengan demikian, evaluasi terhadap pengaruh atau peran PID terhadap performa SMC menjadi aspek penting untuk dipelajari lebih lanjut, khususnya pada sistem AVR yang memiliki sifat nonlinier dan dinamis. Dalam beberapa pendekatan *cascade control*, penggabungan antara kontroler linier seperti PI dan SMC telah digunakan untuk meningkatkan ketangguhan sistem terhadap gangguan eksternal dan variasi beban. Biasanya, PI digunakan sebagai kontroler eksternal untuk menghasilkan nilai referensi, sedangkan SMC digunakan pada *loop* internal untuk meningkatkan stabilitas dan performa dinamis sistem [6].

Selain itu, sistem AVR sering kali dihadapkan pada kondisi operasi yang tidak pasti seperti variasi beban dan perubahan parameter sistem. Pada penelitian yang berjudul *Nonlinear AVR for power system stabilisers robust phase compensation design* [20] mengusulkan Nonlinear AVR dengan desain kompensasi zona untuk meningkatkan stabilitas sistem. Penelitian ini menunjukkan bahwa pendekatan nonlinear dapat memberikan performa yang lebih baik dibandingkan kontroler linier tradisional. Selain itu, penelitian yang berjudul *Optimized four-parameter PID controller for AVR systems with respect to robustness* [5] juga melakukan penelitian tentang pengoptimalan kontroler PID empat-parameter untuk sistem AVR dengan fokus pada *robustness*. Mereka menggunakan algoritma optimisasi untuk menemukan parameter kontroler yang optimal sehingga sistem tetap stabil meskipun menghadapi ketidakpastian.

Berdasarkan tinjauan tersebut, dapat disimpulkan bahwa integrasi PID ke dalam struktur SMC merupakan pendekatan yang menjanjikan untuk meningkatkan kinerja sistem AVR. Namun, kajian yang secara khusus menganalisis bagaimana peran PID dalam memperbaiki karakteristik transien, kestabilan, serta ketahanan SMC terhadap gangguan

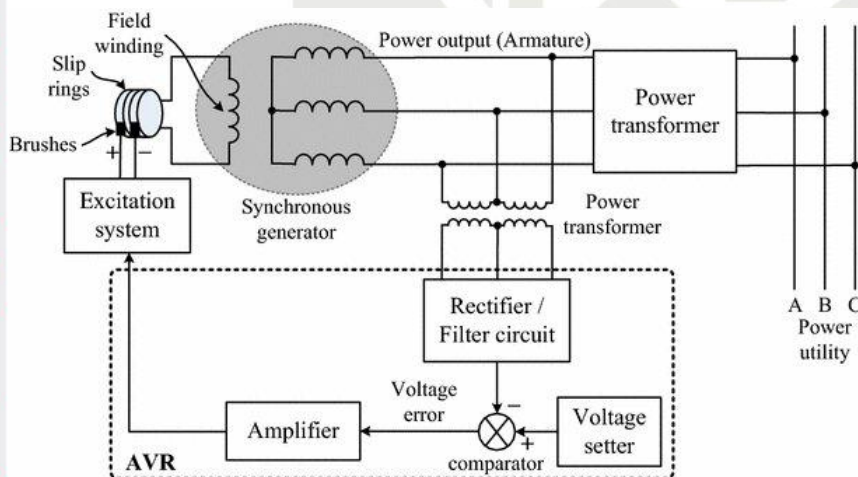
masih relatif terbatas. Oleh karena itu, penelitian ini difokuskan untuk menganalisis peran PID terhadap kinerja kontroler SMC pada sistem AVR.

2. Dasar Teori

2.1 Automatic Voltage Regulator (AVR)

Automatic Voltage Regulator (AVR) merupakan suatu sistem kontrol otomatis yang bertujuan untuk menjaga tegangan terminal generator sinkron tetap pada nilai *setpoint* meskipun terdapat gangguan atau variasi beban [17]. Sebagai bagian dari sistem *exciter* generator, AVR berperan sebagai *feedback controller* yang membandingkan tegangan aktual dengan tegangan referensi guna menghasilkan sinyal kesalahan (*error signal*) yang kemudian digunakan untuk menyesuaikan arus *exciter* [16].

Fungsi utama dari AVR meliputi tiga fungsi yang krusial yaitu menjaga tegangan terminal generator tetap stabil pada level tertentu sesuai dengan nilai *setpoint*, menyesuaikan arus *exciter* secara dinamis terhadap perubahan kondisi beban atau gangguan eksternal, dan meningkatkan kestabilan sistem tenaga secara keseluruhan dengan meredam osilasi tegangan serta memberikan respons yang cepat dan presisi [5, 21]. Dalam aplikasi praktis, seperti pada pembangkit listrik tenaga uap, gas, atau hidro, penggunaan AVR sangat krusial untuk memastikan kontinuitas pasokan listrik yang andal dan aman [17].



Gambar 2.1 Blok Diagram AVR pada *Exciter* Generator

Berdasarkan Gambar 2.1 yang disajikan adalah blok diagram sistem AVR pada generator sinkron, yang menunjukkan komponen utama dan alur kerja kontrol tegangan

secara rinci. Pada Gambar 2.1, generator sinkron berfungsi sebagai inti dari sistem, di mana tegangan keluaran dihasilkan melalui induksi elektromagnetik antara rotor dan stator. Tegangan tersebut kemudian ditingkatkan atau dikurangi menggunakan transformator daya, yang bertindak sebagai perangkat untuk mengubah tegangan sesuai dengan persyaratan jaringan utilitas [17].

Komponen penting lainnya adalah *exciter*, yang bertugas menyediakan arus *exciter* ke kumpulan medan (*field winding*) pada rotor generator melalui *slip rings* dan *brushes*. Arus *exciter* ini sangat kritis karena memengaruhi medan magnet yang dibentuk oleh rotor, sehingga mempengaruhi tegangan terminal generator. Tegangan harus tetap stabil maka AVR digunakan sebagai *feedback controller*. AVR bekerja dengan membandingkan tegangan referensi dengan tegangan aktual yang diukur menggunakan sensor tegangan. Sinyal kesalahan (*error signal*) ini kemudian diproses oleh *amplifier* untuk memberikan sinyal pengaturan ke *exciter* [5].

Setelah itu, tegangan keluaran generator diambil melalui transformator daya dan diubah menjadi tegangan DC menggunakan *rectifier/filter circuit*. Rangkaian ini bertujuan untuk menghilangkan komponen AC residual dan memberikan sinyal tegangan yang lebih stabil sebagai *input* ke kontroler AVR. Dalam skema ini, *voltage setter* bertindak sebagai pengaturan tegangan referensi, sedangkan *comparator* membandingkan tegangan referensi dengan tegangan aktual untuk menghasilkan sinyal kesalahan. Sinyal ini kemudian diperkuat oleh *amplifier* sebelum dikirim ke *exciter*, yang menghasilkan arus *exciter* yang sesuai untuk menjaga tegangan terminal pada nilai *setpoint* yang diinginkan [22].

Dalam pemodelan sistem AVR, umumnya digunakan representasi matematis dalam bentuk fungsi transfer orde satu, dua, atau tiga tergantung kompleksitas *exciter* dan respons dinamik yang diinginkan. Model orde satu sering digunakan untuk sistem sederhana dengan asumsi linearitas, sedangkan model orde lebih tinggi digunakan untuk menangkap karakteristik dinamik nonlinier yang lebih realistis. Salah satu standar acuan dalam pemodelan sistem *exciter* dan AVR adalah IEEE Std 421.5-2005 [1], yang menyediakan berbagai jenis model *exciter* seperti DC2A, AC4A, dan ST1A yang masing-masing cocok untuk aplikasi tertentu [5, 22]. Pada penelitian ini, digunakan model dinamis AVR orde tiga yang mencakup representasi lengkap dari blok-blok penyusun sistem, yaitu *amplifier* yang merepresentasikan penguatan sinyal *error*, *exciter* yang menggambarkan hubungan antara tegangan *input* dan arus *exciter*, generator yang menggambarkan respons tegangan terminal

terhadap perubahan arus *exciter*, serta sensor yang merepresentasikan proses pengukuran tegangan terminal.

Model ini dirancang agar mampu menangani ketidakpastian parameter, gangguan eksternal, serta variasi beban yang sering terjadi dalam sistem tenaga nyata. Representasi matematis dari model orde tiga ini biasanya dinyatakan dalam bentuk fungsi alih yang terdiri dari konstanta waktu dan *gain* masing-masing blok, seperti yang dijelaskan dalam persamaan-persamaan berikut:

$$G_{AMP} = \frac{K_A}{\tau_A s + 1} \quad (2.1)$$

$$G_{EXC} = \frac{K_E}{\tau_E s + 1} \quad (2.2)$$

$$G_{GEN} = \frac{K_G}{\tau_G s + 1} \quad (2.3)$$

dimana K_A adalah *gain* masing-masing blok dan τ_A, τ_E, τ_G adalah konstanta waktu yang sesuai dengan karakteristik dinamis sistem.

Lakukan representasi perilaku dinamik dari semua komponen, maka diambil salah satu blok sebagai contoh untuk dimodelkan ke dalam bentuk *transfer function*, yaitu komponen penguat (*amplifier*) dalam sistem AVR. Pendekatan yang digunakan adalah rangkaian RC *low-pass filter*, karena blok ini pada dasarnya berperan sebagai pemroses sinyal dengan karakteristik keterlambatan (*delay*) serta kemampuan meredam perubahan mendadak (*filtering*). Model ini dianggap cukup akurat untuk menangkap karakteristik dasar dari sistem orde satu.

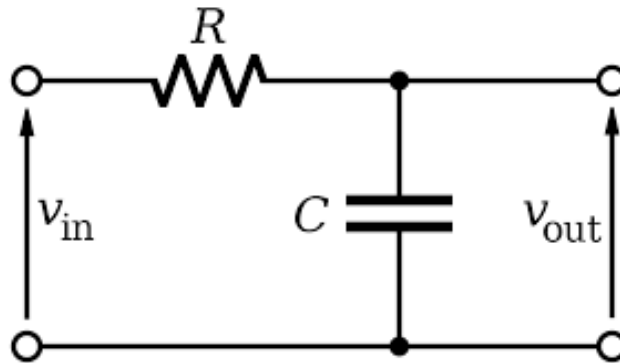
Rangkaian RC tersebut terdiri dari sebuah resistor R dan kapasitor C , disusun secara seri, dengan *input* tegangan $V_{in}(t)$ dan *output* diambil pada ujung kapasitor sebagai $V_{out}(t)$. Meskipun secara fisik komponen seperti *exciter* dan generator dalam sistem AVR tidak secara nyata dibangun dari rangkaian RC, pendekatan ini tetap digunakan untuk menyederhanakan dinamika masing-masing komponen menjadi bentuk orde satu. Dengan menggunakan model matematis seragam berdasarkan sistem orde satu, proses analisis dan perancangan mengontrol menjadi lebih terstruktur dan memungkinkan perbandingan antar blok dilakukan secara adil.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Gambar 2.2 Rangkaian RC *Low-pass filter*

Dari Gambar 2.2 lalu dikaitkan dengan Hukum *Kirchoff* atau *Kirchoff Voltage Law* (KVL) dan menghasilkan persamaan *loop* tegangan yaitu:

$$V_{in}(t) = V_R(t) + V_C(t) \quad (2.4)$$

dengan tahap pertama yaitu pada *loop* pertama dari $V_{in}(t)$ terdapat resistor dikaitkan dengan rumus resistansi dalam bentuk persamaan:

$$V_R(t) = Ri(t) \quad (2.5)$$

berikutnya tahap kedua yaitu pada *loop* kedua dari $V_{out}(t)$ terdapat komponen kapasitor dengan menyamakan variabel $V_{out}(t)$, berikut bentuk *loop* kedua:

$$V_C(t) = V_{out}(t) \quad (2.6)$$

lalu substitusikan ke persamaan KVL dengan menggabungkan persamaan :

$$RC \frac{dV_{out}(t)}{dt} + V_{out}(t) = V_{in}(t) \quad (2.7)$$

didapatkan bentuk persamaan (2.7) setara dengan model matematis sistem dinamik orde satu. Berikutnya ambil transformasi *Laplace* (asumsi kondisi awal nol):

$$RCsV_{out}(s) + V_{out}(s) = V_{in}(s) \quad (2.8)$$

$$V_{out}(s)(RCs + 1) = V_{in}(s) \Rightarrow \frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{RCs + 1}$$

agar persamaan (2.8) bisa digunakan dalam bentuk *transfer function* maka dilakukan penyamaan dengan sistem orde satu yaitu dengan menjadikan

$$\tau = RC \quad (2.9)$$

bentuk persamaan (2.9) sebagai konstanta waktu dari sistem. Selain itu, nilai penguatan awal ditetapkan sebesar $K = 1$ untuk menyamakan kebutuhan setiap komponen dalam sistem AVR serta menjadikan nilai *gain* sebagai *default*, agar lebih mudah dalam proses analisis awal dan normalisasi respon sistem. Model ini dilanjutkan dan diadaptasikan dari pendekatan persamaan orde 1, dimana bentuk umumnya yaitu:

$$\frac{Out}{In} = \frac{K}{1 + \tau} \quad (2.10)$$

atau bentuk dasarnya yaitu:

$$\tau \frac{dy(t)}{dt} + y(t) = Ku(t) \quad (2.11)$$

yang dimana τ adalah konstanta waktu (*time constant*), K adalah penguatan (*gain*), $u(t)$ adalah *input* sistem dan $y(t)$ adalah *output* sistem. Model ini mengasumsikan bahwa respons tiap komponen bersifat linear dan waktu yang kontinu. Hasil analisis dalam domain frekuensi, dilakukan transformasi *Laplace* dari persamaan (2.11) terhadap persamaan differensial:

$$\tau sY(s) + Y(s) = KU(s) \Rightarrow Y(s) = \frac{K}{\tau s + 1}U(s) \quad (2.12)$$

Sehingga dari persamaan (2.12) bisa dibuat *transfer function* umum tiap blok adalah

$$G(s) = \frac{K}{\tau s + 1} \quad (2.13)$$

Berdasarkan struktur umum sistem AVR tanpa sensor, model terdiri dari tiga komponen utama, dengan parameter dari literatur:

Tabel 2.1. Bentuk *Transfer Function* per Komponen

Komponen	Input	Output	Gain (K)	Konstanta Waktu (τ)	Transfer Function
Amplifier	$u(t)$	$V_e(t)$	K_A	τ_A	$\frac{K_A}{\tau_A s + 1}$
Exciter	$V_e(t)$	$V_f(t)$	K_E	τ_E	$\frac{K_E}{\tau_E s + 1}$
Generator	$V_f(t)$	$V_t(t)$	K_G	τ_G	$\frac{K_G}{\tau_G s + 1}$

Pada Tabel 2.1 menunjukkan hubungan *input-output* pada setiap komponen utama AVR beserta fungsi alihnya. *Amplifier* menerima masukan berupa sinyal kesalahan $u(t)$ dan menghasilkan tegangan keluaran $V_e(t)$. Hubungan ini direpresentasikan melalui fungsi alih

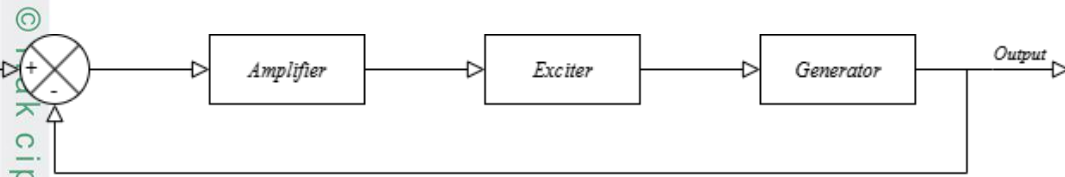
$\frac{K_A}{\tau_A s + 1}$, di mana K_A adalah penguatan dan τ_A adalah konstanta waktu *amplifier*. *Exciter*

menggunakan keluaran *amplifier* $V_e(t)$ sebagai *input* dan menghasilkan tegangan medan $V_f(t)$. Respon dinamisnya digambarkan oleh fungsi alih $\frac{K_E}{\tau_E s + 1}$. Besarnya penguatan K_E

serta konstanta waktu τ_E menentukan seberapa cepat tegangan *exciter* menyesuaikan perubahan pada masukan. Generator menerima masukan berupa tegangan *exciter* $V_f(t)$ dan

menghasilkan tegangan terminal $V_t(t)$. Perilakunya dimodelkan dengan fungsi alih $\frac{K_G}{\tau_G s + 1}$

, di mana K_G adalah penguatan generator dan τ_G merupakan konstanta waktunya. Maka secara rangkaian apabila digabungkan semuanya maka akan didapatkan pada gambar berikut.



Gambar 2.3 Blok Diagram AVR *Close-loop*

Dengan mengalikan masing-masing *transfer function* secara berurutan, diperoleh fungsi transfer sistem AVR (*closed-loop*) yaitu:

$$G(s) = \frac{K_A K_E K_G}{(\tau_A s + 1)(\tau_E s + 1)(\tau_G s + 1)} \quad (2.14)$$

dengan fungsi alih pada persamaan (2.14) menjadi representasi matematis dari hubungan antara *input* referensi $u(t)$ terhadap *output* sistem $V_r(t)$ tanpa adanya pengendali apapun.

Model ini menjadi dasar bagi analisis peran PID terhadap kinerja SMC yang diterapkan pada sistem AVR.

Untuk data yang digunakan dalam sistem AVR berdasarkan penelitian yang sudah pernah dilakukan [23], data tersebut mencakup parameter-parameter yang relevan untuk analisis performa dan optimasi sistem AVR. Data ini meliputi nilai konstanta komponen, konstanta waktu, serta parameter lainnya yang kritis untuk pemodelan dan simulasi sistem AVR. Informasi rinci mengenai data ini tersaji pada Tabel 2.2 berikut:

Tabel 2.2 Data-data sistem AVR [24]

No.	Variabel	Keterangan Variabel	Nilai
1	K_A	Penguatan (<i>gain</i>) <i>Amplifier</i>	10
2	τ_A	Konstanta waktu <i>Amplifier</i>	0,1
3	K_E	Penguatan (<i>gain</i>) <i>Exciter</i>	1
4	τ_E	Konstanta waktu <i>Exciter</i>	0,4
5	K_G	Penguatan (<i>gain</i>) Generator	1
6	τ_G	Konstanta waktu Generator	1.0

Amplifier berfungsi sebagai penguat sinyal $u(t)$ yang diteruskan menjadi tegangan referensi $V_r(t)$. Pada Tabel 2.2 ditunjukkan bahwa *amplifier* memiliki penguatan $K_A = 10$ dengan konstanta waktu $\tau_A = 0,1$ detik. Nilai ini menandakan bahwa *amplifier* mampu merespons perubahan sinyal masukan dengan cepat, sehingga dapat menjaga ketepatan penguatan pada awal proses regulasi. *Exciter* menerima keluaran *amplifier* untuk diubah menjadi tegangan medan $V_f(t)$. Nilai penguatan *exciter* sebesar $K_E = 1$ dengan konstanta waktu $\tau_E = 0,4$ detik menunjukkan bahwa respon *exciter* lebih lambat dibanding *amplifier*, tetapi berperan penting dalam mempertahankan kestabilan fluks magnetik pada rotor [24].

Generator sinkron merupakan komponen akhir yang menghasilkan tegangan terminal $V_t(t)$. Dari Tabel 2.2, generator memiliki penguatan $K_G = 1$ dengan konstanta waktu $\tau_G = 1,0$ detik. Nilai ini menunjukkan bahwa generator adalah komponen dengan respon paling lambat dibandingkan *amplifier* maupun *exciter*. Hal tersebut wajar karena dinamika elektromagnetik generator dipengaruhi oleh kondisi beban dan inersia sistem. Kombinasi ketiga parameter inilah yang membentuk model dinamis AVR dan digunakan sebagai dasar dalam simulasi pengendalian tegangan, sehingga model dapat merepresentasikan kondisi sistem secara lebih realistis [24].

2.2.2 Sistem Exciter pada Generator

Sistem *exciter* merupakan bagian kritis dalam menjaga kestabilan tegangan generator sinkron, terutama saat terjadi perubahan beban atau gangguan eksternal. Sistem ini berfungsi sebagai sumber arus searah yang mengalir ke kumparan medan generator sehingga menghasilkan fluks magnetik yang menentukan besar tegangan *output* generator [1]. Standar yang disebutkan oleh IEEE Std 421.5-2005 memberikan panduan standar untuk berbagai model *exciter* seperti AC1A, DC1A, dan ST1A yang digunakan dalam studi stabilitas sistem tenaga [1]. Dalam operasi normal, pengaturan arus *exciter* dilakukan secara otomatis melalui AVR untuk menjaga tegangan terminal pada nilai *setpoint* yang diinginkan [25]. Penelitian yang dilakukan oleh Máslo et al. (2016) menyederhanakan model IEEE menjadi tiga kategori utama yaitu *exciter* dengan *rectifier* tidak terkontrol, terkontrol, dan statis, untuk memudahkan analisis dalam sistem besar [26]. Pendekatan ini memungkinkan integrasi yang lebih baik antara berbagai jenis *exciter* dalam studi stabilitas sistem tenaga.

Model sistem *exciter* juga dapat direpresentasikan dalam bentuk fungsi alih orde pertama, dengan gain (K) dan konstanta waktu (τ) sebagai parameter utama [23].

Kenaikan arus *exciter* pada kumparan medan generator akan menyebabkan peningkatan fluks magnetik yang dihasilkan, yang pada akhirnya meningkatkan tegangan *output* generator sesuai dengan prinsip dasar hukum *Faraday* tentang induksi elektromagnetik. Hubungan antara tegangan *exciter* (E_f) dan arus *exciter* (I_f) dapat dinyatakan secara matematis sebagai

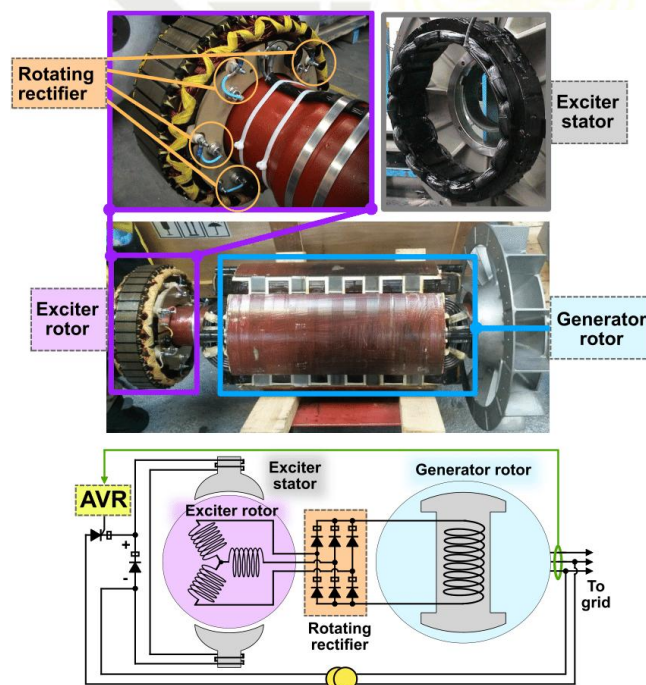
$$E_f = k_f I_f \quad (2.15)$$

di mana k_f adalah konstanta yang bergantung pada desain konstruksi generator. Dengan demikian, semakin besar arus *exciter* yang dialirkan, semakin besar pula tegangan yang dihasilkan oleh generator [27]. Dalam kondisi beban yang tetap, tegangan terminal (V_t) generator berbanding lurus dengan besarnya *exciter*, sehingga pengaturan arus *exciter* secara tepat menjadi kunci dalam menjaga stabilitas tegangan *output* generator. Oleh karena itu, sistem *exciter* harus dirancang sedemikian rupa sehingga mampu memberikan arus *exciter* yang sesuai untuk mempertahankan tegangan terminal pada nilai *setpoint* yang diinginkan meskipun terjadi gangguan atau perubahan kondisi operasi [1].

Perubahan beban pada generator sinkron secara langsung memengaruhi besar tegangan terminal, di mana peningkatan beban umumnya menyebabkan penurunan tegangan akibat peningkatan *drop* tegangan pada impedansi internal generator, sementara penurunan beban dapat menyebabkan kenaikan tegangan yang tidak diinginkan [21]. Untuk mengatasi deviasi tegangan ini, AVR berperan aktif dalam mendeteksi perubahan tegangan melalui sensor tegangan yang menggunakan prinsip pembagi tegangan, kemudian membandingkannya dengan tegangan referensi untuk menghasilkan sinyal *error* [21]. Sinyal *error* tersebut selanjutnya diproses oleh kontroler AVR untuk menghasilkan sinyal kontrol yang dikirim ke sistem *exciter*, sehingga arus *exciter* ke kumparan medan generator dapat disesuaikan secara otomatis. Dengan demikian, tegangan terminal dapat dipertahankan pada nilai *setpoint* yang diinginkan meskipun terjadi perubahan beban secara tiba-tiba [23]. Pendekatan ini memastikan bahwa generator tetap memberikan pasokan tegangan yang stabil dan memenuhi standar kualitas daya yang dibutuhkan dalam sistem tenaga listrik [1].

Sistem *exciter* pada generator sinkron diklasifikasikan ke dalam beberapa jenis utama, yaitu DC *exciter*, AC *exciter*, dan *static exciter*, masing-masing dengan karakteristik

dan mekanisme penyediaan arus *exciter* yang berbeda. DC *exciter* menggunakan generator arus searah (*dynamo*) kecil yang dipasang pada poros turbin untuk menghasilkan arus DC langsung yang dialirkan ke kumparan medan melalui sikat dan cincin geser, meskipun metode ini rentan terhadap perawatan dan keausan komponen mekanis [1]. Sebagai perkembangan lebih lanjut, AC *exciter* menggunakan generator AC kecil (*auxiliary alternator*) yang juga terpasang pada poros utama, di mana tegangan keluarannya diubah menjadi arus DC melalui penyearah (*rectifier*) yang dipasang secara rotatif pada poros rotor, sehingga tidak memerlukan sikat (*brushless excitation*), meningkatkan keandalan dan mengurangi kebutuhan perawatan [1]. Sedangkan *static exciter* merupakan jenis yang paling modern, di mana daya *exciter* diperoleh dari sistem tenaga melalui transformator dan dikonversi menjadi arus DC menggunakan konverter daya statis berbasis *thyristor* atau IGBT, kemudian dialirkan ke rotor melalui sikat, memberikan respons yang cepat dan akurat terhadap perubahan beban [1, 16]. Ketiga jenis *exciter* ini digunakan secara luas tergantung pada kapasitas generator dan kebutuhan sistem tenaga, dengan *static exciter* dan *brushless AC exciter* menjadi pilihan dominan pada instalasi modern karena efisiensi dan keandalannya [1].



Gambar 2.4 Komponen jenis *brushless excitation system*

Sebagai contoh pada Gambar 2.4 yaitu sistem *exciter* yang umum digunakan dalam generator sinkron modern, AC *exciter* dengan *rotating rectifier* merupakan pilihan yang efektif karena kemampuannya menghasilkan arus *exciter* DC tanpa memerlukan sikat atau *slip ring*. Dalam konfigurasi ini, sinyal kontrol dari AVR dikirim ke kumparan stator *exciter* untuk mengatur besar tegangan AC yang dihasilkan oleh *exciter* [1]. Tegangan AC ini kemudian dialirkan ke rotor *exciter* yang berputar bersama poros utama, dan melalui rangkaian penyearah (*rectifier*) yang terpasang secara rotatif pada rotor *exciter*, arus AC diubah menjadi arus searah (DC) [26]. Arus DC hasil penyearahan ini langsung dialirkan ke kumparan medan generator rotor, sehingga menghasilkan fluks magnetik yang diperlukan untuk membangkitkan tegangan AC di stator generator. Karena penyearah berada pada rotor, tidak diperlukan penghubungan listrik mekanis seperti sikat, yang mengurangi keausan dan perawatan serta meningkatkan keandalan sistem [1]. Konfigurasi ini sering disebut sebagai *brushless excitation system*, dan modelnya secara formal diwakili oleh tipe AC5A, yang merupakan representasi standar untuk sistem *exciter* dengan penyearah [1].

2.2.3 Sliding Mode Control (SMC)

SMC adalah metode kontrol nonlinier yang termasuk dalam kategori *Variable Structure Control* (VSC), di mana sinyal kontrol diskrit digunakan untuk memaksa sistem bergerak menuju dan tetap berada pada suatu permukaan tertentu (*sliding surface*) di ruang keadaan. Dalam perancangan SMC, *error* pelacakan didefinisikan sebagai selisih antara sinyal referensi dan keluaran sistem, yang dasar utama dalam pembentukan permukaan *sliding* dan hukum kendali SMC [9].

$$e(t) = r(t) - y(t) \quad (2.16)$$

Dalam SMC, sistem dinamis dirancang agar mengikuti *sliding mode*, yaitu kondisi di mana variabel-variabel sistem berada pada permukaan tertentu yang ditentukan oleh fungsi *sliding* (S). Fungsi ini biasanya dirancang sedemikian rupa sehingga memberikan stabilitas global terhadap gangguan eksternal dan ketidakpastian model sistem [28]. Salah satu karakteristik utama SMC adalah kemampuan kontroler untuk menjamin stabilitas sistem meskipun adanya ketidakpastian parameter atau gangguan, karena kontroler bekerja dengan cara memaksakan sistem untuk berada pada permukaan *sliding* yang telah ditetapkan, sehingga respons sistem menjadi independen terhadap gangguan tersebut [29].

Tujuan utama penggunaan SMC adalah untuk mencapai *robustness* tinggi terhadap ketidakpastian model dan gangguan eksternal. Metode ini memiliki beberapa keunggulan utama, seperti insensitivitas terhadap gangguan internal dan eksternal, akurasi akhir yang tinggi, serta konvergensi waktu terbatas dari variabel *sliding* ke nol [30]. Namun, salah satu tantangan utama dalam implementasi SMC adalah fenomena *chattering*, yaitu getaran yang dihasilkan oleh *switching control law* yang sering kali terjadi di sekitar permukaan *sliding*. Untuk mengatasi hal ini, beberapa pendekatan modern seperti *super-twisting* SMC atau *learning* SMC telah dikembangkan untuk mereduksi efek *chattering* tanpa mengorbankan performa kontrol pada sistem [29]. Oleh karena itu, SMC menjadi pilihan strategis dalam aplikasi kontrol nonlinier, terutama dalam sistem AVR yang cenderung bersifat kompleks dan tidak pasti.

SMC menggunakan konsep *sliding surface* sebagai hukum target dinamika sistem, di mana pengendali dirancang untuk memaksa lintasan keadaan sistem menuju dan bertahan pada permukaan

$$s(x) = 0 \quad (2.17)$$

yang menentukan dinamika sistem saat berada dalam *sliding mode*. *Sliding surface* dirancang sedemikian rupa sehingga ketika sistem mencapai permukaan ini, dinamika sistem menjadi stabil, berdimensi lebih rendah, dan tidak terpengaruh oleh gangguan atau ketidakpastian parameter [31]. Salah satu formulasi umum dari *sliding surface* adalah dalam bentuk linear:

$$s(x) = Cx \quad (2.18)$$

di mana C adalah matriks desain yang dipilih untuk memastikan stabilitas dan performa respons yang diinginkan. Salah satu pendekatan umum dalam desain *sliding surface* adalah penggunaan metode *eigenvalue placement* atau *linear quadratic minimization*, yang memungkinkan perancang sistem untuk menempatkan *pole* atau meminimalkan indeks performa kuadratik sehingga dinamika sistem selama *sliding mode* menjadi stabil dan terkontrol [28].

Setelah *sliding surface* dirancang, langkah berikutnya adalah menentukan komponen kontrol ekuivalen. Komponen ini diperoleh dengan mengasumsikan bahwa

sistem telah berada pada kondisi *sliding mode*, sehingga turunan *sliding surface* memenuhi kondisi berikut

$$\dot{s}(t) = 0 \quad (2.19)$$

yang menyatakan bahwa sistem telah memasuki *sliding mode*, di mana dinamika sistem menjadi tidak sensitif terhadap gangguan dan ketidakpastian parameter tertentu [28]. Dengan mensubstitusikan dinamika sistem ke dalam persamaan $\dot{s}(t)$ dan menyelesaikannya terhadap sinyal kontrol, diperoleh bentuk kontrol ekuivalen u_{eq} . Kontrol ekuivalen merepresentasikan aksi kontrol ideal yang diperlukan untuk mempertahankan lintasan sistem tetap berada pada permukaan *sliding* tanpa adanya efek *switching*. Oleh karena itu, u_{eq} bersifat kontinu dan menggambarkan dinamika nominal sistem AVR.

Pada prinsipnya, pergerakan sistem dalam SMC dapat dibagi menjadi dua tahap utama, yaitu *reaching process* dan *sliding process*. Tahap pertama adalah *reaching process*, di mana lintasan keadaan (*trajectory*) sistem dipaksa bergerak dari kondisi awal menuju permukaan luncur (*sliding surface*). Setelah lintasan sistem mencapai permukaan tersebut, maka dimulailah tahap kedua, yaitu *sliding process*, di mana lintasan akan bergerak sepanjang permukaan luncur menuju titik keseimbangan di sekitar titik *origin* (0,0).

Untuk memastikan sistem dapat mencapai permukaan luncur dalam waktu tertentu, digunakan hukum *reaching law*. Hukum *reaching law* digunakan untuk menjamin lintasan sistem mencapai permukaan *sliding* dalam waktu terbatas (*finite time*) pada fase *reaching* [32]. Salah satu bentuk hukum *fixed-time convergent sliding mode reaching law* dapat dituliskan sebagai berikut:

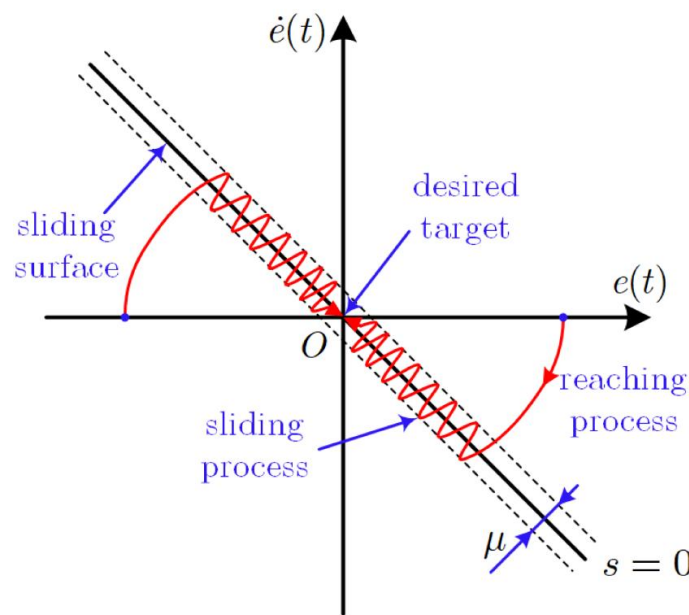
$$\dot{s} = -k_2 \left[\lambda_2 \operatorname{sgn}^{p_2}(s) + \operatorname{sgn}^{q_2}(s) \right] - \mu \operatorname{sgn}(s) \quad (2.20)$$

dengan $k_2 > 0$, $\lambda_2 > 0$, $0 < p_2 < 1$, $q_2 > 1$ dan $\mu > 0$ adalah *switching gain* dari pengendali. Parameter ini berfungsi mengatur kecepatan sistem menuju permukaan luncur, sekaligus mempengaruhi kestabilan sistem terhadap gangguan.

Jika gangguan total sistem tidak diketahui, maka nilai μ sulit ditentukan secara tepat. Nilai μ yang terlalu besar dapat menimbulkan *chattering* (osilasi frekuensi tinggi di sekitar permukaan luncur), sementara nilai μ yang terlalu kecil menyebabkan sistem

kehilangan sifat *robustness*. Oleh karena itu, diperlukan strategi kompensasi untuk menjaga kinerja sistem pada kondisi gangguan yang tidak pasti.

Gambar 2.5 berikut memperlihatkan ilustrasi fase *reaching* dan *sliding process* dalam SMC. Pada bagian awal lintasan, sistem dipaksa untuk bergerak menuju permukaan lurur (ditunjukkan dengan garis merah menuju $s=0$), kemudian meluncur sepanjang permukaan menuju titik kesetimbangan (*desired target*) [33].



Gambar 2.5 Ilustrasi fase *reaching* dan *sliding process*

Setelah *sliding surface* dirancang, langkah selanjutnya adalah memastikan bahwa sistem dapat mencapai permukaan tersebut dalam waktu berhingga. Proses ini dikenal sebagai fase *reaching*, di mana hukum kontrol digunakan untuk memaksa sistem menuju $s(x) = 0$. Salah satu bentuk hukum *reaching* yang umum digunakan adalah:

$$\dot{s} = -k \cdot \text{sign}(s) \quad (2.21)$$

dimana $k > 0$ adalah parameter kontroler yang memastikan kecepatan *reaching* dan ketahanan terhadap gangguan [32]. Fungsi *sign* menyebabkan *switching control* yang membuat sistem bergerak menuju permukaan *sliding* dari kondisi awal manapun. Meskipun efektif dalam memastikan kedatangan sistem ke permukaan *sliding* dalam waktu tertentu, pendekatan ini rentan menyebabkan fenomena *chattering* akibat *switching* yang cepat.

SMC menggunakan hukum kontrol yang terdiri dari dua komponen utama, yaitu komponen ekuivalen (u_{eq}) dan komponen switching (u_{sw}), yang masing-masing berfungsi mempertahankan sistem pada permukaan *sliding* dan memaksa sistem menuju permukaan *sliding* dari kondisi awal [9, 34]. Secara matematis dinyatakan sebagai:

$$u = u_{eq} + u_{sw} \quad (2.22)$$

Komponen ekuivalen (u_{eq}) bertugas untuk menjaga sistem tetap berada di permukaan *sliding* setelah mencapai kondisi *sliding mode*, sedangkan komponen switching (u_{sw}) memaksa sistem untuk mencapai permukaan *sliding* dari kondisi awal manapun dan mempertahankan sistem di sana meskipun terdapat gangguan atau ketidakpastian. Dalam banyak kasus, komponen *switching* dirancang dalam bentuk:

$$u_{sw} = -k \cdot \text{sign}(s(t)) \quad (2.23)$$

di mana $k > 0$ adalah parameter penguatan dan s adalah fungsi *sliding*. Komponen switching ini sangat penting dalam menjamin *robustness* terhadap gangguan yang terklasifikasi sebagai *matched disturbances*, yaitu gangguan yang berada dalam kanal *input* kontrol. Dengan adanya komponen *switching*, sistem tetap mampu mempertahankan dinamika *sliding* meskipun ada ketidakpastian atau gangguan eksternal, karena sifat *discontinuous* dari kontrol ini memaksa sistem untuk kembali ke permukaan *sliding* secara cepat [34].

Untuk menjamin kestabilan sistem pada SMC, digunakan pendekatan kestabilan Lyapunov yang umum diterapkan dalam teori *sliding mode control* [28, 34]. Fungsi Lyapunov kandidat didefinisikan sebagai

$$V = \frac{1}{2} s^2 \quad (2.24)$$

yang bersifat positif definit. Turunan fungsi Lyapunov terhadap waktu diberikan oleh

$$\dot{V} = s\dot{s} \quad (2.25)$$

Dengan menerapkan hukum *reaching law* yang dirancang sehingga $\dot{s}$ berlawanan arah dengan s , maka diperoleh $\dot{V} < 0$ untuk semua $s \neq 0$. Kondisi ini menunjukkan bahwa

lintasan sistem akan selalu bergerak menuju permukaan *sliding* dan bertahan di sekitarnya, sehingga kestabilan sistem AVR secara global dapat dijamin meskipun terdapat gangguan dan ketidakpastian parameter [34].

SMC memiliki beberapa keunggulan yang menjadikannya salah satu metode kontrol nonlinier paling populer, terutama dalam aplikasi sistem dengan ketidakpastian parameter dan gangguan eksternal. Salah satu keunggulan utamanya adalah kemampuannya untuk menjamin *robustness* terhadap gangguan dan ketidakpastian model, selama sistem berada dalam kondisi *sliding mode*. Selain itu, SMC juga menawarkan konvergensi *finite-time*, artinya sistem dijamin mencapai permukaan *sliding* dalam waktu berhingga, tidak seperti kontroler lain yang membutuhkan waktu tak berhingga untuk konvergensi asimtotik [28]. Namun, salah satu kelemahan utama dari SMC adalah fenomena *chattering*, yaitu osilasi frekuensi tinggi pada sinyal kontrol yang terjadi di sekitar permukaan *sliding*. Fenomena ini disebabkan oleh dua faktor utama yaitu idealisasi model yang mengabaikan dinamika cepat dari aktuator atau sensor, dan implementasi diskrit pada kontroler digital dengan frekuensi *sampling* terbatas, yang menyebabkan frekuensi *switching* tidak dapat melebihi frekuensi *sampling* [35]. *Chattering* dapat mengakibatkan kerusakan mekanis, keausan komponen, dan penurunan akurasi kontrol, sehingga menjadi penghambat utama dalam implementasi nyata kontroler SMC [36].

Solusi dari fenomena *chattering* yang menjadi kendala utama dalam implementasi SMC, berbagai teknik pengurangan telah dikembangkan, di antaranya penggunaan *boundary layer*, *higher-order SMC*, dan pendekatan berbasis jaringan saraf. Pendekatan-pendekatan tersebut disampaikan sebagai tinjauan literatur dan tidak diterapkan dalam penelitian ini. Salah satu metode paling umum adalah *boundary layer method*, di mana fungsi *signum* *discontinue* digantikan dengan fungsi kontinu seperti *saturation* atau *tanh* di sekitar permukaan *sliding*, sehingga menghasilkan kontrol yang lebih halus dan mengurangi osilasi frekuensi tinggi [17]. Pendekatan ini efektif karena menghindari *switching* tak hingga dengan membatasi perubahan sinyal kontrol dalam lapisan tertentu di sekitar $s = 0$. Sebagai alternatif yang lebih canggih, *higher-order sliding mode control* (HOSMC), khususnya *super-twisting algorithm* (STA), dirancang untuk menerapkan sinyal *switching* pada turunan lebih tinggi dari variabel *sliding*, sehingga menghasilkan sinyal kontrol yang kontinu secara keseluruhan dan secara signifikan mengurangi *chattering* tanpa mengorbankan *robustness* [28]. Dalam konteks ini, penelitian yang berjudul *Super-twisting sliding mode control with defined boundary layer for chattering reduction of permanent magnet linear synchronous*

motor mengusulkan *super-twisting sliding mode control with defined boundary layer* (δ STA), yang mengganti fungsi *signum* dengan dua fungsi *saturation* berbeda, sehingga mampu mengurangi amplitudo *chattering* secara lebih efektif sambil menjaga kestabilan sistem [37]. Selain itu, pendekatan seperti *neural network-based SMC* atau *Recurrent Neural Network (RNN)* juga mulai dikembangkan untuk memodelkan dan menghaluskan sinyal *switching* secara *real-time*, memungkinkan sistem beradaptasi terhadap dinamika yang tidak terduga dan mengurangi efek *chattering* secara cerdas [31].

2.2.4 Kontrol PID (*Proportional-Integral-Derrivative*)

Kontroler PID merupakan salah satu metode kontrol *loop* umpan balik yang paling umum dan kuat digunakan dalam industri untuk menjaga variabel proses agar tetap pada nilai *setpoint* yang diinginkan, meskipun menghadapi gangguan atau perubahan dinamika sistem. Struktur kontroler ini banyak diadopsi karena sifatnya yang sederhana, mudah diimplementasikan, dan mampu memberikan performa yang andal pada berbagai jenis proses, mulai dari sistem tenaga hingga proses kimia dan manufaktur [38]. Keunggulan utama kontroler PID terletak pada kombinasi tiga komponen yang saling melengkapi: komponen proporsional yang merespons kesalahan secara langsung, komponen integral yang menghilangkan kesalahan tetap (*steady-state error*), serta komponen derivatif yang memperkirakan tren kesalahan dan meningkatkan stabilitas sistem [21]. Berdasarkan literatur, kontroler PID telah terbukti efektif dalam aplikasi AVR untuk mengatur tegangan *exciter* generator sinkron, bahkan dalam kondisi perubahan beban yang mendadak [17]. Oleh karena itu, kontroler PID tetap menjadi pilihan utama dalam sistem kontrol industri modern, baik dalam bentuk analog maupun digital, karena fleksibilitas dan keandalannya dalam mencapai kinerja optimal [38].

Komponen proporsional (P) dalam kontroler PID menghasilkan sinyal kontrol yang proporsional terhadap besarnya kesalahan (*error*) pada waktu saat itu, yang dinyatakan sebagai

$$K_p e(t) \quad (2.26)$$

di mana K_p adalah penguatan proporsional dan $e(t)$ adalah selisih antara *setpoint* dan *output* sistem. Komponen ini berperan penting dalam mempercepat respons dinamis sistem karena memberikan aksi kontrol segera saat terdeteksi adanya deviasi, sehingga mengurangi

waktu naik (*rise time*) [39]. Namun, penguatan K_p yang terlalu besar dapat menyebabkan sistem menjadi terlalu responsif, yang berpotensi menimbulkan *overshoot* yang signifikan dan osilasi sebelum mencapai keadaan stabil, bahkan dapat mengarah pada ketidakstabilan jika tidak dikombinasikan dengan komponen integral dan derivatif secara tepat [40]. Oleh karena itu, pemilihan nilai K_p yang optimal sangat krusial untuk mencapai keseimbangan antara kecepatan respons dan stabilitas sistem, terutama dalam aplikasi seperti AVR yang memerlukan presisi tinggi dalam pengaturan tegangan terminal [21].

Komponen integral (I) dalam kontrol PID menghasilkan sinyal kontrol yang proporsional terhadap akumulasi kesalahan (*error*) dari waktu ke waktu, dinyatakan secara matematis sebagai

$$K_i \int e(t) dt \quad (2.27)$$

di mana K_i adalah konstanta integral. Komponen ini berperan penting dalam mengeliminasi *steady-state error*, yaitu kesalahan yang masih tersisa setelah sistem mencapai kondisi mantap, yang tidak dapat diatasi oleh komponen proporsional saja [40]. Dengan mengintegrasikan *error* sepanjang waktu, kontroler integral terus menyesuaikan *output*-nya hingga *error* benar-benar bernilai nol, sehingga sangat efektif dalam menjaga akurasi jangka panjang sistem [21]. Namun, perlakuan integral yang terlalu agresif (dengan nilai K_i yang terlalu besar) dapat menyebabkan akumulasi *error* yang berlebihan, yang berpotensi menghasilkan *overshoot* dan osilasi sebelum sistem stabil, bahkan dapat menurunkan stabilitas sistem secara keseluruhan [39]. Oleh karena itu, penalaan parameter integral harus dilakukan secara hati-hati untuk menyeimbangkan antara kemampuan menghilangkan *steady-state error* dan mencegah respons yang terlalu lambat atau berlebihan [1].

Komponen derivatif (D) dalam kontrol PID menghasilkan sinyal kontrol yang proporsional terhadap laju perubahan kesalahan (*error*), dinyatakan secara matematis sebagai

$$K_d \frac{de(t)}{dt} \quad (2.28)$$

di mana K_d adalah konstanta derivatif. Komponen ini berfungsi untuk memprediksi tren perubahan *error* di masa depan berdasarkan kemiringan *error* saat ini, sehingga mampu

memberikan aksi korektif lebih awal dan meredam respons berlebihan seperti *overshoot* serta osilasi, yang pada gilirannya meningkatkan kestabilan sistem dan mempercepat waktu *settling* [40]. Dengan menambahkan efek peredaman, komponen derivatif sangat efektif dalam mengurangi kecenderungan sistem untuk berosilasi sekitar *setpoint*, terutama pada sistem dengan dinamika lambat atau inersia tinggi [17]. Namun, kepekaan terhadap perubahan cepat ini membuat komponen D sangat sensitif terhadap *noise* dan gangguan pada sinyal pengukuran, karena *noise* biasanya memiliki perubahan frekuensi tinggi yang menyebabkan turunan menjadi sangat besar dan menghasilkan sinyal kontrol yang tidak diinginkan [39]. Oleh karena itu, ketika diimplementasikan sering digunakan filter orde rendah atau kontrol PID dengan derivatif pada variabel *output* (*derivative-on-measurement*) untuk mengurangi efek *noise* tanpa mengorbankan kinerja stabilisasi [38].

Kontroler PID menghasilkan sinyal kontrol berdasarkan kombinasi tiga komponen: proporsional, integral, dan derivatif, yang secara matematis dinyatakan dalam bentuk waktu kontinu sebagai:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (2.29)$$

di mana $u(t)$ adalah sinyal keluaran kontroler, $e(t)$ adalah kesalahan antara *setpoint* dan *output* sistem, serta K_p , K_i , dan K_d masing-masing merupakan penguatan proporsional, integral, dan derivatif [40]. Bentuk ini dikenal sebagai struktur paralel (ideal), yang merupakan representasi standar dalam analisis dan desain kontroler, di mana ketiga komponen bekerja secara independen dan dijumlahkan secara langsung [39]. Dalam banyak literatur dan aplikasi industri, parameter integral dan derivatif sering dinyatakan dalam bentuk konstanta waktu, yaitu waktu integral (T_i) dan waktu derivatif (T_d), yang terkait

dengan parameter penguatan melalui hubungan $K_i = \frac{K_p}{T_i}$ dan $K_d = K_p T_d$. Substitusi hubungan ini ke dalam bentuk paralel menghasilkan bentuk standar kontroler PID yang sering digunakan dalam implementasi praktis [41]. Representasi dalam domain frekuensi (domain s) dari kontroler PID dengan struktur paralel ini dinyatakan sebagai:

$$C(s) = K_p + sK_i + K_d s \quad (2.30)$$

yang memungkinkan analisis stabilitas dan penalaan parameter dilakukan secara sistematis dalam domain *Laplace* [21].

2.2.5 Respon Transien Sistem Orde II

Respons transien merupakan bagian dari dinamika sistem yang terjadi sebelum mencapai kondisi tunak (*steady state*), yaitu periode awal setelah sistem menerima gangguan atau perubahan *setpoint*, di mana *output* sistem berubah secara dinamis menuju nilai akhirnya [42]. Karakteristik respons transien, seperti waktu naik (*rise time*), waktu puncak (*peak time*), *overshoot*, dan waktu *settling*, digunakan sebagai indikator utama dalam menilai kualitas performansi sistem kendali, termasuk aspek kecepatan, stabilitas, dan akurasi respons [43]. Sistem dengan respons transien yang baik ditandai oleh waktu *settling* yang singkat, *overshoot* minimal, dan tidak adanya osilasi berlebihan, sehingga dapat memenuhi standar performansi yang diharapkan [16].

Respons transien suatu sistem dikarakterisasi oleh sejumlah parameter utama yang menjadi tolak ukur kinerja dalam menilai kecepatan, stabilitas, dan akurasi sistem kontrol. Waktu naik (*rise time, t_r*) didefinisikan sebagai waktu yang dibutuhkan respons sistem untuk meningkat dari 10% hingga 90% dari nilai akhirnya, sedangkan waktu tunda (*delay time, t_d*) adalah waktu yang diperlukan sistem untuk mencapai 50% dari nilai akhir [16]. Waktu tunak (*settling time, t_s*) merupakan durasi hingga respons sistem masuk dan bertahan dalam *band* $\pm 2\%$ terhadap nilai akhir, yang menunjukkan seberapa cepat sistem mencapai kondisi stabil [17]. *Overshoot* (M_p) dihitung sebagai persentase lonjakan maksimum *output* yang melebihi nilai *setpoint* terhadap nilai akhir, dan menjadi indikator penting dari kestabilan sistem [18]. Selain itu, *Root Mean Square Error* (RMSE) digunakan sebagai ukuran statistik untuk mengevaluasi rata-rata deviasi antara respons sistem dan sinyal referensi selama periode waktu tertentu, sehingga memberikan gambaran mengenai akurasi keseluruhan kontrol [23]. Parameter-parameter tersebut secara bersama-sama menjadi dasar evaluasi kinerja dalam sistem kendali modern.

Respons transien sistem orde kedua dapat dianalisis secara kuantitatif melalui dua parameter utama, yaitu rasio redaman (*damping ratio, ζ*) dan frekuensi alami (*natural frequency, ω_n*), yang menentukan perilaku dinamis sistem saat merespons perubahan *setpoint* atau gangguan. Rasio redaman menggambarkan seberapa cepat osilasi sistem diredam, sedangkan frekuensi alami menunjukkan kecepatan respons sistem dalam kondisi tidak teredam. Keduanya berperan langsung dalam menentukan karakteristik respons

transien seperti waktu tunak (t_s) dan *overshoot* maksimum (M_p), yang merupakan indikator utama performa kontrol. Secara matematis, waktu tunak dapat diestimasi dengan rumus:

$$t_s \approx \frac{4}{\zeta \omega_n} \quad (2.31)$$

sedangkan *overshoot* maksimum dihitung sebagai:

$$M_p = e^{-\frac{\pi \zeta}{\sqrt{1-\zeta^2}}} \times 100\% \quad (2.32)$$

Selain parameter rasio redaman dan frekuensi alami pada sistem orde dua, karakteristik transien juga dapat dianalisis melalui konstanta waktu (τ) pada sistem orde satu. Konstanta waktu ini sering digunakan untuk menyederhanakan pemahaman respon dinamis suatu sistem, karena memberikan gambaran langsung mengenai kecepatan respon terhadap perubahan *input*. Hubungan antara nilai τ dengan karakteristik utama respon sistem ditunjukkan pada Tabel 2.3.

Tabel 2.3 hubungan konstanta waktu (τ) dengan karakteristik respon

Konstanta Waktu (τ)	Karakteristik Respon
Besar (τ tinggi)	Respon lambat, t_s panjang, <i>overshoot</i> kecil
Kecil (τ rendah)	Respon cepat, t_s singkat, <i>overshoot</i> cenderung besar
Sedang	Respon moderat, kompromi antara kecepatan dan kestabilan

Tabel 2.3 menunjukkan hubungan antara konstanta waktu (τ) dari sistem orde satu dengan karakteristik respons transien, seperti waktu naik, waktu tunak, dan *overshoot*. Konstanta waktu berperan penting dalam menentukan seberapa cepat sistem memberikan respons terhadap perubahan *input*. Nilai τ yang besar menghasilkan respons yang lambat, ditandai dengan waktu tunak (t_s) yang panjang serta *overshoot* yang relatif kecil karena sistem bergerak lebih hati-hati menuju kondisi tunak [16]. Sebaliknya, nilai τ yang kecil mempercepat dinamika sistem, sehingga waktu tunak menjadi singkat, namun cenderung menimbulkan *overshoot* lebih tinggi akibat pergerakan sistem yang lebih agresif [21].



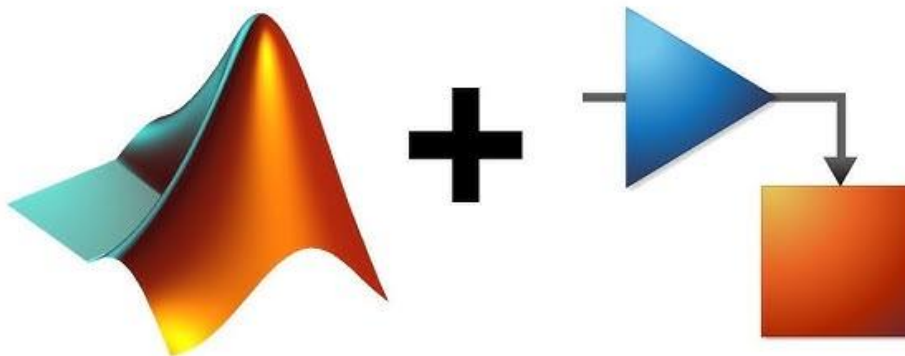
Sementara itu, nilai τ pada rentang sedang memberikan kompromi antara kecepatan respon dan kestabilan, sehingga sering dijadikan acuan awal dalam perancangan kontrol. Dengan demikian, pemilihan konstanta waktu yang tepat menjadi faktor kunci agar sistem dapat mencapai performa optimal sesuai dengan kebutuhan aplikasinya.

2.2.6 Software MATLAB/Simulink



Gambar 2.6 Logo Software Matlab

MATLAB (*Matrix Laboratory*) merupakan perangkat lunak berbasis komputasi numerik yang dikembangkan oleh “*The MathWorks*” dan telah menjadi standar dalam lingkungan akademik maupun industri teknik. *Software* ini dirancang khusus untuk kebutuhan komputasi teknis, dengan fungsi utama meliputi pemrograman, perhitungan matematis, analisis data, visualisasi hasil, serta pemodelan dan simulasi sistem dinamis [44]. Keunggulan MATLAB terletak pada kemampuannya dalam melakukan operasi matriks secara efisien, yang menjadi dasar dari banyak metode perhitungan dalam teknik elektro dan sistem kontrol. Selain itu, MATLAB juga menyediakan berbagai *toolbox* khusus, seperti *Control System Toolbox*, yang memfasilitasi analisis respon waktu maupun frekuensi, serta memudahkan perancangan kontroler untuk berbagai jenis sistem [16].



Gambar 2.7 Logo Matlab dengan Simulink

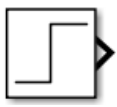
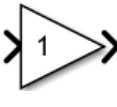
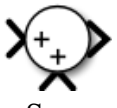
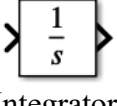
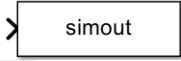
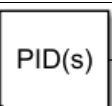
Simulink merupakan bagian integral dari MATLAB yang dikembangkan oleh *The MathWorks* sebagai lingkungan grafis untuk pemodelan, simulasi, dan analisis sistem dinamis. Lingkungan ini menggunakan pendekatan berbasis diagram blok (*block diagram environment*) yang memungkinkan pengguna menyusun model dengan metode *drag-and-drop* dari pustaka blok yang tersedia, sehingga mempermudah representasi sistem fisik maupun elektrik tanpa perlu menulis kode pemrograman secara intensif [45]. Keunggulan utama Simulink adalah kemampuannya dalam memodelkan dan mensimulasikan berbagai jenis sistem, baik linier maupun nonlinier, serta integrasinya yang erat dengan MATLAB yang mendukung analisis respons transien, tuning parameter, dan visualisasi hasil secara langsung [8]. Karena fleksibilitas dan kemudahannya, Simulink banyak digunakan dalam penelitian dan pengembangan di bidang teknik elektro, khususnya untuk desain serta evaluasi sistem kontrol [23].

MATLAB/Simulink menawarkan keunggulan signifikan dalam penelitian sistem kontrol karena didukung oleh berbagai *toolbox* yang mempermudah analisis dan desain sistem dinamis, seperti *Control System Toolbox*, *Simscape*, dan *Optimization Toolbox*. *Toolbox-toolbox* ini memungkinkan peneliti untuk melakukan analisis respons waktu (baik transien maupun keadaan mantap), analisis respons frekuensi, serta perancangan dan penalaan kontroler secara efisien dan akurat [45]. Dengan *Control System Toolbox*, pengguna dapat memodelkan sistem dalam bentuk fungsi alih atau ruang keadaan, melakukan analisis stabilitas, dan menghitung parameter kinerja seperti *overshoot*, waktu naik, dan waktu tunak secara otomatis [8]. Selain itu, Simulink memungkinkan simulasi sistem nonlinier dan multibodi melalui lingkungan diagram blok, sementara *Optimization Toolbox* membantu dalam penalaan parameter kontroler secara otomatis untuk mencapai performa optimal [38]. Karena kemampuannya dalam memvalidasi teori melalui simulasi numerik sebelum implementasi nyata, MATLAB/Simulink telah menjadi standar dalam penelitian akademik maupun industri, terutama dalam bidang teknik elektro dan rekayasa sistem tenaga [17].

Dalam simulasi menggunakan MATLAB Simulink, berbagai blok diagram digunakan untuk membangun model dan mengamati respon sistem terhadap masukan yang diberikan. Setiap blok memiliki fungsi spesifik yang berkontribusi dalam pemrosesan sinyal, perhitungan integral, serta implementasi kendali. Tabel 2.4 berikut menyajikan daftar blok Simulink yang digunakan beserta keterangannya:



Tabel 2.4 Blok diagram Simulink

Blok Simulink	Keterangan
 Step	Digunakan untuk membangkitkan sinyal fungsi step sebagai masukan/ <i>input</i>
 Gain	Digunakan untuk melipatgandakan /memperkuat penguatan dengan nilai yang konstan dan dapat diubah – ubah. Nilai <i>gain</i> dapat berupa bilangan skalar, vektor, maupun matrik.
 Sum	Digunakan untuk menjumlahkan atau mengurangi <i>input/output</i> .
 Integrator	Digunakan untuk menyatukan sinyal, dapat pula untuk menyatakan integral.
 To Workspace	Digunakan untuk menyimpan hasil simulasi ataupun analisis sistem kedalam <i>workspace</i> matlab yang berupa variabel, data atau respon sistem yang akan digunakan untuk analisis lebih lanjut.
 Scope	Digunakan untuk menampilkan keluaran berbentuk fisik.
 Mux	Mux digunakan untuk menggabungkan beberapa sinyal <i>input</i> menjadi satu sinyal <i>output</i>
 PID(s)	Digunakan untuk mengimplementasikan kontroler proporsional-integral-derivatif (PID).

1. Unsur-unsur yang terdapat dalam blok diagram simulink adalah sebagai berikut:

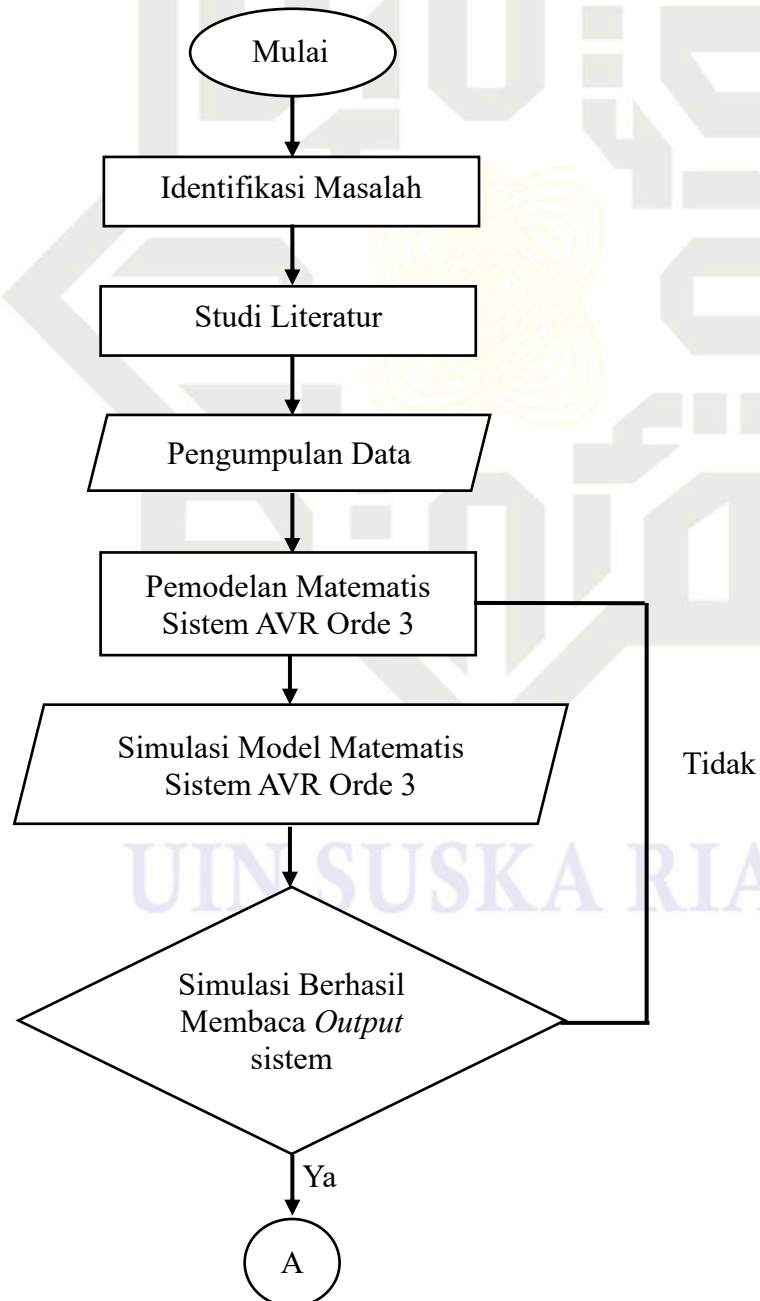
- Penggunaan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - Penggunaan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

BAB III

METODOLOGI PENELITIAN

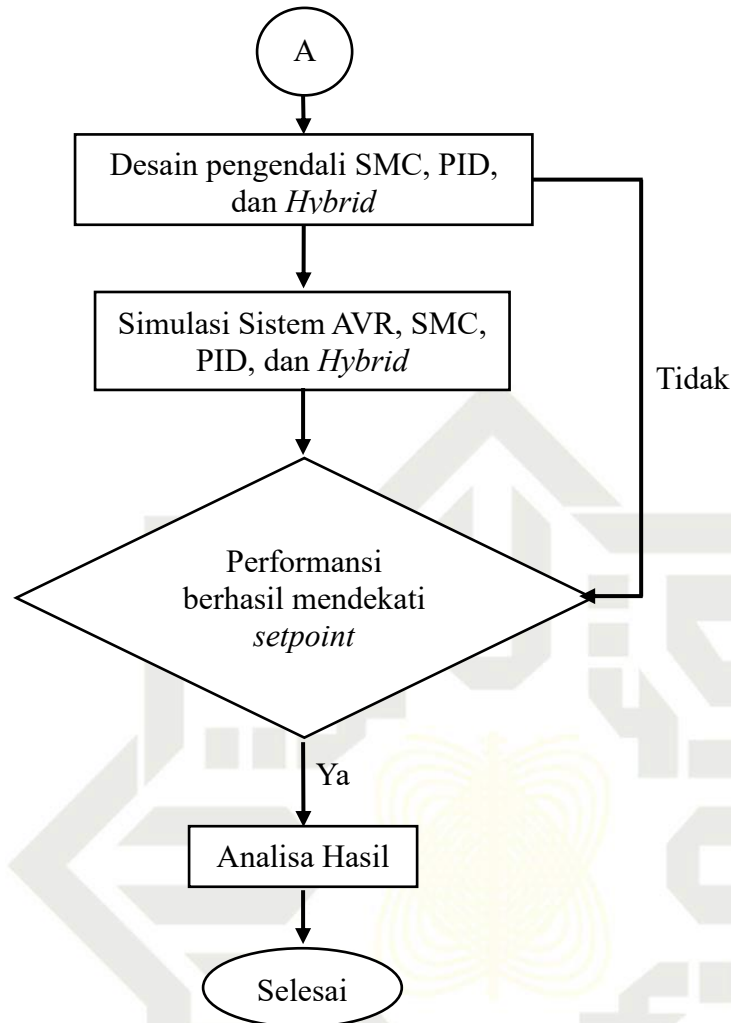
Diagram Alur Penelitian

Dalam penelitian ini, digunakan diagram alur untuk menggambarkan langkah-langkah penelitian secara sistematis mulai dari identifikasi masalah, studi literatur, hingga perancangan kontroler dan simulasi. Diagram alur ini berfungsi untuk memperjelas urutan dan keterkaitan antar langkah dalam proses penelitian. Diagram alur penelitian ditunjukkan pada Gambar 3.1 berikut ini.



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Gambar 3.1 Diagram Alur Penelitian

3.2 Tahapan Penelitian

Berdasarkan diagram alur penelitian pada Gambar 3.1, ada beberapa tahapan yang dilakukan dalam penelitian sebagai berikut:

1. Identifikasi Masalah

Tahap pertama dalam penelitian ini adalah melakukan identifikasi masalah yang berkaitan dengan ketidakstabilan tegangan terminal pada generator sinkron akibat perubahan beban. Ketika beban mengalami fluktuasi, tegangan keluaran cenderung menyimpang dari nilai referensi sehingga dapat menurunkan kualitas daya listrik. Permasalahan ini menjadi penting karena AVR konvensional seringkali belum mampu memberikan regulasi tegangan yang cepat dan stabil, terutama ketika menghadapi gangguan eksternal atau kondisi

nonlinier. Oleh karena itu, diperlukan rancangan kontrol yang lebih adaptif untuk meningkatkan kestabilan tegangan terminal generator.

Studi Literatur

Tahap selanjutnya adalah melakukan studi literatur guna memperoleh pemahaman yang lebih mendalam mengenai prinsip kerja AVR, model matematis sistem exciter generator, serta berbagai metode kontrol yang telah dikembangkan. Literatur yang digunakan mencakup buku teks sistem kontrol klasik, standar IEEE terkait sistem exciter, serta artikel jurnal terbaru yang membahas penggunaan kontroler modern seperti PID, SMC, maupun pendekatan *hybrid*. Hasil kajian pustaka ini menjadi dasar konseptual dalam merancang kontroler yang sesuai untuk sistem AVR.

Pengumpulan Data

Setelah memperoleh landasan teori, tahap berikutnya adalah pengumpulan data yang diperlukan untuk pemodelan sistem. Data yang dikumpulkan mencakup parameter-parameter generator sinkron, *exciter*, serta karakteristik dinamis dari AVR. Sumber data dapat berasal dari literatur, standar referensi, maupun hasil penelitian terdahulu yang relevan seperti disebutkan di Tabel 2.2. Data ini digunakan untuk dijadikan parameter yang dilakukan untuk perancangan kontrol *hybrid* AVR dan memastikan bahwa model matematis yang dibangun mampu merepresentasikan sistem AVR secara akurat, sehingga hasil simulasi dapat mendekati kondisi nyata.

Permodelan Matematis Sistem AVR

Permodelan dilakukan dengan menyusun *transfer function* (fungsi-alih) tiap komponen utama AVR yaitu *amplifier* (A) pada persamaan (2.1), *exciter* (E) pada persamaan (2.2), dan generator (G) pada persamaan (2.3) yang sebelumnya dibentuk dari persamaan (2.13). Kemudian mengalikan ketiganya menjadi model orde-tiga seperti persamaan (2.14).

Berikut lanjutan penurunan permodelan dengan memasukkan variabel yang datanya sudah ditetapkan,

$$G(s) = \frac{10 \cdot 1 \cdot 1}{(0,1s + 1)(0,4s + 1)(1,0s + 1)}$$

kalikan dan kembangkan penyebutnya, dan menghilangkan desimal di penyebut maka dikali 10,

Hak Cipta Ditanggung Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan atau menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

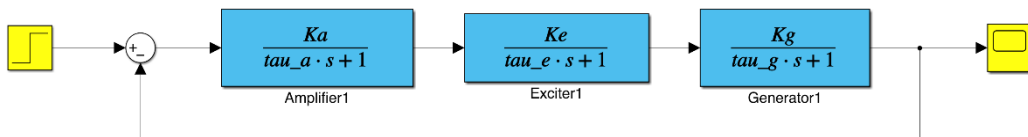
$$\left[\frac{10}{0,04s^3 + 0,54s^2 + 1,5s + 1} \times \frac{25}{25} \right]$$

diperoleh bentuk *transfer function* dari sistem AVR yaitu berupa orde tiga,

$$G(s) = \frac{250}{s^3 + 13,5s^2 + 37,5s + 25} \quad (3.1)$$

Pengujian Sistem AVR

Pengujian sistem AVR dilakukan untuk memastikan bahwa model matematis yang telah diturunkan dapat diimplementasikan pada simulasi berbasis MATLAB/Simulink. Konfigurasi pengujian dilakukan dalam mode *closed-loop* dengan menggunakan *input step* sebagai representasi perubahan tegangan referensi. Pada tahap ini, setiap blok utama dari AVR yaitu *amplifier*, *exciter*, dan generator direpresentasikan dalam bentuk fungsi alih sesuai parameter yang telah diperoleh. Berikut rangkaian Simulink sistem AVR ditunjukkan pada Gambar 3.2.



Gambar 3.2 Rangkaian blok Simulink *close-loop* sistem AVR

Untuk mempermudah implementasi simulasi, langkah-langkah pengujian disusun dalam bentuk algoritma sebagaimana ditunjukkan pada berikut ini.

Tabel 3.1 Algoritma Pengujian Sistem AVR

Algoritma 1: Pengujian Sistem AVR
Mulai 1. Membuat nama program matlab yaitu “AVR_openloop.m” 2. Membuat perintah <code>clc; clear all; close all;</code> (membersihkan <i>comand window</i>, menghapus variabel di <i>baseworkspace</i> dan menutup jendela <i>figure</i> yang masih terbuka)

Hak Cipta Dilindungi Undang-Undang	3. Mendefinisikan waktu: waktu mulai (T_{st}), waktu <i>sampling</i> (T_s) dan waktu akhir (T_{end}). 4. Definisikan parameter berdasarkan Tabel 2.2. 5. Tentukan <i>setpoint</i> sebesar 220 V (tegangan pada umumnya). 6. Rangkai Simulink Matlab dengan menyusun blok komponen secara seri dan memberikan blok <i>Step</i> di awal rangkaian dan <i>Scope</i> di akhir rangkaian. 7. Panggil Simulink Matlab dengan nama “aAVR_openloop.slx”. 8. Jalankan simulasi untuk memperoleh <i>figure scope</i> berupa respon sistem AVR. 9. Simpan data hasil simulasi sebagai keluaran awal.
End	

6. Desain SMC

Pada tahap ini dilakukan perancangan SMC untuk sistem AVR. Perancangan dimulai dengan mendefinisikan model matematis sistem dalam bentuk fungsi alih, kemudian diturunkan ke persamaan diferensial sebagai dasar perumusan hukum kendali. Selanjutnya, ditetapkan *error tracking*, dirancang permukaan luncur (*sliding surface*), dan diturunkan hukum kendali berbasis prinsip Lyapunov serta *reaching law*. Setiap langkah perancangan akan dijelaskan secara sistematis mulai dari model matematis hingga diperoleh bentuk akhir persamaan kontrol u_{smc} . Dimulai dari bentuk TF AVR diambil dari persamaan (3.1) yaitu :

$$\frac{Y(s)}{U(s)} = \frac{250}{s^3 + 13,5s^2 + 37,5s + 25} \quad (3.2)$$

untuk mempermudah proses analisis, setiap konstanta didefinisikan ulang sebagai parameter umum:

$$a = 250, \quad b = 13.5, \quad c = 37.5, \quad d = 25 \quad (3.3)$$

Dengan demikian, model sistem dapat dituliskan:

$$\frac{Y(s)}{U(s)} = \frac{a}{s^3 + bs^2 + cs + d} \quad (3.4)$$

setelah itu dilakukan invers transformasi Laplace:

$$\begin{aligned}\ddot{y}(t) + b\dot{y}(t) + cy(t) + dy(t) &= au(t) \\ \ddot{y}(t) &= -b\dot{y}(t) - cy(t) - dy(t) + au(t)\end{aligned}\quad (3.5)$$

Untuk mendesain SMC diperlukan variabel *error*, definisi dari *error tracking* yaitu:

$$e(t) = y_r(t) - y(t) \quad (3.6)$$

Jika persamaan *error* dinamik dipisah dengan menurunkan persamaan (3.6) sebagai berikut:

$$\dot{e}(t) = \dot{y}_r(t) - \dot{y}(t), \quad \ddot{e} = \ddot{y}_r(t) - \ddot{y}(t), \quad \ddot{e} = \ddot{y}_r(t) - \ddot{y}(t) \quad (3.7)$$

Permukaan luncur orde dua diambil dari persamaan (2.20) digunakan agar dinamika *tracking* stabil dan halus, yaitu:

$$s(t) = \ddot{e}(t) + \lambda_1 \dot{e}(t) + \lambda_0 e(t) \quad (3.8)$$

Pada penelitian ini pemilihan orde-2 bertujuan agar dinamika transien (baik kecepatan respon maupun redaman) dapat diatur melalui parameter λ . Selain itu, bentuk ini memudahkan integrasi dengan model AVR yang merupakan sistem orde-3, karena melibatkan sinyal $\ddot{e}$. Selanjutnya *sliding surface* yang diturunkan terhadap waktu dapat dituliskan sebagai berikut:

$$\dot{s} = \ddot{e}(t) + \lambda_1 \dot{e}(t) + \lambda_0 e(t) \quad (3.9)$$

Agar persamaan (3.9) dapat ditulis dalam bentuk yang lebih informatif, turunan-turunan *error* disubstitusikan dari persamaan turunan (3.7) diperoleh:

$$\dot{s}(t) = (\ddot{y}_r - \ddot{y})(t) + \lambda_1 (\dot{y}_r - \dot{y})(t) + \lambda_0 (y_r - y)(t) \quad (3.10)$$

lalu disubstitusikan dari persamaan (3.5) dengan bentuk sebagai berikut:

$$\begin{aligned}\dot{s}(t) &= \ddot{y}_r(t) - (-b\dot{y}(t) - cy(t) - dy(t) + au(t))(t) + \lambda_1 (\dot{y}_r(t) - \dot{y}(t)) + \lambda_0 (y_r(t) - y(t)) \\ \dot{s}(t) &= \ddot{y}_r(t) + b\dot{y}(t) + cy(t) + dy(t) - au(t) + \lambda_1 (\dot{y}_r(t) - \dot{y}(t)) + \lambda_0 (y_r(t) - y(t)) \\ \dot{s}(t) &= \ddot{y}_r(t) + b\dot{y}(t) + cy(t) + dy(t) - au(t) + \lambda_1 \dot{y}_r(t) - \lambda_1 \dot{y}(t) + \lambda_0 y_r(t) - \lambda_0 y(t)\end{aligned}$$

Kelompokkan suku referensi dan suku keluaran sehingga diperoleh bentuk akhir turunan AVR terhadap SMC yaitu:

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

$$\begin{aligned}\dot{s}(t) &= -au(t) + (\ddot{y}_r(t) + \lambda_1 \dot{y}_r(t) + \lambda_0 y_r(t) + \ddot{y}(b - \lambda_1) + \dot{y}(c - \lambda_0) + dy(t)) \\ \dot{s}(t) &= -au(t) + \Phi(t)\end{aligned}\quad (3.11)$$

dengan fungsi bantu:

$$\Phi(t) = (\ddot{y}_r(t) + \lambda_1 \dot{y}_r(t) + \lambda_0 y_r(t) + \ddot{y}(b - \lambda_1) + \dot{y}(c - \lambda_0) + dy(t)) \quad (3.12)$$

Untuk menentukan kontrol $u_{equivalent}$, menurut konsep *sliding mode* berdasarkan pada persamaan (2.19) yang disubstitusikan menjadi :

$$0 = -au_{eq}(t) + \Phi(t)$$

Sehingga kontrol u_{eq} diperoleh:

$$u_{eq}(t) = \frac{\Phi(t)}{a} \quad (3.13)$$

Setelah didapatkan kontrol u_{eq} , maka perlu membuat kontrol $u_{natural}$ dengan memasukkan persamaan kaidah *Lyapunov* atau *reaching law* diambil dari persamaan (2.21) untuk menjamin kestabilan sistem, berikut kaidah yang digunakan:

$$\dot{s}(t) = -\eta \times \text{sign}(s(t)) \quad (3.14)$$

berikutnya untuk mereduksi *chattering* maka menggunakan saturasi sebagai *boundary layer* sehingga bentuk akhir $\dot{s}$ sebagai berikut:

$$\dot{s}(t) = -\eta \times \text{sat}\left(\frac{s(t)}{\varepsilon}\right) \quad (3.15)$$

dari persamaan (3.11) lalu disubstitusikan ke persamaan (2.22) untuk membentuk komponen $u_{natural}$ diperoleh:

$$\begin{aligned}\dot{s}(t) &= -a[u_{eq} + u_{nat}] + \Phi(t) \\ \dot{s}(t) &= -au_{eq} - au_{nat} + \Phi(t)\end{aligned}$$

lalu dilanjutkan dengan substitusikan persamaan (3.13) yang telah dijabarkan sehingga:

Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

$$\dot{s}(t) = -a \left(\frac{\Phi(t)}{a} \right) - au_{nat} + \Phi(t)$$

$$\dot{s}(t) = -au_{nat}$$

Setelah sederhana fungsi $\dot{s}$ agar persamaan menyesuaikan dengan *reaching law* maka diperoleh kontrol $u_{natural}$:

$$-au_{nat} = -\eta \cdot sat \left(\frac{s(t)}{\varepsilon} \right)$$

$$u_{nat} = \frac{\eta}{a} \cdot sat \left(\frac{s(t)}{\varepsilon} \right)$$
(3.16)

Bentuk akhir dari model matematis SMC dengan menggabungkan kontrol u_{eq} dengan kontrol $u_{natural}$ terhadap sistem AVR yaitu:

$$u_{smc}(t) = u_{eq} + u_{nat}$$

$$u_{smc}(t) = \frac{1}{a} [\Phi(t)] + \frac{\eta}{a} sat \left(\frac{s(t)}{\varepsilon} \right)$$
(3.17)

Jika dijabarkan secara eksplisit, maka diperoleh:

$$u_{smc}(t) = \frac{1}{250} \left[(\ddot{y}_r(t) + \lambda_1 \ddot{y}_r(t) + \lambda_0 \dot{y}_r(t) + \ddot{y}(13,5 - \lambda_1)) + \frac{-\eta}{250} sat \left(\frac{s(t)}{\varepsilon} \right) \right] + \dot{y}(37,5 - \lambda_0) + 25y(t)$$
(3.18)

Sebagai acuan untuk merancang sebuah SMC yang dimasukkan ke Simulink. Parameter $\lambda_1, \lambda_0, \eta$, dan batas *boundary layer* ε ditentukan melalui proses *tuning* agar respon sistem menunjukkan kinerja terbaik, yaitu *rise time* cepat, *overshoot* rendah, serta *error* keadaan tunak minimal.

Tabel 3.2 Parameter SMC hasil tuning

Parameter	Nilai
λ_1	100
λ_0	1
η	100
ε	0.1

Pemilihan parameter SMC pada Tabel 3.2 diperoleh melalui proses *tuning* bertahap berdasarkan pengamatan langsung terhadap respon sistem AVR di MATLAB/Simulink. Tahap ini dilakukan untuk menilai bagaimana setiap kombinasi parameter memengaruhi karakteristik transien, khususnya kestabilan dan kecenderungan sistem mengalami *overshoot* maupun *chattering*. Pada konfigurasi awal dengan nilai parameter yang lebih agresif, respon sistem memang mencapai *setpoint* dengan cepat, namun muncul *overshoot* dan riak di sekitar referensi. Oleh karena itu, parameter kemudian disetel secara lebih konservatif hingga diperoleh kombinasi yang menghasilkan respon sangat halus tanpa osilasi berlebih, meskipun keluaran belum sepenuhnya mencapai nilai *setpoint*. Kondisi ini sengaja dipertahankan sebagai dasar untuk menguji kontribusi pengendali PID pada struktur *hybrid* dalam mengoreksi kekurangan SMC tersebut.

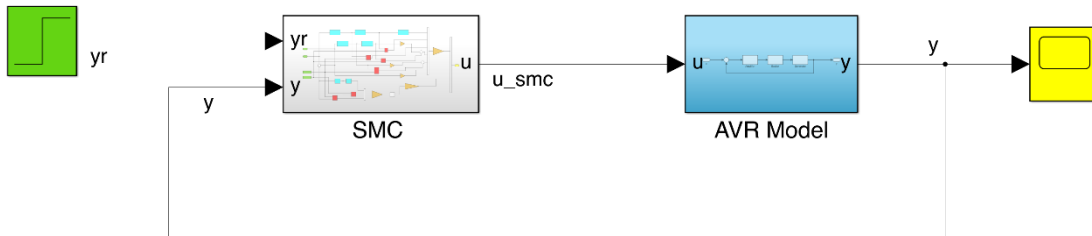
Kombinasi λ_1 , λ_0 , η , dan batas *boundary layer* ε yang terpilih memberikan respon paling stabil dengan bentuk transien yang mendekati orde satu. Nilai λ_1 yang besar ($\lambda_1 = 100$) memperkuat peran turunan *error* sehingga sistem memperoleh redaman yang kuat dan tidak menunjukkan *overshoot* maupun osilasi pada saat awal respon. Sebaliknya, nilai λ_0 yang relatif kecil ($\lambda_0 = 1$) membuat komponen *error* murni tidak terlalu agresif, sehingga keluaran bergerak perlahan menuju nilai referensi dan cenderung berhenti sedikit di bawah *setpoint*. Parameter $\eta = 100$ dipilih sebagai penguat aksi *switching* yang masih mampu menjaga pergerakan menuju permukaan luncur, namun dengan intensitas koreksi yang tidak terlalu besar sehingga tidak menimbulkan lonjakan tegangan. Sementara itu, nilai *boundary layer* $\varepsilon = 0,1$ menjaga fungsi $\text{sat}(s/\varepsilon)$ tetap cukup halus sehingga fenomena *chattering* yang muncul berada pada tingkat yang sangat kecil dan tidak mengganggu kestabilan sistem. Dengan demikian, konfigurasi pada Tabel 3.2 menghasilkan respon AVR yang stabil, aman, dan bebas *overshoot*, tetapi masih menyisakan *steady-state error* kecil terhadap *setpoint*. Kekurangan inilah yang kemudian menjadi motivasi penambahan pengendali PID pada struktur *hybrid* untuk meningkatkan ketepatan pencapaian tegangan referensi.



Gambar 3.3 Rangkaian blok Simulink Kontrol SMC

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Setelah dirangkai SMC pada Gambar 3.3 kemudian menghubungkan rangkaian SMC ke sistem AVR utama, ini akan menjadi sistem yang mempunyai kontrol. Berikut disajikan rangkaian gabungan AVR dengan SMC pada Gambar 3.4 berikut.



Gambar 3.4 Rangkaian blok Simulink gabungan SMC dengan AVR

Selain dalam bentuk blok Simulink pada tahap perancangan dibentuk juga melalui langkah-langkah algoritma. Algoritma ini menjabarkan tahapan implementasi SMC mulai dari membaca masukan referensi, menghitung *error* dan turunannya, membentuk *sliding surface*, hingga menghasilkan sinyal kontrol sesuai hukum *Lyapunov reaching law*. Penyajian algoritma pada Tabel 3.3 bertujuan untuk memudahkan proses implementasi dan memastikan setiap tahapan perhitungan dapat diikuti secara runtut dan konsisten.

Tabel 3.3 Algoritma Desain SMC pada Sistem AVR

Algoritma 2: Desain SMC pada Sistem AVR	
Mulai	
1.	Membuat nama program matlab yaitu “AVR_SMC_onlyy.m”
2.	Membuat perintah <code>clc; clear all; close all;</code> (membersihkan <i>comand window</i> , menghapus variabel di <i>baseworkspace</i> dan menutup jendela <i>figure</i> yang masih terbuka)
3.	Mendefinisikan waktu: waktu mulai (T_{st}), waktu <i>sampling</i> (T_s) dan waktu akhir (T_{end}).
4.	Definisikan parameter berdasarkan Tabel 2.2.
5.	Tentukan <i>setpoint</i> sebesar 220 V (tegangan pada umumnya).
6.	Rangkai Simulink Matlab dengan menyusun blok sesuai pada persamaan (3.18) dan memberikan blok <i>Step</i> di awal rangkaian dan <i>Scope</i> di akhir rangkaian
7.	Panggil Simulink Matlab dengan nama “aAVR_openloop.slx”.

8. Jalankan simulasi untuk memperoleh <i>figure scope</i> berupa respon sistem AVR dengan SMC.
9. Simpan data hasil simulasi sebagai keluaran awal.
End

7. Desain PID

Kontroler PID merupakan salah satu jenis pengendali yang paling banyak digunakan dalam sistem kendali karena strukturnya sederhana namun efektif dalam memperbaiki karakteristik respon sistem. Kontrol PID bekerja dengan mengombinasikan tiga aksi kendali, yaitu *proportional* (P) untuk mempercepat respon sistem, *integral* (I) untuk mengurangi *error* keadaan tunak (*steady-state error*), dan *derivative* (D) untuk memperbaiki redaman sehingga mengurangi *overshoot* dan osilasi. Dalam penelitian ini, kontrol PID digunakan bukan hanya sebagai pengendali konvensional pada sistem AVR, tetapi juga sebagai *compensator* yang berperan dalam memperbaiki karakteristik transien dan kehalusan sinyal kontrol pada metode SMC. Dengan demikian, penerapan PID bertujuan untuk menganalisis sejauh mana pengaruhnya terhadap peningkatan performa kontroler SMC dalam menjaga kestabilan tegangan sistem AVR.

Secara matematis, persamaan kontrol PID dalam domain waktu dituliskan sebelumnya pada persamaan (2.29) sebagai:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

sedangkan dalam domain Laplace dituliskan sebagai:

$$U(s) = \left(K_p + \frac{K_i}{s} + K_d s \right) E(s) \quad (3.19)$$

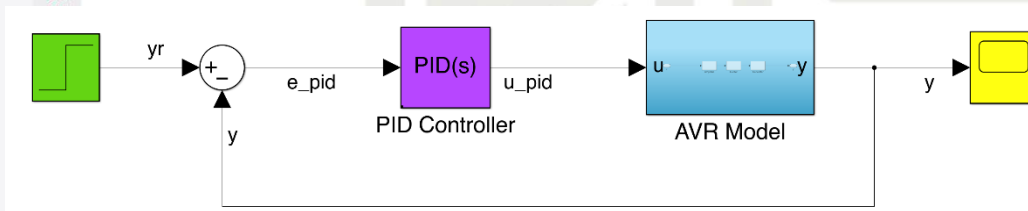
dengan K_p , K_i , dan K_d masing-masing merupakan konstanta penguatan proporsional, integral dan derivatif.

Penalaan parameter PID dilakukan menggunakan metode *tuning* Matlab hingga diperoleh kombinasi parameter yang mampu menghasilkan respon terbaik sesuai kriteria performansi, yaitu *rise time* yang cepat, *overshoot* kecil, dan *error* keadaan tunak minimal. Dari hasil proses tuning, didapatkan parameter PID sebagai berikut:

Tabel 3.4 Parameter PID hasil tuning

Parameter	Nilai
K_p	0.001
K_i	0.25
K_d	0.35

Dari parameter yang sudah ditentukan maka selanjutnya yaitu merangkai sistem AVR dengan pengendali PID pada berikut ini.



Gambar 3.5 Rangkaian Blok Simulink PID dan AVR

Implementasi kontrol PID ini dilakukan pada lingkungan Simulink dengan menggunakan blok *PID Controller*. Blok kontroler kemudian dihubungkan dengan model matematis AVR dalam konfigurasi *closed-loop*, di mana sinyal *error* diperoleh dari selisih antara referensi tegangan dan *output* sistem. Parameter K_p , K_i , dan K_d yang telah ditentukan dimasukkan ke dalam blok PID, kemudian simulasi dijalankan untuk memperoleh respon sistem sebagai dasar evaluasi performansi.

Tabel 3.5 Algoritma Desain PID pada Sistem AVR

Algoritma 3: Desain PID pada Sistem AVR	
Mulai	
1.	Membuat nama program matlab yaitu “AVR_PID_oonly.m”
2.	Membuat perintah <code>clc; clear all; close all;</code> (membersihkan <i>comand window</i> , menghapus variabel di <i>baseworkspace</i> dan menutup jendela <i>figure</i> yang masih terbuka)

Hak Cipta Dilindungi Undang-Undang	3. Mendefinisikan waktu: waktu mulai (T_{st}), waktu <i>sampling</i> (T_s) dan waktu akhir (T_{end}). 4. Definisikan parameter berdasarkan Tabel 2.2. 5. Tentukan <i>setpoint</i> sebesar 220 V (tegangan pada umumnya). 6. Rangkai Simulink Matlab dengan menyusun blok PID serta menyesuaikan parameter PID seperti pada Tabel 3.4 dan memberikan blok <i>Step</i> di awal rangkaian dan <i>Scope</i> di akhir rangkaian 7. Panggil Simulink Matlab dengan nama “AVR_pid_only.slx”. 8. Jalankan simulasi untuk memperoleh <i>figure scope</i> berupa respon sistem AVR dengan kontrol PID. 9. Simpan data hasil simulasi sebagai keluaran awal.
End	

Penentuan parameter kontrol PID pada penelitian ini dilakukan menggunakan pendekatan metode heuristik, yaitu proses penyetelan bertahap yang didasarkan pada pengamatan langsung terhadap karakteristik respon sistem. Tahap awal dilakukan dengan memberikan nilai kecil pada setiap konstanta PID sebagai konfigurasi konservatif untuk menjaga stabilitas sistem dan mencegah terjadinya lonjakan awal yang berlebihan. Selanjutnya, nilai K_p ditingkatkan secara perlahan untuk memperoleh percepatan respon tanpa menimbulkan *overshoot* yang tidak diinginkan. Setelah itu, K_i disesuaikan untuk mengurangi kesalahan keadaan tunak yang masih tersisa, sedangkan K_d ditetapkan untuk meningkatkan redaman dan mengurangi kecenderungan osilasi pada zona transien. Proses penalaan ini diulang hingga diperoleh kombinasi parameter yang memberikan respon paling seimbang sesuai kriteria performansi AVR yang diharapkan.

Selain parameter yang digunakan pada pengujian utama, dilakukan pula eksplorasi *tuning* heuristik dengan variasi nilai K_p untuk mengamati pengaruhnya terhadap karakteristik *rise time* dan kestabilan sistem. Hasil eksplorasi tersebut disajikan pada Lampiran A, dan menunjukkan bahwa peningkatan K_p cenderung mempercepat *rise time* namun diikuti oleh peningkatan osilasi, sehingga parameter yang dipilih pada penelitian ini ditetapkan sebagai kompromi antara kecepatan respon dan kestabilan sistem.

Pendekatan heuristik dipilih karena sifatnya yang fleksibel terhadap perubahan dinamika sistem AVR, sehingga mampu menghasilkan parameter yang lebih realistis dibandingkan metode analitik murni. Selain itu, penyetelan PID pada struktur *hybrid* ini juga diarahkan untuk mengkompensasi kelemahan pengendali SMC khususnya kecenderungan *chattering* dan ketidakmampuannya mencapai *setpoint* pada konfigurasi konservatif sehingga sinyal kontrol yang dihasilkan menjadi lebih halus dan keluaran dapat bergerak tepat menuju nilai referensi. Dengan demikian, parameter PID yang diperoleh tidak hanya berfungsi sebagai pengendali mandiri, tetapi juga sebagai komponen kompensasi yang memperbaiki kualitas respon pada sistem *hybrid* SMC–PID.

8. Desain Hybrid

Integrasi antara pengendali SMC dan PID dilakukan untuk memanfaatkan keunggulan masing-masing metode pada sistem AVR orde tiga. SMC dikenal memiliki ketahanan yang baik terhadap ketidakpastian parameter dan gangguan eksternal, namun pada implementasi ini menunjukkan keterbatasan ketika diterapkan langsung pada model AVR orde tiga, yaitu keluaran cenderung berhenti di bawah *setpoint* meskipun bentuk respon sudah stabil dan tanpa osilasi besar. Selain itu, sifat dasar SMC yang mengandalkan aksi *switching* tetap berpotensi menimbulkan *chattering* yang dapat menurunkan kualitas aktuasi. Di sisi lain, pengendali PID relatif sederhana dan efektif untuk mengurangi *steady-state error* serta memperhalus respon, tetapi kurang tangguh bila berdiri sendiri pada kondisi nonlinier dan saat terjadi ketidakpastian model. Oleh karena itu, penggabungan kedua metode ini diharapkan dapat menghasilkan sistem kendali yang tangguh, cepat, stabil, serta mampu mencapai nilai *setpoint* dengan respon yang tetap halus.

Dalam perancangan struktur *hybrid*, SMC ditempatkan sebagai pengendali utama karena kemampuannya menjaga kestabilan dan *robustness* sistem terhadap variasi parameter dan gangguan eksternal. Namun, respon SMC tunggal pada AVR orde tiga menunjukkan bahwa keluaran masih menyisakan *steady-state error* dan tidak sepenuhnya mencapai *setpoint*, meskipun transiennya relatif rapi dan bebas *overshoot*. Untuk memperbaiki karakteristik ini, pengendali PID ditambahkan sebagai pengendali pendamping (*compensator*) yang memberikan koreksi halus pada sinyal SMC. Pendekatan ini dikenal sebagai *parallel hybrid structure*, di mana SMC bertugas mengendalikan dinamika utama dan menjaga ketahanan sistem, sedangkan PID berfungsi mengangkat keluaran dari posisi

di bawah *setpoint* menuju nilai referensi secara lebih presisi, sekaligus mengurangi osilasi kecil di daerah transien dan meminimalkan *chattering* yang muncul pada keluaran.

Pemilihan struktur paralel dilakukan karena jika PID ditempatkan sebagai pengendali utama atau disusun seri, sifat *robust* dari SMC berpotensi berkurang dan sistem menjadi lebih sensitif terhadap ketidakpastian model. Dengan konfigurasi paralel, kontribusi SMC terhadap kestabilan dan ketangguhan sistem tetap terjaga, sementara PID memberikan perbaikan pada aspek kehalusan respon dan ketepatan pencapaian *setpoint*. Dengan demikian, integrasi SMC–PID dalam struktur paralel memberikan kompromi yang seimbang antara kecepatan respon, kestabilan transien, kemampuan mencapai *setpoint*, dan ketahanan terhadap gangguan pada sistem AVR orde tiga.

Bentuk matematis kontrol *hybrid* SMC–PID dituliskan sebagai:

$$u(t) = u_{smc}(t) + u_{pid}(t) \quad (3.20)$$

dengan:

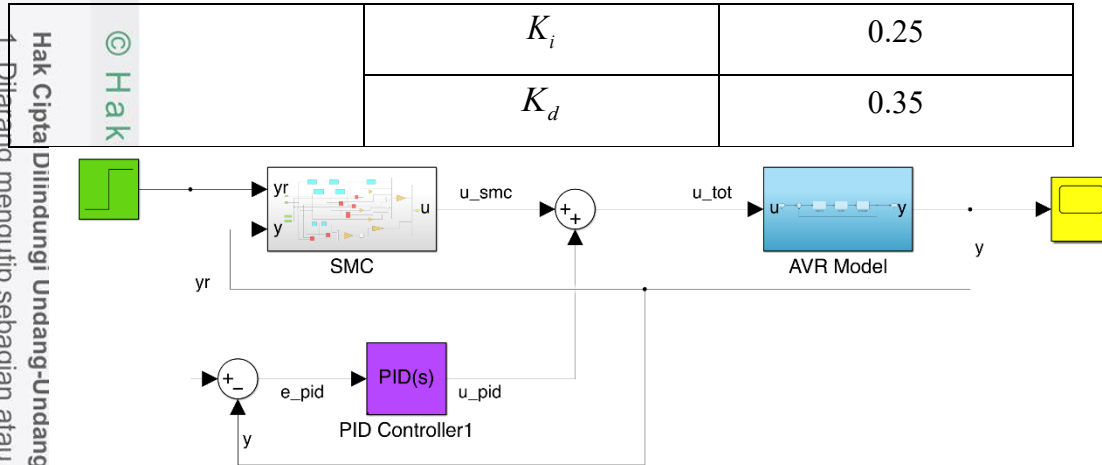
$$u_{pid}(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

$$u_{smc}(t) = \frac{1}{250} \left[(\ddot{y}_r(t) + \lambda_1 \ddot{y}_r(t) + \lambda_0 \dot{y}_r(t) + \dot{y}(13,5 - \lambda_1)) + \frac{\eta}{250} \text{sat}\left(\frac{s(t)}{\varepsilon}\right) \right]$$

sehingga hukum kendali *hybrid* merupakan kombinasi antara sinyal kontrol *robust* dari SMC dan sinyal korektif halus dari PID. Parameter *hybrid* terdiri dari gabungan parameter PID dan SMC yang ditentukan melalui proses *tuning*. Tabel 3.6 berikut menunjukkan hasil *tuning* yang digunakan dalam penelitian ini:

Tabel 3.6 Parameter *hybrid* SMC–PID hasil *tuning*

Kontroler	Parameter	Nilai
SMC	λ_1	100
	λ_0	1
	η	100
	ε	0.1
PID	K_p	0.001



Gambar 3.6 Rangkaian Blok Simulink *hybrid* SMC-PID pada AVR

Implementasi *hybrid* SMC–PID pada Simulink dilakukan dengan membangun dua jalur kontrol secara paralel seperti pada

Gambar 3.6. Sinyal *error* dihitung dari selisih antara referensi dan keluaran sistem, kemudian diproses dalam blok PID untuk menghasilkan u_{pid} , serta diproses dalam blok SMC untuk menghasilkan u_{smc} . Kedua sinyal ini kemudian dijumlahkan melalui blok Sum, dan hasilnya menjadi sinyal kendali akhir $u(t)$ yang diberikan ke sistem AVR.

Tabel 3. Algoritma Desain PID pada Sistem AVR

Algoritma 4: Desain PID pada Sistem AVR
Mulai 1. Membuat nama program matlab yaitu “AVR_hybriid.m” 2. Membuat perintah <code>clc; clear all; close all;</code> (membersihkan <i>comand window</i>, menghapus variabel di <i>baseworkspace</i> dan menutup jendela <i>figure</i> yang masih terbuka) 3. Mendefinisikan waktu: waktu mulai (T_{st}), waktu <i>sampling</i> (T_s) dan waktu akhir (T_{end}). 4. Definisikan parameter berdasarkan Tabel 2.2. 5. Tentukan <i>setpoint</i> sebesar 220 V (tegangan pada umumnya). 6. Definisikan parameter SMC dari Tabel 3.2 dan parameter kontrol PID dari Tabel 3.4.

Hak Cipta Dilindungi Undang-Undang	7. Rangkai Simulink Matlab dengan menyusun blok sesuai pada persamaan (3.18) dan rangkai Simulink Matlab dengan menambah blok PID sebagai pengendali tambahan serta menyesuaikan parameter PID seperti pada Tabel 3.4 dan memberikan blok <i>Step</i> di awal rangkaian dan <i>Scope</i> di akhir rangkaian 8. Panggil Simulink Matlab dengan nama “AVR_hybrid.slx”. 9. Jalankan simulasi untuk memperoleh <i>figure scope</i> berupa respon sistem AVR dengan kontrol <i>hybrid</i> SMC-PID. 10. Simpan data hasil simulasi sebagai keluaran awal.
End	

Melalui rancangan pada Tabel 3.7 ini, kontroler *hybrid* SMC–PID diharapkan mampu memberikan performa terbaik, yaitu kombinasi antara *robustness* dari SMC dan kehalusan respon dari PID. Bukti efektivitas struktur paralel ini akan ditunjukkan melalui simulasi dan analisis pada Bab 4.

BAB V

KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil perancangan, pemodelan, dan simulasi sistem Automatic Voltage Regulator (AVR) yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Pengendalian AVR menggunakan SMC *static* mampu meningkatkan kestabilan sistem dan mempercepat respon transien dibandingkan AVR konvensional, namun masih menunjukkan keterbatasan dalam hal ketepatan pencapaian *setpoint*. Hal ini ditunjukkan oleh nilai *steady-state error* (ESS) yang relatif besar pada pengujian SMC murni, meskipun sistem telah mencapai kondisi stabil.
2. Keterbatasan SMC statik dalam mencapai *setpoint* tercermin dari nilai ESS yang tinggi, yang merupakan konsekuensi dari struktur kendali SMC yang berfokus pada konvergensi menuju permukaan *sliding*, bukan pada pemaksaan *error* keadaan tunak menjadi nol. Hasil ini konsisten dengan karakteristik SMC *static* yang digunakan dalam penelitian.
3. Integrasi kendali PID ke dalam struktur SMC terbukti efektif dalam menyelesaikan permasalahan *steady-state error*. Berdasarkan hasil simulasi, pengendali *Hybrid* SMC–PID mampu menurunkan ESS secara signifikan hingga mendekati nol, sehingga tegangan keluaran AVR dapat mengikuti nilai referensi dengan tingkat akurasi yang jauh lebih baik dibandingkan SMC murni maupun AVR konvensional.
4. Pengendali *Hybrid* SMC–PID menunjukkan performa paling stabil secara keseluruhan, dengan kombinasi *overshoot* yang kecil, respon yang halus, serta kemampuan mempertahankan tegangan keluaran pada berbagai kondisi pengujian. Hal ini menunjukkan bahwa peran PID sebagai kompensator mampu melengkapi keunggulan SMC dalam hal *robustness* sekaligus memperbaiki akurasi keadaan tunak sistem.

Secara keseluruhan, penelitian ini membuktikan bahwa penambahan kendali PID pada struktur SMC berperan signifikan dalam meningkatkan kinerja pengendalian AVR, khususnya dalam menyelesaikan permasalahan ESS dan

meningkatkan ketepatan pencapaian *setpoint*, tanpa mengorbankan kestabilan dan ketahanan sistem terhadap variasi kondisi operasi.

Saran

Penelitian ini masih memiliki ruang pengembangan yang dapat dilakukan pada penelitian selanjutnya. Secara umum, pengujian yang dilakukan dalam penelitian ini mencakup beberapa kategori berikut:

1. Implementasi kontrol *Hybrid* SMC–PID dapat diuji pada model AVR dengan dinamika *plant* yang lebih kompleks atau kondisi operasi yang lebih bervariasi, seperti perubahan beban yang cepat, gangguan harmonisa, maupun fluktuasi frekuensi.
2. Fenomena *chattering* pada SMC dapat diminimalkan lebih lanjut melalui penggunaan metode *higher-order sliding mode* atau penerapan fungsi aproksimasi kontinu yang lebih adaptif.
3. Penelitian lanjutan dapat mengintegrasikan kendali *Hybrid* ini dengan teknik observasi modern, seperti *Luenberger Observer* atau *Kalman Filter*, agar mampu memperkirakan variabel sistem secara lebih presisi pada kondisi gangguan berat.

Selain itu, implementasi pada perangkat keras *real-time* seperti DSP atau mikrokontroler berperforma tinggi dapat memberikan validasi eksperimental terhadap hasil simulasi yang diperoleh. Dengan pengembangan yang tepat, pengendali *Hybrid* SMC–PID berpotensi menjadi solusi yang lebih efektif dan andal dalam aplikasi AVR pada sistem pembangkit listrik modern.

DAFTAR PUSTAKA

- [1] IEEE Recommended Practice for Excitation System Models for Power System Stability Studies," *IEEE Std 421.5-1992*, pp. 1-56, 1992, doi: 10.1109/IEEESTD.1992.106975.
- [2] A. M. Prasetia, L. Sartika, and A. Ma'ruf, "Penggunaan pid pada automatic voltage regulator (avr) untuk kestabilan tegangan terminal generator sinkron 3 fasa," *Elektrika Borneo*, vol. 9, no. 2, 2023.
- [3] G. Y. Puspitaputri, C. W. Priananda, and D. F. Syahbana, "Automatic Voltage Regulator (AVR) Generator dengan Mikrokontroler Menggunakan Metode Hill Climbing," *Jurnal Teknik ITS*, vol. 10, no. 2, pp. B171-B176, 2021.
- [4] L. Liliana, "Analisis Kinerja Automatic Voltage Regulator (AVR) Sebagai Pengontrol Suplai Tegangan Eksitasi Generator Ac 3 Phasa: Peformance Analysis of Automatic Voltage Regulator (AVR) as a 3-Phase AC Generator Excitation Voltage Supply Controller," *Indonesian Journal of Electrical Engineering and Renewable Energy (IJEERE)*, vol. 1, no. 1, pp. 37-44, 06/20 2021, doi: 10.57152/ijeere.v1i1.88.
- [5] S. Veinović, D. Stojić, and D. Joksimović, "Optimized four-parameter PID controller for AVR systems with respect to robustness," *International Journal of Electrical Power & Energy Systems*, vol. 135, p. 107529, 2022/02/01/ 2022, doi: <https://doi.org/10.1016/j.ijepes.2021.107529>.
- [6] L. Wu, J. Liu, S. Vazquez, and S. K. Mazumder, "Sliding mode control in power converters and drives: A review," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 3, pp. 392-406, 2021.
- [7] H. Sukisno and Sunyoto, "MATLAB/SIMULINK UNTUK STUDI PENGENDALIAN DAYA REAKTIF DAN TEGANGAN PADA SISTEM TENAGA LISTRIK DENGAN AUTOMATIC VOLTAGE REGULATOR (AVR)," vol. 2, p. 2002, 05/26 2020.
- [8] K. Ogata, *Modern control engineering*. 2020.
- [9] V. I. Utkin, *Sliding modes in control and optimization*. Springer Science & Business Media, 2013.
- [10] J. J. E. Slotine and W. Li, *Applied nonlinear control* (no. 1). Prentice hall Englewood Cliffs, NJ, 1991.
- [11] C. Edwards and S. K. Spurgeon, *Sliding mode control: theory and applications*. CRC press, 1998.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

© Hak Cipta Dilindungi Undang-Undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

- [12] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control," *International journal of Control*, vol. 76, no. 9-10, pp. 924-941, 2003.
- [13] E. Köse and S. Coşkun, "Time-delay AVR system analysis using PSO-based PID controller," *Avrupa Bilim ve Teknoloji Dergisi*, no. 18, pp. 981-991, 2020.
- [14] N. Pachauri, "Water cycle algorithm-based PID controller for AVR," *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, vol. 39, no. 3, pp. 551-567, 2020.
- [15] L. Sartika, A. M. Prasetia, and F. R. Sudirman, "Kontrol tegangan pada terminal generator menggunakan AVR berbasis neural network," *JURNAL ELTEK*, vol. 22, no. 1, pp. 29-37, 2024.
- [16] A. Zaidi, I. I. Basith, and V. Khan, "Intelligent PID Controller for Automatic Voltage Regulation," *Electrical and Electronic Engineering*, vol. 12, no. 1, pp. 1-12, 26/09/2022 2022, doi: 10.5923/j.eee.20221201.01.
- [17] A. A. Mahal and A.-R. S. Hadi, "Robust control for automatic voltage regulator system based on learning sliding mode control," *International Journal of Advanced Mechatronic Systems*, vol. 11, no. 1, pp. 26-39, 2024, doi: 10.1504/ijamechs.2024.137556.
- [18] M. Furat and I. Eker, "Chattering-eliminated adaptive sliding-mode control: an experimental comparison study," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 24, no. 2, pp. 605-620, 2016.
- [19] C. Lu and J. Yuan, "Adaptive Super-Twisting Sliding Mode Control of Permanent Magnet Synchronous Motor," *Complexity*, vol. 2021, no. 1, p. 1957510, 2021, doi: <https://doi.org/10.1155/2021/1957510>.
- [20] A. Kumar, "Nonlinear AVR for power system stabilisers robust phase compensation design," *IET Generation, Transmission & Distribution*, vol. 14, no. 21, pp. 4927-4935, 2020, doi: <https://doi.org/10.1049/iet-gtd.2020.0092>.
- [21] I. Irianto, A. J. Arman Jaya, and A. Amin Setiadji, "Implementasi Kontroler PID Pada AVR (Automatic Voltage Regulator) Untuk Pengaturan Tegangan Eksitasi Generator Sinkron 3 Fasa," *EEPIS Final Project*, 2011.
- [22] D. Sajnekar, M. Kolhe, S. Deshpande, R. Moharil, N. Patidar, and K. Ogura, "Design of PID controller for automatic voltage regulator and validation using hardware in the loop technique," *International Journal of Smart Grid and Clean Energy*, vol. 7, pp. 75-89, 04/01 2018, doi: 10.12720/sgce.7.2.75-89.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
- [23] A. M. Prasetya, L. Sartika, and A. Saad, "Optimasi AVR Pada Generator Sinkron Menggunakan Metode Adaptive Neuro Fuzzy Inference System (ANFIS)," *MEDIA ELEKTRIKA*, vol. 16, no. 2, pp. 137-150, 2024.
 - [24] A. P. Prasetyo, "Pemodelan, Simulasi, dan Analisis Sistem Regulator Tegangan Otomatis Generator Sinkron Dengan Pengendali PID," *SIMETRIS*, vol. 18, no. 2, pp. 50-56, 2024.
 - [25] P. Kundur, "Power system stability," *Power system stability and control*, vol. 10, no. 1, pp. 7-1, 2007.
 - [26] K. Máslo, A. Kasembe, and M. Kolcun, "Simplification and unification of IEEE standard models for excitation systems," *Electric Power Systems Research*, vol. 140, pp. 132-138, 2016.
 - [27] B. M. Weedy, B. J. Cory, N. Jenkins, J. B. Ekanayake, and G. Strbac, *Electric power systems*. John Wiley & Sons, 2012.
 - [28] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding mode control and observation*. Springer, 2014.
 - [29] V. I. Utkin and I. Vadim, "Sliding mode control," *Variable structure systems: from principles to implementation*, vol. 66, no. 1, 2004.
 - [30] M. R. Homaeinezhad and F. Fotoohi Nia, "Robust nonlinear model predictive sliding mode control algorithm for saturated uncertain multivariable mechanical systems," *Journal of Vibration and Control*, vol. 29, no. 7-8, pp. 1524-1538, 2023, doi: 10.1177/10775463211065883.
 - [31] H. Han, X. Wu, and J. Qiao, "Design of Robust Sliding Mode Control With Adaptive Reaching Law," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 11, pp. 4415-4424, 2020, doi: 10.1109/TSMC.2018.2852626.
 - [32] C. J. Fallaha, M. Saad, H. Y. Kanaan, and K. Al-Haddad, "Sliding-Mode Robot Control With Exponential Reaching Law," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 2, pp. 600-610, 2011, doi: 10.1109/TIE.2010.2045995.
 - [33] X. Zhang, H. Li, and M. Shao, "Fixed-Time-Convergent Sliding Mode Control with Sliding Mode Observer for PMSM Speed Regulation," *Sensors*, vol. 24, no. 5, p. 1561, 2024. [Online]. Available: <https://www.mdpi.com/1424-8220/24/5/1561>.
 - [34] S. Islam and X. P. Liu, "Robust Sliding Mode Control for Robot Manipulators," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 6, pp. 2444-2453, 2011, doi: 10.1109/TIE.2010.2062472.

- [35] V. Utkin and L. Hoon, "Chattering Problem in Sliding Mode Control Systems," in *International Workshop on Variable Structure Systems, 2006. VSS'06.*, 5-7 June 2006 2006, pp. 346-350, doi: 10.1109/VSS.2006.1644542.
- [36] H. Lee and V. I. Utkin, "Chattering suppression methods in sliding mode control systems," *Annual Reviews in Control*, vol. 31, no. 2, pp. 179-188, 2007/01/01/ 2007, doi: <https://doi.org/10.1016/j.arcontrol.2007.08.001>.
- [37] L. Tan, J. Gao, Y. Luo, and L. Zhang, "Super-twisting sliding mode control with defined boundary layer for chattering reduction of permanent magnet linear synchronous motor," *Journal of Mechanical Science and Technology*, vol. 35, no. 5, pp. 1829-1840, 2021/05/01 2021, doi: 10.1007/s12206-021-0403-9.
- [38] G. K. McMillan, "Industrial Applications of PID Control," in *PID Control in the Third Millennium: Lessons Learned and New Approaches*, R. Vilanova and A. Visioli Eds. London: Springer London, 2012, pp. 415-461.
- [39] A. G. Daful, "Comparative study of PID tuning methods for processes with large & small delay times," in *2018 Advances in Science and Engineering Technology International Conferences (ASET)*, 6 Feb.-5 April 2018 2018, pp. 1-7, doi: 10.1109/ICASET.2018.8376915.
- [40] "Intelligent PID Controller for Automatic Voltage Regulation," doi: 10.5923/j.eee.20221201.01
- [41] T. S. Schei, "Automatic tuning of PID controllers based on transfer function estimation," *Automatica*, vol. 30, no. 12, pp. 1983-1989, 1994/12/01/ 1994, doi: [https://doi.org/10.1016/0005-1098\(94\)90060-4](https://doi.org/10.1016/0005-1098(94)90060-4).
- [42] Y. C. Kim, L. H. Keel, and S. P. Bhattacharyya, "Transient response control via characteristic ratio assignment," *IEEE Transactions on Automatic Control*, vol. 48, no. 12, pp. 2238-2244, 2003, doi: 10.1109/TAC.2003.820153.
- [43] E. Joelianto and D. Williamson, "Transient response improvement of feedback control systems using hybrid reference control," *International Journal of Control*, vol. 82, no. 10, pp. 1955-1970, 2009/10/01 2009, doi: 10.1080/00207170902866120.
- [44] A. Gilat, *MATLAB: An introduction with Applications*. John Wiley & Sons, 2017.
- [45] N. S. Nise, *Control systems engineering*. John Wiley & Sons, Inc, 2011.
- [46] IEEE Recommended Practice for Excitation System Models for Power System Stability Studies," *IEEE Std 421.5-2016 (Revision of IEEE Std 421.5-2005)*, pp. 1-207, 2016, doi: 10.1109/IEEESTD.2016.7553421.
- [47] K. R. Padiyar, *Power System Dynamics: Stability and Control*. Wiley, 1999.

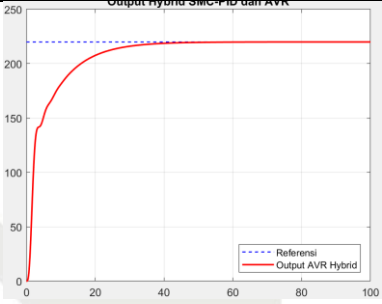
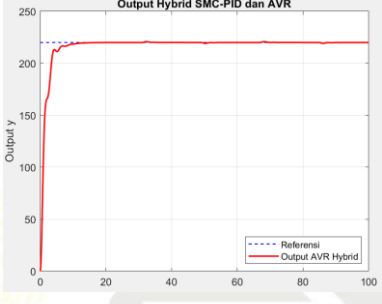
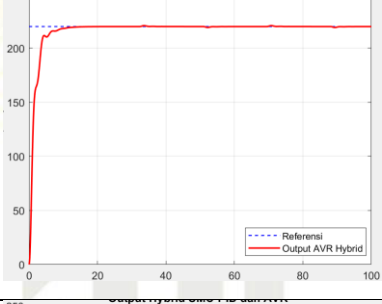
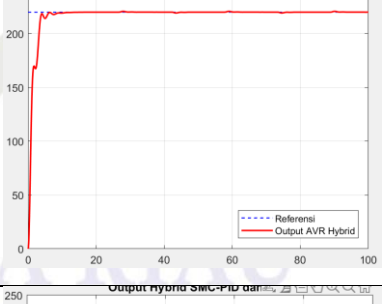
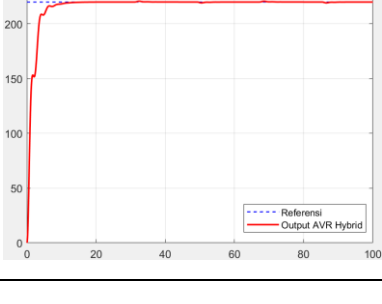
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

LAMPIRAN A

B1. Tabel Heuristik terhadap Parameter PID pada Hybrid SMC-PID AVR

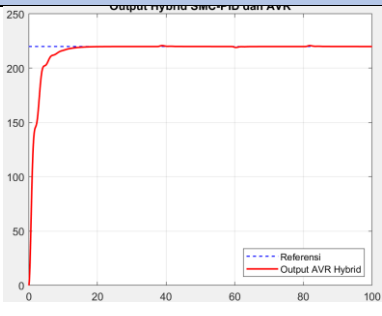
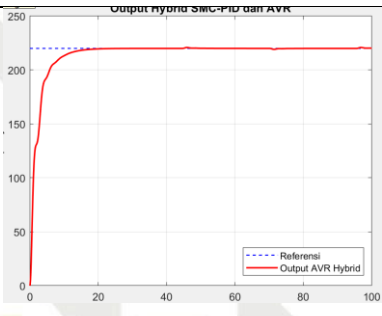
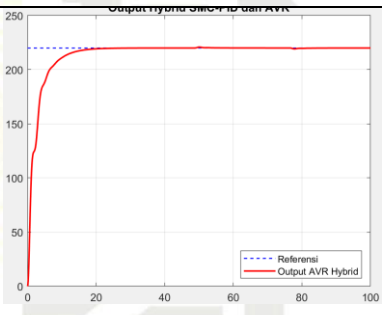
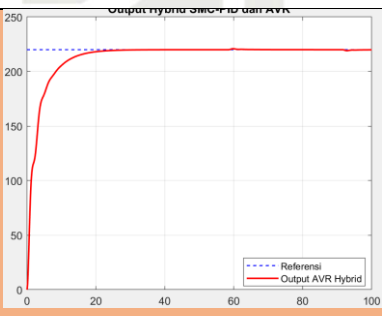
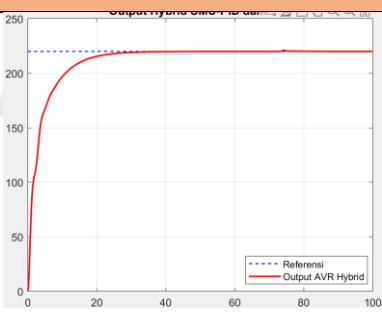
Kp	Ki	Kd	Tr (s)	Ts (s)	Overshoot (%)	ESS (V)	Catatan bentuk grafik
0	0.1422	0	13.9744	29.0758	0	0.0013	
0.001	0.5	0.4	3.0036	6.2216	0.3740	0.0015	
0.001	0.48	0.38	3.0650	6.5924	0.3810	0.0066	
0.001	0.58	0.5	2.8265	5.4158	0.3522	0.0040	
0.001	0.48	0.5	3.0872	6.0306	0.3605	0.0019	

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

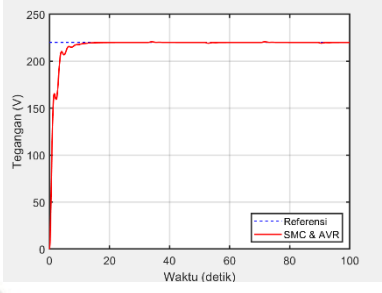
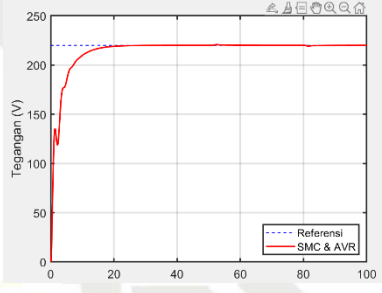
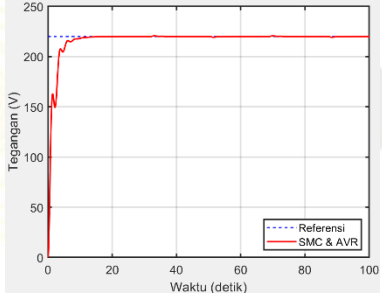
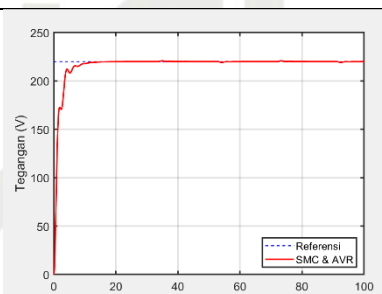
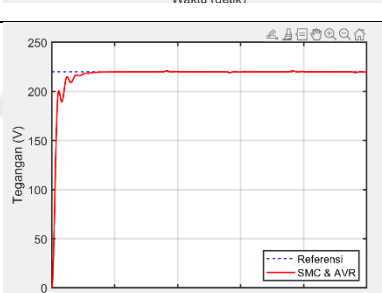
masalah.

asim Riau

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Kp	Ki	Kd	Tr (s)	Ts (s)	Overshoot (%)	ESS (V)	Catatan bentuk grafik
0.001	0.4	0.4	3.5028	9.2205	0.3793	0.0011	
0.001	0.33	0.4	5.2006	11.8091	0.2899	0.2088	
0.001	0.3	0.4	5.7221	12.9483	0.3885	0.0019	
0.001	0.25	0.35	7.5201	15.7365	0.4492	0.1071	
0.001	0.20	0.30	9.8873	20.1616	0.4107	0.0074	

B-2. Tabel Heuristik terhadap Parameter Kp - PID pada Hybrid SMC-PID AVR

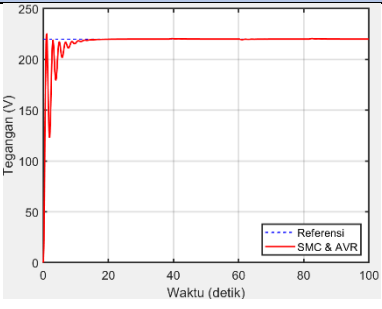
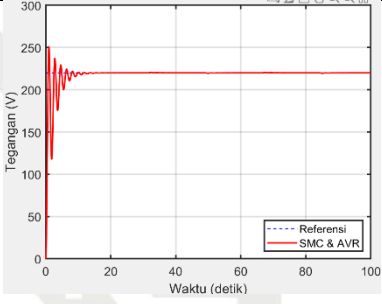
Kp	Ki	Kd	Tr (s)	Ts (s)	Overshoot (%)	ESS (V)	Catatan bentuk grafik
0.1	0.5	0.4	3.0363	7.8579	0.3587	0.0099	
0.1	0.3	0.5	6.3726	13.6434	0.3544	0.0061	
0.1	0.5	0.5	3.0223	7.5395	0.3427	0.0044	
0.1	0.5	0.3	3.0691	8.1525	0.3791	0.0230	
0.1	0.5	0.1	1.5688	7.3118	0.4577	0.1245	

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Kp	Ki	Kd	Tr (s)	Ts (s)	Overshoot (%)	ESS (V)	Catatan bentuk grafik
0.1	0.6	0.1	1.2087	6.9729	1.2476	0.0147	
0.3	0.6	0.1	1.1284	8.4091	0.6102	0.0088	
0.3	0.8	0.3	0.8640	7.6028	5.7928	4.4391e-04	
0.5	0.6	0.3	0.8693	7.4299	0.3309	0.0681	
0.5	0.6	0.5	0.7743	8.5175	0.2894	0.0184	

Kp	Ki	Kd	Tr (s)	Ts (s)	Overshoot (%)	ESS (V)	Catatan bentuk grafik
0.8	0.6	0.5	0.5816	9.7581	2.3989	0.0015	
1	0.8	0.6	0.4771	9.2072	14.0647	0.0013	

i tanpa mencantumkan dan menyebutkan sumber:

penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

LAMPIRAN B

B.4. Kode Program AVR Tanpa Pengendali

```
%% program model sistem AVR openloop
% DM x Robby Hibban
% 28 Oktober 2025

%% clear histori
clear all; close all; clc;

%% time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

%% parameter
Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
Vref = 220;

%% call simulink
sim("aAVR_openloop.slx")

%% plot
figure;
ax = axes;
hold(ax, 'on');
set(ax, 'FontSize', 12, 'LineWidth', 1.5);
grid(ax, 'on'); box(ax, 'on');
plot(T, y1(:,1), 'b--', 'LineWidth', 2); % Garis biru output AVR Hybrid
hold on
plot(T, y1(:,2), 'r', 'LineWidth', 1.5); % Garis merah putus referensi
xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show', 'Location', 'southeast', 'FontSize', 11);
legend('Referensi', 'Output')
hold(ax, 'off');

%% Info Grafik
info = stepinfo(y, T); % Rise time, settling time, peak time, overshoot
rise_time = info.RiseTime
format compact
settling_time = info.SettlingTime
format compact
peak_time = info.PeakTime
format compact
overshoot = info.Overshoot
format compact
```

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



```
% Error steady-state (Ess)
r_final = Vref(end); % referensi
y_final = y(end); % output
ess = abs(r_final - y_final)
format compact

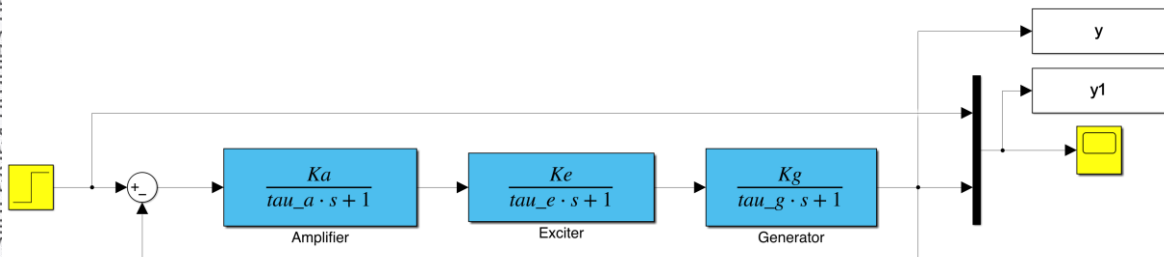
% Time delay (50% steady-state point)
half_value = 0.5 * y_final;
idx_delay = find(y >= half_value, 1);
time_delay = T(idx_delay)
format compact
```

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Rangkaian Simulink Sistem AVR Tanpa Pengendali

B2. Kode Program AVR dan SMC

```
% program model sistem AVR dan SMC
% DM x Robby Hibban
% 28 Oktober 2025

% Clear histori
clear all; close all; clc;

% Time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

% Parameter
Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
Vref = 220;

% Definisikan SMC
a2 = 13.5; a1 = 37.5; a0 = 25; n_nat = 250; gain_utama = 1/250; % nilai AVR
lamda_1 = 100; lamda_0 = 1; n_sat = -100; ep_slon = 0.1; % parameter SMC

% Call simulink
sim('AVR_SMC_updateSESESE.slx')
```

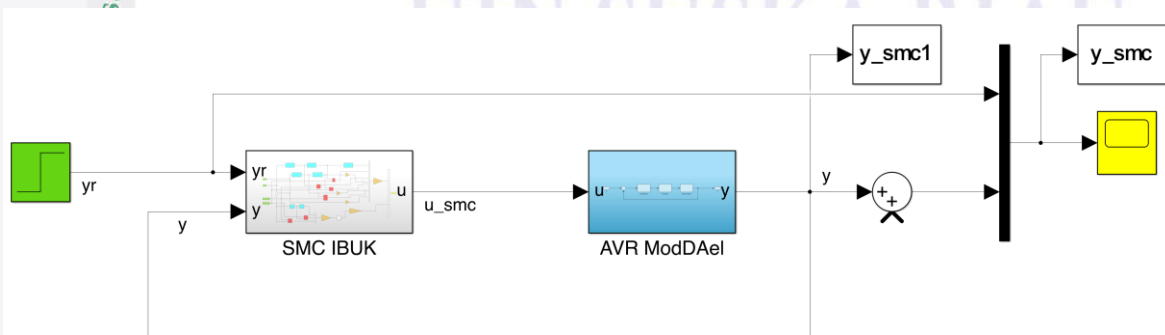
1. Diarangi mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Diarangi mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

```
% plot
figure;
ax = axes;
hold(ax,'on');
set(ax,'FontSize',12,'LineWidth',1.5);
grid(ax,'on'); box(ax,'on');
plot(T,y_smc(:,1),'b--','LineWidth',1); % Output
hold on
plot(T,y_smc(:,2),'r','LineWidth',1.5); % Referensi
xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show','Location','southeast','FontSize',11);
legend('Referensi','Output')
hold(ax,'off');

% Info Grafik
info = stepinfo(y_smc1, T); % Rise time, settling time, peak time, overshoot
rise_time = info.RiseTime
format compact
settling_time = info.SettlingTime
format compact
peak_time = info.PeakTime
format compact
overshoot = info.Overshoot
format compact

% Error steady-state (Ess)
r_final = Vref(end); % referensi
y_final = y_smc1(end); % output
ess = abs(r_final - y_final)
format compact

% Time delay (50% steady-state point)
half_value = 0.5 * y_final;
idx_delay = find(y_smc1 >= half_value, 1);
time_delay = T(idx_delay)
format compact
```



Rangkaian Simulink Sistem AVR dengan SMC

B.3. Kode Program AVR dan Hybrid SMC-PID

```
%% program model sistem AVR dan Hybrid
% DM x Robby Hibban
% 28 Oktober 2025

%% Clear histori
clear all; close all; clc;

%% time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

%% parameter
Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
Vref = 220;

%% definisikan TF AVR
num = 250;
den = [1 13.5 27.5 25];
G = tf(num,den);

%% definisikan SMC
a2 = 13.5; a1 = 37.5; a0 = 25; n_nat = 250; gain_utama = 1/250; % nilai AVR
lamda_1 = 100; lamda_0 = 1; n_sat = -100; ep_slon = 0.1; % parameter SMC

%% definisikan PID
Kp = 0.001; Ki = 0.25; Kd = 0.35;

%% call simulink
sim('AVR_hybrid_bagus.slx')
yd=y_hybd1;

%% plot
figure;
ax = axes;
hold(ax,'on');
set(ax,'FontSize',12,'LineWidth',1.5);
grid(ax,'on'); box(ax,'on');
plot(T, y_hybd(:,1), 'b', 'LineWidth', 1); % Garis biru output AVR Hybrid
hold on
plot(T, y_hybd(:,2), 'r--', 'LineWidth', 1.5); % Garis merah referensi
xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show','Location','southeast','FontSize',11);
legend('Referensi','SMC & AVR','SMC-PID & AVR')
```

Hak Cipta Dilindungi Undang-Undang

1. Perangkat mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

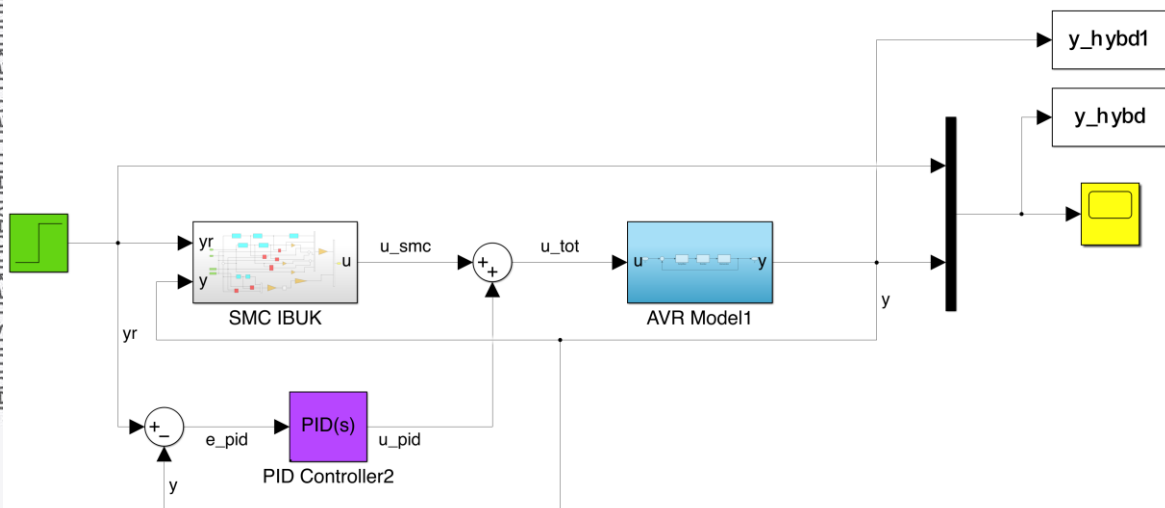
b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

```
% Info Grafik
info = stepinfo(y_hybd1, T); % Rise time, settling time, peak time, overshoot
rise_time = info.RiseTime
format compact
settling_time = info.SettlingTime
format compact
peak_time = info.PeakTime
format compact
overshoot = info.Overshoot
format compact

% Error steady-state (Ess)
r_final = Vref(end); % referensi
y_final = y_hybd1(end); % output
ess = abs(r_final - y_final)
format compact

% Time delay (50% steady-state point)
half_value = 0.5 * y_final;
idx_delay = find(y_hybd1 >= half_value, 1);
time_delay = T(idx_delay)
format compact
```



Rangkaian Simulink Sistem AVR dan *Hybrid* SMC-PID

B-4. Kode Program dan Simulink AVR-Hybrid dengan Disturbance di Sinyal *Input*

```
% program model sistem AVR dan Hybrid
% DM x Robby Hibban
% 28 Oktober 2025

%% Clear histori
```



```

clear all; close all; clc;

%% time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

%% parameter
Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
Vref = 220;

%% definisikan SMC
a2 = 13.5; a1 = 37.5; a0 = 25; n_nat = 250; gain_utama = 1/250; % nilai AVR
lamda_1 = 100; lamda_0 = 1; n_sat = -100; ep_slon = 0.1; % parameter SMC

%% definisikan PID
Kp = 0.001; Ki = 0.25; Kd = 0.35;

%% call simulink
sim("AVR_hybrid_sigedit.slx")

%% plot
figure;
ax = axes;
hold(ax,'on');
set(ax,'FontSize',12,'LineWidth',1.5);
grid(ax,'on'); box(ax,'on');
plot(T, y_hybd_sg, 'b--', 'LineWidth', 1.8, 'DisplayName','Referensi');
plot(T, y_hybd_sg1, 'r', 'LineWidth', 1.8, 'DisplayName','Output');
xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show','Location','southeast','FontSize',11);

% --- Garis batas zona ---
h1 = xline(20,'k--','LineWidth',1);
h2 = xline(80,'k--','LineWidth',1);
set(h1,'HandleVisibility','off');
set(h2,'HandleVisibility','off');

% --- Label zona (posisi Y sesuaikan sendiri) ---
text(10, 40, 'I','FontSize',14,'FontWeight','bold'); % Zona I
text(50, 40, 'II','FontSize',14,'FontWeight','bold'); % Zona II
text(90, 40, 'III','FontSize',14,'FontWeight','bold'); % Zona III
hold(ax,'off');

%% ===== INFO PARAMETER RESPON (MANUAL) =====
% Semua parameter dihitung manual berbasis definisi teori
% Tr : rise time 10%-90%
% Td : time delay 50%

```

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



```
% Ts : settling time ±5%
% ESS : steady-state error (rata-rata akhir)
% OVS : overshoot (% terhadap referensi)

t = T;
segTime = [ 0 20; % Sesi I
            20 80; % Sesi II
            80 100]; % Sesi III

segName = {'Sesi I', 'Sesi II', 'Sesi III'};

for k = 1:3
    t0 = segTime(k,1);
    t1 = segTime(k,2);

    ----- Ambil data per sesi -----
    idx = (t >= t0) & (t < t1);
    t_seg = t(idx) - t0; % waktu lokal
    y_seg = y_hybd_sg1(idx); % output
    r_seg = y_hybd_sg(idx); % referensi

    ----- Nilai dasar -----
    y0 = y_seg(1);
    y_ss = mean(r_seg(end-10:end)); % referensi akhir
    Delta = y_ss - y0;

    % ===== RISE TIME (10% - 90%) =====
    y10 = y0 + 0.10*Delta;
    y90 = y0 + 0.90*Delta;

    k10 = find((y_seg - y10).*sign(Delta) >= 0, 1, 'first');
    k90 = find((y_seg - y90).*sign(Delta) >= 0, 1, 'first');

    if isempty(k10) || isempty(k90)
        Tr = NaN;
    else
        Tr = t_seg(k90) - t_seg(k10);
    end

    % ===== TIME DELAY (50%) =====
    y50 = y0 + 0.50*Delta;
    k50 = find((y_seg - y50).*sign(Delta) >= 0, 1, 'first');

    if isempty(k50)
        Td = NaN;
    else
        Td = t_seg(k50);
    end

    % ===== SETTLING TIME (±5%) =====
    band = 0.05*abs(Delta);
    err = abs(y_seg - y_ss);

    idx_out = find(err > band);
    if isempty(idx_out)
        Ts = 0;
    else
        last_out = idx_out(end);
    end
end
```

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber.
- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

```

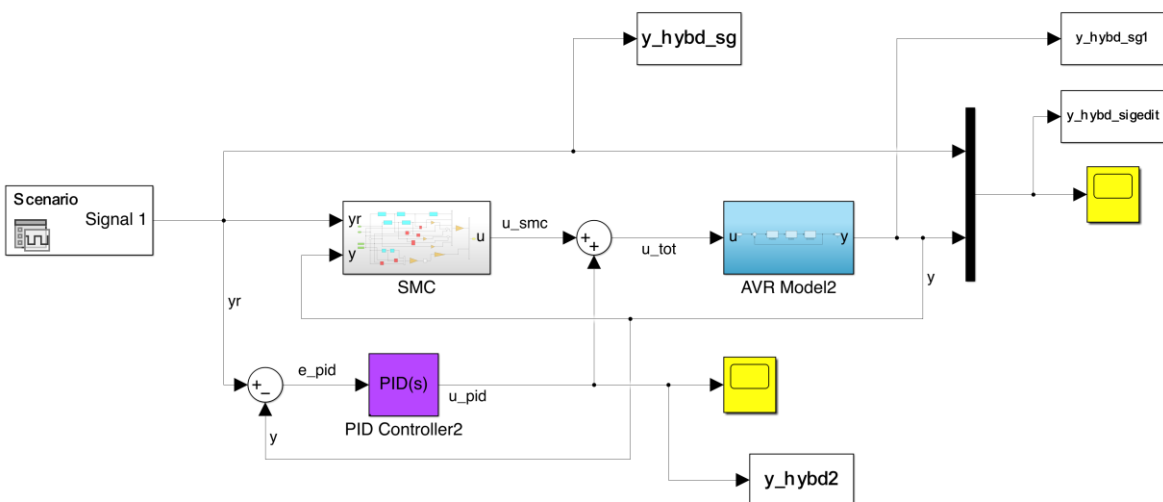
if last_out == length(t_seg)
    Ts = NaN;
else
    Ts = t_seg(last_out+1);
end
end

===== OVERSHOOT =====
if Delta > 0 % step naik
    y_peak = max(y_seg);
    if y_peak > y_ss
        Overshoot = (y_peak - y_ss)/abs(Delta)*100;
    else
        Overshoot = 0;
    end
else % step turun
    Overshoot = NaN; % tidak didefinisikan
end

===== STEADY-STATE ERROR =====
idx_ss = t_seg >= 0.7*t_seg(end); % 30% akhir sesi
ESS = y_ss - mean(y_seg(idx_ss));

% ===== OUTPUT =====
fprintf('== %s (t = %.1f s s.d. %.1f s) ==\n', segName{k}, t0, t1);
fprintf('Overshoot = %.3f %%\n', Overshoot);
fprintf('Rise time (Tr) = %.3f s\n', Tr);
fprintf('Settling time (Ts) = %.3f s\n', Ts);
fprintf('Time delay (Td) = %.3f s\n', Td);
fprintf('ESS = %.4f V\n\n', ESS);
end

```



Rangkaian Simulink AVR-Hybrid dengan *Disturbance* di Sinyal Input

B-5. Kode Program dan Simulink AVR-Hybrid dengan *Disturbance* di Sinyal Kendali (u)

```

%% Program model sistem AVR dan Hybrid
% DMx Robby Hibban
% 28 Oktober 2025

```

```

%% clear histori
clear all; close all; clc;

%% time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

%% parameter
Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
Vref = 220;

%% definisikan SMC
a2 = 13.5; a1 = 37.5; a0 = 25; n_nat = 250; gain_utama = 1/250; % nilai AVR
lamda_1 = 100; lamda_0 = 1; n_sat = -100; ep_slon = 0.1; % parameter SMC

%% definisikan PID
Kp = 0.001; Ki = 0.25; Kd = 0.35;

%% Call simulink
sim("AVR_hybrid_sigedituuu.slx")

%% plot
figure;
ax = axes;
hold(ax,'on');
set(ax,'FontSize',12,'LineWidth',1.5);
grid(ax,'on'); box(ax,'on');
plot(T, y_hybd_sigedit(:,1), 'b--', 'LineWidth', 1.8,
'DisplayName','Referensi');
plot(T, y_hybd_sigedit(:,2), 'r', 'LineWidth', 1.8,
'DisplayName','Output');
xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show','Location','southeast','FontSize',11);

% -- Garis batas zona ---
h1 = xline(20,'k--','LineWidth',1);
h2 = xline(80,'k--','LineWidth',1);
set(h1,'HandleVisibility','off');
set(h2,'HandleVisibility','off');

% -- Label zona (posisi Y sesuaikan sendiri) ---
text(10, 140, 'I','FontSize',14,'FontWeight','bold'); % Zona I
text(50, 140, 'II','FontSize',14,'FontWeight','bold'); % Zona II
text(90, 140, 'III','FontSize',14,'FontWeight','bold'); % Zona III
hold(ax,'off');

%% ===== INFO PARAMETER RESPON (MANUAL) =====
% Semua parameter dihitung manual berbasis definisi teori
% Tr : rise time 10%-90%
% Td : time delay 50%
% Ts : settling time ±5%
% ES : steady-state error (rata-rata akhir)
% OV : overshoot (% terhadap referensi)

```

1. Diarangi mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Diarangi mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



```

t = T;
segTime = [ 0 20; % Sesi I
            20 80; % Sesi II
            80 100]; % Sesi III

segName = {'Sesi I', 'Sesi II', 'Sesi III'};

for k = 1:3
    t0 = segTime(k,1);
    t1 = segTime(k,2);

    % ----- Ambil data per sesi -----
    idx = (t >= t0) & (t < t1);
    t_seg = t(idx) - t0; % waktu lokal
    y_seg = y_hybd_sigedu(idx); % output
    r_seg = y_hybd_sigedu1(idx); % referensi

    % ----- Nilai dasar -----
    y0 = y_seg(1);
    y_ss = mean(r_seg(end-10:end)); % referensi akhir
    Delta = y_ss - y0;

    % ===== RISE TIME (10% - 90%) =====
    y10 = y0 + 0.10*Delta;
    y90 = y0 + 0.90*Delta;

    k10 = find((y_seg - y10).*sign(Delta) >= 0, 1, 'first');
    k90 = find((y_seg - y90).*sign(Delta) >= 0, 1, 'first');

    if isempty(k10) || isempty(k90)
        Tr = NaN;
    else
        Tr = t_seg(k90) - t_seg(k10);
    end

    % ===== TIME DELAY (50%) =====
    y50 = y0 + 0.50*Delta;
    k50 = find((y_seg - y50).*sign(Delta) >= 0, 1, 'first');

    if isempty(k50)
        Td = NaN;
    else
        Td = t_seg(k50);
    end

    % ===== SETTLING TIME (±5%) =====
    band = 0.05*abs(Delta);
    err = abs(y_seg - y_ss);

    idx_out = find(err > band);
    if isempty(idx_out)
        Ts = 0;
    else
        last_out = idx_out(end);
        if last_out == length(t_seg)
            Ts = NaN;
        else
            Ts = t_seg(last_out+1);
        end
    end
end

```

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

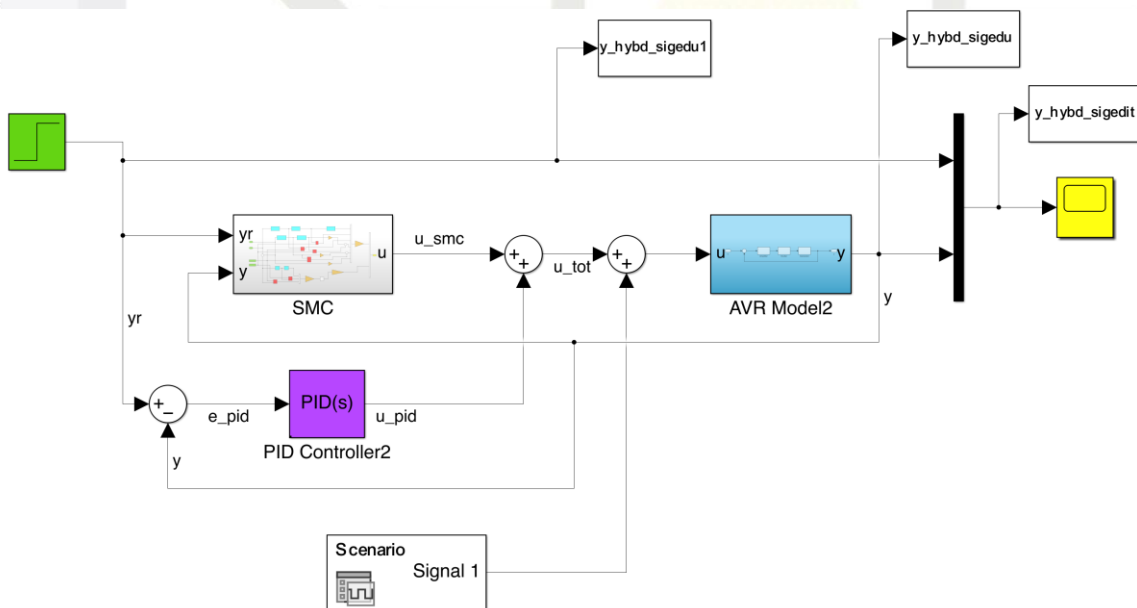
Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber.
- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

```

end
end
% ===== OVERSHOOT =====
if Delta > 0 % step naik
    y_peak = max(y_seg);
    if y_peak > y_ss
        Overshoot = (y_peak - y_ss)/abs(Delta)*100;
    else
        Overshoot = 0;
    end
else % step turun
    Overshoot = NaN; % tidak didefinisikan
end
% ===== STEADY-STATE ERROR =====
idx_ss = t_seg >= 0.7*t_seg(end); % 30% akhir sesi
ESS = y_ss - mean(y_seg(idx_ss));
% ===== OUTPUT =====
fprintf('=== %s (t = %.1f s s.d. %.1f s) ===\n', segName{k}, t0, t1);
fprintf('Overshoot = %.3f %%\n', Overshoot);
fprintf('Rise time (Tr) = %.3f s\n', Tr);
fprintf('Settling time (Ts) = %.3f s\n', Ts);
fprintf('Time delay (Td) = %.3f s\n', Td);
fprintf('ESS = %.4f V\n\n', ESS);
end

```



Rangkaian Simulink AVR-Hybrid dengan *Disturbance* di Sinyal Kendali (u)

B-6. Kode Program dan Simulink AVR-Hybrid dengan *Disturbance* di Sinyal Output (y)

```

% program model sistem AVR dan Hybrid
% Dikx Robby Hibban
% 28 Oktober 2025

```

1. Diarangi mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Diarangi mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

```

%% clear histori
clear all; close all; clc;

%% time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

%% parameter
Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
Vref = 220;

%% definisikan SMC
a2 = 13.5; a1 = 37.5; a0 = 25; n_nat = 250; gain_utama = 1/250; % nilai AVR
lamda_1 = 100; lamda_0 = 1; n_sat = -100; ep_slon = 0.1; % parameter SMC

%% definisikan PID
Kp = 0.001; Ki = 0.25; Kd = 0.35;

%% call simulink
sim("AVR_hybrid_sigidityyy.slx")

%% plot
figure;
ax = axes;
hold(ax,'on');
set(ax,'FontSize',12,'LineWidth',1.5);
grid(ax,'on'); box(ax,'on');

plot(T, y_hybd_sigedit(:,1), 'b--', 'LineWidth', 1.8, 'DisplayName','Referensi');
plot(T, y_hybd_sigedit(:,2), 'r', 'LineWidth', 1.8, 'DisplayName','Output');

xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show','Location','southeast','FontSize',11);

% --- Garis batas zona ---
h1 = xline(20,'k--','LineWidth',1);
h2 = xline(80,'k--','LineWidth',1);
set(h1,'HandleVisibility','off');
set(h2,'HandleVisibility','off');

% --- Label zona (posisi Y sesuaikan sendiri) ---
text(10, 140, 'I','FontSize',14,'FontWeight','bold'); % Zona I
text(50, 140, 'II','FontSize',14,'FontWeight','bold'); % Zona II
text(90, 140, 'III','FontSize',14,'FontWeight','bold'); % Zona III
hold(ax,'off');

%% ===== INFO PARAMETER RESPON (MANUAL) =====
% Semua parameter dihitung manual berbasis definisi teori
% Tr : rise time 10%-90%
% Td : time delay 50%
% Ts : settling time ±5%
% ES : steady-state error (rata-rata akhir)
% OS : overshoot (% terhadap referensi)

```




Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

```
t = T;
segTime = [ 0 20; % Sesi I
            20 80; % Sesi II
            80 100]; % Sesi III
segName = {'Sesi I', 'Sesi II', 'Sesi III'};
for k = 1:3
    t0 = segTime(k,1);
    t1 = segTime(k,2);
    % ----- Ambil data per sesi -----
    idx = (t >= t0) & (t < t1);
    t_seg = t(idx) - t0; % waktu lokal
    y_seg = y_hybd_sigidy(idx); % output
    r_seg = y_hybd_sigidy1(idx); % referensi
    % ----- Nilai dasar -----
    y0 = y_seg(1);
    y_ss = mean(r_seg(end-10:end)); % referensi akhir
    Delta = y_ss - y0;
    % ===== RISE TIME (10% - 90%) =====
    y10 = y0 + 0.10*Delta;
    y90 = y0 + 0.90*Delta;
    k10 = find((y_seg - y10).*sign(Delta) >= 0, 1, 'first');
    k90 = find((y_seg - y90).*sign(Delta) >= 0, 1, 'first');
    if isempty(k10) || isempty(k90)
        Tr = NaN;
    else
        Tr = t_seg(k90) - t_seg(k10);
    end
    % ===== TIME DELAY (50%) =====
    y50 = y0 + 0.50*Delta;
    k50 = find((y_seg - y50).*sign(Delta) >= 0, 1, 'first');
    if isempty(k50)
        Td = NaN;
    else
        Td = t_seg(k50);
    end
    % ===== SETTLING TIME (±5%) =====
    band = 0.05*abs(Delta);
    err = abs(y_seg - y_ss);
    idx_out = find(err > band);
    if isempty(idx_out)
        Ts = 0;
    else
        last_out = idx_out(end);
        if last_out == length(t_seg)
            Ts = NaN;
        else
            Ts = t_seg(last_out);
        end
    end
end
```

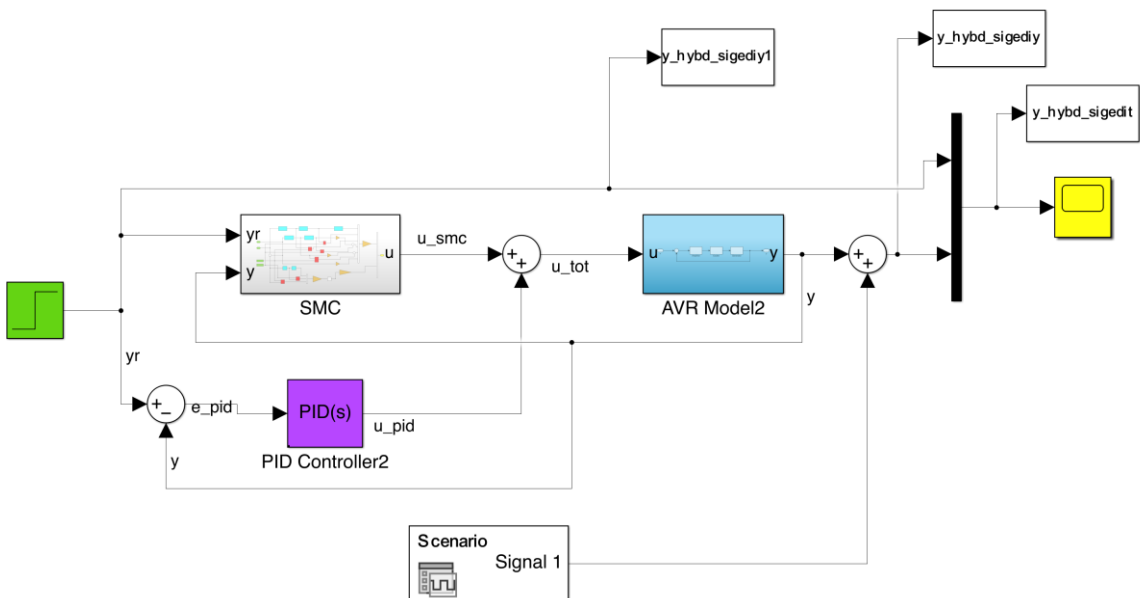
Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber.
- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

```

Ts = t_seg(last_out+1);
end
end
% ===== OVERSHOOT =====
if Delta > 0 % step naik
    y_peak = max(y_seg);
    if y_peak > y_ss
        Overshoot = (y_peak - y_ss)/abs(Delta)*100;
    else
        Overshoot = 0;
    end
else % step turun
    Overshoot = NaN; % tidak didefinisikan
end
% ===== STEADY-STATE ERROR =====
idx_ss = t_seg >= 0.7*t_seg(end); % 30% akhir sesi
ESS = y_ss - mean(y_seg(idx_ss));
% ===== OUTPUT =====
fprintf('== %s (t = %.1f s s.d. %.1f s) ==\n', segName{k}, t0, t1);
fprintf('Overshoot = %.3f %%\n', Overshoot);
fprintf('Rise time (Tr) = %.3f s\n', Tr);
fprintf('Settling time (Ts) = %.3f s\n', Ts);
fprintf('Time delay (Td) = %.3f s\n', Td);
fprintf('ESS = %.4f V\n\n', ESS);
end

```



Rangkaian Simulink AVR-Hybrid dengan *Disturbance* di Sinyal Output (*y*)

B-7. Kode Program Noise 6%

```
clear all; close all; clc;

%% Jalankan sistem AVR
load AVR_to_noise2; % Pastikan berisi variabel 'yd' hasil simulasi AVR
yd = yd'; % Transposisi bila perlu

%% Variasi level noise (% dari sinyal maksimum)
noise_levels = 0.02, 0.06, 0.10;

for i = 1:length(noise_levels)
    sigma = max(yd) * noise_levels(i); % Skala noise berdasarkan level
    var_s = sigma.^2;
    R = diag([var_s var_s]); % Matriks kovarians (opsional)

    %% Generate noise acak (Gaussian)
    q = randn(size(yd)); % Noise distribusi normal (mean=0, var=1)

    %% Normalisasi agar mean=0, var=1 lalu skalakan sesuai sigma
    q1 = (q - mean(q)) ./ std(q); % Normalisasi
    v = q1 .* sigma; % Noise aktual sesuai skala sigma

    %% Simpan hasil noise ke file berbeda
    filename = sprintf('data_noiseerefresh_%02d.mat', round(noise_levels(i)*100));
    save(filename, 'v');
    fprintf('File %s dibuat (sigma = %.2f%%)\n', filename, noise_levels(i)*100);
end

%% plot Noise
figure;
ax = axes;
hold(ax, 'on');
set(ax, 'FontSize', 12, 'LineWidth', 1.5);
grid(ax, 'on'); box(ax, 'on');
plot(T, v)
xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show', 'Location', 'southeast', 'FontSize', 11);
legend('Referensi', 'Output')
hold(ax, 'off');
```

B-8. Kode Program dan Simulink AVR-Hybrid dengan Noise 6%

```
% Program model sistem AVR dan Hybrid
% DM-x Robby Hibban
% 28 Oktober 2025

% Clear histori
clear all; close all; clc;

% Time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

% parameter
```

1. Diarangi mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Diarangi mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



```

Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
Vref = 220;

%% definisikan SMC
a2 = 13.5; a1 = 37.5; a0 = 25; n_nat = 250; gain_utama = 1/250; % nilai AVR
lamda_1 = 100; lamda_0 = 1; n_sat = -100; ep_slon = 0.1; % parameter SMC

%% definisikan PID
Kp = 0.001; Ki = 0.25; Kd = 0.35;

%% call simulink
% untuk AVR Hybrid w/ noise
load data_noise refresh_06
vs=timeseries(v,T);
sim('AVR_hybrid_noiseeyy.slx')

%% plot
figure;
ax = axes;
hold(ax,'on');
set(ax,'FontSize',12,'LineWidth',1.5);
grid(ax,'on'); box(ax,'on');
plot(T, y_hybd_sgnoi(:,1), 'b--', 'LineWidth', 1);
hold on
plot(T, y_hybd_sgnoi(:,2), 'r', 'LineWidth', 0.5);
xlabel('Waktu (detik)');
ylabel('Tegangan (V)');
legend('show','Location','southeast','FontSize',11);
legend('Referensi','Output')
hold(ax,'off');

%% Info Grafik
t = T(:); % ini mengubah row → column
y = y_hybd_sgnoi1(:); % output noisy (kolom)
r = y_hybd_sgnoi2(:); % referensi (kolom)
% Samakan panjang sinyal jika perlu
N = min([length(t), length(y), length(r)]);
t = t(1:N);
y = y(1:N);
r = r(1:N);

% Hitung error & RMSE
e = r - y;
RMSE = sqrt(mean(e.^2));
fprintf('RMSE = %.4f V\n', RMSE);

```

Hak Cipta Dilindungi Undang-Undang

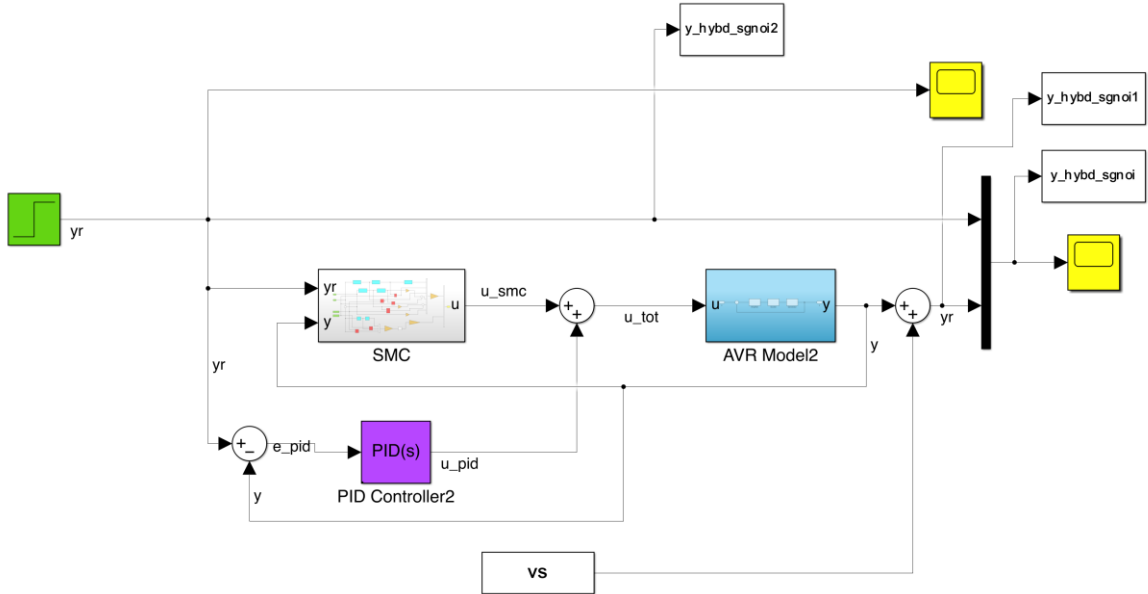
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

State Islamic University of Sultan Syarif Kasim Riau

UN SUSKA RIAU



Rangkaian Simulink AVR-Hybrid dengan Noise 6%

B. Kode Program dan Simulink AVR-Hybrid dengan Perubahan Beban

```
% program model sistem AVR dan Hybrid
% DM x Robby Hibban
% 28 Oktober 2025

% clear histori
clear all; close all; clc;

% time
T_st = 0;
Ts = 0.1;
T_end = 100;
T = T_st : Ts : T_end;

% parameter
Ka = 10; Ke = 1; Kg = 1; Ks = 1;
tau_a = 0.05; tau_e = 1.0; tau_g = 2.0; tau_s = 0.05;
TgTL = 0.002; tgTL = 0.12;
T_L = 0.1;
T_LL = 0.1; T_L2 = 0.28; T_L3 = 0.38;
Vref = 220;
% definisikan SMC
a2 = 13.5; a1 = 37.5; a0 = 25; n_nat = 250; gain_utama = 1/250; % nilai AVR
lamda_1 = 100; lamda_0 = 1; n_sat = -100; ep_slon = 0.1; % parameter SMC

% definisikan PID
Kp = 0.001; Ki = 0.25; Kd = 0.35;

% call simulink
sim('AVR_hybrid_loadchange.slx')

% ===== PLOT UTAMA =====
figure;
ax1 = axes;
```

1. Urutlah ringkasan atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



```

hold(ax1,'on');
set(ax1,'FontSize',12,'LineWidth',1.2);
grid(ax1,'on'); box(ax1,'on');

% warna otomatis
colors = lines(4);

% Plot referensi
plot(T, y_hybdall(:,1), '--', 'Color', colors(1,:), 'LineWidth', 1.3);

% Plot masing-masing TL
plot(T, y_hybd4, 'Color', colors(2,:), 'LineWidth', 1.6); % TL_3
plot(T, y_hybd3, 'Color', colors(4,:), 'LineWidth', 1.6); % TL_2
plot(T, y_hybd2, 'Color', colors(3,:), 'LineWidth', 1.6); % TL_1

xlabel('Waktu (detik)');
ylabel('Tegangan (V)');

legend({'Referensi', 'Output TL_3 = 0.38', 'Output TL_2 = 0.28', 'Output TL_1 = 0.1'}, 'Location', 'southeast', 'FontSize', 11);

%% Info Grafik
signals = {y_hybd4, y_hybd3, y_hybd2};
names = {'Output TL_3', 'Output TL_2', 'Output TL_1'};

r_final = Vref(end);

fprintf('\n=== HASIL STEPINFO UNTUK PERUBAHAN BEBAN (TL Variasi) ===\n');

for i = 1:length(signals)

    y = signals{i};
    info = stepinfo(y, T, r_final);

    % Steady-state error
    ess = abs(r_final - y(end));

    % Time delay (50%)
    half_value = 0.5 * r_final;
    idx_delay = find(y >= half_value, 1);
    if ~isempty(idx_delay)
        Td = T(idx_delay);
    else
        Td = NaN;
    end

    fprintf('\n[%s]\n', names{i});
    fprintf('Peak Time : %.4f s\n', info.PeakTime);
    fprintf('Overshoot : %.4f %\n', info.Overshoot);
    fprintf('Rise Time : %.4f s\n', info.RiseTime);
    fprintf('Settling Time : %.4f s\n', info.SettlingTime);
    fprintf('Time Delay (Td) : %.4f s\n', Td);
    fprintf('Steady-State Err: %.4f V\n', ess);

end

```

Hak Cipta Dilindungi Undang-Undang

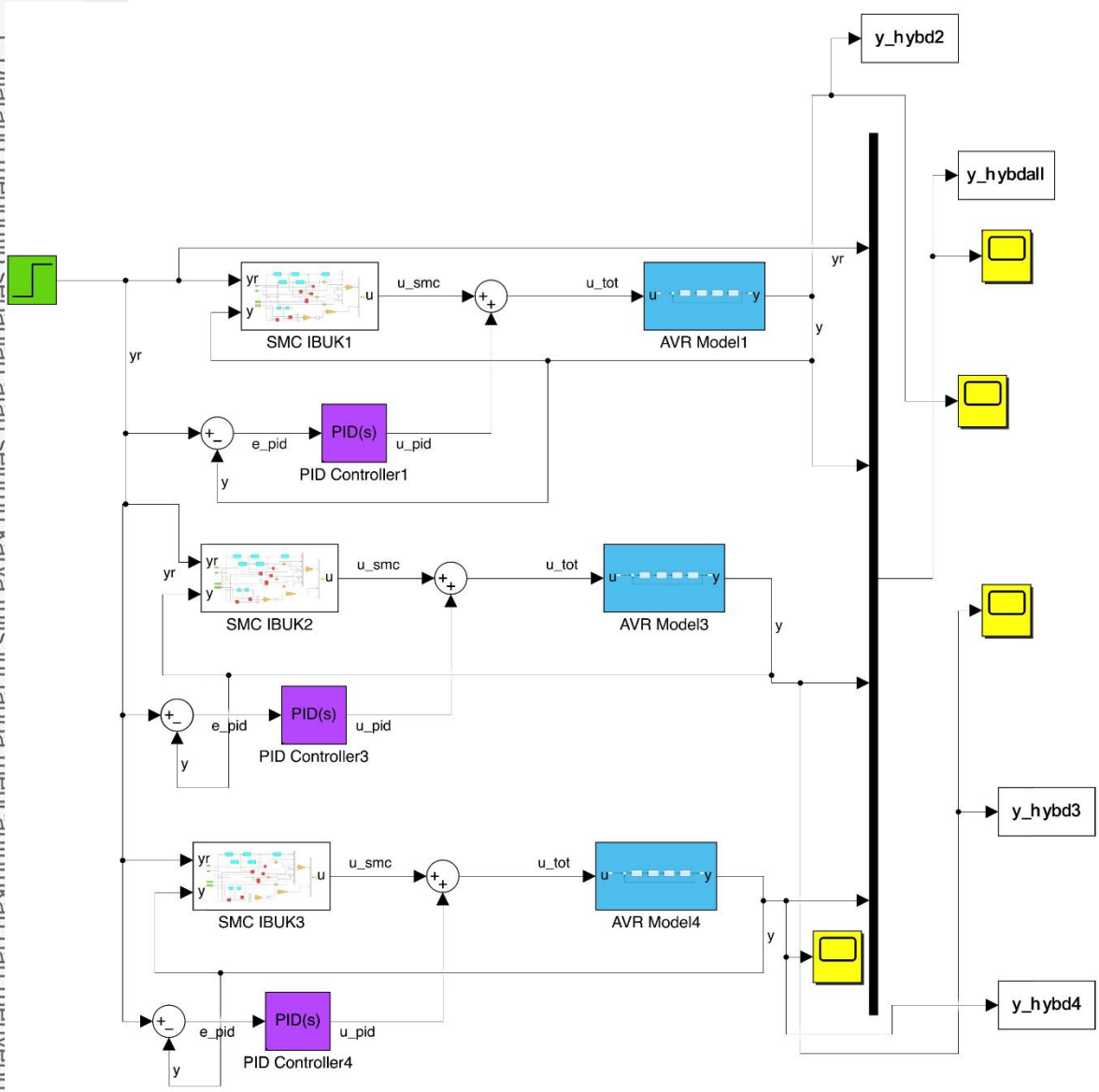
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

1. Urut dari yang terpenting atau seluruh karya tulis ini tanpa meniadakan dan menyuarakan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Rangkaian Simulink AVR-Hybrid dengan *Load Change*

DAFTAR RIWAYAT HIDUP



Sayyid Robbi Al Hibban, lahir di Pekanbaru 03 Mei 2004, ayah bernama Eddy Pariyanto dan nama ibu Niti. Penulis pertama kali menempuh jenjang pendidikan berumur 6 tahun yaitu di SDIT At-Taqwa, waktu itu tahun 2010 dan selesai pada tahun 2016, dan pada tahun yang sama juga penulis melanjutkan sekolah menengah pertama di SMPIT At-Taqwa dan menyelesaikan pendidikannya pada tahun 2019. Pada tahun yang sama juga penulis melanjutkan pendidikan di SMA Plus Taruna Andalan dan waktu SMA saya mengambil jurusan IPA karena sesuai minat penulis. Pada 2022 penulis selesai pendidikan di SMA dan pada tahun yang sama juga penulis diterima di melanjutkan pendidikan ke perguruan tinggi lulus melalui jalur SNMPTN 2022 diprodi pilihan pertama yaitu Teknik Elektro UIN Suska Riau dan sampai sekarang. Pada semester 5, penulis memilih konsentrasi pada bidang Elektronika Instrumentasi. Penulis Berhasil menyelesaikan studi pada tahun 2025.

Atas izin Allah SWT, berkat semangat dan ketekunan penulis, akhirnya dapat menyelesaikan tugas akhir ini. Penulis berharap, karya ini dapat memberikan manfaat serta berguna bagi siapapun yang membutuhkan.

Akhir kata penulis mengucapkan rasa syukur yang sebesar-besarnya kepada Allah SWT atas terselesaikan tugas akhir ini yang berjudul “**Analisis Peran PID terhadap Kinerja Sliding Mode Controller (SMC) pada Sistem Automatic Voltage Regulator (AVR)**”.

No. HP

: 0853-6137-6088

Email

: hibbanaja.1@gmail.com

UIN SUSKA RIAU

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

Hak Cipta

ig

© Hak

Suska Riau

State Islamic University of Sultan Syarif Kasim Riau