

# KOMPUTASI ELEKTRONIK & TENAGA LISTRIK

*by* Suhendar Suhendar

---

**Submission date:** 04-Sep-2022 10:50PM (UTC+0700)

**Submission ID:** 1892335204

**File name:** a3\_Komputasi\_Elektronik.pdf (6.49M)

**Word count:** 36102

**Character count:** 208395



**PENGANTAR**  
**KOMPUTASI**  
**ELEKTRONIK**  
**& TEKNIK TENAGA**

Implementasi Pemrograman Berbantu MATLAB

**SUHENDAR**

**SUHENDAR**

**PENGANTAR KOMPUTASI ELEKTRONIK & TEKNIK TENAGA**  
 Implementasi Pemrograman Berbantu MATLAB



**PENGANTAR**  
**KOMPUTASI**  
**ELEKTRONIK**  
**& TEKNIK TENAGA**

Implementasi Pemrograman Berbantu MATLAB



**M** Indonesiamediaedukasi@gmail.com



087871944890

Jalan Lingkar Caringin, Cisoka Tangerang  
 Banten Kode Pos 15730

ISBN 978-603-403-017-8 (PDF)



9 786234 530179

**PENGANTAR**

# **KOMPUTASI ELEKTRONIK & TENAGA LISTRIK**

Implementasi Pemrograman Berbantu MATLAB

**SUHENDAR**



PENGANTAR

# KOMPUTASI ELEKTRONIK & TENAGA LISTRIK

Impelemntasi Pemrograman Berbantu MATLAB

145

Penulis : SUHENDAR  
ISBN : 978-623-453-017-9  
Editor : Dama Tesniyadi  
Desain Sampul : Tim Desain Media Edukasi  
Layout : Pitriyani

Cetakan Pertama, Agustus 2021  
v + 207 hlm. ; 15 x 23 cm

**Penerbit:**

Media Edukasi Indonesia (Anggota IKAPI)  
Jalan Lingkar Caringin Cisoka Tangerang  
Banten Kode Pos 15730  
Email: [indonesiamediaedukasi@gmail.com](mailto:indonesiamediaedukasi@gmail.com)  
WhatsApp Only: 087871944890

Hak cipta dilindungi oleh Undang-Undang.  
Dilarang mengutip atau memperbanyak sebagian  
atau seluruh isi buku ini dalam bentuk apapun  
juga tanpa izin tertulis dari penerbit.

Syukur Alhamdulillah ke hadirat Allah SWT atas hidayah dan kekuatan yang diberikan-Nya sehingga penulis dapat menyelesaikan buku teks berjudul **Pengantar Komputasi Kendali Elektronik Dan Sistem Tenaga (*Implementasi Pemrograman berbantu MATLAB*)**. Tanpa karunia berupa petunjuk, kekuatan, dan kesehatan yang diberikan oleh Allah SWT, niscaya penulis tidak akan mampu melakukan pengujian dan analisis untuk penelitian lebih lanjut guna menyusun dan menuntaskan buku ini. Buku referensi ini lebih spesifik bertujuan untuk melengkapi kebutuhan literatur pada beberapa matakuliah, yaitu matakuliah Rangkaian Listrik & Elektronik dan mata kuliah Komputasi Sistem Tenaga Listrik di Jurusan Teknik Elektro dan Teknik Metalurgi Fakultas Teknik Universitas Sultan Ageng Tirtayasa (UNTIRTA) Banten. Buku ini ditulis berdasarkan hasil-hasil penelitian dan pengujian-pengujian di laboratorium. Konsep dasar tentang analisis rangkaian listrik dan sistem tenaga yang disajikan dalam buku ini akan memberikan pemahaman kepada para pengguna karena proses analisisnya berbantu sistem komputasi menggunakan perangkat lunak MATLAB.

Para mahasiswa dan dosen dapat menggunakan buku ini sebagai pelengkap referensi lain dalam beberapa matakuliah yang bersesuaian, sedangkan bagi para pengguna umum atau praktisi dapat menggunakan buku ini sebagai bahan untuk meningkatkan pemahaman secara teori dan konsep penggunaan computer untuk membantu menganalisis model rangkaian listrik dan elektronik serta analisis jaringan pada system tenaga listrik. Sebelum mempelajari materi ajar yang ada dalam buku ini, mahasiswa terlebih dahulu harus mengerti konsep dasar rangkaian listrik, rangkaian elektronik, dan analisa sistem tenaga listrik.

Pendekatan yang digunakan dalam buku ini adalah teoritis dan praktis. Pendekatan teoritis dikembangkan guna memberikan pemahaman dasar kepada para pembaca tentang konsep komputasi menggunakan MATLAB. Bagaimana melakukan analisis terhadap rangkaian listrik dan elektronik serta analisis jaringan listrik untuk system tenaga pada tegangan tinggi. Pendekatan teori dan praktis buku ini disajikan dalam beberapa bab yang dibagi menjadi beberapa bagian. Empat bab pertama, yaitu Bab 1 sampai dengan Bab 4.

Untuk lebih memperdalam pemahaman para mahasiswa dan para pembaca, maka dalam buku ini disajikan beberapa contoh analisis dan implementasi model komputasi menggunakan MATLAB. Pada setiap akhir bab dalam buku ini dilengkapi juga dengan pertanyaan ulangan dan soal-soal latihan tambahan untuk diujicobakan lebih lanjut secara langsung oleh para mahasiswa dan para pembaca.

Akhirnya pada kesempatan ini, penulis menghaturkan terima kasih kepada Teman-teman Dosen dan para mahasiswa Jurusan Teknik Elektro UNTIRTA Banten, terima kasih atas dukungannya. Kepada Susrini istriku tersayang dan anak-anakku tercinta: Nazhmi, Naufal, Naila, Sultan, dan Puri Indah yang dengan sabar senantiasa mendorong dan menyemangati Abi untuk menyelesaikan buku ini.

Semoga semua pihak yang telah memberikan bantuan, dukungan, dan do'anya dibalas oleh Allah SWT dengan pahala Surga-Nya, Amin.

Cilegon, Agustus 2021

**Penulis**

## DAFTAR ISI

---

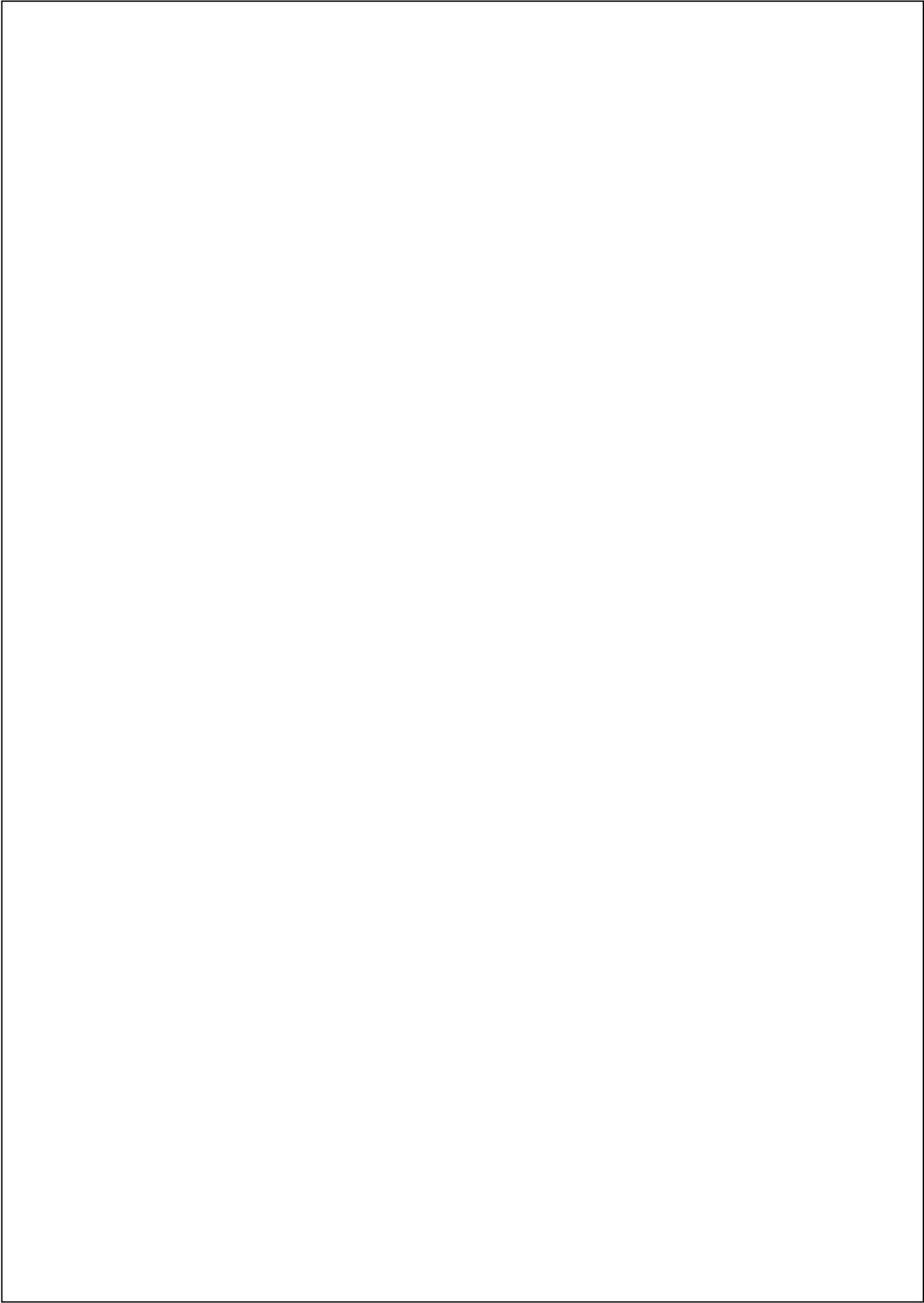
118

<b>KATA PENGANTAR</b> .....	<b>i</b>
<b>DAFTAR ISI</b> .....	<b>iii</b>
<b>BAB 1 PENDAHULUAN</b> .....	<b>1</b>
Tujuan Instruksional .....	1
A. Definisi Komputasi .....	1
B. Teknik Komputasi .....	2
C. Komputasi Berbantu Matlab .....	4
Pertanyaan dan Soal .....	8
<b>BAB 2 PENGANTAR PEMROGRAMAN MATLAB</b> .....	<b>9</b>
Tujuan Instruksional .....	9
A. Lingkungan Kerja dalam Matlab .....	9
B. Memulai Matlab .....	12
C. Operasi Dasar Matlab .....	20
Pertanyaan dan Soal .....	23
<b>BAB 3 PEMROGRAMAN MATRIX DALAM MATLAB</b> .....	<b>24</b>
Tujuan Instruksional .....	24
A. Dasar Matematika pada Matlab .....	24
B. Persamaan Matematika pada Matlab .....	26
C. Implementasi dalam Bentuk Grafik .....	28
D. Array dan Matriks .....	30
E. Pemrograman Matriks pada Matlab .....	40
Pertanyaan dan Soal .....	51
<b>BAB 4 PEMROGRAMAN VISUAL MATLAB</b> .....	<b>52</b>
Tujuan Instruksional .....	52
A. Visualisasi Dasar .....	52
B. Visualisasi 2 Dimensi .....	58

	C. Visualisasi 3 Dimensi .....	60
	Pertanyaan dan Soal.....	64
<b>18</b>	<b>BAB 5 KOMPUTASI KENDALI DASAR .....</b>	<b>65</b>
	Tujuan Instruksional .....	65
	A. Membuka Jendela Simulink .....	65
	B. Menggambar Model .....	66
	C. Penggunaan Simulink dalam Analisa Respon Kendali .....	67
	D. Model Transformasi Laplace.....	68
	E. Respon Transien .....	72
	F. Kesalahan Tunak .....	75
	G. Stabilitas .....	78
	Pertanyaan dan Soal.....	81
	<b>BAB 6 KOMPUTASI KARAKTERISTIK KENDALI DASAR.....</b>	<b>83</b>
	Tujuan Instruksional .....	83
	A. Tempat Kedudukan Akar .....	83
	B. Kontroler P, PI, dan PID.....	90
	Pertanyaan dan Soal.....	97
	<b>BAB 7 KOMPUTASI MODEL RUANG KEADAAN.....</b>	<b>99</b>
	Tujuan Instruksional .....	99
	A. Membentuk Persamaan Ruang Keadaan .....	99
	B. Ketidakunikan Persamaan Keadaan .....	101
	C. Merubah Fungsi Alih Menjadi Ruang Keadaan .....	103
	D. Merubah Persamaan Ruang Keadaan Menjadi Fungsi Alih.....	105
	E. Menggambar Diagram Blok Persamaan Ruang Keadaan dengan Simulink .....	107
	Pertanyaan dan Soal.....	109
	<b>BAB 8 KOMPUTASI ELEKTRONIK .....</b>	<b>110</b>
	Tujuan Instruksional .....	110
	A. Analisis Rangkaian DC .....	112
	B. Analisis Rangkaian Dioda .....	121
	C. Karakteristik Dioda Zener .....	122
	Pertanyaan dan Soal.....	126



<b>BAB 9</b>	<b>PENGANTAR TEORI TENAGA LISTRIK .....</b>	<b>127</b>
	Tujuan Instruksional .....	127
	A. Landasan Teori .....	128
	Pertanyaan dan Soal.....	150
<b>BAB 10</b>	<b>PERANCANGAN KOMPUTASI VISUAL TENAGA LISTRIK ..</b>	<b>151</b>
	Tujuan Instruksional .....	151
	A. Perancangan Tampilan Awal.....	152
	B. Perancangan Karakteristik Motor DC .....	154
	C. Perancangan Karakteristik Motor Ac 1 Fasa.....	158
	D. Perancangan Karakteristik Motor Induksi 3 Fasa.....	163
	E. Perancangan Karakteristik Motor Sinkron .....	164
	F. Diagram Alir Sistem.....	165
	Pertanyaan dan Soal.....	168
<b>BAB 11</b>	<b>KOMPUTASI VISUAL TENAGA LISTRIK.....</b>	<b>169</b>
	Tujuan Instruksional .....	169
	A. Grafik Hasil Pengujian Motor DC.....	169
	B. Grafik Hasil Pengujian Motor Ac Satu Fasa .....	180
	C. Grafik Hasil Pengujian Motor Induksi Tiga Fasa.....	186
	D. Grafik Hasil Pengujian Motor Sinkron .....	191
	Pertanyaan dan Soal.....	193
<b>BAB 12</b>	<b>CONTOH – CONTOH LAIN PROGRAM MATLAB TENAGA LISTRIK .....</b>	<b>195</b>
	Tujuan Instruksional .....	195
	A. Membuat Grafik Karakteristik Torsi-Slip Motor Induksi 3 Fasa ...	195
	B. Perhitungan Arus Gangguan pada Sistem Tenaga Listrik .....	197
	C. Perhitungan Arus Saluran Menggunakan Analisis Mesh dan Komponen Simetris .....	203
	D. Menentukan Daya Optimal Dari Stasiun Pembangkit dan Biaya Total Bahan Bakar Dengan Memperhitungkan Rugi-Rugi Daya Pada Sistem .....	205
	<b>DAFTAR PUSTAKA .....</b>	<b>210</b>



20

**TUJUAN INSTRUKSIONAL:**

*Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:*

- 1) *Mengerti dan menjelaskan kembali tentang definisi dan konsep dasar komputasi*
- 2) *Mengerti dan menjelaskan kembali tentang teknik komputasi*
- 3) *Mengerti penggunaan MATLAB untuk mengimplementasikan proses komputasi*
- 4) *Mengerti fungsi-fungsi utama pada tampilan MATLAB*

97

**A. DEFINISI KOMPUTASI**

Komputasi sebetulnya bisa diartikan sebagai cara untuk menemukan pemecahan masalah dari data input dengan menggunakan suatu algoritma. Lahirnya proses komputasi berasal dari kegiatan hitung menghitung yang dilakukan manusia sejak berabad-abad yang lalu sejak manusia mengenal angka dan perhitungan seperti melakukan perdagangan dengan sistem bertukar (barter), sistem kalender menghitung tanggal, hari, bulan dan tahun, dan perhitungan lainnya dengan menggunakan alat hitung seperti kayu, batu, pena dan lain-lain. Seiring dengan perkembangan zaman, proses perhitungan tidak lagi se-simple bertahun-tahun yang lalu. Zaman berkembang begitu pula dengan perhitungan. Jika berbicara mengenai perhitungan, pastilah kita memerlukan data apa saja yang akan kita hitung. Dengan perkembangan zaman, data yang dihitung juga berkembang, tidak stack pada satu jenis data saja. Bisa lebih dari beberapa data, puluhan, ratusan bahkan ribuan. Hal demikianlah yang membuat lahirnya sebuah cara yang disebut Komputasi Modern yang dapat digunakan untuk melakukan atau memecahkan perhitungan yang banyak dan rumit dengan alat menggunakan komputasi modern yakni komputer. Namun bukan berarti komputer hanya bisa menyelesaikan proses perhitungan saja, komputer juga memiliki kemampuan untuk menyelesaikan banyak tugas manusia seperti menulis, mencari informasi, dan masih banyak lagi.

Komputasi juga sering diartikan sebagai sebuah komputer secara fisik. Sebagai contoh dari sistem fisik yaitu komputer digital, komputer *quantum*, komputer penganalisa DNA, dan komputer molekular. Sudut pandang ini dipelajari

di cabang ilmu teori fisik yang disebut *Physic of Computation*. Bahkan ada sudut pandang yang lebih radikal berbasis dalil *Digital Physic* yang menyatakan bahwa evolusi alam semesta itu sendiri adalah sebuah proses komputasi – disebut *Pancomputationalism*.

Ilmu atau sains berdasarkan obyek kajiannya dibedakan antara Fisika, Kimia, Biologi dan Geologi. Ilmu dapat pula digolongkan berdasarkan metodologi dominan yang digunakannya, yaitu ilmu pengamatan/percobaan (*observational/experimental science*), ilmu teori (*theoretical science*) dan ilmu komputasi (*computational science*). Yang terakhir ini bisa dianggap bentuk yang paling baru yang muncul bersamaan dengan perkembangan kekuatan pemrosesan dalam komputer dan perkembangan teknik-teknik metode numerik dan metode komputasi lainnya. Dalam ilmu (sains) tradisional seperti Fisika, Kimia dan Biologi, penggolongan ilmu berdasarkan *metodologi dominannya* juga mawujud, yang ditunjukkan dengan munculnya bidang-bidang khusus berdasarkan penggolongan tsb. lengkap dengan jurnal-jurnal yang relevan untuk melaporkan hasil-hasil penelitiannya.

Sebagai contoh dalam kimia, melengkapi kimia percobaan (*experimental chemistry*) dan kimia teori (*theoretical chemistry*), berkembang pula kimia komputasi (*computational chemistry*), seperti juga di bidang Biologi dikenal Biologi Teori (*theoretical biology*) serta Biologi Komputasi (*computational biology*), lengkap dengan jurnalnya seperti *Journal of Computational Chemistry* dan *Journal of Computational Biology*. Cara penggolongan yang digunakan berbeda dengan cara penggolongan lain berdasarkan obyek kajian, seperti penggolongan kimia atas Kimia Organik, Kimia Anorganik, dan Biokimia. Walaupun dengan titik pandang yang berbeda, ilmu komputasi sebagai bentuk ketiga dari ilmu (sains) telah banyak disampaikan oleh berbagai pihak.

Stephen Wolfram dan bukunya yang terkenal: *A New Kind of Science*, dan Jürgen Schimhuber. *Implementasi Komputasi di Berbagai Bidang Ilmu Pengetahuan*, diantaranya Fisika, Kimia, Matematika, Ekonomi, Biologi, Geografi, dan lain-lain.

## B. TEKNIK KOMPUTASI

Jika ditinjau dari penggunaan alat, metode, dan teori bahwa Komputasi adalah kegiatan mendapatkan penyelesaian atau *solusi* atas persoalan yang dinyatakan dalam model matematis. Secara matematis pada umumnya model mengambil bentuk  $f(x) = y$ , dengan  $x = \text{himpunan informasi}$  yang tersembunyi dalam model, berupa besaran-besaran yang nilainya harus ditetapkan agar persoalan nyata dapat dipecahkan,  $y = \text{himpunan data}$  yang tersedia, berupa besaran-besaran yang nilainya telah diketahui, dan  $f(.) = \text{operator matematis model tersebut}$ . Secara singkat dalam komputasi diberikan  $f(.)$  serta nilai numeris  $y$ , lakukanlah aktivitas

untuk memperoleh nilai numeris  $x$ , agar  $f(x) = y$  memenuhi. Secara matematis,  $x$  diperoleh melalui operasi invers atas  $y$ . Konkritnya:  $x = f^{-1}(y)$ , dengan  $f^{-1}$  operator matematis untuk melaksanakan operasi invers yang dimaksudkan. Masalah utama: dalam praktek tidak banyak operator  $f$  dan  $f^{-1}$  diketahui atau langsung dapat ditetapkan dengan mudah. Oleh karena itu proses komputasi sering harus melalui jalan yang tak langsung.

Teknik komputasi adalah perangkat ilmu tentang alat (biasanya sebuah komputer), metode (yang disebut algoritma) dan teori (bukti matematis bahwa komputasi memberi hasil yang benar) yang diperlukan untuk melaksanakan komputasi tersebut. Sementara itu dalam melakukan kegiatan komputasi untuk menyelesaikan suatu persoalan, seorang teknisi harus memperhatikan interaksi dari alat (komputer yang digunakan), metode (yaitu program yang dimiliki), dan sifat unik dari soal yang dihadapi, sebab dalam praktek soal-soal memiliki tingkat kesulitan yang berbeda-beda: ada soal yang relatif sangat gampang, ada yang sulit, tetapi juga ada soal yang sangat sulit.

Alat komputasi paling kuno adalah kertas, potlot dan karet penghapus. Alat komputasi yang lebih maju adalah kertas, potlot, karet penghapus dan mistar hitung. Selanjutnya mistar diganti dengan kalkulator (elektronis), yang dapat dibawa di dalam saku. Alat komputasi yang modern adalah komputer, yang biasanya merupakan peralatan elektronis dengan berbagai kemampuan tertentu: (1) dapat melakukan operasi penyimpanan, karena dalam komputer ada memori; (2) dapat melakukan operasi-operasi tertentu atas yang disimpan dalam memori; (3) dapat menyajikan kembali isi memori itu dalam media penampil menurut format yang dikehendaki oleh pemakai. Itu adalah rumusan singkat tentang apa yang disebut komputer. Beberapa hal masih harus ditambahkan agar sebuah gambaran yang utuh diperoleh. Dalam bidang komputer dikenal tiga serangkai *data - program - informasi*.

Dalam bahasa sehari-hari data adalah fakta tersurat (dalam bentuk catatan atau tulisan) tentang suatu obyek. Dalam dunia komputer data adalah segala sesuatu yang dapat disimpan dalam memori menurut format tertentu. Informasi adalah segala sesuatu yang ditampilkan oleh komputer dalam sebuah media penampil tersebut diatas, biasanya sebagai hasil dari sebuah proses komputasi. Bagi teknisi yang berhadapan langsung dengan persoalan yang nyata, data menghadirkan fakta tersurat yang secara spesifik ada pada persoalan itu, dan yang secara khas terpancar sebagai isyarat sebagaimana diterima (ditangkap) oleh teknisi yang sedang menangani persoalan tersebut. Dalam konteks itu informasi adalah fakta tersirat yang muncul dalam benak teknisi itu sesudah mencermati dan mengolahnya dengan tertib, berdasarkan model yang diyakini sebagai hal yang benar ada dalam keseluruhan persoalan tersebut.

Program adalah deretan operasi yang sengaja ditulis untuk sebuah proses komputasi. Program adalah resep tentang bagaimana komputasi itu harus

dilaksanakan. Sebagai sebuah fakta tentang obyek komputasi, program disimpan dalam memori komputer untuk dijalankan. -- Menjalankan komputer adalah membuat komputer melaksanakan tiap operasi yang terdapat dalam program, satu demi satu, dari operasi pertama, kedua, ..., dan seterusnya sampai dengan operasi yang terakhir. Maka operasi sama dengan *instruksi* dan himpunan instruksi yang dimiliki atau dikenal oleh komputer merupakan *bahasa komputer*, karena dengan bahasa itu manusia dapat berkomunikasi dan menyampaikan kehendaknya kepada komputer.

Informasi <sup>26</sup> adalah fakta tersirat (dalam bentuk catatan atau tulisan) tentang obyek yang dibicarakan. Disini (sekali lagi) sengaja digunakan istilah <sup>26</sup> "tersirat" (dan dipertentangkan dengan kata "tersurat") untuk menegaskan bahwa informasi adalah fakta tersembunyi dibalik himpunan fakta yang sudah dicatat, dan baru diketemukan sesudah diolah atau dicerna. Maka informasi identik dengan data. Perbedaan pokok adalah dalam hubungannya dengan pemakai data/informasi tersebut. Data adalah mentah, karena hasil harus dicerna lagi, informasi bersifat matang karena langsung dapat digunakan (misalnya untuk pengambilan keputusan).

## C. KOMPUTASI BERBANTU MATLAB

### 1. Proses Instalasi Matlab

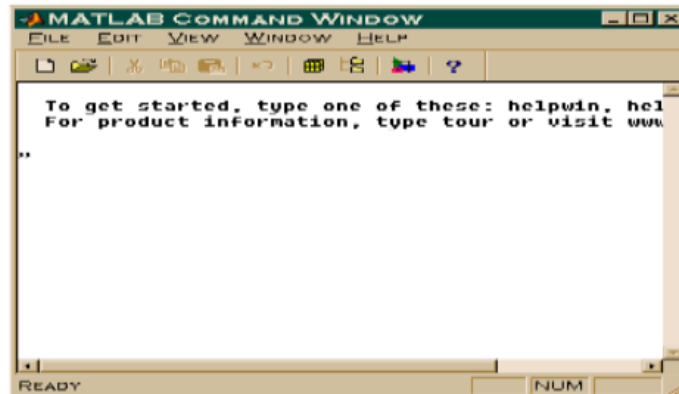
Orang yang biasa menginstall tidak akan mengalami kesulitan dengan menginstal MATLAB kecuali bila sistem operasi yang digunakan LINUX. Untuk pengguna LINUX, tata cara instalasinya dapat dilihat di website <http://www.mathworks.com> yang merupakan website resmi MATLAB.

Pemakai prosesor pentium III ke atas tidak akan menemui kesulitan dalam menginstall karena mampu mensupport MATLAB R-14 versi 7. Sedangkan untuk pemakai prosesor di bawahnya misalnya pentium I, harus mencari versi MATLAB yang sesuai misalnya MATLAB R-11 versi 5.3 yang sudah memadai karena didalamnya terdapat paket SIMULINK. MATLAB dapat dijalankan pada semua Power Macintosh (dengan mikroprosesor 68020/68030/68040 atau 68881/68882) tetapi tidak bisa jalan pada Macintosh dengan mikroprosesor 68LC040.

Penulis menyarankan menginstall paket MATLAB secara keseluruhan. Tetapi bila kapasitas hardisk-nya kecil, saat memilih peralatan yang akan diinstall, pilihlah yang penting-penting saja dengan mengklik pada kotak pilihan. File help yang berbentuk pdf dan html serta MATLAB untuk server bagi Anda yang sudah mahir mungkin tidak dibutuhkan.

## 2. Command Window

Saat program MATLAB dijalankan dan setelah muncul secara singkat simbol MATLAB (grafik membran-L) yang Anda lihat sekarang adalah lembar kerja utama kita yang disebut command window. Untuk pengguna Macintosh, tampilannya tidak jauh berbeda.

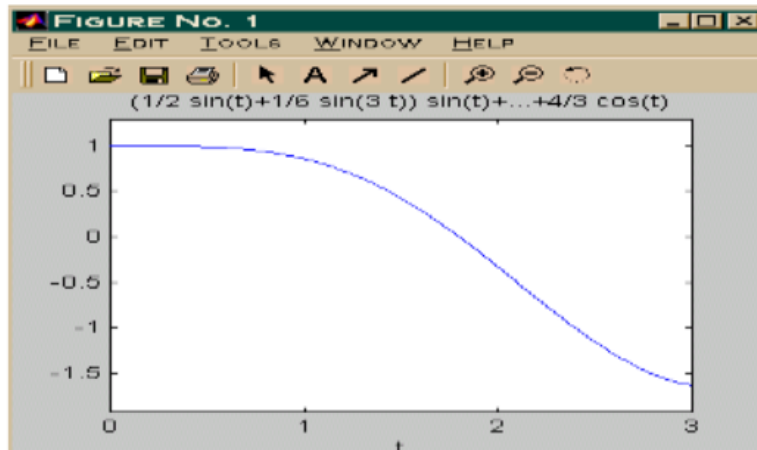


**Gambar 1.1** Command Window

Pada lembar kerja (*Workspace*) ini terdapat *toolbox* dan layar putih dengan tanda `>>` di sebelah kirinya yang siap diberi perintah. Pada lembar kerja ini, MATLAB siap menerima instruksi, baik instruksi langsung seperti pada alat hitung/kalkulator dimana jawabannya langsung diperoleh atau instruksi yang berupa program seperti layaknya bahasa pemrograman tingkat tinggi.

## 3. Grafik/Figure

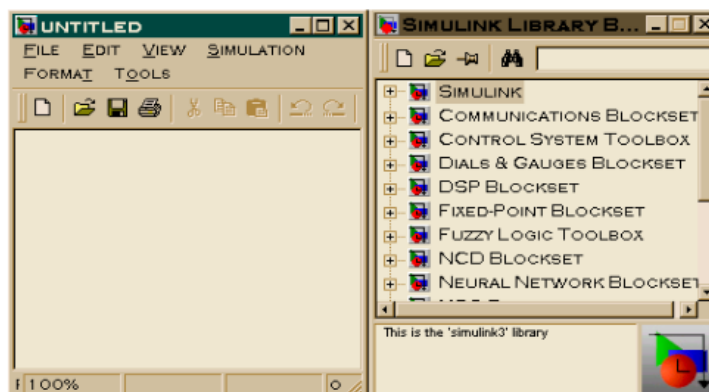
Saat anda memberi instruksi membuat grafik/plot akan muncul jendela grafik/figure pada layar komputer anda. Atau Anda dapat membuka grafik lama yang tersimpan dengan mengklik File – New – Figure. File grafik yang tersimpan berekstensi FIG.



Gambar 1.2 Jendela Grafik

#### 4. Jendela Model/Simulink

Disamping command window yang formatnya seperti sistem operasi DOS, pada MATLAB disertakan paket SIMULINK yang berbentuk seperti pada gambar di bawah ini, dapat anda buka dengan mengklik file – new – model.



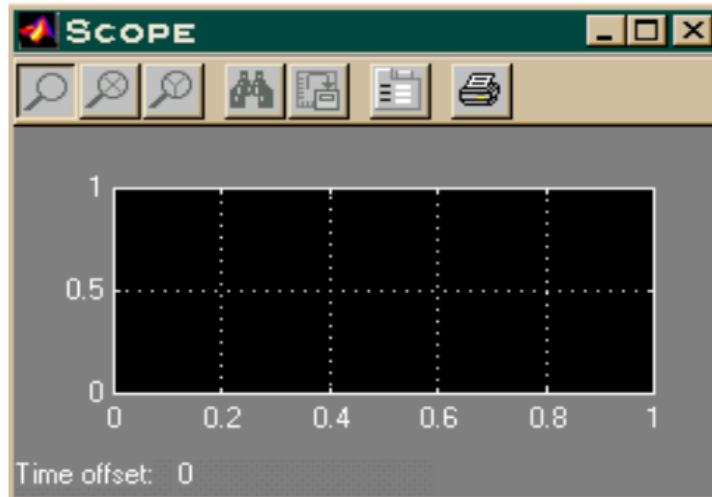
Gambar 1.3 Jendela Model / SIMULINK

Kita dapat menggambar model dengan cara klik dan drag model yang ada di Simulink Library Browser ke jendela Model.

#### 5. Jendela Simulasi

Jendela yang masih merupakan bagian dari SIMULINK ini menggambarkan respon sistem sesuai gambar model.





**Gambar 1.4** Jendela SCOPE pada SIMULINK

## 6. Jendela M-File

Jendela M-File adalah jendela yang dipergunakan untuk pemrograman. Cara memasukinya adalah dengan klik **file – New – M-File**. Program disimpan dalam ekstensi m dan cdr.

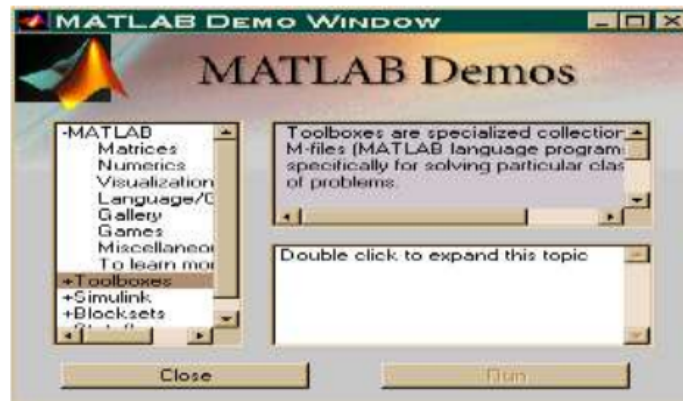


**Gambar 1.5** Jendela M-File

## 7. Matlab Demos

Dalam menggambar grafik maupun simulasi kadang kala kita lupa atau belum mengenal aturan penulisannya. Sedangkan [185](#)ku yang secara lengkap membahas seluruh fasilitas yang ada pada MATLAB sangat sulit. Maka Anda tidak perlu khawatir karena di MATLAB tersedia suatu fasilitas pembantu yang

dinamakan **MATLAB DEMOS**. Silahkan Anda ketik demo pada command window, maka akan tampak jendela seperti di bawah ini.



**Gambar 1.6** Jendela MATLAB DEMO

## **PERTANYAAN DAN SOAL**

---

1. Jelaskan kembali tentang definisi komputasi dan beberapa pengelompokkannya!!
2. Jelaskan dan berikan beberapa contoh tentang teknik komputasi?
3. Bagaimana MATLAB dapat digunakan untuk membantu proses komputasi?
4. Sebutkan beberapa fungsi utama pada windows MATLAB yang umum digunakan untuk proses komputasi!!

## PENGANTAR PEMROGRAMAN MATLAB

---

20

### TUJUAN INSTRUKSIONAL:

Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:

- 1) Mengetahui lingkungan kerja salam MATLAB
  - 2) Dapat memulia membuka MATLAB
  - 3) Mampu membedakan beberapa windows MATLAB
  - 4) Mengerti dan mengujicobakan beberapa contoh sintaks dasar pada MATLAB
- 

### A. LINGKUNGAN KERJA DALAM MATLAB

9

MATLAB merupakan bahasa pemrograman yang hadir dengan fungsi dan karakteristik yang berbeda dengan bahasa pemrograman lain yang sudah ada lebih dahulu seperti Delphi, Basic maupun C++. MATLAB merupakan bahasa pemrograman level tinggi yang dikhususkan untuk kebutuhan komputasi teknis, visualisasi dan pemrograman seperti komputasi matematik, analisis data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan. Matlab hadir dengan membawa warna yang berbeda. Hal ini karena matlab membawa keistimewaan dalam fungsi-fungsi matematika, fisika, statistik, dan visualisasi. Matlab dikembangkan oleh MathWorks, yang pada awalnya dibuat untuk memberikan kemudahan mengakses data matrik pada proyek LINPACK dan EISPACK. Saat ini matlab memiliki ratusan

fungsi yang dapat digunakan sebagai *problem solver* mulai dari simple sampai masalah-masalah yang kompleks dari berbagai disiplin ilmu.

15

Beberapa Lingkungan Kerja Matlab yang merupakan bagian dari Window MATLAB, diantaranya:

#### a. Current Directory

Window ini menampilkan isi dari direktori kerja saat menggunakan matlab. Kita dapat mengganti direktori ini sesuai dengan tempat direktori kerja yang diinginkan. Default dari alamat direktori berada dalam folder works tempat program files Matlab berada.

29

## b. Command History

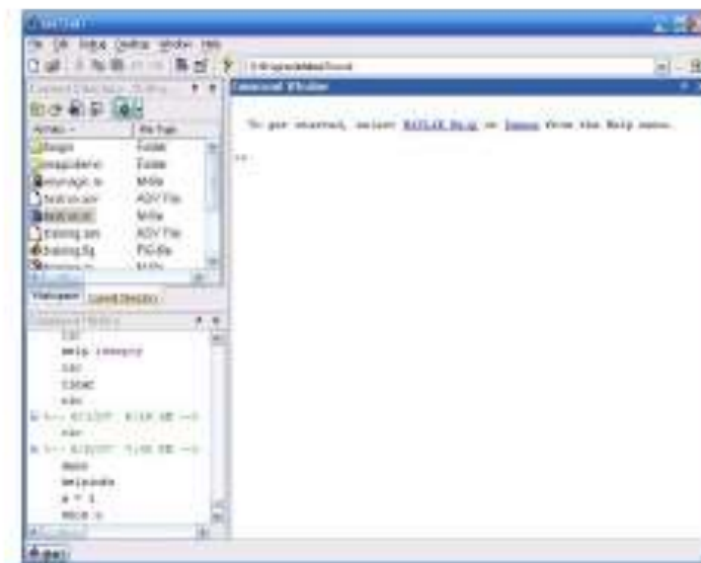
Window ini berfungsi untuk menyimpan perintah-perintah apa saja yang sebelumnya dilakukan oleh pengguna terhadap matlab.

## c. Command Window

Window ini adalah window utama dari Matlab. Disini adalah tempat untuk menjalankan fungsi, mendeklarasikan variable, menjalankan proses-proses, serta melihat isi variable.

## 1. Workspace

Workspace berfungsi untuk menampilkan seluruh variabel-variabel yang sedang aktif pada saat pemakaian matlab. Apabila variabel berupa data matriks berukuran besar maka user dapat melihat isi dari seluruh data dengan melakukan double klik pada variabel tersebut. MATLAB secara otomatis akan menampilkan window “array editor” yang berisikan data pada setiap variabel yang dipilih user. Gambar berikut menampilkan tampilan antar muka dari matlab versi 7.0.



Gambar 2.1 Windows MATLAB

9

## 2. Getting Help

Matlab menyediakan fungsi help yang tidak berisikan tutorial lengkap mengenai Matlab dan segala keunggulannya. User dapat menjalankan fungsi ini dengan menekan tombol pada toolbar atau menulis perintah ‘**helpwin**’ pada

command window. Matlab juga menyediakan fungsi demos yang berisikan video tutorial matlab serta contoh-contoh program yang bias dibuat dengan matlab.

### 3. Interrupting dan Terminating dalam Matlab

Untuk menghentikan proses yang sedang berjalan pada matlab dapat dilakukan dengan menekan tombol **Ctrl-C**. Sedangkan untuk keluar dari matlab dapat dilakukan dengan menuliskan perintah **exit** atau **quit** pada comamnd window atau dengan menekan menu exit pada bagian menu file dari menu bar.

Sebagai sebuah system, MATLAB tersusun dari 5 bagian utama:

#### a. Development Environment

Merupakan sekumpulan perangkat dan fasilitas yang membantu anda untuk menggunakan fungsi-fungsi dan file-file MATLAB. Beberapa perangkat ini merupakan sebuah graphical user interfaces (GUI). Termasuk didalamnya adalah MATLAB desktop dan Command Window, command history, sebuah editor dan debugger, dan browsers untuk melihat help, workspace, files, dan search path.

#### b. MATLAB Mathematical Function Library

Merupakan sekumpulan algoritma komputasi mulai dari fungsi-fungsi dasar seperti: sum, sin, cos, dan complex arithmetic, sampai dengan fungsi-fungsi yang lebih kompleks seperti matrix inverse, matrix eigenvalues, Bessel functions, dan fast Fourier transforms.

#### c. MATLAB Language

Merupakan suatu high-level matrix/array language dengan control flow statements, functions, data structures, input/output, dan fitur-fitur object-oriented programming. Ini memungkinkan bagi kita untuk melakukan "pemrograman dalam lingkup sederhana " untuk mendapatkan hasil yang cepat, dan "pemrograman dalam lingkup yang lebih besar" untuk memperoleh hasil-hasil dan aplikasi yang kompleks.

#### d. Graphics

MATLAB memiliki fasilitas untuk menampilkan vector dan matrices sebagai suatu grafik. Didalamnya melibatkan high-level functions (fungsi-fungsi level tinggi) untuk visualisasi data dua dikensi dan data tiga dimensi, image processing, animation, dan presentation graphics. Ini juga melibatkan fungsi level rendah yang memungkinkan bagi anda untuk membiasakan diri

untuk memunculkan grafik mulai dari bentuk yang sederhana sampai dengan tingkatan graphical user interfaces pada aplikasi MATLAB anda.

2

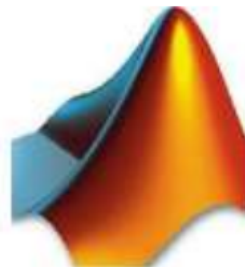
#### e. MATLAB Application Program Interface (API)

Merupakan suatu library yang memungkinkan program yang telah anda tulis dalam bahasa C dan Fortran mampu berinteraksi dengan MATLAB. Ini melibatkan fasilitas untuk pemanggilan routines dari MATLAB (dynamic linking), pemanggilan MATLAB sebagai sebuah computational engine, dan untuk membaca dan menuliskan MAT-files.

## B. MEMULAI MATLAB

126

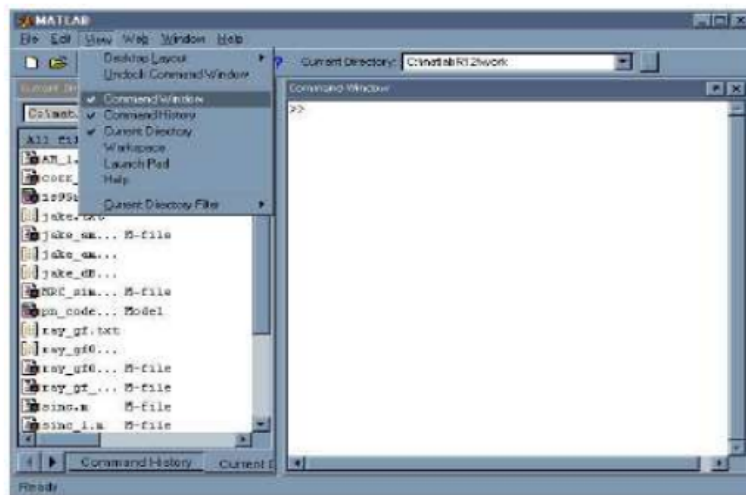
Perhatikan Dekstop pada layar monitor PC, anda mulai MATLAB dengan melakukan *double-clicking* pada *shortcut icon* MATLAB.



Gambar 2.1 Icon MATLAB pada desktop PC

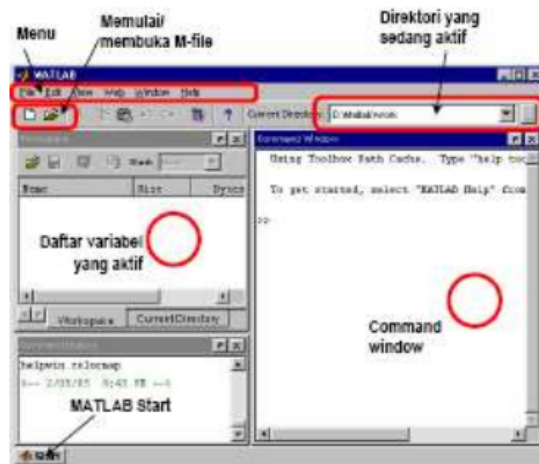
34

Selanjutnya anda akan mendapatkan tampilan seperti pada Gambar berikut ini.



Gambar 2.2 Tampilan awal Matlab

Sedangkan untuk mengakhiri sebuah sesi MATLAB, anda bisa melakukan dengan dua cara, pertama pilih **File -> Exit MATLAB** dalam window utama MATLAB yang sedang aktif, atau cara <sup>18</sup>dua lebih mudah yaitu cukup ketikkan type **quit** dalam Command Window. Mengeksekusi ikon MATLAB di layar komputer ataupun melalui tombol **Start** di Windows. Setelah proses *loading* program, jendela utama MATLAB akan muncul seperti berikut ini.



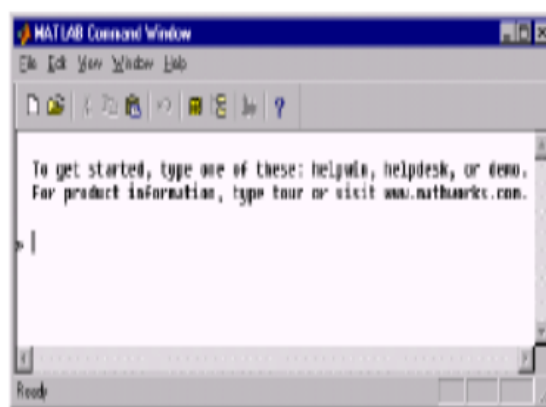
**Gambar 2.3** Jendela Utama MATLAB

## 1. Beberapa Windows Pada MATLAB

Ada beberapa macam window yang tersedia dalam MATLAB, yang dapat dijelaskan sebagai berikut:

### a. MATLAB Command window/editor

MATLAB Command window/editor merupakan window yang dibuka pertama kali setiap kali MATLAB dijalankan.



**Gambar 2.3** MATLAB Command Windows

MATLAB dengan cara mengetikkan barisan-barisan ekspresi MATLAB, seperti mengakses help window dan lain-lainnya. Jika perintah-perintah yang sudah diketikkan dan hasil yang ditampilkan pada layar command window akan disimpan maka dapat dilakukan dengan menggunakan command diary. Sebagai contoh jika ingin disimpan output

```
m =
```

```
1 2 3
```

```
5 2 1
```

direktori c:\backup dengan nama file mydata.txt, maka dapat dilakukan dengan mengetikkan :

```
» m = [1 2 3;5 2 1]
```

```
m =
```

```
1 2 3
```

```
5 2 1
```

```
» diary 'c:\backup\mydata.txt'
```

dan jika ingin menutup file ini maka ketikkan

```
» diary off
```

96

Command windows juga digunakan untuk memanggil tool Matlab seperti editor, debugger atau fungsi. Ciri dari window ini adalah adanya prompt (>>) yang menyatakan matlab siap menerima perintah. Perintah dapat berupa fungsi-fungsi pengaturan file (seperti perintah DOS/UNIX) maupun fungsifungsi bawaan/toolbox MATLAB sendiri. Berikut ini beberapa fungsi pengaturan file dalam MATLAB :

*dir / ls* : Digunakan untuk melihat isi dari sebuah direktori aktif.

*cd* : Digunakan untuk melakukan perpindahan dari direktori aktif.

*pwd* : Digunakan untuk melihat direktori yang sedang aktif

*mkdir* : Digunakan untuk membuat sebuah direktori

*what* : Digunakan untuk melihat nama file m dalam direktori aktif

*who* : Digunakan untuk melihat variabel yang sedang aktif

*whos* : Digunakan untuk menampilkan nama setiap variabel

*delete* : Digunakan untuk menghapus file

*clear* : Digunakan untuk menghapus variabel

*clc* : Digunakan untuk membersihkan layar

*doc* : Digunakan untuk melihat dokumentasi The MathWorks, Inc. dalam format html secara online.

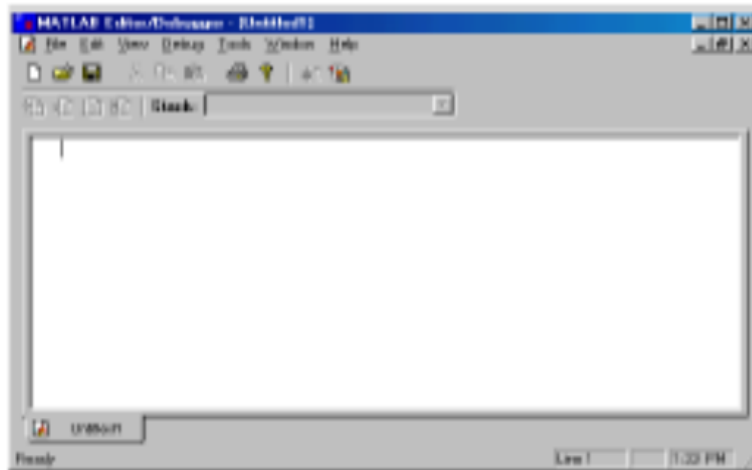
*demo* : Digunakan untuk mencoba beberapa tampilan demo yang disediakan oleh Matlab



## b. MATLAB Editor/Debugger (Editor M-File/Pencarian Kesalahan)

Window ini merupakan tool yang disediakan oleh Matlab 5 keatas. Berfungsi sebagai editor script Matlab (M-file). Walaupun sebenarnya script ini untuk pemrograman Matlab dapat saja menggunakan editor yang lain seperti notepad, wordpad bahkan word. Untuk mengakses window m-file ini dapat dilakukan dengan cara:

- 1) Memilih File kemudian pilih New
- 2) Pilih m-file, maka MATLAB akan menampilkan editor window:



Gambar 2.4 MATLAB Editor

selain dengan cara di atas untuk menampilkan editor MFile ini dapat juga dilakukan dengan :

```
>> edit
```

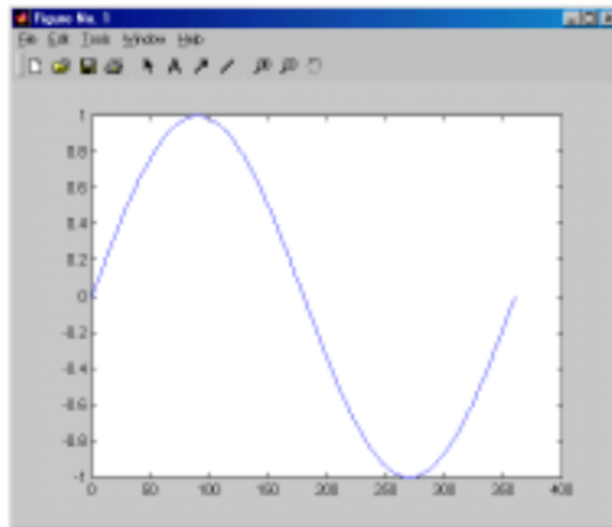
## c. Figure Windows

Window ini adalah hasil visualisasi dari script Matlab. Namun Matlab memberi kemudahan bagi programmer untuk mengedit window ini sekaligus memberikan program khusus untuk itu. Sehingga window ini selain berfungsi sebagai visualisasi output dapat juga sekaligus menjadi media input yang interaktif.

**Contoh :**

```
>> x=0:2:360;  
>> y=sin(x*pi/180);  
>> plot(x,y)
```

hasil yang diperoleh adalah :



Gambar 2.5 Figure Windows

58

#### d. MATLAB help window

MATLAB menyediakan sistem *help* yang dapat diakses dengan perintah *help*. Misalnya, untuk memperoleh informasi mengenai fungsi *elfun* yaitu fungsi untuk trigonometri, eksponensial, complex dan lain-lain, maka hanya perlu mengetikkan perintah berikut :

» help elfun

dan kemudian menekan enter maka di layar akan muncul informasi dalam bentuk teks pada layar MATLAB yaitu: Elementary math functions.

#### Trigonometric

sin	- Sine.
sinh	- Hyperbolic sine.
asin	- Inverse sine.
Asinh	- Inverse hyperbolic sine.
cos	- Cosine.
Cosh	- Hyperbolic cosine.
acos	- Inverse cosine.
acosh	- Inverse hyperbolic cosine.
tan	- Tangent.
tanh	- Hyperbolic tangent.
atan	- Inverse tangent.

atan2	- Four quadrant inverse tangent.
atanh	- Inverse hyperbolic tangent.
sec	- Secant.
sech	- Hyperbolic secant.
asec	- Inverse secant.
asech	- Inverse hyperbolic secant.
csc	- Cosecant.
csch	- Hyperbolic cosecant.
acsc	- Inverse cosecant.
acsch	- Inverse hyperbolic cosecant.
cot	- Cotangent.
coth	- Hyperbolic cotangent.
acot	- Inverse cotangent.
acoth	- Inverse hyperbolic cotangent.

### Exponential

exp	- Exponential.
Log	- Natural logarithm.
log10	- Common (base 10) logarithm.
log2	- Base 2 logarithm and dissect floating

### Point Number

pow2	- Base 2 power and scale floating point number.
sqrt	- Square root.
nextpow2	- Next higher power of 2.

### Complex

abs	- Absolute value.
angle	- Phase angle.
complex	- Construct complex data from real and

### Imaginary Parts

conj	- Complex conjugate.
imag	- Complex imaginary part.
real	- Complex real part.
unwrap	- Unwrap phase angle.
isreal	- True for real array.

`cplxpair` - Sort numbers into complex conjugate pairs.

### Rounding and remainder

- `fix` - Round towards zero.
- `floor` - Round towards minus infinity.
- `Ceil` - Round towards plus infinity.
- `round` - Round towards nearest integer.
- `mod` - Modulus (signed remainder after division).
- `rem` - Remainder after division.
- `sign` - Signum.

Se<sup>58</sup> `help` untuk informasi di atas dapat juga dilihat informasi lainnya misalnya perintah yang sangat berguna untuk mempelajari pemrograman MATLAB adalah `intro`, yang membahas konsep-konsep dasar tentang bahasa MATLAB. Selain itu, juga terdapat banyak program demonstrasi yang mengilustrasikan berbagai kapabilitas MATLAB, yang dapat dimulai dengan perintah `demo`. Atau untuk lebih lengkapnya dapat dilihat di tampilan MATLAB dengan cara memilih menu Window kemudian pilih help window, dan untuk mengetahui informasi yang ada maka dapat dilakukan dengan mengklik dua kali info yang ada di MATLAB Help Window atau dengan mengetikkan informasi yang ingin didapatkan pada sudut sebelah kiri MATLAB Help Window. Sebagai contoh tampilan Help window tersebut adalah sebagai berikut:



Gambar 2.6 MATLAB Help Window

Sebelum melihat lebih jauh tentang kemampuan MATLAB untuk programming ataupun penyelesaian masalah-masalah numerik maka dapat dilakukan beberapa percobaan sederhana seperti berikut ini:

**Ketikkan :**

```
» x=12
» x=6
» x*y
```

```
ans =
      72
```

dan perhatikan perbedaannya dengan :

```
» p=12;
» q=6;
» r=p*q;
```

dapat dilihat bahwa **tanpa ;** MATLAB akan mengassign hasil output ke variabel ans (answer).

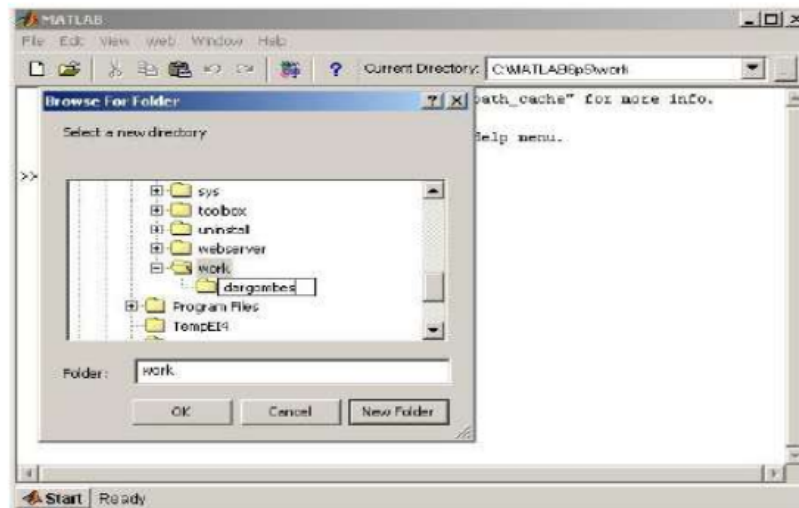
Untuk contoh penggunaan fasilitas grafik dari MATLAB, maka dapat dilihat berikut ini dengan mengetikkan :

```
» x=[5 10 15 20 25];
» y=cos(x);
» plot(x,y);
```

dapat dijelaskan bahwa x diassign ke suatu array [5 10 15 20 25], kemudian y =cos(x) akan mengassign y ke array [cos(5) cos(10) cos(15) cos(20) cos(25)].

## 2. Menentukan Direktori Tempat Bekerja

Anda dapat bekerja dengan MATLAB secara default pada directory Work ada di dalam Folder MATLAB. Tetapi akan lebih bagus dan rapi jika anda membuat satu directory khusus dengan nama yang sudah anda kuskan, “dargombes” atau nama yang lain yang mudah untuk diingat. Hal ini akan lebih baik bagi anda untuk membiasakan bekerja secara rapi dan tidak mencampur program yang anda buat dengan program orang lain. Untuk itu Arahkan pointer mouse anda pada kotak bertanda ... yang ada disebelah kanan tanda panah kebawah (yang menunjukkan folder yang sedang aktif). Pilih new directory, selanjutnya ketikkan “dargombes”, dan diikuti dengan click Ok.



**Gambar 2.7** Membuat Folder baru tempat program

### C. OPERASI DASAR MATLAB

**MATLAB** mempunyai berbagai fungsi matematika umum yang biasa digunakan dalam matematik . Sebagian besar fungsi tersebut hampir sama dengan bila anda menuliskannya secara matematis. Sebagai contoh:

```
pi
ans =
    3.1416
```

```
y=sin(pi/6)
y =
    0.5000
```

```
y=asin(0.5)
y =
    0.5236
```

130

**Contoh 1: Persamaan fungsi**

$$M=4x^2+3y+10$$

$$N=e(2*x)+x$$

$$O=\text{sqrt}((1/(x+y))+(1/(t+z)))$$

$$P=4*(e(-x/2))*\text{sin}(\text{pi}*x)$$

Tentukan nilai fungsi tersebut dengan nilai-nilai  $t=25$ ,  $x=43$ ,  $y=15.25$ ,  
 $z=8.2$  !

Untuk penyelesaian di atas adalah sbb:

**Pertama**, dibuat dulu variabel nya

```
t=25;x=43;y=15.25;z=8.2;
```

**Kedua**, masukkan nilai tersebut ke masing-masing fungsinya

```
M=4*x^2+3*y+10
```

```
M =
```

```
7.4518e+003
```

```
N=exp(2*x)+x
```

```
N =
```

```
2.2352e+037
```

```
O=sqrt((1/(x+y))+1/(t+z)))
```

```
O =
```

```
0.2175
```

```
P=4*(exp(-x/2))*sin(pi*x)
```

```
P =
```

```
1.6223e-023
```

39

## Contoh 2 : Penentuan volume silinder berlubang

Sebuah silinder mempunyai diameter luar 6.3125 dan diameter dalam 5.762, tentukan volumenya bila diketahui rumus **untuk** mencari volumenya:

( 3 3 )

3

$$V = 4\pi RE - RI$$

Dimana RE adalah diameter luar dan RI adalah diameter dalam

### Penyelesaian :

```
re=6.3125;ri=5.762;
```

```
v=4/3*pi*(re^3-ri^3);
```

```
disp(['Volume = ',num2str(v)])
```

```
Volume = 252.3169
```

## 1. Operator Logika dan Relasional

MATLAB menyediakan operasi logika dan relasional, hal ini diperlukan untuk menjawab pertanyaan benar atau salah dan salah satu manfaat yang penting dari kemampuan ini adalah untuk mengontrol urutan eksekusi sederetan perintah MATLAB (biasanya dalam M-File) berdasarkan pada hasil pertanyaan benar/salah.

Sebagai masukan pada semua ekspresi relasi dan logika, MATLAB menganggap semua angka tidak nol sebagai benar, nol sebagai salah. Hasil dari semua ekspresi logika relasi dan logika adalah satu untuk benar dan nol untuk salah dengan tipe array logika yaitu hasilnya memuat bilangan 1 dan 0 yang tidak saja dapat digunakan untuk statemen matematika akan tetapi dapat juga untuk pengalamatan Operator relasi MATLAB terdiri dari semua perbandingan:

**Tabel 2.1** Operator Relasi MATLAB

Operator Relasi	Deskripsi
<	Kurang dari
>	Lebih dari
<=	Kurang dari atau sama dengan
>=	Lebih dari atau sama dengan
=	Sama dengan
~=	Tidak sama dengan

Operator relasi MATLAB dapat digunakan untuk membandingkan dua array berukuran sama atau untuk membandingkan array dengan skalar. Operator logika menyediakan cara untuk menggabungkan atau menegaskan ekspresi relasi. Selanjutnya Operator-operator logika dalam MATLAB yaitu:

**Tabel 2.2** Operasi Logika MATLAB

Operator Relasi	Deskripsi
&	AND
	OR
~	NOT

## 2. Menyimpan dan Memanggil Data

Untuk Menyimpan dan memanggil data dari file pilih **File Save Workspace As ...** . Untuk memanggil data digunakan pilihan **Load Workspace As** atau **Open** pada menu file. Sedangkan untuk mengimport data , untuk **Matlab** versi 6 keatas pilih **file Import Data ..** . **Matlab** juga menyediakan dua perintah ---- **save dan load** ----- yang jauh lebih fleksibel. Perintah **save** untuk menyimpan satu atau lebih variabel dalam file format Yang sesuai dengan pilihan anda.



**Contoh:**

```
clear all
```

```
    x=1:10;y=10:10:10:100; % membuat array baru save
```

Saving to: Matlab.mat menyimpan semua variabel **Matlab** dalam format biner di file **Matlab.mat** save data menyimpan semua variabel **Matlab** dalam format biner di file data.mat save data\_x x menyimpan variabel x dalam format biner di file data\_x.mat save data\_xy x y /ascii menyimpan variabel x dan y dalam format biner di file data\_xy dalam format ascii **untuk** membuka data digunakan perintah load.

**Contoh;**

```
load data_x.mat
```

**PERTANYAAN DAN SOAL**

---

1. Sebutkan beberapa lingkungan kerja MATLAB!!
2. Bagaimana proses instalasi dan cara memulai menggunakan program MATLAB?
3. Jelaskan beberapa perbedaan windows yang terdapat dalam MATLAB!
4. Coba tuliskan beberapa sintak dasar dalam MATLAB untuk menghitung luas bujur sangkar

## BAB 3

### PEMROGRAMAN MATRIX DALAM MATLAB

---

20

#### TUJUAN INSTRUKSIONAL:

*Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:*

- 1) *Mengerti konsep dasar implementasi matematika dalam MATLAB*
  - 2) *Menjelaskan model persamaan matematika pada MATLAB*
  - 3) *Menuliskan persamaan untuk implemntasi grafik MATLAB*
  - 4) *Menjelaskan kembali model implementasi array dan matriks pada MATLAB*
  - 5) *Menuliskan kembali contoh pemrograman Matriks pada MATLAB*
- 

Fungsi matematika adalah fungsi umum dan terpenting dari MATLAB. Bahkan seorang profesor yang sekaligus pengguna MATLAB mengatakan bahwa alasan mengapa MATLAB sangat berguna untuk pemrosesan sinyal adalah bukan karena MATLAB dirancang secara khusus untuk memproses sinyal, melainkan karena MATLAB dirancang untuk matematika.

#### A. DASAR MATEMATIKA PADA MATLAB

Seperti kalkulator scientific kita tinggal mer<sup>62</sup>itik masalah matematik di command window dengan format seperti matematika biasa. MATLAB mempunyai berbagai fungsi umum yang penting untuk matematika, teknik dan ilmu pengetahuan lain. Sebagai tambahan atas fungsi-fungsi umum tersebut, MATLAB menyediakan ratusan fungsi khusus dan algoritma yang berguna untuk menyelesaikan permasalahan tertentu.. Berikut ini satu contoh perhitungan logaritma.

```
57
» x=sqrt(3)/2
    x =
    0.8660
» y=acos(x)
    y =
    0.5236
» y_deg=y*180/pi
    y_deg =
```

30.0000

Hitungan terakhir di atas adalah jika kita ingin melihat sudut  $y$  dalam derajat dikarenakan MATLAB hanya bekerja dalam radian. Pada command window MATLAB saat ini terdapat tiga variabel, yang dapat kita lihat dengan mengetik instruksi `who`.

» `who`

Your variables are:

x    y    y\_deg

Bila kita ingin menghapus satu atau lebih variabel yang ada di MATLAB maka instruksi yang digunakan adalah `clear`. MATLAB juga menyediakan panduan `help`. Ketik `help` lalu diketik instruksi yang akan dicari informasinya.

17

» **help clear**

`CLEAR` Clear variables and functions from memory.  
`CLEAR` removes all variables from the workspace.  
`CLEAR VARIABLES` does the same thing.  
`CLEAR GLOBAL` removes all global variables.  
`CLEAR FUNCTIONS` removes all compiled M-functions.  
`CLEAR MEX` removes all links to MEX-files and all M-functions.  
`CLEAR ALL` removes all variables, globals, functions and MEX links.  
`CLEAR CLASSES` is the same as `CLEAR ALL` except that class definitions are also cleared. If any objects exist outside the workspace (say in `userdata` or persistent in a locked m-file) a warning will be issued and the class definition will not be cleared. `CLEAR CLASSES` must be used if the number or names of fields in a class are changed

14

**Tabel 3.1** Fungsi-fungsi Umum

<code>abs(x)</code>	harga mutlak atau besarnya bilangan kompleks
<code>acos(x)</code>	invers kosinus
<code>acosh(x)</code>	invers kosinus hiperbolik
<code>angle(x)</code>	sudut suatu
<code>asin(x)</code>	invers sinus
<code>asinh(x)</code>	invers sinus hiperbolik
<code>atan(x)</code>	invers tangent
<code>atan2(x)</code>	invers tangen untuk empat kuadran
<code>atanh(x)</code>	invers tangen hiperbolik
<code>ceil(x)</code>	pembulatan ke arah plus tak berhingga
<code>conj(x)</code>	konjugat bilangan kompleks

27

<code>cos(x)</code>	Kosinus
<code>cosh(x)</code>	kosinus hiperbolik
<code>exp(x)</code>	eksponensial: $e^x$
<code>fix(x)</code>	pembulatan ke arah nol
<code>floor(x)</code>	pembulatan ke arah minus tak berhingga
<code>gcd(x,y)</code>	faktor persekutuan terbesar bilangan bulat x dan y
<code>imag(x)</code>	bagian imajiner suatu bilangan kompleks
<code>lcm(x,y)</code>	kelipatan persekutuan terkecil bilangan bulat x dan y
<code>log(x)</code>	logaritma natural
<code>log10(x)</code>	logaritma biasa
<code>real(x)</code>	bagian real suatu bilangan kompleks
<code>rem(x,y)</code>	sisa pembagian: rem(x,y) menghasilkan sisa pembagian x/y
<code>round(x)</code>	pembulatan ke arah bilangan bulat terdekat
<code>sign(x)</code>	menghasilkan tanda dari argumen, misalnya: sign(1.2)=1, sign(-23.4)=-1, sign(0)=0
<code>sin(x)</code>	Sinus
<code>sinh(x)</code>	sinus hiperbolik
<code>sqrt(x)</code>	akar kuadrat
<code>tan(x)</code>	Tangent

## B. PERSAMAAN MATEMATIKA PADA MATLAB

Sebelumnya Anda diperkenalkan bagaimana MATLAB dapat dipergunakan seperti kalkulator canggih yang dapat diprogram. Akan tetapi kalkulator biasa tidak dapat bekerja untuk memanipulasi ekspresi matematika tanpa menggunakan bilangan. Maka di sini akan dibahas aplikasi MATLAB untuk persamaan-persamaan matematika yang sering dijumpai dalam dunia teknik, khususnya pada sistem kontrol.

### 1. Membuat Variabel Simbolik

Berikut adalah contoh keunggulan MATLAB dibanding kalkulator scientific biasa karena dapat menghitung simbol-simbol yang bukan angka. Caranya adalah dengan merubah terlebih dahulu variabel numerik menjadi variabel simbolik.

```

» x=sym('x');
» diff(sin(x))
ans =
cos(x)

```

Perhatikan format penulisan statemen pertama yang mendefinisikan x sebagai simbol sehingga MATLAB dapat mendiferensialkan sinus. Statemen pertama bisa juga ditulis dalam bentuk: syms x. Untuk mengetahui suatu variabel angka atau simbol dengan cara sebagai berikut:

```
» class(x)
ans =
sym
```

## 2. Persamaan Aljabar

Berik<sup>184</sup>ni contoh mengoperasikan persamaan aljabar biasa  $f = 5x^3 + x^2 + 4$  dan  $g = x^2 + x - 2$

```
» syms x
» f=5*x^3+x^2+4 % mendefinisikan ekspresi simbolik f dan g
f =106
5*x^3+x^2+4
» g=x^2+x-2
g =
x^2+x-2
» f^(3*x)
ans =
(5*x^3+x^2+4)^(3*x)

» h=subs(ans,x,1)
h =
1000
```

Instruksi terakhir mengganti/mensubstitusi variabel x pada persamaan ans (kependekan dari *answer/jawaban*) dengan 1 dan diperoleh hasil 1000.

Instruksi double adalah kebalikan dari sym, yaitu merubah simbol menjadi numerik. Untuk mengintegalkan atau mendiferensialkan fungsi f dan g dengan ca<sup>189</sup> sebagai berikut:

```
» diff(f)
ans =
57 15*x^2+2*x
» int(g)
ans =
1/3*x^3+1/2*x^2-2*x
» pretty(ans)
1/3. x^3 + 1/2. x^2 - 2x
```

Perintah terakhir adalah untuk memperindah bentuk persamaan jawaban sehingga memiliki bentuk seperti penulisan matematika biasa.

### 3. Persamaan Diferensial

Saat kita memodelkan suatu sistem fisis, yang pertama kita peroleh adalah persamaan diferensial. MATLAB membantu menyelesaikan persamaan diferensial yang merupakan kendala bagi orang yang baru belajar sistem kontrol. Cara penulisan diferensial pada MATLAB adalah dengan huruf kapital D untuk orde 1, D2 untuk orde 2, D3 untuk orde 3 dan seterusnya. Untuk lebih jelasnya misalnya kita diminta untuk menyelesaikan persamaan diferensial dengan harga awal:

$$\frac{d^2y}{dt^2} = \cos(2t) - y \quad \frac{dy}{dt}(0) = 0 \quad y(0) = 1$$

```
43 » y=dsolve('D2y=cos(2*t)-y, Dy(0)=0, y(0)=1')
```

```
y =
```

```
(1/2*sin(t)+1/6*sin(3*t))*sin(t)+(1/6*cos(3*t)
```

```
1/2*cos(t))*cos(t)+4/3*cos(t)
```

```
» pretty(y)
```

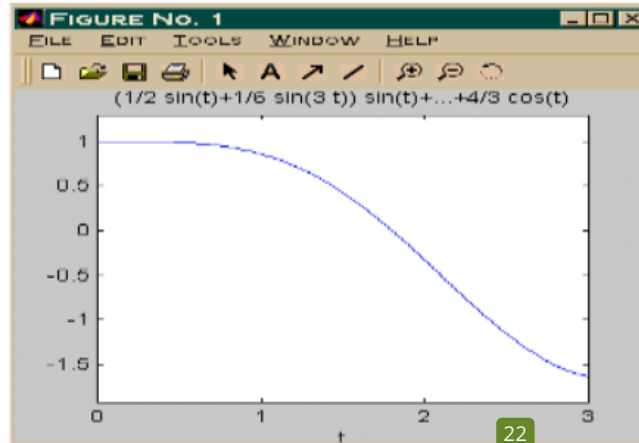
```
43 (1/2 sin(t) + 1/6 sin(3 t)) sin(t) + (1/6 cos(3 t) - 1/2 cos(t)) cos(t) + 4/3 cos(t)
```

### C. IMPLEMENTASI DALAM BENTUK GRAFIK

Bila ingin melihat bentuk grafik keluaran persamaan diferensial di atas yang berdimensi dua dapat diketik :

```
» ezplot(y,[0 3])
```

Dan kemudian muncul lembar grafik MATLAB yang berbentuk sebagai berikut:



**Gambar 3.1** Grafik Persamaan  $\frac{d^2y}{dt^2} = \cos(2t) - y$   $\frac{dy}{dt}(0) = 0$   $y(0) = 1$

Menggambar grafik 3D secara manual amat sulit, tetapi MATLAB mampu menggambarannya dengan cepat dan akurat. Untuk menjadi ahli dalam menggambar grafik, Anda dapat melihat menu help dengan cara menetik pada command window.

32

» help graph3d

Three dimensional graphs.

#### Elementary 3-D plots.

plot3 - Plot lines and points in 3-D space.

mesh - 3-D mesh surface.

surf - 3-D colored surface.

fill3 - Filled 3-D polygons.

#### Color control.

colormap - Color look-up table.

caxis - Pseudocolor axis scaling.

shading - Color shading mode.

hidden - Mesh hidden line removal mode.

brighten - Brighten or darken color map.

colordef - Set color defaults.

graymon - Set graphics defaults for gray-scale monitors.

#### Lighting.

surfl - 3-D shaded surface with lighting.

lighting - Lighting mode.

material - Material reflectance mode.

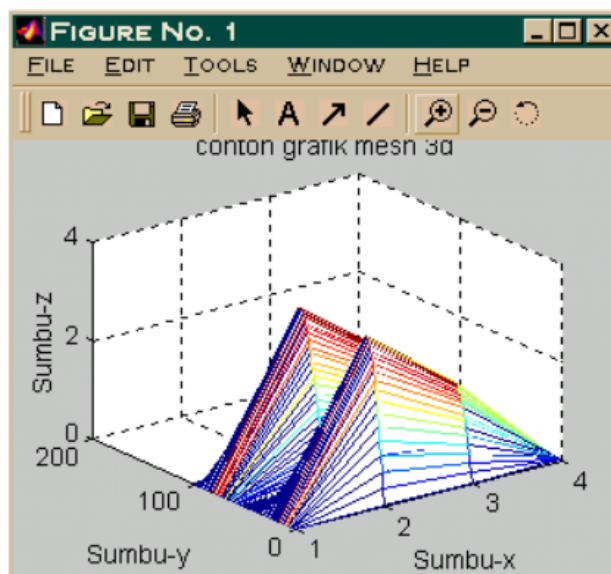
specular - Specular reflectance.  
diffuse - Diffuse reflectance.  
Surfnorm - Surface normals.

Sedangkan untuk tatacara penulisannya ada baiknya Anda belajar sendiri dengan cara mengetik demo pada command window hingga muncul jendela MATLAB DEMOS.

Berikut ini adalah contoh penggunaan salah satu fasilitas yang ada di grafik tiga dimensi MATLAB yaitu mesh/jala.

```
» [X,Y]=meshgrid(-pi:pi/50,-pi:pi/50:pi);  
» Z=X.*sin(X).*Y.*sin(Y);  
» mesh(Z)  
» title('contoh grafik mesh 3d')  
» xlabel('Sumbu-x'),ylabel('Sumbu-y'),zlabel('Sumbu-z');
```

Saat mesh(Z) di-enter akan muncul grafik sesuai fungsi yang ditulis di command window. Untuk memberi penjelasan nama judul dengan instruksi title sedangkan xlabel, ylabel dan zlabel memberi keterangan pada tiap sumbu.



**Gambar 3.2** Grafik Mesh/Jala 3D

#### **D. ARRAY DAN MATRIKS**



Operasi dengan satu besaran skalar saja memang merupakan dasar matematika. Namun jika dalam sesaat Anda ingin melakukan operasi yang sama pada beberapa bilangan, perulangan operasi skalar akan menghabiskan waktu dan tentu saja tidak praktis.

### 1. Penulisan Array

Array sesungguhnya adalah matriks vektor yang mempunyai satu baris atau kolom saja. Misalnya kita ingin mencari harga cosinus sudut dari 0 sampai  $\pi$ .

```

79
» x=[0 0.1*pi 0.2*pi 0.3*pi 0.4*pi 0.5*pi]
      x =
      0  0.3142  0.6283  0.9425  1.2566  1.5708
» y=cos(x)
      y =
      1.0000  0.9511  0.8090  0.5878  0.3090  0.0000

```

Perhatikan tata cara penulisannya! Setelah nama variabel dan tanda sama dengan array ditulis di antara kurung [ ]. Antar sel dipisah dengan spasi (boleh lebih dari satu spasi). Di antara tanda kurung [ ] dapat juga kita sisipi data dari Microsoft Excel® dengan cara copy-paste biasa.

### 2. Alamat Array

Array contoh sebelumnya adalah matriks 1x6 atau dengan kata lain matriks dengan satu baris dan enam kolom yang lebih dikenal dengan nama vektor. Dalam MATLAB elemen-elemen array diakses dengan *subscript*. Misalnya  $x(1)$  berarti elemen pertama,  $x(2)$  elemen kedua dan seterusnya.

```

» x(1)
      ans =
      0

» x(2)
      ans =
      0.3142

```

Untuk mengakses suatu blok elemen, MATLAB menyediakan notasi kolom.

```

» x(1:4)
      ans =
      0  0.3142  0.6283  0.9425

```

### 3. Membentuk Array

Untuk jumlah elemen array yang banyak, tidak praktis dengan cara mengetikkan satu persatu array-nya. Ada dua cara memasukan nilai x dengan notasi kolom. Cara pertama:

```
» x=(0:0.1:0.5)*pi
```

```
x =
```

```
0 0.3142 0.6283 0.9425 1.2566 1.5708
```

Dengan cara pertama, notasi kolom (0:0.1:0.5) menghasilkan array yang dimulai dengan 0, meningkat setiap 0,1 dan berhenti pada 0,5 (Matlab menggunakan notasi matematika eropa dimana titik berarti koma untuk notasi kita).

Selanjutnya cara kedua:

```
» x=linspace(0,0.5*pi,6)
```

```
x =
```

```
0 0.3142 0.6283 0.9425 1.2566 1.5708
```

Dengan cara kedua, notasi kolom (0,0.5\*pi,6) menghasilkan array yang dimulai dengan 0, diakhiri  $0,5\pi$  dengan jumlah elemen enam. Cara di atas sering digunakan di MATLAB dimana jarak antar elemen linear.

Untuk yang tidak linear misalnya logaritmik, yang biasa dipakai pada diagram BODE, adalah sebagai berikut:

```
» logspace(0,2,6)
```

```
ans =
```

```
1.0000 2.5119 6.3096 15.8489 39.8107 100.0000
```

Maksud notasi kolom (0,2,6) adalah suatu array yang dimulai  $10^0$ , diakhiri  $10^2$  dengan jumlah elemen enam.

**Tabel 3.2** Instruksi Pada Array

<b>Pengalaman Array</b>	
<code>x=[2 2*pi sqrt(2) 2-3j]</code>	menciptakan vektor baris x yang memuat elemen-elemen yang diberikan

<code>x=awal: akhir</code>	membuat vektor baris x dimulai dengan awal, kenaikan satu, diakhiri pada atau sebelum akhir
<code>x=awal:kenaikan:akhir</code>	membuat vektor baris x diawali dengan awal, kenaikan sebesar kenaikan, diakhiri pada atau sebelum akhir
<code>x=linspace(awal,akhir,n)</code>	menciptakan vektor baris x diawali dengan awal, berakhir dengan akhir, mempunyai n elemen
<code>x=logspace(awal,akhir,n)</code>	menciptakan vektor kolom dengan elemen berjarak logaritmis dimulai dengan $10^{\text{awal}}$ , diakhiri dengan $10^{\text{akhir}}$ , mempunyai n elemen
<b>1</b> <code>A(r,c)</code>	mengalami subarray dalam A dengan indeks baris yang diinginkan dalam r dan kolom yang diinginkan dalam c
<code>A(r,:)</code>	mengalami subarray dalam A dengan indeks baris yang dialami dalam r dan semua kolom diambil
<code>A(:,r)</code>	mengalami subarray dalam A dengan semua baris diambil dan indeks kolom yang diinginkan dalam r
<code>A(:)</code>	mengalami semua elemen dalam A sebagai vektor kolom diambil kolom per kolom
<code>A(i)</code>	mengalami subarray dalam A dengan indeks tunggal i, dianggap A merupakan vektor kolom a(:)
<code>A(x)</code>	mengalami subarray dalam A dengan array logika x; x harus mempunyai ukuran yang sama dengan A
<b>Pencarian Array</b>	
<code>i=find(x)</code>	menghasilkan indeks dari array x dimana elemen-elemennya tidak nol
<code>[r,c]=find(X)</code>	menghasilkan indeks baris dan kolom dari array X dimana elemen-elemennya tidak nol

#### 4. Membentuk Matriks

Matriks adalah perluasan dari array/vektor. Cara penulisannya hampir sama dengan array hanya saja untuk memisahkan antara baris satu dengan lainnya digunakan titik koma (;) atau menekan enter.

Berikut ini adalah operasi dasar pada matriks, yaitu matriks transpose, determinan dan invers.

```
120
» A=[1 2 3; 4 5 6; 8 9 0]
    A =
     1  2  3
```

```
4 5 6
8 9 0
```

```
» B=A' % transpose matriks A
```

```
B =
1 4 8
2 5 9
3 6 0
```

```
» C=det(A) % 48determinan matriks A
```

```
C =
30
```

```
» D=inv(A) % invers matriks A
```

```
D =
-1.8000 0.9000 -0.1000
1.6000 -0.8000 0.2000
-0.1333 0.2333 -0.1000
```

## 5. Memanipulasi Matriks

Sekali suatu matriks dibentuk, <sup>72</sup>MATLAB menyediakan cara mudah untuk menyisipkan, mengambil dan mengatur kembali sebagian dari matriks tersebut dengan mengidentifikasi *subscript* yang berkaitan. Penguasaan akan hal ini merupakan kunci untuk menggunakan MATLAB secara efisien. Berikut ini contoh memanipulasi array dan matriks.

```
82» A=[1 2 3; 4 5 6; 7 8 9]
```

```
A =
1 2 3
4 5 6
7 8 9
```

```
» A(3,3)=0 % mengubah elemen pada baris3 kolom 3 menjadi nol
```

```
A =
1 2 3
4 5 6
7 8 0
```

```
» A(:,3)=4 % mengubah seluruh elemen kolom 3 menjadi 4
```

```
A =  
    1    2    4  
    4    5    4  
    7    8    4
```

```
187  
» B=A(1:3,1:3) % matriks B dibuat dengan urutan baris A yang terbalik
```

```
127  
B =  
    7    8    4  
    4    5    4  
    1    2    4
```

```
» C=[A B(:,[1 3])] % membuat C dgn A yg disisipi kolom 1 & 3 matriks B
```

```
C =  
    1    2    4    7    4  
    4    5    4    4    4  
    7    8    4    1    4
```

## 6. Karakteristik Matriks

Karakteristik matriks menggambarkan perihal suatu matriks yaitu ukuran matriks, rank dan nilai eigen.

```
» C
```

```
C =  
    1    2    4    7    4  
    4    5    4    4    4  
    7    8    4    1    4
```

```
» size(C) % mengetahui ukuran matriks C
```

```
ans =  
     3     5
```

Untuk mengetahui seluruh ukuran matriks yang telah dibuat pada command window, dengan cara sebagai berikut:» whos

Name	Size	Bytes Class
A	3x3	72 double array
B	3x3	72 double array

```

C          3x5          120 double array
D          3x4          96 double array
E          3x3          72 double array
F          3x4          96 double array
ans       1x2          16 double array
Grand total is 68 elements using 544 bytes

```

Nilai eigen  $\lambda$  adalah salah satu karakteristik matriks yang sering dijumpai pada sistem kontrol. Misal kita ingin mencari nilai  $\lambda$  matriks A.

```

» A
57
A =
    1    2    3
    4    5    6
    7    8    9

```

```

» eig(A)
ans =
    16.1168
    -1.1168
    -0.0000

```

Rank (peringkat) matriks secara teori dicari dengan mengubah matriks menjadi bentuk eselon baris tereduksi. Misal kita ingin mencari rank matriks G:

```

» G
G =
    1    2    4
    2    4    8
    1    2    3

```

```

» [H,pivot]=rref(G)
H =
    1    2    0
    0    0    1
    0    0    0
pivot =
    1    3

```

```

» length(pivot)
ans =
    2

```

Instruksi terakhir menghasilkan rank matriks  $G = 2$ . Atau kita dapat dengan cepat menghitung rank matriks tanpa melakukan pivot (fungsi `rref`) dengan bantuan fungsi-fungsi khusus yang telah disediakan MATLAB.

```
» rank(G)
ans =
    2
```

Berikut ini tabel yang berisi fungsi-fungsi yang berkaitan dengan matriks.

**Tabel 3.3** Instruksi MATLAB pada Matriks

Fungsi-fungsi Matriks	
<code>balance(A)</code>	penskalaan untuk meningkatkan ketepatan nilai eigen
<code>cdf2rdf(A)</code>	bentuk kompleks diagonal ke bentuk real diagonal
<code>chol(A)</code>	faktorisasi Cholesky
<code>cholinc(A,droptol)</code>	faktorisasi Cholesky tidak lengkap
<code>cond(A)</code>	bilangan kondisi matriks
<code>condest(A)</code>	perkiraan bilangan kondisi matriks 1-norm
<code>condeig(A)</code>	kondisi w.r.t nilai eigen
<code>det(A)</code>	determinan
<code>d=eig(A), [V,D]=eig(A)</code>	nilai eigen dan vektor eigen
<code>eigs(A)</code>	sedikit nilai eigen dan vektor eigen
<code>expm(A)</code>	pemangkatan matriks
<code>expm1(A)</code>	implementasi M-file dari <code>expm</code>
<code>expm2(A)</code>	pemangkatan matriks menggunakan deret Taylor
<code>expm3(A)</code>	pemangkatan matriks menggunakan nilai dan vektor eigen
<code>funm(A,'fun')</code>	menghitung fungsi matriks umum
<code>hess(A)</code>	bentuk Hessenberg
<code>inv(A)</code>	invers matriks
<code>logm(A)</code>	logaritma matriks
<code>lscov(A,b,V)</code>	kuadrat terkecil dengan kovarian diketahui
<code>lu(A)</code>	faktor dari eliminasi Gauss
<code>luinc(A,droptol)</code>	faktorisasi LU tidak lengkap

<code>nnls(A,b)</code>	kuadrat terkecil non-negatif
<code>norm(A)</code>	normal matriks dan vector
<code>norm(A,1)</code>	normal-1
<code>norm(A,2)</code>	normal-2 Euclidan
<code>norm(A,inf)</code>	ketidak-terbatasan
<code>norm(A,p)</code>	normal-P (vektor saja)
<code>norm(A,'fro')</code>	normal-F
<code>normest(A)</code>	memperkirakan normal-2 untuk matriks jarang besar
<code>null(A)</code>	bidang nol
<code>orth(A)</code>	Ortogonalisasi
<code>pinv(A)</code>	Pseudoinverse
<code>planerot(A)</code>	rotasi bidang Given
<code>poly(A)</code>	karakteristik polynomial
<code>polyeig(A1,A2,...)</code>	menyelesaikan masalah nilai eigen polynomial
<code>polyvalm(A)</code>	mengevaluasi matriks polynomial
<code>qr(A)</code>	dekomposisi ortogonal-triangular
<code>qrdelete(Q,R,j)</code>	menghapus kolom dari faktorisasi QR
<code>qrinsert(Q,R,j,x)</code>	menyisipkan kolom dalam faktorisasi QR
<code>qz(A,B)</code>	nilai eigen umum
<code>rank(A)</code>	jumlah baris atau kolom bebas linear
<code>rcond(A)</code>	perkiraan komdisi resiprok
<code>rref(A)</code>	bentuk eselon baris tereduksi
<code>rsf2csf(U,T)</code>	bentuk real Schur ke bentuk kompleks Schur
<code>sqrtn(A)</code>	akar kuadrat matriks
<code>subspace(A,B)</code>	sudut antara dua bidang
<code>svd(A)</code>	dekomposisi nilai singular
<code>svds(A,K)</code>	sedikit nilai-nilai singular
<code>trace(A)</code>	jumlah elemen-elemen diagonal

## 7. Operasi Pada Matriks

MATLAB mampu melaksanakan operasi penjumlahan, pengurangan, perkalian, pembagian dan pemangkatan hanya bila kondisi terpenuhi. Tidak ada salahnya pembaca membuka kembali teori aljabar linear matriks. Penjumlahan dan pengurangan matriks membutuhkan orde yang sama sedangkan perkalian boleh dilaksanakan jika kolom matriks pertama sama dengan jumlah baris matriks kedua.

» `A=[1 2 3 4; 5 6 7 8; 9 10 11 12]`

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$



9 10 11 12

» B=[1 1 1 1; 2 2 2 2; 3 3 3 3]

B =

```
1 1 1 1
2 2 2 2
3 3 3 3
```

» C=[1 17; 4 5 6; 7 8 9; 10 11 12]

C =

```
1 2 3
4 5 6
7 8 9
10 11 12
```

» D=A+B

D =

```
2 3 4 5
7 8 9 10
12 13 14 15
```

» E=A\*C

E =

```
70 80 90
158 184 210
246 288 330
```

» F=A.^B

F =

```
1 2 3 4
25 36 49 64
729 1000 1331 1728
```

Matriks D pada instruksi di atas adalah hasil operasi penjumlahan, matriks E hasil perkalian sedangkan matriks F hasil pemangkatan dengan matriks B.

## 8. Persamaan Linear

Salah satu masalah pertama pada MATLAB adalah menyelesaikan persamaan aljabar linear dari sekumpulan persamaan linear, misalnya:

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 8 & 9 & 0 \end{bmatrix} \cdot \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix} = \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix}$$

$$A \cdot X = B$$

Jika terdapat penyelesaian, maka terdapat beberapa metode untuk menemukan penyelesaian itu, seperti eliminasi Gauss – Jordan, faktorisasi LU, atau penggunaan langsung dari  $A^{-1}$ . Secara analitis, penyelesaian ditulis sebagai  $X = A^{-1} \cdot B$ .

MATLAB menyelesaikan persamaan di atas dengan cepat dan akurat:

```
88
» A=[1 2 3; 4 5 6; 8 9 0]
```

```
A =
     1     2     3
     4     5     6
     8     9     0
```

```
» B=[100; 200; 300]
```

```
B =
    100
    200
    300
```

```
» X=inv(A)*B
```

```
X =
   -30.0000
    60.0000
     3.3333
```

Mungkin kita masih dapat menghitung penyelesaian persamaan di atas secara manual, tetapi jika matriks yang akan dihitung berukuran besar, misalnya  $1000 \times 1000$ , tentu saja akan mengalami kesulitan.

## E. PEMROGRAMAN MATRIKS PADA MATLAB

Untuk membuat program dalam bahasa MATLAB pertama-tama bukalah jendela M-File dengan cara seperti yang telah dijelaskan di bab II. Ada dua jenis m-file yaitu *script file* dan *function file*. *Script file* tidak menggunakan argumen input atau mengembalikan argumen output. *Function file* dapat menggunakan argumen input atau menghasilkan argumen output.

## 1. Membuat Script File

Langkah pertama membuat program m-file adalah dengan mengetik instruksi pada jendela MATLAB Editor/Debugger. Misalnya kita ingin membuat program *script file* yang menggambarkan perbandingan respon dua buah sistem:

$$y1 = 1 - e^{-0.5t} \sin t$$

$$y2 = 1 - e^{-1t} \sin t$$

Ketikan pada jendela MATLAB Editor/Debugger program sebagai berikut:

**%program membandingkan respon**

```
t=0:1:15.;  
y=1.-(exp(-.5.*t).*sin(t));  
plot(t,y),grid  
title('perbandingan respon sistem')  
gtext('1.-(exp(-.5.*t).*sin(t))')  
hold;  
y=1.-(exp(-1.*t).*sin(t));  
plot(t,y),grid  
gtext('1.-(exp(-1.*t).*sin(t))')
```



**Gambar 3.3** Tampilan Grafik1.m pada MATLAB Editor/Debugger

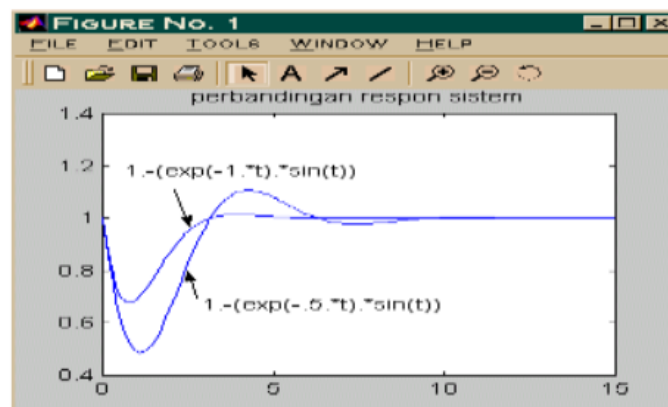
Kemudian simpanlah program Anda dengan cara mengklik **File** dalam layar **MATLAB Editor/Debugger** dan pilih **Save As...** Tuliskan nama file yang baru Anda buat misalnya grafik1 dan tekan tombol **Save**. Pastikan file Anda tersimpan dalam direktori yang ada dalam jalur pencarian (*path*) MATLAB, yang default-nya pada subdirektori work.

Untuk menjalankan program yang baru saja Anda buat, aktifkan kembali command window. Lalu tekan enter setelah mengetik grafik1:

» grafik1

Current plot held

Di bawah grafik muncul kalimat current plot held, yang sesuai dengan instruksi pada program yaitu hold;. Maka akan muncul hasil program yaitu grafik respon:



**Gambar 3.4** Grafik Hasil Pemrograman Script File

Anda dapat menambahkan tanda panah pada persamaan sebagai penunjuk dengan cara mengklik gambar panah pada toolbox.

## 2. Membuat *Function File*

Seperti telah dijelaskan di atas bahwa *function file* memproses input agar dihasilkan output sesuai dengan program m-file yang dibuat. Misal kita ingin membuat program yang mensortir suatu array dimana nanti diharapkan memperoleh array yang elemennyaurut dari kecil ke besar.

```
function [b, j]=sortir(a)
%fungsi sortir, dengan urutan menaik
```

```

%keluarannya, parameter j dengan permutasi
%menemukan vektor b dari vektor a
[b, j]=sort(a);

```

Gunakan prosedur yang sama seperti membuat *script file*. Misalnya kita beri nama *sortir*. Untuk menjalankannya kita kembali ke command window. Tentu saja kita tidak dapat langsung menjalankannya tanpa adanya input. Oleh karena itu kita harus memberi input vektor yang akan kita urutkan.

```

» a=[13 15 19 22 100 44 4 98];
» [b, j]=sortir(a)    % mengambil function file dari m-file

```

```

b =
    4    13    15    19    22    44    98   100
j =
    7     1     2     3     4     6     8     5

```

```

» sortir(a)
ans =
    4    13    15    19    22    44    98   100

```

Vektor *b* adalah hasil *sortir* dari vektor *a* yang kita masukkan sedangkan vektor *j* jumlah permutasi yang dilakukan. Perintah yang terakhir kita ketik apabila hanya ingin mensortir saja.

### 23 3. Kontrol Program

MATLAB menyediakan 4 alat yang dapat digunakan programmer saat menulis program, yaitu:

- For
- While
- If – else – end , dan
- Switch – Case

Berikut ini contoh penggunaan kontrol program for untuk menghitung harga kosinus dari  $n=0$  hingga  $\pi$ . Di sini akan dihitung sinus dari 0 hingga  $\pi$  dengan 11 point.

```

» for t=0:10
    y(t+1)=sin(pi*t/10);

```

```

end
» y
y =
Columns 1 through 7
0 0.3090 0.5878 0.8090 0.9511 1.0000 0.9511
Columns 8 through 11
0.8090 0.5878 0.3090 0.0000

```

Program di atas dapat juga ditulis dengan kontrol program while sebagai berikut:

```

193
» while t>10
t=0 18
y(t+1)=sin(pi*t/10);
end
» y
y =
Columns 1 through 7
0 0.3090 0.5878 0.8090 0.9511 1.0000 0.9511
Columns 8 through 12
0.8090 0.5878 0.3090 0.0000 -0.3090

```

Berikut adalah contoh penggunaan kontrol program if – else – end. Bila barang yang dibeli lebih dari 10, maka dapat diskon 20%. Jika pembelian kurang dari 10, tidak dapat diskon. Pertama-tama buat *function file* dari MATLAB Debugger.

```

23
if barang>10
    harga=(1-0.2)*hargabrg*barang
else
    harga=hargabrg*barang
end

```

Lalu simpan dengan nama harga, sehingga diperoleh m-file dengan file harga.m. Coba program dijalankan lewat command window.

```

» barang=5
    barang =

```

```

5
» hargabrg=100
    hargabrg =
        100
» harga      % memanggil m-file harga.m
    harga =
        500
» clear      % membersihkan variabel
» barang=100
    barang =
        100
» hargabrg=100
    hargabrg =
        100
» harga      % memanggil m-file harga.m
    harga =
        8000

```

Pada program di atas saat jumlah barang yang dibeli lima dengan harga satuan 100 didapat harga = 500. Tetapi jika pembelian 100 (lebih dari 10 buah) harganya bukan 10.000 melainkan diskon 20%=8.000.

Bila sederetan perintah harus dikerjakan dengan didasarkan pada penggunaan berulang-ulang suatu tes dengan argumen yang sama, konstruksi switch-case akan lebih tepat digunakan. Konstruksi control ini memiliki bentuk:

```

61
switch ekspresi
    case test_ekspresi1
        deret_perintah1
    case {test_ekspresi2, test_ekspresi3, test_ekspresi4}
        deret_perintah2
    otherwise
        deret_perintah 3
end

```

23  
Berikut ini adalah contoh penggunaan switch case dimana merubah beberapa satuan menjadi centimeter, cm. Buatlah m-file sebagai berikut lalu simpan dengan nama ubahunit.

```

switch units
case{'inch','in'}
    y=x*2.54;
case{'meter','m'}
    y=x/100;
case{'centimeter','cm'}
    y=x;
otherwise
    disp{['Unit tidak diketahui: 'units]}
    y=nan;
end;

```

Setelah itu masuk ke command window. Misal kita masukan harga x=10 in, dan akan diubah menjadi centimeter, cm.

```

» x=10.;
» units='in';

```

```

» ubahunit          % panggil m-file ubahunit.m
» y
    y =
    25.4000

```

#### 4. Fungsi Inline dan Feval

Fungsi inline dipakai selama satu sesi MATLAB tanpa membuka MATLAB Debugger. Berikut ini contoh membuat *inline function* guna mencari harga diskriminan persamaan kuadrat.

```

10 » disk=inline('sqrt(b.^2-4*a*c)','a','b','c')

```

```

disk =

```

```

    Inline function:

```

```

13 disk(a,b,c) = sqrt(b.^2-4*a*c)

```

```

» disk(1,3,2)

```

```

ans =

```

```

    1

```



Feval memanfaatkan m-file yang telah kita buat. Misalnya kita lihat kembali *function file* harga.m yang menyatakan bila jumlah barang yang dibeli lebih dari 10 akan mendapatkan diskon 20% (lihat subbab 3.6.3).

» barang=100.

```
barang =
    100
```

» hargabrg=100

```
hargabrg =
    100
```

» feval('harga') % memanggil m-file harga.m

```
harga =     8000
```

## 5. <sup>60</sup> Operator Logika dan Relasi

Perbandingan-perbandingan dalam MATLAB dilakukan dengan bantuan operator-operator berikut:

**Tabel 3.4** Operator Logika

Operasi	Simbol
<	Kurang dari
<=	Kurang dari sama dengan
>	Lebih dari
>=	Lebih dari sama dengan
==	Sama dengan
~=	Tidak sama dengan

Operator == membandingkan dua variabel dan mengembalikan yang satu jika mereka sama dan <sup>140</sup> sebaliknya diisi angka nol. Fungsi relasi dan logika serta fungsi-fungsi pengujian dapat dilihat pada tabel berikut ini:

**Tabel 3.5** Tabel Fungsi Logika pada MATLAB

Fungsi Relasi dan Logika	
xor(x,y)	Operasi eksklusif OR. Menghasilkan satu jika salah satu dari

	x atau y tidak nol (benar). Menghasilkan nol jika baik x maupun y adalah nol (salah)/keduanya tidak nol (benar).
any(x)	Menghasilkan satu jika ada elemen dalam suatu vektor yang tidak nol. Menghasilkan satu untuk setiap kolom suatu matriks yang mempunyai elemen tidak nol.
all(x)	Menghasilkan satu jika semua elemen dalam suatu vektor x tidak nol. Menghasilkan satu untuk setiap kolom dalam suatu matriks x yang semua elemennya tidak nol.
<b>Fungsi-fungsi Penguji</b>	
isa(X, 'nama')	Benar jika X mempunyai objek kelas 'nama'
iscell(X)	Benar jika argumennya adalah suatu sel array.
iscellstr(S)	Benar jika argumennya adalah sela suatu array string.
ischar(S)	Benar jika argumennya adalah karakter.
isempty(X)	Benar jika argumennya kosong
isequal(A,B)	Benar jika A dan B identik
isfield(S, 'nama')	Benar jika 'nama' adalah field dari struktur S.
isfinite(X)	Benar jika elemen terhingga.
isglobal(X)	Benar jika argumennya adalah variabel global.
ishandle(h)	Benar jika argumennya adalah objek handle valid.
ishold	Benar jika status hold plot saat ini adalah ON.
isieee	Benar jika komputer melakukan aritmatika IEEE
isinf(X)	Benar jika elemen tak terhingga.
isletter(S)	Benar jika elemen adalah huruf-huruf alfabet.
islogical(X)	Benar jika argumennya adalah array logika.
ismember(A,B)	Benar jika elemen di A juga terdapat di B.
isnan(X)	Benar jika elemen adalah nan.
isnumeric(X)	Benar jika argumen adalah array numerik
Isppc	Benar untuk Macintosh dengan prosesor powerpc
isprime(X)	Benar jika elemen adalah bilangan prima.
isreal(X)	Benar jika argumen tidak mempunyai bagian imajiner.
isspace(S)	Benar jika argumen adalah karakter putih.
issparse(A)	Benar jika argumen adalah matriks jarang.
isstruct(S)	Benar jika argumen merupakan suatu struktur.
isstudent	Benar jika digunakan The Student Edition of MATLAB.
isunix	Benar jika sistem operasinya adalah UNIX.
isvms	Benar jika komputernya adalah VMS.

Berikut adalah contoh penggunaan fungsi logika untuk mencari elemen berangka satu.

```

» a=[ 1 1 2 3 4 7 1 12]

a =
    1    1    2    3    4    7    1   12

» indx=(a==1)      % mencari angka 1 di vektor a
indx =
    1    1    0    0    0    0    1    0

112
» b=a(indx)        % vektor b berisi angka 1 saja dari a
b =
    1    1    1

» indx=find(a==1)  % letak angka 1 pada vektor a
indx =
    1    2    7

```

## 6. String

String adalah sederetan karakter yang bukan angka/numerik. Karakter yang tampak berformat ASCII. Berikut ini adalah contoh penggunaan string:

```

» str='SUHENDAR'
str = SUHENDAR

» str1=double(str)      % merubah string ke numerik
str1 =
Columns 1 through 12
    82    97   104   109    97   100   121    97    32    84   114   105
Columns 13 through 24
    97   115    32    72    97   110   100    97   121    97   110   116
Column 25
    111

```

Tabel berikut ini merangkum fungsi-fungsi pada MATLAB yang berkaitan dengan operasi string.

**Tabel 3.6** Konversi dan Fungsi-fungsi String

<b>Konversi String</b>	
base2dec	string berbasis x ke bilangan bulat desimal
bin2dec	string biner ke bilangan bulat desimal
Char	string ke ASCII
dec2base	bilangan bulat desimal ke string berbasis x
dec2bin	bilangan desimal ke string biner
dec2hex	bilangan desimal ke string heksadesimal
double	ASCII ke string
fprintf	menulis teks ke file atau monitor
hex2dec	string heksadesimal ke bilangan bulat desimal
hex2num	string heksadesimal ke bilangan floating point IEEE
int2str	integer ke string
mat2str	matriks numerik ke string yang dapat dioperasikan oleh fungsi eval
num2str	bilangan ke string
sprintf	bilangan ke string dengan kontrol format
sscanf	konversi string ke bilangan dengan kontrol format
str2num	string ke bilangan
<b>Fungsi-fungsi String</b>	
blank(n)	menghasilkan string dengan n spasi
deblank(s)	menghasilkan karakter blank akhir dari string
eval(string)	mengevaluasi string sebagai perintah MATLAB
feval(f,x,y,...)	mengevaluasi fungsi yang diberikan dengan string
findstr(s1,s2)	menemukan suatu string dalam string yang lain
ischar(s)	benar jika inputnya adalah string
isletter(s)	benar jika terdapat karakter alfabet
isspace(s)	benar jika terdapat karakter putih
lasterr	string pesan kesalahan MATLAB terakhir
lower(s)	mengkonversi string ke huruf kecil
strcat(s1,s2,...)	melakukan penggabungan string horisontal
strcmp(s1,s2)	benar jika kedua string identik
strjust(s)	membuat string rata kanan
strmatch(s1,s2)	menemukan kemungkinan kesesuaian string
strncmp(s1,s2,n)	benar jika n karakter pertama identik
strrep(s1,s2)	mengganti suatu string dengan string yang lain
strtok(s)	menemukan token pertama dalam suatu string
strvcat(s1,s2,...)	menggabungkan string secara vertikal
upper(s)	konversi string ke huruf kapital

## PERTANYAAN DAN SOAL

---

1. Jelaskan fungsi matematika dalam MATLAB!
2. Tuliskan contoh persamaan matematika dan gambarkan dalam bentuk grafik pada pemrograman MATLAB!
3. Hitunglah invers dari variabel X dalam matriks berikut menggunakan pemrograman MATLAB:

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 8 & 9 & 0 \end{bmatrix} \cdot \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix} = \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix}$$

## PEMROGRAMAN VISUAL MATLAB

20

### TUJUAN INSTRUKSIONAL:

Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:

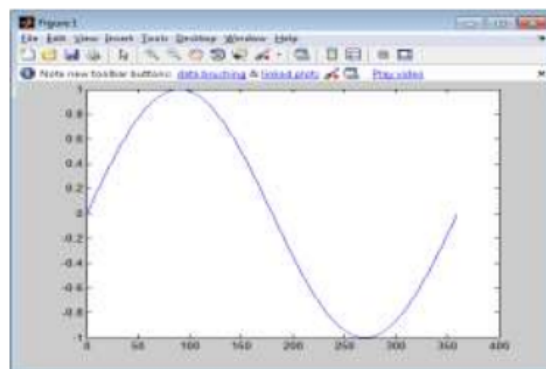
- 1) Mengerti konsep dasar pemrograman visual berbantu MATLAB
- 2) Mengerti konsep dasar pemrograman visual 2D berbantu MATLAB
- 3) Mengerti konsep dasar pemrograman visual 3D berbantu MATLAB

### A. VISUALISASI DASAR

MATLAB memiliki kemampuan dalam memvisualisasikan data numeric. Ilustrasi yang dapat ditampilkan menggunakan pemrograman MATLAB diperlihatkan dalam bentuk grafik satu, dua, atau tiga dimensi. Sebagai contoh sederhana, pada *Command Prompt* MATLAB, dapat dituliskan *script* sebagai berikut:

```
>> sudut=[0:360];
>>y=sin(sudut*pi/180);
>>plot(sudut,y)
```

Maka dengan menambahkan perintah `plot` pada baris ketiga pada keseluruhan *script*, MATLAB akan menampilkan grafik satu dimensi seperti gambar berikut.

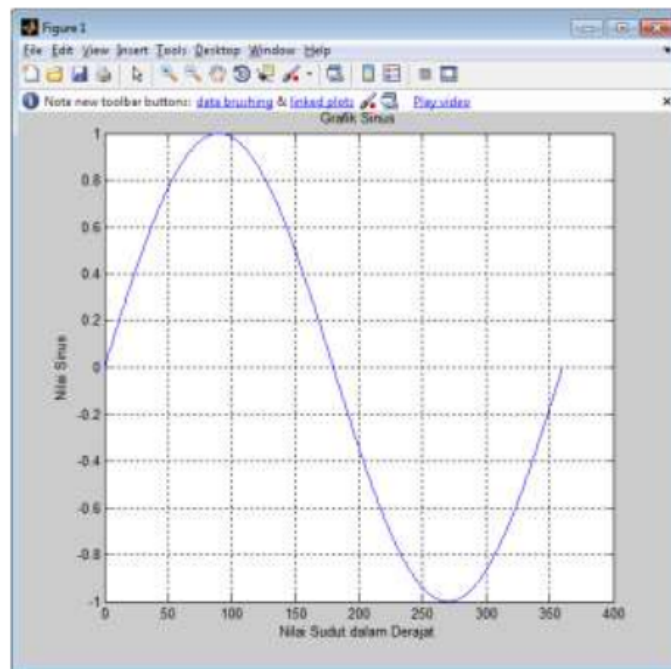


**Gambar 4.1** Tampilan Gambar satu Dimensi untuk Grafik Sinusoidal

Cara menampilkan grafik menggunakan bahasa pemrograman MATLAB, jauh lebih sederhana dibanding dengan menggunakan bahasa pemrograman lain seperti Delphi atau Visual Basic.

Tampilan gambar 4.1 dapat dilengkapi dengan judul, label, dan memunculkan grid pada grafik dengan menambahkan beberapa script pada baris-baris berikutnya seperti contoh berikut.

```
>> sudut=[0:360];  
>>y=sin(sudut*pi/180);  
>>plot(sudut,y)  
>>title('Grafik Sinus');  
>>xlabel('Nilai Sudut dalam Derajat');  
>>ylabel('Nilai Sinus');  
>>grid on;
```



**Gambar 4.2** Tampilan Grafik dilengkapi Judul, x dan y Label, Grid On

Selain itu, tampilan grafik dapat kita tambahkan dengan beberapa karakter lainnya seperti simbol untuk mengubah garis, tanda, dan warna pada kurva grafik. Perintah untuk memplot beberapa variable pada grafik yang dilengkapi dengan simbol menggunakan script dasar berikut:

```
plot(x,y,simbol)
```

Beberapa simbol yang disediakan oleh MATLAB, disajikan seperti dalam tabel berikut.

**Tabel 4.1** Simbol untuk memformat tampilan grafik MATLAB

SIMBOL	GARIS	TANDA	WARNA
-	Solid line		
:	Dotted line		
-.	Dash dot line		
--	Dash line		
.		Point	
o		Circle	
x		Cross	
+		Plus sign	
*		Asterisk	
s		Square	
d		Diamond	
v		Triangle (down)	
^		Triangle (up)	
<		Triangle (left)	
>		Triangle (right)	
p		Pentagram	
h		Hexagram	
b			Blue
g			Green
r			Red
c			Cyan
m			Magenta
y			Yellow
k			Black
k			white

Sedangkan tampilan huruf dalam grafik dapat diformat menggunakan sintak dasar berikut.

```
set(gca,fontProperty,value)
```

Beberapa jenis perintah fontproperty yang dapat digunakan, dituliskan pada tabel berikut.



**Tabel 4.2** Daftar fonproperty dalam pemrograman MATLAB

Font Property	Value
Fontname	Arial, Time new roman, dan lain-lain
Fontweight	Bold, normal
FontSize	1, 2, 3, 4 s.d. 30
fontangle	Italic, normal

Simbol-simbol dan beberapa fontproperty di atas dapat diimplementasikan dalam pemrograman MATLAB sebagai berikut.

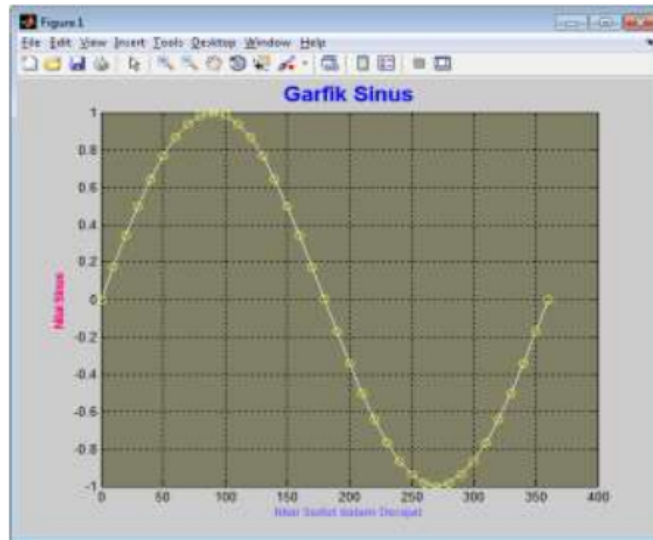
1. Buka MATLAB
2. Pada command window, ketik: `>>edit`, kemudian tekan enter maka akan muncul MATLAB editor
3. Pada MATLAB editor, ketik script program berikut:

```
Sudut=[0:10:360];  
y=sin(sudut*pi/180);
```

```
plot(sudut,y,'oy');  
hold on  
plot(sudut,y,'-w');  
grid on;
```

```
set(gca,'color',[0.5 0.5 0.4],'fontname','arial');  
title('Garfik Sinus','color',[0 0 1],'fontweight','bold','fontsize',16);  
xlabel('Nilai Sudut dalam Derajat','color',[.5 .4 1],'fontweight', ...  
    'bold','fontsize',9);  
ylabel('Nilai Sinus','color',[1 0 .5],'fontweight','bold','fontsize',9);
```

4. Script pada nomor 3 disimpan dengan nama file **latihangrafik1.m**
5. Pada command window MATLAB, panggil nama file tersebut dengan mengetiknya sebagai berikut  
`>> latihangrafik1`
6. Kemudian tekan enter untuk menjalankan program dan menghasilkan tampilan seperti grafik berikut ini.



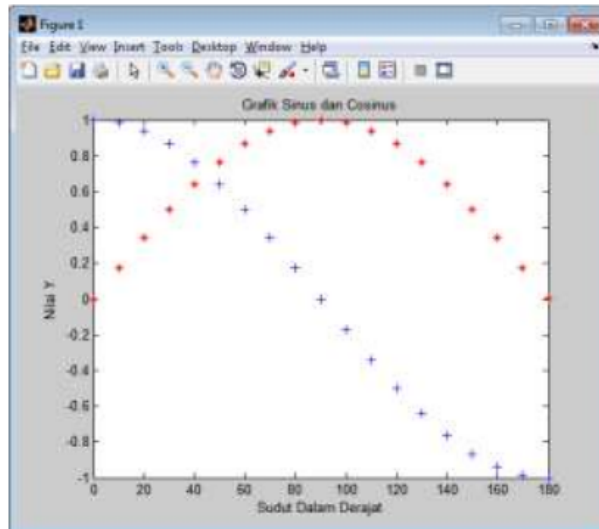
**Gambar 4.3** Tampilan Grafik dilengkapi beberapa Properti

Grafik pada MATLAB dapat juga ditampilkan dalam bentuk overlay dan parallel menggunakan script-script yang dituliskan dalam command window ataupun dalam editornya MATLAB seperti contoh berikut.

#### 1. Menampilkan grafik overlay

```
>> sudut=[0:10:180];
>> y1=sin(sudut*pi/180);
>> y2=cos(sudut*pi/180);
>> plot(sudut,y1,'r');
>> hold on;
>> plot(sudut,y2,'b');
>> title('Grafik Sinus dan Cosinus');
>> xlabel('Sudut Dalam Derajat');
>> ylabel('Nilai Y');
```

Maka akan menampilkan grafik berikut.



**Gambar 4.4** Tampilan Grafik Overlay

## 2. Menampilkan grafik parallel

Perintah atau sintak dasar untuk menampilkan grafik secara parallel dalam MATLAB adalah:

**subplot(n\_baris,n\_kolom,i\_grafik)**

Pada editor MATLAB tuliskan script berikut ini.

```
4
sudut=[0:10:180];
y1=sin(sudut*pi/180);
y2=cos(sudut*pi/180);
y3=y1+y2
```

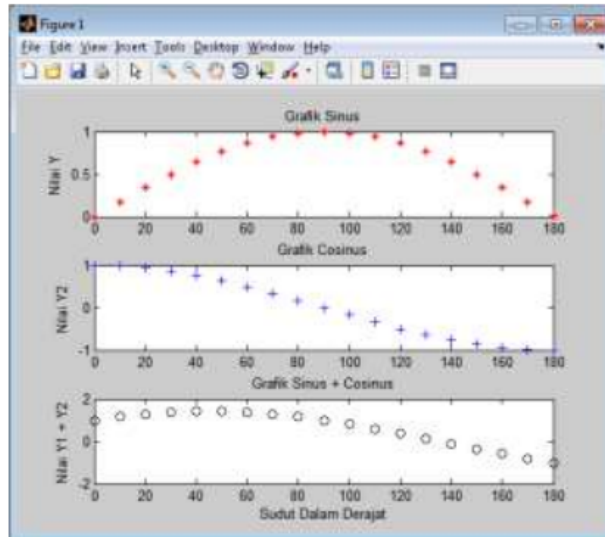
```
subplot(3,1,1);
plot(sudut,y1,'*r');
title('Grafik Sinus');
ylabel('Nilai Y');
```

```
subplot(3,1,2);
plot(sudut,y2,'+b');
title('Grafik Cosinus');
ylabel('Nilai Y2');
```

```
subplot(3,1,3);
plot(sudut,y3,'ok');
title('Grafik Sinus + Cosinus');
```

```
xlabel('Sudut Dalam Derajat');
ylabel('Nilai Y1 + Y2');
```

Maka akan tampil grafik sebagai berikut.



**Gambar 4.5** Tampilan Grafik Paralel

## B. VISUALISASI 2 DIMENSI

Ketika kita dapat menempatkan satu atau lebih titik data pada bidang kartesian atau polar menggunakan bahasa pemrograman MATLAB, maka tampilan seperti itu dinamakan visualisasi data dua dimensi (2D). Visualisasi data 2D dapat dibuat dalam bentuk batang (bar), stem, tangga (stairs), dan lain-lain.

Perintah dasar untuk menampilkan grafik 2D pada masing-masing bentuk adalah:

```
190 bar(x,y); stem(x,y); stairs(x,y)
```

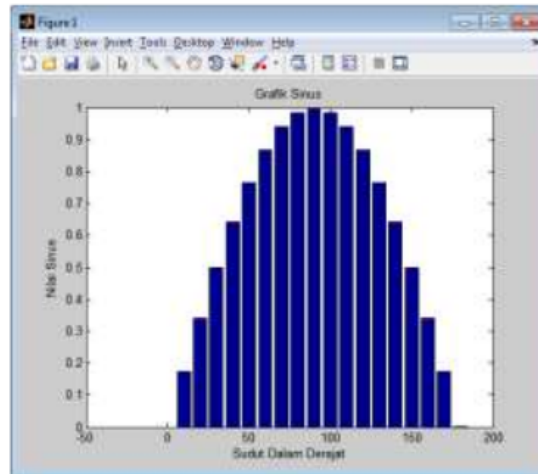
### 1. Menampilkan Grafik 2D Dalam Bentuk Batang

Pada command window MATLAB tuliskan script seperti berikut.

```
>> sudut=[0:10:180];
>> y1=sin(sudut*pi/180);
>> bar(sudut,y)
>> bar(sudut,y1)
>> title('Grafik Sinus');
>> xlabel('Sudut Dalam Derajat');
```

```
>> ylabel('Nilai Sinus');
```

Maka akan menghasilkan tampilan grafik batang sebagai berikut.



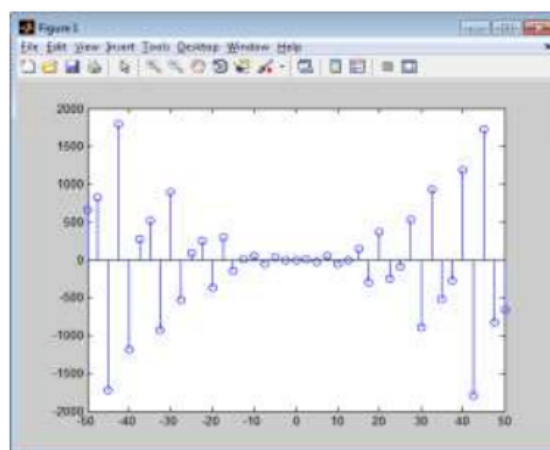
**Gambar 4.6** Tampilan Grafik 2D bentuk Batang (bar)

## 2. Menampilkan Grafik 2D Dalam Bentuk Stem

Pada command window MATLAB tuliskan script seperti berikut.

```
>>x=[-50:2.5:50];  
>> y1=(x.^2).*sin(x);  
>> stem(x,y1);
```

Maka akan menghasilkan tampilan grafik stem sebagai berikut.



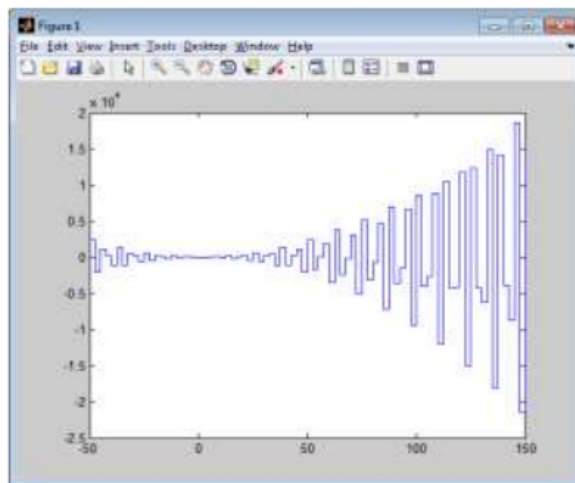
**Gambar 4.7** Tampilan Grafik 2D bentuk Stem

### 3. Menampilkan Grafik 2D Dalam Bentuk Stem

Pada command window MATLAB tuliskan script seperti berikut.

```
>>x=[-50:2.5:150];  
>> y1=(x.^2).*cos(x);  
>> stairs(x,y1);
```

Maka akan menghasilkan tampilan grafik stem sebagai berikut.



**Gambar 4.8** Tampilan Grafik 2D bentuk Tangga (Stairs)

### C. VISUALISASI 3 DIMENSI

Tampilan grafik 3 dimensi dalam MATLAB sangat penting untuk dipelajari. Teknik ini berguna terutama dalam bidang pemodelan yang ada kaitannya dengan data dalam bentuk ruang. Visualisasi data 3 dimensi akan menempatkan satu atau lebih titik data pada suatu media berbentuk ruang. Data ditampilkan dalam bentuk ruang  $(x,y,z)$ . Nilai  $z$  adalah besarnya nilai bobot pada titik  $(x,y)$ . Nilai ini dapat berupa ketinggian, temperature, kecepatan, dan besaran-besaran lainnya.

Beberapa perintah atau sintaks dasar dan sintak bantu lainnya untuk menampilkan grafik 3 dimensi dalam MATLAB, perintah dan fungsinya masing-masing dituliskan seperti dalam tabel berikut.

**Tabel 4.3** Sintak Untuk Menampilkan Grafik 3 Dimensi Dalam MATLAB

Sintak	Fungsi dan Kegunaan
plot3(x,y,z)	Untuk menampilkan data dalam ruang 3 dimensi pada

	koordinat x,y, dan z
mesh(z)	Untuk menampilkan data dalam ruang 3 dimensi berupa grid-grid yang menghubungkan antara titik
surf(z)	Untuk menampilkan data dalam ruang 3 dimensi berupa grid-grid dengan warna tertentu sesuai bobot yang diberikan
surfl(z)	Untuk menampilkan data dalam ruang 3 dimensi berupa grid-grid dengan warna tertentu berdasarkan teknik pencahayaan
pcolor(z)	Untuk menampilkan data dalam ruang 3 dimensi dalam bentuk permukaan 2 dimensi (tampak atas) berupa grid-grid dengan warna tertentu sesuai bobot yang diberikan
imagesc(z)	Untuk menampilkan data dalam ruang 3 dimensi dalam bentuk permukaan 2 dimensi (tampak atas) berupa grid-grid dengan menggunakan interpolasi warna
oh=contour(z) clabel(oh)	Untuk membuat garis kontur data 3 dimensi
oh=contourf(z) clabel(oh)	Untuk membuat garis kontur data 3 dimensi, pada setiap kontur diberikan warna sesuai bobot konturnya

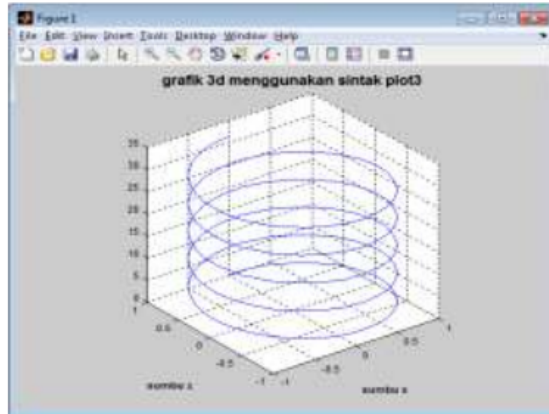
Sebagai contoh, tuliskanlah script berikut dalam editor MATLAB. Pada command window MATLAB ketikkan perintah edit kemudian enter.

```
>>edit
```

**Contoh 1:**

```
104 t = 0:pi/50:10*pi;
plot3(sin(t),cos(t),t)
grid on
axis square
set(gca,'fontsize',8);
76 title('grafik 3d menggunakan sintak plot3','fontweight','bold','fontsize',12);
xlabel('sumbu x','fontweight','bold');
ylabel('sumbu y','fontweight','bold');
ylabel('sumbu z','fontweight','bold');
```

maka tampilan grafik yang muncul seperti gambar berikut.



**Gambar 4.9** Tampilan Grafik 3D plot3 untuk Contoh 1

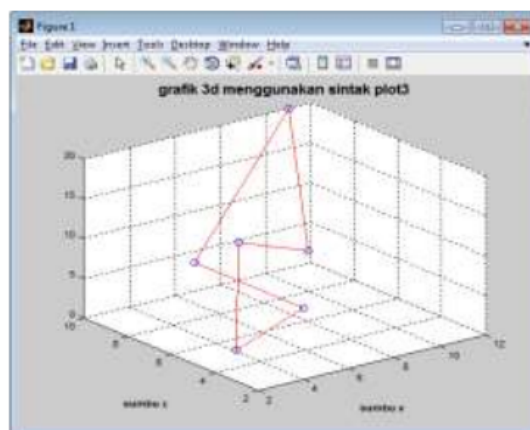
**Contoh 2:**

```

74 [5 6 12;2 3 4; 4 2 9; 4 7 9;11 10 20;9 7 7;5 6 12];
plot3(t(:,1),t(:,2),t(:,3),'ob');
hold on;
plot3(t(:,1),t(:,2),t(:,3),'-r');
grid on;
set(gca,'fontsize',8);
76 title('grafik 3d menggunakan sintak plot3','fontweight','bold','fontsize',12);
xlabel('sumbu x','fontweight','bold');
ylabel('sumbu y','fontweight','bold');
ylabel('sumbu z','fontweight','bold');

```

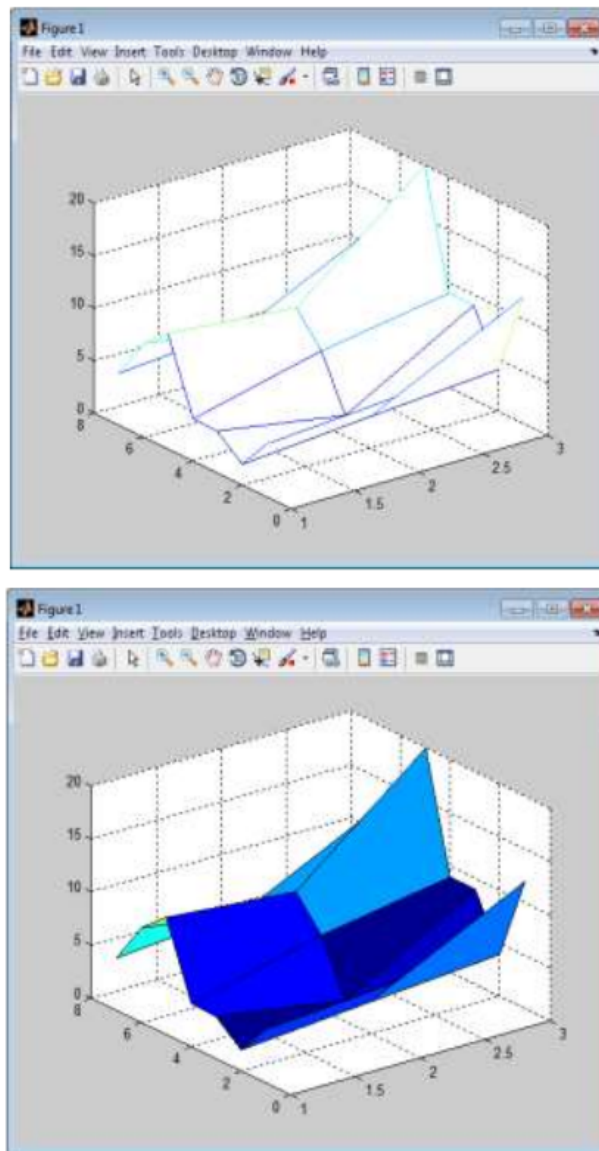
maka tampilan grafik yang muncul seperti gambar berikut.



**Gambar 4.10** Tampilan Grafik 3D plot3 untuk Contoh 2

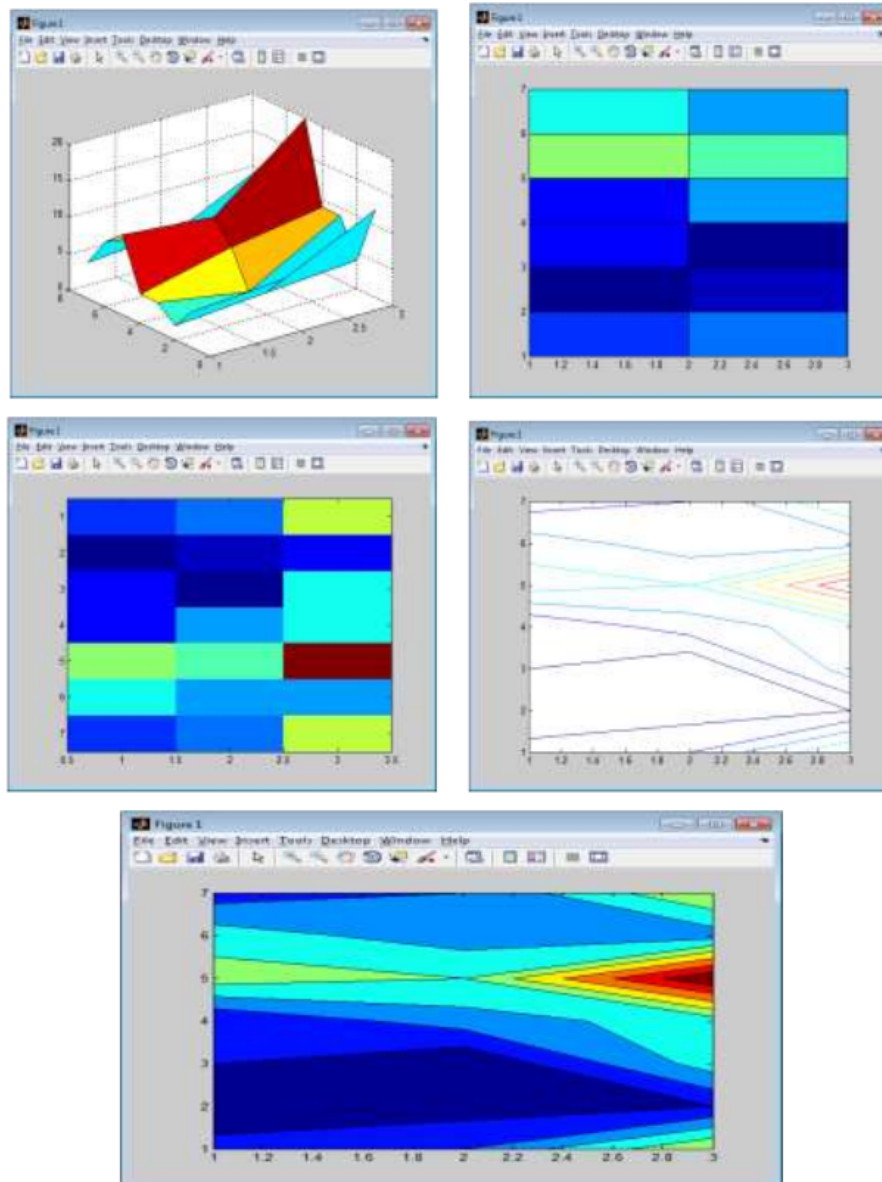


Sedangkan sintak mesh dan surf dengan script seperti pada contoh 2 akan menghasilkan tampilan gambar seperti berikut.



**Gambar 4.11** Tampilan Grafik 3D untuk Sintak mesh dan surf

Perintah atau sintak-sintak lainnya (surfl, pcolor, imagesc, contour, dan contour) jika diujicobakan dengan script pada contoh 2, maka akan menghasilkan tampilan grafik secara berturut-turut seperti gambar berikut.



**Gambar 4.12** Tampilan Grafik 3D untuk Sintak `surf`, `pcolor`, `imagesc`, `contour`, dan `contour`

## PERTANYAAN DAN SOAL

---

1. Buatlah grafik fungsi cosines dan tangent!
2. Buatlah grafik fungsi cosines dan tangent dalam bentuk 2D!
3. Buatlah grafik fungsi cosines dan tangent dalam bentuk 3D!

## BAB 5

### KOMPUTASI KENDALI DASAR

---

20

#### TUJUAN INSTRUKSIONAL:

Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:

- 1) Mempraktekkan cara membuka jendela Simulink
  - 2) Menjelaskan cara menggambar model pada jendela Simulink
  - 3) Menjelaskan kembali fungsi penggunaan Simulink dalam respon kendali
  - 4) Memodelkan Transformasi Laplace menggunakan Simulink MATLAB
  - 5) Memodelkan Respon Transien menggunakan Simulink MATLAB
  - 6) Memodelkan Kesalahan Tunak menggunakan Simulink MATLAB
  - 7) Memodelkan Analisa Stabilitas Menggunakan Simulink MATLAB
- 

#### A. MEMBUKA JENDELA SIMULINK

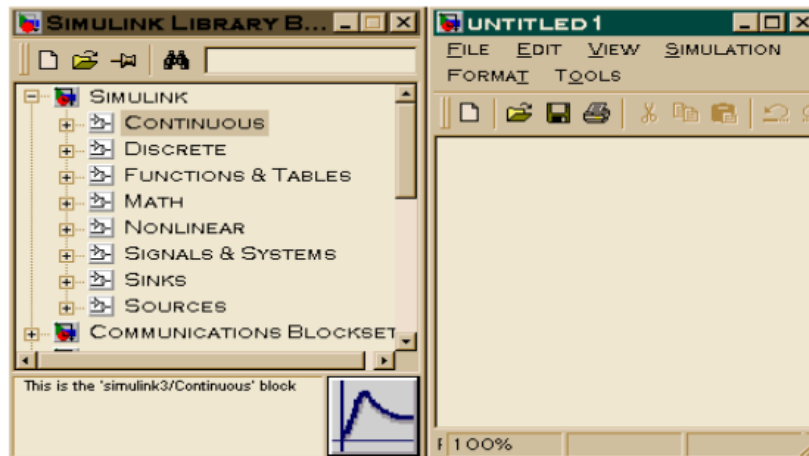
SIMULINK yang masih satu paket dengan MATLAB berguna untuk mempermudah dan memperbaiki visualisasi pemrograman. Sesuai dengan makna katanya, SIMULINK bermanfaat dalam mensimulasikan sistem kontrol yang kita analisa. Versi terakhir SIMULINK saat tulisan ini dibuat adalah versi 6 yang satu paket dengan MATLAB 7 R-14.

SIMULINK dibuka dengan mengklik **File – New – Model**.



### Gambar 5.1 Membuka Jendela SIMULINK

Kemudian muncul dua jendela yaitu Simulink Library Browser dan Jendela Model.



Gambar 5.2 Jendela SIMULINK

Untuk membuka SIMULINK browser kita dapat juga dengan cara mengklik icon bergambar :



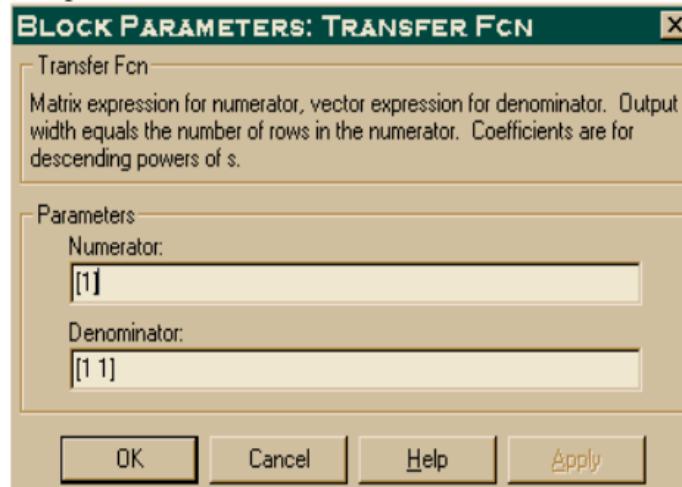
Pada jendela Simulink Library Browser terdapat fasilitas-fasilitas yang berguna sesuai dengan bidang ilmu kita. Kita dapat melihat apa yang tersedia pada tiap-tiap bagian Simulink Browser dengan cara mengklik tanda '+' yang ada di sebelah kiri nama fungsi. Misalnya setelah kita mengklik tanda '+' pada bagian SIMULINK akan muncul continuous, discrete, function and tables dan sebagainya.

### B. MENGGAMBAR MODEL

Model kita gambar dalam jendela Model dengan cara klik – drag simbol komponen dari jendela Simulink Library Browser. Misal kita ingin meletakkan diagram blok pada jendela model. Setelah tanda '+' di sebelah kiri kata SIMULINK kita klik lagi tanda '+' di bagian CONTINUOUS yang akan memunculkan DERIVATIVE, INTEGRATOR dan lain-lain termasuk di dalamnya TRANSFER FUNCTION.

Pindahkan dengan cara klik – drag ke jendela Model. Untuk mengisi harga fungsi alihnya, double klik pada kotak fungsi alih yang akan memunculkan menu input

fungsi alih. Selain dalam format fungsi alih, tersedia pula format lainnya misalnya ruang keadaan dan pole-zero.



**Gambar 5.3** Menu Input Fungsi Alih

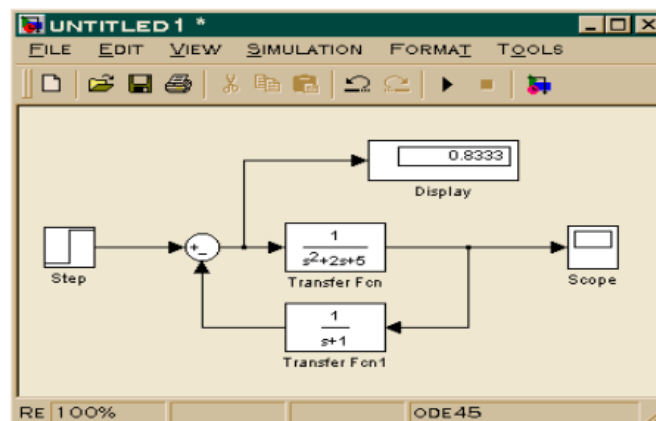
Silahkan Anda membuka-buka seluruh menu yang tersedia pada Simulink Library Browser agar mengetahui apa saja yang tersedia di SIMULINK.

### C. PENGGUNAAN SIMULINK DALAM ANALISA RESPON KENDALI

Misalnya kita diminta menganalisa respon transien dan kesalahan tunak masukan step sistem lingkaran tertutup yang berfungsi alih:

$$G(s) = \frac{1}{s^2 + 2s + 5}$$

dengan umpan balik menggunakan sensor yang berfungsi alih:  $H(s) = \frac{1}{s+1}$ .

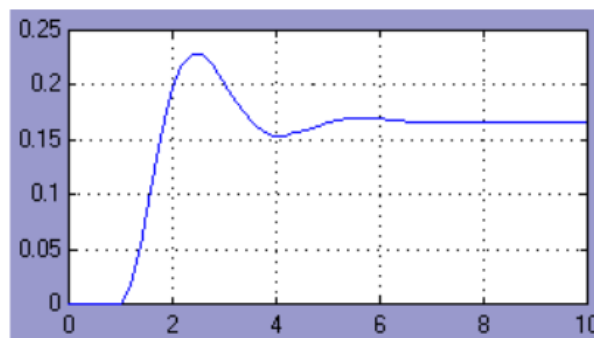


### Gambar 5.4 Penggambaran Sistem dengan Jendela Model

Mula-mula buat blok fungsi alih dan umpan balik, double klik untuk memasukkan harganya. Display dihubungkan dengan keluaran titik jumlah guna mengetahui besar kesalahan tunaknya. Scope dipasang di keluaran guna mengetahui respon transien sistem. Baik display maupun scope di klik – drag dari **SIMULINK – SINK**. Fungsi alih sensor dibalik dengan cara klik kanan pada **blok – FORMAT – FLIP BLOCK**. Masukan STEP di ambil dari **SIMULINK – SOURCES – STEP**.

Dan yang terakhir titik jumlah diambil dari **SIMULINK – MATH – SUM**. Jangan lupa karena default SUM adalah menjumlah (++) maka agar keluaran mengurangi masukan kita double klik pada titik jumlah lalu diedit agar dihasilkan (+-).

Setelah tombol ►/RUN ditekan diperoleh harga kesalahan tunak pada display sebesar 0,8333 dan untuk mengetahui respon sistem, double klik pada SCOPE setelah itu. Dan klik gambar teropong untuk melihat lebih dekat.



Gambar 5.5 Respon Sistem

Pada grafik di atas terbukti bahwa kesalahan tunak yang terjadi adalah  $1 - 0,17 = 0,83$  yang sama dengan tampilan pada kotak display di jendela model. Dari materi pada bab VI Anda tentu dapat mencari harga-harga karakteristik sistem tersebut yaitu  $t_d = 1,5$  detik,  $t_r = 1,9$  detik,  $t_p = 2,5$  detik,  $M_p = 0,045$  dan  $t_s = 4$  detik. Serta sistem tersebut stabil karena keluaran tidak terus membesar.

#### D. MODEL TRANSFORMASI LAPLACE

Sistem dinamik melibatkan kecepatan dan percepatan sehingga persamaan matematikanya akan melibatkan diferensiasi. Persamaan matematik itu disebut persamaan diferensial. Ada banyak cara menyelesaikan persamaan diferensial, misalnya metode operator D, bernoulli dan sebagainya. Salah satu cara yang sering dipakai adalah transformasi laplace. Transformasi laplace mampu merubah persamaan diferensial menjadi persamaan aljabar biasa dalam variabel laplace.

Kemudian setelah persamaan itu diselesaikan, dilakukan transformasi balik menjadi persamaan dalam variabel semula.

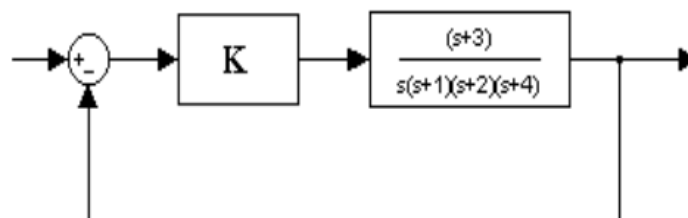
## 1. Fungsi Alih

Fungsi alih atau fungsi transfer adalah perbandingan keluaran terhadap masukan dalam variabel laplace. Variabel laplace diberi simbol  $s$  yang didefinisikan:  $s = \sigma + j\omega$  dimana  $\sigma$  bagian real dan  $j\omega$  bagian imajiner. Disebut fungsi alih karena mengalihkan/mentransfer masukan menjadi keluaran tertentu.

## 2. Diagram Blok

Diagram blok menggambarkan aliran proses sistem kontrol yang berupa kotak dan anak panah. Anak panah menggambarkan arah proses yang menggambarkan besaran fisika misalnya listrik, panas dan lain-lain tergantung sistem fisiknya. Kotak merupakan pemroses sinyal yang masuk menjadi sinyal keluar yang merupakan hasil pemodelan matematik dalam variabel laplace.

Terdapat dua titik pada diagram blok yaitu titik jumlah dan titik pisah. Titik jumlah menjumlah/mengurangi dua sinyal sedangkan titik pisah tidak melakukan operasi hanya membagi dua sinyal masing-masing identik (tidak bertambah/berkurang).



**Gambar 5.6** Diagram Blok

Pada sistem kontrol klasik, transformasi laplace sangat terkenal karena mampu merubah persamaan diferensial yang rumit menjadi persamaan aljabar biasa. Awalnya kita mentransformasikan persamaan dalam variable  $t$  dengan rumus transformasi:

$$\mathcal{L}\{f(t)\} = F(s) = \int_0^{\infty} f(t)e^{-st} dt$$

Tetapi kita tidak perlu mengintegalkan semua fungsi yang terlibat dalam persamaan diferensial. Kita tinggal melihat pada tabel bentuk yang sesuai dicari.

**Tabel 5.1** Transformasi Laplace

No	$f(t)$	$F(s)$
1.	Impuls satuan $\delta(t)$	1
2.	Langkah satuan $1(t)$	$1/s$
3.	$T$	$1/s^2$
4.	$\frac{t^{n-1}}{(n-1)!}$ ( $n=1,2,3,\dots$ )	$1/s^n$
5.	$t^n$ ( $n=1,2,3,\dots$ )	$\frac{n!}{s^{n+1}}$
6.	$e^{-at}$	$\frac{1}{s+a}$
7.	$te^{-at}$	$\frac{1}{(s+a)^2}$
8.	$\frac{1}{(n+1)!} t^{n+1} e^{-st}$ ( $n=1,2,3,\dots$ )	$\frac{1}{(s+a)^{n+1}}$
9.	$t^n e^{-st}$ ( $n=1, 2, 3, \dots$ )	$\frac{n!}{(s+a)^{n+1}}$
10.	$\sin \omega t$	$\frac{\omega}{s^2 + \omega^2}$
11.	$\cos \omega t$	$\frac{s}{s^2 + \omega^2}$
12.	$\sinh \omega t$	$\frac{\omega}{s^2 - \omega^2}$
13.	$\cosh \omega t$	$\frac{s}{s^2 - \omega^2}$
14.	$\frac{1}{a} (1 - e^{-at})$	$\frac{1}{s(s+a)}$
15.	$\frac{1}{(b-a)} (e^{-at} - e^{-bt})$	$\frac{1}{(s+a)(s+b)}$
16.	$\frac{1}{(b-a)} (be^{-bt} - ae^{-at})$	$\frac{s}{(s+a)(s+b)}$
17.	$\frac{1}{ab} \left[ 1 + \frac{1}{(b-a)} (be^{-at} - ae^{-bt}) \right]$	$\frac{1}{s(s+a)(s+b)}$
18.	$e^{-st} \sin \omega t$	$\frac{\omega}{(s+a)^2 + \omega^2}$
19.	$e^{-st} \cos \omega t$	$\frac{s+a}{(s+a)^2 + \omega^2}$



20.	$\frac{\omega_n}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t} \sin \omega_n \sqrt{1-\xi^2} t$	$\frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$
21.	$-\frac{1}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1-\xi^2} t - \phi)$ $\phi = \tan^{-1} \frac{\sqrt{1-\xi^2}}{\xi}$	$\frac{s}{s^2 + 2\xi\omega_n s + \omega_n^2}$
22.	$1 - \frac{1}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1-\xi^2} t + \phi)$ $\phi = \tan^{-1} \frac{\sqrt{1-\xi^2}}{\xi}$	$\frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$
23.	$1 - \cos \omega t$	$\frac{\omega^2}{s(s^2 + \omega^2)}$
24.	$\omega t - \sin \omega t$	$\frac{\omega^3}{s^2(s^2 + \omega^2)}$
25.	$\frac{d}{dt} f(t)$	$sF(s) - f(0+)$
26.	$\int_{0+}^t f(t) dt$	$F(s)/s$

Dengan bantuan MATLAB kita dapat langsung menghitung transformasi laplace berbagai macam fungsi, dengan cara sebagai berikut:

Sintaks:  $L=Laplace(F)$   
 $L=Laplace(F,t)$   
 $L=Laplace(F,w,z)$

» `laplace(exp(-a*t).*sin(4*t))`

ans =

$4/((s+a)^2+16)$

» `pretty(ans)`

4  
-----  
(s + a)<sup>2</sup> + 16

Kebalikan transformasi laplace adalah invers transformasi laplace yang mengembalikan hasil operasi matematika dalam variabel s menjadi variabel semula yaitu variabel t.

Sintaks:

```

18
F=iLaplace(L)
F=iLaplace(L,t)
F=iLaplace(L,y,x)

```

Berikut ini contoh seperti pada tabel laplace no. 6:

```

18
» syms s a
» y=ilaplace(1/(s+a))
y =
exp(-a*t)

```

Berikut ini dirangkum transformasi-transformasi integral lainnya yang tersedia di MATLAB.

**Tabel 5.2** Transformasi Integral

Transformasi Integral	
Fourier	Transformasi Fourier
Laplace	Transformasi Laplace
Ztrans	Transformasi Z
Ifourier	Invers Transformasi Fourier
Ilaplace	Invers Transformasi Laplace
Iztrans	Invers Transformasi Z

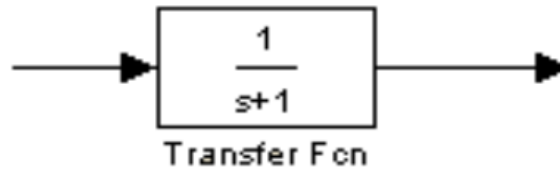
## E. RESPON TRANSIEN

Respon transien adalah reaksi awal sistem kontrol terhadap masukan tertentu. Transien berarti transisi dari keadaan sebelum menjadi keadaan setelah diberi masukan. Jenis masukan yang sering digunakan untuk menguji respon transien sistem antara lain:

- Masukan Impu  $r(t)=\delta(t)$
- Masukan Step,  $r(t)=1$
- Masukan Ramp,  $r(t)=t$
- Masukan Sinusoida,  $r(t)=\sin \omega t$

Sebagian besar alat bekerja dengan respon step, misalnya AC, penggerak antenna dan mekanisme servo. Pada sistem kontrol elektronik biasanya melibatkan frekuensi sehingga masukan yang tepat untuk analisa adalah sinus.

**Contoh sistem orde-1.**



**Gambar 5.7** Sistem Orde-1

Berikut ini contoh penulisan fungsi alih dalam MATLAB.

```

» num=[0 1];           % cara pertama menulis fungsi alih
» den=[1 1];
» g=tf([1],[1 1])     % cara kedua menulis fungsi alih
    
```

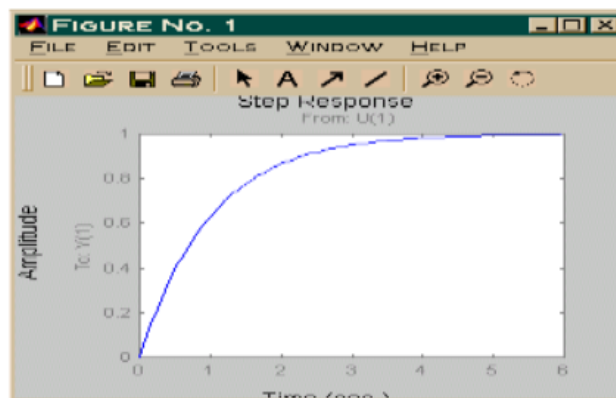
Transfer function:

$$\frac{1}{s + 1}$$

```

» step(num,den)       % masukan step
» step(g)             % masukan step untuk cara kedua
    
```

Pada jendela grafik akan muncul grafik sebagai berikut:



**Gambar 5.8** Respon Sistem Contoh Terhadap Masukan Step

Cara praktis menganalisa tanggapan step suatu sistem adalah dengan menganggap kondisi awal sistem nol. Lima karakteristik pada tanggapan transien adalah:

1. Waktu tunda,  $t_d$ , yaitu waktu yang dibutuhkan sistem naik setengah dari set point.
2. Waktu naik,  $t_r$ , yaitu waktu yang dibutuhkan sistem naik menyentuh 90%-100% set point.
3. Waktu puncak,  $t_p$ , yaitu waktu yang dibutuhkan saat sistem menyentuh nilai tertinggi.
4. Overshoot Maksimum,  $M_p$ , Jarak lebih respon sistem terhadap set point atau biasanya dinyatakan dalam prosentase.
5. Waktu turun,  $t_s$ , adalah waktu yang ditempuh sistem untuk mencapai harga 2%-5% kesalahan terhadap setpoint.

Pembaca tentu saja dapat menjawab lima karakteristik sistem dengan melihat grafik respon pada gambar VI.2.  $t_d=1s$ ,  $t_r=2s$ ,  $t_s=3,5s$  dan tidak ada overshoot.

Bagaimana dengan masukan ramp, sinus dan masukan lain? Selain step dan impulse, kita harus memodifikasi fungsi alih. Misalnya kita ingin mengetahui respon ramp. Berdasarkan teori, fungsi alih adalah perbandingan keluaran dengan masukan. Jadi keluaran adalah masukan dikali fungsi alih. Karena masukan ramp= $1/s^2$ , kita dapat melihat respon dengan:

1. Masukan impulse terhadap fungsi alih yang dikali dengan  $1/s^2$
2. Dengan masukan step terhadap fungsi alih yang dikali dengan  $1/s$ .

Berikut ini cara melihat respon ramp dengan cara kedua.

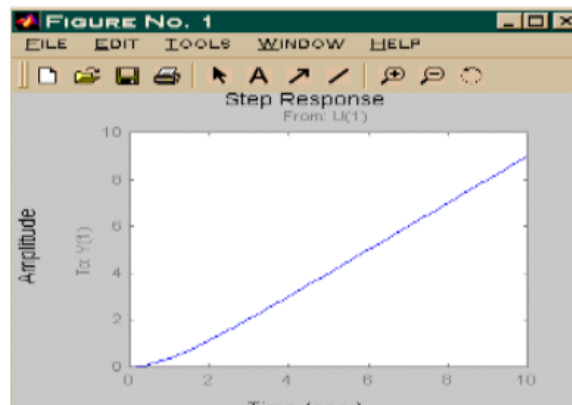
»  $g=tf([1],[1 1 0])$  % fungsi dikali  $1/s$

Transfer function:

$$\frac{1}{s^2 + s}$$

»  $step(g)$  % walau step, sebenarnya ini adalah masukan ramp

Diperoleh grafik:



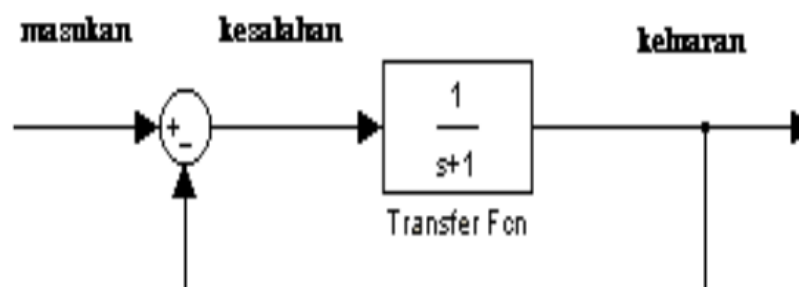
**Gambar 5.9** Grafik Sistem Respon Ramp yang memanfaatkan fungsi Step

Di bab VII nanti Anda akan menemukan fasilitas dalam MATLAB yang lebih mudah dalam menganalisa respon transien sistem.

#### F. KESALAHAN TUNAK

Kesalahan adalah penyimpangan sistem kontrol dari set point sedangkan tunak (*steady state*) adalah kondisi mantap yang secara matematis dinyatakan limit mendekati waktu tak hingga. Secara praktek, kesalahan tunak berarti kesalahan yang muncul dengan cara mengukur kesalahan sistem setelah menunggu sistem itu diam/mantap.

Kita menghitung kesalahan tunak secara matematik dengan teorema limit dari selisih masukan dengan keluaran. Kita ambil contoh menghitung kesalahan tunak sistem di bawan ini.



**Gambar 5.10** Kesalahan Tunak  $1/(s+1)$

Fungsi alih lingkaran tertutup gambar di atas adalah:

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

$$\frac{E(s)}{R(s)} = \frac{1}{1+G(s)} \quad E(s) = \frac{R(s)}{1+G(s)}$$

Dengan teorema final value problem dalam Laplace diperoleh:

$$e(\infty) = \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sF(s) = \lim_{s \rightarrow 0} \frac{sR(s)}{1+G(s)}$$

Bila pada soal kita ingin menganalisis kesalahan tunak sistem untuk masukan step  $R(s)=1/s$  didapat:

$$e(\infty) = \lim_{s \rightarrow 0} \frac{sR(s)}{1+G(s)} = \lim_{s \rightarrow 0} \frac{s(1/s)}{1 + \frac{1}{s+1}} = 0,5$$

Sekarang mari kita coba dengan MATLAB. Fungsi alih lingkaran tertutup sistem di atas adalah :

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1+G(s)} = \frac{\frac{1}{s+1}}{1 + \frac{1}{s+1}} = \frac{1}{s+2}$$

Coba analisa respon sistem untuk masukan step  $R(s)=1/s$

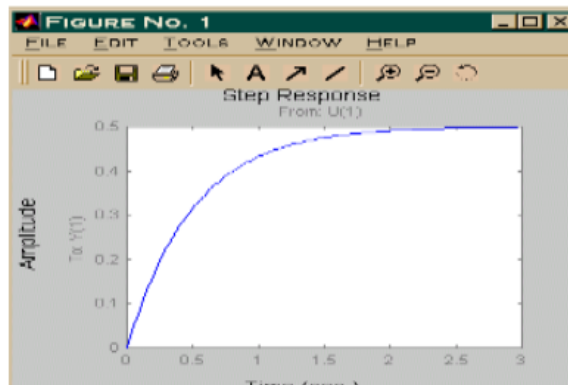
» `gclose=tf([1],[1 2])` % fungsi alih lingkaran tertutup system

Transfer function:

$$\frac{1}{s+2}$$

» `step(gclose)`

Dan diperoleh grafik:



**Gambar 5.11** Grafik Respon Sistem  $1/(S+1)$  Lingkaran tertutup

Kalau Anda cermati harga keluaran tunaknya yang sebesar 0,5 dapat diambil kesimpulan bahwa kesalahan tunak sistem tersebut adalah : masukan – keluaran =  $1 - 0,5 = 0,5$  yang sesuai dengan hitungan manual sebelumnya.

Kita dapat juga menggambar respon lingkaran tertutup dengan hanya melihat fungsi alih lingkaran terbukanya:

```

128
» g=tf([1],[1 1]) % lingkaran terbuka
Transfer function:
1
-----
s + 1

» gclose=minreal(g/(1+g)) % merubah ke lingkaran tertutup
Transfer function:
1
-----
s + 2

» step(gclose)

```

Yang akan memunculkan grafik yang serupa dengan sebelumnya. Berikut ini tabel yang memuat fungsi-fungsi lain pada MATLAB yang bermanfaat dalam menganalisa respon.

**Tabel 5.3** Fungsi MATLAB untuk Respon Sistem Kontrol

Step	respon step
impulse	respon impulse
initial	respon sistem status-ruang dengan keadaan awal ditentukan
Lsim	respon terhadap sembarang input
ltiview	analisis respon berbasis GUI
gensig	menghasilkan signal input untuk lsim
stepfun	menghasilkan input step

### G. STABILITAS

Sistem dikatakan stabil jika tanggapan sistem terhadap gaya pemaksa mengecil. Sedangkan sistem dikatakan tidak stabil jika tanggapan sistem terhadap gaya pemaksa terus membesar menuju besaran tak hingga menyebabkan sistem rusak seperti yang terjadi pada kasus meledaknya pesawat ulang alik Colombia milik Amerika Serikat. Kasus ketidakstabilan kerap terjadi pada sistem lingkaran tertutup karena adanya umpan balik yang bila rancangannya kurang tepat malah dapat menimbulkan kerusakan sistem.

Berikut ini cara perhitungan kestabilan klasik yang dinamakan Routh-Hurwitz. Suatu sistem memiliki persamaan karakteristik:  $as^4 + bs^3 + cs^2 + ds + e = 0$ . Perhatikan cara memasukan nilai pada tabel berikut ini sesuai dengan ordenya.

**Tabel 5.2** Tabel Routh-Hurwitz

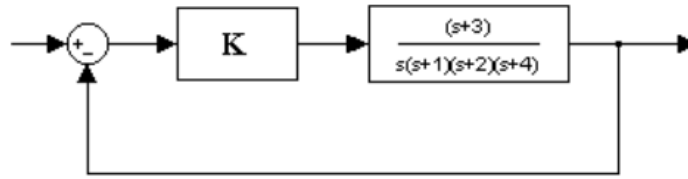
$s^4$	a	C	e
$s^3$	b	d	0
$s^2$	$\frac{(bxc)-(avd)}{b} = f$	$\frac{(bxe)-(ax0)}{b} = g$	0
$s^1$	$\frac{(fxd)-(bxg)}{f} = h$	$\frac{(fx0)-(bx0)}{f} = 0$	
$s^0$	$\frac{(hvg)-(fx0)}{h} = i$		

Lihat pada kolom pertama berisi angka a, b, f h dan i. Routh-Hurwitz mengatakan bahwa sistem stabil jika di kolom pertama ini tidak ada perubahan tanda.

Misal sistem yang kita analisa memiliki fungsi alih:

$$G(s) = \frac{s + 3}{s(s + 1)(s + 2)(s + 4)}$$





**Gambar 5.12** Contoh Perhitungan Kestabilan

Bila harga penguatan  $K = 5$ , apakah sistem stabil? Mari kita mulai menghitung dengan kriteria Routh-Hurwitz. Penyederhanaan blok sistem di atas menghasilkan fungsi alih:

$$G_{tertutup}(s) = \frac{5(s+3)}{s(s+1)(s+2)(s+4)+5(s+3)}$$

Bila yang ingin kita analisa adalah respon terhadap masukan impulse  $R(s)=1$  maka keluarannya didapat:

$$C(s) = G_{tertutup}(s) = \frac{5(s+3)}{s(s+1)(s+2)(s+4)+5(s+3)}$$

Diperoleh :

$$C(s) = \frac{5s+15}{s^4+7s^3+14s^2+13s+15}$$

Persamaan karakteristik untuk sistem lingkaran tertutup di atas adalah :

$$s^4 + 7s^3 + 14s^2 + 13s + 15 = 0 \text{ (penyebut disama dengankan dengan nol)}$$

Kemudian buat tabel Routh-Hurwitz berdasarkan orde pangkatnya yang dapat dilihat teknik peletakan nilainya pada tabel berikut ini:

**Tabel 5.4** Tabel Routh-Hurwitz

$s^4$	1	14	15
$s^3$	7	13	0
$s^2$	$\frac{(7 \times 14) - (13 \times 1)}{7} = 12,14$	$\frac{(7 \times 15) - (1 \times 0)}{7} = 15$	0
$s^1$	$\frac{(12,14 \times 13) - (15 \times 7)}{12,14} = 4,35$	$\frac{(12,14 \times 0) - (0 \times 7)}{12,14} = 0$	
$s^0$	$\frac{(4,35 \times 15) - (0 \times 12,14)}{4,35} = 15$		

Yang perlu diperhatikan adalah cara perhitungan ordo-2 ke bawah. Lihat pada kolom pertama berisi angka 1,7, 12.14, 4.35 dan 0 yang semuanya positif dan tidak ada perubahan tanda. Routh-Hurwitz mengatakan bahwa sistem stabil jika di kolom pertama ini tidak ada perubahan tanda. Bila kita ingin memperkuat sistem dengan meningkatkan K, apakah kinerja sistem akan bertambah baik, terutama kestabilannya? Gunakan cara yang sama dengan cara di atas.

Keuntungan cara Routh-Hurwitz adalah kemampuan mengetahui kestabilan sistem tanpa bantuan komputer. Sedangkan kelemahan utamanya adalah kita hanya tahu kestabilan tanpa mengetahui bentuk respon sistem tersebut. Oleh karena itu cara Routh-Hurwitz mulai ditinggalkan seiring berkembangnya komputer. Berikut ini program singkat melihat respon sistem untuk K=5 dan K yang lebih besar misalnya 10. Apakah kinerjanya baik?

```
» g5=zpk([-3],[0 -1 -2 -4],5) % fungsi alih untuk K=5 terbuka
```

```
Zero/pole/gain:
```

```
5 (s+3)
```

```
-----  
s (s+1) (s+2) (s+4)
```

```
» g10=zpk([-3],[0 -1 -2 -4],10) % fungsi alih untuk K=10 terbuka
```

```
Zero/pole/gain:
```

```
10 (s+3)
```

```
-----  
s (s+1) (s+2) (s+4)
```

```
» gc5=minreal(g5/(1+g5)) % fungsi alih untuk K=5 tertutup
```

```
Zero/pole/gain:
```

```
5 (s+3)
```

```
-----  
(s+2.473) (s+4.203) (s^2 + 0.3238s + 1.443)
```

```
» gc10=minreal(g10/(1+g10)) % fungsi alih untuk K=10 tertutup
```

```
Zero/pole/gain:
```

```
10 (s+3)
```

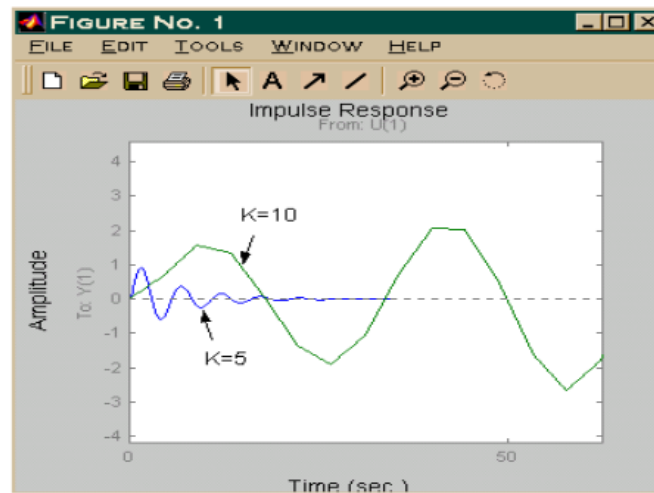
```
-----  
(s+2.63) (s+4.391) (s^2 - 0.02073s + 2.598)
```

```
» impulse(gc5) % masukan impuls pada K=5
```

```
» hold % grafik sebelumnya tetap dipakai
```

Current plot held  
 » impulse(gc10) % input impulse pada K=10

Diperoleh grafik sebagai berikut:



**Gambar 5.13** Grafik Respon Sistem dengan K=5 dan K=10

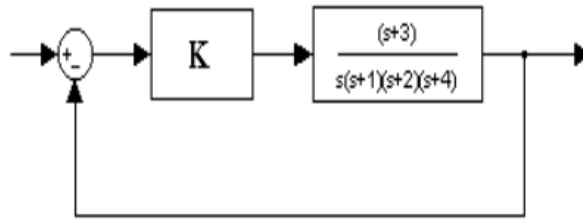
Pada grafik di atas ternyata bila K diperbesar dari 5 menjadi 10 bukannya kinerjanya baik malah sistem menjadi tidak stabil. Untuk menggambar hubungan K dengan letak-letak pole di bidang-s yang mempengaruhi kestabilan dapat Anda lihat di bab yang membahas Tempat Kedudukan Akar.

## PERTANYAAN DAN SOAL

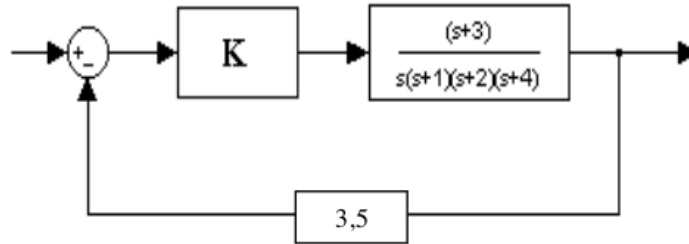
1. Gambarkan grafik respon sistem dari persamaan berikut:

$$G(s) = \frac{1}{s^3 + 2s^2 + 3s + 8}$$

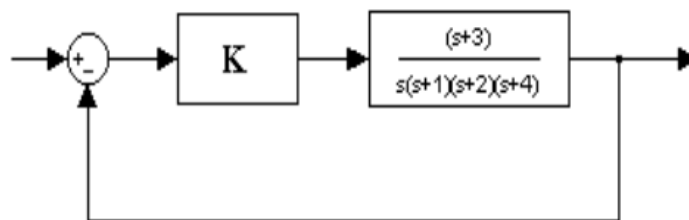
2. Tuliskan persamaan karakteristik dari blok diagram sistem kendali di bawah ini jika nilai  $K = S^2 + 4$



3. Tuliskan model persamaan matematika dari blok diagram sistem kendali di bawah ini!



4. Gambarkan grafik respon transien dari blok diagram sistem kendali di bawah ini!



5. Tentukan dengan menggambar grafikny, apakah sistem dengan persamaan di bawah ini stabil/tidak stabil?

$$G(s) = \frac{s + 4}{s(s^2 + 1)(s + 2)(s + 2)}$$

## BAB 6

### KOMPUTASI KARAKTERISTIK KENDALI DASAR

20

#### TUJUAN INSTRUKSIONAL:

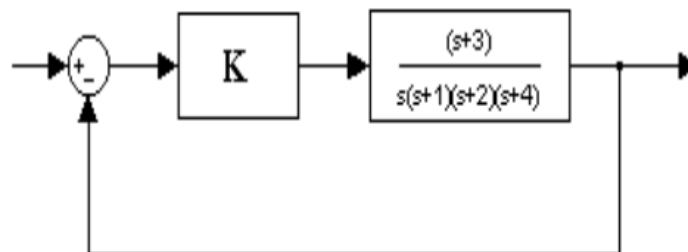
Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:

- 1) Memodelkan dan mengimplementasikan analisa tempat kedudukan akar
- 2) Memodelkan dan mengimplementasikan analisa Kontroller P, PI, PID

#### A. TEMPAT KEDUDUKAN AKAR

##### 1. Pengaruh Penguatan Terhadap Stabilitas

Lihat kembali bab VI mengenai kinerja sistem kendali terutama bab 6.3 tentang stabilitas. Berikut disajikan diagram blok sistem orde-3 seperti gambar 6.6



Gambar 6.1 Pengaruh K terhadap Stabilitas

Telah dihitung baik dengan kriteria Routh-Hurwitz maupun dengan simulasi MATLAB bahwa  $K=5$  stabil sedangkan  $K=10$  tidak stabil. Dalam perancangan kita tidak hanya berpatokan pada beberapa harga  $K$  Sebagai perancang kita ingin tahu letak akar-akar pada bidang- $s$  untuk seluruh harga penguatan  $K$ . Letak pole-pole untuk  $K=1$  didapat dengan menyelesaikan persamaan karakteristik:

$$s(s+1)(s+2)(s+4)+(s+3)=0$$

$$s^4+7s^3+14s^2+9s+3=0$$

yang merupakan persamaan s pangkat 4. Bila setelah kita hitung diperoleh akar s ada bagian real-nya yang positif maka sudah dipastikan sistem itu tidak stabil.

Dengan MATLAB:

```

» pk=[1 7 14 9 3]           % polinomial pers. karakteristik
    pk =
         1     7    14     9     3
» s=roots(pk)              % mencari akar polinomial
    s =
   -4.0415
   -2.1764
  -0.3910 + 0.4338i
  -0.3910 - 0.4338i

```

Sistem untuk K=1 stabil terbukti dari hasil akar s yang seluruh real-nya berharga negatif (-4.0415, -2.1764 dan -0.3910). Namun kita akan kerepotan bila diminta menggambar tempat kedudukan akar dengan cara seperti itu. Saat komputer belum berkembang ada teknik menggambar tempat kedudukan akar dengan metode grafis yang aturannya sebagai berikut:

- a. Jumlah Percabangan  
Jumlah cabang dalam TKA sebanding dengan jumlah pole lingkaran tertutup
- b. Simetri  
Tempat Kedudukan Akar akan simetris di sekitar sumbu real.
- c. Segmen sumbu real.  
Untuk K>0, TKA terletak di sebelah kiri jumlah ganjil sumbu real
- d. Titi awal dan titik akhir  
TKA berawal pada pole dan berakhir pada zero
- e. Aturan Tambahan:

$$\sigma = \frac{\sum Poles - \sum Zeros}{Poles - Zeros}$$

$$M = \tan \frac{(2k + 1)\pi}{Poles - Zeros}$$

dengan k=0, ± 1, ±2, ...

66

f. Syarat Sudut:

$$\angle G(s)H(s) = \pm 180^\circ (2k + 1) \quad (k = 0, 1, 2, \dots)$$

g. Syarat besaran:

$$|G(s)H(s)| = 1$$

Namun di sini kita akan berkonsentrasi penggambaran tempat kedudukan akar dengan program MATLAB.

## 2. Menggambar Tempat Kedudukan Akar dengan Matlab

Dengan MATLAB penggambaran tempat kedudukan akar menjadi lebih mudah, kita tinggal mengetik perintah sebagai berikut pada command window:

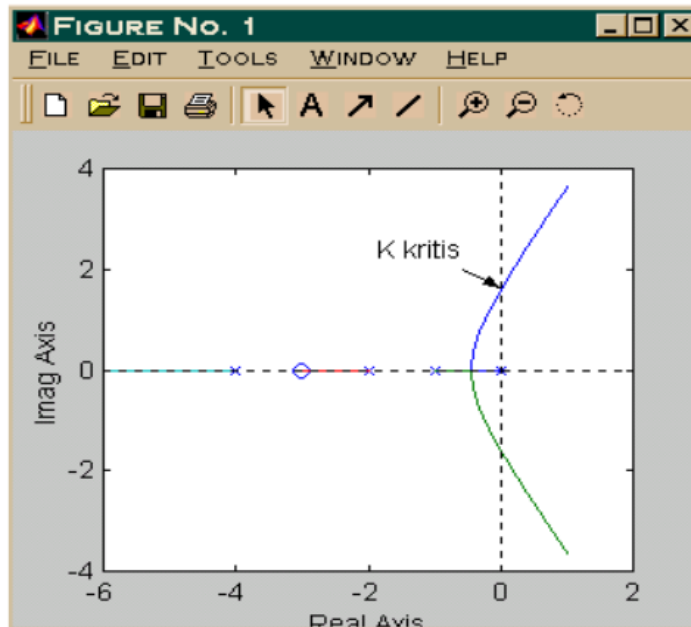
```
» g=zpk([-3],[0 -1 -2 -4],1) % fungsi alih lingkaran terbuka
```

Zero/pole/gain:

12  
(s+3)

-----  
s (s+1) (s+2) (s+4)

```
» rlocus(g)
```



Gambar 6.2 Tempat Kedudukan Akar

Karena pada grafik di atas ada garis yang menyeberangi sumbu imajiner ke kanan, maka tidak seluruh K menghasilkan sistem yang stabil. Ada harga K batas yang disebut K kritis dimana bila harga K tersebut dilampaui, sistem menjadi tidak stabil. Untuk mencari harga K kritis itu kita butuh kriteria Routh-Hurwitz.

Karena persamaan karakteristiknya:

$$s^4 + 7s^3 + 14s^2 + (8+K)s + 3K = 0$$

didapat tabel kriteria Routh Hurwitz.

**Tabel 6.1** Tabel Routh -Hurwitz

$s^4$	1	14	3K
$s^3$	7	8+K	0
$s^2$	$\frac{90-K}{7}$	$\frac{21K-0}{7} = 3K$	
$s^1$	$\frac{-K^2-65K+720}{90-K}$		
$s^0$	3K		

Syarat stabil adalah kolom pertama tidak boleh negatif. Sehingga titik kritisnya diperoleh dari:

$$\frac{-K - 65K + 720}{90 - K} = 0$$

» p=[-1 -65 720]

p =

-1 -65 720

» roots(p)

ans =

-74.6456

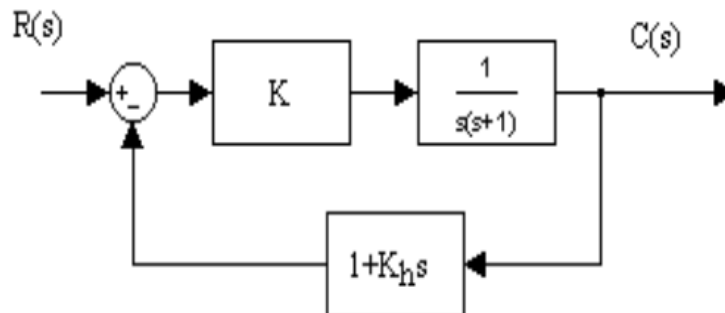
9.6456

Didapat K=9,6456. Ini berarti agar sistem tersebut stabil, kekuatan yang diijinkan maksimal 9,6456. Berarti bila  $K > 9,6456$  ada pole yang terletak di sebelah kanan sumbu imajiner pada bidang-s dan sistem akan tidak stabil.



### 3. Perancangan Sistem Kontrol dengan Tempat Kedudukan Akar

Gambar berikut ini adalah contoh sistem servo dengan umpan balik kecepatan. Kita ingin mendisain mesin ini agar didapat overshoot maksimum sebesar 0,2 dan waktu puncak 1 detik. Berapakah besar penguat K dan  $K_h$  yang cocok?



**Gambar 6.3** Diagram Blok Sistem Servo Berumpan balik Kecepatan

Overshoot maksimum  $M_p$  diberikan oleh persamaan berikut:

$$M_p = e^{-(\zeta / \sqrt{1-\zeta^2})\pi}$$

Karena nilai ini harus sama dengan 0,2 maka:

$$e^{-(\zeta / \sqrt{1-\zeta^2})\pi} = 0,2$$

$$\text{atau } \frac{\zeta\pi}{\sqrt{1-\zeta^2}} = 1,61$$

$$\text{dan } \zeta = 0,456$$

Waktu puncak diketahui 1 detik. Oleh karena itu diperoleh:

$$t_p = \frac{\pi}{\omega_d} = 1$$

$$\omega_d = 3,14$$

Karena  $\zeta$  adalah 0,456, maka  $\omega_n$  sama dengan

$$\omega_n = \frac{\omega_d}{\sqrt{1-\zeta^2}} = 3,53$$

Karena frekuensi alami  $\omega_n$  sama dengan akar K:

$$K = \omega_n^2 = 12,5$$

Maka  $K_h$  diperoleh dari persamaan sebagai berikut dengan berdasarkan soal  $B=1$  dan  $J=1$ :

$$\zeta = \frac{B + KK_h}{2\sqrt{KJ}}$$
$$K_h = \frac{2\sqrt{K}\zeta - 1}{K} = 0,178$$

Waktu naik,  $t_r$ : waktu naik (rise time) didefinisikan:

$$t_r = \frac{\pi - \beta}{\omega_d}$$

dengan

$$\beta = \tan^{-1} \frac{\omega_d}{\sigma} = \tan^{-1} 1,95 = 1,10$$

Jadi,  $t_r$  diperoleh :  $t_r = 0,65$  detik

Waktu turun,  $t_s$ : Misal kita pakai standar 2%,

$$t_s = \frac{4}{\sigma} = 2,48 \text{ det}$$

Sekarang kita coba buktikan dengan MATLAB. Tempat kedudukan akar sistem servo di atas adalah:

```
» g=zpk([], [0 -1], 1) % lingkat terbuka sistem servo
```

```
Zero/pole/gain:
```

$$\frac{1}{s(s+1)}$$

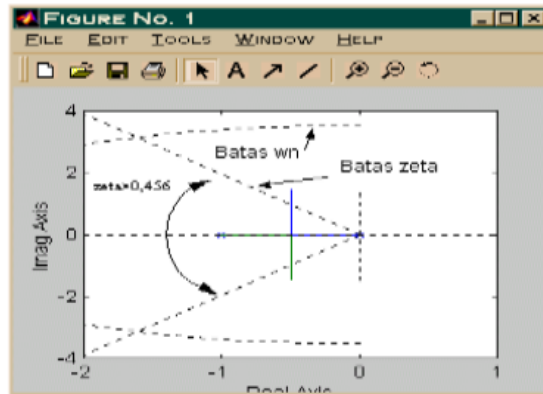
```
» rlocus(g)
```

Nilai redaman  $\zeta$  (zeta) diketahui = 0,456 dan  $\omega_n = 3,53$  maka kita dapat mengetahui batasan-batasan harga K dengan instruksi:

```
» g=zpk([], [0 -1], 1)
```

Zero/pole/gain:  $\frac{1}{s(s+1)}$

- » rlocus(g)
- » zeta=0.456;
- » wn=3.53;
- » sgrid(zeta, wn)                      % membuat batas wilayah zeta dan wn



**Gambar 6.4** Tempat Kedudukan Akar Sistem Servo

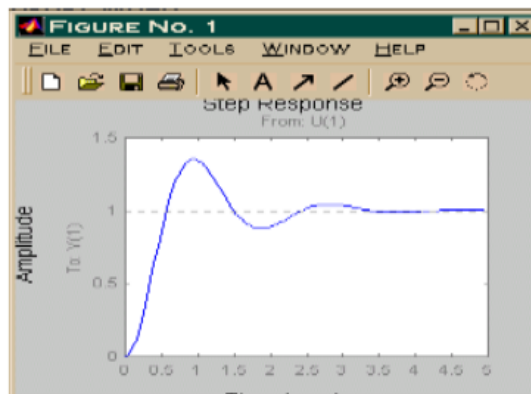
Responnya adalah:

- » g=tf([12.5],[1 2.225 12.5])                      % lingkaran tertutup mesin servo

Transfer function:  
12.5

-----  
 $s^2 + 2.225 s + 12.5$

- » step(g)



### Gambar 6.5 Grafik Step Respon Sistem Servo

Berikut ini tabel yang memuat instruksi-instruksi penting mengenai tempat kedudukan akar.

**Tabel 6.2** Alat Disain Klasik

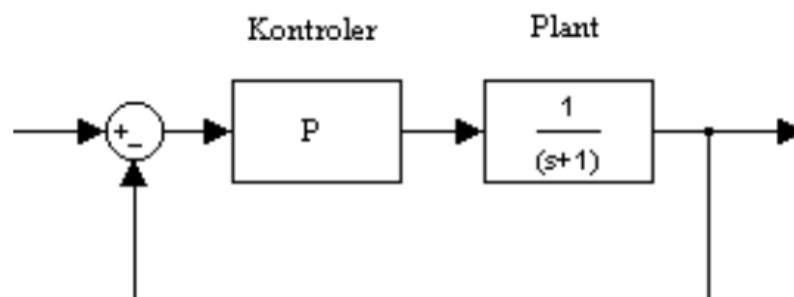
Rlocus	akar locus Evan
Rlocfind	penentuan akar perolehan locus secara interaktif
Acker	penempatan kutub SISO
Place	penempatan kutub MIMO
Estim	estimator form dengan estimator perolehan diberikan
Reg	regulator form dengan status umpan balik dan estimator perolehan diberikan

## B. KONTROLER P, PI, DAN PID

Pada bab sebelumnya Anda telah diperkenalkan dengan penguat K dan hubungannya dengan stabilitas sistem pada sistem kontrol lingkaran tertutup. Bab ini membahas lebih jauh kinerja sistem kontrol bukan hanya pada kestabilan melainkan juga pada kecepatan dan akurasi.

### 1. Kontroler Proportional (P)

Kontroler proportional (P) identik dengan penguat K pada bab yang lalu. Agar lebih jelas berikut ini diberikan contoh ilustrasi.



**Gambar 6.6** Kontroler Proportional

```
» g1=zpk([],[-1],1) % fungsi alih tanpa kontroler P (P=1)
   Zero/pole/gain:
   1
   -----
```

```

(s+1)
» g1c=minreal(g1/(1+g1))           % lingkaran tertutup g1
Zero/pole/gain:
1
-----
(s+2)

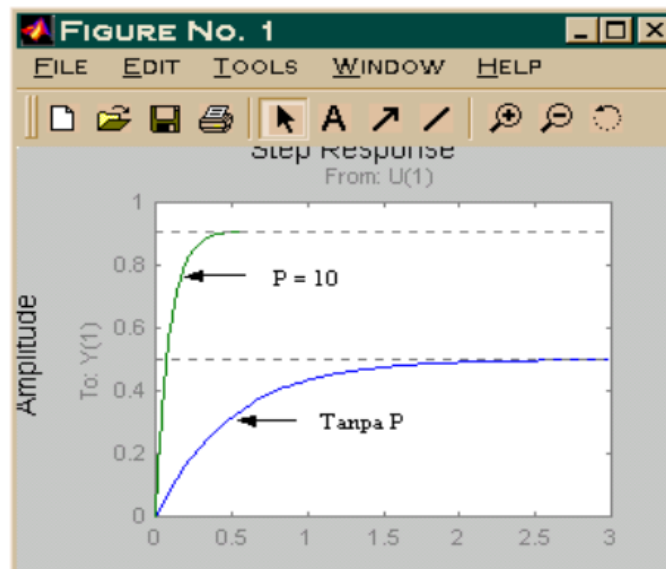
» step(g1c)                         % input step g1 tertutup
» g2=zpk([],[-1],10)                % fungsi alih dengan kontroler P=10
Zero/pole/gain:
10
-----
(s+1)

» g2c=minreal(g2/(1+g2))           % lingkaran tertutup g2
Zero/pole/gain:
10
-----
(s+11)

» hold                               % grafik yang lalu tetap dipakai
Current plot held

» step(g2c)                         % input step g2 tertutup

```



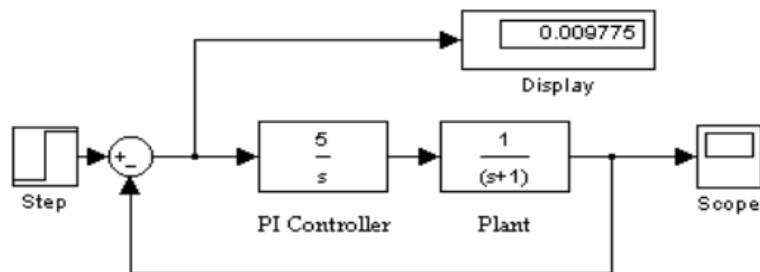
**Gambar 6.7** Perbandingan Kontroler P dengan Tanpa Kontroler

Kalau Anda perhatikan dapat disimpulkan bahwa kontroler proportional menambah akurasi (dari kesalahan tunak 0,5 menjadi sekitar 0,1) dan kecepatan/waktu turun (dari 2 detik menjadi 0,5 detik).

## 2. Kontroler Proportional Integrator (PI)

Integrator ideal adalah penambahan pole di titik asal (titik nol) pada sistem. Tetapi integrator menambah lambat suatu sistem sehingga dibutuhkan tambahan kontroler proportional.

Masih dengan sistem gambar IX.1, misal kita tambah kontroler PI,  $PI = \frac{5}{s}$ , pada sistem itu. Kita coba dengan cara lain yaitu dengan jendela model.



**Gambar 6.8** Efek Kontroler PI ( simulasi dengan SIMULINK)

Dan ternyata kesalahan tunaknya berkurang dari 0,5 menjadi mendekati nol (tanpa kesalahan). Untuk melihat kecepatan sistem cobalah double klik gambar SCOPE. Berikut ini cara melihat perbandingan kecepatan lewat command window.

```
» gpi=zpk([], [0 -1], 5) % fungsi alih dengan kontroler PI=5/s
```

Zero/pole/gain:

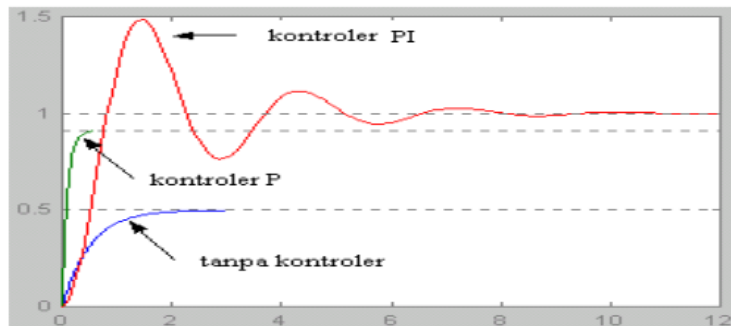
```
5
-----
s (s+1)
```

```
» gpic=minreal(gpi/(1+gpi)) % fungsi lingkaran tertutup
```

Zero/pole/gain:

```
5
```

-----  
( $s^2 + s + 5$ )  
» step(gpic)



**Gambar 6.9** Perbandingan Kontroler PI, P dan Tanpa Kontroler

Perhatikan, ternyata PI memiliki kelemahan yaitu berosilasi dan waktu turunnya jadi lama sekitar enam detik. Ingat bahwa integrator murni memakai prinsip menambah pole di titik asal (titik nol) yang pada hakikatnya adalah titik perbatasan antara stabil dan tidak stabil yang sangat berbahaya untuk perancangan. Bergeser saja sedikit ke kanan letak pole integrator, sistem itu sudah tidak stabil.

Ada teknik lain untuk mengatasi hal itu yaitu dengan kontroler lag. Prinsipnya adalah menambah pole di sekitar titik asal (tentu saja di sebelah kirinya) dan untuk mengurangi efek pole tambahan itu, ditambah zero yang letaknya dekat pole tambahan itu. Guna mengurangi efek lag (ketinggalan) dan memperbaiki margin fasa (lihat bab X) ditambahkan kompensator lead, sehingga dikenal kontroler Lag – lead.

### 3. Kontroler Proportional Diferensiator (Derivatif)

Prinsip dari kontroler ini adalah menambah zero pada sistem kontrol. Penambahan zero ini meningkatkan kecepatan sistem. Kelemahannya adalah pada kontroler ini muncul gangguan noise yang harus diperhitungkan bila sistem bekerja dengan input yang memiliki frekuensi besar. Disamping itu kontroler ini memerlukan catu daya yang berarti menambah biaya.

### 4. Kontroler PID

Ini merupakan jenis kontroler termahal gabungan antara proportional, integrator dan diferensiator (disingkat PID) yang rumus umumnya:

$$K_p = \left( 1 + \frac{1}{T_i} + T_d s \right) \quad (6.1)$$

dimana  $K_p/T_i=K_i$  dan  $K_p T_d=K_d$ .

Di sini kita menggunakan penulisan:

$$K_p + K_d s + \frac{K_i}{s} \quad (6.2)$$

Atau bila disederhanakan:

$$\frac{K_d s^2 + K_p s + K_i}{s} \quad (6.3)$$

Kita cari harga-harga  $K_d$ ,  $K_p$  dan  $K_i$  yang sesuai agar didapat hasil yang terbaik. Dengan bantuan komputer khususnya program MATLAB prinsip *trial and error* jadi lebih cepat. Bayangkan kalau kita menghitung tiap harga  $K$  tertentu tentu saja akan banyak menghabiskan waktu.

*Trial and Error* yang paling nyaman adalah dengan jendela model SIMULINK karena kita tinggal merubah besar  $K$  tanpa menulis ulang instruksi, gambarlah pada jendela model sistem tersebut.

## 5. Aturan Ziegler-Nichols

Ziegler dan Nichols mengusulkan aturan-aturan untuk menentukan nilai penguatan proporsional  $K_p$ , waktu integral  $T_i$  dan waktu turunan  $T_d$  yang didasarkan pada karakteristik respon transien suatu sistem yang diketahui. Dasarnya adalah persen overshoot maksimum 25% dan respon terhadap masukan step. Ada dua metode yang mereka lontarkan yang sebagian besar penggunaannya dengan cara eksperimental (trial dan error).

Kita ambil contoh lagi. Misalnya diketahui sistem lingkaran tertutup yang fungsi alih lingkaran terbukanya,

$$G(s) = \frac{1}{s^2 + 2s + 5}$$

Akan kita rancang dengan spesifikasi:

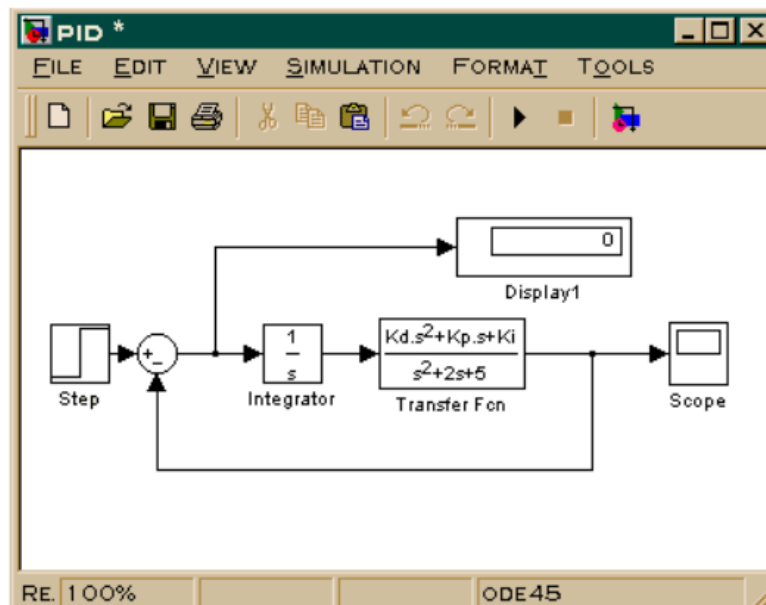
- Sistem Stabil
- $M_p < 0,3$  (30% overshoot)
- Waktu turun  $< 3$  detik, Kesalahan tunak  $< 0,1$



Langkah yang ditempuh adalah sebagai berikut:

- Simulasikan sistem tanpa kompensator, apakah sudah sesuai dengan spesifikasi.
- Rancang kontroler P ( $K_p$ ), simulasikan!
- Rancang kontroler PD ( $K_d$ ), simulasikan! Rancang kontrol PI ( $K_i$ ), simulasikan!
- Bila kurang memuaskan, kembali ke langkah 2!

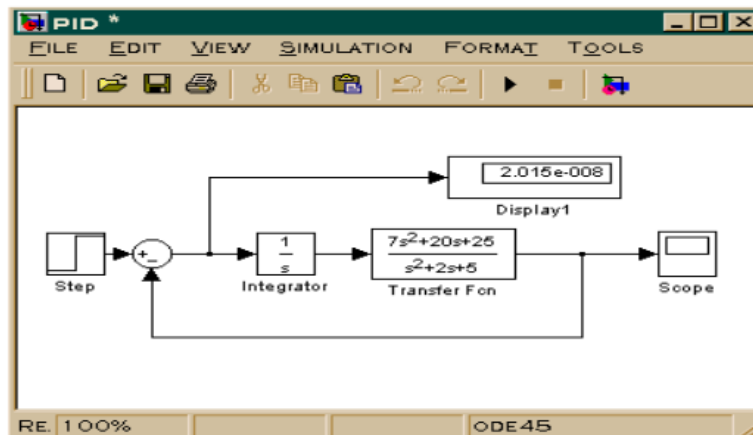
Untuk simulasi yang cepat dapat digunakan jendela model SIMULINK yang tersedia di MATLAB. Tanpa kontroler setelah disimulasi diperoleh kesalahan tunak = 0,8333 dan waktu turun sekitar 4 detik yang berarti masih jauh dari spesifikasi yang diharapkan. Kita tidak dapat membentuk fungsi alih seperti pada persamaan umum PID, persamaan (8.3), karena fungsi alih tidak mengizinkan orde pembilang lebih besar dari penyebut. Oleh karena itu diperlukan sedikit modifikasi seperti gambar berikut.



**Gambar 6.10** Perancangan Kontroler PID dengan SIMULINK dengan Modifikasi Fungsi Alih

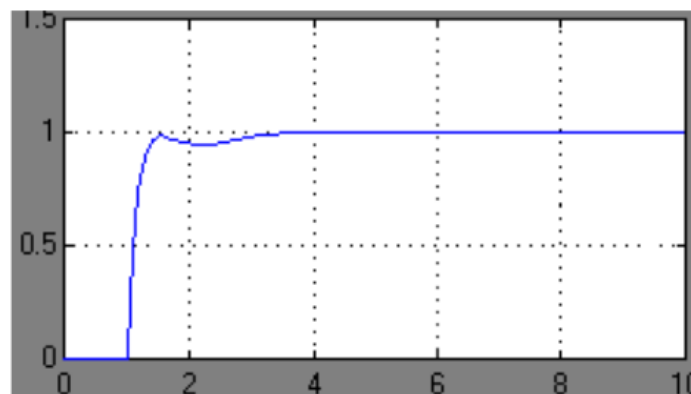
Fungsi alih sistem di atas adalah identik dengan soal hanya saja pembilang PID di pindah ke plant untuk menghindari PID yang orde pembilangnya lebih besar dari penyebut. Double klik pada blok fungsi alih (*transfer function*) dan ganti  $K_d$ ,  $K_p$  dan  $K_i$  dengan angka lalu jalankan (RUN). Bila hasilnya belum sesuai dengan spesifikasi yang diharapkan, ganti lagi harga  $K_p$ ,  $K_d$  dan  $K_i$ . Berdasarkan tata cara

trial and error, sebaiknya cari dahulu Kd (derivatif) yang sesuai baru kemudian Kp dan Ki. Berikut ini hasil trial and error penulis, dan pembaca mungkin saja menemukan hasil yang lebih baik.



**Gambar 8.11** Hasil Perancangan kontroler PID

Penulis menemukan kontroler yang sesuai dengan spesifikasi dengan fungsi alih  $PID(s) = 20 + 7s + \frac{25}{s}$ . Dari gambar dilihat kesalahan tunaknya memenuhi spesifikasi (mendekati nol). Untuk mengetahui waktu turun dan overshoot apakah sudah sesuai, tinggal mendouble klik pada blok scope.



**Gambar 6.12** Respon Sistem dengan PID.

Kita lihat pada grafik di atas, overshoot hampir tidak ada dan yang lebih penting kecepatan sistem kontrol (waktu naik kira-kira 1,5 detik dan waktu turun kira-kira 2,2 detik) yang sesuai dengan spesifikasi. Pembaca dapat memahami sendiri saat perancangan kontroler di atas perubahan apa yang terjadi bila harga-harga K di atas dirubah. Mungkin secara tidak sengaja pembaca menemukan kondisi dimana harga-harga K menyebabkan sistem tidak stabil.

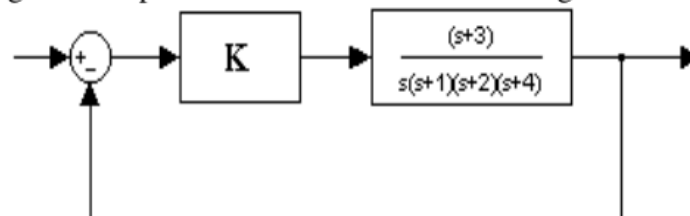
Anda juga dapat langsung mendesain blok PID dengan cara klik tanda ‘+’ pada **SIMULINK EXTRAS** dan **ADDITIONAL LINEAR** yang kemudian tampak blok PID CONTROLLER. Tabel berikut merinci efek yang diberikan oleh masing-masing kontroler.

**Tabel 6.3** Efek Kontroler Terhadap Kinerja Sistem

Respon L. tertutup	Waktu Naik	Overshoot	Waktu Turun	Kesalahan Tunak
<b>K<sub>p</sub></b>	Turun	Naik	Sedikit Berubah	Turun
<b>K<sub>i</sub></b>	Turun	Naik	Naik	Hilang
<b>K<sub>d</sub></b>	Sedikit Berubah	Turun	Turun	Sedikit Berubah

## PERTANYAAN DAN SOAL

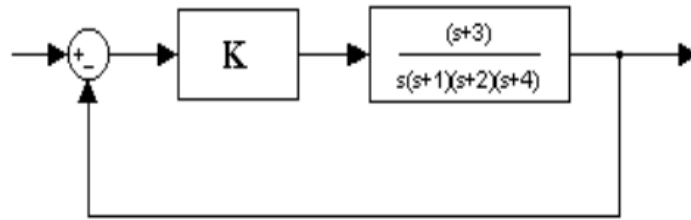
1. Gambarkan grafik tempat kedudukan akar dari blok diagram berikut!



2. Gambarkan grafik tempat kedudukan akar dari persamaan berikut!

$$G(s) = \frac{s + 4}{s(s^2 + 1)(s + 2)(s + 2)}$$

3. Gambarkan grafik sistem Proporsional (P), Integral (I), Derivatif (D), PI, dan PID berdasarkan blok diagram berikut:



## KOMPUTASI MODEL RUANG KEADAAN

20

### TUJUAN INSTRUKSIONAL:

Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:

- 1) Mengerti konsep analisa Ruang Keadaan
- 2) Menjelaskan kembali dan mengerti model ketidakunikan Ruang
- 3) Memodelkan perubahan Fungsi Alih menjadi Ruang Keadaan
- 4) Memodelkan perubahan Ruang Keadaan menjadi Fungsi Alih
- 5) Menggambarkan Diagram Blok Ruang Keadaan menggunakan Simulink

Sejauh ini kita telah menganalisa sistem kontrol dengan teknik konvensional atau sering disebut metode klasik yang cirinya adalah satu masukan dan satu keluaran (*SISO*). Ternyata metode klasik tidak memadai untuk sistem-sistem kontrol yang banyak dirancang saat ini. Misalnya sistem pemanas air, ternyata tidak hanya dipengaruhi oleh laju pemanasan saja, melainkan juga oleh laju aliran air. Pada AC, ternyata pendinginan tidak hanya dipengaruhi sistem pendinginnya melainkan juga oleh kecepatan kipas dan gerak sirip.

186

Pada bab ini kita akan membahas sistem dengan masukan dan keluaran yang lebih dari satu (*MIMO*) dengan metode yang disebut metode ruang keadaan.

### A. MEMBENTUK PERSAMAAN RUANG KEADAAN

Ada baiknya Anda buka kembali buku tentang matriks dan vektor, karena persamaan keadaan adalah persamaan dalam bentuk matriks. Jika  $x$  adalah variabel keadaan suatu sistem, maka persamaan keadaannya didefinisikan sebagai berikut:

35

$$\dot{x}_1 = X_1(x_1, x_2, \dots, x_n)$$

$$\dot{x}_2 = X_2(x_1, x_2, \dots, x_n)$$

⋮

$$\dot{x}_n = X_n(x_1, x_2, \dots, x_n)$$

Dengan syarat bahwa untuk setiap himpunan nilai  $\dot{x}_1, \dot{x}_2, \dots, \dot{x}_n$  terdapat suatu himpunan nilai  $x_1, x_2, x_3, \dots, x_n$  yang unik dan sebaliknya. Jadi jika  $x$  merupakan vektor keadaan maka :  $\dot{x} = Px$  juga merupakan vektor keadaan, dengan syarat bahwa matriks  $P$  nonsingular. Vektor-vektor keadaan yang berbeda membawa informasi yang sama mengenai perilaku sistem.

Biasanya kita lebih mudah memahami sesuatu dari contoh. Misal sistem dengan persamaan diferensial :

$$\ddot{y} = 6\ddot{y} + 11\dot{y} + 6y = 6u \quad (7.1)$$

dengan  $y$  adalah keluaran dan  $u$  adalah masukan sistem. Kita pilih variabel baru sebagai berikut:

$$\begin{aligned} x_1 &= y \\ x_2 &= \dot{y} \\ x_3 &= \ddot{y} \end{aligned}$$

Kemudian kita dapatkan:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= -6x_1 - 11x_2 - 6x_3 + 6u \end{aligned}$$

Lihat persamaan terakhir di atas adalah hasil penyelesaian  $\ddot{y}$  dari persamaan diferensial pada soal. Kemudian kita buat matriks ruang keadaannya dengan 3 persamaan diferensial.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 6 \end{bmatrix} u \quad (7.2)$$

Persamaan keluaran diberikan oleh:

$$y = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (7.3)$$

Persamaan (7.2) dan (7.3) dapat dituliskan dalam bentuk standar sebagai berikut:

$$\begin{aligned} \dot{y} &= Ax + Bu \\ y &= Cx \end{aligned} \quad (7.4)$$

dengan

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 6 \end{bmatrix}, C = [1 \ 0 \ 0]$$

## B. KETIDAKUNIKAN PERSAMAAN KEADAAN

Persamaan keadaan (7.4) ternyata bukan satu satunya persamaan yang merepresentasikan sistem tersebut. Kita akan coba membentuk persamaan yang lain dari sistem itu. Bentuk fungsi alihnya dalam laplace.

$$\frac{Y(s)}{U(s)} = \frac{6}{s^3 + 6s^2 + 11s + 6} = \frac{6}{(s+1)(s+2)(s+3)}$$

Coba Anda cari pecahan parsialnya dengan MATLAB, Anda akan memperoleh:

$$\frac{Y(s)}{U(s)} = \frac{3}{s+1} + \frac{-6}{s+2} + \frac{3}{s+3}$$

Sehingga didapat

$$Y(s) = \frac{3}{s+1}U(s) + \frac{-6}{s+2}U(s) + \frac{3}{s+3}U(s) \quad (7.5)$$

Marilah kita definisikan

$$X_1(s) = \frac{3}{s+1}U(s) \quad (7.6)$$

$$X_2(s) = \frac{-6}{s+2}U(s) \quad (7.7)$$

$$X_3(s) = \frac{3}{s+3}U(s) \quad (7.8)$$

Dengan membalik transformasi laplace persamaan (7.6), (7.7) dan (7.8) kita peroleh:

$$\begin{aligned} \dot{x}_1 &= -x_1 + 3u \\ \dot{x}_2 &= -2x_2 + 6u \\ \dot{x}_3 &= -3x_3 + 3u \end{aligned}$$

Dalam bentuk notasi matriks vektor kita dapatkan:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 3 \\ -6 \\ 3 \end{bmatrix} u \quad (7.9)$$

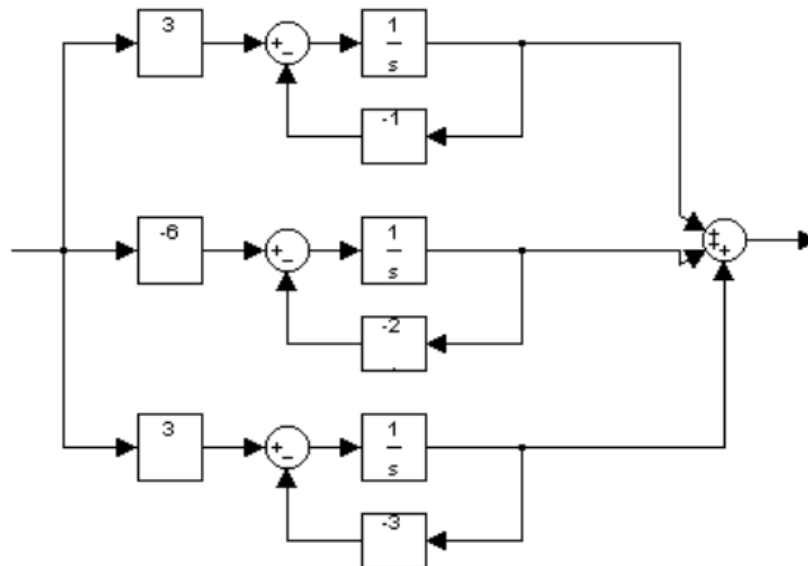
Berdasarkan persamaan (7.5), diperoleh persamaan keluarannya:

$$y = x_1 + x_2 + x_3$$

atau:

$$y = [1 \quad 1 \quad 1] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (7.9)$$

Gambar diagram bloknya adalah sebagai berikut:



**Gambar 7.1** Diagram Blok Sistem dengan Persamaan Keadaan (7.9)

Mengapa persamaan (7.9) dan persamaan (7.2) berbeda padahal keduanya merupakan representasi dari sistem yang sama? Hal ini terjadi karena cara membentuk persamaan ruang keadaan kedua persamaan itu berbeda. Sehingga tiap persamaan ruang keadaan bukan merupakan persamaan yang unik. Kasus tersebut terjadi karena invariansi nilai eigen.



Tetapi walaupun tiap persamaan keadaan tidak unik, baik persamaan (7.2) maupun (7.9) memiliki nilai eigen yang sama yaitu -1, -2 dan -3. Buka kembali command window MATLAB anda.

```
» A1=[0 1 0; 0 0 1; -6 -11 -6]      % Matriks A persamaan (10.2)
```

```
A1 =
    0    1    0
    0    0    1
   -6   -11   -6
```

```
» eig(A1)
```

```
ans =
  -1.0000
  -2.0000
  -3.0000
```

```
» A2=[-1 0 0; 0 -2 0; 0 0 -3]      % Matriks A persamaan (10.9)
```

```
A2 =
   -1    0    0
    0   -2    0
    0    0   -3
```

```
» eig(A2)
```

```
ans =
   -1
   -2
   -3
```

Lihat, terbukti bahwa nilai eigen kedua persamaan sama. Bagaimana jika nilai eigen kedua persamaan itu berbeda? Bila hal ini terjadi berarti kedua persamaan yang nilai eigennya berbeda itu bukan representasi sistem yang sama.

### C. MERUBAH FUNGSI ALIH MENJADI RUANG KEADAAN

Masih sistem dengan persamaan (11.1) dengan fungsi alih:

$$\frac{Y(s)}{U(s)} = \frac{12 \cdot 6}{s^3 + 6s^2 + 11s + 6} = \frac{6}{(s+1)(s+2)(s+3)}$$

Berikut instruksi pada command window guna merubah fungsi alih menjadi persamaan ruang keadaan.

```
» g=tf([6],[1 6 11 6])
```

69

Transfer function:

6

$$s^3 + 6s^2 + 11s +$$

» ss(g)

a =

	x1	x2	x3
x1	-6	-2.75	-0.75
x2	4	0	0
x3	0	2	0

b =

	u1
x1	1
x2	0
x3	0

c =

	x1	x2	x3
y1	0	0	0.75

d =

	u1
y1	0

Anda lihat, ternyata matriks A berbeda sehingga persamaan ruang keadaannya berbeda baik dengan persamaan (7.2) maupun (7.3). Tetapi sesungguhnya persamaan itu sama karena nilai eigen matriks a di atas:

» A3=[-6 -2.75 -0.75; 4 0 0; 0 2 0]

A3 =

-6.0000	-2.7500	-0.7500
4.0000	0	0
0	2.0000	0

» eig(A3)

ans =

-3.0000  
-2.0000  
-1.0000

Ternyata nilai eigen persamaan ruang keadaan yang baru kita bentuk lewat MATLAB sama dengan nilai eigen kedua persamaan keadaan sebelumnya. Nilai eigen sering disebut akar karakteristik sistem.

Bentuk standar MATLAB dalam merubah fungsi alih menjadi ruang keadaan adalah sebagai berikut:

```

>> m=[6]; % Pembilang Fungsi Alih
>> den=[1 6 11 6]; % Penyebut Fungsi Alih
>> [A,B,C,D]=tf2ss(num,den) % Merubah Fungsi Alih menjadi Ruang Keadaan

```

```

A =
    -6   -11    -6
     1     0     0
     0     1     0
B =
     1
     0
     0
C =
     0     0     6
D =
     0

```

#### D. MERUBAH PERSAMAAN RUANG KEADAAN MENJADI FUNGSI ALIH

Tentu saja kita dapat merubah persamaan keadaan menjadi fungsi alih dalam variabel s.

Format penulisan:

```
[n,d]=ss2tf(A,B,C,D,iu)
```

dengan iu merupakan masukan yang lebih dari satu.

Masih dengan contoh sebelumnya, coba Anda rubah kembali ke fungsi alihnya dengan instruksi sebagai berikut:

```

>> [num,den]=ss2tf(A,B,C,D)
num =
     0  0.0000  0.0000  6.0000
den =
  1.0000  6.0000  11.0000  6.0000

```

Ternyata hasilnya tepat sesuai dengan fungsi alih semula. Kita ambil contoh yang lebih kompleks.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -25 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Mengikuti format penulisannya, masukkan instruksi pada command window sebagai berikut:

```
35
» A=[0 1; -25 -4];
» B=[1 1; 0 1];
» C=[1 0; 0 1];
» D=[0 0; 0 0];
» [n,d]=ss2tf(A,B,C,D,1)
```

```
n =
    0    1    4
    0    0 -25
d =
    1    4    25
```

```
» [n,d]=ss2tf(A,B,C,D,2)
```

```
n =
    0  1.0000  5.0000
    0  1.0000 -25.0000
d =
    1    4    25
```

Apa maksud hasil keluaran pada MATLAB di atas?. Simbol n berarti pembilang dan d berarti penyebut (singkatan dari numerator dan denominator). Berikut adalah pembacaan dari hasil keluaran MATLAB tersebut.

Terhadap masukan pertama  $u_1$ :

$$\frac{Y_1(s)}{U_1(s)} = \frac{s+4}{s^2+4s+25} \qquad \frac{Y_2(s)}{U_1(s)} = \frac{-25}{s^2+4s+25}$$

Terhadap masukan kedua  $u_2$ :

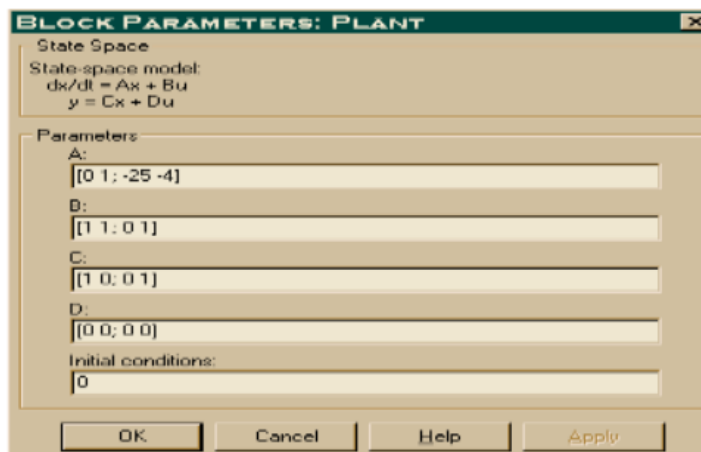
$$\frac{Y_1(s)}{U_2(s)} = \frac{s+5}{s^2+4s+25} \qquad \frac{Y_2(s)}{U_2(s)} = \frac{s-25}{s^2+4s+25}$$

## E. MENGGAMBAR DIAGRAM BLOK PERSAMAAN RUANG KEADAAN DENGAN SIMULINK

Masih ingat cara membuka jendela model? Pada bab terdahulu Anda telah diperkenalkan cara menggambar diagram blok dengan fasilitas yang ada pada MATLAB yaitu SIMULINK. Versi terakhir SIMULINK saat buku ini ditulis adalah versi 6. Seperti biasa, klik **File – New – Model** akan membawa Anda ke jendela model. Misal kita ingin melihat respon sistem dengan persamaan ruang keadaan sistem contoh bab 11.4.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -25 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$
$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Pertama-tama kita ingin menggambar blok diagram persamaan ruang keadaan di atas. Pada Simulink Library Browser, klik tanda ‘+’ **SIMULINK** dan **CONTINUOUS**, yang akan memunculkan menu-menu yang termasuk di dalamnya **STATE-SPACE**. Klik dan drag menu state-space ke jendela model. Double klik pada state-space saat Anda ingin memasukkan persamaan ruang keadaan. Masukkan harganya seperti di bawah ini. Perhatikan spasi antar elemen matriks parameter.

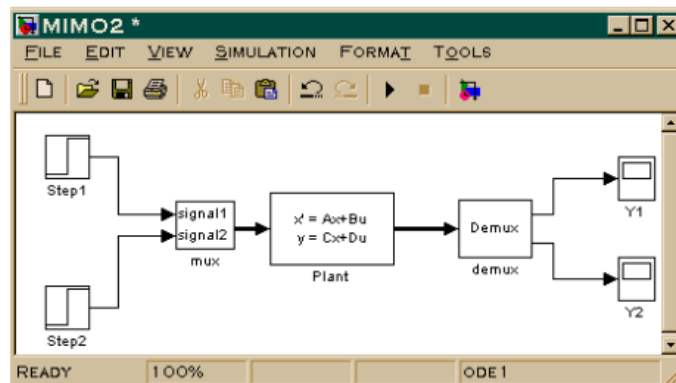


**Gambar 7.2** Menu Input Persamaan Ruang Keadaan Pada SIMULINK

Sistem dengan dua masukan dan dua keluaran mengharuskan kita menerapkan blok khusus Multiplexing dan Demultiplexing. Klik tanda ‘+’ pada **SIMULINK** dan **SIGNALS AND SYSTEMS**. Tampak di sana blok **MUX** dan

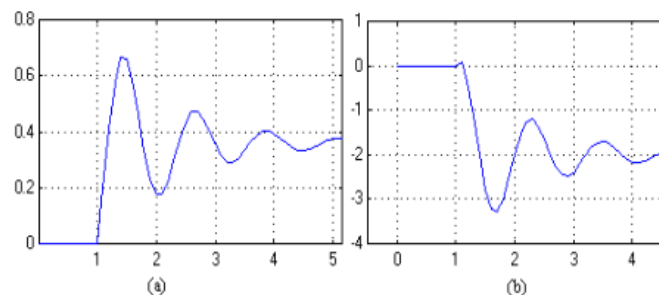
**DEMUX.** Klik dan drag ke jendela model, dimana MUX di bagian input Plant sedangkan DEMUX di bagian output. Double klik pada blok MUX dan DEMUX rubah sesuai dengan sistem di atas dimana jumlah input dan output kita pilih 2. Mungkin saat Anda mengklik dan drag MUX tampak bentuk yang berbeda dengan bentuk di gambar XI.3. Oleh karena itu Anda harus memperbesar ukuran dengan cara mengklik blok tersebut lalu menggerakkan ujungnya agar membesar.

Sistem kita memerlukan dua buah masukan step, dan kita ingin mengetahui responnya pada dua terminal keluaran. Buka Simulink Library Browser lagi, Klik tanda '+' pada **SIMULINK** dan **SOURCES** lalu pilih **STEP**. Setelah klik dan drag, rubah setting dengan double klik, dan isilah agar lebih akurat **sample time** pada menu properties harga yang kecil, misalnya 0,1. Buat setting untuk kedua masukan. Untuk melihat keluaran buat blok scope dengan klik tanda '+' pada **SIMULINK** dan **SINKS** lalu klik dan drag **SCOPE** ke jendela model Anda. Buat seperti pada gambar berikut.



**Gambar 7.3** Penggambaran dengan SIMULINK

Lalu jalankan dengan mengklik icon ► dan bila ingin berhenti klik icon ■. Lalu untuk melihat hasilnya double klik blok SCOPE Y1 dan SCOPE Y2. Untuk memperjelas dan memperbesar grafik klik pada bagian yang ingin Anda lihat. Hasilnya akan tampak seperti gambar di bawah ini (Warna putih pada background dibuat dengan invert color).



### Gambar 7.4 Keluaran pada SCOPE

Gambar (a) adalah keluaran pada SCOPE Y1 sedangkan gambar (b) keluaran pada SCOPE Y2. Cobalah Anda ganti masukan dengan input yang lain, misalnya: sinus, random, ramp dan lain-lain. Berlatihlah dengan mencoba menggambar blok sistem multi masukan dan multi keluaran yang lain.

Tabel berikut memuat fungsi lain yang berguna dalam analisa sistem kontrol dalam ruang keadaan.

**Tabel 7.1** Model Status-Ruang

ras,drss	model status-ruang stabil acak
ss2ss	transformasi koordinat status
Canon	bentuk kanonik status-ruang
ctrb,obsv	matriks pengontrolan dan pengamatan
Gram	gramian pengontrolan dan pengamatan
Ssbal	realisasi penyeimbangan diagonal status-ruang
Balreal	penyeimbangan input-output berbasis gramian
Modred	reduksi model status
Mineral	realisasi minimal dan pembatalan kutub/nol
Augstate	penambahan output dengan menambahkan status

### PERTANYAAN DAN SOAL

---

Tulis dan gambarkan kondisi dan kinerja sistem kontrol dalam ruang keadaan dari persamaan berikut:

$$\frac{R(s)}{C(s)} = \frac{12s + 2,5}{s^3 + 2s^2 + 8s + 6} = \frac{5}{(s+1)(s+2)(s+4)}$$

## BAB 8

# KOMPUTASI ELEKTRONIK

---

20

### TUJUAN INSTRUKSIONAL:

Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:

- 1) Mengerti konsep dasar komputasi elektronik
  - 2) Membuat script MATLAB untuk memproses model komputasinya
  - 3) Mengimplementasikan rancangan proses komputasi elektronika dasar ke dalam pemrograman berbasis MATLAB
  - 4) Membuat script MATLAB untuk menampilkan grafik atas hasil komputasi elektronik
- 

14

Pada bab ini akan dibahas dan dijelaskan beberapa contoh proses komputasi untuk menyelesaikan permasalahan rangkaian listrik dan elektronik. Proses komputasi ini akan sangat membantu para mahasiswa dalam memberikan dan mengimplementasikan ilmu dasar pemrograman yang telah diikuti sebelumnya. Buku ini memberikan pemahaman lebih jauh bagaimana mengimplementasikan model-model analisis matematik rangkaian listrik dan elektronik berbantu pemrograman MATLAB.

Untuk menyingkat dan memperjelas pemahaman, buku ini secara langsung akan memberikan contoh-contoh penyelesaian masalah atau soal yang diselesaikan dengan menuliskan sederatan script pada program MATLAB. Di bawah ini diberikan contoh soal dan cara penyelesaiannya dengan menuliskan script pada command window MATLAB ataupun pada editornya.

### Contoh 1.

Sebuah tegangan yang melintasi suatu resistor berdasarkan hukum Ohm:  $v=i.R$ . Daya yang dibangkitkan sepanjang rangkaian itu adalah sebesar:  $P=i^2.R$ . Jika diasumsikan nilai resistansi 10 ohm dan arus bertambah secara linier setiap 2 ampere mulai dari 0 sampai dengan 10 ampere, maka besarnya perubahan tegangan dan disipasi daya akibat kenaikan arus adalah:



Tuliskan script program untuk memecahkan masalah di atas pada editor MATLAB seperti script berikut.

**Solusi:**

```
16
R=10; % Resistance value
i=(0:2:10); % Generate current values
v=i.*R; % array multiplication to obtain voltage
p=(i.^2)*R; % power calculation
sol=[i v p]
```

Maka hasil yang ditampilkan pada command window MATLAB adalah 3 barisan angka mulai dari nilai arus, tegangan, dan daya (pada baris ke-3).

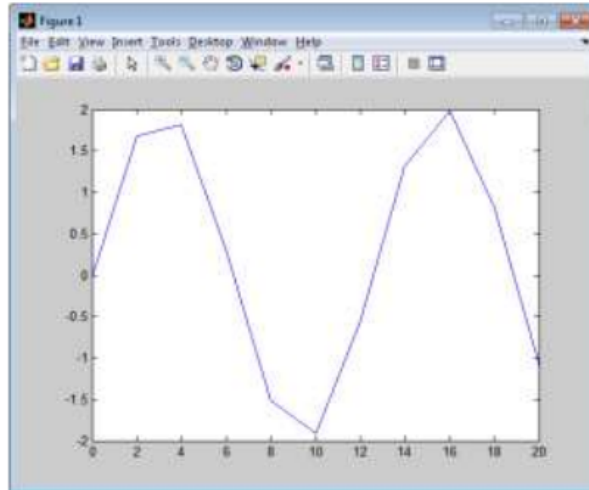
```
sol =
Columns 1 through 6
0     2     4     6     8    10
Columns 7 through 12
0    20    40    60    80   100
Columns 13 through 18
0    40   160   360   640  1000
```

**Contoh 2.**

16  
Jika  $x(t) = 2t$  dan  $y(t) = \sin(t)$ , maka untuk  $t = 0$  sampai dengan  $10 \text{ ms}$ . Tentukan nilai  $x(t)$  dan  $y(t)$  dan gambarkan grafik fungsinya!

**Solusi:**

```
16
for i= 0:10
x(i+1) = 2*i;
y(i+1) = 2*sin(i);
end
plot(x,y)
```

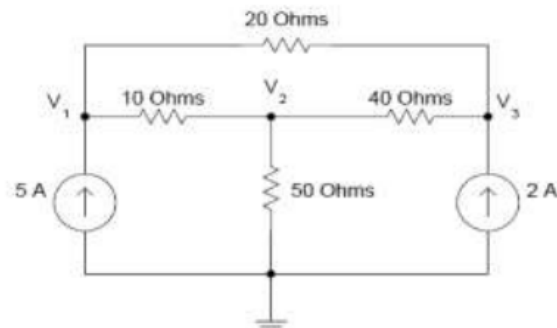


**Gambar 8.1** Grafik Fungsi X terhadap Y untuk contoh 2

## A. ANALISIS RANGKAIAN DC

### 1. Analisis Tegangan Simpul (Nodal Analysis)

Untuk menghitung dan menentukan besarnya tegangan simpul pada sebuah rangkaian listrik tertutup seperti gambar 8.2, dapat dengan mudah dianalisis berbantu MATLAB menggunakan rumus yang sudah dipelajari pada teori matakuliah sebelumnya.



**Gambar 8.2** Rangkaian Tegangan Simpul

#### Contoh 1.

Penyelesaian menggunakan script MATLAB pada rangkaian tegangan simpul untuk gambar 8.2 terlebih dahulu harus ditentukan rumus-rumus yang ada pada masing-masing loop tersebut. Berdasarkan analisis tegangan yang terjadi pada

masing-masing loop, model matriks jaringan tetangan tersebut dituliskan seperti berikut ini.

$$\begin{bmatrix} 0.15 & -0.1 & -0.05 \\ -0.1 & 0.145 & -0.025 \\ -0.05 & -0.025 & 0.075 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \\ 2 \end{bmatrix}$$

Pada editor MATLAB, kita tuliskan script sebagai berikut.

```

13
Y = [ 0.15 -0.1 -0.05;
      -0.1 0.145 -0.025;
      -0.05 -0.025 0.075];
I = [5;
      0;
      2];
fprintf('Nodal voltages V1, V2 and V3 are \n')
v = inv(Y)*I
diary

```

Nilai tegangan V1, V2, dan V3 yang ditunjukkan pada command windows MATLAB adalah:

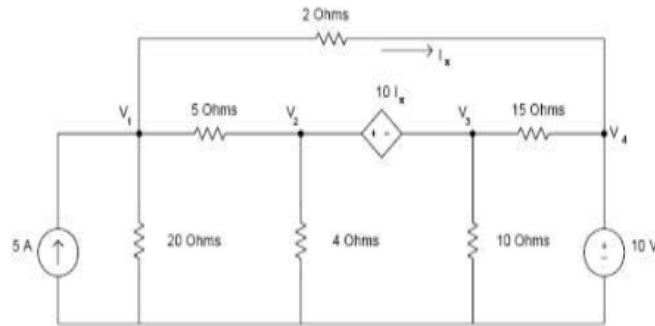
```

V =
404.2857
350.0000
412.8571

```

**Contoh 2** lainnya untuk menghitung dan menentukan besarnya tegangan simpul pada rangkaian tertutup seperti gambar 8.3 adalah:

$$\begin{bmatrix} 0.75 & -0.2 & 0 & -0.5 \\ -5 & 1 & -1 & 5 \\ -0.2 & 0.45 & 0.1667 & -0.06667 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \\ 0 \\ 10 \end{bmatrix}$$



**Gambar 8.3** Rangkaian Tegangan Simpul untuk Contoh 2

Script MATLAB yang dituliskan pada editor:

```

Y = [0.75 -0.2 0 -0.5;
-5 1 -1 5;
-0.2 0.45 0.166666667 -0.066666667;
0 0 1];
I = [5 0 0 10]';
fprintf('Nodal voltages V1,V2,V3,V4 are \n')
V = inv(Y)*I
diary

```

Nilai tegangan  $V_1$ ,  $V_2$ ,  $V_3$ , dan  $V_4$  yang ditunjukkan pada command windows MATLAB adalah:

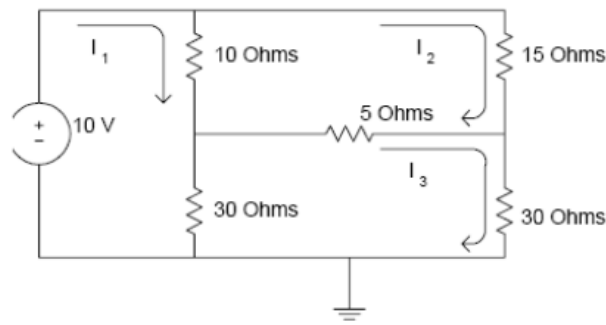
```

V =
18.1107
17.9153
-22.6384
10.0000

```

## 2. Analisis LOOP

Untuk menghitung dan menentukan besarnya arus yang mengalir pada sebuah rangkaian listrik tertutup seperti gambar 8.4, dapat dengan mudah dianalisis berbantu MATLAB menggunakan rumus yang sudah dipelajari pada teori matakuliah sebelumnya.



**Gambar 8.4** Model Rangkaian untuk Analisis LOOP

$$\begin{bmatrix} 40 & -10 & -30 \\ -10 & 30 & -5 \\ -30 & -5 & 65 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 0 \\ 0 \end{bmatrix}$$

3. Script MATLAB yang dituliskan pada editor:

```

13
Z = [40 -10 -30;
-10 30 -5;
-30 -5 65];
V = [10 0 0]';
I = inv(Z)*V;
IRB = I(3) - I(2);
fprintf('the current through R is %8.3f Amps \n',IRB)
PS = I(1)*10;
fprintf('the power supplied by 10V source is %8.4f watts \n',PS)
diary

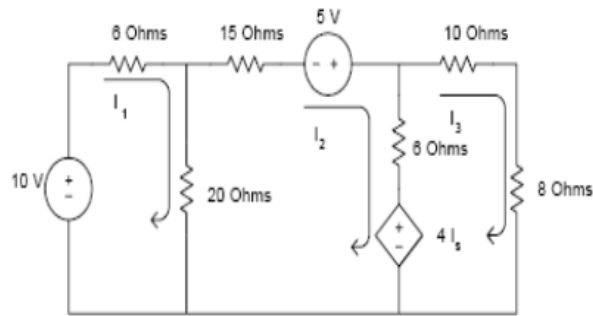
```

Bersanya nilai arus dan tegangan yang ditampilkan pada command windows adalah:

```

R = 0.037 Amps
V = 10V
Daya = 4.7531 watts

```



Gambar 8.5 Model Rangkaian untuk Analisis LOOP untuk Contoh 4

#### Contoh 4.

Solusi untuk menghitung besarnya arus pada rangkaian tertutup gambar 8.5 diselesaikan seperti script berikut.

$$\begin{bmatrix} 26 & -20 & 0 \\ -16 & 41 & -6 \\ -4 & -6 & 24 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 5 \\ 0 \end{bmatrix}$$

Pada nilai hambatan 8 Ohm, maka besarnya daya:

$$P = R_3 \cdot I^2$$

$$P = 8 \cdot I^2$$

untuk  $I_3 = I_1$ , script MATLAB yang dituliskan pada editor:

```
Z = [26 -20 0;
-16 40 -6;
16 6 24];
V = [10 5 0]';
I = inv(Z)*V;
P = 8*I(3)^2;
fprintf('Power dissipated in 8 ohm resistor is %8.2f Watts\n',P)
fprintf('Current in 10V source is %8.2f Amps\n',I(1))
diary
```

maka hasilnya adalah:

$$R = 0.037 \text{ Amps}$$

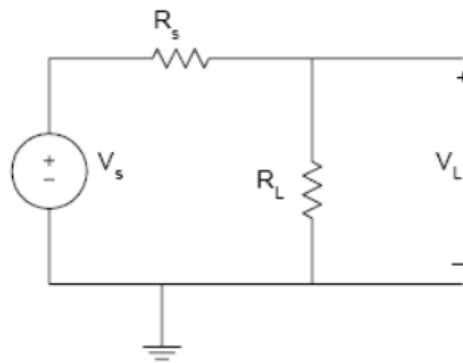
$$P = 4.7531 \text{ watts pada tegangan 10 Volt}$$

$$P = 0.44 \text{ Watts pada hambatan 8 Ohm}$$

$I = 0.73 \text{ A}$  pada tegangan 10 Volt

### 3. Maximum Power Transfer

Asumsikan bahwa sumber tegangan  $V_s$  dihubungkan dengan hambatan  $R_s$  dan sebuah beban  $R_L$  seperti ditunjukkan pada gambar 8.6 berikut.



**Gambar 8.6** Rangkaian Disipasi Daya Maksimum

$$V_L = \frac{V_s R_L}{R_s + R_L} \quad P_L = \frac{V_L^2}{R_L} = \frac{V_s^2 R_L}{(R_s + R_L)^2}$$

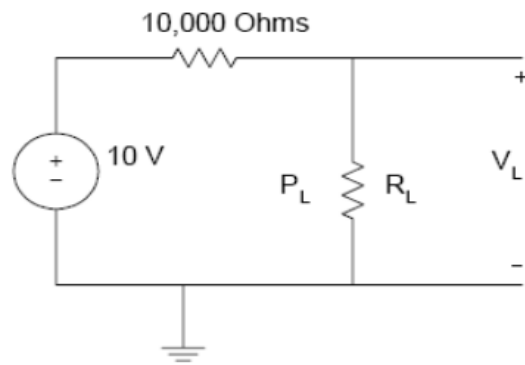
$$\frac{dP_L}{dR_L} = \frac{(R_s + R_L)^2 V_s - V_s^2 R_L (2)(R_s + R_L)}{(R_s + R_L)^4}$$

$$\frac{dP_L}{dR_L} = 0$$

$$(R_s + R_L) - 2R_L = 0 \quad R_L = R_s$$

#### Contoh 5.

Untuk nilai  $R_L$  yang bervariasi antara 0 s.d 50 kOhm. Bagaimana tegangan yang muncul pada nilai  $R_L = 10 \text{ kOhm}$  sebagaimana ditunjukkan oleh gambar 8.7 berikut.



**Gambar 8.7** Rangkaian Resistif untuk menguji Daya Maksimum Rangkaian

script MATLAB yang dituliskan pada editor:

```

16 rl = 0:1e3:50e3;
k = length(rl); % components in vector rl
for i=1:k
pl(i) = ((vs/(rs+rl(i)))^2)*rl(i);
end
dp = diff(pl)./diff(rl);
30 rld = rl(2:length(rl)); % length of rld is 1 less than that of rl
prod = dp(1:length(dp) - 1).*dp(2:length(dp));
crit_pt = rld(find(prod < 0));
max_power = max(pl); % maximum power is calculated
fprintf('Maximum power occurs at %8.2f Ohms\n',crit_pt)
fprintf('Maximum power dissipation is %8.4f Watts\n', max_power)
plot(rl,pl,'+')
13 title('Power delivered to load')
xlabel('load resistance in Ohms')
ylabel('power in watts')

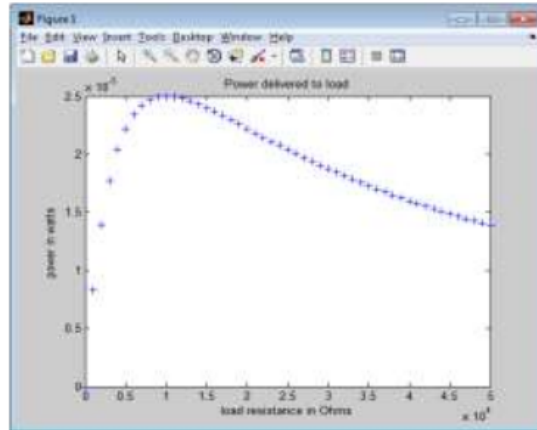
```

Hasil yang ditampilkan adalah:

**Daya maksimum pada hambatan 10000.00 Ohm**  
 **$P_{Max} = 0.0025$  Watt**

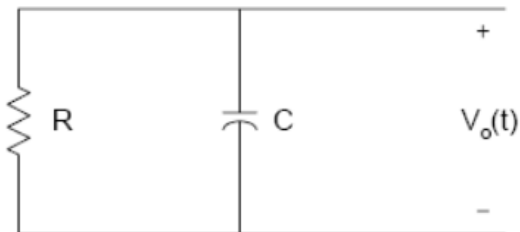
Sedangkan grafik daya maksimum ditunjukkan seperti tampilan gambar 8.8 berikut.





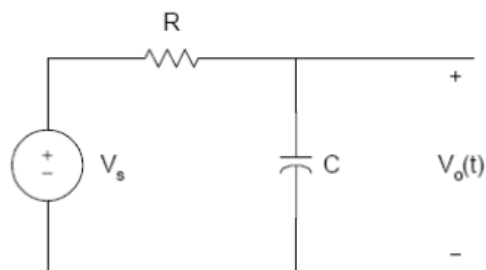
**Gambar 8.8** Tampilan Grafik Daya Maksimum untuk Contoh 5.

#### 4. Analisis Transien Jaringan RC



**Gambar 8.9** Jaringan RC sumber terbuka

$$C \frac{dv_o(t)}{dt} + \frac{v_o(t)}{R} = 0 \quad \frac{dv_o(t)}{dt} + \frac{v_o(t)}{CR} = 0 \quad v_o(t) = V_m e^{-\left(\frac{t}{CR}\right)}$$



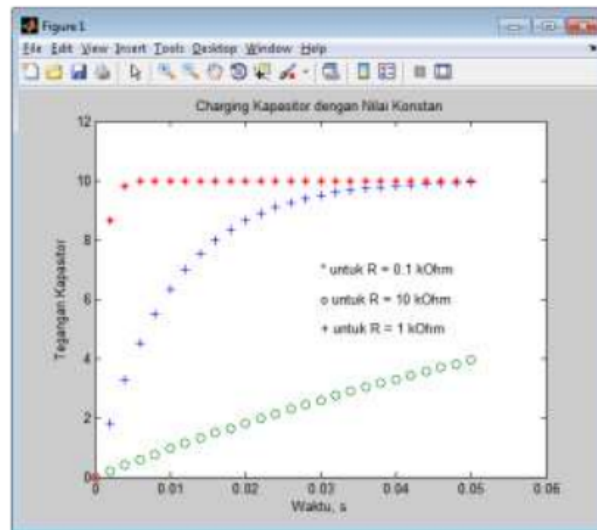
**Gambar 8.10** Charging Kapasitor

Dengan mengimplementasikan Hukum Kirchoff (KCL), diperoleh rumus berikut.

$$C \frac{dv_o(t)}{dt} + \frac{v_o(t) - V_s}{R} = 0 \quad v_o(t) = V_s \left( 1 - e^{-\left(\frac{t}{CR}\right)} \right)$$

### Contoh 6.

Jika  $C = 10\text{F}$  untuk  $R = 1\text{ kOhm}$ ,  $10\text{ kOhm}$ , dan  $0,1\text{ kOhm}$  adalah:



Gambar 8.11 Charging Kapasitor untuk contoh 6

Script MATLAB untuk gambar 8.11 dituliskan sebagai berikut.

```
30
r1=1e3;
tau1 = c*r1;
t = 0:0.002:0.05;
v1 = 10*(1-exp(-t/tau1));
r2 = 10e3;
tau2 = c*r2;
v2 = 10*(1-exp(-t/tau2));
r3 = .1e3;
tau3 = c*r3;
v3 = 10*(1-exp(-t/tau3));
plot(t,v1,'+',t,v2,'o', t,v3,'*')
axis([0 0.06 0 12])
title('Charging Kapasitor dengan Nilai Konstan')
```

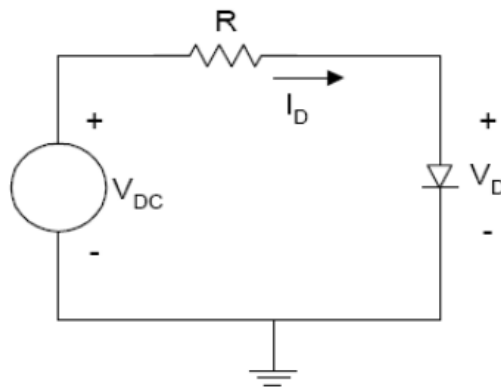
```

xlabel('Waktu, s')
title('Tegangan Kapasitor')
text(0.03, 5.0, '+ untuk R = 1 kOhm')
text(0.03, 6.0, 'o untuk R = 10 kOhm')
text(0.03, 7.0, '* untuk R = 0.1 kOhm')

```

## B. ANALISIS RANGKAIAN DIODA

Arus dan tegangan diode pada sebuah rangkaian DC seperti ditunjukkan pada gambar 8.12, dapat dianalisis dengan terlebih dahulu menentukan model formula dari rangkaian tersebut.



**Gambar 8.12** Rangkaian Dasar Dioda

Penggunaan hukum Kirchoff pada rangkaian diode, berturut-turut:

$$V_{DC} = RI_D + V_D \quad i_D = I_S e^{(v_D/nV_T)}$$

Jika pada gambar 8.12 nilai hambatan 10 kOhm dan Tegangan DC 10 Volt, arus saturasi Reverse Dioda  $10^{-12}$  amper pada  $n = 2$  dan temperature  $25^\circ\text{C}$ . Gambarkan karakteristik kurva diode forward bias.

### Solusi:

```

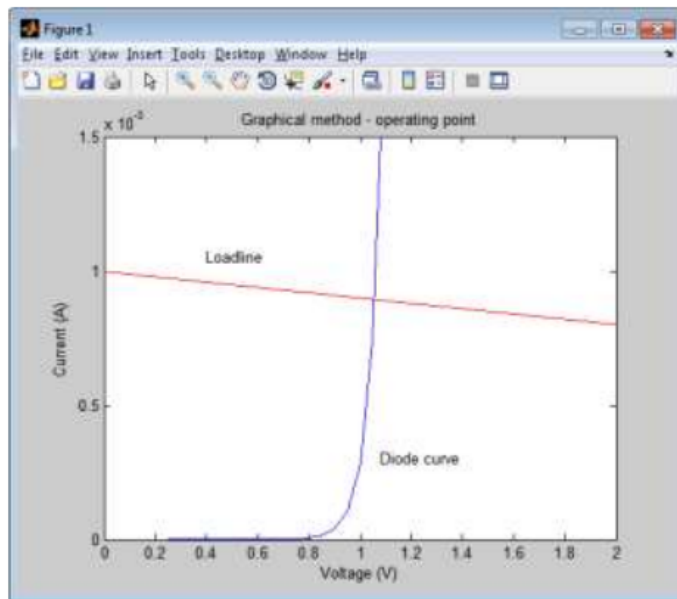
22
k = 1.38e-23;q = 1.6e-19;
t1 = 273 + 25; vt = k*t1/q;
v1 = 0.25:0.05:1.1;
i1 = 1.0e-12*exp(v1/(2.0*vt));
% load line 10=(1.0e4)i2 + v2
vdc = 10;

```

```

r = 1.0e4;
v2 = 0:2:10;
i2 = (vdc - v2)/r;
% plot
plot(v1,i1,'b', v2,i2,'r')
axis([0,2, 0, 0.0015])
title('Graphical method - operating point')
xlabel('Voltage (V)')
ylabel('Current (A)')
text(0.4,1.05e-3,'Loadline')
text(1.08,0.3e-3,'Diode curve')

```

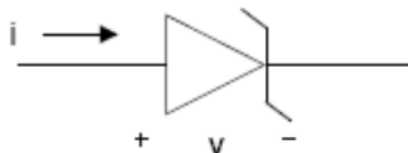


**Gambar 8.13** Karakteristik Garis Beban Dioda Forward Bias

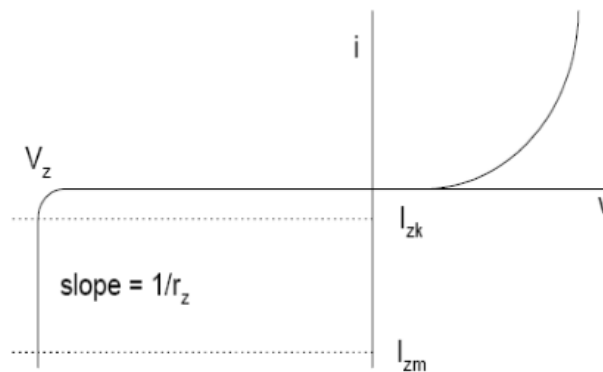
165

### C. KARAKTERISTIK DIODA ZENER

Dioda Zener adalah pn junction Dioda yang bekerja atas dasar pengontrolan pada tegangan reverse-bias. Simbol dan karakteristik Dioda Zener diperlihatkan pada gambar berikut.



**Gambar 8.14** Simbol Dioda Zener



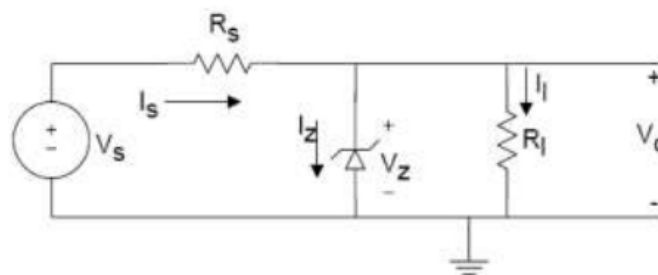
**Gambar 8.15** Karakteristik I-V Dioda Zener

Maka beberapa persamaan yang berlaku pada rangkaian diode Zener untuk disipasi daya dan arus diode zener, ditunjukkan pada persamaan di bawah ini.

$$I_{ZM} = \frac{P_Z}{V_Z} \qquad r_z = \frac{\Delta V_Z}{\Delta I_Z}$$

**Contoh.**

Jika sebuah diode zener dihubungkan dengan sebuah rangkaian tertutup yang dihubungkan dengan sebuah tegangan sumber dan beberapa hambatan seperti gambar berikut.



**Gambar 8.16** Rangkaian Dioda Zener Shunt pada Regulator Tegangan

$$V_o = V_z \qquad I_s = I_z + I_L$$

$$I_S = \frac{V_S - V_Z}{R_S} \qquad I_L = \frac{V_O}{R_L}$$

$$R_S = \frac{V_{S,\min} - V_Z}{I_{L,\max} + I_{Z,\min}} \qquad R_S = \frac{V_{S,\max} - V_Z}{I_{L,\min} + I_{Z,\max}}$$

$$(V_{S,\min} - V_Z)(I_{L,\min} + I_{Z,\max}) = (V_{S,\max} - V_Z)(I_{L,\max} + I_{Z,\min})$$

$$I_{Z,\min} = 0.1I_{Z,\max}$$

$$I_{Z,\max} = \frac{I_{L,\min}(V_Z - V_{S,\min}) + I_{L,\max}(V_{S,\max} - V_Z)}{V_{S,\min} - 0.9V_Z - 0.1V_{S,\max}}$$

Jika pada gambar. 16 di atas diberikan nilai-nilai sebagai berikut:

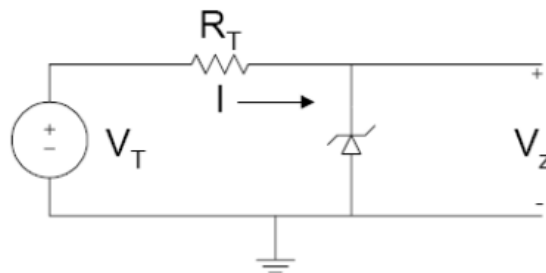
$$30 \leq V_S \leq 35\text{V}; \quad R_L = 10\text{K}, \quad R_S = 2\text{K} \quad V_Z = -20 + 0.05I$$

untuk  $-100 \text{ mA} \leq I < 0$ , maka menggunakan analisa berbantu MATLAB.

1. Tentukan grafik karakteristik tegangan breakdown zener!
2. Gambarkan grafik garis beban untuk tegangan  $V_S = 30$  volt dan  $V_S = 35$  volt
3. Hitunglah tegangan output pada saat  $V_1 = 30$  volt dan  $V_2 = 35$  volt

### SOLUSI:

Jika digunakan Theorema Thevenin maka dapat digambarkan rangkaian ekivalen sebagai berikut.



**Gambar 8.17** Rangkaian Ekivalen untuk soal gambar.16

$$V_T = \frac{V_S R_L}{R_L + R_S} \qquad R_T = R_L \parallel R_S$$

Pada saat:

$$R_L = 10\text{K}, R_S = 2\text{K}, R_T = (10)(2\text{K}) / 12 \text{K} = 1.67 \text{K}\Omega$$

Dan

$$V_S = 30\text{V}, V_T = (30)(10) / 12 = 25 \text{V}$$

$$V_S = 35\text{V}, V_T = (35)(10) / 12 = 29.17 \text{V}$$

Maka:

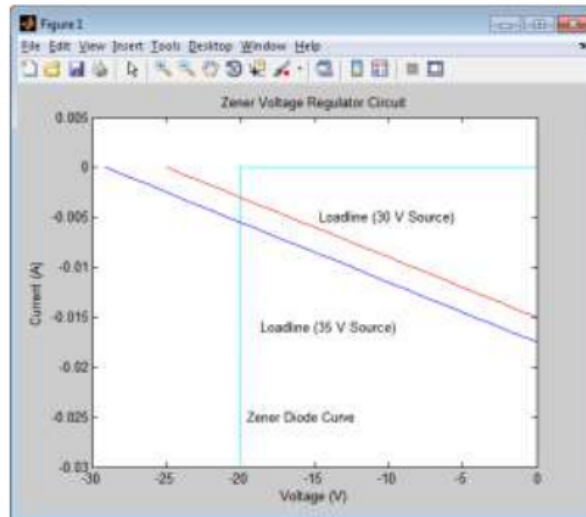
$$V_Z = V_T - R_T I = -20 + 0.05I$$

$$\Rightarrow I = \frac{(V_T + 20)}{R_T + 0.05}$$

$$V_Z = V_T - R_T I = V_T - \frac{R_T (V_T + 20)}{R_T + 0.05}$$

Komputasi dan tampilan grafik dapat dibuat berbantu MATLAB dengan menuliskan sintaks sebagai berikut.

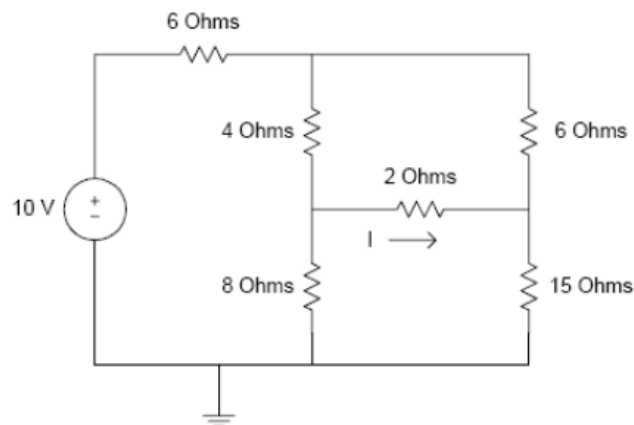
```
ip1 = -0.0030      vp1 = -20.0001
ip2 = -0.005      vp2 = -20.0003
ip1 = -0.0030      vp1 = -20.0001
ip2 = -0.0055      vp2 = -20.0003
```



**Gambar 8.18** Karakteristik Garis Beban Dioda Zener

## PERTANYAAN DAN SOAL

Hitunglah arus yang mengalir pada rangkaian di bawah ini dan tampilkan grafik karakteristiknya menggunakan MATLAB!!





## BAB 9

### PENGANTAR TEORI TENAGA LISTRIK

---

20

#### TUJUAN INSTRUKSIONAL:

Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:

- 1) Mengerti tentang klasifikasi dan konsep dasar motor listrik
  - 2) Mengerti konsep dasar karakteristik motor listrik
  - 3) Mengerti rumus-rumus dasar untuk proses komputasi karakteristik motor-motor listrik DC dan AC
  - 4) Mampu membedakan karakteristik motor-motor listrik untuk divisualisasikan menggunakan proses komputasi MATLAB
- 

65 Karakteristik motor-motor listrik layak diketahui untuk mencerminkan unjuk kerja dari motor listrik tersebut selama kondisinya, pada mata kuliah teknik tenaga listrik mahasiswa dapat mengetahui karakteristik dari setiap motor listrik dengan membuat grafik atau kurva karakteristik motor dengan cara perhitungan secara konvensional, begitu pula pada praktikum motor-motor listrik karena keterbatasan alat dan bahan pada laboratorium mesin-mesin listrik khususnya untuk pengamatan grafik sebagai sarana pendukung pada praktikum motor-motor listrik menyebabkan kurang dan terlambatnya mahasiswa dalam memahami dan menganalisa karakteristik dari motor listrik. Oleh karena itu, diperlukan suatu alternatif untuk mengefektifkan proses pembelajaran mata kuliah teknik tenaga listrik khususnya untuk memudahkan mahasiswa dalam menganalisa karakteristik motor-motor listrik serta alternatif pengganti untuk mengeliminasi keterbatasan alat dan bahan praktikum proses pelaksanaan praktikum motor-motor listrik.

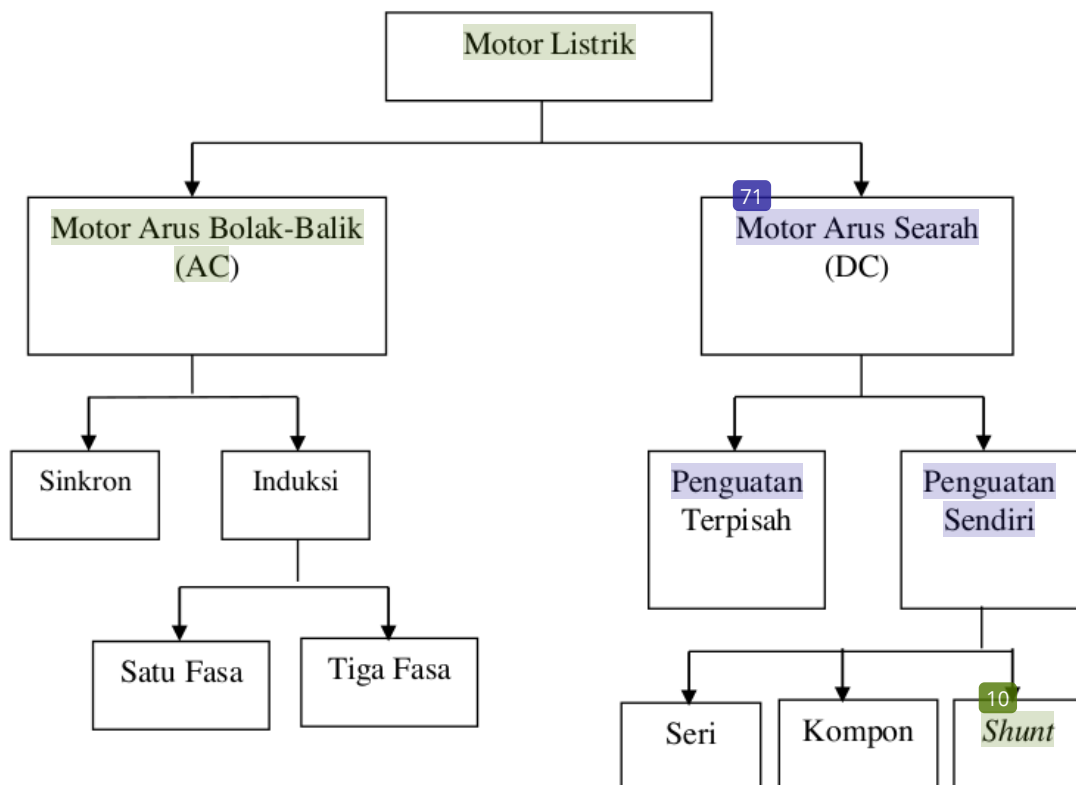
Salah satu alat pengolahan data yang memungkinkan simulasi sistem dinamis berbentuk grafik dengan cara yang sederhana adalah GUI (*Graphical User Interface*) MATLAB, dalam MATLAB fasilitas GUI dipakai untuk media tampilan grafik sebagai pengganti perintah teks untuk *users* berinteraksi. Hal tersebut dapat dilakukan karena kekayaan akan fungsi matematika, fisika dan visualisasi. Berdasarkan uraian diatas maka pada penelitian ini akan dirancang laboratorium virtual untuk studi analisis karakteristik motor-motor listrik menggunakan GUI (*Graphical User Interface*) MATLAB yang diharapkan dapat dimanfaatkan sebagai

alat pembelajaran yang efektif pada mata kuliah teknik tenaga listrik dan juga membantu mengoptimalkan proses praktikum mesin-mesin listrik.

## A. LANDASAN TEORI

### 1. Motor Listrik

Motor listrik merupakan sebuah perangkat elektromagnetis yang mengubah energi listrik menjadi energi mekanik. Motor listrik mempunyai dua jenis yaitu motor arus bolak-balik (AC) dan motor arus searah (DC). Gambar 9.1 menunjukkan klasifikasi jenis utama dari motor listrik.



Gambar 9.1 Klasifikasi Jenis Utama Motor Listrik

### 2. Motor DC

Motor arus searah atau motor DC adalah mesin yang merubah energi listrik arus searah menjadi energi mekanik yang berupa putaran. Berdasarkan fisiknya motor arus searah secara umum terdiri atas bagian yang diam dan bagian yang berputar, pada bagian yang diam (*stator*) merupakan tempat diletakkannya

19

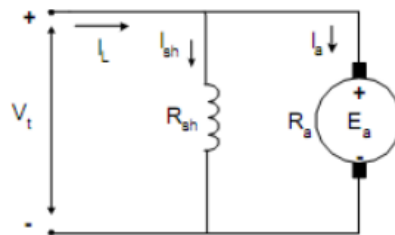
kumparan medan yang berfungsi untuk menghasilkan fluksi magnet sedangkan pada bagian yang berputar (*rotor*) ditempati oleh rangkaian jangkar seperti kumparan jangkar, komutator dan sikat. Motor DC bekerja berdasarkan prinsip interaksi antara dua fluksi magnetik.

Kumparan medan akan menghasilkan fluksi magnet yang arahnya dari kutub utara menuju kutub selatan dan kumparan jangkar akan menghasilkan fluksi magnet yang melingkar, interaksi antara kedua fluksi magnet ini menimbulkan suatu gaya. Berdasarkan jenis penguatannya yaitu hubungan rangkaian kumparan medan dengan kumparan jangkar, motor DC dibedakan menjadi:

1. Motor DC penguatan terpisah
2. Motor DC penguatan sendiri, yang terdiri atas:
  - a. Motor DC penguatan seri.
  - b. Motor DC penguatan *shunt*.
  - c. Motor DC kompon panjang.
  - d. Motor DC kompon pendek.

a. **Motor DC Shunt**

Rangkaian ekuivalen motor DC *shunt* ditunjukkan pada Gambar 9.2 dibawah ini.



**Gambar 9.2** Rangkaian Ekuivalen Motor DC *Shunt*

28

Motor *shunt* kumparan jangkarnya dihubungkan langsung pada terminal sehingga paralel dengan kumparan jangkar. Persamaan-persamaan yang berlaku pada motor *shunt* adalah:

$$V_t = E_a + I_a \cdot R_a \quad (\text{Volt}) \quad (9-1)$$

$$I_{sh} = \frac{V_t}{R_{sh}} \quad (\text{Ampere}) \quad (9-2)$$

$$I_L = I_a + I_{sh} \quad (\text{Ampere}) \quad (9-3)$$

Keterangan:  $V_t$  : tegangan terminal (Volt)

47

$I_{sh}$  : arus kumparan medan *shunt* (Ampere)

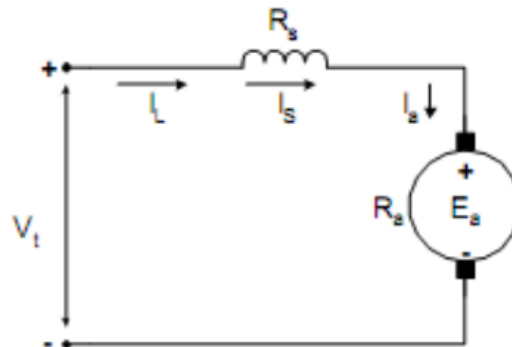
$R_{sh}$  : tahanan medan *shunt* (ohm)

$I_L$  : arus dari jala-jala (Ampere)

### b. Motor DC Seri

36

Rangkaian ekivalen motor DC seri ditunjukkan pada Gambar 9.3 dibawah ini.



**Gambar 9.3** Rangkaian Ekivalen Motor DC Seri

28

Motor DC seri, kumparan medan dihubungkan secara seri dengan rangkaian jangkar, oleh sebab itu arus yang mengalir pada kumparan medan seri sama dengan arus yang mengalir pada kumparan jangkar. Persamaan-persamaan yang berlaku pada motor arus searah penguatan seri adalah:

$$V_t = E_a + I_s \cdot R_s + I_a \cdot R_a \quad (\text{Volt}) \quad (9-4)$$

$$I_L = I_a = I_s \quad (\text{Ampere}) \quad (9-5)$$

Maka:

$$V_t = E_a + I_a (R_a + R_s) \quad (\text{Volt}) \quad (9-6)$$

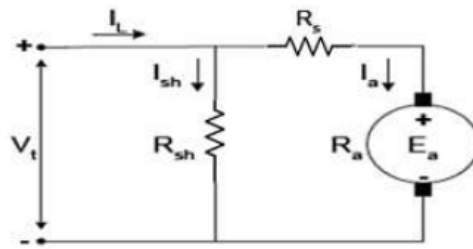
71

Keterangan:  $I_s$  : arus kumparan medan seri (Ampere)

$R_s$  : tahanan medan seri (ohm)

### c. Motor DC Kompon Panjang

Rangkaian ekivalen motor DC penguatan kompon panjang ditunjukkan pada Gambar 9.4.



Gambar 9.4 Rangkaian Ekuivalen Motor DC Kompon Panjang

Motor DC penguatan kompon panjang dan medan stator merupakan gabungan antara kumparan seri dan kumparan *shunt*, pada motor ini nilai arus yang masuk rotor sama dengan arus yang ada pada medan seri. Persamaan umum motor DC penguatan kompon panjang dan persamaan daya motor sebagai berikut:

$$I_L = I_a + I_{sh} \text{ (Ampere)} \quad (9-7)$$

$$V_t = E_a + I_a \cdot (R_s + R_a) \text{ (Volt)} \quad (9-8)$$

$$P_{in} = V_t \cdot I_L \text{ (Watt)} \quad (9-9)$$

$$I_L \cdot R_s = \text{tegangan jatuh pada kumparan seri (Ampere)} \quad (9-10)$$

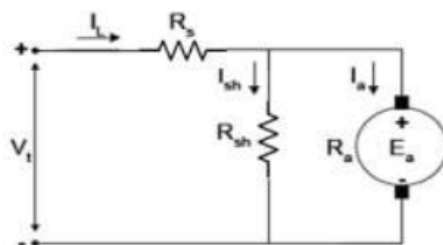
$$(I_s)^2 \cdot R_s = \text{rugi daya pada kumparan seri (Ampere)} \quad (9-11)$$

$$(I_{sh})^2 \cdot R_{sh} = \text{rugi daya pada kumparan shunt (Ampere)} \quad (9-12)$$

$$(I_a)^2 \cdot R_a = \text{rugi daya pada jangkar (Ampere)} \quad (9-13)$$

d. Motor DC Kompon Pendek

Motor DC kompon pendek adalah motor yang medan statornya merupakan gabungan antara kumparan seri dan kumparan paralel, pada motor kompon pendek arus masukan akan terbagi dua yaitu arus masuk ke medan *start* dan sebagian masuk ke medan seri. Rangkaian ekuivalen motor DC penguatan kompon panjang ditunjukkan pada Gambar 9.5 dibawah ini.



Gambar 9.5 Rangkaian Ekuivalen Motor DC Kompon Pendek

Persamaan umum motor arus searah penguatan kompon pendek:

$$I_L = I_a + I_{sh} \text{ (Ampere)} \quad (9-14)$$

$$V_t = E_a + I_L \cdot R_s + I_a \cdot R_a \text{ (Volt)} \quad (9-15)$$

$$P_{in} = V_t \cdot I_L \text{ (Watt)} \quad (9-16)$$

110

### 3. Persamaan Kecepatan Pada Motor DC

Pengaturan kecepatan pada motor DC berlaku persamaan:

$$E_a = V_t - I_a R_a \text{ (Volt)} \quad (9-17)$$

$$E_a = \frac{P\Phi Z n}{60A} \text{ (Volt), sehingga:} \quad (9-18)$$

$$\frac{P\Phi Z n}{60A} = V_t - I_a R_a, \text{ atau:} \quad (9-19)$$

$$n = \frac{V_t - I_a R_a}{\Phi} \frac{60 A}{PZ} \text{ (Rpm), atau:} \quad (9-20)$$

$$n = K \frac{V_t - I_a R_a}{\Phi}, \text{ dengan } K = \frac{60 A}{PZ} \quad (9-21)$$

$$n = K \frac{E_a}{\Phi} \text{ (Rpm)} \quad (9-22)$$

Motor DC seri, ambil kondisi awal:

$$n_1 \propto \frac{E_{a1}}{\Phi_1}, \text{ dengan } E_{a1} = V_t - I_{a1} R_a \quad (9-23)$$

$$n_2 \propto \frac{E_{a2}}{\Phi_2}, \text{ dengan } E_{a2} = V_t - I_{a2} R_a \quad (9-24)$$

$$\text{Jadi } \frac{n_2}{n_1} = \frac{E_{a2}}{E_{a1}} \times \frac{\Phi_1}{\Phi_2} \quad (9-25)$$

Sebelum kutub-kutub magnet jenuh, maka  $\Phi \propto I_a$

$$\text{Jadi } \frac{n_2}{n_1} = \frac{E_{a2}}{E_{a1}} \times \frac{I_{a1}}{I_{a2}} \quad (9-26)$$

Untuk motor DC *shunt*:

Persamaan dasar yang digunakan sama dengan motor DC seri:

$$\frac{n_2}{n_1} = \frac{E_{a2}}{E_{a1}} \times \frac{\Phi_1}{\Phi_2}, \text{ jika } \Phi_1 = \Phi_2;$$

Maka  $\frac{n_2}{n_1} = \frac{E_{a2}}{E_{a1}}$ , dalam praktek untuk motor shunt  $\Phi$  tetap (9-27)

Keterangan: konstanta (bergantung pada ukuran fisik motor)

$\Phi$  : fluksi setiap kutub (Weber)

$I_a$  : arus jangkar (Ampere)

$P$  : jumlah kutub

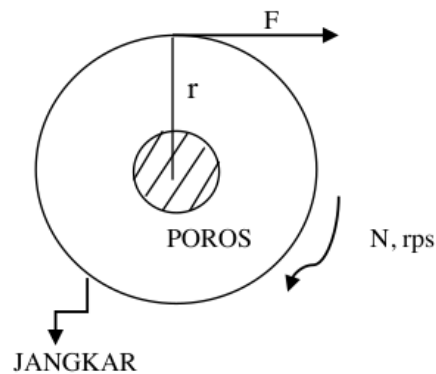
$Z$  : jumlah konduktor

$A$  : cabang paralel

$n$  : kecepatan (Rpm)

#### 4. Torsi Motor

Besarnya torsi yang dihasilkan oleh motor DC dapat diketahui dengan melakukan analisis seperti pada Gambar 9.6 sebagai berikut:



Gambar 9.6 Jangkar Motor

$$T = F \times r \quad (9-27)$$

Newton meter = T Joule

Usaha yang dilakukan oleh gaya  $F$  dalam 1 putaran:  $F \times 2\pi R$  joule

Kerja yang dilakukan per detik:

$$W = \frac{F \times 2\pi R}{P} = F \times 2\pi r \cdot N \quad (9-28)$$

$$W = (F \times r) \times 2\pi N \text{ (Joule)} \quad (9-29)$$

$$\omega = 2\pi N \quad (9-30)$$

$$\text{Jadi, usaha yang dilakukan per detik} = T \times \omega \text{ (Joule)} \quad (9-31)$$

Maka, tenaga yang menggerakkan =  $T \times \omega$  (Watt) (9-32)

Torsi jangkar dari motor dengan kecepatan N rps, jika T dalam Nw-m maka usaha yang dilakukan per detik =  $T \times 2 \pi N$  (Watt) (9-33)

Tenaga listrik yang dirubah ke dalam tenaga mekanik dalam jangkar adalah:  
 $P_{mek} = E_a I_a$  (Watt) (9-34)

Jadi:  $T \times 2 \pi N = E_a I_a$ ,

atau  $T = \frac{E_a I_a}{2 \pi N}$  (Nw-m) =  $0.159 \frac{E_a I_a}{N}$  (Nw-m) =  $0.0162 \frac{E_a I_a}{N}$  (kg-m) (9-35)

jika  $E_a = \Phi z N \left(\frac{P}{A}\right)$  (Volt)

$T = \frac{\Phi z N \left(\frac{P}{A}\right) I_a}{2 \pi N}$  (Nw-m) (9-36)

$T = 0.159 \Phi z I_a \left(\frac{P}{A}\right)$  (Nw-m) (9-37)

$T = 0.0162 \Phi z I_a \left(\frac{P}{A}\right)$  (Kg-m) (9-38)

150

## 5. Karakteristik Motor Dc

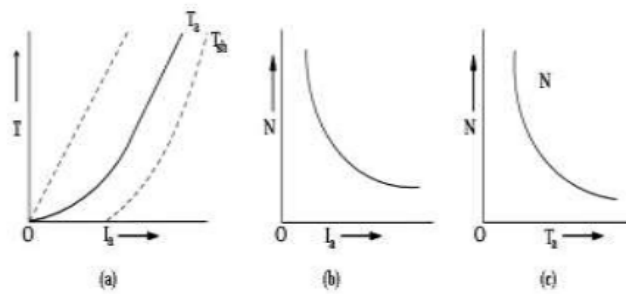
### a. Karakteristik Motor DC Seri

Karakteristik motor DC seri dapat dilihat dari karakteristik torsi terhadap arus jangkar, karakteristik kecepatan dengan arus jangkar dan karakteristik kecepatan dengan torsi. Karakteristik motor DC seri ditunjukkan dengan penambahan arus jangkar  $I_a$  yang menyebabkan fluks bertambah dan torsi juga besar.

Perubahan kecepatan sesuai dengan rumus  $N \sim \frac{E_a}{\Phi}$ , perubahan  $E_a$  terhadap arus beban adalah kecil dan dapat diabaikan untuk waktu tertentu, jika  $I_a$  bertambah maka  $\Phi$  juga bertambah atau  $I_a \sim \Phi$ , jadi  $N \sim \frac{E_a}{I_a}$ .

Karakteristik N/T, kecepatan dengan torsi disebut juga karakteristik mekanis dapat diperoleh dari  $T \sim I_a^2$  jadi  $T \sim \frac{1}{N^2}$ . Ketiga karakteristik dari motor DC seri dapat dilihat pada Gambar 9.7.



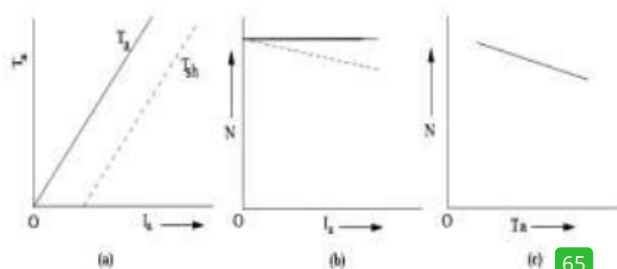


**Gambar 9.7** Kurva Karakteristik Motor DC Seri. (a) Karakteristik Torsi-Arus Jangkar; (b) Karakteristik Kecepatan-Arus Jangkar; (c) Karakteristik Kecepatan-Torsi.

**b. Karakteristik Motor DC Shunt**

Telah diketahui bahwa dalam motor DC,  $T_a \sim I_a \Phi$  karena beroperasi dari suatu tegangan sumber yang konstan fluksi juga konstan (dengan mengabaikan reaksi jangkar), maka  $T_a \sim I_a$ , jelas terlihat pada kurva bahwa arus yang sangat besar dibutuhkan untuk *start* beban yang berat, oleh karena itu motor DC shunt tidak boleh *start* dalam keadaan berbeban berat seperti ditunjukkan dalam kurva karakteristik torsi dengan arus jangkar.

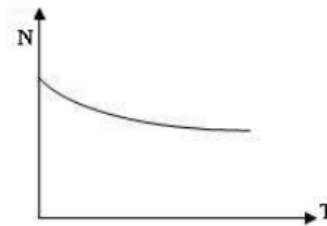
Kecepatan motor DC seperti yang telah kita ketahui, menunjukkan  $n \sim \frac{E_a}{\Phi}$ , fluksi dan GGL lawan  $E_a$  dalam motor DC shunt hampir konstan dibawah kondisi normal, dengan demikian kecepatan motor DC shunt selalu konstan walaupun arus jangkar berubah-ubah nilainya, dengan kata lain ketika beban bertambah dan fluks berkurang karena drop tahanan jangkar dan reaksi jangkar,  $E_a$  berkurang lebih sedikit daripada fluks sehingga dengan demikian kecepatan motor menurun sedikit dengan penambahan beban yang ditunjukkan dengan kurva karakteristik kecepatan dengan arus jangkar dan torsi.



**Gambar 9.8** Kurva Karakteristik Motor DC Shunt. (a) Karakteristik Torsi-Arus Jangkar; (b) Karakteristik Kecepatan-Arus Jangkar; (c) Karakteristik Kecepatan-Torsi.

### c. Karakteristik Motor DC Kompon

Motor DC kompon mempunyai komponen fluks yang konstan dan komponen lainnya yang sebanding terhadap arus jangkarnya dan juga bebannya, Karena itu motor kompon kumulatif memiliki kopel mula (*starting*) lebih besar dari pada motor DC lainnya. Motor DC kompon kumulatif mempunyai kecepatan tanpa beban terbatas dan dapat dioperasikan dengan aman pada keadaan tanpa beban, Kurva karakteristik dari motor DC kompon ditunjukkan pada Gambar 2.9 dibawah ini.

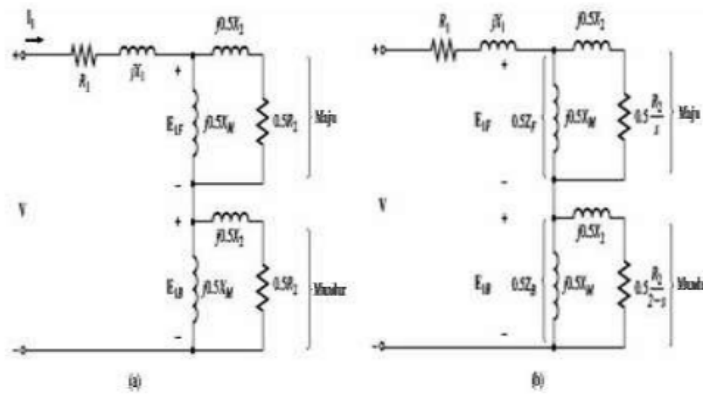


**Gambar 9.9** Kurva Karakteristik Kecepatan dan Torsi Motor DC Kompon

## 6. Motor Induksi Satu Fasa

Motor induksi satu fasa memiliki dua belitan stator, yaitu belitan fasa utama dan belitan fasa bantu. Belitan fasa utama menggunakan penampang kawat tembaga lebih besar sehingga memiliki impedansi lebih kecil, sedangkan belitan bantu dibuat dari tembaga berpenampang kecil dan jumlah belitannya lebih banyak sehingga impedansinya lebih besar dibanding impedansi belitan utama. Motor induksi satu fasa memiliki beberapa jenis yaitu motor fasa belah, motor kapasitor, dan motor kutub bayangan (*shaded pole*).

Rangkaian ekuivalen motor induksi satu fasa ditunjukkan pada Gambar 9.10, pada Gambar 2.10(a) motor dalam keadaan diam atau belum berputar, sementara pada gambar 2.10(b) motor dalam keadaan beroperasi. Rangkaian ekuivalen motor induksi ini dipengaruhi oleh medan magnet maju dan mundur yang digambarkan secara terpisah.



98

**Gambar 9.10** Rangkaian Ekivalen Motor Induksi Satu Fasa. (a) Rangkaian Ekivalen Motor Induksi Keadaan Diam; (b) Rangkaian Ekivalen Motor Induksi Dalam Keadaan Beroperasi.

Persamaan umum yang didapat dari rangkaian ekivalen di atas:

$$Z_F = R_F + jX_F = \frac{(R_2/s + jX_2)(jX_M)}{(R_2/s + jX_2) + jX_M} \quad (9-39)$$

$$Z_B = R_B + jX_B = \frac{[R_2/(2-s) + jX_2](jX_M)}{[R_2/(2-s) + jX_2] + (jX_M)} \quad (9-40)$$

$$I_1 = \frac{V}{R_1 + jX_1 + 0.5 Z_F + 0.5 Z_B} \quad (9-41)$$

50

Daya celah udara untuk medan magnet maju didapatkan:

$$P_{AG\ F} = I_1^2 (0.5 R_F) \quad (9-42)$$

Daya celah udara untuk medan magnet mundur didapatkan:

$$P_{AG\ B} = I_1^2 (0.5 R_B) \quad (9-43)$$

50

Total daya celah udara pada motor induksi satu fasa adalah:

$$P_{AG} = P_{AG\ F} - P_{AG\ B} \quad (9-44)$$

Jadi torsi induksi didapatkan:

$$T_{ind} = \frac{P_{AG}}{\omega_s} \quad (9-45)$$

Perubahan daya dari daya elektrik ke daya mekanik untuk motor induksi satu fasa didapatkan:

$$P_{conv} = T_{ind} \omega_m \quad (9-46)$$

Karena  $\omega_m = (1 - s)\omega_s$

Maka:

$$P_{conv} = T_{ind} (1 - s)\omega_s \quad (9-47)$$

Dapat juga dituliskan:

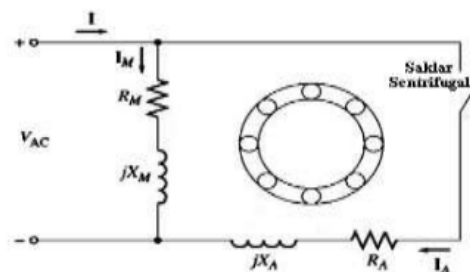
$$P_{conv} = (1 - s)P_{AG} \quad (9-48)$$

### a. Motor Fasa belah

78

Motor fasa belah yang terdiri atas dua kumparan, kumparan utama dan kumparan bantu, antara kumparan utama dan kumparan bantu berbeda arus 90 derajat listrik dan juga dilengkapi dengan saklar sentrifugal, pada saat motor belum bekerja saklar sentrifugal dalam keadaan menutup. Kumparan utama selalu dirancang mempunyai nilai resistansi rendah dan nilai reaktansi tinggi dibanding dengan kumparan bantu yang selalu mempunyai nilai reaktansi rendah dan resistansi tinggi. Kedua kumparan ini dihubungkan ke sumber jala-jala dengan kondisi nilai resistansi dan reaktansi kumparan masing-masing tidak sama nilainya maka sudut fasa arus yang mengalir melalui kumparan utama.

Kumparan utama bekerja terus menerus selama motor berputar, sedangkan kumparan bantu bekerja pada saat motor start awal karena setelah motor berputar 70-80% dari putaran nominal saklar sentrifugal memutuskan kumparan bantu. Kumparan bantu selain untuk membantu menghasilkan momen kopel pada saat motor start awal juga berfungsi untuk menentukan arah berputarnya poros motor. Gambar 9.11 menunjukkan rangkaian ekuivalen motor fasa belah.

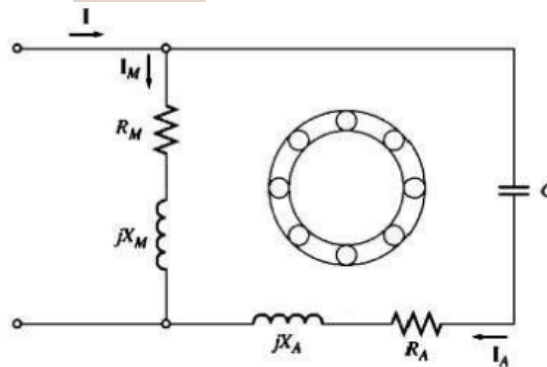


Gambar 9.11 Rangkaian Ekuivalen Motor Fasa Belah

19

## b. Motor Kapasitor

Motor kapasitor adalah motor arus bolak-balik satu fasa yang kumparan bantunya dipasang seri dengan kap. Kapasitor pada motor ini berfungsi untuk memperoleh beda fasa  $90^0$  antara arus kumparan utama dan arus kumparan bantu dan diperoleh kopel mula yang lebih besar sehingga rotor dapat putar. Rangkaian ekivalen dari motor kapasitor permanen ditunjukkan Gambar 9.12.



Gambar 9.12 Rangkaian Ekivalen Motor Kapasitor Permanen

Motor kapasitor memiliki beberapa tipe yaitu:

### 1) Motor Kapasitor Jalan (*Run Capacitor*)

Motor *run capacitor* dilengkapi dengan sebuah kapasitor jalan (*run capacitor*). Motor ini tidak dilengkapi dengan saklar sentrifugal sehingga kumparan bantu dan *run capacitor* bekerja terus menerus selama motor berputar, *run capacitor* selain berfungsi untuk menghasilkan momen kopel pada saat *start* awal motor juga berfungsi untuk memperbaiki faktor daya.

### 2) Motor Kapasitor *Start* (*Start Capacitor*)

Motor *start capacitor* dilengkapi dengan sebuah kapasitor *start* (*start capacitor*) dan saklar sentrifugal. *Start capacitor* berfungsi untuk menghasilkan momen kopel yang besar, hal ini disebabkan kapasitas kapasitonya besar. Kumparan bantu selain diseri dengan saklar sentrifugal juga diseri dengan kapasitor *start*. Kumparan bantu dan kapasitor akan diputuskan oleh saklar sentrifugal setelah motor berputar 70-80% dari putaran nominal.

27

### 3) Motor Kapasitor *Start* dan *Run*

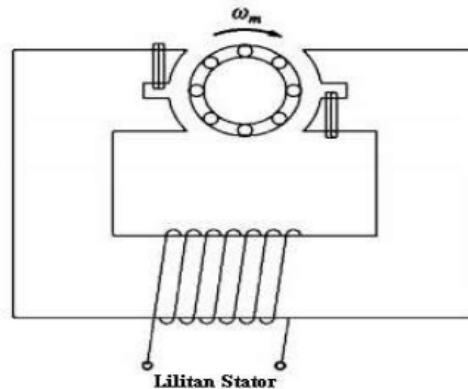
Motor kapasitor *start* dan *run* terdapat dua buah kapasitor yang terhubung semua sehingga diperoleh beda fasa yang cukup besar antara arus pada lilitan utama dan arus pada lilitan bantu dan diperoleh torsi

awal yang sangat besar, setelah putaran motor mencapai 70-80% putaran normal.

91

### c. Motor Kutub Bayangan

Motor kutub bayangan (*shaded pole motors*) yaitu motor induksi satu fasa yang pada kutubnya disisipkan kumparan tambahan dengan prinsip kerja apabila pasokan catu daya diberikan maka pada kumparan bayangan timbul arus induksi yang menyebabkan fluksnya tertinggal dari fluks yang berada di kumparan lain.



Gambar 9.13 Kutub Utama Motor Kutub Bayangan

24

### 7. Slip

Motor Induksi tidak dapat berputar pada kecepatan sinkron, jika hal ini terjadi maka rotor akan tetap diam relatif terhadap fluksi yang berputar, maka tidak akan ada GGL yang diinduksikan dalam rotor tidak ada arus yang mengalir pada rotor dan karenanya tidak akan menghasilkan kopel. Kecepatan rotor sekalipun tanpa beban harus lebih kecil sedikit dari kecepatan sinkron agar adanya tegangan induksi pada rotor dan akan menghasilkan arus di rotor, arus induksi ini akan berinteraksi dengan fluks listrik sehingga menghasilkan kopel. Selisih antara kecepatan rotor dengan kecepatan sinkron disebut slip. Slip dapat dinyatakan dalam putaran setiap menit tetapi lebih umum dinyatakan sebagai persen dari kecepatan sinkron.

$$s = \frac{n_s - n_r}{n_s} \times 100\% \quad (9-49)$$

172

Keterangan:  $s$  : slip

$n_s$  : kecepatan sinkron

$n_r$  : kecepatan rotor

Persamaan (2-49) di atas menunjukkan bahwa:

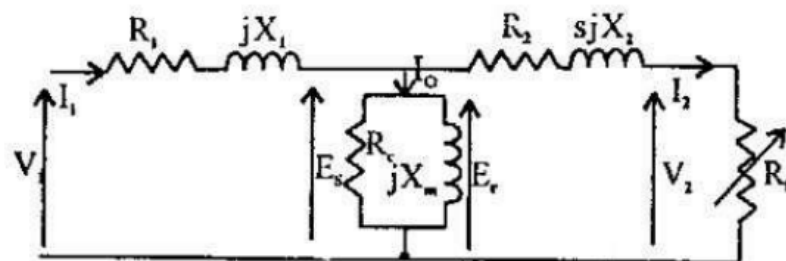
1. Saat  $s = 1$  dimana  $n_r = 0$ , berarti rotor masih dalam keadaan diam atau akan berputar.
2.  $s = 0$  menyatakan bahwa  $n_s = n_r$ , ini berarti rotor berputar sampai kecepatan sinkron. Hal ini dapat terjadi jika ada arus dc yang diinjeksikan ke belitan rotor, atau rotor digerakkan secara mekanik.
3.  $0 < s < 1$ , ini berarti kecepatan rotor diantara keadaan diam dengan kecepatan sinkron. Kecepatan rotor dalam keadaan inilah dikatakan kecepatan tidak sinkron. Biasanya slip untuk mendapatkan efisiensi yang tinggi pada saat beban penuh adalah 0.04.

## 8. Motor Induksi Tiga Fasa

Motor induksi merupakan motor arus bolak-balik (AC) yang paling luas digunakan dan dapat dijumpai dalam setiap aplikasi industri maupun rumah tangga. Motor induksi tiga fasa sangat banyak dipakai sebagai penggerak di perindustrian karena banyak memiliki keuntungan, tetapi ada juga kelemahannya.

Keuntungan motor induksi tiga fasa antara lain, konstruksinya yang sederhana dan kuat, biaya murah tetapi dapat diandalkan, efisiensi yang tinggi pada kondisi kerja normal, dan perawatannya mudah. Kerugian motor induksi tiga fasa diantaranya, kecepatannya tidak bisa bervariasi tanpa merubah efisiensi dan kecepatannya tergantung beban.

Motor induksi tiga fasa berdasarkan rotornya mempunyai dua jenis yaitu motor induksi tiga fasa sangkar tupai (*squirrel-cage motor*) dan motor induksi tiga fasa rotor belitan (*wound-rotor motor*). Kedua motor ini bekerja pada prinsip yang sama dan mempunyai konstruksi stator yang sama tetapi berbeda dalam konstruksi rotor. Rangkaian ekuivalen motor induksi tiga fasa ditunjukkan pada Gambar 9.14, pada prinsipnya proses induksi yang terjadi pada motor induksi hampir sama seperti pada transformator yang berbeban resistif, perbedaan mendasar antara keduanya adalah transformator merupakan mesin listrik statis sementara motor induksi adalah mesin listrik dinamis.



Gambar 9.14 Rangkaian Ekuivalen Motor Induksi Tiga Fasa

Dari rangkaian ekivalen diatas dapat dituliskan persamaan:

$$V_1 = E_s + I_1(R_1 + jX_1) \quad (9-50)$$

$$E_2 = I_2 Z_2 = I_2(R_2 + jsX_2) \quad (9-51)$$

Ketika motor dibebani, maka akan mengalir arus  $I_2$  sebesar:

$$I_2 = \frac{E_2}{\sqrt{(R_2)^2 + (sX_2)^2}} \quad (9-52)$$

3

Prinsip kerja motor induksi tiga fasa mengikuti langkah-langkah sebagai berikut:

1. Apabila catu daya arus bolak-balik tiga fasa dihubungkan pada kumparan stator (jangkar), maka akan timbul medan putar dengan kecepatan:

$$n_s = \frac{120 \times f}{p} \text{ (Rpm)} \quad (9-53)$$

68

Keterangan:  $n_s$  : Kecepatan medan putar stator atau kecepatan sinkron

$f$  : frekuensi (Hertz)

$p$  : jumlah kutub

2. Medan putar stator akan memotong batang konduktor pada rotor, sehingga pada kumparan rotor (medan) timbul tegangan induksi atau GGL sebesar:

$$E_2 = 4.44 f N_2 \Phi_m \quad (9-54)$$

5

Keterangan:  $E_2$  : Tegangan induksi pada rotor saat rotor dalam keadaan diam (Volt)

$N_2$  : Jumlah lilitan kumparan rotor

$\Phi_m$  : Fluksi maksimum (Wb)

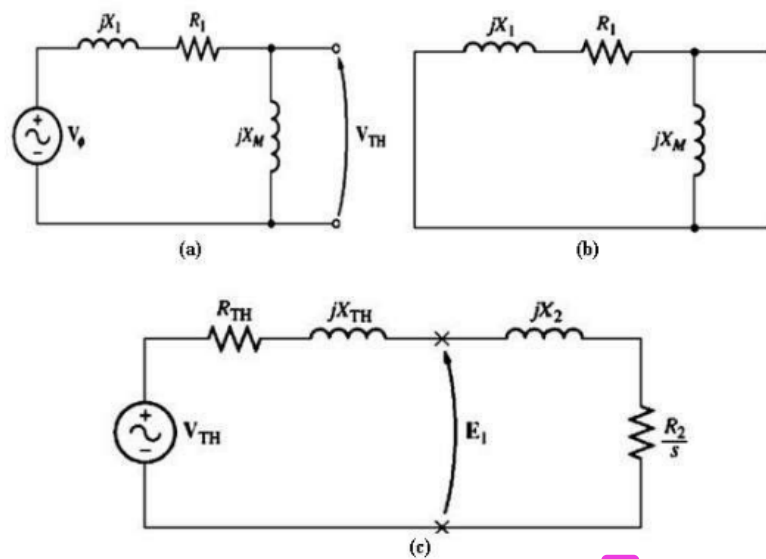
3. Kumparan rotor merupakan rangkaian tertutup, maka GGL akan menghasilkan arus.
4. Adanya arus di dalam medan magnet akan menimbulkan gaya  $F$  pada rotor.
5. Jika kopel mula yang dihasilkan oleh gaya  $F$  cukup besar untuk memikul kopel beban, rotor akan berputar searah medan putar stator.
6. Perputaran rotor akan semakin meningkat hingga mendekati kecepatan sinkron. Perbedaan kecepatan medan stator ( $n_s$ ) dan kecepatan rotor ( $n_r$ ) disebut slip.



7. Bila  $n_s = n_r$  tegangan tidak akan terinduksi dan arus tidak akan mengalir pada kumparan rotor, karenanya tidak dihasilkan kopel. Kopel ditimbulkan jika  $n_r < n_s$ .

## 9. Turunan Persamaan Torsi Motor Induksi dengan Teorema Thevenin

Mengacu pada rangkaian ekivalen per fasa motor induksi disederhanakan, didapat rangkaian tegangan dan impedansi thevenin yang ditunjukkan pada Gambar 2.15 berikut:



**Gambar 9.15** Rangkaian Ekivalen Thevenin Motor Induksi. (a) Rangkaian Ekivalen Tegangan Masuk Thevenin Motor Induksi; (b) Rangkaian Ekivalen Impedansi Masuk Thevenin Motor Induksi; (c) Rangkaian Ekivalen Sederhana Motor Induksi Tiga Fasa

Gambar 2.15 di atas menunjukkan sumber terbuka untuk mencari tegangan thevenin dari motor induksi dari pembagi tegangan, didapat:

$$V_{TH} = V_{\phi} \frac{X_M}{\sqrt{R_1^2 + (X_1 + X_M)^2}} \quad (9-55)$$

Persamaan yang didapat dari Gambar 2.15(a):

$$V_{TH} = \frac{jX_M}{R_1 + jX_1 + jX_M} V_{\phi} \quad (9-56)$$

$$V_{TH} = \frac{X_M}{\sqrt{R_1^2 + (X_1 + X_M)^2}} V_p \quad (9-57)$$

Persamaan yang didapat dari Gambar 2.15(b):

$$Z_{TH} = \frac{jX_M(R_1 + jX_1)}{R_1 + j(X_1 + X_M)} \quad (9-58)$$

Torsi induksi dari motor induksi tiga fasa:

$$T = \frac{3V_{TH}^2 R_2/s}{\omega_s [(R_{TH} + R_2/s)^2 + (X_{TH} + X_2)^2]} \quad (9-59)$$

$$I = \frac{V_{TH}}{[(R_{TH} + R_2/s) + j(X_{TH} + X_2)]} \quad (9-60)$$

$$\text{Slip maksimal} = \frac{R_2}{\sqrt{R_{th}^2 + (X_{th} + X_2)^2}} \quad (9-61)$$

Daya motor Induksi tiga fasa:

$$P_{in} = 3 \times V \times I \times \cos \varphi \quad (9-62)$$

$$P_{cu} = 3 \times (I^2) \times (R_2/slip) \quad (9-63)$$

$$P_{mek} = (1 - Slip) \times P_{cu} \quad (9-64)$$

71

## 10. Motor Sinkron

Motor sinkron adalah motor arus bolak-balik tiga fasa yang dijalankan pada kecepatan sinkron tanpa slip. Secara umum penggunaan motor sinkron difungsikan sebagai generator akan tetapi motor sinkron tetap digunakan oleh industri yang membutuhkan ketelitian putaran dan putaran konstan.

Sebuah motor sinkron selalu beroperasi pada kecepatan konstan, pada kondisi tidak berbeban, tetapi apabila motor diberi beban maka motor akan selalu akan berusaha untuk tetap pada putaran konstan dan motor akan melepaskan kondisi sinkronnya apabila beban yang ditanggung terlalu besar (torsi *pull-out*).

64

Motor sinkron memiliki kekurangan didalam melakukan start dengan sendirinya. Karena tidak memiliki torsi *start* awal, oleh karena itu motor sinkron memerlukan beberapa alat bantu untuk membantu di dalam *start* awal sehingga masuk didalam kondisi sinkron, pada sebuah induksi motor, rotor harus memiliki slip. Kecepatan rotor harus kurang atau terlambat dari perputaran fluks stator supaya arus diinduksikan ke rotor. Jika induksi rotor motor tersebut itu bertujuan untuk mencapai kecepatan sinkron, maka tidak ada garis gaya yang memotong

melalui rotor, sehingga tidak ada arus yang akan diinduksikan ke rotor dan tidak ada torsi yang akan dikembangkan.

Motor sinkron memiliki karakteristik sebagai berikut:

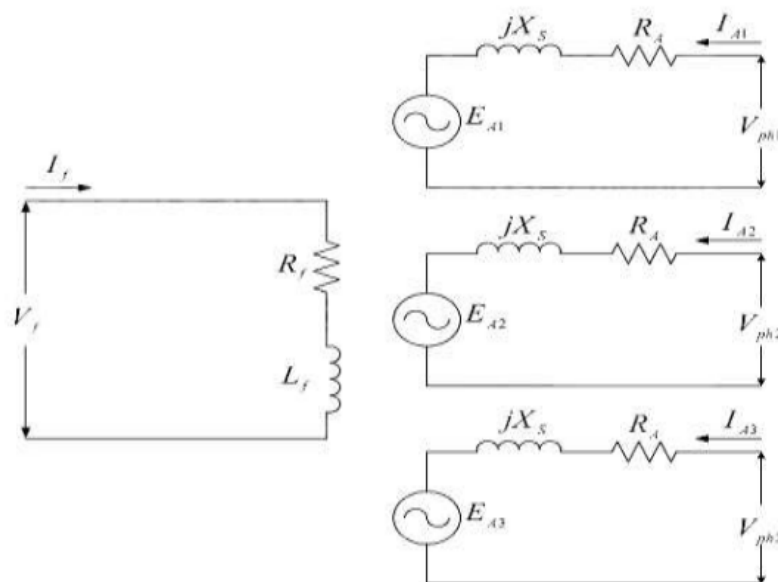
1. Sebuah stator tiga fasa sama dengan motor induksi, stator yang memiliki tegangan menengah sering digunakan.
2. Sebuah motor yang bersinggungan (bidang yang berputar) yang memiliki jumlah kutub yang sama sebagai statornya dan dipasang oleh sumber eksternal arus searah.

Tipe *brush* dan *brushless exciters* digunakan untuk memasok medan arus searah ke rotor. Arus pada rotor membentuk suatu hubungan kutub magnetic utara-selatan pada kutub-kutub rotor, yang memungkinkan rotor untuk mengunci fluks stator yang berputar.

3. Dimulai sebagai sebuah motor induksi rotor motor sinkron juga mempunyai sebuah *squirrel-cage winding* yang dikenal sebagai *amortisseur winding*, yang berfungsi menghasilkan torsi untuk menyalakan motor.
4. Motor sinkron akan dijalankan pada kecepatan sinkron sesuai dengan rumu:

$$\text{Synchronous rpm} = \frac{120 \times f}{p} \quad (9-62)$$

Rangkaian ekivalen motor sinkron 3 fasa ditunjukkan pada Gambar 9.16 dibawah ini.



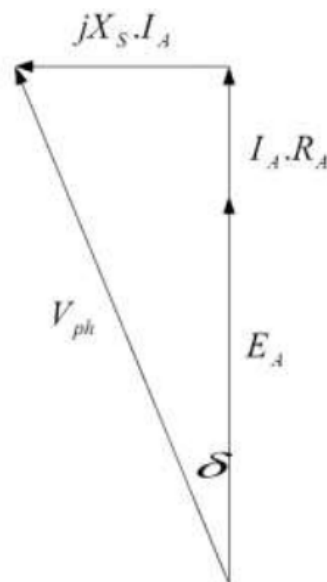
Gambar 9.16 Rangkaian Ekivalen Motor Sinkron Tiga Fasa

Persamaan hukum Kirchoof untuk tegangan dari motor sinkron adalah:

$$V_{ph} = E_A + j.X_s I_A + (I_A R_A) \quad (9-63)$$

64

Persamaan (9-61) di atas dapat digambarkan diagram fasor motor sinkron seperti ditunjukkan pada Gambar 9.17 berikut:



**Gambar 9.17** Diagram Fasor Motor Sinkron Dengan Faktor Daya Satu

64

Keterangan:  $E_A$  : tegangan jangkar (GGL lawan) (Volt)

$I_A$  : arus jangkar (Ampere)

$V_{ph}$  : tegangan terminal (Volt)

$X_s$  : reaktansi motor sinkron

$\delta$  : sudut torsi

Persamaan yang didapat dari diagram fasor motor sinkron adalah:

$$P_{mek} = 3E_A \cdot I_A \quad (9-64)$$

$$V_{ph} \cdot \sin \delta = X_s \cdot I_A \quad (9-65)$$

Maka diperoleh:

$$I_A = \frac{V_{ph} \cdot \sin \delta}{X_s} \quad (9-66)$$

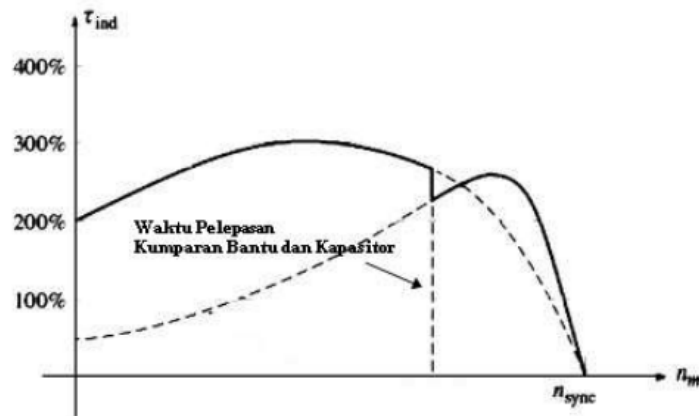
$$\text{Jadi: } P = \frac{3E_A \cdot V_{ph} \cdot \sin \delta}{X_s} \quad (9-67)$$

$$T_{imd} = \frac{P_{mek}}{\omega_s} \quad (9-68)$$

## 11. Karakteristik Motor AC

### a. Karakteristik Motor Kapasitor

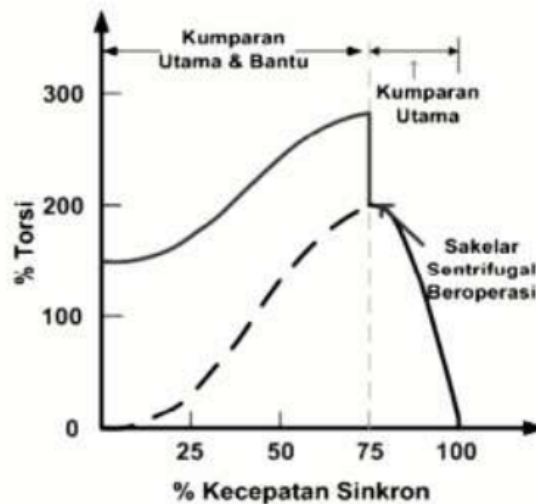
Motor kapasitor memiliki torsi awal yang cukup besar, untuk beberapa aplikasi penggunaan motor kapasitor kurang memuaskan walaupun motor tipe ini digunakan untuk mengatasi kelemahan pada motor fasa belah. Gambar 9.18 menunjukkan karakteristik dari motor kapasitor.



**Gambar 9.18** Karakteristik Motor Kapasitor.

### b. Karakteristik Motor Fasa Belah

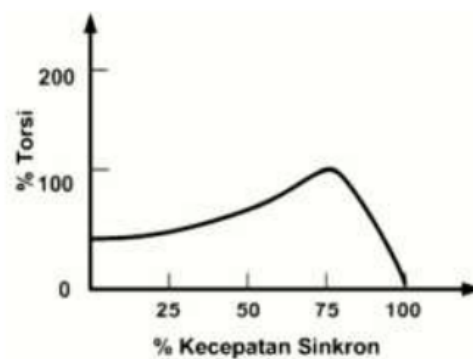
Karakteristik motor fasa belah dapat dilihat dari Gambar 9.19 di bawah ini, torsi awal dari motor fasa belah mencapai 150% dari torsi beban penuhnya, dan saklar sentrifugal dilepaskan pada 75% kecepatan normal pada motor.



**Gambar 9.19** Karakteristik Motor Fasa Belah

c. **Karakteristik Motor Kutub Bayangan**

Motor kutub bayangan biasanya digunakan pada alat-alat yang tidak membutuhkan torsi yang besar dan tidak membutuhkan banyak pemeliharaan. Kurva karakteristik dari motor kutub bayangan ditunjukkan pada gambar 9.20 berikut.



**Gambar 9.20** Karakteristik Motor Kutub Bayangan

d. **Karakteristik Motor Induksi 3 Fasa**

Motor induksi rotor sangkar tupai dibuat dalam 4 kelas berdasarkan *National Electrical Manufacturers Association* (NEMA).

53

### 1) Motor Kelas A

Motor kelas A mempunyai rangkaian resistansi rotor kecil, beroperasi pada slip sangat kecil yaitu kurang dari 0.01 dalam keadaan berbeban, motor kelas A digunakan untuk keperluan torsi *start* yang sangat kecil.

### 2) Motor Kelas B

Motor kelas B digunakan untuk keperluan umum mempunyai torsi *starting* normal dan arus *starting* normal, regulasi kecepatan putar pada saat *full load* rendah dibawah 5% dan torsi *starting* sekitar 150% dari *rated*, walaupun arus *starting* normal biasanya mempunyai besar 600% dari *full load*.

### 3) Motor Kelas C

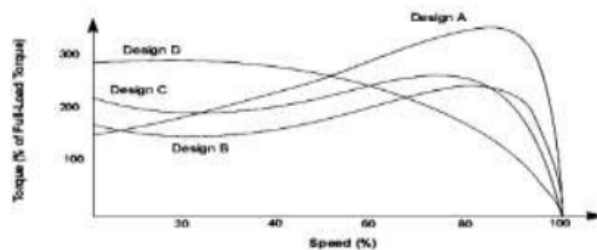
Motor kelas C mempunyai torsi *starting* yang lebih besar dibandingkan motor kelas B, arus *starting* normal, slip kurang dari 0.05 pada kondisi *full load*, torsi *starting* sekitar 200% dari *rated*, dan biasanya digunakan untuk konveyor, pompa, kompresor dan lain sebagainya.

53

### 4) Motor Kelas D

Motor kelas D mempunyai torsi *starting* yang besar dan arus *starting* relatif rendah, slip besar, pada slip beban penuh mempunyai efisiensi lebih rendah dibandingkan kelas motor lainnya dan torsi *starting* sekitar 300%.

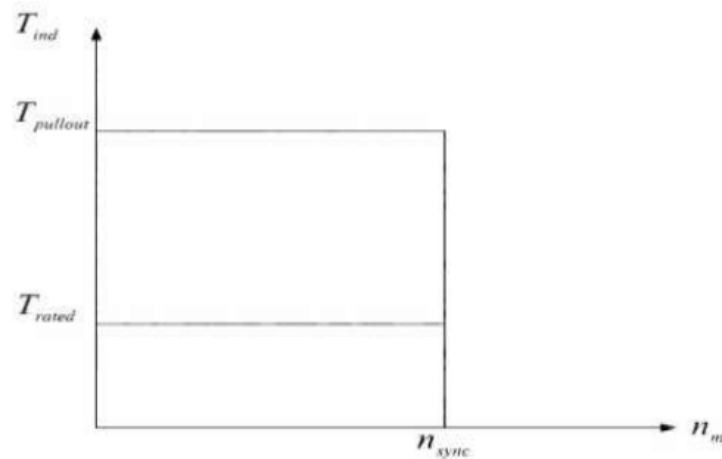
Karakteristik torsi dan kecepatan pada motor induksi rotor sangkar dari perbedaan standar NEMA ditunjukkan pada gambar 9.21 dibawah ini.



**Gambar 9.21** Karakteristik Torsi dan Kecepatan pada Motor Induksi

### e. Karakteristik Motor Sinkron

Karakteristik motor sinkron dapat dilihat pada Gambar 9.22, kecepatan putaran motor sinkron terkunci pada frekuensi elektrik yang disuplai, oleh karena itu kecepatan motor akan tetap konstan dari kondisi beban nol sampai torsi maksimum motor.



Gambar 9.22 Karakteristik Motor Sinkron

### PERTANYAAN DAN SOAL

---

1. Jelaskan perbedaan klasifikasi jenis motor listrik!!
2. Jelaskan karakteristik-karakteristik yang dimiliki oleh masing-masing motor listrik!!
3. Bagaimana karakteristik motor listrik dapat diproses komputasi menggunakan MATLAB



## BAB 10

### PERANCANGAN KOMPUTASI VISUAL TENAGA LISTRIK

---

20

#### TUJUAN INSTRUKSIONAL:

*Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:*

- 1) *Mengerti konsep perancangan komputasi visual menggunakan MATLAB*
  - 2) *Mengerti penggunaan masing-masing menu MATLAB untuk mendisain visualisasi*
  - 3) *Tenaga Listrik menggunakan MATLAB Mengimplementasikan beberapa parameter karakteristik motor DC untuk dibuatkan menu tampilan secara visual*
  - 2) *Mengimplementasikan beberapa parameter karakteristik motor AC dan Motor Sinkron untuk dibuatkan menu tampilan secara visual*
- 

Penyusunan perancangan akan melakukan serangkaian kegiatan untuk menyelesaikan proses komputasi ini. Metodologi yang digunakan untuk menyelesaikan proses komputasi ini diantaranya adalah:

1. Pembuatan rancangan dilakukan percobaan dengan membuat *software* (program) yang berhubungan dengan perancangan motor dan karakteristik yang akan diuji..
2. Pengujian terhadap rancangan *software* yang telah dibuat dan analisa hasil pengujian rancangan dilakukan dengan evaluasi serta membuat kesimpulan terhadap hasil pengujian perancangan.

Perancangan meliputi pembahasan dan pembuatan perangkat pengujian berupa simulasi *software* karakteristik-karakteristik motor listrik yang disesuaikan dengan pembahasan mata kuliah teknik tenaga listrik yang ada pada jurusan Elektro Fakultas Teknik Untirta. Perancangan motor dan karakteristik yang akan diuji yaitu motor DC dan AC 1 fasa dan 3 fasa, motor DC meliputi motor DC *shunt*, motor DC seri, dan motor DC kompon, perancangan ketiga motor tersebut akan menghasilkan karakteristik listrik (torsi terhadap arus jangkar), karakteristik kecepatan terhadap arus jangkar dan karakteristik mekanis (torsi terhadap kecepatan).

Motor AC satu fasa yang dirancang meliputi motor induksi satu fasa, motor fasa belah, motor kutub bayangan, dan motor kapasitor. Perancangan motor induksi

satu fasa akan menghasilkan karakteristik torsi terhadap slip, sementara pada perancangan motor fasa belah, motor kutub bayangan dan motor kapasitor akan menghasilkan dua karakteristik yaitu karakteristik torsi terhadap slip dan karakteristik torsi terhadap kecepatan. Perancangan pada setiap motor dilakukan dengan memasukkan persamaan-persamaan motor yang dibutuhkan untuk perhitungan karakteristik masing-masing motor dengan membuat parameter variabel sebagai bentuk dari laboratorium virtual sehingga dapat membuktikan teori.

## A. PERANCANGAN TAMPILAN AWAL

Pada sub bab ini dirancang tampilan awal dari laboratorium virtual menggunakan *software* GUI MATLAB. Tampilan awal ini dibuat di *blank* GUI (*default*) dengan nama file LabVirtual\_MML\_JTE\_UNTIRTA sebagai menu utama dari laboratorium virtual sebelum pengguna memilih simulasi motor untuk dilihat karakteristiknya. Tahap awal yang dilakukan untuk tampilan awal laboratorium virtual ini adalah membuat gambar *background* agar tampilan terlihat menarik, kemudian membuat beberapa *toolbar* sebagai pilihan menu dengan fungsi yang berbeda dan dapat dipilih oleh pengguna. Halaman tampilan menu utama laboratorium virtual dapat dilihat pada Gambar 10.1.



**Gambar 10.1** Tampilan Halaman Awal Laboratorium Virtual

Tampilan halaman awal laboratorium virtual mempunyai empat buah *toolbar* untuk memuat tampilan berikutnya yaitu:

1. *Toolbar* "Tentang" mempunyai satu buah sub *toolbar* yaitu sub *toolbar* "Tentang Laboratorium Virtual" yang berfungsi untuk memuat halaman tentang laboratorium virtual yang berisi penjelasan dari laboratorium virtual, dan tujuan

dari laboratorium virtual. Halaman tentang laboratorium virtual ditunjukkan pada Gambar 10.2.



**Gambar 10.2** Tampilan Halaman Tentang Laboratorium Virtual

2. *Toolbar "Motor DC"* mempunyai tiga buah sub *toolbar* yaitu Sub *toolbar "Motor DC Shunt"* dan "Motor DC Seri" yang berfungsi untuk memuat halaman motor dc shunt dan seri
3. *Toolbar "Motor AC"* mempunyai dua buah sub *toolbar* yaitu:
  - a. Sub *toolbar "Motor AC 1 Fasa"* yang mempunyai empat buah sub menu yang berisi motor induksi 1 fasa, motor fasa belah, motor kutub bayangan, dan motor kapasitor yang berfungsi untuk memuat halaman motor ac 1 fasa tersebut.
  - b. Sub *toolbar "Motor AC 3 Fasa"* yang mempunyai dua buah sub menu yang berisi motor induksi tiga fasa dan motor sinkron yang berfungsi untuk memuat halaman motor ac tiga fasa tersebut.
4. *Toolbar "Bantuan"* mempunyai dua buah sub *toolbar* yaitu:
  - a. Sub *toolbar "Teori Singkat Motor DC"* yang berfungsi menampilkan dasar teori singkat karakteristik dan persamaan umum dari motor DC. Sub *toolbar "Teori Singkat Motor AC 1 Fasa"* yang berfungsi menampilkan dasar teori singkat dan karakteristik dari motor induksi 1 fasa, motor fasa belah, motor kapasitor dan motor kutub bayangan.
  - b. Sub *toolbar "Teori Singkat Motor AC 3 Fasa"* yang berfungsi menampilkan dasar teori singkat dan karakteristik dari motor induksi 3 fasa dan motor sinkron.

- c. Sub *toolbar* "Petunjuk Penggunaan" berfungsi menampilkan petunjuk penggunaan laboratorium virtual sebagai tata cara pengujian karakteristik motor pada laboratorium virtual.
5. *Toolbar* "Keluar" berfungsi untuk menutup program atau keluar dari program. Tampilan ketika <sup>34</sup> mengeksekusi *toolbar* keluar dibuat menggunakan *modal question dialog* seperti pada Gambar 3.3 di bawah ini.



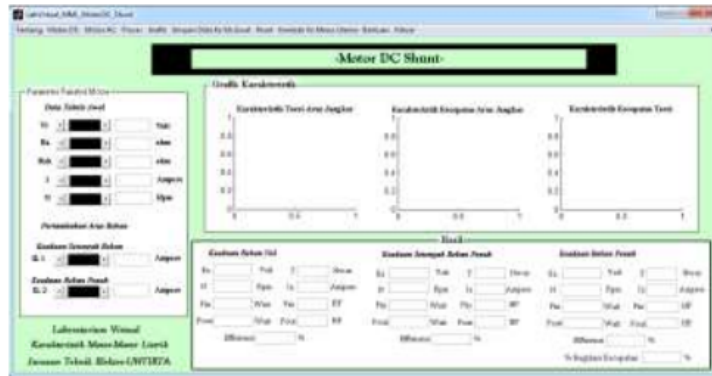
**Gambar 10.3** Tampilan Kotak Dialog Keluar

## B. PERANCANGAN KARAKTERISTIK MOTOR DC

<sup>94</sup> Pada sub bab ini dirancang laboratorium virtual untuk mendapatkan <sup>94</sup> karakteristik motor DC yang meliputi karakteristik torsi terhadap arus jangkar, karakteristik kecepatan terhadap arus jangkar dan karakteristik kecepatan terhadap torsi pada saat beban nol, setengah beban <sup>51</sup> dan keadaan beban penuh. Motor DC pada perancangan ini meliputi motor DC shunt, motor DC seri, motor DC kompon panjang dan motor DC kompon pendek. Parameter perancangan motor DC pada menggunakan pendek <sup>176</sup> dari parameter motor yang terdapat pada laboratorium mesin mesin listrik Jurusan Teknik Elektro Universitas Sultan Ageng Tirtayasa.

### 1. Perancangan Karakteristik Motor DC Shunt

<sup>156</sup> Perancangan karakteristik motor DC shunt dibuat untuk mendapatkan karakteristik kelakuan motor DC shunt saat arus beban motor tersebut bertambah. Tampilan karakteristik motor DC shunt ditunjukkan pada Gambar 10.4.



**Gambar 10.4** Tampilan Halaman Karakteristik Motor DC Shunt

Tampilan halaman jenis motor DC shunt memuat beberapa *toolbar*, parameter data teknis, *slider*, *static text*, *edit text*, panel dan *axes*, yaitu:

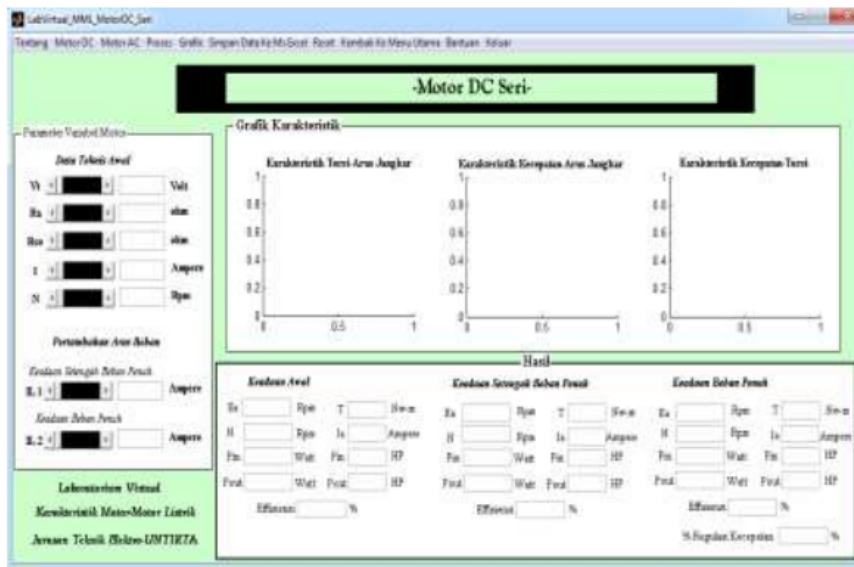
1) *Toolbar*

- a) *Toolbar* "Tentang" berfungsi untuk menampilkan halaman yang menjelaskan tentang laboratorium virtual, petunjuk penggunaan, dasar teori serta rumus singkat dari motor dan mempunyai sub menu yaitu "Tentang Laboratorium Virtual", "Tentang Motor DC", dan "Tentang Motor AC".
- b) *Toolbar* "Motor DC" berfungsi untuk menampilkan halaman karakteristik motor dc dan mempunyai sub menu yaitu motor DC shunt, motor DC seri, dan motor DC kompon yang mencakup kompon panjang dan kompon pendek.
- c) *Toolbar* "Motor AC" berfungsi untuk menampilkan halaman karakteristik motor AC dan mempunyai sub menu yaitu motor AC 1 fasa dan motor ac 3 fasa.
- d) *Toolbar* "Proses" berfungsi untuk mengeksekusi perhitungan dan menampilkan hasilnya pada panel "Hasil".
- e) *Toolbar* "Grafik" mempunyai dua buah sub *toolbar* yang berfungsi untuk merubah hasil perhitungan menjadi grafik dan menampilkannya di panel "Grafik Karakteristik" dan juga *figure*.
- f) *Toolbar* "Simpan Data Ke Ms.Excel" berfungsi untuk menyimpan data parameter dan hasil yang telah diuji coba ke dalam ms.excel.
- g) *Toolbar* "Reset" berfungsi untuk mengosongkan kembali semua parameter dan hasil pada halaman karakteristik menjadi kembali pada tampilan awal halaman karakteristik.
- h) *Toolbar* "Kembali Ke Menu Utama" berfungsi untuk menampilkan halaman menu utama dari laboratorium virtual mesin-mesin listrik jurusan teknik elektro Untirta.
- i) *Toolbar* "Bantuan" berfungsi untuk menampilkan halaman teori singkat motor DC, teori singkat motor AC 1 fasa, teori singkat motor AC 3 fasa, dan petunjuk penggunaan laboratorium virtual.

- j) *Toolbar "Keluar"* berfungsi untuk menutup program atau keluar dari program.
- 2) Parameter Data Teknis
- Vt berupa *slider* yang berfungsi memasukkan *input* tegangan berupa *range* angka dengan *min* = 100 dan *max* = 125, untuk mendapatkan nilai tegangan pertama Vt yang diinginkan dengan cara menggeser *control* pada *slider*.
  - Ra berupa *slider* yang berfungsi memasukkan *input* resistansi jangkar berupa *range* angka dengan *min* = 0.1 dan *max* = 0.5, dan untuk mendapatkan nilai resistansi jangkar yang diinginkan dengan cara menggeser *control* pada *slider*.
  - Rsh berupa *slider* yang berfungsi memasukkan *input* tahanan shunt berupa *range* angka dengan *min* = 100 dan *max* = 140, dan untuk mendapatkan nilai resistansi *shunt* yang diinginkan dengan cara menggeser *control* pada *slider*.
  - I berupa *slider* yang berfungsi memasukkan *input* arus berupa *range* angka dengan *min* = 3.5 dan *max* = 5, dan untuk mendapatkan nilai arus yang diinginkan dengan cara menggeser *control* pada *slider*.
  - N berupa *slider* yang berfungsi memasukkan *input* kecepatan berupa *range* angka dengan *min* = 1100 dan *max* = 1800, dan untuk mendapatkan nilai kecepatan yang diinginkan dengan cara menggeser *control* pada *slider*.
  - IL 1 berupa *slider* yang berfungsi memasukkan *input* perubahan arus beban yang pertama berupa *range* angka dengan *min* = 10 dan *max* = 20, dan untuk mendapatkan nilai arus beban yang diinginkan dengan cara menggeser *control* pada *slider*.
  - IL 2 berupa *slider* yang berfungsi memasukkan *input* perubahan arus beban yang pertama berupa *range* angka dengan *min* = 20 dan *max* = 40, dan untuk mendapatkan nilai arus beban yang diinginkan dengan cara menggeser *control* pada *slider*.
- 3) Panel "*Grafik Karakteristik*" berfungsi untuk menampilkan grafik hasil dari perhitungan untuk dilihat karakteristiknya, panel ini akan menampilkan tiga grafik yaitu grafik karakteristik torsi terhadap arus jangkar, grafik karakteristik kecepatan terhadap arus jangkar dan grafik kecepatan terhadap torsi.
- 4) Panel "*Hasil*" berfungsi untuk menampilkan hasil perhitungan dari data parameter yang telah diisi dari keadaan beban nol, keadaan setengah beban penuh, dan keadaan beban penuh dari motor. Berupa *edit text* yang akan menampilkan hasil dari tegangan jangkar, torsi, kecepatan, arus jangkar, daya *input* dalam watt, daya *output* dalam watt, daya *input* dalam HP, daya *output* dalam HP, *effisiensi* dan regulasi kecepatan.

## 2. Perancangan Karakteristik Motor DC Seri

Perancangan karakteristik motor DC seri dibuat untuk menemukan karakteristik perubahan kecepatan, torsi dan arus jangkar saat arus beban bertambah. Tampilan karakteristik motor DC seri ditunjukkan pada Gambar 10.5.



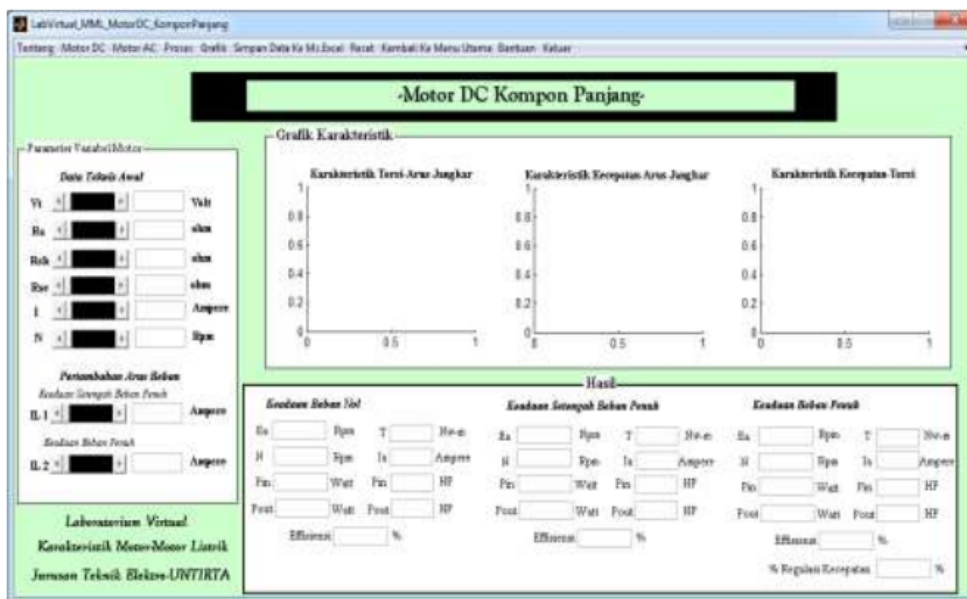
**Gambar 10.5** Tampilan Halaman Karakteristik Motor DC Seri

Tampilan halaman jenis motor DC seri hampir sama dengan motor dc shunt yang memuat beberapa *toolbar*, parameter data teknis, *slider*, *static text*, *edit text*, panel dan *axes*, hanya pada motor DC seri parameter data teknis Rsh diganti menjadi Rse karena nilai resistansi seri yang dibutuhkan untuk perhitungan karakteristik motor tersebut. *Range* nilai resistansi seri pada *slider* motor DC seri adalah  $min = 0.1$  dan  $max = 0.15$ .

## 3. Perancangan Karakteristik Motor DC Kompon

Perancangan karakteristik motor DC kompon panjang dan kompon pendek dibuat untuk mendapatkan karakteristik perubahan kecepatan, arus jangkar dan torsi saat arus beban bertambah. Tampilan halaman jenis motor DC kompon panjang dan pendek adalah sama. Tampilan halaman jenis motor DC kompon memuat beberapa *toolbar*, parameter data teknis, *slider*, *static text*, *edit text*, panel dan *axes* yang sama seperti motor-motor yang telah dijelaskan sebelumnya, hanya pada kedua motor ini parameter data teknis awal ditambahkan Rsh dan Rse sebagai tahanan *shunt* dan seri karena keduanya diperlukan pada perhitungan. *Range* nilai Rsh pada *slider* motor DC kompon panjang adalah  $min = 100$  dan  $max = 140$ ,

sedangkan untuk resistansi seri adalah  $min = 0.1$  dan  $max = 0.15$ . Tampilan karakteristik motor DC kompon panjang ditunjukkan pada Gambar 10.6.



**Gambar 10.6** Tampilan Halaman Karakteristik Motor DC Kompon Panjang

### C. PERANCANGAN KARAKTERISTIK MOTOR AC 1 FASA

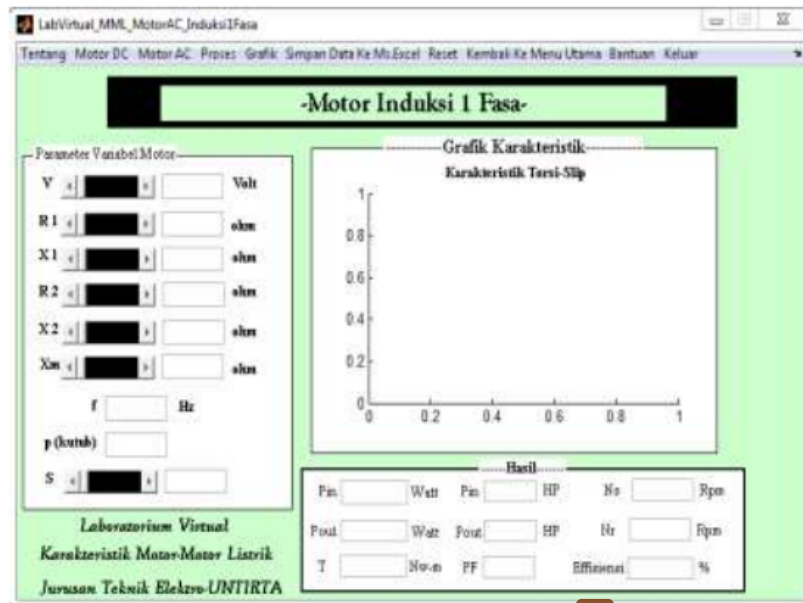
Pada sub bab ini dirancang laboratorium virtual untuk mendapatkan karakteristik yang dimiliki oleh motor AC 1 fasa yaitu karakteristik torsi dan slip serta karakteristik torsi dan kecepatan. Motor AC 1 fasa pada perancangan ini meliputi motor induksi 1 fasa, motor fasa belah, motor kapasitor dan motor kutub bayangan. Perancangan laboratorium virtual untuk mengetahui karakteristik motor DC ini menggunakan pendekatan dari parameter motor AC yang ada pada *datasheet* motor dan pada jurnal-jurnal hasil penelitian yang berkaitan dengan motor 1 fasa.

#### 1. Perancangan Karakteristik Motor Induksi 1 Fasa

Merujuk pada Gambar 2.10(a) dan (b) diketahui bahwa motor induksi fasa tunggal dipengaruhi oleh medan magnet maju dan mundur sehingga motor induksi 1 fasa tidak mempunyai kopel mula atau motor tidak dapat berputar jika tidak diberikan suatu gerakan oleh alat bantu. Perancangan karakteristik motor induksi 1 fasa pada laboratorium virtual dirancang untuk mendapatkan karakteristik motor sebelum motor diberikan alat bantu yang dimaksudkan agar pengguna mengetahui



karakteristik awal dari motor induksi 1 fasa. Tampilan karakteristik motor induksi 1 fasa ditunjukkan pada Gambar 10.7.



**Gambar 10.7** Tampilan Halaman karakteristik Motor Induksi 1 Fasa

Tampilan halaman jenis motor induksi 1 fasa memuat beberapa *toolbar*, parameter data teknis, *slider*, *static text*, *edit text*, panel dan *axes*, yaitu:

1) *Toolbar*

*Toolbar* yang dibuat pada halaman karakteristik motor induksi 1 fasa sama seperti *toolbar* yang dibuat pada karakteristik motor-motor sebelumnya dan telah dijelaskan sebelumnya pada perancangan karakteristik motor DC shunt.

2) Parameter Data Teknis

- a) V berupa *slider* yang berfungsi memasukkan *input* tegangan berupa *range* angka dengan  $min = 100$  dan  $max = 220$ , untuk mendapatkan nilai tegangan pertama  $V_t$  yang diinginkan dengan cara menggeser *control* pada *slider*.
- b) R1 berupa *slider* yang berfungsi memasukkan *input* resistansi berupa *range* angka dengan  $min = 10$  dan  $max = 50$ , dan untuk mendapatkan nilai resistansi yang diinginkan dengan cara menggeser *control* pada *slider*.
- c) X1 berupa *slider* yang berfungsi memasukkan *input* reaktansi berupa *range* angka dengan  $min = 20$  dan  $max = 40$ , dan untuk mendapatkan nilai reaktansi yang diinginkan dengan cara menggeser *control* pada *slider*.
- d) R2 berupa *slider* yang berfungsi memasukkan *input* resistansi berupa *range* angka dengan  $min = 10$  dan  $max = 25$ , dan untuk mendapatkan nilai resistansi yang diinginkan dengan cara menggeser *control* pada *slider*.

- e)  $X_2$  berupa *slider* yang berfungsi memasukkan *input* reaktansi berupa *range* angka dengan  $min = 20$  dan  $max = 60$ , dan untuk mendapatkan nilai reaktansi yang diinginkan dengan cara menggeser *control* pada *slider*.
  - f)  $X_m$  berupa *slider* yang berfungsi memasukkan *input* reaktansi magnetisasi berupa *range* angka dengan  $min = 300$  dan  $max = 500$ , dan untuk mendapatkan nilai reaktansi magnetisasi yang diinginkan dengan cara menggeser *control* pada *slider*.
  - g)  $F$  berupa *edit text* yang berfungsi memasukkan *input* frekuensi yang dapat dituliskan langsung oleh pengguna.
  - h)  $P$  (kutub) berupa *edit text* yang berfungsi memasukkan *input* kutub motor yang dapat dituliskan langsung oleh pengguna.
  - i)  $S$  berupa *slider* yang berfungsi memasukkan *input* slip berupa *range* angka dengan  $min = 0.01$  dan  $max = 0.05$ , dan untuk mendapatkan nilai slip yang diinginkan dengan cara menggeser *control* pada *slider*.
- 3) Panel "Grafik Karakteristik" berfungsi untuk menampilkan grafik hasil dari perhitungan untuk dilihat karakteristiknya, panel ini akan menampilkan satu buah grafik yaitu grafik karakteristik torsi terhadap kecepatan.
  - 4) Panel "Hasil" berfungsi untuk menampilkan hasil perhitungan dari data parameter yang telah diisi dan menampilkan daya *input* motor dalam watt, daya *input* motor dalam HP, daya *output* motor dalam watt, daya *output* motor dalam HP, torsi, kecepatan stator, kecepatan rotor, faktor daya dan *effisiensi*.

## 2. Perancangan Karakteristik Motor Fasa Belah

Perancangan karakteristik motor fasa belah dibuat untuk mendapatkan karakteristik torsi terhadap slip dan torsi terhadap kecepatan dari motor induksi 1 fasa yang menggunakan dua buah kumparan stator (kumparan utama dan kumparan bantu). Resistansi dan reaktansi parameter *variable* motor yang dapat disesuaikan oleh pengguna pada motor fasa belah adalah resistansi dan reaktansi pada kumparan bantu, sementara resistansi dan reaktansi pada kumparan utama telah dimasukkan kedalam perhitungan dalam program. Hal ini dimaksudkan agar pada saat perhitungan resistansi dan reaktansi kumparan bantu tidak berubah. Parameter resistansi dan reaktansi pada kumparan utama dapat dilihat pada Tabel 10.2.

Gambar 10.8 menunjukkan tampilan halaman karakteristik motor fasa belah. Tampilan halaman jenis motor induksi 1 fasa memuat beberapa *toolbar*, parameter data teknis kumparan bantu, *slider*, *static text*, *edit text*, panel dan *axes*, yaitu:

- 1) *Toolbar*  
*Toolbar* yang dibuat pada halaman karakteristik motor fasa belah sama seperti *toolbar* yang dibuat pada karakteristik motor-motor sebelumnya.
- 2) Parameter Data Teknis

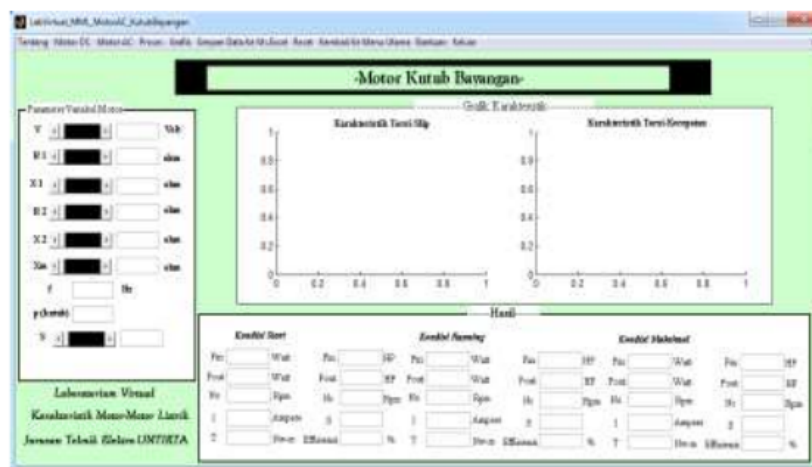
- a) V berupa *slider* yang berfungsi memasukkan *input* tegangan berupa *range* angka dengan *min* = 100 dan *max* = 220, untuk mendapatkan nilai tegangan pertama  $V_t$  yang diinginkan dengan cara menggeser *control* pada *slider*.
  - b) R1 berupa *slider* yang berfungsi memasukkan *input* resistansi berupa *range* angka dengan *min* = 10 dan *max* = 50, dan untuk mendapatkan nilai resistansi yang diinginkan dengan cara menggeser *control* pada *slider*.
  - c) X1 berupa *slider* yang berfungsi memasukkan *input* reaktansi berupa *range* angka dengan *min* = 20 dan *max* = 40, dan untuk mendapatkan nilai reaktansi yang diinginkan dengan cara menggeser *control* pada *slider*.
  - d) R2 berupa *slider* yang berfungsi memasukkan *input* resistansi berupa *range* angka dengan *min* = 10 dan *max* = 25, dan untuk mendapatkan nilai resistansi yang diinginkan dengan cara menggeser *control* pada *slider*.
  - e) X2 berupa *slider* yang berfungsi memasukkan *input* reaktansi berupa *range* angka dengan *min* = 20 dan *max* = 60, dan untuk mendapatkan nilai reaktansi yang diinginkan dengan cara menggeser *control* pada *slider*.
  - f)  $X_m$  berupa *slider* yang berfungsi memasukkan *input* reaktansi magnetisasi berupa *range* angka dengan *min* = 300 dan *max* = 500, dan untuk mendapatkan nilai reaktansi magnetisasi yang diinginkan dengan cara menggeser *control* pada *slider*.
  - g) F berupa *edit text* yang berfungsi memasukkan *input* frekuensi yang dapat dituliskan langsung oleh pengguna.
  - h) P (kutub) berupa *edit text* yang berfungsi memasukkan *input* kutub motor yang dapat dituliskan langsung oleh pengguna.
  - i) S berupa *slider* yang berfungsi memasukkan *input* slip berupa *range* angka dengan *min* = 0.01 dan *max* = 0.05, dan untuk mendapatkan nilai slip yang diinginkan dengan cara menggeser *control* pada *slider*.
- 3) Panel "Grafik Karakteristik" berfungsi untuk menampilkan grafik hasil dari perhitungan untuk dilihat karakteristiknya, panel ini akan menampilkan dua buah grafik yaitu grafik karakteristik torsi terhadap kecepatan dan torsi terhadap slip.
  - 4) Panel "Hasil" berfungsi menampilkan hasil perhitungan dari data parameter yang telah diisi dari kondisi *start*, kondisi *running*, dan kondisi beban maksimal dari motor. Berupa *edit text* yang akan menampilkan hasil dari nilai kapasitansi, daya *input* dalam watt, daya *output* dalam watt, daya *input* dalam HP, daya *output* dalam HP, kecepatan stator, kecepatan rotor, arus motor, slip, torsi dan efisiensi.



Gambar 10.8 Tampilan Halaman Karakteristik Motor Fasa Belah

### 3. Perancangan Karakteristik Motor Kutub Bayangan

Perancangan karakteristik motor kutub bayangan dibuat untuk mendapatkan karakteristik torsi terhadap slip dan torsi terhadap kecepatan dari motor yang mempunyai kumparan tambahan pada kutub-kutubnya. Tampilan karakteristik motor kutub bayangan ditunjukkan pada Gambar 10.9.

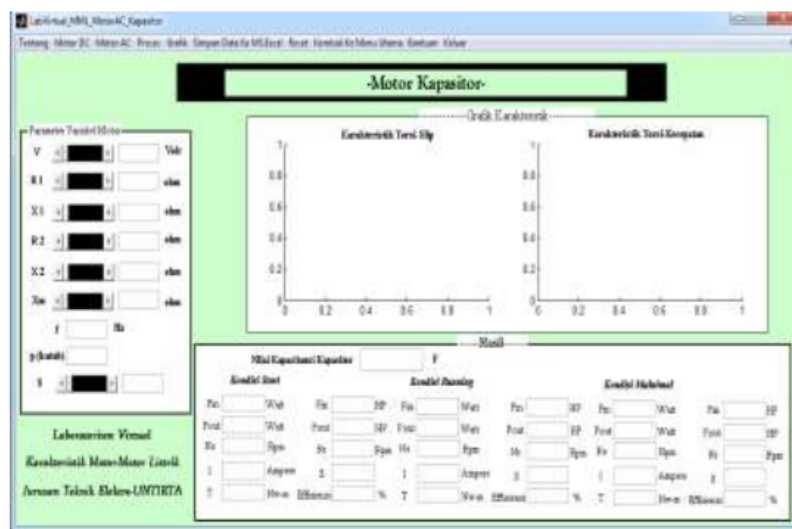


Gambar 10.9 Tampilan Halaman Karakteristik Motor Kutub Bayangan

Tampilan halaman karakteristik jenis motor kutub bayangan hampir sama dengan motor fasa belah yang memuat beberapa *toolbar*, parameter data teknis kumparan bantu pada motor fasa belah, *slider*, *static text*, *edit text*, panel dan *axes*.

#### 4. Perancangan Karakteristik Motor Kapasitor

Perancangan karakteristik motor kapasitor dibuat untuk mendapatkan karakteristik torsi terhadap slip dan torsi terhadap kecepatan dari motor yang mempunyai sebuah kapasitor yang dipasang seri terhadap kumparan bantu. Motor kapasitor hampir sama dengan motor fasa belah yang menggunakan dua buah kumparan stator (kumparan utama dan kumparan bantu) dan pada perancangan ini resistansi dan reaktansi kumparan bantu dibuat sama seperti pada motor fasa belah serta tampilan halaman karakteristik jenis kapasitor yang memuat beberapa *toolbar*, parameter data teknis, *slider*, *static text*, *edit text*, panel dan *axes* adalah sama, namun pada panel hasil ditambahkan sebuah *edit text* untuk menampilkan hasil dari nilai kapasitansi.



Gambar 10.10 Tampilan Halaman Karakteristik Motor Kapasitor

#### D. PERANCANGAN KARAKTERISTIK MOTOR INDUKSI 3 FASA

Sub bab ini menjelaskan perancangan laboratorium virtual motor induksi 3 fasa untuk mendapatkan karakteristik motor yaitu karakteristik torsi terhadap slip dan torsi terhadap kecepatan. Perancangan parameter resistansi dan reaktansi karakteristik motor induksi 3 fasa dibuat *range* nilai tertentu agar dapat dilihat perbandingan karakteristiknya sesuai dengan standarisasi NEMA (*National Electrical Manufactures Association*). *Range* nilai parameter resistansi dan reaktansi ditunjukkan Tabel 10.3.

**Tabel 10.1** Parameter Resistansi dan Reaktansi Motor Induksi 3 Fasa

Parameter	Range Nilai	
	<i>min</i>	<i>max</i>
Resistansi 1 (R1)	0.1 ohm	1.5 ohm
Reaktansi 1 (X1)	1.0 ohm	1.5 ohm
Resistansi 2 (R2)	0.1 ohm	1.5 ohm
Reaktansi 2 (X2)	1.0 ohm	1.5 ohm
Reaktansi Magnetisasi (Xm)	10 ohm	20 ohm

Tampilan halaman jenis motor induksi 3 fasa hampir sama dengan motor-motor yang telah dijelaskan sebelumnya, kecuali nilai tegangan dengan *range* nilai *min* = 100 dan *max* = 380, serta nilai resistansi dan reaktansi yang berbeda yang telah ditunjukkan pada Tabel 3.3. Tampilan halaman karakteristik motor induksi 3 fasa ditunjukkan pada Gambar 3. 11.



**Gambar 10.11** Tampilan Halaman Karakteristik Motor Induksi 3 Fasa

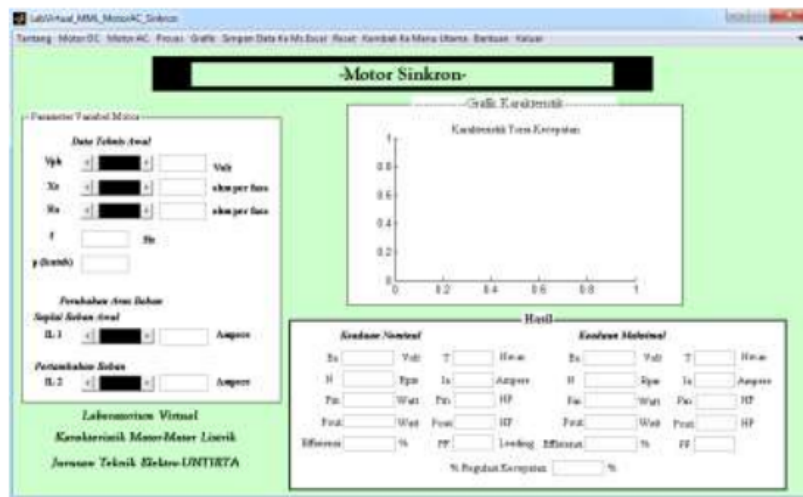
## E. PERANCANGAN KARAKTERISTIK MOTOR SINKRON

Perancangan motor sinkron dilakukan untuk mendapatkan karakteristik torsi terhadap kecepatan dengan pengaruh penambahan arus beban, tidak sama dengan perancangan sebelumnya perancangan motor ini hanya mengamati satu buah grafik, hal ini dikarenakan motor sinkron tidak mempunyai slip.

Perancangan parameter motor sinkron dibuat menggunakan *slider* dan *edit text* agar pengguna dapat langsung memasukkan nilai parameter yang diinginkan, parameter data teknis awal motor meliputi tegangan fasa (Vph) dengan *range* nilai *min* = 220 Volt dan *max* = 380 Volt, frekuensi (f), reaktansi sinkron per fasa (Xs)

dengan *range* nilai 1.5 hingga 5 ohm per fasa, resistansi jangkar ( $R_a$ ) dengan *range* nilai 0.5 hingga 1 ohm per fasa serta kutub motor ( $p$ ).

Parameter tambahan dalam perancangan ini yaitu berupa PF (*Power Factor*) atau  $\cos \phi$  untuk data teknis awal yaitu sebesar 0.8 *leading* dan motor sinkron terhubung delta, hal ini dimaksudkan untuk mempermudah perhitungan awal pada motor sinkron. Gambar 3.12 menunjukkan halaman tampilan karakteristik motor sinkron.



**Gambar 10.12** Tampilan Halaman Karakteristik Motor Sinkron

## F. DIAGRAM ALIR SISTEM

Diagram alir sistem perancangan laboratorium virtual mesin-mesin listrik dengan menggunakan GUI MATLAB dimulai dengan merancang tampilan awal berupa menu utama laboratorium virtual beserta pelengkap nya seperti tentang laboratorium virtual yang berisi penjelasan singkat tentang laborarotorium virtual, tujuan dan petunjuk penggunaan laboratorium, kemudian tahap berikutnya merancang laboratorium virtual jenis motor DC dan motor AC, dan tahap terakhir membuat tampilan bantuan pada setiap motor tampilan masing-masing motor yang berisi petunjuk pengujian karakteristik, dasar teori dan karakteristik singkat tentang motor. Setelah perancangan semua laboratorium virtual selesai dikerjakan maka pertama-tama melakukan percobaan pada setiap motor yang dirancang agar menampilkan grafik karakteristik motor dan hasil perhitungan motor tersebut sesuai dengan rumus yang digunakan pada setiap motor dalam perancangan program. Tata cara percobaan atau pengujian program pada laboratorium virtual dari pada salah satu motor adalah sebagai berikut:

1. Buka program MATLAB pastikan *current directory* adalah tempat menyimpan program GUI, kemudian pada *command window* ketikkan perintah

"LabVirtual\_MML\_JTE\_UNTIRTA" dan akan keluar tampilan menu utama laboratorium virtual dengan beberapa menu didalamnya.

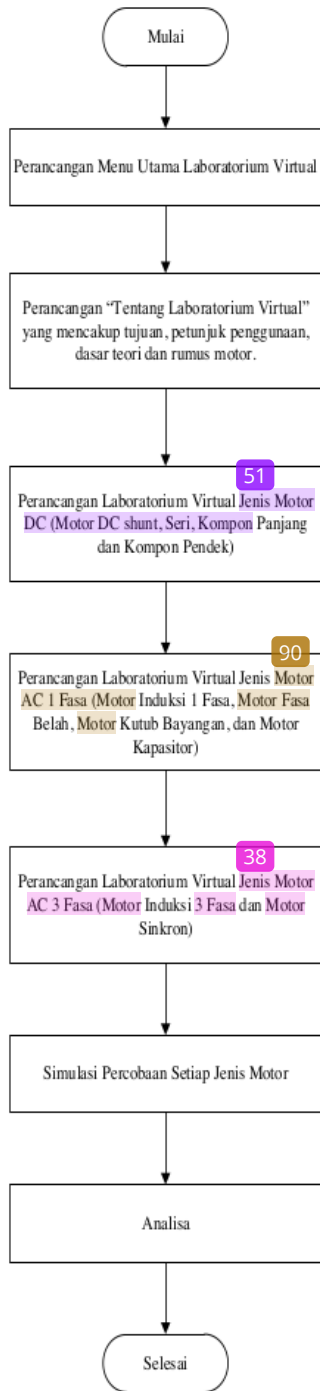
2. Pilih salah satu motor yang akan diuji pada *toolbar* motor AC atau motor DC, sebagai contoh untuk memilih motor DC shunt, pada *toolbar* motor DC tekan pada motor DC shunt kemudian akan muncul halaman tampilan motor DC shunt.
3. Proses percobaan dimulai dengan memasukkan parameter data teknis awal motor, baik menggunakan *slider* (menggeser *range* nilai yang telah ditentukan) atau dengan memasukkan nilai secara manual sesuai keinginan pengguna, semua parameter harus diisi penuh agar program dapat dieksekusi. Seperti pada tampilan motor DC shunt isi parameter  $V_t$ ,  $R_a$ ,  $R_{sh}$ ,  $I$ ,  $N$  dan parameter penambahan arus beban pada keadaan setengah beban penuh IL 1 dan IL 2 pada beban penuh.

Tahap-tahap pengujian pada masing-masing motor dijelaskan pada *toolbar* bantuan pada tampilan laboratorium virtual.

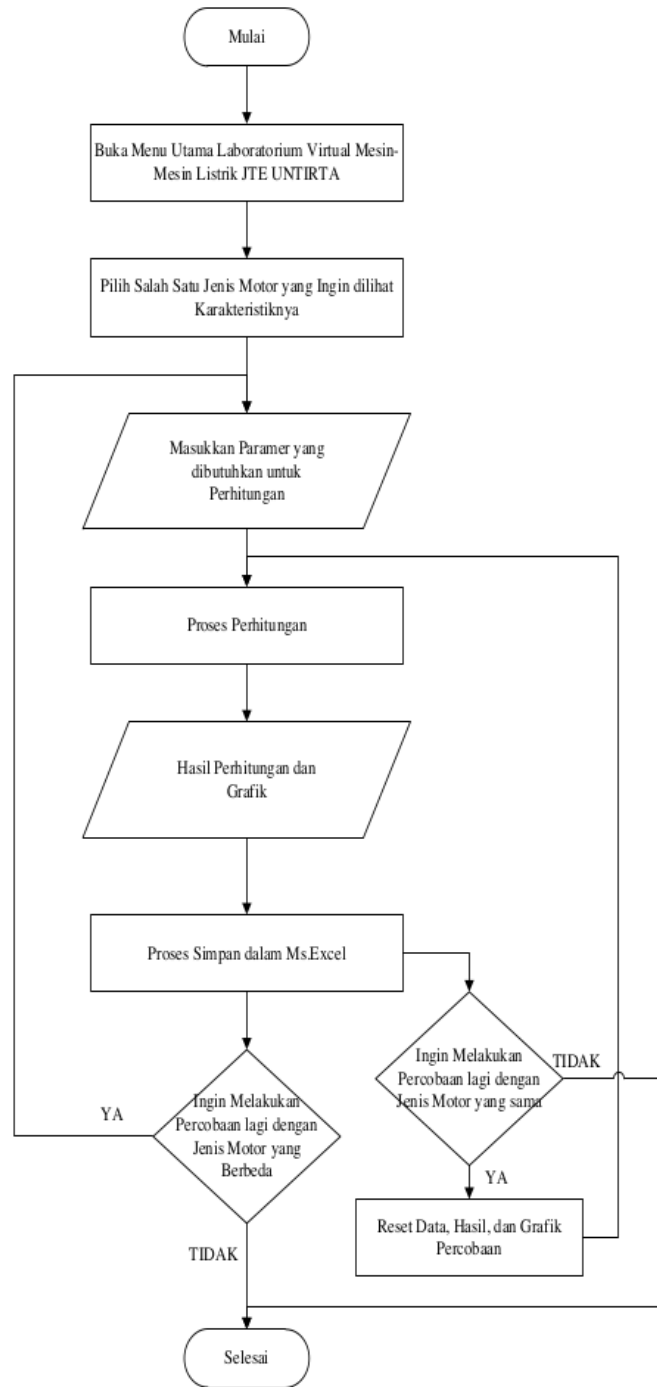
4. Setelah semua parameter telah diisi maka pengguna dapat melihat hasil perhitungan pengujian dengan menekan *toolbar* proses, dan untuk melihat grafik atau kurva karakteristik motor tekan *toolbar* tampilkan grafik. Grafik dapat dilihat pada tampilan laboratorium virtual motor dengan menekan sub *toolbar* pada tampilkan grafik atau untuk lebih jelas melihat keluaran grafik tekan sub *toolbar* tampilkan grafik dalam *figure*.
5. Setelah selesai pengujian data hasil percobaan dapat disimpan dalam Ms.Excel dengan menekan *toolbar* simpan data ke Ms. Excel dan untuk membuka data hasil percobaan dapat membuka pada folder penyimpanan program dengan nama *file* untuk motor DC shunt adalah lab virtual percobaan motor dc shunt, agar pengguna dapat menyimpan sendiri atau menyimpan ulang hasil pengujian dengan cara *save as file* pada folder lain yang dipilih untuk hasil pengujian.
6. Grafik hasil pengujian dapat disimpan dengan cara menampilkannya terlebih dahulu pada *figure* lalu simpan file pada folder lain seperti pada simpan data ke Ms.Excel.
7. Jika ingin melakukan pengujian pada motor yang sama, dapat menekan *toolbar* *reset* untuk mengosongkan kembali semua parameter, hasil dan grafik, tetapi jika ingin melakukan pengujian motor lain dapat langsung memilih motor yang ingin diuji dengan memilih pada *toolbar*.
8. Untuk mengakhiri program atau menutup program pengguna dapat menekan *toolbar* keluar.

Diagram alir sistem perancangan dan pengujian laboratorium virtual ditunjukkan pada Gambar 10.13.





(a)



(b)

**Gambar 103.15** Diagram Alir Sistem, (a) Diagram Alir Perancangan, (b) Diagram Alir Pengujian

## **PERTANYAAN DAN SOAL**

---

1. Jelaskan hal-hal yang harus anda pahami dan mengerti terkait dengan proses perancangan komputasi visual menggunakan pemrograman MATLAB!!
2. Bagaimana merancang komputasi visual sistem tenaga listrik menggunakan MATLAB??
3. Bagaimana mengimplementasikan proses komputasi visual karakteristik motor DC, AC, dan motor Sinkron menggunakan menu GUI MATLAB??

# BAB 11

## KOMPUTASI VISUAL TENAGA LISTRIK

---

20

### TUJUAN INSTRUKSIONAL:

*Setelah mempelajari bab ini, para mahasiswa diharapkan mampu:*

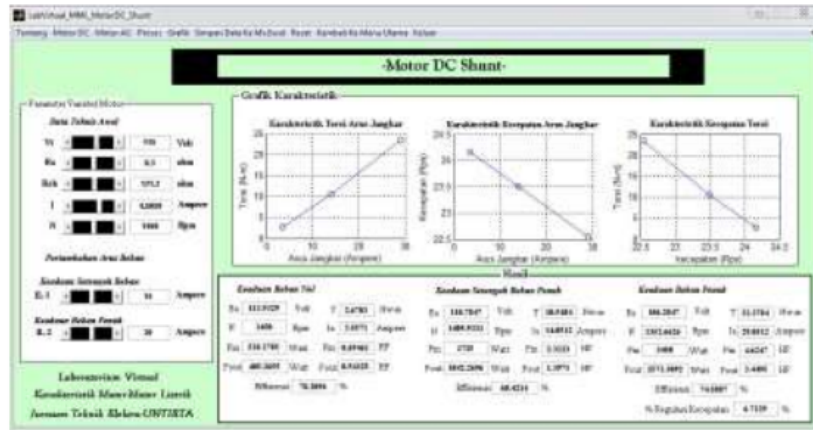
- 1) *Memproses komputasi untuk pengujian Motor DC: Shunt, Seri, dan kompon*
  - 2) *Memproses komputasi untuk pengujian Motor AC: Kutub Bayangan dan motor induksi 3 fasa*
  - 3) *Memproses komputasi untuk pengujian Motor Sinkron*
- 

### A. GRAFIK HASIL PENGUJIAN MOTOR DC

Pengujian pada motor DC dilakukan pada semua motor DC yang dirancang, antara lain motor DC shunt, seri, kompon panjang dan kompon pendek karena setiap motor tersebut mempunyai karakteristik yang berbeda. Motor DC mempunyai tiga karakteristik yaitu berupa karakteristik listrik, karakteristik kecepatan terhadap arus jangkar serta karakteristik mekanik. Setiap pengujian akan menunjukkan setiap karakteristik tersebut berupa data hasil perhitungan dan grafik yang dapat disimpan oleh pengguna. Analisa dilakukan berdasarkan hasil pengujian dan dapat dibuktikan dengan merujuk ke persamaan dan dasar teori yang telah dijelaskan.

#### 1. Grafik Hasil Pengujian Motor DC Shunt

Pengujian ini dilakukan untuk mengetahui karakteristik pada motor DC shunt saat arus beban bertambah sehingga didapatkan karakteristik torsi terhadap arus jangkar yang disebut karakteristik elektris, karakteristik kecepatan terhadap arus jangkar, dan karakteristik kecepatan terhadap torsi yang disebut karakteristik mekanik. Gambar 11.1 berikut menunjukkan salah satu pengujian pada motor DC shunt.



Gambar 11.1 Pengujian Motor DC Shunt

Pengujian dilakukan dengan memasukkan parameter data teknis pada laboratorium virtual motor DC shunt dan memberi penambahan arus beban pada kondisi setengah beban penuh dan beban penuh. Pengujian yang telah dilakukan memperlihatkan bahwa karakteristik pada motor DC shunt jika arus jangkar besar torsi akan bertambah secara linear dengan bertambahnya arus jangkar dan kecepatan motor akan menurun. Hal ini dipengaruhi oleh bertambahnya arus beban pada saat keadaan setengah beban penuh yaitu 15 A dan arus beban penuh 30 A, sementara arus medan besarnya konstan karena kumparan medan langsung terhubung dengan sumber tegangan yang dianggap konstan sesuai dengan persamaan (9-2), (9-3) dan (9-1).

$$I_{sh} = \frac{V_t}{R_{sh}} = \frac{115}{121.2} = 0.948 \text{ A}$$

$$I_{a_0} = I_{L_0} - I_{sh} = 4.5059 - 0.948 = 3.557 \text{ A}$$

$$I_{a_1} = I_{L_1} - I_{sh} = 15 - 0.948 = 14.052 \text{ A}$$

$$I_{a_2} = I_{L_2} - I_{sh} = 30 - 0.948 = 29.052 \text{ A}$$

$$E_{a_0} = V_t - (I_{a_0} \times R_a) = 115 - (3.557 \times 0.3) = 113.9329 \text{ V}$$

$$E_{a_1} = V_t - (I_{a_1} \times R_a) = 115 - (14.052 \times 0.3) = 110.7844 \text{ V}$$

$$E_{a_2} = V_t - (I_{a_2} \times R_a) = 115 - (29.052 \times 0.3) = 106.2844 \text{ V}$$

Berdasarkan grafik yang terlihat pada tampilan laboratorium virtual dapat dianalisa bahwa karakteristik kecepatan motor DC dapat dikatakan konstan walau saat penambahan beban motor mulai berkurang kecepatannya tetapi perubahan kecepatan diantara tanpa beban dan beban penuh lebih kecil daripada tipe motor lainnya, dapat dilihat bahwa pada saat beban nol kecepatan motor adalah 24.16 Rps

= 1450 Rpm pada torsi 2.67032 Nm, pada keadaan setengah beban penuh adalah 23.49 Rps = 1409.93 Rpm dengan torsi 10.54835 Nm, dan pada keadaan beban penuh adalah 22.54 Rps = 1352.66 Rpm dengan torsi beban penuh 23.37839 Nm.

Sesuai dengan persamaan (9-27) dan (9-35) pada motor DC shunt, saat bertambahnya arus beban sebesar 15 Adan 30 A, karena fluks tetap maka:

$$N_1 = \frac{E_{a_1} N_0}{E_{a_0}} = \frac{110.78 \times 1450}{113.93} = 1409.9 \text{ Rpm} = 23.49 \text{ Rps}$$

$$N_2 = \frac{E_{a_2} N_0}{E_{a_0}} = \frac{106.2847 \times 1450}{113.93} = 1352.69 \text{ Rpm} = 22.54 \text{ Rps}$$

$$T_0 = \frac{E_{a_0} \times I_{a_0}}{2\pi n_0} = \frac{113.9329 \times 3.557}{6.28 \times 24.16} = 2.67 \text{ Nm}$$

$$T_1 = \frac{E_{a_1} \times I_{a_1}}{2\pi n_1} = \frac{110.7844 \times 14.052}{6.28 \times 23.49} = 10.55 \text{ Nm}$$

$$T_2 = \frac{E_{a_2} \times I_{a_2}}{2\pi n_2} = \frac{106.2844 \times 29.052}{6.28 \times 22.54} = 21.81 \text{ Nm}$$

Persentasi perubahan kecepatan dari tanpa beban sampai beban penuh adalah:

$$\% \text{ Regulasi Kecepatan} = \frac{1450 - 1352.6626}{1450} \times 100 = 6.7129 \%$$

Kecepatan, torsi dan arus jangkar pada motor DC shunt membuat keseimbangan sangat cepat menghasilkan hanya perubahan kecil pada kecepatannya, oleh karena itu motor DC shunt banyak digunakan secara luas karena karakteristik kecepatannya yang konstan. Data Pengujian pada motor DC shunt disimpan dalam bentuk *excel* dengan menekan *toolbar* "Simpan Data Ke Ms.Excel", data pengujian motor DC shunt ditunjukkan pada Tabel 11.1.

**Tabel 11.1** Data Pengujian Motor DC Shunt

<b>Data Teknis Awal</b>		<b>Vt</b>	115	<b>Volt</b>
		<b>Ra</b>	0.3	<b>ohm</b>
		<b>Rsh</b>	121.2	<b>ohm</b>
		<b>I</b>	4.5059	<b>Ampere</b>
		<b>N</b>	1450	<b>Rpm</b>
<b>Pertambahan Beban</b>		<b>IL 1</b>	15	<b>Ampere</b>
		<b>IL 2</b>	30	<b>Ampere</b>
<b>Hasil</b>	<b>Keadaan Beban Nol</b>	<b>Ea</b>	113.93288	<b>Volt</b>
		<b>N</b>	1450	<b>Rpm</b>
		<b>T</b>	2.6703199	<b>Nw-m</b>

		<b>Ia</b>	3.5570551	<b>Ampere</b>
		<b>Pin</b>	518.1785	<b>Watt</b>
		<b>Pout</b>	405.26555	<b>Watt</b>
		<b>Effisiensi</b>	78.209641	<b>%</b>
	<b>Keadaan Setengah Beban Penuh</b>	<b>Ea</b>	110.78465	<b>Volt</b>
		<b>N</b>	1409.9331	<b>Rpm</b>
		<b>T</b>	10.548355	<b>Nw-m</b>
		<b>Ia</b>	14.051155	<b>Ampere</b>
		<b>Pin</b>	1725	<b>Watt</b>
		<b>Pout</b>	1042.2696	<b>Watt</b>
		<b>Effisiensi</b>	60.421429	<b>%</b>
	<b>Keadaan Beban Penuh</b>	<b>Ea</b>	106.28465	<b>Volt</b>
		<b>N</b>	1352.6626	<b>Rpm</b>
<b>T</b>		21.809017	<b>Nw-m</b>	
<b>Ia</b>		29.051155	<b>Ampere</b>	
<b>Pin</b>		3450	<b>Watt</b>	
<b>Pout</b>		2573.3092	<b>Watt</b>	
<b>Effisiensi</b>		74.588674	<b>%</b>	
<b>Regulasi Kecepatan</b>	<b>Regulasi Kecepatan</b>	6.7129259	<b>%</b>	

Dari data pengujian dapat dianalisa bahwa daya *output* bertambah besar seiring kenaikan arus beban, hal ini terjadi karena arus beban mempengaruhi arus jangkar sehingga rugi-rugi pada jangkar akan semakin naik, sementara rugi-rugi medan konstan. Analisa perhitungan untuk pembuktian kenaikan daya *output* pada saat beban nol hingga mencapai beban penuh sebagai berikut:

$$P_{cu_0} = (I_{a_0})^2 \cdot R_a = (3.557^2 \times 0.3) = 3.7956 \text{ W}$$

$$P_{cu_1} = (I_{a_1})^2 \cdot R_a = (14.052^2 \times 0.3) = 59.237 \text{ W}$$

$$P_{cu_2} = (I_{a_2})^2 \cdot R_a = (29.052^2 \times 0.3) = 253.205 \text{ W}$$

$$P_{rugi\ tetap} = P_{in_0} - P_{cu_0} = 518.1785 - 3.7956 = 514.3829 \text{ W}$$

$$P_{medan} = (I_{sh})^2 \cdot R_{sh} = 0.948^2 \times 121.2 = 108.922 \text{ W}$$

$$P_{out_0} = P_{in_0} - P_{cu_0} - P_{medan} = 518.17 - 3.7956 - 108.92 = 405.46 \text{ W}$$

$$\begin{aligned} P_{out_1} &= P_{in_1} - P_{cu_1} - P_{medan} - P_{rugi\ tetap} \\ &= 1725 - 59.237 - 108.922 - 514.3829 = 1042.45 \text{ W} \end{aligned}$$

$$P_{out_2} = P_{in_2} - P_{cu_2} - P_{medan} - P_{rugi\ tetap}$$

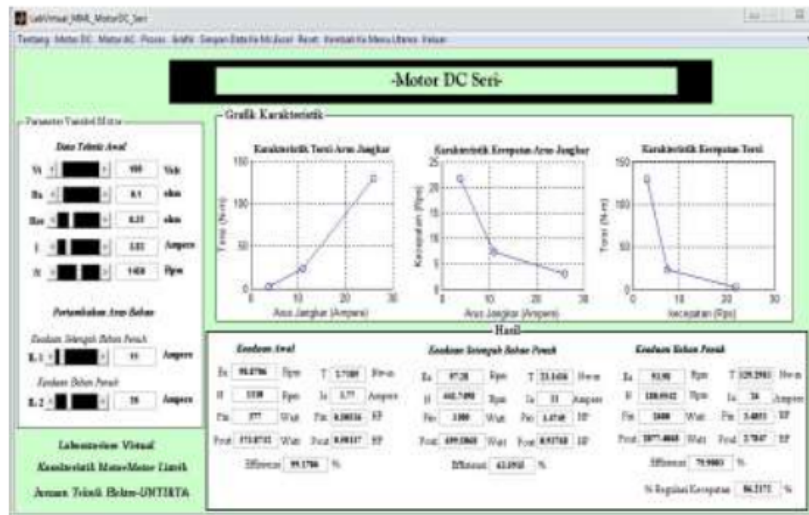
$$= 3450 - 253.205 - 108.922 - 514.3829 = 2573.49 \text{ W}$$

Sehingga efisiensi pada motor DC shunt yang diuji pada saat kondisi beban penuh adalah 74.58867 %.

92

## 2. Grafik Hasil Pengujian Motor DC Seri

Pengujian motor DC seri dilakukan untuk menemukan karakteristik perubahan kecepatan motor dan torsi apabila arus beban motor bertambah, pada pengujian ini sama seperti pengujian yang dilakukan oleh motor DC shunt dengan memberi parameter data teknis pada motor dan pemberian arus beban setengah beban penuh dan beban penuh. Gambar 4.3 menunjukkan pengujian terhadap motor DC seri.



Gambar 11.2 Pengujian Motor DC Seri

Hasil perhitungan dan grafik pada pengujian motor DC seri menunjukkan penurunan yang sangat drastis pada kecepatan motor saat arus beban bertambah, hal ini disebabkan karena pada motor DC seri arus medan dan arus beban adalah sama, merujuk pada persamaan (2-5). Keluaran torsi sebanding dengan arus jangkar dan fluks medan, hal ini yang menjadikan torsi sebanding dengan kuadrat arus jangkar, untuk beban ringan seperti yang ditampilkan pada hasil keadaan awal motor DC seri hanya menghasilkan torsi 2.65 Nm yaitu torsi yang sangat kecil untuk arus jangkar 3.65 A, dan pada saat arus beban bertambah 12 A dan 22 A motor DC seri menghasilkan torsi yang lebih besar daripada motor DC shunt yaitu 28.60 Nm dan 97.95 Nm. Perubahan kecepatan sesuai dengan persamaan (2-26) pada motor DC seri karena fluks tetap maka:

$$\frac{n_1}{n_0} =$$

$$n_0 \times \frac{E_{a1}}{E_{a0}} \times \frac{I_{a0}}{I_{a1}} = 1310 \times \frac{96.16}{98.832} \times \frac{3.65}{12} = 387.68 \text{ Rpm} = 6.466 \text{ Rps}$$

$$\frac{n_2}{n_0} =$$

$$n_0 \times \frac{E_{a2}}{E_{a0}} \times \frac{I_{a0}}{I_{a2}} = 1310 \times \frac{92.96}{98.832} \times \frac{3.65}{22} = 204.42 \text{ Rpm} = 3.407 \text{ Rps}$$

107

Persentasi perubahan kecepatan dari tanpa beban sampai beban penuh adalah:

$$\% \text{ Regulasi Kecepatan} = \frac{1310 - 204.42}{1310} \times 100 = 84.39 \%$$

Sesuai dengan persamaan (2-35) torsi motor dapat diketahui dengan mendapatkan tegangan jangkar dan arus jangkar dengan perubahan arus hingga beban penuh adalah:

$$T_0 = \frac{E_{a0} I_{a0}}{2\pi n_0} = \frac{98.832 \times 12}{6.28 \times 21.8} = 2.63 \text{ Nm}$$

$$T_1 = \frac{E_{a1} I_{a1}}{2\pi n_1} = \frac{96.16 \times 22}{6.28 \times 6.466} = 28.44 \text{ Nm}$$

$$T_2 = \frac{E_{a2} I_{a2}}{2\pi n_2} = \frac{92.96 \times 22}{6.28 \times 3.407} = 95.58 \text{ Nm}$$

Motor DC shunt dapat beroperasi tanpa beban, sementara motor DC seri tidak, motor DC shunt dapat beroperasi tanpa beban disebabkan tersedia fluks yang kuat, sementara pada motor DC seri ter<sup>139</sup> dari rangkaian ekivalen pada Gambar 11.3 apabila tidak ada arus beban maka arus jangkar sama dengan nol, harga fluks juga nol, sehingga dapat dilihat pada persamaan (9-20) diperoleh kecepatan yang tak terhingga, dan sebaliknya dengan nilai arus jangkar yang besar kecepatan akan mendekati nol atau semakin berkurang. Data Pengujian motor DC seri yang tersimpan dalam Ms.Excel ditunjukkan pada Tabel 11.2.

**Table 11.2** Data Pengujian Motor DC Seri

<b>Data Teknis Awal</b>	<b>Vt</b>	100	<b>Volt</b>
	<b>Ra</b>	0.1	<b>ohm</b>
	<b>Rse</b>	0.22	<b>ohm</b>
	<b>I</b>	3.65	<b>Ampere</b>
	<b>N</b>	1310	<b>Rpm</b>



<b>Pertambahan Beban</b>		<b>IL 1</b>	12	<b>Ampere</b>
		<b>IL 2</b>	22	<b>Ampere</b>
<b>Hasil</b>	<b>Keadaan Beban Nol</b>	<b>Ea</b>	98.832	<b>Volt</b>
		<b>N</b>	1310	<b>Rpm</b>
		<b>T</b>	2.63093888	<b>Nw-m</b>
		<b>Ia</b>	3.65	<b>Ampere</b>
		<b>Pin</b>	365	<b>Watt</b>
		<b>Pout</b>	363.66775	<b>Watt</b>
		<b>Effisiensi</b>	99.635	<b>%</b>
		<b>Keadaan Setengah Beban Penuh</b>	<b>Ea</b>	96.16
	<b>N</b>		387.685702	<b>Rpm</b>
	<b>T</b>		28.4372452	<b>Nw-m</b>
	<b>Ia</b>		12	<b>Ampere</b>
	<b>Pin</b>		1200	<b>Watt</b>
	<b>Pout</b>		821.93225	<b>Watt</b>
	<b>Effisiensi</b>		68.4943542	<b>%</b>
	<b>Keadaan Beban Penuh</b>	<b>Ea</b>	92.96	<b>Volt</b>
		<b>N</b>	204.427826	<b>Rpm</b>
		<b>T</b>	95.5807408	<b>Nw-m</b>
		<b>Ia</b>	22	<b>Ampere</b>
		<b>Pin</b>	2200	<b>Watt</b>
		<b>Pout</b>	1787.93225	<b>Watt</b>
		<b>Effisiensi</b>	81.2696477	<b>%</b>
<b>Regulasi Kecepatan</b>		<b>%Regulasi Kecepatan</b>	84.3948224	<b>%</b>

Dilihat dari data pengujian dapat dianalisa bahwa daya *output* bertambah besar seiring kenaikan arus beban, hal ini terjadi karena arus beban mempengaruhi arus jangkar sehingga rugi-rugi pada jangkar akan semakin naik, sementara rugi medan pada perhitungan motor ini diabaikan. Perhitungan untuk daya *input* dan *output* motor DC seri sama dengan perhitungan pada motor shunt dengan mengabaikan rugi-rugi medan.

### 3. Grafik Hasil Pengujian Motor DC Kompon

Pengujian ini dilakukan untuk mengetahui karakteristik pada motor DC kompon panjang dan kompon pendek saat arus beban bertambah sehingga didapatkan karakteristik torsi terhadap arus jangkar yang disebut karakteristik

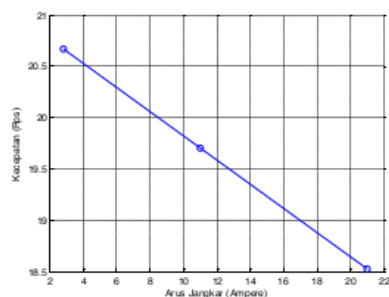
elektris, karakteristik kecepatan terhadap arus jangkar, dan karakteristik kecepatan terhadap torsi yang disebut karakteristik mekanik. Pengujian pada motor ini masing-masing dilakukan agar mengetahui pula perbedaan karakteristik antara motor DC kompon panjang dan kompon pendek dengan parameter data teknis yang sama. Tabel 11.3 menunjukkan data pengujian motor DC kompon panjang dan kompon pendek.

**Tabel 11.3** Data Pengujian Motor DC Kompon Panjang dan Kompon Pendek

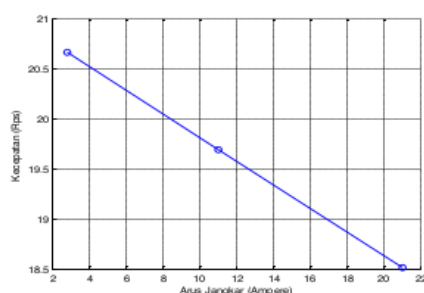
<b>Data Teknis</b>	<b>Vt</b>	107	<b>Volt</b>	
	<b>Ra</b>	0.3	<b>ohm</b>	
	<b>Rsh</b>	108	<b>ohm</b>	
	<b>Rse</b>	0.3	<b>ohm</b>	
	<b>I</b>	3.8	<b>Ampere</b>	
	<b>N</b>	1240	<b>Rpm</b>	
<b>Pertambahan Beban</b>	<b>IL 1</b>	12	<b>Ampere</b>	
	<b>IL 2</b>	22	<b>Ampere</b>	
<b>Hasil</b>		<b>Kompon Panjang</b>	<b>Kompon Pendek</b>	
<b>Kedadaan Beban Nol</b>	<b>Ea</b>	105.3144444	105.0172222	<b>Volt</b>
	<b>N</b>	1240	1240	<b>Rpm</b>
	<b>T</b>	2.279552945	2.273119508	<b>Nw-m</b>
	<b>Ia</b>	2.809259259	2.809259259	<b>Ampere</b>
	<b>Pin</b>	406.6	406.6	<b>Watt</b>
	<b>Pout</b>	404.2324187	404.2324187	<b>Watt</b>
	<b>Effisiensi</b>	99.41771243	99.41771243	<b>%</b>
<b>Kedadaan Setengah Beban Penuh</b>	<b>Ea</b>	100.3944444	100.0972222	<b>Volt</b>
	<b>N</b>	1182.070624	1181.906671	<b>Rpm</b>
	<b>T</b>	8.933383162	8.908171046	<b>Nw-m</b>
<b>Kedadaan Setengah Beban Penuh</b>	<b>Ia</b>	11.00925926	11.00925926	<b>Ampere</b>
	<b>Pin</b>	1284	1284	<b>Watt</b>
	<b>Pout</b>	843.4064444	843.4064444	<b>Watt</b>
	<b>Effisiensi</b>	65.68586016	65.68586016	<b>%</b>

<b>Keadaan Beban Penuh</b>	<b>Ea</b>	94.39444444	94.09722222	<b>Volt</b>
	<b>N</b>	1111.425045	1111.061149	<b>Rpm</b>
	<b>T</b>	17.04781026	16.99969731	<b>Nw-m</b>
	<b>Ia</b>	21.00925926	21.00925926	<b>Ampere</b>
	<b>Pin</b>	2354	2354	<b>Watt</b>
	<b>Pout</b>	1817.350889	1817.350889	<b>Watt</b>
	<b>Effisiensi</b>			
	<b>i</b>	77.20267158	77.20267158	<b>%</b>
<b>% Regulasi Kecepatan</b>		10.36894802	10.39829446	<b>%</b>

Berdasarkan hasil pengamatan pada data pengujian dan grafik pada motor 38 kompon panjang dan kompon pendek dengan parameter data teknis yang sama, penurunan kecepatan 38 akibat pertambahan arus beban sangat kecil. Hal ini disebabkan karena tahanan jangkar  $R_a$  sangat kecil sehingga penurunan kecepatan dari tanpa beban hingga beban penuh adalah kecil, sehingga motor ini dapat dikategorikan sebagai motor yang mempunyai kecepatan konstan seperti motor DC shunt karena sebenarnya motor DC kompon berada diantara karakteristik motor seri dan motor shunt. Gambar 11.4 menunjukkan grafik kecepatan dan arus jangkar motor DC kompon panjang dan kompon pendek.



(a)



(b)

51

**Gambar 11.4** Grafik Kecepatan dan Arus Jangkar; (a) Motor DC Kompon Panjang, (b) Motor DC Kompon Pendek

Perbedaan yang mempengaruhi motor DC kompon panjang dan kompon pendek adalah letak dari resistansi seri motor tersebut sehingga mempengaruhi perhitungan untuk tegangan jangkar walau perbedaan nilainya sangat kecil, dapat dilihat pada Gambar 2.4 dan 2.5 yaitu rangkaian ekuivalen motor DC kompon panjang dan kompon pendek, dengan perhitungan merujuk pada persamaan (9-6) dan (9-8) maka nilai tegangan jangkar pada motor kompon panjang adalah:

$$Ea_0 = Vt - Ia_0(R_a + R_s) = 107 - 2.81(0.3 + 0.3) = 105.314 V$$

$$Ea_1 = 107 - 11.01(0.3 + 0.3) = 100.394 V$$

$$Ea_2 = 107 - 21.01(0.3 + 0.3) = 94.394 V$$

Sementara perhitungan pada motor kompon pendek adalah:

$$Ea_0 = Vt - ((Ia_0 \times R_a) + (IL_0 \times R_s)) = 107 - ((2.81 \times 0.3) + (3.8 \times 0.3)) = 105.017 V$$

$$Ea_1 = 107 - ((11.01 \times 0.3) + (12 \times 0.3)) = 100.097 V$$

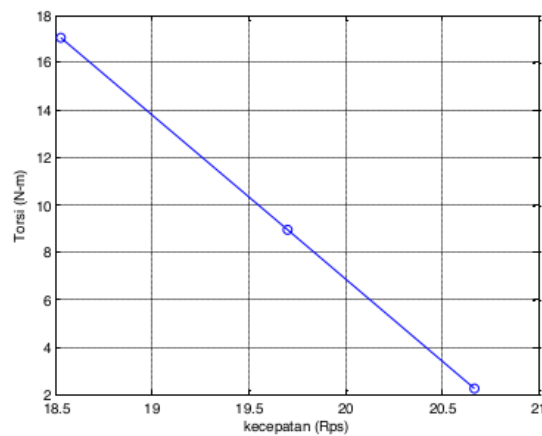
$$Ea_2 = 107 - ((21.01 \times 0.3) + (22 \times 0.3)) = 94.097 V$$

Perbedaan tegangan jangkar pada kedua motor tersebut mempengaruhi kecepatan putaran pada motor dan torsi motor tersebut, pada grafik penurunan kecepatan terlihat sama antara kedua motor namun sebenarnya ada sedikit perbedaan kecepatan antara motor tersebut seperti terlihat pada data hasil pengujian. Motor DC kompon panjang berputar dengan arus beban penuh pada kecepatan 1111.425045 Rpm dan torsi 17.04781026 Nm, sedangkan pada motor DC kompon pendek pada arus beban penuh berputar pada kecepatan 1111.061149 Rpm dan torsi 16.99969731 Nm dan dapat dikatakan sesuai dengan persamaan (2-27) dan pada grafik keluaran dari laboratorium virtual yang berbentuk *figure* seperti ditunjukkan pada Gambar 4.5 untuk motor DC kompon panjang. Persentasi perubahan kecepatan pada motor DC kompon panjang dari tanpa beban sampai beban penuh adalah:

$$\% \text{Regulasi Kecepatan} = \frac{1240 - 1111.425}{1240} \times 100 = 10.368 \%$$

Sementara untuk motor DC kompon pendek adalah:

$$\% \text{Regulasi Kecepatan} = \frac{1240 - 1111.0611}{1240} \times 100 = 10.398 \%$$



**Gambar 11.5** Grafik Torsi dan Kecepatan Motor DC Kompon Panjang

Daya *input* dan *output* pada kedua motor DC kompon panjang dan kompon pendek adalah sama karena tegangan terminal dan perubahan beban yang dipasok ke motor adalah sama dapat dilihat dengan perhitungan berikut:

$$Pin_0 = Vt \times I_{L0} = 107 \times 3.8 = 406.6 \text{ W} = 0.545 \text{ HP}$$

$$Pin_1 = 107 \times 12 = 1284 \text{ W} = 1.721 \text{ HP}$$

$$Pin_2 = 107 \times 22 = 2354 \text{ W} = 3.15 \text{ HP}$$

$$Pcu_0 = (Ia_0)^2 \cdot R_a = (2.81^2 \times 0.3) = 2.368 \text{ W}$$

$$Pcu_1 = (Ia_1)^2 \cdot R_a = (11.01^2 \times 0.3) = 36.36 \text{ W}$$

$$Pcu_2 = (Ia_2)^2 \cdot R_a = (21.01^2 \times 0.3) = 132.42 \text{ W}$$

$$P_{rugi \text{ tetap}} = Pin_0 - Pcu_0 = 406.6 - 2.368 = 404.232 \text{ W}$$

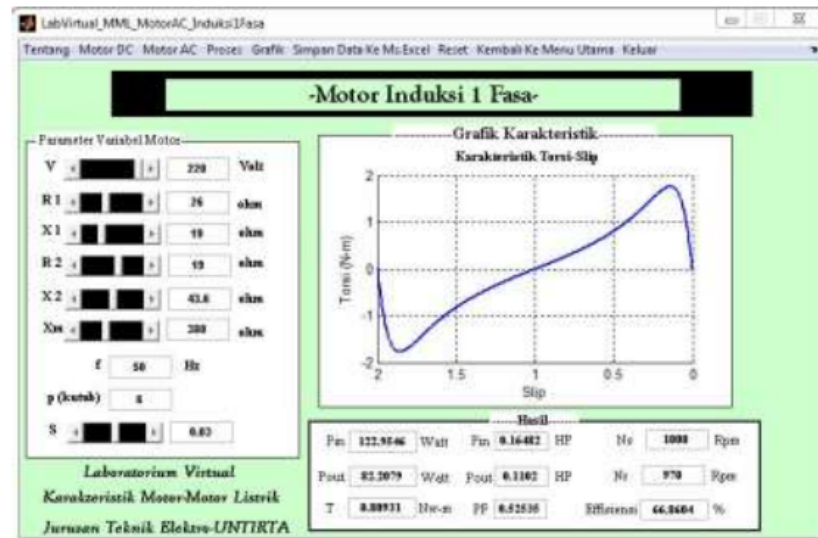
$$Pout_0 = Pin_0 - Pcu_0 = 406.6 - 2.368 = 404.232 \text{ W}$$

$$\begin{aligned} Pout_1 &= Pin_1 - Pcu_1 - P_{rugi \text{ tetap}} \\ &= 1284 - 36.36 - 404.232 = 843.408 \text{ W} \end{aligned}$$

$$\begin{aligned} Pout_2 &= Pin_2 - Pcu_2 - P_{rugi \text{ tetap}} \\ &= 2354 - 132.42 - 404.232 = 1817.348 \text{ W} \end{aligned}$$

## B. GRAFIK HASIL PENGUJIAN MOTOR AC SATU FASA

Pengujian motor AC satu fasa dilakukan untuk mengetahui karakteristik dari motor induksi satu fasa, motor satu fasa yang di rancang pada laboratorium virtual ini adalah motor induksi satu fasa, motor fasa belah, motor kapasitor, dan motor kutub bayangan. Pengujian motor induksi satu fasa dilakukan agar pengguna dapat mengetahui bahwa motor induksi tidak dapat berputar apabila tidak diberi kumparan bantu, pengujian motor induksi 1 fasa ditunjukkan pada Gambar 11.6.



Gambar 11.6 Pengujian Motor Induksi Satu Fasa

Pengujian dilakukan dengan memberi nilai parameter pada motor dan juga slip untuk mengetahui besarnya daya, arus, torsi dan kecepatan walaupun motor sebenarnya tidak dapat bergerak. Berdasarkan pengamatan grafik pada motor induksi satu fasa ini mempunyai torsi yang berlawanan arah maju dan mundur yang pada dasarnya kopel tersebut mempunyai kemampuan untuk menggerakkan motor dengan arah maju atau mundur, tetapi pada keadaan start yaitu slip sama dengan 1 kemampuan motor untuk maju sama besar dengan kemampuan gerak mundurnya, oleh sebab itu motor tetap diam maka perlu adanya alat bantu untuk memberikan sedikit torsi untuk maju, motor akan berputar mengikuti kopel tersebut, seperti yang dilakukan motor fasa belah, motor kapasitor dan motor kutub bayangan. Analisis perhitungan untuk membuktikan besarnya komponen impedansi maju dan mundur pada motor induksi satu fasa dengan slip 0.03 adalah:

$$Z_F = \frac{(R_2/s + jX_2)(jX_M)}{(R_2/s + jX_2) + jX_M} = \frac{(19/0.03 + j43.6) \times j380}{19/0.03 + j43.6 + j380} = \frac{j240665.4 - 16568}{633.33 + j423.6}$$

$$= \frac{241235.0169 \angle 93.938}{761.934 \angle 33.776} = 316.6 \angle 60.162 = 157.524 + j274.63 \Omega$$

$$Z_B = \frac{[R_2/(2-s) + jX_2](jX_M)}{[R_2/(2-s) + jX_2] + (jX_M)} = \frac{((19/(2-0.03)) + (j43.6)) \times (j380)}{(19/(2-0.03)) + (j43.6) + (j380)}$$

$$= \frac{j3664.97 - 16568}{9.64467 + j423.6} = \frac{16968.518 \angle 167.52}{423.709 \angle 88.69} = 40.047 \angle 78.83 = 7.7579 + j39.288 \Omega$$

Maka didapat arus dan daya *input* pada motor induksi 1 fasa sebesar:

$$I = \frac{V}{R_1 + jX_1 + 0.5 Z_F + 0.5 Z_B} = \frac{220}{26 + j19 + 78.762 + j137.315 + 3.8789 + j19.644}$$

$$= \frac{220 \angle 0}{108.64 + j175.959} = \frac{220 \angle 0}{206.795 \angle 58.308} = 1.06385 \angle -58.308 A$$

$$P_{in} = V \cdot I \cdot \cos \theta = 220 \times 1.06385 \times (\cos -58.308) = 122.944 W$$

$$P_F = I^2 R_F = 1.06385^2 (0.5 \times 157.524) = 89.141 W$$

$$P_B = I^2 R_B = 1.06385^2 (0.5 \times 7.7579) = 4.390 W$$

$$P = P_F - P_B = 89.141 - 4.390 = 84.751 W$$

$$T_{ind} = \frac{P_{AG}}{\omega_s} = \frac{84.751}{1000 \times 6.28 / 60} = 0.809 Nm$$

$$P_{conv_F} = (1 - S) \times P_F = (1 - 0.03) \times 89.141 = 86.46677 W$$

$$P_{conv_B} = (1 - S) \times P_B = (1 - 0.03) \times 4.39 = 4.2583 W$$

$$P_{out} = P_{conv} = P_{conv_F} - P_{conv_B} = 86.4667 - 4.2583 = 82.20 W$$

Persamaan 98 yang digunakan untuk menghitung besarnya torsi yang dihasilkan oleh motor induksi satu fasa merujuk pada rangkaian ekuivalen motor pada Gambar 11.10 dengan persamaan umum untuk perhitungan (9-45) hingga didapat nilai torsi nominal pada motor ini adalah sebesar 0.809 Nm dengan kecepatan stator motor apabila berputar adalah 1000 Rpm dari perhitungan persamaan (9-53):

$$n_s = \frac{120 \times 50}{6} = 1000 Rpm = 16.67 Rps$$

sedangkan untuk kecepatan rotor motor apabila motor bergerak pada slip 0.03 adalah:

$$N_r = (1 - S) \times N_s = (1 - 0.03) \times 1000 = 970 Rpm = 16.1667 Rps$$

Data 50 pengujian motor induksi satu fasa yang disimpan dalam Ms.Excel ditunjukkan pada Tabel 11.4.

**Tabel 11.4** Pengujian Motor Induksi 1 Fasa

<b>Data Teknis Awal</b>	<b>V</b>	220	<b>Volt</b>
	<b>R1</b>	26	<b>ohm</b>
	<b>X1</b>	19	<b>ohm</b>
	<b>R2</b>	19	<b>ohm</b>
	<b>X2</b>	43.6	<b>ohm</b>
	<b>Xm</b>	380	<b>ohm</b>
	<b>f</b>	50	<b>Hz</b>
	<b>p</b>	6	
	<b>S</b>	0.03	
<b>Hasil</b>	<b>Pin</b>	122.95464	<b>Watt</b>
	<b>Pin</b>	0.1648186	<b>HP</b>
	<b>Pout</b>	82.207937	<b>Watt</b>
	<b>Pout</b>	0.1101983	<b>HP</b>
	<b>T</b>	0.8093072	<b>Nm</b>
	<b>PF</b>	0.5253532	
	<b>Effisiens i</b>	66.860375	<b>%</b>

### 1. Grafik Hasil Pengujian Motor Kutub Bayangan

55

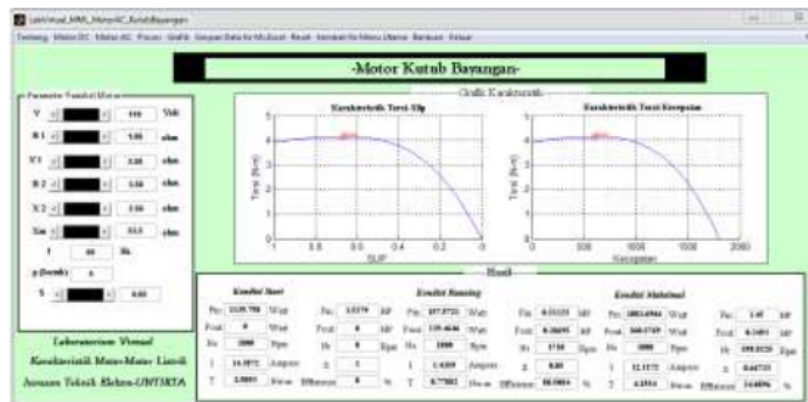
Pengujian ini dilakukan untuk melihat karakteristik motor induksi 1 fasa jenis motor kutub bayangan. Motor kutub bayangan dapat berputar karena pada kutub-kutubnya disisipkan kumparan tambahan. Parameter pengujian motor ini menggunakan soal tentang motor induksi satu fasa 50 pada *e-book* BL Theraja, pada soal parameter motor ini dihitung menggunakan rangkaian ekivalen motor induksi satu fasa dalam keadaan diam dengan pengaruh medan magnet maju mundur, sementara pada pengujian ini parameter digunakan untuk motor dengan kondisi berputar tanpa ada pengaruh perubahan medan magnet maju mundur. Parameter motor induksi satu fasa sebagai berikut:

$V = 110 \text{ V}$ ,  $60 \text{ Hz}$ , 4 kutub, Slip  $0.05$ .

$R1 = 1.86 \Omega$ ;  $X1 = 2.56 \Omega$ ;  $Xm = 53.5 \Omega$ ;  $R2 = 3.56$ ;  $X2 = 2.56$ .



Gambar pengujian motor kutub bayangan ditunjukkan pada Gambar 4.8.



**Gambar 11.8** Pengujian Motor Kutub Bayangan

110

Hasil pengujian motor kutub bayangan yang ditunjukkan pada Gambar 4.8 dapat dilihat bahwa dengan parameter yang telah ditentukan motor menghasilkan torsi 3.9093 Nm pada saat slip 1 atau kondisi *start* dengan kecepatan stator sebesar 1800 Rpm dan kecepatan rotor adalah 0 karena motor dalam kondisi diam. Dapat dihitung dengan persamaan (2-53):

$$N_s = \frac{120 \times 60}{4} = 1800 \text{ Rpm}$$

$$N_r = (1 - s) \times N_s = (1 - 1) \times 1800 = 0$$

Arus *starting* pada pengujian motor ini lebih tinggi dari arus nominal motor berputar yaitu sebesar 14.3872 A. Saat kondisi nominal pada slip yang ditentukan sebesar 0.05 motor berputar dengan kecepatan rotor 1710 Rpm dengan torsi nominal kecil sebesar 0.77872 Nm dan arus nominal sebesar 1.435 A.

Analisis perhitungan untuk arus pada saat keadaan *starting* dan saat keadaan nominal dapat dibuktikan dengan persamaan (2-60) dengan menggunakan teroema thevenin perfasa:

$$V_{TH} = \frac{X_M}{\sqrt{R_1^2 + (X_1 + X_M)^2}} \times V_p = \frac{53.5}{\sqrt{(1.86^2) + (2.56 + 53.5)^2}} \times 110 = 104.919 \text{ V}$$

$$Z_{TH} = \frac{jX_M(R_1 + jX_1)}{R_1 + j(X_1 + X_M)} = \frac{j53.5(1.86 + j2.56)}{1.86 + j(2.56 + 53.5)} = \frac{j8.4 - 12.6}{0.5 + j17.55} = 1.6921 + j2.4992$$

Pada kondisi diam atau slip = 1, maka:

$$I = \frac{V_{TH}}{[(R_{TH} + R_2/s) + j(X_{TH} + X_2)]} = \frac{104.919}{(1.6921 + (3.56/1) + j(2.4992 + 2.56))} = 14.35 \angle - 3.969 A$$

Dan pada kondisi nominal slip = 0.05:

$$I = \frac{104.919}{(1.6921 + (3.56/0.05) + j(2.4992 + 2.56))} = 14.387 \angle - 43.928 A$$

Saat kondisi maksimal atau pada beban penuh slip pada motor kutub bayangan ini cukup tinggi yaitu sebesar 0.667, dengan torsi 4.15 Nm, dan daya keluaran yang kecil yaitu sebesar 260.57 W sehingga pada saat kondisi beban penuh motor kutub bayangan yang diuji mempunyai efisiensi yang sangat rendah. Torsi awal yang cukup tinggi daripada torsi nominal menunjukkan bahwa tipe motor kutub bayangan yang diuji merupakan motor induksi satu fasa kelas D karena mempunyai torsi *starting* yang tinggi dan slip besar dan pada beban penuh mempunyai efisiensi yang lebih rendah dari pada motor induksi kelas lain. Analisis perhitungan torsi pada pengujian motor kutub bayangan untuk pembuktian pada kondisi nominal motor dengan slip 0.05 sebagai berikut:

$$\omega_s = \frac{(Ns \times 2\pi)}{60} = \frac{1800 \times 6.28}{60} = 188.4$$

Menggunakan persamaan thevenin untuk torsi per fasa slip 1 atau motor keadaan diam:

$$T = \frac{V_{TH}^2 R_2/s}{\omega_s [(R_{TH} + R_2/s)^2 + (X_{TH} + X_2)^2]} = \frac{104.9191 \times 3.56/1}{188.4 \times ((1.6921 + 3.56/1)^2 + (2.4992 + 2.56)^2)}$$

$$= 3.9093 Nm$$

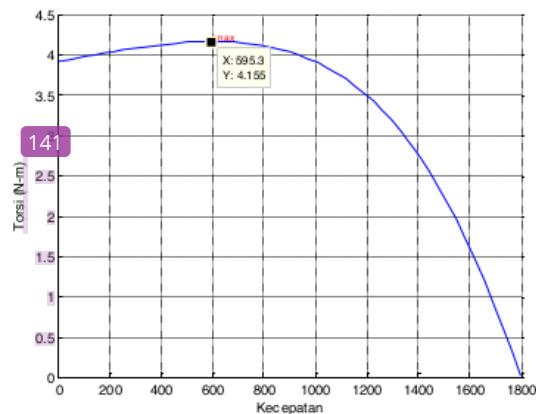
Torsi pada slip nominal = 0.05:

$$T = \frac{V_{TH}^2 R_2/s}{\omega_s [(R_{TH} + R_2/s)^2 + (X_{TH} + X_2)^2]} = \frac{104.9191 \times 3.56/0.05}{188.4 \times ((1.6921 + 3.56/0.05)^2 + (2.4992 + 2.56)^2)}$$

$$= 0.7782 Nm$$

Torsi maksimal motor didapat dari perhitungan pada slip maksimal atau pada beban penuh, slip pada pengujian motor kutub bayangan ini cukup besar sesuai dengan motor induksi kelas D, dan torsi maksimal pada pengujian ini pada

pembacaan grafik yang ditampilkan dalam bentuk *figure* sebesar 4.155 Nm dengan kecepatan 595.3 Rpm, sementara pada perhitungan dihasilkan torsi maksimal sebesar 4.1554 Nm dan kecepatan sebesar 598.8125 Rpm. Hal ini membuktikan bahwa grafik sesuai dengan perhitungan torsi pada teorema thevenin. Gambar 4.9 menunjukkan torsi maksimal pada motor kutub bayangan yang diuji.



**Gambar 11.9** Grafik Pengujian Karakteristik Motor Kutub Bayangan

Tabel 11.7 menunjukkan data pengujian motor kutub bayangan yang disimpan dalam bentuk Ms.Excel, dari tabel tersebut data pengujian motor kutub bayangan dapat dianalisis dan disimpan sebagai perbedaan terhadap hasil pengujian dari motor lain atau hasil pengujian motor kutub bayangan dengan parameter yang berbeda.

**Tabel 11.7** Data Pengujian Motor Kutub Bayangan

<b>Parameter Variabel Motor</b>		<b>V</b>	110	<b>Volt</b>
		<b>R1</b>	1.86	<b>ohm</b>
		<b>X1</b>	2.56	<b>ohm</b>
		<b>R2</b>	3.56	<b>ohm</b>
		<b>X2</b>	2.56	<b>ohm</b>
		<b>Xm</b>	53.5	<b>ohm</b>
		<b>f</b>	60	<b>Hz</b>
		<b>p</b>	4	
		<b>Slip</b>	0.05	
<b>Hasil</b>	<b>Kondisi Start</b>	<b>Pin</b>	1139.798042	<b>Watt</b>
		<b>Pin</b>	1.527879413	<b>HP</b>

Kondisi Running	<b>Pout</b>	0	<b>Watt</b>
	<b>Pout</b>	0	<b>HP</b>
	<b>Ns</b>	1800	<b>Rpm</b>
	<b>Nr</b>	0	<b>Rpm</b>
	<b>I</b>	14.38721262	<b>Ampere</b>
	<b>T</b>	3.909328795	<b>Nw-m</b>
	<b>Effisiens i</b>	0	<b>%</b>
	<b>Pin</b>	157.5720965	<b>Watt</b>
	<b>Pin</b>	0.211222649	<b>HP</b>
	<b>Pout</b>	139.4645998	<b>Watt</b>
	<b>Pout</b>	0.186949866	<b>HP</b>
	Kondisi Maksimal	<b>Ns</b>	1800
<b>Nr</b>		1710	<b>Rpm</b>
<b>I</b>		1.43591983	<b>Ampere</b>
<b>Slip</b>		0.05	
<b>T</b>		0.778823875	<b>Nw-m</b>
<b>Effisiens i</b>		88.50843703	<b>%</b>
<b>Pin</b>		1081.696552	<b>Watt</b>
<b>Pin</b>		1.449995378	<b>HP</b>
<b>Pout</b>		260.5768863	<b>Watt</b>
<b>Pout</b>		0.349298775	<b>HP</b>
<b>Ns</b>		1800	<b>Rpm</b>
<b>Nr</b>		598.8125236	<b>Rpm</b>
<b>I</b>	12.11721759	<b>Ampere</b>	
<b>Slip</b>	0.667326376		
<b>T</b>	4.155434085	<b>Nw-m</b>	
<b>Effisiens i</b>	24.0896475	<b>%</b>	

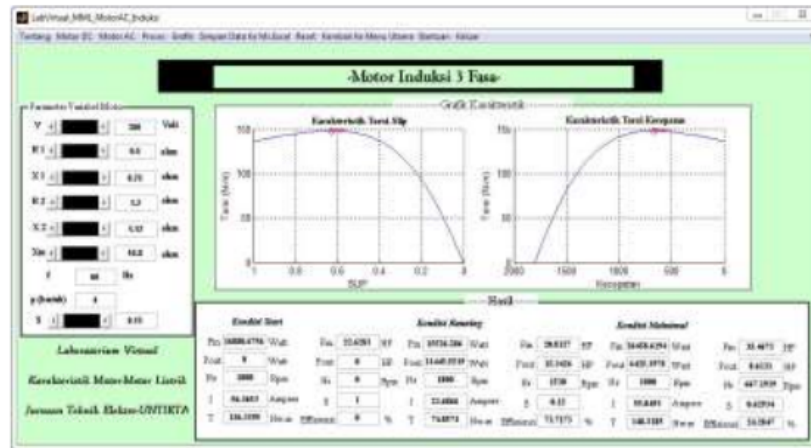
36

### C. GRAFIK HASIL PENGUJIAN MOTOR INDUKSI TIGA FASA

196

Pengujian ini dilakukan untuk mendapatkan karakteristik motor induksi tiga fasa yang mencakup karakteristik torsi terhadap slip dan karakteristik torsi terhadap kecepatan. Parameter pengujian motor induksi tiga fasa ini berdasarkan penelitian

yang telah dilakukan oleh Finayani dan Alhan dalam jurnal analisis karakteristik motor induksi NEMA D untuk melakukan penghematan energi.  $R_1 = 0.5$ ;  $X_1 = 0.75$ ;  $R_2 = 1.2$ ;  $X_2 = 1.12$ ;  $X_m = 16.8$ ;  $S = 0.15$  dan  $N_s = 1800$  RPM. Gambar tampilan pengujian motor induksi tiga fasa ditunjukkan pada Gambar 11.9.

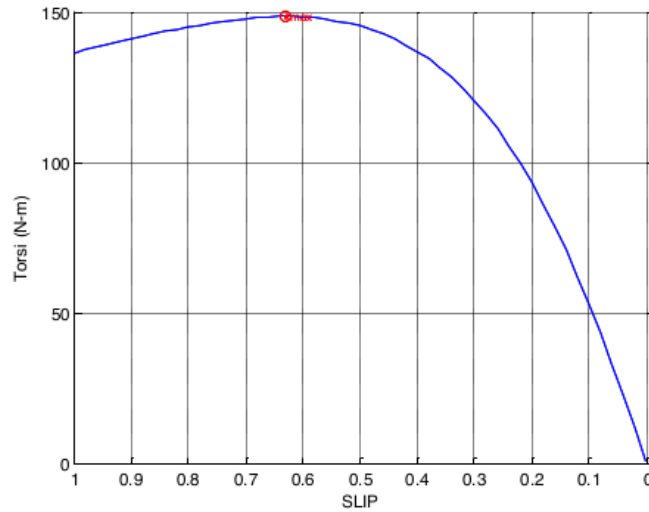


148

**Gambar 11.9** Pengujian Motor Induksi Tiga Fasa

Hasil pengujian motor induksi tiga fasa yang ditunjukkan pada Gambar 11.9 dapat dilihat bahwa dengan parameter yang telah ditentukan motor menghasilkan torsi 74.8574 Nm pada saat slip 0.15 dengan kecepatan rotor sebesar 1530 Rpm, sementara pada saat torsi maksimal dengan parameter motor yang sama torsi motor sebesar 148.3815 dengan kecepatan 667.1919 Rpm dan torsi awal motor sebesar 136.3195 Nm. Gambar 11.10 menunjukkan grafik hubungan antara torsi dan slip yang disimpan dalam bentuk *figure*, terlihat bahwa motor pada slip = 0,  $T = 0$  pada kecepatan sinkron nya yaitu kecepatan stator sama dengan kecepatan rotor.

Torsi awal motor induksi pada pengujian cukup besar hal ini dikarenakan motor induksi yang diuji adalah motor induksi tipe NEMA-D yang mempunyai karakteristik torsi awal tinggi, slip besar pada saat keadaan maksimal dan mempunyai efisiensi yang rendah. Pada penelitian yang dilakukan oleh Finayani dan Alhan slip maksimal pada motor ini sebesar 0.7, slip tersebut tidak dihitung melainkan melalui pembacaan grafik, pada pengujian laboratorium virtual dapat dilihat pada gambar 4.9 atau pada Gambar 4.10 grafik karakteristik torsi terhadap slip, slip yang dihasilkan motor adalah sebesar 0.62934 dan pada pembacaan grafik sebesar 0.6279 dan analisis dilakukan berdasarkan pada perhitungan bukan pada pembacaan grafik.



**Gambar 11.10** Karakteristik Torsi Terhadap Slip Motor Induksi Tiga Fasa

Sesuai dengan perhitungan pada persamaan pada (2-55) hingga persamaan (2-61) analisis perhitungan untuk membuktikan perhitungan torsi dan slip pada saat keadaan maksimal, sebagai berikut:

$$V_{TH} = \frac{X_M}{\sqrt{R_1^2 + (X_1 + X_M)^2}} V_p = \frac{16.8}{\sqrt{(0.5^2) + (0.75 + 16.8)^2}} \times 219.39 = 209.932 \text{ V}$$

$$Z_{TH} = \frac{jX_M(R_1 + jX_1)}{R_1 + j(X_1 + X_M)} = \frac{j16.8(0.5 + j0.75)}{0.5 + j(0.75 + 16.8)} = \frac{j8.4 - 12.6}{0.5 + j17.55} = 0.8753 \angle 57.956$$

$$= 0.457 + j 0.73 \Omega$$

$$\text{Slip maksimal} = \frac{R_2}{\sqrt{R_{th}^2 + (X_{th} + X_2)^2}} = \frac{1.2}{\sqrt{(0.457^2 + (0.73 + 1.12)^2)}} = 0,629$$

$$N_s = \frac{120 \times 60}{4} = 1800 \text{ Rpm}$$

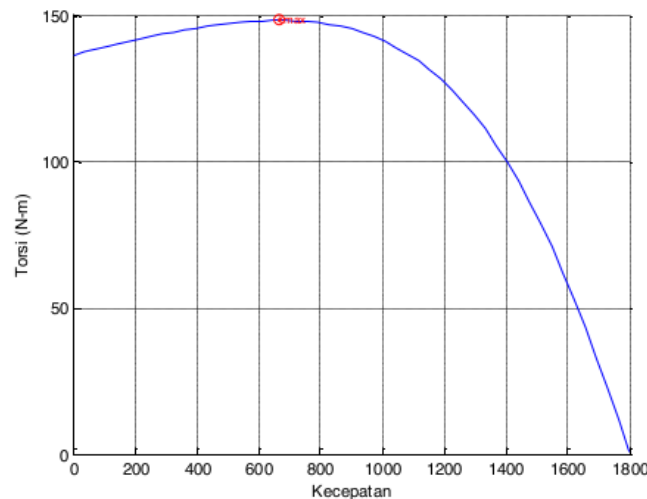
$$\omega_s = \frac{(N_s \times 2\pi)}{60} = \frac{1800 \times 6.28}{60} = 188.4$$

$$T = \frac{3V_{TH}^2 R_2 / s}{\omega_s [(R_{TH} + R_2 / s)^2 + (X_{TH} + X_2)^2]} = \frac{3 \times 209.932^2 \times 1.2 / 0.629}{188.4 \times ((0.457 + 1.2 / 0.629)^2 + (0.73 + 1.12)^2)} = 148.51 \text{ Nm}$$

Pada program Matlab  $T_{maks} = 148.3815$  (selisih disebabkan oleh ketelitian angka dibelakang koma).

Kecepatan motor pada motor induksi tiga fasa pada saat kondisi *start* adalah 1800 Rpm tetapi motor ini belum bergerak karena yang kecepatan tersebut adalah

kecepatan stator, sementara kecepatan rotor adalah nol karena slip motor adalah 1 atau motor dalam keadaan diam, pada kondisi maksimal motor adalah sebesar 1800 Rpm dan kecepatan pada rotor sebesar 667.19 Rpm, hal ini disebabkan slip pada keadaan maksimal sebesar 0.629 atau dapat dikatakan bahwa perbedaan kecepatan anantara kecepatan stator dan rotor adalah sebesar 62.9% karena slip yang cukup besar pengaruh beban motor dinaikkan. Torsi maksimal ini disebut torsi *pull-out* atau *breakdown torque* Gambar 11.11 menunjukkan kecepatan pada saat torsi maksimal adalah sebesar 667.2 Rpm pada saat torsi 148.3 (pembacaan grafik).



**Gambar 11.11** Grafik Kecepatan Terhadap Torsi Motor Induksi Tiga Fasa

Daya *output* pada pengujian motor ini sangat kecil dibanding dengan daya *input* pada motor, hal ini menyebabkan efisiensi motor menjadi sangat rendah tetapi sesuai dengan karakteristik motor induksi tiga fasa NEMA D bahwa motor mempunyai efisiensi yang rendah dibanding pada motor lain. Rugi-rugi yang diperhitungkan pada pengujian motor ini adalah rugi-rugi tembaga sehingga rugi-rugi tembaga akan mempengaruhi perhitungan pada daya keluar pada saat kondisi motor diam hingga pada saat keadaan maksimal, sementara pada penelitian sebelumnya hanya daya *input* yang dianalisa sementara daya *output* motor diabaikan. Pada penelitian sebelumnya mengambil nilai torsi konstan yang dilihat dari grafik 65 Nm sehingga didapatkan daya 10300 W pada slip 0.15. Analisis perhitungan daya pada laboratorium virtual untuk membuktikan daya *input* dan daya *output* pada motor induksi adalah sebagai berikut:

Pada kondisi *running* atau nominal dengan slip 0.15 arus maksimal motor:

$$I = \frac{V_{TH}}{[(R_{TH} + \frac{R_2}{s}) + j(X_{TH} + X_2)]} = \frac{209.932}{((0.457 + \frac{1.2}{0.629}) + j(0.73 + 1.12))} = 69.90 \angle -38.0539 \text{ A}$$

$$\cos \varphi = \cos -38.0539 = 0.787$$

Daya pada motor induksi tiga fasa pada keadaan maksimal:

$$P_{in} = 3 \times V \times I \times \cos \varphi = 3 \times 219.393 \times 69.9 \times 0.787 = 36207.28 \text{ W}$$

$$P_{cu} = 3 \times (I^2) \times \left( \frac{R_2}{slip} \right) = 3 \times 69.9^2 \times \left( \frac{1.2}{0.629} \right) = 27964.445 \text{ W}$$

$$P_{mek} = (1 - Slip) \times P_{cu} = (1 - 0.629) \times 27964.445 = 10374.8091 \text{ W}$$

Karena rugi-rugi lain diabaikan maka  $P_{mekanik} = P_{output}$  motor.

Data pengujian motor induksi tiga fasa dapat disimpan dalam bentuk Ms.Excel dengan cara menek **5** toolbar simpan data ke Ms.Excel pada tampilan halaman laboratorium virtual **motor induksi tiga fasa**, data yang disimpan **seperti ditunjukkan pada Tabel 11.8**.

**Tabel 11.8** Proses Sismpan Data Ke Ms. Excel

Parameter Variabel Motor		V	380	Volt
		R1	0.5	ohm
		X1	0.75	ohm
		R2	1.2	ohm
		X2	1.12	ohm
		Xm	16.8	ohm
		f	60	Hz
		p	4	
		Slip	0.15	
Hasil	Kondisi Start	Pin	37098.44843	Watt
		Pout	0	Watt
		Pout	0	HP
		Ns	1800	Rpm
		Nr	0	Rpm
		I	84.48474839	Ampere
		Slip	1	
		T	136.3195073	Nw-m
		Effisiens i	0	%

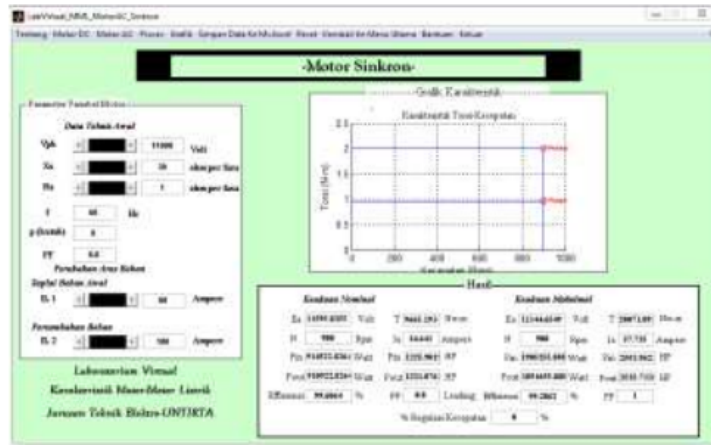


	<b>Kondisi Running</b>	<b>Pin</b>	15590.05005	<b>Watt</b>
		<b>Pout</b>	11993.74068	<b>Watt</b>
		<b>Pout</b>	16.07740038	<b>HP</b>
		<b>Ns</b>	1800	<b>Rpm</b>
		<b>Nr</b>	1530	<b>Rpm</b>
		<b>I</b>	24.24723623	<b>Ampere</b>
		<b>Slip</b>	0.15	
		<b>T</b>	74.8573771	<b>Nw-m</b>
		<b>Effisiensi</b>		
	<b>i</b>	76.93202169	%	
	<b>Kondisi Maksimal</b>	<b>Pin</b>	36232.28806	<b>Watt</b>
		<b>Pout</b>	10362.74429	<b>Watt</b>
		<b>Pout</b>	13.89107814	<b>HP</b>
		<b>Ns</b>	1800	<b>Rpm</b>
		<b>Nr</b>	667.1919251	<b>Rpm</b>
		<b>I</b>	69.90993813	<b>Ampere</b>
		<b>Slip</b>	0.629337819	
		<b>T</b>	148.3185197	<b>Nw-m</b>
<b>Effisiensi</b>				
<b>i</b>	28.60085533	%		

181

#### D. GRAFIK HASIL PENGUJIAN MOTOR SINKRON

Pengujian motor AC sinkron dilakukan untuk mengetahui karakteristik dari motor sinkron berupa torsi terhadap kecepatan, karena motor sinkron tidak mempunyai slip seperti motor AC induksi satu fasa dan motor induksi tiga fasa maka perlu diketahui unjuk kerja dari motor ini. Gambar 4.12 menunjukkan gambar pengujian dari motor sinkron.



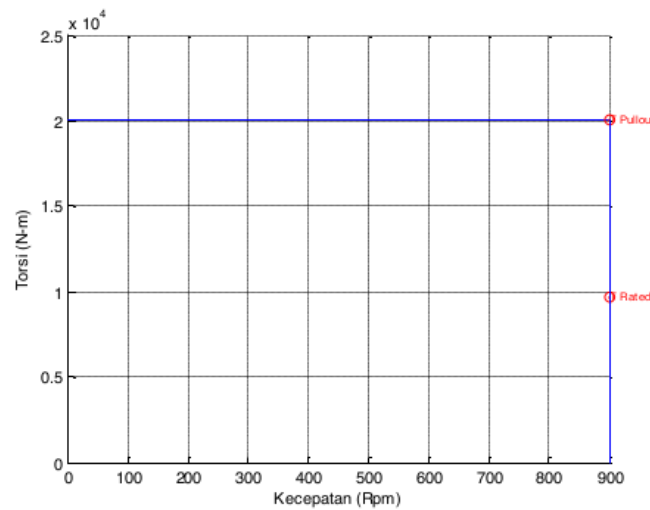
**Gambar 11.12** Pengujian Motor Sinkron

Hasil pengujian pada motor sinkron menunjukkan bahwa kecepatan motor sinkron konstan terhadap penambahan beban yang diberikan, sementara torsi beban semakin bertambah. Setiap penambahan beban torsi motor akan bertambah karena dipengaruhi penambahan nilai  $\sin \delta$  namun kecepatan sinkron motor tetap, jadi setiap penambahan beban sampai dibawah batas kekuatan torsi tertinggi medan putar stator, kecepatan akan tetap sehingga kurva torsi motor terhadap kecepatan seperti terlihat pada Gambar 11.12 atau lebih jelasnya dapat dilihat pada Gambar 11.13 dalam bentuk *figure*. Torsi maksimum atau *pull out-torque* terjadi saat  $\delta = 90$  derajat, dan torsi normalnya akan lebih kecil dari harga tersebut. Keadaan arus awal pada motor sinkron dipengaruhi oleh arus beban awal pada motor sebesar 60 A, maka arus awal motor dan tegangan jangkar pada motor tersebut sebesar:

$$I_A = \frac{I_L}{\sqrt{3}} = \frac{60}{\sqrt{3}} = 34.641 \text{ A}$$

$$E_A = V_{ph} - j.X_s I_A - (I_A R_A) = 11000 - (j30 \times 34.641) - (34.641 \times 1) \\ = 11596 - j852.17 = 11627.27 \angle - 4.203 \text{ V}$$

Pada grafik pengujian motor sinkron didapat torsi maksimal mencapai nilai 20071.9 Nm dengan kecepatan tetap 900 RPM, nilai yang didapat pada grafik dapat dibuktikan dengan perhitungan torsi motor sinkron pada persamaan (2-63) hingga pada persamaan (2-68).



**Gambar 11.13** Pengujian Motor Sinkron Tiga Fasa

### PERTANYAAN DAN SOAL

---

1. Berikan penjelasan dan interpretasi atas hasil pengujian karakteristik motor DC berdasarkan grafik visual yang ditampilkan menggunakan GUI MATLAB!!
2. Berikan penjelasan dan interpretasi atas hasil pengujian karakteristik motor AC berdasarkan grafik visual yang ditampilkan menggunakan GUI MATLAB!!
3. Berikan penjelasan dan interpretasi atas hasil pengujian karakteristik motor Sinkron berdasarkan grafik visual yang ditampilkan menggunakan GUI MATLAB!!
4. Jelaskan beberapa kesimpulan atas hasil tampilan grafik visual untuk pengujian sistem tenaga motor-motor listrik!!



## CONTOH – CONTOH LAIN PROGRAM MATLAB TENAGA LISTRIK

20

### TUJUAN INSTRUKSIONAL:

Setelah mempelajari bab ini, para mahasiswa diharapkan mampu memahami konsep lanjutan dan mengujicobakan beberapa contoh-contoh lainnya di bawah ini untuk diujicoba menggunakan pemrograman MATLAB.

### A. MEMBUAT GRAFIK KARAKTERISTIK TORSI-SLIP MOTOR INDUKSI 3 FASA

Sintaks program yang harus dituliskan pada editor MATLAB, dijelaskan berikut ini.

```

8 Nilai parameter-parameter dari motor
r1 = 0.60;           % nilai hambatan stator dalam ohm
x1 = 1.25;          % nilai reaktansi stator dalam ohm
8 = 0.40;           % nilai hambatan rotor dalam ohm
x2 = 1.15;          % nilai reaktansi rotor dalam ohm
rc = 55.8; xm = 13.6; % Resistansi dan reaktansi rangkaian
                    % penguat

v1 = 380 / sqrt(3); % Tegangan Phase Netral V1
v2 = 360 / sqrt(3); % nilai tegangan kedua
v3 = 340 / sqrt(3); % nilai tegangan ketiga
v4 = 320 / sqrt(3); % nilai tegangan keempat
v5 = 300 / sqrt(3); % nilai tegangan kelima
ns = 1500;          % Synchronous speed (8 min)= 120f/p
ws = (ns*2*pi)/50; % Synchronous speed (rad/s)

% Menghitung tegangan Thevenin dan impedansinya
vth1 = v1*rc*xm / sqrt((r1*rc-x1*xm)^2 + (rc*xm+r1*xm+rc*x1)^2);
vth2 = v2*rc*xm / sqrt((r1*rc-x1*xm)^2 + (rc*xm+r1*xm+rc*x1)^2);
    
```

```

vth3 = v3*rc*xm /sqrt((r1*rc-x1*xm)^2 + (rc*xm+r1*xm+rc*x1)^2);
vth4 = v4*rc*xm /sqrt((49*rc-x1*xm)^2 + (rc*xm+r1*xm+rc*x1)^2);
vth5 = v5*rc*xm /sqrt((r1*rc-x1*xm)^2 + (rc*xm+r1*xm+rc*x1)^2);
116 = ((j*xm*r1*rc)-(rc*xm*x1)) /((r1*rc-x1*xm)+j*(rc*xm+r1*xm+rc*x1));
rth = real(zth); xth = imag(zth);

```

```

s = (0:1:5000)/5000;
s(1) = 0.001;
nm = (1-s) * ns;

```

99 Menentukan Torsi untuk menggambarkan karakteristik Torsi-Slip  
for i= 1:5001

```

t(i)=(3*vth1^2*63/s(i))/(ws*((rth+r2/s(i))^2+(xth+x2)^2));
t2(i)=(3*vth2^2*(r2)/s(i))/(ws*((rth+(r2)/s(i))^2+(xth+x2)^2));
t3(i)=(3*vth3^2*(r2)/s(i))/(ws*((rth+(r2)/s(i))^2+(xth+x2)^2));
t4(i)=(3*vth4^2*(r2)/s(i))/(ws*((rth+(r2)/s(i))^2+(xth+x2)^2));
t5(i)=(3*vth5^2*(r2)/s(i))/(ws*((rth+(r2)/s(i))^2+(xth+x2)^2));
end

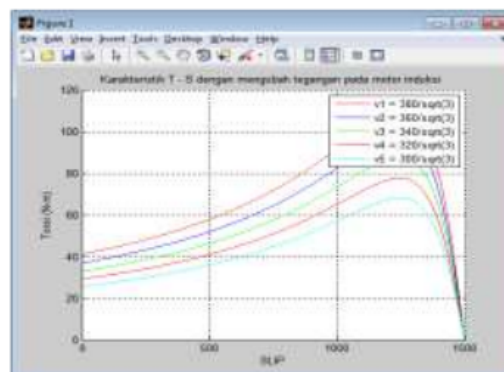
```

```

% Menggambar karakteristik Torsi-Slip untuk tiga variasi tegangan
figure(1);
hold on;
plot (nm,t,'r-',nm,t2,'b-',nm,t3,'g-',nm,t4,'r-',nm,t5,'c-');
title ('Karakteristi 125- S dengan mengubah tegangan pada motor induksi ');
legend ('v1 = 380/sqrt(3)', 'v2 = 360/sqrt(3)', 'v3 = 340/sqrt(3)', 'v4 = 320/sqrt(3)', 'v5 = 300/sqrt(3)');
xlabel ('SLIP');
ylabel ('Torsi (Nm)');
grid on;

```

Grafik yang akan ditampilkan seperti pada gambar berikut.



**Gambar 12.1** Grafik Fungsi Torsi-Slip Motor Induksi 3 Fasa

## B. PERHITUNGAN ARUS GANGGUAN PADA SISTEM TENAGA LISTRIK

Sintaks program berikut dapat diujicobakan menggunakan MATLAB dengan cara menuliskannya pada editor MATLAB dan menjalankannya. Pada command windows MATLAB dapat anda lihat tampilan hasil seperti gambar 12.2.

```
clc
disp(' *****
175 disp(' *****Perhitungan Arus Gangguan Pada Sistem Tenaga Listrik *****
disp(' *****
disp('press any key to continue.....')
pause;
% Matriks transformasi komponen simetris :
40 -0.5+j*0.866;
A = [1 1 1; 1 a^2 a; 1 a a^2];
% Data Sistem
disp(' Masukan Data XG,1,2,3 XT,1,2,3,XL ,1,2,3')
V3_a_0 = input (' Masukan Data Variabel V3_a_0 =');
Zf = input (' Masukan Data Variabel Zf =');
XG_reaktor = input (' Masukan Data XG1_reaktor =');
XG1_1 = input (' Masukan Data Variabel XG1_1 =');
XG1_2 = input (' Masukan Data Variabel XG1_2 =');
XG1_0 = input (' Masukan Data Variabel XG1_0 =');
XG2_1 = input (' Masukan Data Variabel XG2_1 =');
XG2_2 = input (' Masukan Data Variabel XG2_2 =');
XG2_0 = input (' Masukan Data Variabel XG2_0 =');
XT1_1 = input (' Masukan Data Variabel XT1_1 =');
XT1_2 = input (' Masukan Data Variabel XT1_2 =');
XT1_0 = input (' Masukan Data Variabel XT1_0 =');
XT2_1 = input (' Masukan Data Variabel XT2_1 =');
XT2_2 = input (' Masukan Data Variabel XT2_2 =');
XT2_0 = input (' Masukan Data Variabel XT2_0 =');
XL12_1 = input (' Masukan Data Variabel XL12_1 =');
XL12_2 = input (' Masukan Data Variabel XL12_2 =');
XL12_0 = input (' Masukan Data Variabel XL12_0 =');
XL13_1 = input (' Masukan Data Variabel XL13_1 =');
XL13_2 = input (' Masukan Data Variabel XL13_2 =');
XL13_0 = input (' Masukan Data Variabel XL13_0 =');
```

```

XL23_1 = input (' Masukan Data Variabel XL23_1 =');
XL23_2 = input (' Masukan Data Variabel XL23_2 =');
XL23_0 = input (' Masukan Data Variabel XL23_0 =');

```

```
%Daftar Menu
```

```

MENU =[
    ' Pilih Jenis Gangguan :                               ';
    '   1. Hitung Arus Gangguan 3 Fasa Seimbang             ';
    '   2. Hitung Arus Gangguan 1 Fasa Ke Tanah            ';
    '   3. Hitung Arus Gangguan 1 phasa ke phasa           ';
    '   4. Hitung Arus Gangguan 2 phasa Ke Tanah          ';
    '   5. Keluar                                           '];
input_num = input('Pilih Jenis Gangguan [1..5]: ');
while (input_num > 0)&(input_num < 5),
    if input_num==1;
        clc;
        disp('')
        disp('!=====!!')
        disp('!== ARUS GANGGUAN TIGA FASA SEIMBANG PADA BUS 3 ==!')
        disp('!-----!')
        disp('!          I3_a          !')
        disp('!          (per unit)      !')
        disp('!=====!!')
        fprintf('          '),disp([I3_a_F])
        disp('Tekan ENTER Untuk Melanjutkan Perhitungan Gangguan...');
        pause;
        clc;
        fprintf('');disp([MENU]);
        input_num = input('Pilih Jenis Gangguan [1..5]: ');
    end;
    if input_num==2;
        clc;
        disp('!=====!!')
        disp('!== ARUS GANGGUAN SATU PHASA KE TANAH PADA BUS 3 ==!')
        disp('!-----!')

```



```

disp('! I3_a      ! I3_b      ! I3_c  !')
disp('! (per unit)      ! (per unit)      ! (per unit)  !')
disp('!===== !')
fprintf('      ');disp([I3_a,I3_b,I3_c])
disp('Tekan ENTER Untuk Melanjutkan Perhitungan Gangguan...');
pause;
clc;
fprintf('      ');disp([MENU]);

input_num = input('Pilih Jenis Gangguan [1..5]: ');

end;
if input_num==3;
    clc;
    disp('!===== !')
    disp('! ARUS GANGGUAN SATU PHASA KE PHASA PADA BUS 3  !')
    disp('!----- !')
    disp('! I3_a      ! I3_b      ! I3_c  !')
    disp('! (per unit)      ! (per unit)      ! (per unit)  !')
    disp('!===== !')
    fprintf('      '),disp([I3_a,I3_b,I3_c])
    disp('Tekan ENTER Untuk Melanjutkan Perhitungan Gangguan...');
    pause;
    clc;
    fprintf('      ');disp([MENU]);
    input_num = input('Pilih Jenis Gangguan [1..5]: ');

end;
if input_num==4;
    clc;
    disp('!===== !')
    disp('! ARUS GANGGUAN DUA PHASA KE TANAH PADA BUS 3  !')
    disp('!----- !')
    disp('! I3_a      ! I3_b      sudut ! I3_c      Sudut !I3(F)      Sudut !')
    disp('! (per unit)      !(per unit) (drajat)! (perunit) (drajat) ! (perunit)(derajat)!')
    disp('!===== !')

```

```

fprintf('!%6.0f      ! %7.3f %7.3f      !%7.3f %4.3f ! %8.3f %4.3f
!\n',I3_a,I3_b,theta2,I3_c,theta3,I3F,theta)
disp('!===== !')
disp('Tekan ENTER Untuk Melanjutkan Perhitungan Gangguan...');
pause;
clc;
fprintf('      ');disp([MENU]);
input_num = input('Pilih Jenis Gangguan [1..5]: ');
end;
end

%('Impedansi Transformasi Delta ke Y ditinjau dari bus 3 untuk urutan');
%('positif dan negatif ');
Z1s_pn = (XL12_1*XL13_1)/(XL12_1+XL13_1+XL23_1);
Z2s_pn = (XL12_1*XL23_1)/(XL12_1+XL13_1+XL23_1);
Z3s_pn = (XL13_1*XL23_1)/(XL12_1+XL13_1+XL23_1);
% ('Impedansi Urutan Positif dan Negatif');
Zs1_pn = XG1_1+XT1_1+Z1s_pn;
Zs2_pn = XG2_1+XT2_1+Z2s_pn;
Zpn = ((Zs1_pn*Zs2_pn)/(Zs1_pn+Zs2_pn))+Z3s_pn;
Z33_1 = Zpn;
Z33_2 = Z33_1;
%('Impedansi Transformasi Delta ke Y ditinjau dari bus 3 untuk urutan nol ');
Z1s_0 = (XL12_0*XL13_0)/(XL12_0+XL13_0+XL23_0);
Z2s_0 = (XL12_0*XL23_0)/(XL12_0+XL13_0+XL23_0);
Z3s_0 = (XL13_0*XL23_0)/(XL12_0+XL13_0+XL23_0);
%('Impedansi Urutan NOL');
Zs1_0 = XG_reaktor+XG1_0+XT1_0+Z1s_0;
Zs2_0 = XT2_0+Z2s_0;
Z33_0 = ((Zs1_0*Zs2_0)/(Zs1_0+Zs2_0))+Z3s_0;
% *****
% ***** PERHITUNGAN ARUS GANGGUAN *****
% *****
% Perhitungan Arus Gangguan tiga fasa seimbang pada bus 3 :
I3_a_F = V3_a_0/(Z33_1+Zf);
disp('      ')
disp(' !===== !')

```

```

disp('!=== ARUS GANGGUAN TIGA FASA SEIMBANG PADA BUS 3
====!')
disp('!-----!')
disp('!          I3_a          !')
disp('!          (per unit)      !')
disp('!=====!!')
fprintf('          '),disp([I3_a_F]);

```

**% Perhitungan Arus Gangguan Satu Fasa ke Tanah pada bus 3 :**

```

I3_0 = V3_a_0/(Z33_1+Z33_2+Z33_0+3*Zf);
I3_1 = I3_0;
I3_2 = I3_0;
I = A*[I3_0;I3_0;I3_0];
I3_a = I(1);
I3_b = I(2);
I3_c = I(3);

```

```

disp('!=====!!')
disp('!===ARUS GANGGUAN SATU PHASA KE TANAH PADA BUS 3 ===!!')
disp('!-----!')
disp('!  I3_a      !  I3_b      !  I3_c      !')
disp('! (per unit) ! (per unit) ! (per unit) !')
disp('!=====!!')
fprintf('          ');disp([I3_a,I3_b,I3_c])

```

**% Perhitungan Arus Gangguan Satu Fasa ke Fasa pada bus 3 :**

```

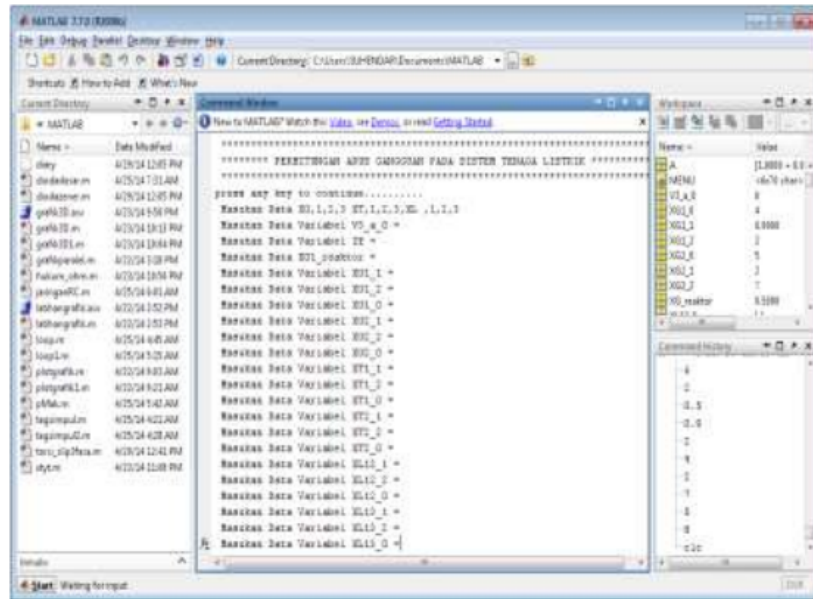
I3_0 = 0;
I3_1 = V3_a_0/(Z33_1+Z33_2+Zf);
I3_2 = -I3_1;
I = A*[I3_0; I3_1; I3_2];
I3_a = I(1);
I3_b = I(2);
I3_c = I(3);
disp('!=====!!')
disp('!  ARUS GANGGUAN SATU PHASA KE PHASA PADA BUS 3
!')
disp('!-----!')

```

```

disp('! I3_a      ! I3_b      ! I3_c      !')
disp('! (per unit)      ! (per unit)      ! (per unit)      !')
disp('!=====!!')
fprintf('          '),disp([I3_a,I3_b,I3_c])
% Perhitungan Arus Gangguan Dua Fasa ke Tanah pada bus 3 :
I3_1 = V3_a_0/(((Z33_2*(Z33_0+3*Zf))/(Z33_2+Z33_0+3*Zf))+Z33_1);
I3_2 = -(V3_a_0-Z33_1*I3_1)/(Z33_0+3*Zf);
% Arus setiap Fasa :
I = A*[I3_0 I3_1;I3_2];
I3_a = abs(I(1));
I3_b = abs(I(2));
I3_c = abs(I(3));
theta1 = 180/pi*atan2(imag(I(1)),real(I(1)));
theta2 = 180/pi*atan2(imag(I(2)),real(I(2)));
theta3 = 180/pi*atan2(imag(I(3)),real(I(3)));
% Arus Gangguan :
I3_F = I(2) + I(3);
I3F = abs(I3_F);
theta = 180/pi*atan2(imag(I3_F),real(I3_F));
disp('!=====!!')
disp('!          ARUS GANGGUAN DUA PHASA KE TANAH PADA BUS 3
!')
disp('!-----!!')
disp('! I3_a      ! I3_b      sudut ! I3_c      Sudut ! I3(F)      Sudut !')
disp('! (per unit)      !(per unit) (drajat)! (perunit) (drajat) ! (perunit)(derajat)!!')
disp('!=====!!')
fprintf('!%6.0f      ! %7.3f %7.3f      !%7.3f %4.3f ! %8.3f %4.3f
!\n',I3_a,I3_b,theta2,I3_c,theta3,I3F,theta)
disp('!=====!!')

```



**Gambar 12.2** Tampilan Menu Command Windows MATLAB untuk Perhitungan Arus Gangguan Pada Sistem Tenaga Listrik

### C. PERHITUNGAN ARUS SALURAN MENGGUNAKAN ANALIS MESH DAN KOMPONEN SIMETRIS

```

% =====
% = Perhitungan Arus Saluran Dengan Menggunakan Analisis Mesh =%
% ==                               Dan                               ==
% ==                               Komponen Simetris                               ==
% =====
% Matriks transformasi komponen simetris :
a = -0.5+j*0.866;
% data tegangan dari fasa ke fasa :
VL = sqrt(3) *VLN;
% data reaktansi seri per fasa dan reaktansi kopling bersama antara fasa :
Zs = j*12;
Zm = j*4;

% perhitungan arus saluran dengan menggunakan analisa mesh :
Zmesh = [(Zs-Zm) -(Zs-Zm) 0
          0 (Zs-Zm) -(Zs-Zm)
          1 1 1];

```

```

Vmesh = [VL*cos(pi/6)+j*VL*sin(pi/6)
          VL*cos(-pi/2)+j*VL*sin(-pi/2)
          0];
Ymesh = inv(Zmesh);
Iabc = Ymesh*Vmesh;
Iabcm = abs(Iabc);
sudut = angle(Iabc)*180/pi;
disp('')
disp('!=====!')
disp(! ARUS SALURAN DENGAN PERHITUNGAN !)
disp(! ANALISA MESH !)
disp(!-----!)
disp(! Iabc Sudut !)
disp(! (Ampere) (derajat) !)
disp('!=====!')
fprintf('!%11.3f %7.0f !\n',Iabcm(1),sudut(1))
fprintf('!%11.3f %7.0f !\n',Iabcm(2),sudut(2))
fprintf('!%11.3f %7.0f !\n',Iabcm(3),sudut(3))
disp('!=====!')
% Perhitungan arus saluran dengan menggunakan komponen simetris :
V012 = [0; VLN; 0];
Z012 = [Zs+2*Zm 0 0
        0 Zs-Zm 0
        0 0 Zs-Zm];
I012 = inv(Z012)*V012;
Iabc = A*I012;
Iabck = abs(Iabc);
sudut = angle(Iabc)*180/pi;
disp(! ARUS SALURAN DENGAN PERHITUNGAN !)
disp(! KOMPONEN SIMETRIS !)
disp(!-----!)
disp(! Iabc sudut !)
disp(! (Ampere) (derajat) !)
disp('!=====!')
fprintf('!%11.3f %7.0f !\n',Iabck(1),sudut(1))
fprintf('!%11.3f %7.0f !\n',Iabck(2),sudut(2))
fprintf('!%11.3f %7.0f !\n',Iabck(3),sudut(3))

```

```
disp('!=====!')
```

maka jika program tersebut dijalankan, pada command windowsnya MATLAB akan ditampilkan hasil sebagai berikut.

```
!=====!  
! ARUS SALURAN DENGAN PERHITUNGAN !  
!   ANALISA MESH   !  
!-----!  
!  Iabc   Sudut   !  
!      (Ampere) (derajat) !  
!=====!  
! 12.500   -90   !  
! 12.500   150   !  
! 12.500   30    !  
!=====!  
! ARUS SALURAN DENGAN PERHITUNGAN !  
!   KOMPONEN SIMETRIS   !  
!-----!  
!  Iabc   sudut   !  
! (Ampere) (derajat) !  
!=====!  
! 12.500   -90   !  
! 12.499   150   !  
! 12.500   30    !  
!=====!
```

#### **D. MENENTUKAN DAYA OPTIMAL DARI STASIUN PEMBANGKIT DAN BIAYA TOTAL BAHAN BAKAR DENGAN MEMPERHITUNGGAN RUGI-RUGI DAYA PADA SISTEM**

```
disp('!=====!')  
Disp('!   Menentukan Daya Optimal Dari Stasiun Pembangkit   !')  
Disp('!           Dan Biaya Total Bahan Bakar           !')  
Disp('! Dengan Memperhitungkan Rugi-Rugi Daya Pada Sistem !')  
disp('!=====!')  
epsilon=0.0001;  
alpha=[200;180;140];
```

```

beta=[7;6.3;6.8];
gamma=[0.008;0.009;0.007];
gamma_23=[0.009;0.007];
gamma_13=[0.008;0.007];
gamma_12=[0.008;0.009];
PD=150;
delta_P=0.001;
lambda=8;
disp('!_____!')
disp('!      !      !      !      !      !')
disp('! Iterasi ! lambda ! P1 ! P2 ! P3 !')
disp('!      ! ($/MW-jam) ! (MW) ! (MW) ! (MW) !')
disp('!_____!')

iterasi=0;
while abs(delta_P)>=epsilon

% daya dari masing-masing stasiun pembangkit:
P1=(lambda-beta(1))/(2*(gamma(1)+lambda*0.000218));
P2=(lambda-beta(2))/(2*(gamma(2)+lambda*0.000228));
P3=(lambda-beta(3))/(2*(gamma(3)+lambda*0.000179));

% rugi-rugi daya nyata pada sistem :
PL=0.000218*(P1)^2+0.000228*(P2)^2+0.000179*(P3)^2;
delta_P=PD+PL-(P1+P2+P3);
J1=(gamma(1)+0.000218*beta(1))/(2*(gamma(1)+lambda*0.000218)^2);
J2=(gamma(2)+0.000228*beta(2))/(2*(gamma(2)+lambda*0.000228)^2);
J3=(gamma(3)+0.000179*beta(3))/(2*(gamma(3)+lambda*0.000179)^2);
JT=J1+J2+J3;
delta_lambda=(delta_P)/JT;
lambda=delta_lambda+lambda;

%=====
J_23_1=(gamma_23(1)+0.000228*beta(2))/(2*(gamma_23(1)+lambda*0.000228)^2);
J_23_2=(gamma_23(2)+0.000179*beta(3))/(2*(gamma_23(2)+lambda*0.000179)^2);

```



```

J_T_23=J_23_1+J_23_2;
%=====
J_13_1=(gamma_13(1)+0.000218*beta(1))/(2*(gamma_13(1)+lambda*0.000218)^
2);
J_13_2=(gamma_13(2)+0.000179*beta(3))/(2*(gamma_13(2)+lambda*0.000179)^
2);
J_T_13=J_13_1+J_13_2;
%=====
J_12_1=(gamma_12(1)+0.000218*beta(1))/(2*(gamma_12(1)+lambda*0.000218)^
2);
J_12_2=(gamma_12(1)+0.000228*beta(2))/(2*(gamma_12(1)+lambda*0.000228)^
2);
J_T_12=J_12_1+J_12_2;
%=====

if P1>85
    P1=85;
    PL=0.000218*(P1)^2+0.000228*(P2)^2+0.000179*(P3)^2;
    delta_P=PD+PL-(P1+P2+P3);
    delta_lambda=(delta_P)/J_T_23;

    lambda=lambda+delta_lambda;
    P2=(lambda-beta(2))/(2*gamma(2));
    P3=(lambda-beta(3))/(2*gamma(3));

else if P1<10
    P1=10;
    PL=0.000218*(P1)^2+0.000228*(P2)^2+0.000179*(P3)^2;
    delta_P=PD+PL-(P1+P2+P3);
    delta_lambda=(delta_P)/J_T_23;
    lambda=lambda+delta_lambda;
    P2=(lambda-beta(2))/(2*gamma(2));
    P3=(lambda-beta(3))/(2*gamma(3));
else if P2<10
    P2=10;
    PL=0.000218*(P1)^2+0.000228*(P2)^2+0.000179*(P3)^2;
    delta_P=PD+PL-(P1+P2+P3);

```

```

    delta_lambda=(delta_P)/J_T_13;
    lambda=lambda+delta_lambda;
    P1=(lambda-beta(1))/(2*gamma(1));
    P3=(lambda-beta(3))/(2*gamma(3));
else if P2>80
    P2=80;
    PL=0.000218*(P1)^2+0.000228*(P2)^2+0.000179*(P3)^2;
    delta_P=PD+PL-(P1+P2+P3);
    delta_lambda=(delta_P)/J_T_13;
    lambda=lambda+delta_lambda;
    P1=(lambda-beta(1))/(2*gamma(1));
    P3=(lambda-beta(3))/(2*gamma(3));
    else if P3<10
        P3=10;
21 PL=0.000218*(P1)^2+0.000228*(P2)^2+0.000179*(P3)^2;
        delta_P=PD+PL-(P1+P2+P3);
        delta_lambda=(delta_P)/J_T_12;
        lambda=lambda+delta_lambda;
        P1=(lambda-beta(1))/(2*gamma(1));
        P2=(lambda-beta(3))/(2*gamma(2));
    else if P3>70
        P3=70;
21 PL=0.000218*(P1)^2+0.000228*(P2)^2+0.000179*(P3)^2;
        delta_P=PD+PL-(P1+P2+P3);
        delta_lambda=(delta_P)/J_T_12;
        lambda=lambda+delta_lambda;
        P1=(lambda-beta(1))/(2*gamma(1));
        P2=(lambda-beta(3))/(2*gamma(2));
    end
end
end
end
end
end
iterasi=iterasi+1;
fprintf('%5.0f ! %8.4f !%9.4f !%9.4f !%10f !\n',iterasi,lambda,P1,P2,P3)

```

```

end
disp('!_____!')
%biaya total bahan bakar :
Ct=sum(alpha)+sum([P1 P2 P3]*beta)+sum([(P1)^2 (P2)^2 (P3)^2]*gamma);
disp(' ')
disp(['! BIAYA TOTAL BAHAN BAKAR=' num2str(Ct), ' ', '$/jam  '])
disp('!_____!')

```

maka jika program tersebut dijalankan, pada command windowsnya MATLAB akan ditampilkan hasil sebagai berikut.

```

!=====!
!   MENENTUKAN DAYA OPTIMAL DARI STASIUN
PEMBANGKIT   !
!           DAN BIAYA TOTAL BAHAN BAKAR           !
!   DENGAN MEMPERHITUNGKAN RUGI-RUGI DAYA PADA
SISTEM   !
!=====!
!_____!
!   !   !   !   !   !
! Iterasi ! lambda ! P1 ! P2 ! P3 !
!       ! ($/MW-jam) ! (MW) ! (MW) ! (MW) !
!_____!
! 1 ! 7.2169 ! 13.5582 ! 23.1629 ! 70.000000 !
! 2 ! 7.6671 ! 11.3300 ! 43.0668 ! 25.141109 !
! 3 ! 7.6787 ! 34.4885 ! 63.5976 ! 51.783584 !
! 4 ! 7.6789 ! 35.0781 ! 64.1206 ! 52.462251 !
! 5 ! 7.6789 ! 35.0904 ! 64.1315 ! 52.476377 !
! 6 ! 7.6789 ! 35.0907 ! 64.1317 ! 52.476670 !
!_____!
!           BIAYA TOTAL BAHAN BAKAR=1592.6494 $/jam
!_____!

```

## PERTANYAAN DAN SOAL

Berikan penjelasan untuk masing-masing fungsi baris program pada sub bab 12.1, 12.2, 12.3, dan 12.4 di atas

## DAFTAR PUSTAKA

---

Anoname. *Pedoman Pengantar Mengenai Programmable Logic controller*. Mitsubishi

AT, Sobirirn & Suhendar (2003). *Teori Dan Praktek Programmable Logic Control*. PT. Bukaka Teknik Utama, Bogor Jawa Barat

AT, Sobirin & Suhendar (2005). *Modul Praktek Programmable Logic Control*. PT. Bukaka Teknik Utama, Bogor Jawa Barat

Budyanto, M & Wijaya, A. (2003). *Pengenalan dasar-Dasar Programmable Logic Controller (Disertai Contoh Aplikasinya)*. Gava media, Yogyakarta.

Chapallaz, J.M, dkk. (1992). *Manual On Induction Motor Used As Generator*. GTZ, Germany

Factory Automation Division. (2002). *Pengenalan Programmable Logic Control (Sysmac C-Series)*. PT. Mandala Sumber Rejeki, Yogyakarta

Herwin, Suhendar, Rocky Alfan. (2013). *Rancang Bangun Teknik Antarmuka Jaringan Syaraf Listrik Dengan Modul Praktikum Kendali Proses*. Laporan Penelitian. Untirta. Banten

Link, Wolfgang, 2002, *Pengukuran, Pengendalian, dan Pengaturan dengan PC (Praktikum Otomasi dengan Pengaturan Numerik)*, Elex Media Komputindo, Jakarta

Muamar, Ahmad, 2004, *Sistem Kontrol I/O dan Kontrol Suara*, Andi Yogyakarta

Maulana, Fadjar, Nana, Yossy & Suhendar. (2012). *Desain Dan Aplikasi Program Berbasis Programmable Logic Controller Untuk Praktikum Sistem Kendali Proses*. Laporan Praktek Lapangan Indsutri. Untira Banten

Nurul Hidayat & Suhendar. (2012). *Sistem Kontrol Industri (Kendali On-Off Valve Pada Tangki Distribusi Air)*. Laporan Praktek lapangan Industri. UNISMA Bekasi

- Omron. (1999). *Omron User's Manual (position Control Unit For Sysmac C-Series)*. Omron Corporation.
- Pakpahan, Sahat. (1994). *Kontrol Otomatik (Teori Dan Penerapan)*. Erlangga, Jakarta.
- Petruzela, Frank D. (2004). *Industrial Electronic*, Prentice-Hall. Inc.
- Suhendar & angoro. (2012). <sup>109</sup> *Rancang Bangun Perangkat Simulasi Berbasis Jaringan Saraf Listrik Sebagai Virtual Praktikum Sistem Kendali Proses. Laporan Penelitian Dosen Madya*. Universitas Sultan Ageng Tirtayasa (UNTIRTA) Banten.
- Suhendar & Irfanuddin, Fajar. (2012). *Desain Dan Aplikasi Program Berbasis Programmable Logic Controller Untuk Praktikum Sistem Kendali Proses. Laporan. Penelitian Dosen Madya*. Universitas Sultan Ageng Tirtayasa (UNTIRTA) Banten.
- Suhendar. (2010). *Sistem Penggerak Indsutri dan Penggunaan Motor-Motor Listrik*. Teknik Elektro, Universitas Sultan Ageng Tirtayasa (UNTIRTA) Banten.
- Sumanto. (1996). *Motor-Motor Listrik*. Andi, Yogyakarta.
- Yahya, Zakir. (1996). *Kontrol Motor Induksi*. FPTK IKIP Padang, Padang

# KOMPUTASI ELEKTRONIK & TENAGA LISTRIK

## ORIGINALITY REPORT

23%

SIMILARITY INDEX

22%

INTERNET SOURCES

6%

PUBLICATIONS

8%

STUDENT PAPERS

## PRIMARY SOURCES

1	<a href="http://slideplayer.info">slideplayer.info</a> Internet Source	1%
2	<a href="http://repository.its.ac.id">repository.its.ac.id</a> Internet Source	1%
3	<a href="http://eprints.polsri.ac.id">eprints.polsri.ac.id</a> Internet Source	1%
4	<a href="http://dokumen.tips">dokumen.tips</a> Internet Source	1%
5	<a href="http://text-id.123dok.com">text-id.123dok.com</a> Internet Source	1%
6	<a href="http://17until22pm.blogspot.com">17until22pm.blogspot.com</a> Internet Source	<1%
7	<a href="http://www.math.tamu.edu">www.math.tamu.edu</a> Internet Source	<1%
8	<a href="http://pt.scribd.com">pt.scribd.com</a> Internet Source	<1%
9	<a href="http://ilmukomputer.org">ilmukomputer.org</a> Internet Source	<1%
10	<a href="http://vdocuments.site">vdocuments.site</a> Internet Source	<1%

11	<a href="https://coggle.it">coggle.it</a> Internet Source	<1 %
12	<a href="https://www.coursehero.com">www.coursehero.com</a> Internet Source	<1 %
13	<a href="https://sed142b1e1a29c1ce.jimcontent.com">sed142b1e1a29c1ce.jimcontent.com</a> Internet Source	<1 %
14	<a href="https://adoc.tips">adoc.tips</a> Internet Source	<1 %
15	<a href="https://123dok.com">123dok.com</a> Internet Source	<1 %
16	<a href="https://ebin.pub">ebin.pub</a> Internet Source	<1 %
17	<a href="https://docslide.us">docslide.us</a> Internet Source	<1 %
18	<a href="https://ml.scribd.com">ml.scribd.com</a> Internet Source	<1 %
19	<a href="https://repositori.kemdikbud.go.id">repositori.kemdikbud.go.id</a> Internet Source	<1 %
20	<a href="https://staff.uny.ac.id">staff.uny.ac.id</a> Internet Source	<1 %
21	<a href="https://wreforum.org">wreforum.org</a> Internet Source	<1 %
22	<a href="https://ocw.kfupm.edu.sa">ocw.kfupm.edu.sa</a> Internet Source	<1 %
23	<a href="https://rahmadya.com">rahmadya.com</a> Internet Source	<1 %

---

24	<a href="http://repositori.usu.ac.id">repositori.usu.ac.id</a> Internet Source	<1 %
25	<a href="http://afrileya.blogspot.com">afrileya.blogspot.com</a> Internet Source	<1 %
26	<a href="http://ictsmpcenter.blogspot.com">ictsmpcenter.blogspot.com</a> Internet Source	<1 %
27	<a href="http://es.scribd.com">es.scribd.com</a> Internet Source	<1 %
28	<a href="http://apriyantokharisma.blogspot.com">apriyantokharisma.blogspot.com</a> Internet Source	<1 %
29	<a href="http://aripradani.blogspot.com">aripradani.blogspot.com</a> Internet Source	<1 %
30	<a href="http://www.docme.ru">www.docme.ru</a> Internet Source	<1 %
31	<a href="http://followscience.com">followscience.com</a> Internet Source	<1 %
32	<a href="http://www.grdfp.polymtl.ca">www.grdfp.polymtl.ca</a> Internet Source	<1 %
33	Submitted to Fakultas Ekonomi Universitas Indonesia Student Paper	<1 %
34	<a href="http://edoc.pub">edoc.pub</a> Internet Source	<1 %
35	<a href="http://kupdf.net">kupdf.net</a> Internet Source	<1 %

---

[ejournal.uhn.ac.id](http://ejournal.uhn.ac.id)



36

Internet Source

&lt;1 %

37

[repository.uhn.ac.id](https://repository.uhn.ac.id)

Internet Source

&lt;1 %

38

[edoc.site](https://edoc.site)

Internet Source

&lt;1 %

39

[desmarianasiringoringo.blogspot.com](https://desmarianasiringoringo.blogspot.com)

Internet Source

&lt;1 %

40

[www.ece.umd.edu](http://www.ece.umd.edu)

Internet Source

&lt;1 %

41

[www.docstoc.com](http://www.docstoc.com)

Internet Source

&lt;1 %

42

[kailuccanoenoe.blogspot.com](https://kailuccanoenoe.blogspot.com)

Internet Source

&lt;1 %

43

[docplayer.com.br](https://docplayer.com.br)

Internet Source

&lt;1 %

44

Submitted to iGroup

Student Paper

&lt;1 %

45

[eprints.itn.ac.id](https://eprints.itn.ac.id)

Internet Source

&lt;1 %

46

[eprints.untirta.ac.id](https://eprints.untirta.ac.id)

Internet Source

&lt;1 %

47

[jurnal.uisu.ac.id](https://jurnal.uisu.ac.id)

Internet Source

&lt;1 %

48

[adammd710.blogspot.com](https://adammd710.blogspot.com)

Internet Source

&lt;1 %

49	Submitted to Kocaeli Üniversitesi Student Paper	<1 %
50	repository.ub.ac.id Internet Source	<1 %
51	slidetodoc.com Internet Source	<1 %
52	zainalkholik.wordpress.com Internet Source	<1 %
53	Repository.Umsu.Ac.Id Internet Source	<1 %
54	repository.itk.ac.id Internet Source	<1 %
55	repository.usd.ac.id Internet Source	<1 %
56	adiindra28.blogspot.com Internet Source	<1 %
57	docshare.tips Internet Source	<1 %
58	vbook.pub Internet Source	<1 %
59	wandypardosi.blogspot.com Internet Source	<1 %
60	doc2men.blogspot.com Internet Source	<1 %
61	usmanalli1991.blogspot.com Internet Source	<1 %

---

62	<a href="http://kuliah-e.blogspot.com">kuliah-e.blogspot.com</a> Internet Source	<1 %
63	<a href="#">Submitted to Omer Halisdemir University</a> Student Paper	<1 %
64	<a href="http://repository.usu.ac.id">repository.usu.ac.id</a> Internet Source	<1 %
65	<a href="http://eprints.unpam.ac.id">eprints.unpam.ac.id</a> Internet Source	<1 %
66	<a href="http://library.binus.ac.id">library.binus.ac.id</a> Internet Source	<1 %
67	<a href="http://triekosaputra.blogspot.com">triekosaputra.blogspot.com</a> Internet Source	<1 %
68	<a href="http://jurnal.polsri.ac.id">jurnal.polsri.ac.id</a> Internet Source	<1 %
69	<a href="http://sistemasdinamicosvii.blogspot.com">sistemasdinamicosvii.blogspot.com</a> Internet Source	<1 %
70	<a href="http://arukichiasari.blogspot.com">arukichiasari.blogspot.com</a> Internet Source	<1 %
71	<a href="http://blogs.itb.ac.id">blogs.itb.ac.id</a> Internet Source	<1 %
72	<a href="http://download.garuda.ristekdikti.go.id">download.garuda.ristekdikti.go.id</a> Internet Source	<1 %
73	<a href="http://ejournal.unkhair.ac.id">ejournal.unkhair.ac.id</a> Internet Source	<1 %
74	<a href="http://wahyu091090.blogspot.com">wahyu091090.blogspot.com</a> Internet Source	<1 %

---

---

75	<a href="http://astlabmatematika.wordpress.com">astlabmatematika.wordpress.com</a> Internet Source	<1 %
76	Submitted to University of Central Florida Student Paper	<1 %
77	<a href="http://his.pusan.ac.kr">his.pusan.ac.kr</a> Internet Source	<1 %
78	<a href="http://smkn1gempol.sch.id">smkn1gempol.sch.id</a> Internet Source	<1 %
79	<a href="http://www2.peq.coppe.ufrj.br">www2.peq.coppe.ufrj.br</a> Internet Source	<1 %
80	<a href="http://fendiramadhan76.blogspot.com">fendiramadhan76.blogspot.com</a> Internet Source	<1 %
81	Submitted to itera Student Paper	<1 %
82	<a href="http://matab-arab-free-course.blogspot.com">matab-arab-free-course.blogspot.com</a> Internet Source	<1 %
83	<a href="http://jurnal.unipasby.ac.id">jurnal.unipasby.ac.id</a> Internet Source	<1 %
84	<a href="http://repositori.umsu.ac.id">repositori.umsu.ac.id</a> Internet Source	<1 %
85	Bill Goodwine. "Engineering Differential Equations", Springer Science and Business Media LLC, 2011 Publication	<1 %
86	<a href="http://digilib.unila.ac.id">digilib.unila.ac.id</a> Internet Source	<1 %

---

87	<a href="https://documents.mx">documents.mx</a> Internet Source	<1 %
88	<a href="http://www.elettrotecnica.unina.it">www.elettrotecnica.unina.it</a> Internet Source	<1 %
89	<a href="http://aryusta09.blogspot.com">aryusta09.blogspot.com</a> Internet Source	<1 %
90	<a href="http://qdoc.tips">qdoc.tips</a> Internet Source	<1 %
91	<a href="http://rizkitomy10.blogs.uny.ac.id">rizkitomy10.blogs.uny.ac.id</a> Internet Source	<1 %
92	<a href="http://lab-elektro.umm.ac.id">lab-elektro.umm.ac.id</a> Internet Source	<1 %
93	<a href="http://vikriguntara1.blogspot.com">vikriguntara1.blogspot.com</a> Internet Source	<1 %
94	<a href="http://idoc.pub">idoc.pub</a> Internet Source	<1 %
95	<a href="http://www.ee.lut.fi">www.ee.lut.fi</a> Internet Source	<1 %
96	Submitted to Universitas Islam Bandung Student Paper	<1 %
97	<a href="http://queenlittle.wordpress.com">queenlittle.wordpress.com</a> Internet Source	<1 %
98	<a href="http://rikikhomarudin09.wordpress.com">rikikhomarudin09.wordpress.com</a> Internet Source	<1 %
99	Submitted to American University of the Middle East	<1 %

100 Submitted to Sultan Agung Islamic University  
Student Paper <1 %

---

101 archive.org  
Internet Source <1 %

---

102 id.scribd.com  
Internet Source <1 %

---

103 alfyandiishaq.wordpress.com  
Internet Source <1 %

---

104 www.mathworks.com  
Internet Source <1 %

---

105 Fredrik Gustafsson, Niclas Bergman.  
"MATLAB® for Engineers Explained",  
Springer Science and Business Media LLC,  
2003  
Publication <1 %

---

106 Richard G. Brereton. "Chemometrics", Wiley,  
2018  
Publication <1 %

---

107 budiartoyoppy.blogspot.com  
Internet Source <1 %

---

108 de.scribd.com  
Internet Source <1 %

---

109 elektro.untirta.ac.id  
Internet Source <1 %

---

110 eprints.undip.ac.id

Internet Source

<1 %

---

111 moam.info  
Internet Source

<1 %

---

112 silo.pub  
Internet Source

<1 %

---

113 ww2.lafayette.edu  
Internet Source

<1 %

---

114 www.cs.gsu.edu  
Internet Source

<1 %

---

115 www.statistikian.com  
Internet Source

<1 %

---

116 Submitted to Colorado Technical University  
Online  
Student Paper

<1 %

---

117 Franky Tanahitumessing. "DESAIN  
AUTOMATIC CONTROL SYSTEM POMPA AIR  
TAWAR PADA KAPAL RAKYAT", ALE  
Proceeding, 2021  
Publication

<1 %

---

118 Submitted to Sogang University  
Student Paper

<1 %

---

119 Submitted to University of Sydney  
Student Paper

<1 %

---

120 myweb.loras.edu  
Internet Source

<1 %

---

121	Submitted to Cranfield University Student Paper	<1 %
122	Fedorov, V.A.. "Homogenization and boundary estimates of shear stiffness for the composites of the tetragonal structure", Composites Part B Engineering, 2016. Publication	<1 %
123	Stefani, Raymond T.. "Design of Feedback Control Systems", Oxford University Press Publication	<1 %
124	Submitted to UIN Syarif Hidayatullah Jakarta Student Paper	<1 %
125	Submitted to UOW Malaysia KDU University College Sdn. Bhd Student Paper	<1 %
126	Submitted to Universitas Putera Batam Student Paper	<1 %
127	<a href="http://hassanshehawy.wordpress.com">hassanshehawy.wordpress.com</a> Internet Source	<1 %
128	<a href="http://control.ee.ethz.ch">control.ee.ethz.ch</a> Internet Source	<1 %
129	<a href="http://dokumen.site">dokumen.site</a> Internet Source	<1 %
130	<a href="http://skripsikomplitt.blogspot.com">skripsikomplitt.blogspot.com</a> Internet Source	<1 %
131	<a href="http://search.jogjalib.com">search.jogjalib.com</a> Internet Source	<1 %



132	<a href="http://yogambote.blogspot.com">yogambote.blogspot.com</a> Internet Source	<1 %
133	Bismil Rabeta, Mohammad A.F Ulhaq, Aswan Tajuddin, Agus Sugiharto. "Simulasi Graphical User Interface Analisis Termodinamika Mesin Turboprop Menggunakan Perangkat Lunak Matlab R2020a", Jurnal Teknologi Kedirgantaraan, 2021 Publication	<1 %
134	I Nyoman Tri Anindia Putra, Ketut Sepdyana Kartini. "PERBANDINGAN METODE PENGENALAN WAJAH MELALUI SURVEILLANCE BERBASIS PENGENALAN WAJAH", SINTECH (Science and Information Technology) Journal, 2021 Publication	<1 %
135	<a href="http://doczz.es">doczz.es</a> Internet Source	<1 %
136	<a href="http://epdf.pub">epdf.pub</a> Internet Source	<1 %
137	<a href="http://file.upi.edu">file.upi.edu</a> Internet Source	<1 %
138	<a href="ftp-dee.poliba.it:8000">ftp-dee.poliba.it:8000</a> Internet Source	<1 %
139	<a href="http://rayhansnajib.blogspot.com">rayhansnajib.blogspot.com</a> Internet Source	<1 %

140	Internet Source	<1 %
141	<a href="http://retii.sttnas.ac.id">retii.sttnas.ac.id</a> Internet Source	<1 %
142	<a href="http://210.42.41.13">210.42.41.13</a> Internet Source	<1 %
143	Submitted to Pontificia Universidad Catolica del Peru Student Paper	<1 %
144	Submitted to Universitas Pelita Harapan Student Paper	<1 %
145	Submitted to Universitas Pendidikan Indonesia Student Paper	<1 %
146	<a href="http://arief-just4fun.blogspot.com">arief-just4fun.blogspot.com</a> Internet Source	<1 %
147	<a href="http://grahailmu.co.id">grahailmu.co.id</a> Internet Source	<1 %
148	<a href="http://knowledgecross.wordpress.com">knowledgecross.wordpress.com</a> Internet Source	<1 %
149	<a href="http://repository.uin-suska.ac.id">repository.uin-suska.ac.id</a> Internet Source	<1 %
150	<a href="http://zoniaelektro.net">zoniaelektro.net</a> Internet Source	<1 %
151	Agus Ilham Palwa Samudra, Misbah Misbah. "RANCANG BANGUN ALAT PENYIRAM	<1 %

KECAMBAH OTOMATIS MENGGUNAKAN  
STM32F746G DISCOVERY DENGAN METODE  
FUZZY LOGIC CONTROL", E-Link: Jurnal  
Teknik Elektro dan Informatika, 2020

Publication

- 
- |     |   |      |
|-----|---|------|
| 152 | Anne Angermann, Michael Beuschel, Martin Rau, Ulrich Wohlfarth. "MATLAB® – Simulink® – Stateflow®", Walter de Gruyter GmbH, 2009<br>Publication | <1 % |
| 153 | Fayers, M.. "James's Conjecture holds for weight four blocks of Iwahori-Hecke algebras", Journal of Algebra, 20071115<br>Publication            | <1 % |
| 154 | Pratap, Rudra. "Getting Started with MATLAB", Oxford University Press<br>Publication  | <1 % |
| 155 | Submitted to Udayana University<br>Student Paper  | <1 % |
| 156 | <a href="http://agungbudi131.blogspot.com">agungbudi131.blogspot.com</a><br>Internet Source   | <1 % |
| 157 | <a href="http://bermatematika.net">bermatematika.net</a><br>Internet Source   | <1 % |
| 158 | <a href="http://control.hut.fi">control.hut.fi</a><br>Internet Source   | <1 % |
| 159 | <a href="http://de.slideshare.net">de.slideshare.net</a><br>Internet Source   | <1 % |
| 160 | <a href="http://ejournal.unesa.ac.id">ejournal.unesa.ac.id</a>  |      |

Internet Source

<1 %

---

161 [latifkurniawan.blogspot.com](http://latifkurniawan.blogspot.com)  
Internet Source

<1 %

---

162 [sutikno.blog.undip.ac.id](http://sutikno.blog.undip.ac.id)  
Internet Source

<1 %

---

163 [ta3leemnet.com](http://ta3leemnet.com)  
Internet Source

<1 %

---

164 [vibdoc.com](http://vibdoc.com)  
Internet Source

<1 %

---

165 [wiranto1995.blogspot.com](http://wiranto1995.blogspot.com)  
Internet Source

<1 %

---

166 [www-ee.swan.ac.uk](http://www-ee.swan.ac.uk)  
Internet Source

<1 %

---

167 [www-micrel.deis.unibo.it](http://www-micrel.deis.unibo.it)  
Internet Source

<1 %

---

168 [www.afdhalilahi.com](http://www.afdhalilahi.com)  
Internet Source

<1 %

---

169 [www.researchinlanders.be](http://www.researchinlanders.be)  
Internet Source

<1 %

---

170 Darmawansyah Darmawansyah, M Khairul Amri Rosa, Ika Novia Anggraini. "Sistem Proteksi Motor Induksi 3 Fasa Terhadap Berbagai Gangguan Menggunakan Mikrokontroler", JURNAL AMPLIFIER : JURNAL ILMIAH BIDANG TEKNIK ELEKTRO DAN KOMPUTER, 2020

<1 %

---

171	Dwi Kresna Wijaya, Doan Perdana, Yoseph Gustommy Biono. "Implementasi dan Analisis Purwarupa Sistem Collision Avoidance pada Mobil Pintar Berbasis Jaringan Sensor Nirkabel", Buletin Pos dan Telekomunikasi, 2017 Publication	<1 %
172	Submitted to LL DIKTI IX Turnitin Consortium Part IV Student Paper	<1 %
173	abduelektro.blogspot.com Internet Source	<1 %
174	anzdoc.com Internet Source	<1 %
175	cicese.repositorioinstitucional.mx Internet Source	<1 %
176	daftaralamat.wordpress.com Internet Source	<1 %
177	elektro.studentjournal.ub.ac.id Internet Source	<1 %
178	eprints.uny.ac.id Internet Source	<1 %
179	hermansaputra-ihdinasirotoImustaqim.blogspot.com Internet Source	<1 %
180	id.123dok.com Internet Source	<1 %

---

181	<a href="http://jurnal.fp.unila.ac.id">jurnal.fp.unila.ac.id</a> Internet Source	<1 %
182	<a href="http://kaydier.wordpress.com">kaydier.wordpress.com</a> Internet Source	<1 %
183	<a href="http://library.uwp.ac.id">library.uwp.ac.id</a> Internet Source	<1 %
184	<a href="http://link.springer.com">link.springer.com</a> Internet Source	<1 %
185	<a href="http://mainceme.info">mainceme.info</a> Internet Source	<1 %
186	<a href="http://repository.radenintan.ac.id">repository.radenintan.ac.id</a> Internet Source	<1 %
187	<a href="http://text.123docz.net">text.123docz.net</a> Internet Source	<1 %
188	<a href="http://www.baliprov.go.id">www.baliprov.go.id</a> Internet Source	<1 %
189	<a href="http://www.burakturgut.net">www.burakturgut.net</a> Internet Source	<1 %
190	<a href="http://www.byteboss.com">www.byteboss.com</a> Internet Source	<1 %
191	<a href="http://xpermanax.wordpress.com">xpermanax.wordpress.com</a> Internet Source	<1 %
192	<a href="http://yps-crypt.blogspot.com">yps-crypt.blogspot.com</a> Internet Source	<1 %

---

193 "Advanced, Contemporary Control", Springer Science and Business Media LLC, 2020  
Publication <1 %

---

194 Leslie T. Fournier. "The Canadian National Railway versus the Canadian Pacific: A Comparative Study", Journal of Political Economy, 1931  
Publication <1 %

---

195 ariestarlight.blogspot.com  
Internet Source <1 %

---

196 skripsiteknikelektro.blogspot.com  
Internet Source <1 %

---

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off