

**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK**



**SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC CONTROL (FLC)
PADA PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA
DENGAN MENGGUNAKAN PERANGKAT LUNAK MATLAB 7.0**

SKRIPSI

Disusun Oleh :

**RIZKY ALAM PURYANTO
NIM. 02.12.075**

SEPTEMBER 2006

LEMBAR PERSETUJUAN

**SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC CONTROL (FLC)
PADA PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA
DENGAN MENGGUNAKAN PERANGKAT LUNAK MATLAB 7.0**

SKRIPSI

*Disusun dan Diajukan Untuk Melengkapi dan Memenuhi Syarat-Syarat
Guna Mencapai Gelar Sarjana Teknik*

Disusun Oleh :

RIZKY ALAM PURYANTO

NIM 02.12.075



Mengetahui,

Ketua Jurusan Teknik Elektro



NIP.Y. 103 950 0274

Diperiksa dan disetujui,

Dosen Pembimbing

Ir. WIDODO PUDJI M, MT

NIP. P. 102 870 0171

**KONSENTRASI TEKNIK ENERGI LISTRIK
JURUSAN TEKNIK ELEKTRO S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2006**

ABSTRAKSI

SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC CONTROL (FLC) PADA PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA DENGAN MENGGUNAKAN PERANGKAT LUNAK MATLAB 7.0

(Rizky Alam Puryanto, 02.12.075, Teknik Energi Listrik (S-1)
(Dosen Pembimbing : Ir. Widodo Pudji Muljanto, MT)

Kata kunci: *Self Organizing* , *Fuzzy* , Motor Induksi , Matlab 7.0

Dalam Skripsi saya ini, menggunakan metode pembelajaran pada jaringan syaraf tiruan yang berupa jaringan kohonen dalam penentuan parameter - parameter logika fuzzy, sehingga kontroler neuro -fuzzy yang terbentuk akan diimplementasikan pada simulasi motor induksi tiga fasa dengan menggunakan perangkat lunak Matlab 7.0 dan dapat kita ketahui performansi sistemnya bahwa pengaturan sistem dengan menggunakan *Fuzzy* berdasarkan pembelajaran kohonen lebih baik jika dibandingkan dengan pengaturan sistem dengan menggunakan PID.

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadirat Tuhan Yang Maha Esa, atas rahmat dan karunia-Nya memungkinkan penulis dapat menyelesaikan skripsi yang berjudul, "SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC CONTROL (FLC) PADA PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA DENGAN MENGGUNAKAN PERANGKAT LUNAK MATLAB 7.0".

Skripsi ini bertujuan untuk memenuhi kurikulum akademik yang harus ditempuh oleh setiap mahasiswa ITN Malang guna mengakhiri pendidikan pada jenjang strata satu jurusan Teknik Elektro konsentrasi Teknik Energi Listrik di Institut Teknologi Nasional Malang.

Atas segala bimbingan, pengarahan dan bantuan yang diberikan, sehingga tersusunnya skripsi ini, maka penulis menyampaikan terima kasih kepada :

1. **Dr. Ir. Abraham Lomi, MSEE**, selaku Rektor ITN Malang.
2. **Ir. Mochtar Asroni, MSME**, selaku Dekan FTI ITN Malang.
3. **Ir. F Yudi Limpraptono, MT**, selaku Ketua Jurusan Teknik Elektro (S1) ITN Malang.
4. **Ir. Widodo Pudji M MT**, selaku Dosen Pembimbing.
5. **Mas Fathan**, sebagai *programmer*.
6. Kedua orang tuaku yang telah membesarkan, mendidik dan selalu mendoakanku sampai terwujud cita-citaku, pengorbananmu tak kan kulupakan selamanya.

7. Kakakku Fanny Arief Poeryanto, yang selalu mendukung dan memberi support, Thank's kak.
8. Rekan-rekan elektro energi listrik yang saling membantu dan saling mengisi dalam pengerjaan skripsi ini.

Akhirnya penulis mengharapkan skripsi dapat berguna dan bermanfaat bagi rekan-rekan mahasiswa khususnya pada jurusan Teknik Elektro konsentrasi Teknik Energi Listrik.

Malang, September 2006

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xiii
DAFTAR GRAFIK.....	xiv
BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Perumusan Masalah.....	2
1.3. Tujuan.....	2
1.4. Batasan Masalah.....	3
1.5. Metodologi	3
1.6. Kontribusi.....	4
1.7. Sistematika	4

BAB II DASAR TEORI

2.1. Teori Motor Induksi	6
2.2.1 Konstruksi Motor Induksi	6
2.2.1.1 Stator	7
2.2.1.2 Rotor	7
2.1.2. Medan Magnet Putar	9
2.1.3. Prinsip Kerja Motor Induksi	10
2.1.4. IGBT (Insulated Gate Bipolar Junction)	11
2.1.5. Inveter 3-fasa	11
2.2. Konsep Pembelajaran Mandiri Pada Jaringan Syaraf Tiruan	
Kohonen	13
2.2.1. Umum	13
2.2.2. Sejarah Perkembangan Jaringan Syaraf Tiruan	14
2.2.3. Komponen Jaringan Syaraf Tiruan	15
2.2.4. Arsitektur Jaringan Syarf Tiruan	15
2.2.5. Proses Pembelajaran	17
2.2.5.1. Pembelajaran Terawasi (Supervise Learning)	19
2.2.5.2. Pembelajaran Tak Terawasi (Unsupervise Learning)	20
2.3. Teori <i>Fuzzy</i>	23
2.3.1. Umum	23
2.3.2. Konsep Dasar Logika Fuzzy	24
2.3.3. Fungsi Keanggotaan (Membranship Function)	25

2.3.4. Variable Linguistik	29
2.3.5. Kontroler Logika Fuzzy.....	31
2.3.7. Penentuan Aturan (Rule Evaluation).....	34
2.3.7.1. Bentuk Aturan Kontrol Fuzzy	37
2.3.7.2. Mekanisme Inference	38
2.3.8. Defuzzyfikasi.....	41
2.3.9. Desain Kontroler	42
2.3.9.1. Bentuk Fungsi Keanggotaan.....	43
2.3.9.2. Proses Pelatihan.....	43
2.4. Alogaritma Proses Pembelajaran Jaringan Syaraf Tiruan Kohonen	44
2.5. Flowchart proses Pembelajaran Jaringan Syarf Tiruan Kohonen	47
2.6. Alogaritma Proses Pembentukan FIS (Fuzzy Inference System) ...	48
2.7. Flowchart Proses Pembentukan FIS (Fuzzy Inference System)	48
2.8. Alogaritma Pemodelan System	50
2.9. Flowchart Pemodelan System	51

**BAB III PERANCANGAN KONTROLLER PADA JARINGAN SYARAF
TIRUAN KOHONEN**

3.1. Perencanaan Kontoler Pada Jaringan Syaraf Tiruan Kohonen	51
3.2. Pengambilan Data Training.....	52
3.3. Proses Pembelajaran Kohonen	56
3.3.1 Proses Pembelajaran Jaringan Kohonen Untuk Kecepatan sudut putar 100 rad/s	56

3.3.2 Proses Pembelajaran Jaringan Kohonen Untuk Kecepatan sudut putar 120 rad/s	58
3.4. Proses Pembentukan FIS (Fuzzy Inference System).....	61
3.4.1. Proses Fuzzyfikasi.....	61
3.4.2. Penentuan Aturan (<i>Rule Evaluation</i>).....	67
3.4.3. Proses Defuzzyfikasi.....	70

BAB IV SIMULASI DAN ANALISA HASIL

4.1. Simulasi Dengan menggunakan Fuzzy berdasarkan Pembelajaran Jaringan Kohonen.....	74
4.1.1. Pengujian pada kecepatan sudut putar 100 rad/s dengan torsi beban 0 Nm	77
4.1.2. Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 0 Nm	78
4.1.3. Pengujian pada kecepatan sudut putar 100 rad/s dengan torsi beban 50 Nm	80
4.1.4. Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 50 Nm	81
4.2. Simulasi Dengan menggunakan PID.....	86
4.2.1. Pengujian pada kecepatan sudut putar 100 rad/s dengan torsi beban 0 Nm	86
4.2.2. Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 0 Nm	87

4.2.3. Pengujian pada keccpatan sudut putar 100 rad/s dengan torsi beban 50 Nm	89
4.2.4. Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 50 Nm	90
4.3. Analisa Performansi Sistem	92

BAB V PENUTUP

5.1. Kesimpulan.....	95
5.2. Saran	97

DAFTAR PUSTAKA	XX
-----------------------------	-----------

LAMPIRAN

DAFTAR GAMBAR

Gambar 2-1	Konstruksi Motor Induksi	6
Gambar 2-2	Stator 3-fasa Motor Induksi.....	7
Gambar 2-3	Rotor Belitan	8
Gambar 2-4	Rotor Sangkar.....	8
Gambar 2-5	Medan Putar Motor Induksi	9
Gambar 2-6	Rangkaian Inverter 3-fasa	12
Gambar 2-7	Susunan Syaraf Manusia.....	13
Gambar 2-8	Susunan Neuron Jaringan Syaraf	15
Gambar 2-9	Jaringan Syaraf Dengan Lapisan Tunggal	16
Gambar 2-10	Jaringan Syaraf Dengan Banyak Lapisan.....	17
Gambar 2-11	Struktur Jaringan Kohonen.....	22
Gambar 2-12	Bagian-bagian Fuzzy Logic.....	24
Gambar 2-13	S-Function	26
Gambar 2-14	Π - Function	27
Gambar 2-15	T-Function.....	28
Gambar 2-16	Trapeziodal Function.....	28
Gambar 2-17	Himpunan Fuzzy Untuk Kecepatan Motor	30
Gambar 2-18	Struktur Sistem Kontrol Logika Fuzzy Secara Umum.....	31
Gambar 2-19	Tahapan Proses fuzzy Logic	32
Gambar 2-20	Proses Fuzzyfikasi	33

Gambar 2-21 Proses Mekanisme Inference.....	39
Gambar 2-22 Proses Penalaran Fuzzy MISO dengan Operasional Max-Min.....	40
Gambar 2-23 Proses defuzzyfikasi.....	41
Gambar 3-1 Pengambilan Data Training.....	52
Gambar 3-4 FIS Editor pada kecepatan sudut putar 100 rad/s.....	64
Gambar 3-5 Membranship Error pada kecepatan sudut putar 100 rad/s.....	64
Gambar 3-6 Membranship D Error pada kecepatan sudut putar 100 rad/s.....	65
Gambar 3-7 Membranship Output pada kecepatan sudut putar 100 rad/s.....	65
Gambar 3-8 FIS Editor pada kecepatan sudut putar 120 rad/s.....	67
Gambar 3-9 Membranship Error pada kecepatan sudut putar 120 rad/s.....	67
Gambar 3-10 Membranship D Error pada kecepatan sudut putar 100 rad/s.....	68
Gambar 3-11 Membranship Output pada kecepatan sudut putar 120 rad/s.....	69
Gambar 4-1 Pemodelan sistem dengan menggunakan Fuzzy.....	74
Gambar 4-2 Teraan parameter motor induksi 3 fasa untuk fuzzy.....	74
Gambar 4-3 Teraan Parameter IGBT Inverter 3 fasa untuk fuzzy.....	75
Gambar 4-4 Blok Diagram Vektor Kontrol Fuzzy.....	76
Gambar 4-5 Pemodelan sistem dengan menggunakan PID.....	83
Gambar 4-6 Teraan parameter motor induksi 3 fasa untuk PID.....	83
Gambar 4-7 Teraan Parameter IGBT Inverter 3 fasa untuk PID.....	84
Gambar 4-8 Blok Diagram Vektor Kontrol PID.....	85

BAB I PENDAHULUAN

1.1 Latar Belakang

Secara umum didalam bidang industri sistem pengaturan memegang peranan yang sangat penting guna meningkatkan efisiensi dalam proses produksi dan menunjang proses berjalannya industri tersebut.

Untuk mengendalikan motor induksi telah banyak dikembangkan berbagai macam kontroler, salah satunya adalah kontroler logika *fuzzy*. *Fuzzy Logic Control (FLC)* merupakan salah satu dari kontroler cerdas yang dalam perancangannya diperlukan pengalaman yang cukup untuk menentukan parameternya.

Secara konvensional kontroler dirancang berdasarkan pada model matematis dari sistem yang akan digunakan, kondisi ini menyulitkan untuk implementasi pada sistem yang sesungguhnya, terutama pada sistem yang kompleks dan tidak linier. Sebagai alternatif telah dikembangkan konsep logika *fuzzy* untuk berbagai macam pengontrolan. Akan tetapi jenis kontroler ini memerlukan pengalaman dari perancangannya. Jika ini tidak tersedia maka tujuan pengontrolan akan sulit tercapai.

Untuk mengatasi ketidak tersediaan pengalaman yang cukup, maka dikembangkan pembelajaran mandiri logika *fuzzy* sehingga pengalaman tidak lagi dominan dalam perancangannya.

Dalam Skripsi saya ini, menggunakan metode pembelajaran pada jaringan syaraf tiruan yang berupa jaringan kohonen dalam penentuan parameter –

parameter logika *fuzzy* yaitu fungsi keanggotaannya. Kontroler *neuro-fuzzy* yang terbentuk akan diimplementasikan pada simulasi motor induksi tiga fasa dengan menggunakan perangkat lunak *Matlab 7.0*

1.2 Perumusan Masalah

Mengacu pada latar belakang diatas maka permasalahan yang timbul adalah:

1. Bagaimana performansi sistem pada saat kondisi berbeban.
2. Bagaimana performansi sistem pada saat kondisi tanpa beban .
3. Bagaimana performansi sistem, ketika diberikan setting point yang berbeda.

1.3 Tujuan Penelitian

Merancang suatu kontroler logika *fuzzy* melalui metode pembelajaran pada jaringan syaraf tiruan yang berupa jaringan kohonen dalam penentuan parameter – parameter kontroler logika *fuzzy* yaitu fungsi keanggotaannya. Kontroler *neuro-fuzzy* yang terbentuk akan diimplementasikan pada simulasi motor induksi tiga fasa dengan menggunakan perangkat lunak *Matlab 7.0*, sehingga dapat dianalisa performansi sistem yang nantinya berupa *rise time*, *settling time*, dan *error steady state*.

1.4 Batasan Masalah

Agar pembahasan skripsi ini lebih terarah sesuai dengan tujuan, maka permasalahan dibatasi oleh hal-hal sebagai berikut:

1. Motor yang digunakan adalah Motor induksi jenis sangkar rotor sangkar (*Squirrel-Cage Rotor*).
2. Analisa dilakukan dalam keadaan mantab.
3. Metode yang digunakan adalah Metode pembelajaran pada jaringan syaraf tiruan yang berupa jaringan kohonen.
4. Penentuan parameter – parameter kontroler logika fuzzy yaitu dengan menentukan nilai tengah dari fungsi keanggotaannya.
5. PID hanya sebagai pembandingan performansi sistem dari pengaturan dengan menggunakan *Fuzzy* melalui pembelajaran jaringan kohonen.
6. Simulasi hasil dengan software *simulink-MATLAB 7.0* secara off-line.

1.5 Metodologi Penelitian

Metode yang dipakai untuk menyelesaikan masalah-masalah diatas adalah:

1. Studi Literatur.
 2. Perancangan *fuzzy* pada jaringan syaraf tiruan kohonen dengan software *simulink-MATLAB 7.0*
 3. Simulasi dengan *simulink-MATLAB 7.0*
 4. Analisa data hasil perancangan.
 5. Penarikan kesimpulan.
-

1.6 Kontribusi

Bahwa sistem pengaturan motor induksi mempunyai peranan yang sangat penting dalam menjaga kestabilan sistem dan meningkatkan efisiensi energi. Sehingga diharapkan dengan penggunaan *Fuzzy Logic Controller* (FLC) ini menjadi alternatif sistem pengendalian yang lebih produktif, ekonomis dan bermanfaat lebih bagi industri-industri yang ada.

1.7 Sistematika Penulisan

Skripsi ini dibagi menjadi lima bab, yaitu :

BAB I : PENDAHULUAN

Berisikan tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian dan sistematik penulisan.

BAB II : DASAR TEORI

Membahas tentang teori dasar mengenai teori tentang motor induksi tiga fasa, teori tentang pembelajaran pada jaringan syaraf tiruan kohonen dan teori tentang logika *fuzzy*.

BAB III: PERANCANGAN FUZZY PADA JARINGAN SYARAF TIRUAN KOHONEN

Membahas tentang perencanaan kontroler fuzzy pada jaringan syaraf tiruan kohonen, pembelajaran jaringan kohonen serta pembentukan FIS (*Fuzzy Inference System*).

BAB IV : SIMULASI DAN HASILNYA

Membahas tentang simulasi pengaturan kecepatan sudut putar motor induksi tiga fasa dengan menggunakan Fuzzy melalui pembelajaran pada jaringan syaraf tiruan kohonen dengan menggunakan dan PID, yang bertujuan untuk mengetahui performansi dari kontroler yang dirancang.

BAB V : PENUTUP

Berisikan kesimpulan dari hasil simulasi dan saran.

BAB II

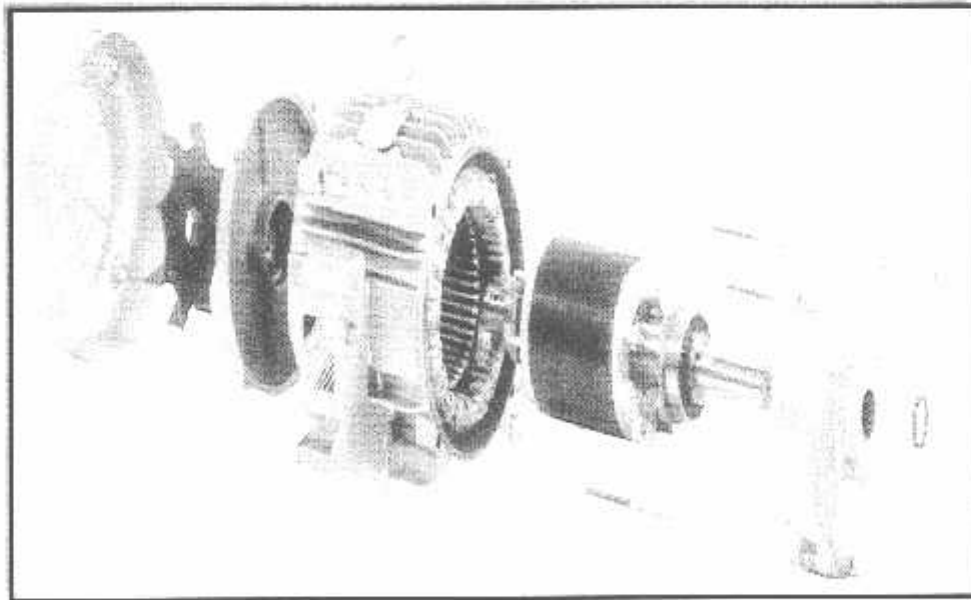
DASAR TEORI

2.1 Umum

Motor arus bolak-balik (Motor AC) adalah suatu mesin yang berfungsi untuk mengubah tenaga listrik arus bolak-balik menjadi tenaga mekanik atau tenaga gerak, dimana tenaga gerak ini berupa perputaran pada motor. Salah satu jenis motor AC ini adalah motor induksi.

2.1.1 Konstruksi Motor Induksi

Konstruksi motor induksi terdiri dari dua bagian utama yaitu stator dan rotor. Hal ini dapat dilihat pada gambar 2-1 di bawah ini :



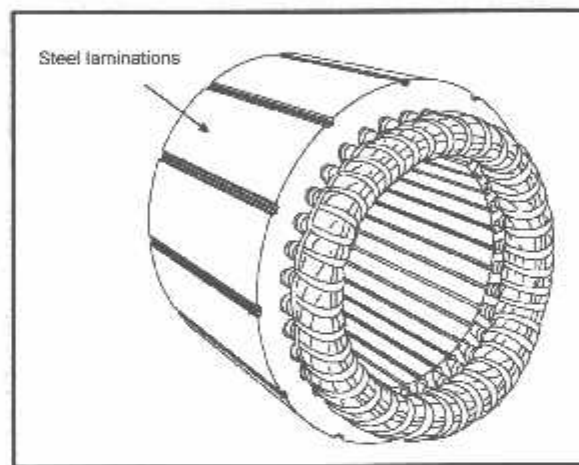
Gambar 2-1 : Konstruksi Motor Induksi 3-fasa

Sumber : <http://www.enr.com>, Dynamics of 3-phase Induction Motors, Part I.

2.1.1.1 Stator

Pada dasarnya konstruksi stator pada motor induksi mempunyai bentuk fisik yang sama dengan mesin sinkron, yang terdiri dari :

- a. Rumah stator terbuat dari besi tuang.
- b. Inti stator dari besi atau baja silikon.
- c. Alur dan gigi materialnya sama dengan inti, alur tempat meletakkan belitan.
- d. Belitan stator dari tembaga.



Gambar 2-2: Stator 3-fasa motor induksi

Sumber : <http://www.enm.com>, Dynamics of 3-phase Induction Motors, Part I.

2.1.1.2 Rotor

Konstruksi dari rotor motor induksi mempunyai 2 bentuk, yaitu :

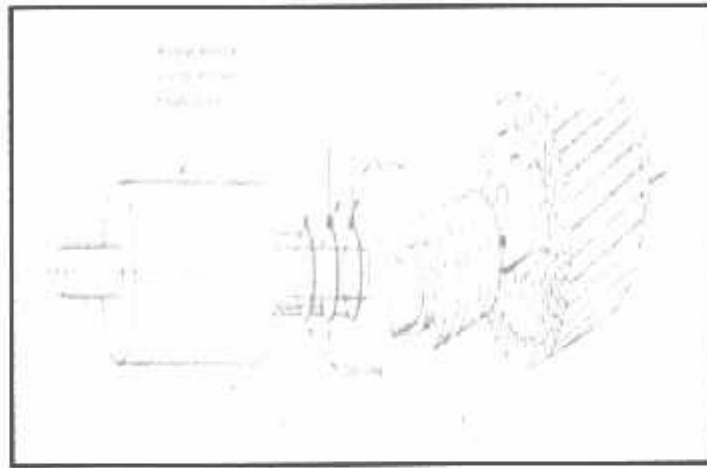
1. Rotor Belitan (*wound rotor*)

Konstruksi rotor belit biasa dipakai pada motor induksi dengan kapasitas yang besar, dimana kumparan rotor dihubungkan dengan cincin seret yang berguna untuk menghubungkan kumparan rotor dengan tahanan yang berfungsi untuk mengatur arus awal

2. Rotor sangkar (*squirrel cage rotor*).

Rotor sangkar-tupai terdiri atas beberapa batang tembaga yang ditanam di alur-alur rotor. Pada kedua sisi rotor terdapat cincin untuk menghubungkan ujung-ujung batang tembaga sehingga akan membentuk suatu sangkar.

Kedua jenis motor induksi 3 fasa ini dapat dilihat pada gambar 2-3 dan 2-4.



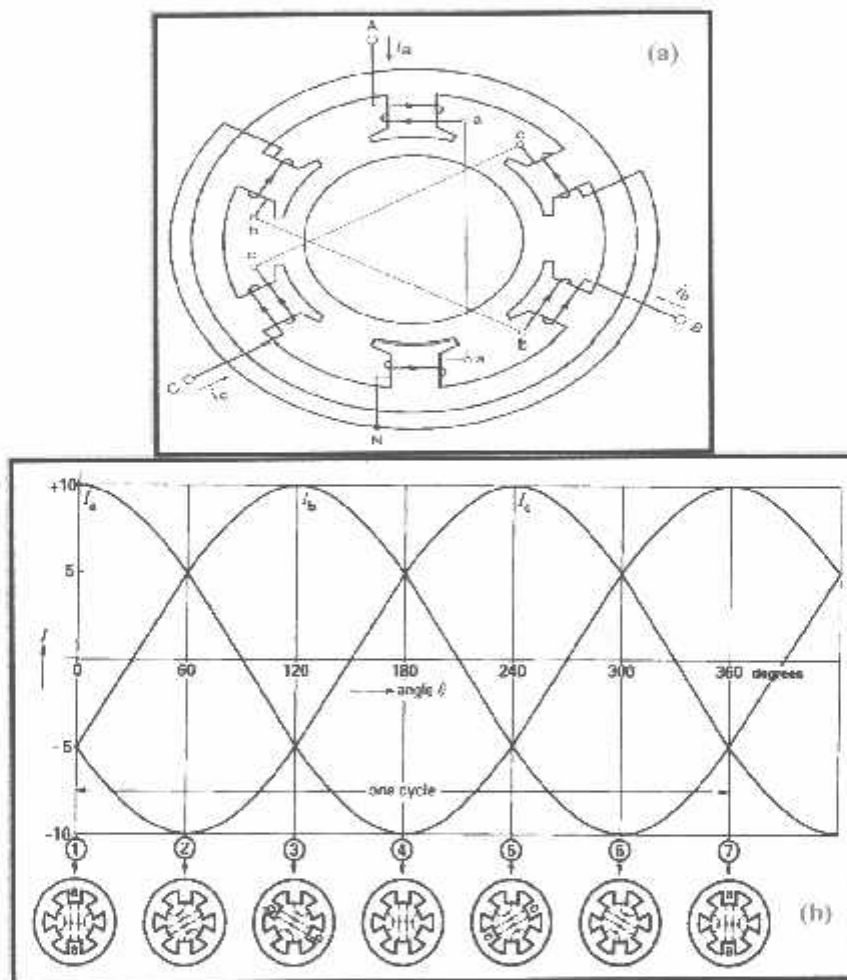
Gambar 2-3 : Rotor belit (a) secara skematis; (b) kondisi sebenarnya
Sumber : Nasyar, Syed A., 1995



Gambar 2-4: Rotor sangkar (a) secara skematis; (b) bila dibelah
Sumber : Nasyar, Syed A., 1995

2.1.2 Medan Magnet Putar

Perputaran motor pada mesin arus bolak-balik ditimbulkan oleh adanya medan putar (fluks yang berputar) yang dihasilkan dalam kumparan statornya. Medan putar ini terjadi apabila kumparan stator dihubungkan dalam fasa banyak (*polyfasa*), umumnya 3 fasa. Hubungan dapat berupa hubungan bintang atau delta.



Gambar 2-5 : Medan Putar Pada Motor Induksi 3-fasa
 Sumber : <http://www.enm.com>, Dynamics of 3-phase Induction Motors, Part I.

Medan putar terjadi apabila kumparan A-a, B-b, C-c dihubungkan 3 fasa dengan beda fasa masing-masing 120° (hubungan bintang, Y) dan dialiri arus sinusoida. Distribusi i_a , i_b , i_c sebagai fungsi waktu adalah seperti gambar 2-5. Pada keadaan t_1 fluks resultan mempunyai arah yang sama dengan arah yang dihasilkan oleh kumparan A-a, sedangkan pada t_3 , fluks resultannya dihasilkan oleh kumparan B-b. Untuk t_4 , fluks resultannya berlawanan arah dengan fluks resultan yang dihasilkan pada t_1 . Dari gambar 2-5 b tersebut terlihat bahwa fluks resultan ini akan berputar satu kali.

2.1.3 Prinsip Kerja Motor Induksi

Prinsip kerja motor induksi secara ringkas dapat diterangkan sebagai berikut :

1. Apabila sumber tegangan 3 fasa dipasang pada kumparan stator akan timbul medan putar dengan kecepatan :

$$n_s = \frac{120f}{P} \text{ rpm}$$

2. Medan putar stator tersebut akan memotong batang konduktor pada rotor sehingga pada kumparan rotor timbul tegangan induksi (*ggl induksi*).
3. Karena kumparan rotor merupakan rangkaian yang tertutup, maka *ggl* (E) akan mengalir arus (I). Adanya arus (I) yang berada dalam medan magnet akan menimbulkan gaya (F) pada rotor.
4. Bila kopel mula yang dihasilkan oleh gaya (F) pada rotor lebih besar dari pada kopel beban, maka rotor akan berputar searah dengan medan putar stator.

5. Seperti yang telah dijelaskan bahwa tegangan induksi timbul karena tepotongnya batang konduktor (rotor) oleh medan putar stator. Artinya agar tegangan terinduksi diperlukan adanya perbedaan relatif antara kecepatan medan stator (n_s) dengan kecepatan berputar rotor (n_r).
6. Perbedaan kecepatan putar antar medan putar dengan kecepatan putar rotor disebut slip (s), dinyatakan dengan :

$$S = \frac{n_s - n_r}{n_s} \times 100\%$$

7. Bila $n_r = n_s$, tegangan tidak akan terinduksi dan arus tidak mengalir pada kumparan jangkar rotor, dengan demikian tidak dihasilkan kopel. Kopel motor akan ditimbulkan apabila n_r lebih kecil dari n_s .
8. Sehingga dari cara kerjanya, motor induksi disebut juga motor tak serempak atau asinkron.

2.1.4 IGBT (*Insulated Gate Bipolar Transistor*)

IGBT adalah transistor daya yang dikendalikan tegangan. Selanjutnya IGBT ini akan digunakan sebagai rangkaian penyusun rangkaian *inverter* (pengkonversi sinyal DC ke AC) yang digunakan untuk mencatu kumparan stator dari motor induksi.

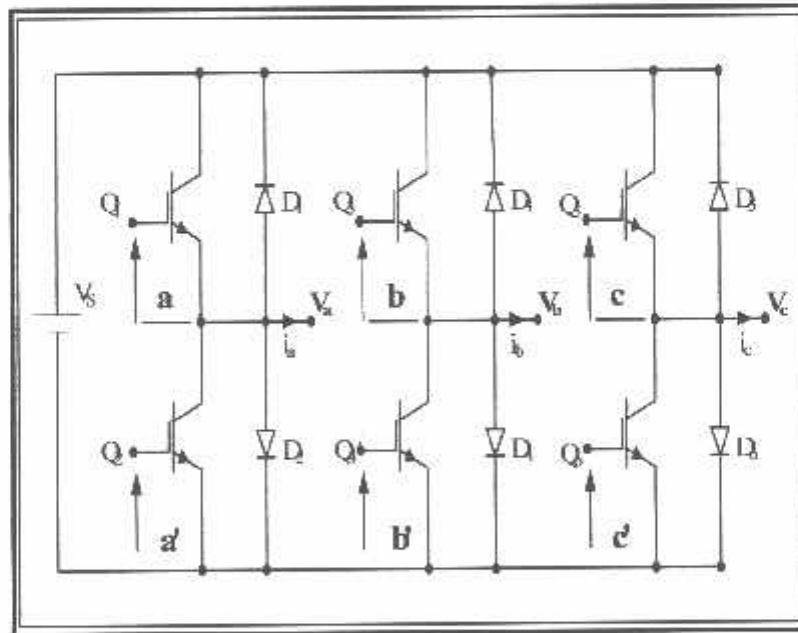
2.1.5 Inverter 3 fasa

Rangkaian *inverter* 3-fasa ditunjukkan pada gambar 2-6 V_a, V_b dan V_c merupakan tegangan keluaran *inverter* yang dicatukan ke kumparan stator motor

induksi. Pada gambar 2-8 terlihat bahwa terdapat 6 dioda (D_1 sampai dengan D_6) yang dipasang dengan seluruh IGBT (Q_1 sampai dengan Q_6).

Keenam dioda tersebut berfungsi sebagai dioda *feedback* yang digunakan sebagai jalur pelepasan energi induktif dari kumparan stator motor induksi yang harus dikembalikan ke sumber saat kondisi IGBT *off*. Misalnya apabila Q_1 dan Q_4 berada dalam kondisi *off*, maka energi induktif akan dikembalikan ke sumber melalui D_3 dan D_2 .

Proses *on* dan *off* dari keenam IGBT dikontrol oleh sinyal *gate* a , a' , b , b' , c dan c' . Pada pengontrolan motor induksi ini, pada saat IGBT bagian atas (Q_1 , Q_3 atau Q_5) dalam keadaan *on*, yaitu saat a , b atau c berlogika 1, maka IGBT bagian bawah yang bersesuaian (Q_2 , Q_4 / Q_6) akan *off* yaitu saat a' , b' atau c' berlogika 0.

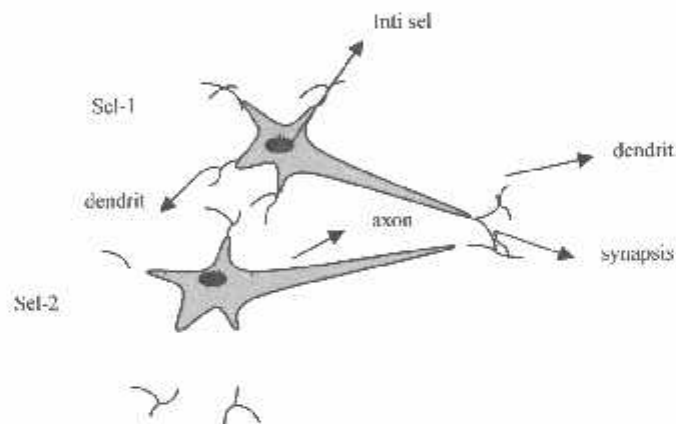


Gambar 2-6 : Rangkaian *Inverter* 3-fasa
Sumber : Rashid, Muhammad H., 1993 : 366

2.2 KONSEP PEMBELAJARAN MANDIRI PADA JARINGAN SYARAF TIRUAN KOHONEN

2.2.1 Umum

Jaringan syaraf tiruan merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran.



Gambar 2-7 : *Susunan syaraf manusia*

Sumber: Ir.Yusuf Ismail Nahkoda, MT, "Diktat Mata Kuliah Logika Fuzzy", ITN Malang

Jaringan syaraf tiruan ini memiliki keunggulan diantaranya sebagai berikut :

1. Toleransi terhadap kesalahan

Jaringan syaraf bila mengalami kerusakan sebagian maka kinerjanya hanya kan sedikit menurun. Sebaliknya, kebanyakan program dan sistem yang direkayasa adalah rapuh, bila sedikit saja bagian tertentu sembarang diambil, akan sangat mungkin keseluruhan fungsi akan terganggu.

2. Kemampuan belajar (*learning, reorganizing itself*)

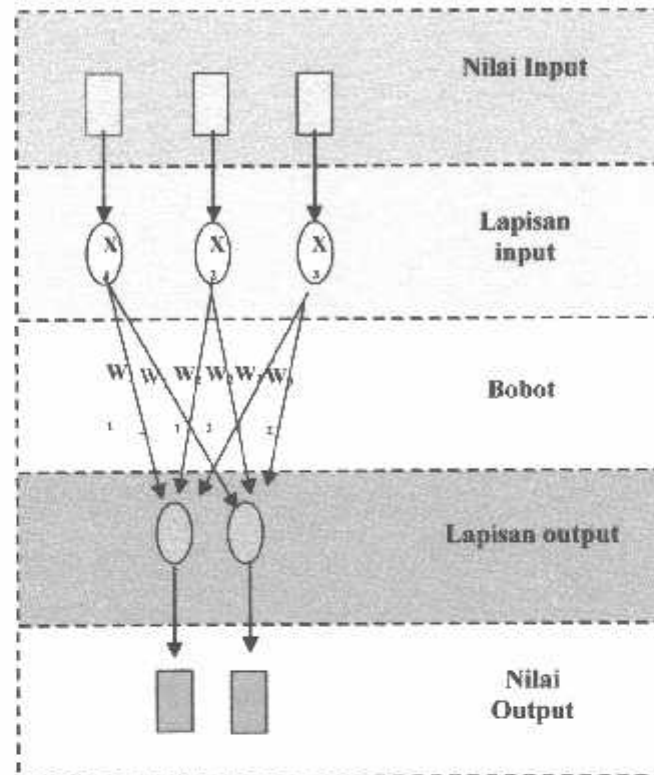
Jaringan syaraf memiliki kemampuan belajar (mereorganisir diri) berdasarkan pengalaman.

3. Cerdas (*intelligent*)

2.2.2 Sejarah perkembangan jaringan syaraf tiruan

Mulai dari ditemukannya, jaringan syaraf tiruan telah mengalami tahap-tahap perkembangan, antara lain :

- Pada tahun 1940-an, ilmuwan menemukan bahwa psikologi dari otak sama dengan mode pemrosesan yang dilakukan oleh peralatan komputer.
 - Pada tahun 1943, McCulloch dan Pitts merancang model formal pertama kali sebagai perhitungan dasar neuron.
 - Pada tahun 1954, Farley dan Clark mensetup model-model untuk relasi adaptif stimulus-respon dalam jaringan *random*.
 - Pada tahun 1958, Rosenbalt mengembangkan konsep dasar tentang *perceptron* untuk klasifikasi pola.
 - Pada tahun 1974, Werbos memperkenalkan algoritma *backpropagation* untuk melatih *perceptron* dengan banyak lapisan.
 - Pada tahun 1975, Little dan Shaw menggambarkan jaringan syaraf dengan menggunakan model probabilistik.
 - Pada tahun 1982, Kohonen mengembangkan metode pembelajaran jaringan syaraf yang tidak terawasi (*unsupervised learning*) untuk pemetaan.
-

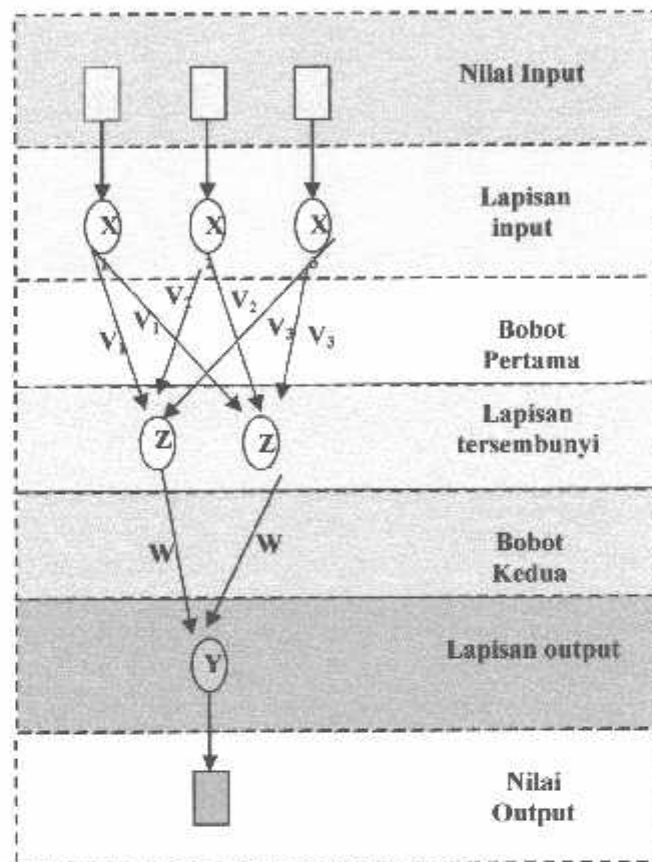


Gambar 2-9 : Jaringan Syaraf dengan lapisan tunggal
 Sumber: Ir.Yusuf Ismail Nahkoda, MT, "Diktat Mata Kuliah Jaringan Syaraf Tiruan," ITN Malang

2. Jaringan dengan banyak lapisan (*Multilayer Net*)

Jaringan dengan banyak lapisan memiliki 1 atau lebih lapisan terletak diantara lapisan input dan lapisan output (memiliki 1 atau lebih lapisan tersembunyi), seperti terlihat pada gambar 2-10 Jumlah hidden unit tergantung pada kebutuhan. Sampai sekarang belum ada rumusan atau ketetapan mengenai jumlah pemakaian hidden unit. Terdapat dua buah layer dengan bobot v dan w . Umumnya, ada lapisan bobot-bobot yang terletak diantara 2 lapisan (*layer*) yang bersebelahan. Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit daripada jaringan dengan lapisan tunggal, tentu saja dengan pembelajaran

yang lebih rumit. Namun demikian, pada banyak kasus, pembelajaran pada jaringan dengan banyak lapisan ini lebih sukses dalam menyelesaikan masalah.



Gambar 2-10 : Jaringan syaraf dengan banyak lapisan

Sumber: Ir.Yusuf Ismail Nahkoda, MT, "Diktat Mata Kuliah Jaringan Syaraf Tiruan,"ITN Malang

2.2.5 Proses Pembelajaran (*Learning*)

Pada otak manusia, informasi yang dilewatkan dari satu neuron yang lainnya berbentuk rangsangan listrik melalui dendrite. Jika rangsangan tersebut diterima oleh suatu neuron, maka neuron tersebut akan membangkitkan output kesemua neuron yang berhubungan dengannya sampai informasi tersebut sampai ketujuannya yaitu terjadinya suatu reaksi. Jika rangsangan yang diterima terlalu

halus, maka output yang dibangkitkan oleh neuron tersebut tidak akan direspon. Tentu saja sangatlah sulit untuk memahami bagaimana otak manusia bisa belajar. Selama proses pembelajaran, terjadi perubahan yang cukup berarti pada bobot-bobot yang menghubungkan antar neuron.

Apabila ada rangsangan yang sama dengan rangsangan yang telah diterima oleh neuron, maka neuron akan memberikan reaksi dengan cepat. Namun, apabila kelak ada rangsangan yang berbeda dengan apa yang telah diterima oleh neuron, maka neuron akan segera beradaptasi untuk memberikan reaksi yang sesuai.

Jaringan syaraf akan mencoba untuk mensimulasikan kemampuan otak manusia untuk belajar. Jaringan syaraf tiruan juga tersusun atas neuron-neuron dan dendrite. Tidak seperti model biologis, jaringan syaraf memiliki struktur yang tidak dapat diubah, dibangun oleh sejumlah neuron, dan memiliki nilai tertentu yang menunjukkan seberapa besar koneksi antara neuron (yang dikenal dengan nama bobot). Perubahan yang terjadi selama proses pembelajaran adalah perubahan nilai bobot. Nilai bobot akan bertambah, jika informasi yang diberikan oleh neuron yang bersangkutan tersampaikan, sebaliknya jika informasi tidak disampaikan oleh suatu neuron ke neuron yang lain, maka nilai bobot yang menghubungkan keduanya akan dikurangi.

Pada saat pembelajaran dilakukan pada input yang berbeda, maka nilai bobot akan diubah secara dinamis hingga mencapai suatu nilai yang cukup seimbang. Apabila nilai ini telah tercapai mengindikasikan bahwa tiap-tiap input telah berhubungan dengan output yang diharapkan.

Metode Pembelajaran pada jaringan syaraf ada 2, antara lain:

1. Pembelajaran terawasi (*Supervised Learning*)
2. Pembelajaran tak terawasi (*Unsupevised Learning*)

2.2.5.1 Pembelajaran terawasi (*Supervised Learning*)

Metode pembelajaran pada jaringan syaraf disebut terawasi jika output yang diharapkan telah diketahui sebelumnya. Dalam proses belajar yang terawasi, seolah-olah ada "guru" yang mengajari jaringan. Contoh: andaikan kita memiliki jaringan syaraf yang akan digunakan untuk mengenali pasangan pola, misalkan pada operasi AND.

Input		Target
0	0	0
0	1	0
1	0	0
1	1	1

Pada proses pembelajaran (training), satu pola input akan diberikan ke satu neuron lagi. pada lapisan input. Pola ini akan dirambatkan disepanjang jaringan syaraf hingga sampai ke neuron pada lapisan output. Lapisan output ini akan membangkitkan pola output yang nantinya akan dicocokkan dengan pola output targetnya. Apabila terjadi perbedaan antara pola output hasil pembelajaran dengan pola target, maka disini akan muncul error. Apabila nilai error ini masih cukup besar, mengindikasikan bahwa masih perlu dilakukan banyak pembelajaran

Dalam proses belajar yang terawasi , seolah-olah ada "guru" yang mengajari jaringan. Cara pelatihan jaringan tersebut adalah dengan memberikan data-data yang disebut training data terdiri atas pasangan input-output yang diharapkan. Data-data itu biasanya, didapat dari pengalaman atau pengetahuan seseorang dalam penyelesaian persoalan. Setelah jaringan dilatih, akan mengingat suatu pola. Jika jaringan diberi input baru, jaringan dapat mengeluarkan output seperti yang diharapkan (*desired* atau *target output*) berdasarkan pola yang sudah ada. Ada banyak metode yang menggunakan prinsip pembelajaran terawasi ini, antara lain:

- | | |
|---------------|-----------------------------|
| 1. Hebb rule | 4. Heteroassociative Memory |
| 2. Perceptron | 5. Counter Propagation |
| 3. Delta Rule | 6. Backpropagation |

Dari keenam metode tersebut, metode yang paling sering digunakan adalah ***Backpropagation***. Ini dikarenakan backpropagation selain cukup simpel, metode ini juga telah terbukti mampu menyelesaikan masalah yang rumit dengan sukses.

2.2.5.2 Pembelajaran tak terawasi (*Unsupervised Learning*)

Pada metode pembelajaran tak terawasi tidak memerlukan target output. Pada metode ini, tidak dapat ditentukan hasil yang seperti apakah yang diharapkan selama proses pembelajaran. Selama proses pembelajaran, nilai bobot disusun dalam suatu range tertentu tergantung pada nilai input yang diberikan.

Tujuan pembelajaran ini adalah pengelompokan unit-unit yang hampir sama dalam suatu area tertentu. Pembelajaran ini biasanya sangat cocok untuk pengelompokan (klasifikasi) pola. Perlu diketahui, bahwa biasanya dalam pelatihan jaringan, selain ada data untuk training (*training set*), juga ada data untuk tes (*test set*). Training data, selain digunakan untuk pelatihan, juga digunakan untuk memantau besarnya error yang terjadi antara output yang dihasilkan jaringan dengan output yang diharapkan.

Training data akan mempengaruhi proses pelatihan. Jadi, pada training data set, baik data input maupun output, semuanya digunakan untuk pelatihan (*training*). Sedangkan test set dipakai dalam perhitungan untuk mengetahui output yang dihasilkan jaringan karena adanya data input. Dalam test tersebut, dapat diketahui seberapa banyak jaringan telah belajar (*iterasi* atau *epoch*) dari pelatihan dan apakah jaringan telah mampu untuk menghasilkan output yang benar dengan bobot yang sudah ada.

Data output pada test set hanya digunakan untuk menentukan besarnya error dengan cara membandingkan output jaringan dengan output yang diharapkan (tidak mempengaruhi proses pelatihan) .tidak ada aturan khusus tentang banyaknya *training data* dan *test data* tersebut. Jadi training data dan test data diambil secukupnya sesuai dengan kebutuhan.

Salah satu contoh yang paling menonjol dari pembelajaran tak terawasi adalah *Self-Organizing Feature Map (SOFM)* yang dikembangkan oleh *Kohonen* (1982,1989) atau *Jaringan Syaraf Tiruan Kohonen*.

2.3 TEORI FUZZY

2.3.1 Umum

Dalam logika konvensional, nilai kebenaran mempunyai kondisi yang pasti yaitu benar atau salah (*true or false*), dengan tidak ada kondisi antara. Prinsip ini dikemukakan oleh Aristoteles sekitar 2000 tahun yang lalu sebagai hukum *Excluded Middle* dan hukum ini telah mendominasi pemikiran logika di dunia (barat) sampai sekarang. Tentu saja, pemikiran mengenai logika konvensional dengan nilai kebenaran yang pasti yaitu benar atau salah dalam kehidupan yang nyata sangatlah tidak mungkin. Logika *fuzzy* (*Fuzzy Logic*) menawarkan suatu logika yang dapat merepresentasikan keadaan dunia nyata.

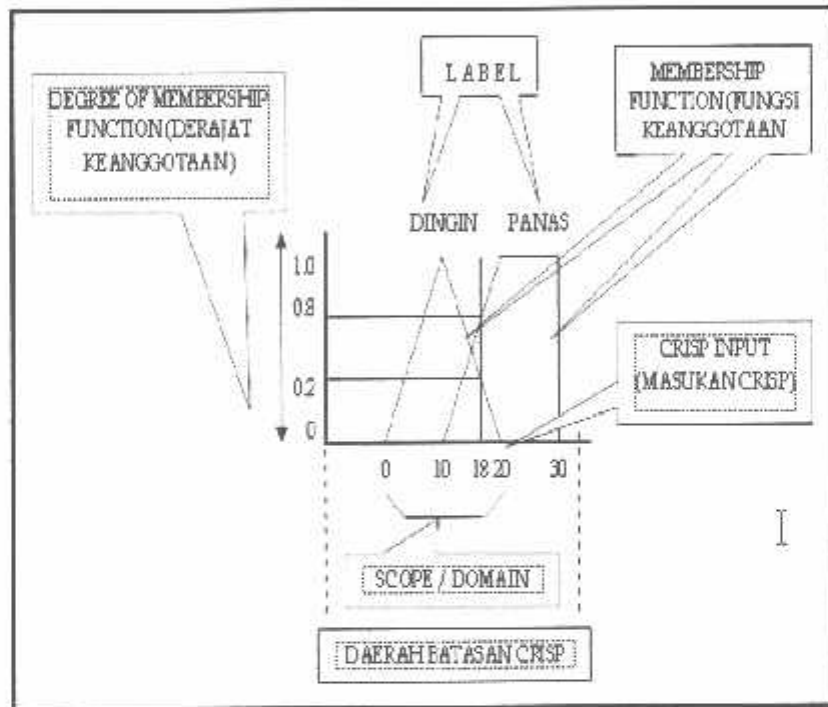
Konsep himpunan fuzzy pertama kali diperkenalkan oleh **Prof. Lotfi A Zadeh** – orang Iran yang menjadi guru besar dari California Universitas di Berkley dalam makalahnya "*fuzzy set*" pada tahun 1965. Dalam makalahnya tersebut dipaparkan ide dasar *fuzzy set* yang meliputi *union*, *intersection* dan *complement*. Ide tersebut terus dimatangkan oleh Zadeh dalam beberapa papernya.

Pelopop aplikasi *fuzzy set* dalam bidang kontrol, yang merupakan aplikasi pertama dan utama dari *fuzzy set* adalah Prof. Ebrahim Mamdani dan kawan-kawan dari *Queen Mary College London*. Logika *fuzzy* dikatakan sebagai logika baru yang lama, sebab ilmu tentang logika *fuzzy* yang modern dan metodis baru ditemukan beberapa tahun yang lalu, padahal sebenarnya konsep tentang logika *fuzzy* itu sendiri ada pada diri kita sejak lama.

2.3.2 Konsep Dasar Logika Fuzzy

Ketidakpastian merupakan dasar pemikiran dari logika fuzzy, dan bagaimana logika fuzzy mampu untuk mempresentasikan ketidakpastian yang ada pada suatu sistem pengendali berbasis logika fuzzy.

Logika Fuzzy mempunyai beberapa konsep dasar dan bagian penting yang harus diketahui yaitu:



Gambar 2-12: Bagian – bagian Fuzzy Logic
Sumber: Diktat Mata Kuliah, "Logika Fuzzy", ITN Malang

Keterangan Gambar:

- Crisp Input (masukan crisp)
 - Masukan yang tegas dan tertentu (seperti 20°), crisp input disini merupakan pernyataan nilai masukan dari sistem kontrol.

- Membership Function (fungsi keanggotaan)
 - Mendefinisikan fuzzy set dengan mentransformasikan atau memetakan masukan crisp dari domainnya ke derajat keanggotaan.
- Label
 - Nama deskriptif yang digunakan untuk mengidentifikasi sebuah fungsi keanggotaan
- Scope
 - Scope atau lingkup merupakan lebar fungsi keanggotaan. Jangkauan konsep, pada umumnya bilangan, tempat dimana fungsi keanggotaan dipetakan. Disini domain dari fuzzy set (fungsi keanggotaan) adalah dari 10^0 sampai 30^0 dan lingkungannya adalah 20^0
- Daerah Batasan Crisp
 - Jangkauan seluruh nilai yang mungkin dapat diaplikasikan pada variabel sistem atau sering disebut daerah semesta keseluruhan dari crisp input.

2.3.3 Membership Function (Fungsi Keanggotaan).

Pada himpunan *fuzzy*, semua informasi terkandung dalam fungsi keanggotaannya. Oleh karena itu penting untuk mengetahui bagian-bagian pada sebuah fungsi keanggotaan *fuzzy*.

Ada beberapa fungsi keanggotaan dalam logika *fuzzy*. Adapun yang sering digunakan adalah bentuk fungsi-S, fungsi- π , fungsi segitiga (*tringular*) dan fungsi trapesium (*trapeosida*).

Contoh berikut ini adalah 4 macam keanggotaan yang dinyatakan dengan :

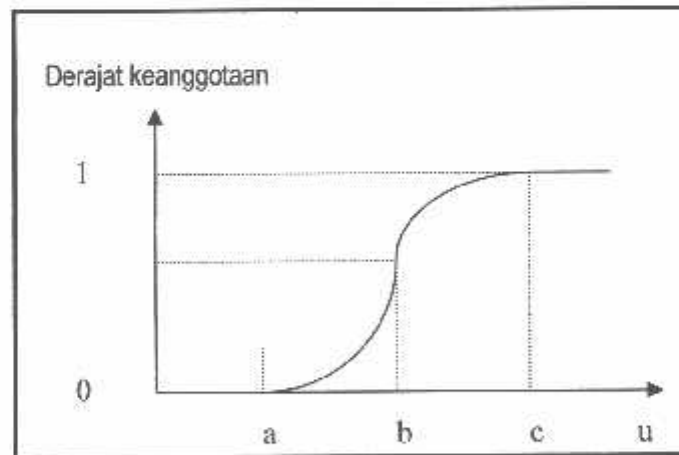
1. Fungsi Keanggotaan S (*S Membership Function*)
2. Fungsi Keanggotaan π (*π Membership Function*)
3. Fungsi Keanggotaan Segitiga (*Triangular Membership Function*)
4. Fungsi Keanggotaan Trapezium (*Trapezoidal Membership Function*)

1. S-Function

Definisi S-function adalah sebagai berikut :

$$S(u,a,b,c) = \begin{cases} 0 & u < a \\ 2\left(\frac{u-a}{c-a}\right) & a \leq u < b \\ 1 - 2\left(\frac{u-a}{c-a}\right) & b \leq u \leq c \\ 1 & u > c \end{cases}$$

Gambar grafik fungsi keanggotaannya ditunjukkan dalam gambar 2-13 :



Gambar 2-13 : S-Function

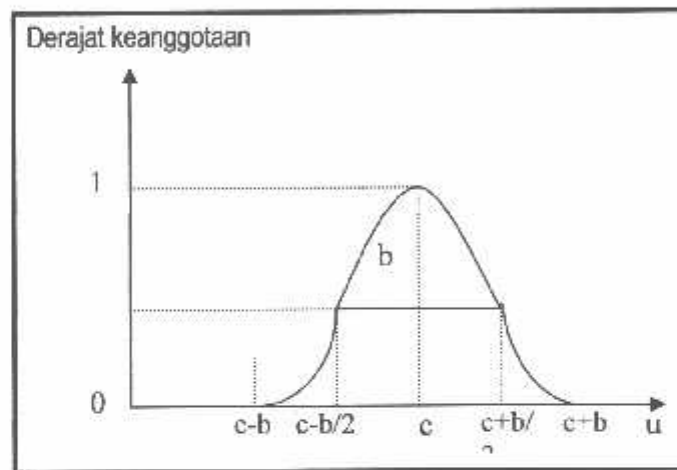
Sumber: Sri kusumadewi & Hari Purnomo, "Aplikasi Logika Fuzzy Untuk Pendukung Keputusan,"

2. π -Function

Definisi π -function adalah sebagai berikut :

$$\pi(u, b, c) = \begin{cases} S(u; c-b, c-\frac{b}{2}, c) & u \leq c \\ 1 - S(u; c+\frac{b}{2}, c+b) & u \geq c \end{cases}$$

Gambar grafik fungsi keanggotaannya ditunjukkan dalam gambar 2-14 :



Gambar 2-14: π -Function

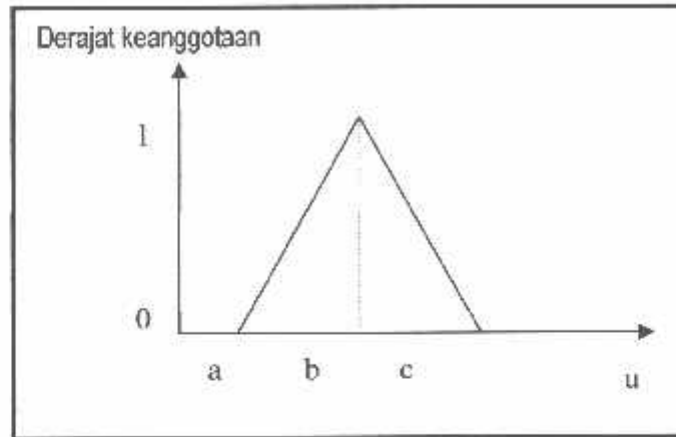
Sumber: Sri kusumadewi & Hari Purnomo "Aplikasi Logika Fuzzy Untuk Pendukung Keputusan,"

3. Triangular -Function.

Triangular-function didefinisikan sebagai berikut :

$$T(u; a, b, c) = \begin{cases} 0 & u < a \\ \frac{u-a}{b-a} & a \leq u \leq b \\ \frac{c-u}{c-b} & b \leq u \leq c \\ 0 & u > c \end{cases}$$

Gambar grafik fungsi keanggotaannya ditunjukkan dalam gambar 2-15 :



Gambar 2-15: T-Function

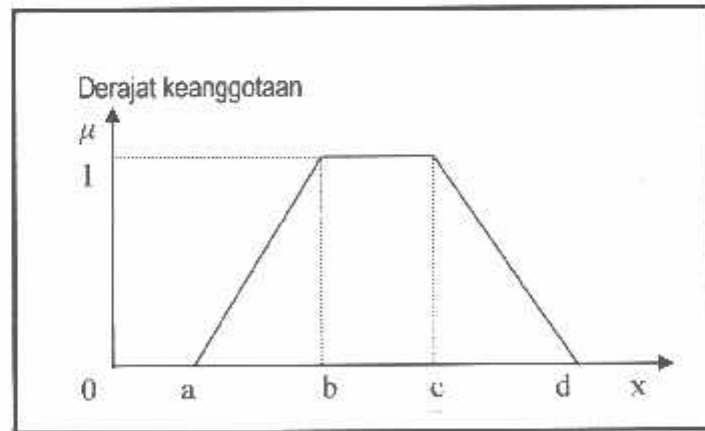
Sumber: Sri kusumadewi & Hari Purnomo "Aplikasi Logika Fuzzy Untuk Pendukung Keputusan,"

4. Trapezoidal - Function

Persamaan untuk bentuk *Trapezoidal - Function* ini adalah :

$$Z(x; a, b, c, d) = \begin{cases} 0 & \text{jika } x < a, x > d \\ 1 & \text{jika } b \leq x \leq c \\ (x-a)/(b-a) & \text{jika } a \leq x \leq b \\ (d-x)/(d-c) & \text{jika } c \leq x \leq d \end{cases}$$

Gambar grafik fungsi keanggotaannya ditunjukkan dalam gambar 2-16 :



Gambar 2-16: Trapezoidal -Function

Sumber: Sri kusumadewi & Hari Purnomo "Aplikasi Logika Fuzzy Untuk Pendukung Keputusan,"

2.3.4 Variabel Linguistik

Inti dari teknik pemodelan *fuzzy* adalah nama suatu himpunan *fuzzy* yang disebut variabel linguistik.

Pendekatan *fuzzy logic* dengan sistem kontrol merupakan sistem pengaturan yang menirukan cara kerja manusia dalam melakukan proses pengambilan keputusan melalui ungkapan-ungkapan kualitatif yang diindranya sebagai contoh, seorang operator yang mengatur suatu proses secara manual akan menggunakan ungkapan-ungkapan seperti sangat besar, sedang, mendekati maksimum, sekitar set poin dan sebagainya.

Dalam sistem kontrol variabel linguistik dapat dinyatakan dengan ungkapan linguistik:

- | | | |
|--|---|---------------------------------|
| <ol style="list-style-type: none"> 1. NB (<i>Negatif Big</i>) 2. NM (<i>Negatif Medium</i>) 3. NS (<i>Negatif Small</i>) 4. Z (<i>Zero</i>) 5. PS (<i>Positif Small</i>) 6. PM (<i>Positif Medium</i>) 7. PB (<i>Positif Big</i>) | } | Untuk variabel input dan output |
|--|---|---------------------------------|

Secara sederhana variabel linguistik dapat dinyatakan dengan pasangan $(\mu, T(\mu), X)$. μ menyatakan nama variabel dan $T(\mu)$ adalah istilah yang menyatakan seperangkat nama dari besaran linguistik untuk himpunan bagian *fuzzy* pada semesta pembicaraan X .

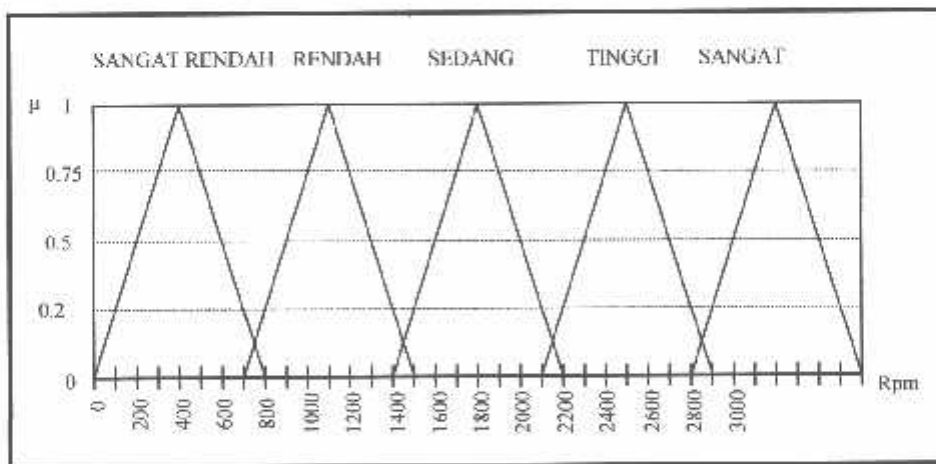
Sebagai contoh : Variabel kecepatan dipakai sebagai variabel linguistik, maka seperangkat istilah T (kecepatan) pada semesta pembicaraan X (kecepatan putaran motor), sehingga dapat dapat dituliskan :

$$\square T(\text{kecepatan}) = \{ \text{sangat rendah, rendah, sedang, tinggi, sangat tinggi} \}$$

Jika jelajah dari semesta pembicaraan X , dimisalkan dengan nilai [0:3600] rpm maka berlaku ketentuan seperti :

- ❶ Kecepatan putaran sangat rendah [0:800]
- ❷ Kecepatan putaran rendah [700:1500]
- ❸ Kecepatan putaran sedang {1400:2200}
- ❹ Kecepatan putaran tinggi [2100:2900]
- ❺ Kecepatan putaran sangat tinggi [2800:3600]

Fungsi keanggotaan untuk data diatas dapat digambarkan sebagai berikut :



Gambar 2-17 : Himpunan *fuzzy* untuk kecepatan putaran motor

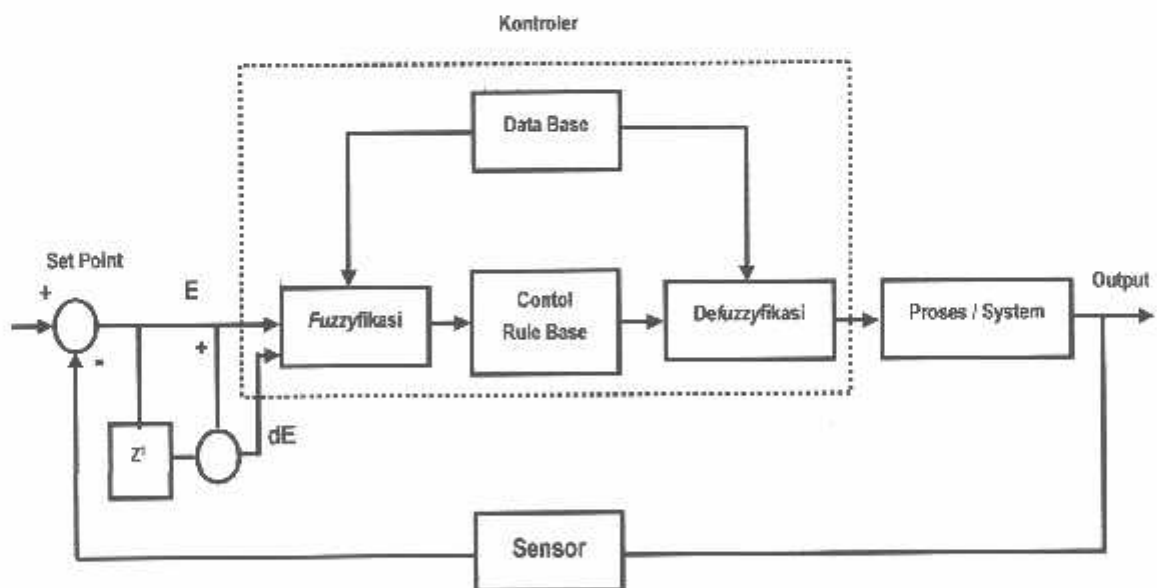
Sumber: Sri Kusumadewi, " *Aplikasi LOGIKA FUZZY Untuk Pendukung Keputusan* ", Penerbit GRAHA ILMU.

2.3.5 KONTROLER FUZZY

Sistem pengaturan cerdas ini dapat dirancang berdasarkan empat pendekatan yaitu :

1. Pendekatan berdasarkan basis pengetahuan (knowledge Base) seorang operator ahli.
2. Pendekatan fuzzy logic dengan variabel linguistik.
3. Pendekatan dengan jaringan syaraf tiruan (artificial neural network).
4. Gabungan antara artificial neural network dan fuzzy logic yang dikenal sebagai Neuro-fuzzy.

Secara umum Struktur Sistem Kontrol Logika *Fuzzy* dengan Error dan Delta Error adalah sebagai berikut:



Gambar 2-18: Struktur Sistem Kontrol Logika *Fuzzy* secara umum.
Sumber: Son Kuswadi, "Kendali Cerdas," EEPIS Press Surabaya

Kontroler *fuzzy* dalam gambar 2-18, mempunyai pengolahan dua sinyal masukan ke kontroler, yaitu sinyal kesalahan (*error* / E), dan sinyal perubahan kesalahan (*delta error* / dE).

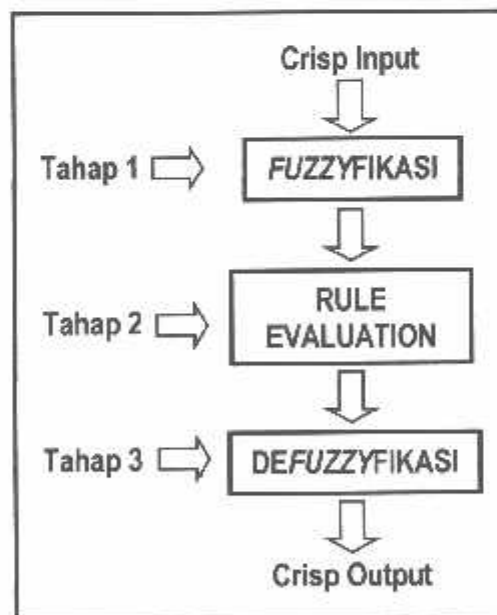
Sinyal E didapatkan dari pengurangan keluaran proses terhadap *set point*, sedangkan sinyal dE didapatkan dari pengurangan sinyal *error* saat ini $E(k)$, dengan sinyal *error* sebelumnya $E(k-1)$.

Kedua sinyal tersebut diolah kontroler *fuzzy*, dengan demikian terdapat dua masukan ke kontroler dan satu keluaran dari kontroler.

Pemecahan masalah dalam proses *fuzzy logic* melibatkan 3 tahap:

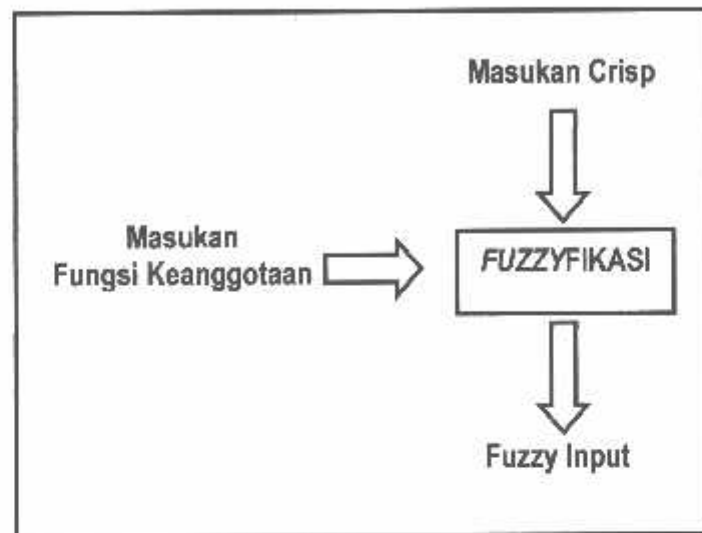
1. *Fuzzyfikasi*
2. *Rule Evaluation*
3. *Defuzzyfikasi*

Pemecahan masalah dalam proses *fuzzy logic* dapat digambarkan sebagai berikut :



Gambar 2-19 : Tahapan dalam proses *fuzzy logic*
 Sumber: Program Fuzzy Logic 2.0 by Motorola Tutorial

2.3.6 FUZZYFIKASI



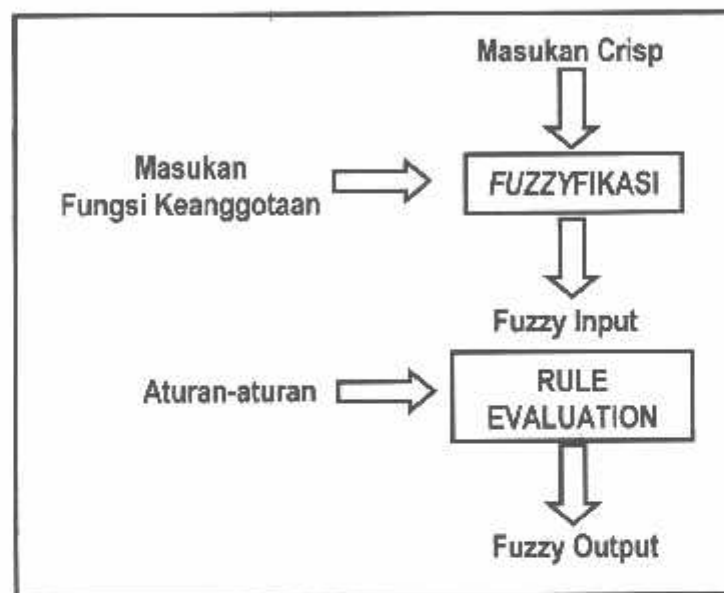
Gambar 2-20 : Proses Fuzzifikasi
 Sumber: Program Fuzzy Logic 2.0 by Motorola Tutorial

Proses fuzzifikasi merupakan proses untuk mengubah variabel non *fuzzy* (variabel non numerik) menjadi variabel *fuzzy* (variabel linguistik). Nilai masukan-masukan yang masih dalam bentuk variabel numerik yang telah dikuantisasi sebelum diolah oleh pengendali logika *fuzzy* harus diubah dahulu ke dalam variabel *fuzzy*. Melalui fungsi keanggotaan yang telah disusun maka nilai-nilai masukan tersebut menjadi informasi *fuzzy* yang berguna untuk proses pengolahan secara *fuzzy* pula.

Secara diagram blok dapat anda lihat pada gambar 2-32, Prosesnya adalah sebagai berikut: suatu besaran analog dimasukkan sebagai input (*crisp* input), lalu input tersebut dimasukkan pada batas *scope / domain* sehingga input tersebut dapat dinyatakan dengan label (NB,NM,NS,Z,PB,PM dan PS) dari *membership*

function. *Membership function* ini biasanya dinamakan *membership function* input.

2.3.7 PENENTUAN ATURAN (*RULE EVALUATION*)



Gambar 2-21: Proses *Fuzzyfikasi*

Sumber: Program Fuzzy Logic 2.0 by Motorola Tutorial

Pada proses *rule evaluation*, *fuzzy* menggunakan Variable linguistik untuk menentukan aksi kontrol yang harus dilakukan dalam merespon nilai *input* yang diberikan. Dalam *rule evaluation* ini, *rule base* yang telah dibuat sangat menentukan handal dan tidaknya *fuzzy* kontroler yang dirancang.

Rule Evaluation merupakan bagian yang menggambarkan dinamika suatu sistem terhadap masukan yang dikarakteristikkan oleh sekumpulan variabel-variabel linguistik dan berbasis pengetahuan seorang operator ahli. Pernyataan tersebut umumnya dinyatakan oleh suatu pernyataan bersyarat.

Dalam pembentukan basis aturan perlu diperhatikan aspek :

1. Variabel masukan dan keluaran error (E) dan perubahan error (dE) tetap menjadi acuan utama bagi aturan kontrol suatu sistem kontrol yang berbasis fuzzy logic ini sehingga akan menghasilkan aksi kontrol (u) yang sesuai.
2. Penurunan aturan kontrol fuzzy

Salah satu cara menurunkan aturan kontrol fuzzy adalah dengan mengumpulkan aturan-aturan kontrol fuzzy yang dibentuk dari analisa perilaku obyek atur. Aturan kontrol diturunkan dengan jalan mengkoreksi simpangan keluaran sistem menuju ke keadaan yang diinginkan.

Agar diperoleh kinerja sistem kontrol yang lebih baik digunakan 7 variabel linguistik yaitu

- | | | |
|--|---|---|
| <ol style="list-style-type: none"> 1. NB (<i>Negatif Big</i>) 2. NM (<i>Negatif Medium</i>) 3. NS (<i>Negatif Small</i>) 4. Z (<i>Zero</i>) 5. PS (<i>Positif Small</i>) 6. PM (<i>Positif Medium</i>) 7. PB (<i>Positif Big</i>) | } | <p>Untuk 2 variabel input (E & dE)
dan 1 output</p> |
|--|---|---|

Penyempurnaan aturan kontrol terus menerus dilakukan oleh para ahli. Mac Vicar Whelan yang meneliti masalah penyempurnaan basis aturan menemukan pola hubungan antara E ,dE dan U.

Penurunan lengkap dari aturan kontrol in ditunjukkan dalam tabel yang disebut sebagai tabel Mac Vicar Whelan. Hasil penyempurnaan ini sangat dianjurkan oleh operator ahli.

Tabel Aturan Kontrol Fuzzy Macvicar-Whelan

E							
NB	NB	NB	NB	NB	NB	NB	PS
NM	NB	NM	NM	NM	NS	Z	PM
NS	NB	NM	NM	Z	PM	PM	PM
Z	NM	NM	NM	Z	PM	PM	PM
PS	NM	NM	NS	PS	PM	PM	PB
PM	NM	Z	PS	PM	PM	PB	PB
PB	NS	PB	PB	PB	PB	PB	PB

2.3.7.1 Bentuk Aturan Kontrol Fuzzy

Seorang operator ahli yang berpengalaman akan menggunakan aturan kontrol yang berbentuk if-then.

a) SISO (Single Input Single Output)

→ Untuk sistem dengan 1 masukan dan 1 keluaran

Aturan kontrolnya adalah :

- Aturan ke-1: Jika A adalah x_1 maka B adalah y_1 .
-
-
- Aturan ke-n : Jika A adalah x_n maka B adalah y_n

dimana:

A = Variabel masukan

B = Variabel keluaran

x,y = Variabel linguistik

b) MISO (Multi Input Single Output)

Sedangkan untuk sistem banyak masukan satu keluaran aturan kontrolnya adalah :

- Aturan ke-1 : Jika A adalah x_1 dan B adalah y_1 maka C adalah z_1
 -
 -
 - Aturan ke-n : Jika A adalah x_n dan B adalah y_n maka C adalah z_n
-

dimana :

$A \& B$ = Variabel masukan (misalnya: E dan dE)

$x \& y$ = Himpunan *fuzzy* dengan variabel linguistik

(misalnya : NB, NM, NS, Z, PS, PM, PB)

C = variabel keluaran,

Z = himpunan fuzzy untuk z (yaitu NB, NS, NM, Z, PS, PM PB)

2.3.7.2 Mekanisme Inferensi

Untuk mengkombinasikan aturan-aturan kontrol yang digunakan kata hubung yang secara umum dinyatakan dengan :

- Rule 1 : Jika E adalah A_1 dan dE adalah B_1 maka u adalah C_1
- Rule 2 : Jika E adalah A_2 dan dE adalah B_2 maka u adalah C_2
-
-
- Rule n : Jika E adalah A_n dan dE adalah B_n maka u adalah C_n

Maksud dari pernyataan diatas :

E = Variabel masukan error yang mempunyai variable linguistik

$A (A_1, \dots, A_n)$

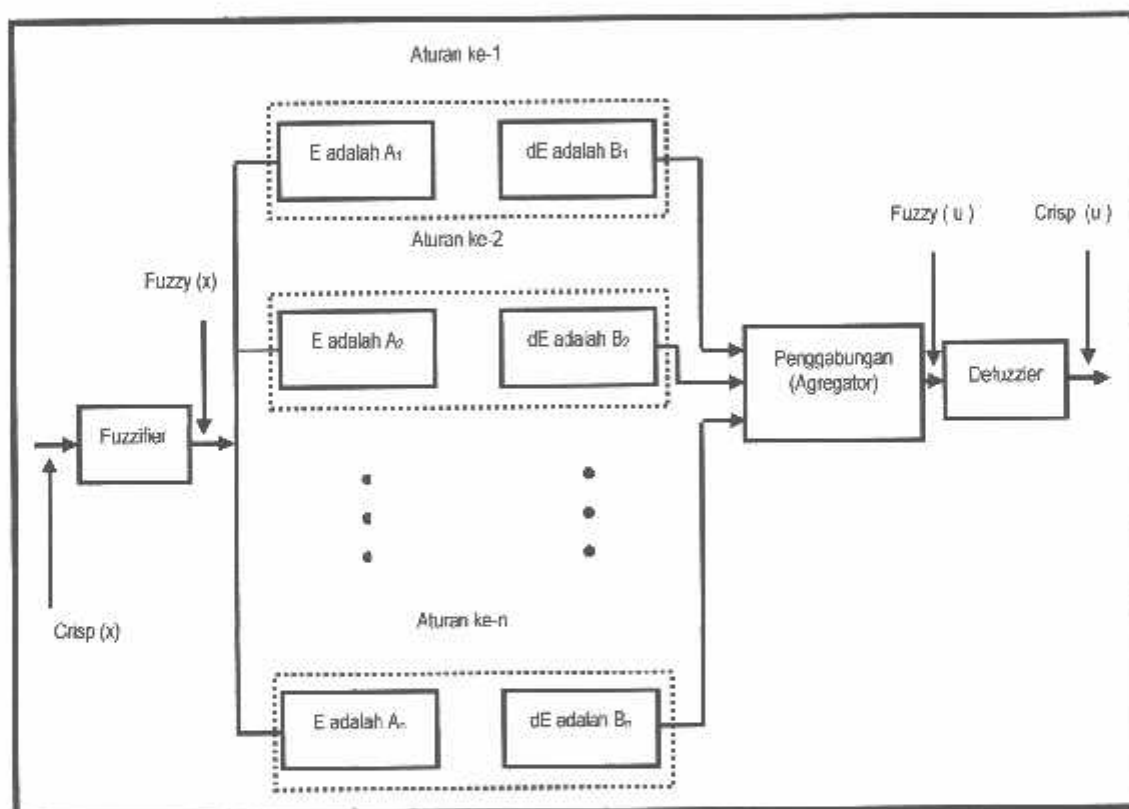
dE = Variabel masukan perubahan error yang mempunyai variable linguistik

$B (B_1, \dots, B_n)$

u = Variabel keluaran yang mempunyai variabel linguistik $C (C_1, \dots, C_n)$

Keseluruhan aturan kontrol ini harus dapat menghasilkan satu keluaran yaitu berupa kesimpulan. Proses pengambilan kesimpulan ini dinyatakan sebagai **"MEKANISME INFERENSI"**.

Dari uraian diatas, *Mekanisme Inferensi* yang terdiri dari beberapa aturan-aturan dapat digambarkan sebagai berikut :

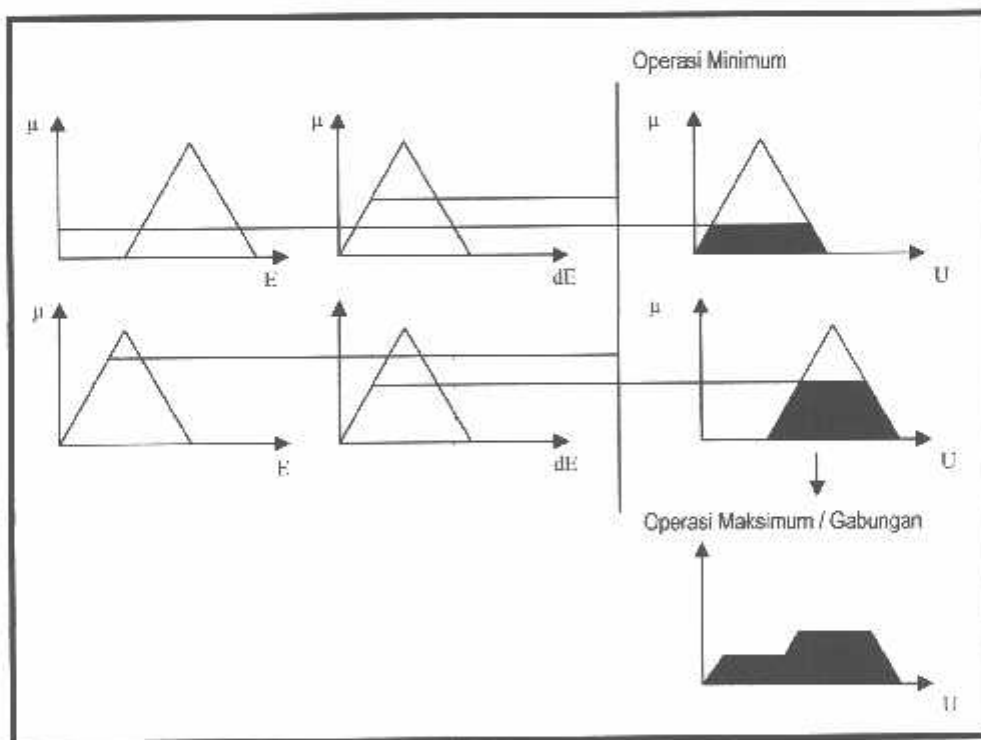


Gambar 2-21: Proses Mekanisme Inferensi
 Sumber: Jun Yan, dkk., 1994

Metode *Inferensi* yang banyak digunakan adalah Metode Inferensi Mamdani, seorang pakar sistem kontrol dari Queen Mari Colledge di London yang mempunyai nama lengkap *Ebrahim H. Mamdani*.

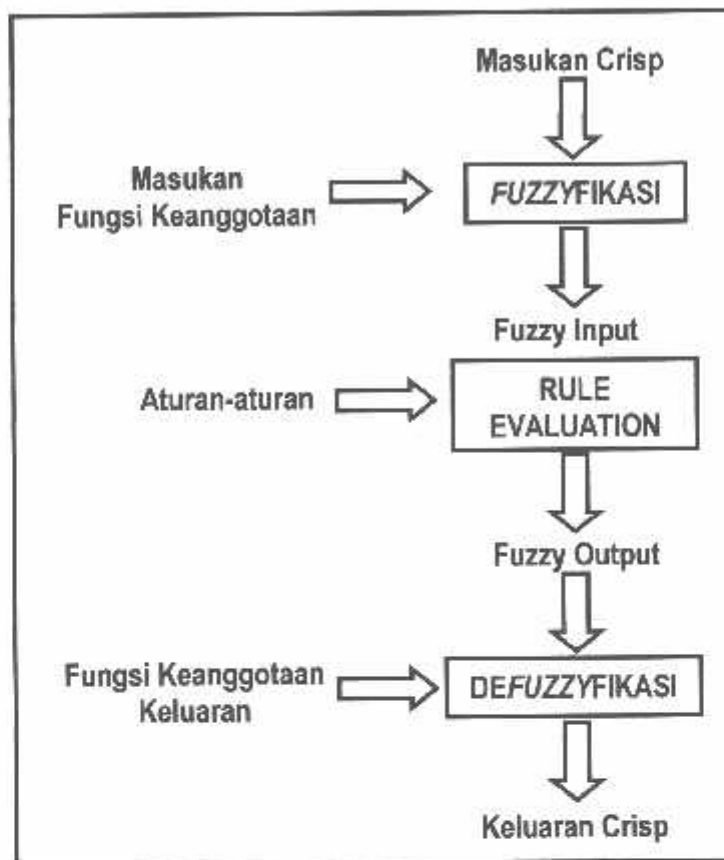
Metode *Inferensi Mamdani* ini pertama kali dipakai untuk mengontrol kombinasi ketel dan mesin uap dengan menggunakan aturan kontrol dan variabel linguistik yang didapat dari operator yang berpengalaman.

Metode *Inferensi* untuk *MISO* (*Multi Input Satu Output*) dari Mamdani yang digunakan adalah Metode Maksimum-Minimum (*Max-Min Method*). Penalaran fuzzy dengan menggunakan Metode Maksimum-Minimum (*Max-Min Method*) dapat digambarkan sebagai berikut.



Gambar 2-22 : Proses penalaran *fuzzy* MISO dengan Operasi Max-Min
Sumber: Jun Yan, dkk., 1994

2.3.8 Defuzzyfikasi



Gambar 2-22: Proses Defuzzyfikasi
 Sumber: Program Fuzzy Logic 2.0 by Motorola Tutorial

Defuzzyfikasi merupakan proses pemetaan himpunan *fuzzy* ke himpunan *crisp*. Proses ini merupakan kebalikan dari proses *fuzzyfikasi* berfungsi untuk mengubah keluaran linguistik ke dalam keluaran *crisp*.

Proses ini berfungsi untuk menentukan suatu nilai *crisp* output. Prosesnya adalah sebagai berikut: suatu nilai *fuzzy* output yang berasal dari *rule evaluation* diambil kemudian dimasukkan ke dalam suatu *membership function* output.

Metode defuzzyfikasi yang umum digunakan adalah "**Centroid Of Area Method**"

Metode defuzzyfikasi dengan centroid of area method mempunyai persamaan :

1. SISO (*Single Input Single Output*) , berlaku :

$$U = \frac{\sum U_i \cdot \mu_i}{\sum \mu_i}$$

2. MISO (*Multi Input Single Output*) , berlaku :

$$U = \frac{\sum U_i \cdot \mu_i}{\sum \mu_i}$$

dimana :

$i = 1, 2, 3, \dots, 49$ menyatakan jumlah aturan kontrol MISO (7 x 7)

$U_i =$ Titik pusat masing-masing himpunan variabel keluaran ke-I

$\mu_i =$ Variabel minimum untuk tiap *variabel linguistik*.

2.3.9 DESAIN KONTROLLER

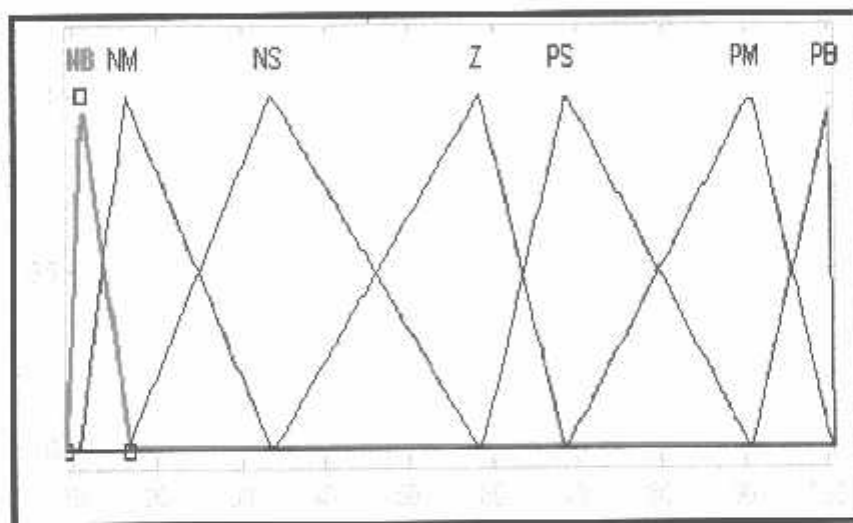
Pada kontroler logika fuzzy konvensional penentuan fungsi keanggotaan (*membership function*) dilakukan berdasarkan pada pengalaman dan pengetahuan perancangannya.

Pada kontroler logika fuzzy di sini pembentukan fungsi keanggotaan (*membership function*) logika fuzzy dilakukan melalui jaringan syaraf tiruan yang berupa jaringan kohonen dan fungsi keanggotaan (*membership function*) logika fuzzy yang terbentuk nantinya merupakan hasil pemetaan input kontroler yang dilakukan oleh *Jaringan Kohonen*.

2.3.9.1 Bentuk Fungsi keanggotaan (*Membranship Function*)

Keanggotaan logika fuzzy yang terbentuk nantinya merupakan hasil pemetaan input kontroler yang dilakukan oleh *Jaringan Kohonen*. Pada kontroler logika fuzzy ini fungsi keanggotaan (*membranship function*) yang digunakan berbentuk segitiga (*triangular*), seperti tampak pada gambar 2- 37.

Penentuan fungsi keanggotaan (*membranship function*) dilakukan dengan menentukan titik tengah segitiga.



Gambar 2-37 : Fungsi keanggotaan segitiga logika fuzzy
Sumber : Jurnal Politeknik Elektronika Negeri Surabaya- ITS.

2.3.9.2 Proses Pelatihan

Proses pelatihan merupakan bagian utama dari pembentukan kontroler logika fuzzy, yang berfungsi untuk menentukan fungsi keanggotaan (*membranship function*) dilakukan dengan menentukan titik tengah segitiga yaitu dengan cara mengelompokkan data hasil pensamplingan berupa *Jaringan Kohonen*.

Jumlah input pada layer input sesuai dengan jumlah input pada kontroler logika fuzzy yang dalam hal ini adalah Error (E) dan perubahan error (dE), sedangkan jumlah output = jumlah fungsi keanggotaan (*membranship function*) input logika fuzzy.

Data untuk melatih *Jaringan Kohonen* ini dapat diperoleh dari *running loop* terbuka atau *running loop* tertutup sistem. Data *loop* terbuka, sudah dapat digunakan untuk membentuk fungsi keanggotaan, tetapi jika respon yang diperoleh belum memuaskan, maka dengan menggunakan hasil pelatihan ini dapat diperoleh data *loop* tertutup, yang selanjutnya digunakan untuk memperbaiki fungsi keanggotaan yang ada.

Dalam proses pelatihan ini, penentuan jumlah fungsi keanggotaan (*membranship function*) dan laju pelatihan serta jumlah output *Jaringan Kohonen* masih harus ditentukan lagi.

2.4 Alogaritma Proses Pembelajaran Jaringan Syaraf Tiruan

Kohonen :

Alogaritma dilakukan berdasarkan proses pembelajaran pada jaringan syaraf tiruan berupa *Jaringan Kohonen*.

Banyaknya kelas untuk jaringan staraf ini adalah sebanyak 7 buah kelas yang mewakili fungsi keanggotaan (*membranship function*) dari kontroler logika fuzzy. Jaringan kohonen ini digunakan untuk menentukan nilai tengah dari tiap-tiap fungsi keanggotaan (*membranship function*) dari kontroler logika fuzzy.

Masukan untuk kontroler logika *fuzzy* ini adalah *error* (E) dan *perubahan error* (dE) dari kecepatan sudut putar dari motor induksi dalam satuan radian/detik. Dari data yang diperoleh tersebut nantinya akan digunakan sebagai data untuk pembelajaran jaringan kohonen.

Alogaritma Pembelajaran Jaringan Syaraf Tiruan Kohonen:

1. Inisialisasi bobot input dalam hal ini adalah Error (E) dan Delta Error (DE) untuk kecepatan sudut putar 100 rad/s dan 120 rad/s.
2. Hitung bobot input:

$$W_{ij} = \frac{MinE_j + MaxE_j}{2} \quad \text{dan} \quad W_{ij} = \frac{MinDE_j + MaxDE_j}{2}$$

dengan w_{ij} adalah bobot antara variable input ke- j ($j = 1, 2, 3, \dots, m$) dengan neuron pada kelas ke- i ($i = 1, 2, 3, \dots, K$); $MinE_j$ dan $MinDE_j$ serta $MaxE_j$ dan $MaxDE_j$ adalah masing-masing nilai terkecil pada variable input ke- i dan nilai terbesar dari variable input ke- i .

3. Hitung bobot bias:

$$b_i = e^{[1 - \ln(1/K)]}$$

dengan b_i adalah bobot bias ke neuron ke- i dan K adalah jumlah kelas.

4. Set parameter learning rate (α) = 0.001

Set maximum epoch (Max Epoch) = 500 Epoch

Set Epoch = 0

5. IF Epoch < MaxEpoch; jika **ya** maka:

a. Epoch = Epoch + 1

- b. Pilih data secara acak (random), misalkan data yang terpilih adalah data ke-z.
 c. Hitung jarak antara data ke-z dengan setiap bobot input ke-i (D_i):

$$D_i = \sqrt{\sum_{j=1}^m (w_{ij} - p_{zj})^2}$$

- d. Hitung: $a_i = -D_i + b_i$; setiap jarak ini dinegatifkan dan ditambahkan dengan bobot biasnya, Misalkan hasilnya = a_i
 e. Cari a_i terbesar: (i). $\text{MaxA} = \max(a_i)$, dengan $i = 1, 2, 3, \dots, K$
 (ii). $\text{Idx} = i$, sedemikian hingga $a_i = \text{MaxA}$
 dicari a_i terbesar, misalkan terletak pada $\text{Idx} = i$, dengan nilai terbesar adalah $\text{MaxA} = \text{Max}(a_i)$, maka neuron ke-i akan menjadi pemenang.

- f. Set Output Neuron ke-i (F_i): $F_i = \begin{cases} 1, & i = \text{Idx} \\ 0, & i \neq \text{Idx} \end{cases}$

- h. Update bobot yang menuju Neuron ke Idx :

$$W_{\text{Idx}-j} = W_{\text{Idx}-j} + \alpha (p_{zj} - W_{\text{Idx}-j})$$

- i. Update bobot bias:

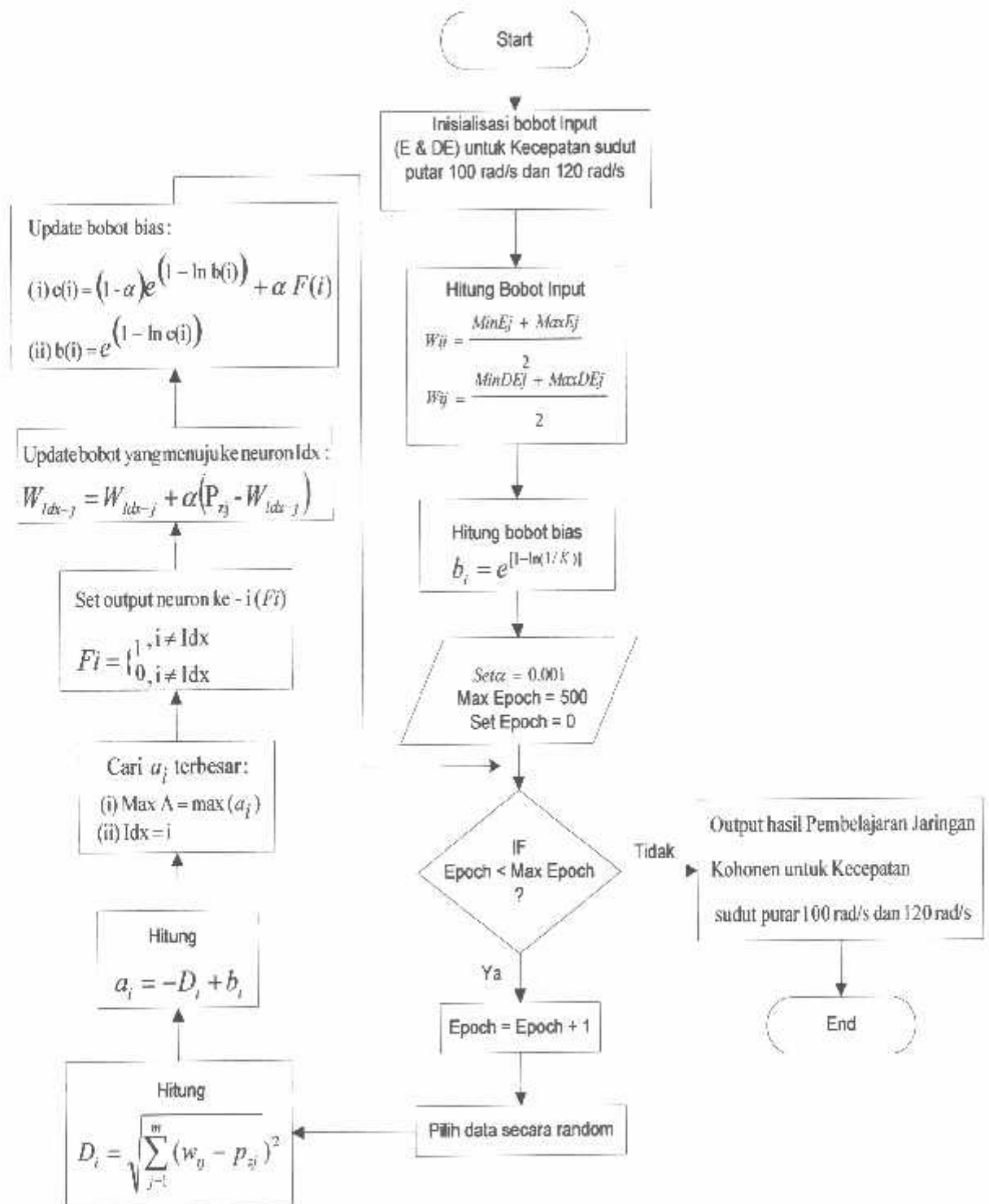
$$(i) c(i) = (-\alpha) e^{1 - \ln b(i)} + \alpha F(i)$$

$$(ii) b(i) = e^{(1 - \ln c(i))}$$

6. IF Epoch < MaxEpoch; jika **tidak** maka menuju ke langkah ke-7.

7. Output hasil neuron pemenang pada tiap-tiap kelas untuk kecepatan sudut putar 100 rad/s dan 120 rad/s.

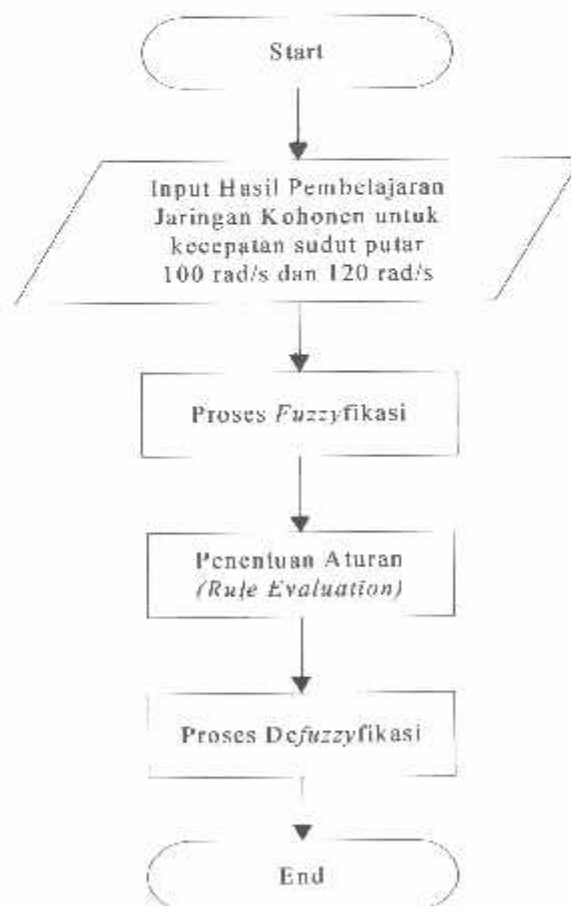
2.5 Flowchart Proses Pembelajaran Jaringan Syaraf Tiruan Kohonen



2.6 Alogaritma proses pembentukan FIS (Fuzzy Infermce System)

1. Start
2. Input Hasil Pembelajaran *Jaringan Kohonen* untuk kecepatan sudut putar 100 rad/s dan 120 rad/s
3. Proses *Fuzzyfikasi*
4. Penentuan Aturan (*Rule Evaluation*)
5. Proses *Defuzzyfikasi*
6. End

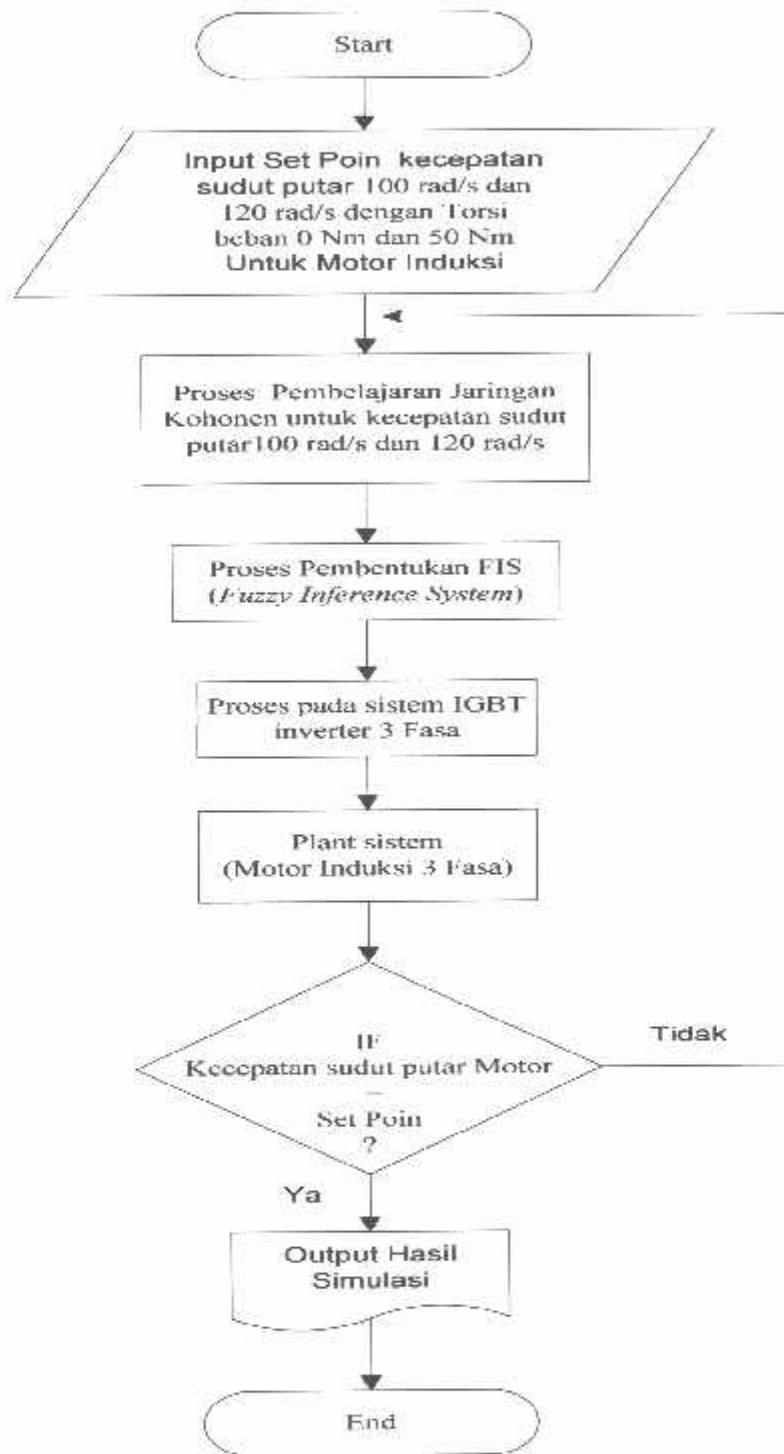
2.7 Flowchart Proses Pembentukan FIS (Fuzzy Inference System)



2.8 Algoritma Pemodelan System

1. Start
 2. Input Set Point kecepatan sudut putar 100 rad/s dengan torsi beban 0 Nm dan 50 Nm serta kecepatan sudut putar 120 rad/s dengan torsi beban 0 Nm dan 50 Nm untuk motor induksi 3 fasa
 3. Proses Pembelajaran *Jaringan Kohonen* dengan kecepatan sudut putar 100 rad/s dengan torsi beban 0 Nm dan 50 Nm serta kecepatan sudut putar 120 rad/s dengan torsi beban 0 Nm dan 50 Nm.
 4. Proses pembentukan FIS (*Fuzzy Inference System*)
 5. Proses pada sistem IGBT Inverter 3 fasa
 6. Proses pada Motor Induksi 3 fasa (*Plant sistem*)
 7. Jika kecepatan sudut putar motor induksi 3 fasa = Set Point, jika **tidak**, maka kembali ke point 3.
 8. Jika kecepatan sudut putar motor induksi 3 fasa = Set Point, jika **ya**, maka dilanjutkan ke point 9
 9. Output hasil simulasi dengan menggunakan perangkat lunak Mallab 7.0
 10. End
-

2.9 FLOWCHART SISTEM



BAB III

PERANCANGAN KONTROLER

PADA JARINGAN SYARAF TIRUAN KOHONEN

3.1 Perencanaan Kontroler pada Jaringan Syaraf Tiruan Kohonen

Perencanaan kontroler ini menggunakan logika *fuzzy* berdasarkan pembelajaran yang telah dilakukan oleh jaringan syaraf tiruan berupa *Jaringan Kohonen*.

Banyaknya kelas untuk jaringan syaraf ini adalah 7 buah kelas yang mewakili 7 fungsi keanggotaan (*membership function*) dari kontroler logika *fuzzy*. *Jaringan Kohonen* ini digunakan untuk menentukan nilai tengah dari tiap-tiap fungsi keanggotaan (*membership function*) *fuzzy*.

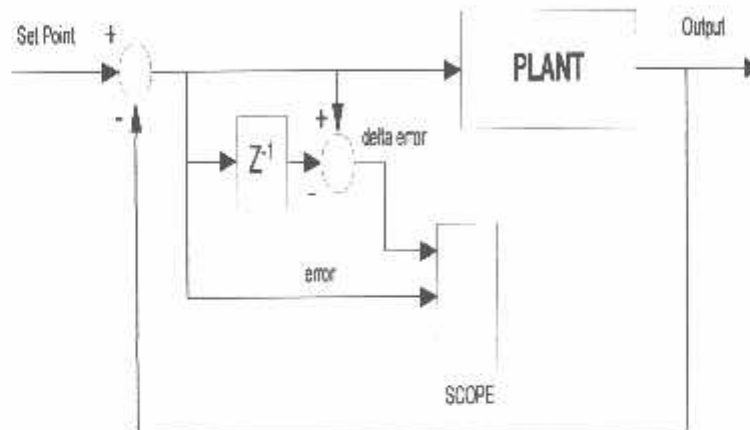
Masukan untuk kontroler logika *fuzzy* ini adalah *error* (E) dan *perubahan error* (dE) dari kecepatan sudut putar dari motor induksi dalam satuan radian/detik (*rad/s*). Dari data masukan (E dan dE) yang diperoleh melalui proses pensamplingan tersebut, nantinya akan digunakan sebagai data untuk pembelajaran *Jaringan Kohonen*.

Tahap-tahap yang dilakukan dalam perancangan *fuzzy* pada jaringan syaraf tiruan yang berupa *Jaringan Kohonen* ini meliputi 3 tahap, antara lain :

- 1) Pengambilan Data *Training*
 - 2) Pembelajaran *Jaringan Kohonen*
 - 3) Pembentukan *Fuzzy Inference System (FIS)*
-

3.2 PENGAMBILAN DATA *TRAINING*

Data masukan (*input*) yang digunakan sebagai data *training* adalah data *error* (E) dan *perubahan error* (dE) serta data keluaran (*output*) *fuzzy* yang diinginkan. Pengambilan data *training* terlihat pada gambar 3-1.



Gambar 3-1: Pengambilan data *training*

Sumber : Perancangan

Data yang didapatkan berjumlah 600 pasangan data yaitu *error* (E) dan *perubahan error* (dE) melalui proses pensamplingan.

Pensamplingan data meliputi:

- 1) Pensamplingan data untuk kecepatan putar 100 rad/s dengan torsi beban 0 Nm dan 50 Nm.
- 2) Pensamplingan data untuk kecepatan putar 120 rad/s dengan torsi beban 0 Nm dan 50 Nm.

Pasangan data yang diperoleh terdiri dari 300 pasangan data untuk beban kecepatan 100 rad/s, 300 lainnya untuk kecepatan 120 rad/s .

Cuplikan data yang telah didapatkan dari proses pensamplingan dapat dilihat dalam tabel 3.1 dan tabel 3.2 untuk kecepatan motor 100 rad/s dengan torsi beban 0 Nm dan 50 Nm serta tabel 3.3 dan 3.4 untuk kecepatan 120 rad/s dengan torsi beban 0 Nm dan 50 Nm.

Tabel 3.1. Cuplikan sinyal *error* dan *delta error* untuk pembelajaran jaringan kohonen pada kecepatan sudut putar 100 rad/s dengan beban 0 Nm

Time(detik)	D_Error(rad/s)	Error(rad/s)
0	100	100
0.1	-0.00066	94.24
0.2	-0.00057	88.785
0.3	-0.00036	83.735
0.4	-0.00058	79.055
0.5	-0.00041	74.706
0.6	-0.00023	70.669
0.7	-0.00024	66.912
0.8	-0.00037	63.407
0.9	-0.00031	60.139
1	-0.00012	57.09

Sumber: Hasil simulasi

*) Data selengkapnya dapat dilihat pada lampiran.

Tabel 3.2. Cuplikan sinyal *error* dan *delta error* untuk pembelajaran jaringan kohonen pada kecepatan sudut putar 100 rad/s dengan beban 50 Nm

Time(detik)	D_Error(rad/s)	Error(rad/s)
0	100	100
0.1	-0.0003	97.151
0.2	-0.00039	94.414
0.3	-0.00011	91.898
0.4	-0.00021	89.567
0.5	-3.56E-05	87.402
0.6	-0.00012	85.401
0.7	-0.00015	83.545
0.8	-0.00033	81.815
0.9	-0.00032	80.199
1	1.00E-05	78.689

Sumber: Hasil simulasi

*) Data selengkapnya dapat dilihat pada lampiran.

Tabel 3.3. Cuplikan sinyal *error* dan *delta error* untuk pembelajaran jaringan kohonen pada kecepatan sudut putar 120 rad/s dengan beban 0 Nm

Time(detik)	D_Error(rad/s)	Error(rad/s)
0	120	120
0.1	-0.03121	113.48
0.2	-0.00245	107.48
0.3	0.004438	102.11
0.4	-0.03106	98.03
0.5	0.023633	94.557
0.6	-0.00605	92.145
0.7	0.0097	89.874
0.8	-0.01859	86.894
0.9	0.015924	84.624
1	-0.01654	82.987

Sumber: Hasil simulasi

*) Data selengkapnya dapat dilihat pada lampiran.

Tabel 3.4. Cuplikan sinyal *error* dan *delta error* untuk pembelajaran jaringan kohonen pada kecepatan sudut putar 120 rad/s dengan beban 50 Nm

Time(detik)	D_Error(rad/s)	Error(rad/s)
0	120	120
0.1	-0.00037	116
0.2	-0.00052	112.17
0.3	-0.0002	108.65
0.4	-0.00048	105.39
0.5	-0.0004	102.35
0.6	-0.0003	99.529
0.7	-0.00012	96.897
0.8	-0.0004	94.459
0.9	-0.0001	92.175
1	-0.0004	90.061

Sumber: Hasil simulasi

*) Data selengkapnya dapat dilihat pada lampiran.

3.3 PEMBELAJARAN *JARINGAN KOHONEN*

Pembelajaran jaringan ini dilakukan untuk mendapatkan nilai tengah dari membership *fuzzy* dengan cara mengelompokkan data hasil pensamplingan.

3.3.1 Proses pembelajaran *Jaringan Kohonen* untuk kecepatan 100 rad/s dengan torsi beban 0 nm dan 50 nm adalah

```

bobot_input =
    0  0
    0  0
    0  0
    0  0
    0  0
    0  0
    0  0

bobot_bias_input =
    19.0280
    19.0280
    19.0280
    19.0280
    19.0280
    19.0280
    19.0280

TRAINR, Epoch 0/500
TRAINR, Epoch 25/500
TRAINR, Epoch 50/500
TRAINR, Epoch 75/500
TRAINR, Epoch 100/500
TRAINR, Epoch 125/500
TRAINR, Epoch 150/500
TRAINR, Epoch 175/500
TRAINR, Epoch 200/500
TRAINR, Epoch 225/500
TRAINR, Epoch 250/500
TRAINR, Epoch 275/500
TRAINR, Epoch 300/500
TRAINR, Epoch 325/500
TRAINR, Epoch 350/500
TRAINR, Epoch 375/500
TRAINR, Epoch 400/500
TRAINR, Epoch 425/500
TRAINR, Epoch 450/500
TRAINR, Epoch 475/500
TRAINR, Epoch 500/500
TRAINR, Maximum epoch reached
F =

Columns 1 through 9
    6  6  6  7  7  7  7  7  7

Columns 10 through 18
    2  2  4  4  4  4  4  4  4

Columns 19 through 27
    4  4  4  4  4  4  4  4  4

Columns 28 through 36
    4  4  4  4  4  4  4  4  4

Columns 37 through 45
    3  3  3  3  3  3  3  3  3

Columns 46 through 54
    3  3  3  3  3  3  3  3  3

Columns 55 through 63
    3  3  3  3  3  3  3  3  3

Columns 64 through 72
    3  3  3  3  5  5  5  5  5

Columns 73 through 81
    5  5  5  5  5  5  5  5  5

Columns 82 through 90
    5  5  5  5  5  5  5  5  5

Columns 91 through 99
    5  5  5  5  5  5  5  5  5

Columns 100 through 108
    5  5  6  6  6  6  6  6  6

```

Columns 109 through 117

7 7 7 7 7 7 7 7 7

Columns 118 through 126

7 7 7 7 7 7 7 7 7

Columns 127 through 135

7 7 7 7 7 7 7 7 7

Columns 136 through 144

7 7 7 7 7 7 7 7 2

Columns 145 through 153

2 2 2 2 2 2 2 2 2

Columns 154 through 162

2 2 2 2 2 2 2 2 2

Columns 163 through 171

2 2 2 2 2 2 2 2 2

Columns 172 through 180

2 2 2 2 2 2 2 2 2

Columns 181 through 189

2 2 2 2 2 2 2 2 2

Columns 190 through 198

2 2 2 2 2 2 2 2 2

Columns 199 through 207

2 2 2 2 6 6 6 6 6

Columns 208 through 216

6 6 6 6 6 6 6 6 6

Columns 217 through 225

6 6 6 6 6 6 6 6 6

Columns 226 through 234

6 6 6 1 1 1 1 1 1

Columns 235 through 243

1 1 1 1 1 1 1 1 1

Columns 244 through 252

1 1 1 1 1 1 1 1 1

Columns 253 through 261

1 1 1 1 1 1 1 1 1

Columns 262 through 270

1 1 1 1 1 1 1 1 1

Columns 271 through 279

1 1 1 1 1 1 1 1 1

Columns 280 through 288

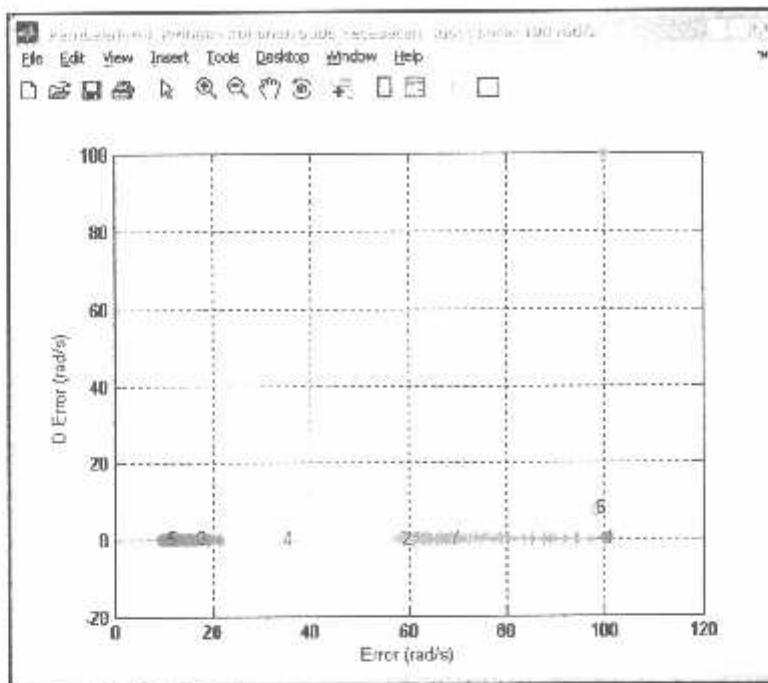
1 1 1 1 1 1 1 1 1

Columns 289 through 297

1 1 1 1 1 1 1 1 1

Columns 298 through 303

1 1 1 1 1 1



Gambar 3-2 : Grafik hasil pembelajaran jaringan kohonen pada kecepatan sudut putar 100 rad/s
Sumber : Simulasi

Keterangan Gambar :

- = Menunjukkan kelas ke-1 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
- = Menunjukkan kelas ke-2 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
- = Menunjukkan kelas ke-3 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
- = Menunjukkan kelas ke-4 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
- = Menunjukkan kelas ke-5 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
- = Menunjukkan kelas ke-6 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
- = Menunjukkan kelas ke-7 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.

3.3.2 Proses pembelajaran *Jaringan Kohonen* untuk kecepatan 120 rad/s dengan torsi

beban 0 Nm dan 50 Nm adalah

bobot_input =

```
0 0
0 0
0 0
0 0
0 0
0 0
0 0
```

bobot_bias_input =

```
19.0280
19.0280
19.0280
19.0280
19.0280
19.0280
19.0280
```

```
TRAINR, Epoch 0/500
TRAINR, Epoch 25/500
TRAINR, Epoch 50/500
TRAINR, Epoch 75/500
TRAINR, Epoch 100/500
TRAINR, Epoch 125/500
TRAINR, Epoch 150/500
TRAINR, Epoch 175/500
TRAINR, Epoch 200/500
TRAINR, Epoch 225/500
TRAINR, Epoch 250/500
TRAINR, Epoch 275/500
TRAINR, Epoch 300/500
TRAINR, Epoch 325/500
TRAINR, Epoch 350/500
TRAINR, Epoch 375/500
TRAINR, Epoch 400/500
TRAINR, Epoch 425/500
TRAINR, Epoch 450/500
TRAINR, Epoch 475/500
TRAINR, Epoch 500/500
TRAINR, Maximum epoch reached.
```

F =

Columns 1 through 9

```
2 1 1 6 2 2 2 2 4
```

Columns 10 through 18

```
4 4 4 4 4 4 4 4 4
```

Columns 19 through 27

```
4 4 4 4 4 4 4 4 4
```

Columns 28 through 36

```
4 4 4 4 4 4 4 4 4
```

Columns 37 through 45

```
4 4 4 4 4 4 4 4 7
```

Columns 46 through 54

```
7 7 7 7 7 7 7 7 7
```

Columns 55 through 63

```
7 7 7 7 7 7 7 7 7
```

Columns 64 through 72

```
7 7 7 7 7 7 7 7 7
```

Columns 73 through 81

```
7 7 7 7 7 3 3 3 3
```

Columns 82 through 90

```
3 3 3 3 3 3 3 3 3
```

Columns 91 through 99

```
3 3 3 3 3 3 3 3 3
```

Columns 100 through 108

```
3 3 2 1 1 1 1 6 2
```

Columns 109 through 117

```
2 2 2 2 4 4 4 4 4
```

Columns 118 through 126

```
4 4 4 4 4 4 4 7 7
```

Columns 127 through 135

```
7 7 3 3 3 3 3 3 3
```

Columns 136 through 144

```
3 3 3 3 3 3 3 3 3
```

Columns 145 through 153

```
3 3 3 3 5 5 5 5 5
```

Columns 154 through 162

```
5 5 5 5 5 5 5 5 5
```

Columns 163 through 171

```
5 5 5 5 5 5 5 5 5
```

Columns 172 through 180

5 5 5 5 5 5 5 5 5

Columns 181 through 189

5 5 5 5 5 5 5 5 5

Columns 190 through 198

5 5 5 5 5 5 5 5 5

Columns 199 through 207

5 5 5 5 2 1 1 1 1

Columns 208 through 216

1 1 1 1 1 1 1 1 1

Columns 217 through 225

1 1 1 1 1 1 1 1 1

Columns 226 through 234

1 1 1 1 1 1 1 1 1

Columns 235 through 243

1 1 1 1 1 1 1 1 6

Columns 244 through 252

6 6 6 6 6 6 6 6 6

Columns 253 through 261

6 6 6 6 6 6 6 6 6

Columns 262 through 270

6 6 6 6 6 6 6 6 6

Columns 271 through 279

6 6 6 6 6 6 6 6 6

Columns 280 through 288

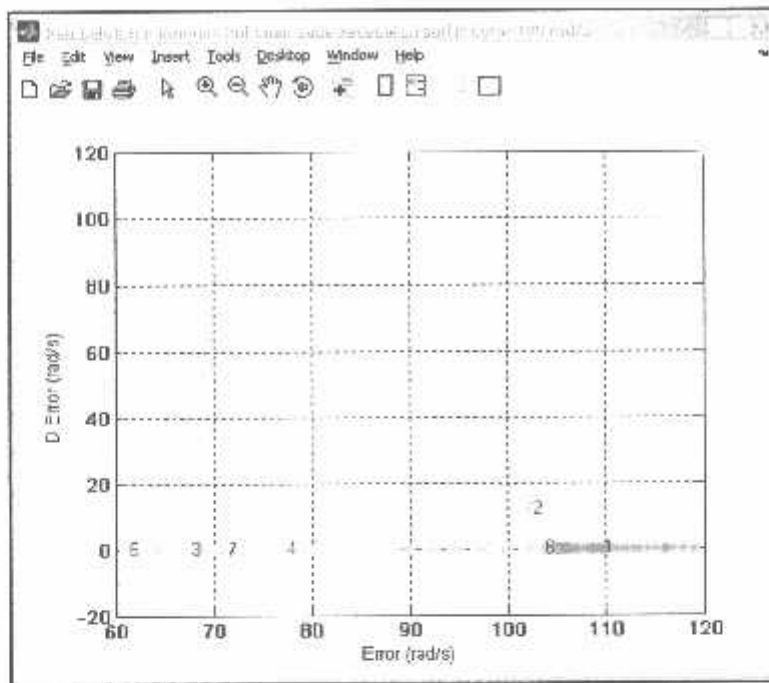
6 6 6 6 6 6 6 6 6

Columns 289 through 297

6 6 6 6 6 6 6 6 6

Columns 298 through 303

6 6 6 6 6 6



Gambar 3-3 : Grafik hasil pembelajaran jaringan kohonen pada kecepatan sudut putar 120 rad/s
Sumber : Simulasi

Keterangan Gambar :

-  = Menunjukkan kelas ke-1 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
-  = Menunjukkan kelas ke-2 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
-  = Menunjukkan kelas ke-3 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
-  = Menunjukkan kelas ke-4 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
-  = Menunjukkan kelas ke-5 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
-  = Menunjukkan kelas ke-6 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.
-  = Menunjukkan kelas ke-7 dari 7 buah kelas yang mewakili 7 *membership function* dari kontroler logika fuzzy.

3.4 PEMBENTUKAN FIS (*FUZZY INFERENCE SYSTEM*)

3.4.1 Pembentukan FIS ini meliputi :

1. Proses *fuzzyfikasi* {Pembentukan *membership error (E)* dan perubahan *error (dE)*},
2. Penentuan Aturan (*Rule Evaluation*).
3. Proses *Defuzzyfikasi*.

3.4.1.1 PROSES *FUZZYFIKASI*

Sinyal *error (E)* dan *perubahan error (dE)* yang didapatkan kemudian dilakukan proses *fuzzyfikasi* dengan cara memetakan sinyal *error (E)* dan *perubahan error (dE)* pada suatu fungsi keanggotaan (*membership function*).

Fungsi Keanggotaan (*membership function*) yang digunakan pada penelitian ini adalah bentuk segitiga (*triangular*).

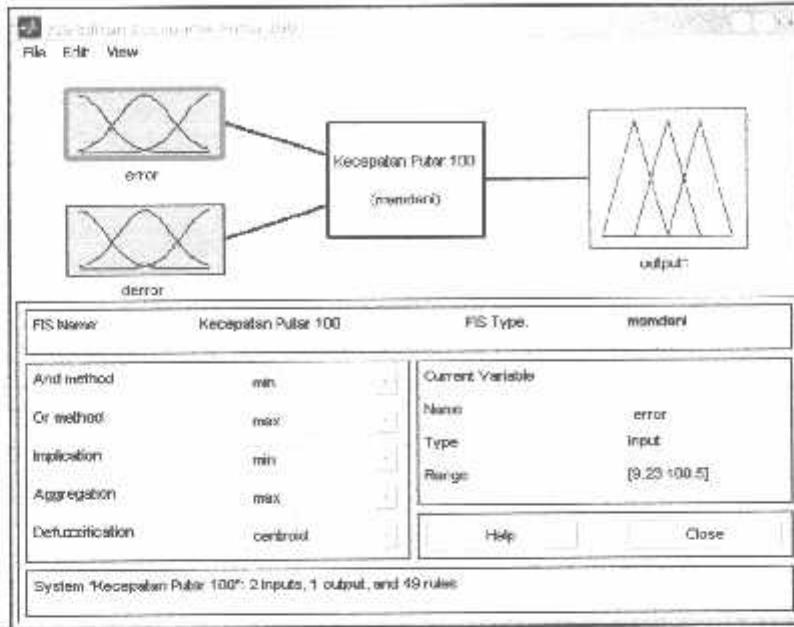
Hasil dari pembelajaran *Jaringan Kohonen* digunakan untuk menentukan parameter-parameter *fuzzy* yaitu titik tengah dari fungsi keanggotaan (*membership function*) baik itu *error (E)* maupun *perubahan error (dE)*.

Berdasarkan hasil pensamplingan data, pada kecepatan sudut putar 100 rad/s dengan torsi beban 0 Nm dan 50 Nm, sinyal *error (E)* dan *perubahan error (dE)* dipetakan dalam FIS Editor dengan memanfaatkan perangkat lunak *Matlab 7.0*.

Setelah dipetakan pada proses *fuzzyfikasi* diperoleh 7 fungsi keanggotaan (*membership function*) antara lain

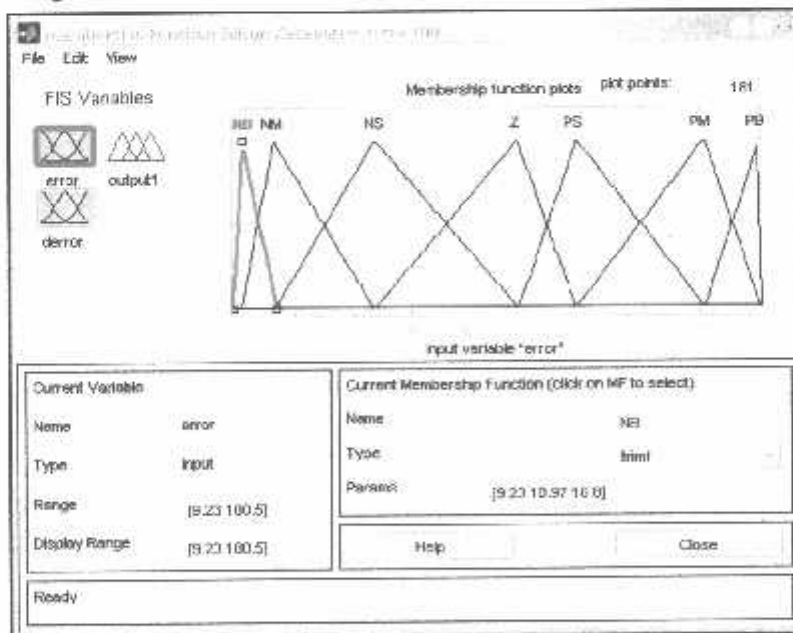
- | | |
|-------------------------------|--------------------------------|
| 1. <i>Negatif Big (NB)</i> | 2. <i>Negative medium (NM)</i> |
| 3. <i>Negative Small (NS)</i> | 4. <i>Zero (Z)</i> |
| 5. <i>Positif Small (PS)</i> | 6. <i>Positif Medium (PM)</i> |
| 7. <i>Positif Big (PB)</i> . | |

Untuk *FIS Editor fuzzy* pada kecepatan sudut putar 100 rad/s dapat dilihat dalam gambar 3-4 :

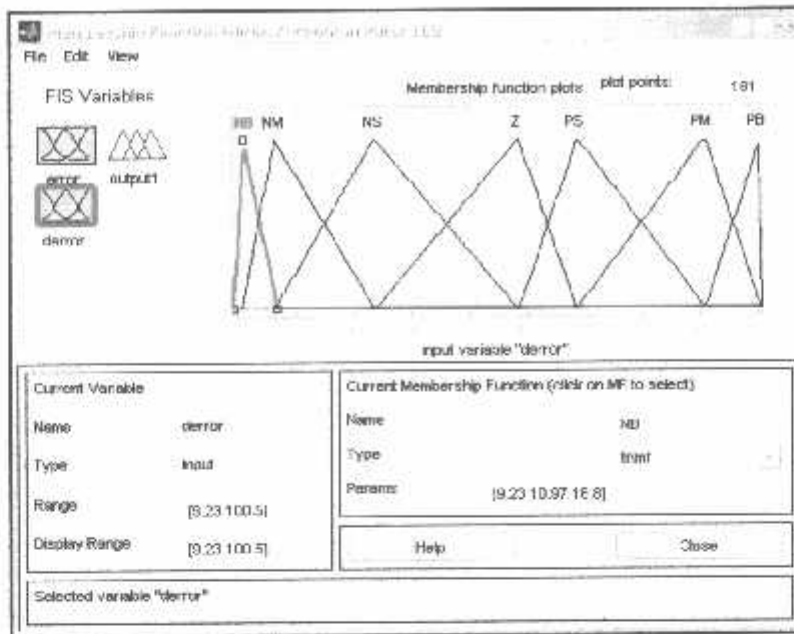


Gambar 3-4: FIS Editor pada kecepatan sudut putar 100 rad/s
Sumber : Perancangan

Sedangkan untuk Fungsi keanggotaan (*membership function*) *error* (*E*) dan *perubahan error* (*dE*) untuk kecepatan sudut putar 100 rad/s ditunjukkan pada gambar 3-5 dan gambar 3-6:

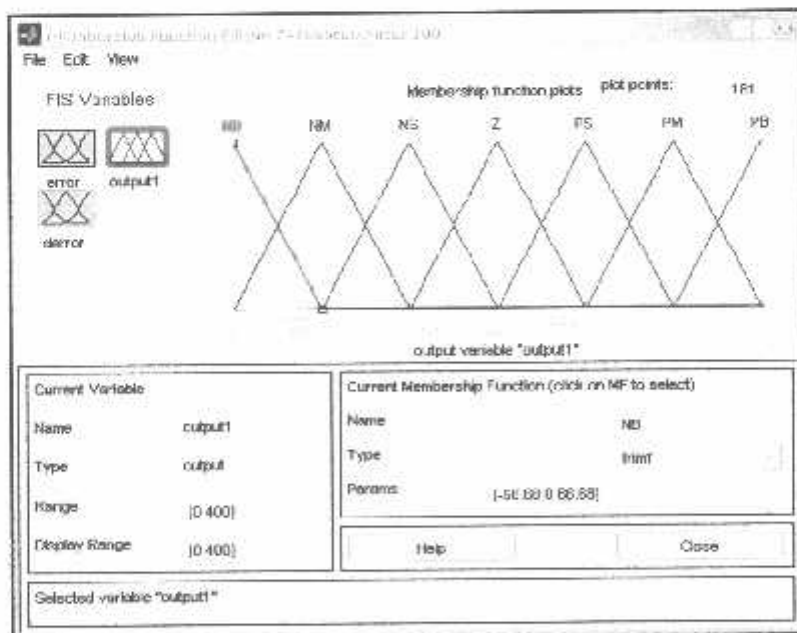


Gambar 3-5: *Membership Error* pada kecepatan sudut putar 100 rad/s
Sumber: Perancangan



Gambar 3-6: *Membership Delta Error* pada kecepatan sudut putar 100 rad/s
Sumber: Perancangan

Dan untuk *membership output fuzzy* pada kecepatan sudut putar 100 rad/s dapat dilihat dalam gambar 3-7 :



Gambar 3-7: *Membership Output* pada kecepatan sudur putar 100 rad/s
Sumber: Perancangan

Dari proses Fuzzyfikasi pada kecepatan sudut putar 100 rad/s didapatkan input nilai tengah fungsi keanggotaan (*membership function*) *error (E)* dan *perubahan error (dE)* sebagai berikut :

Tabel 3.5. Cuplikan input nilai tengah fungsi keanggotaan pada kecepatan sudut putar 100 rad/s

No	MEMBERSHIP FUZZY	Nilai Tengah
1	NB (<i>Negative Big</i>)	10.97
2	NM (<i>Negative Medium</i>)	16.8
3	NS (<i>Negative Small</i>)	34.05
4	Z (<i>Zero</i>)	58.7
5	PS (<i>Positive Small</i>)	68.7
6	PM (<i>Posiyive Medium</i>)	90.7
7	PB (<i>Positive Big</i>)	100.4

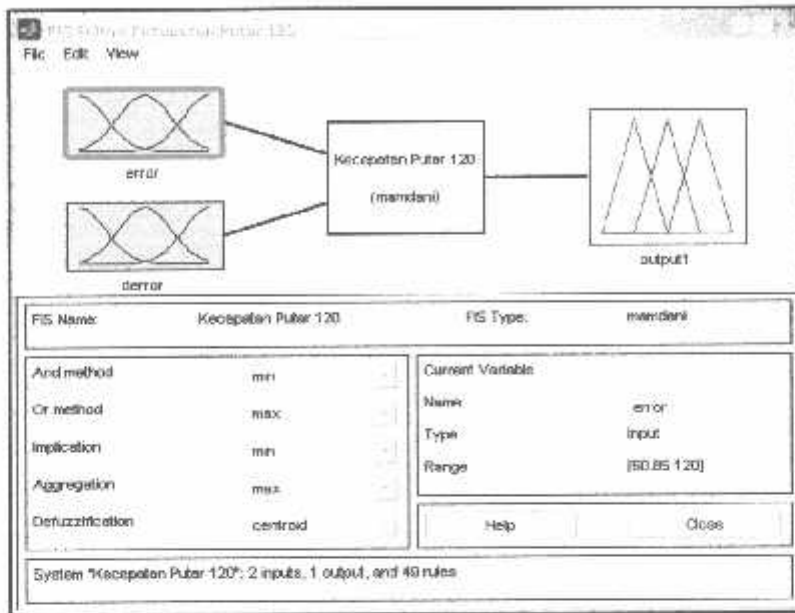
Sumber: Perancangan

Dan berdasarkan hasil pensamplingan data, pada kecepatan sudut putar 120 rad/s dengan torsi beban 0 Nm dan 50 Nm, sinyal *error (E)* dan *perubahan error (dE)* dipetakan dalam FIS Editor dengan memanfaatkan perangkat lunak *Matlab 7.0*.

Setelah dipetakan pada proses fuzzyfikasi diperoleh 7 fungsi keanggotaan (*membership function*) antara lain

1. *Negatif Big (NB)*
2. *Negative medium (NM)*
3. *Negative Small (NS)*
4. *Zero (Z),*
5. *Positif Small (PS)*
6. *Positif Medium (PM)*
7. *Positif Big (PB)*

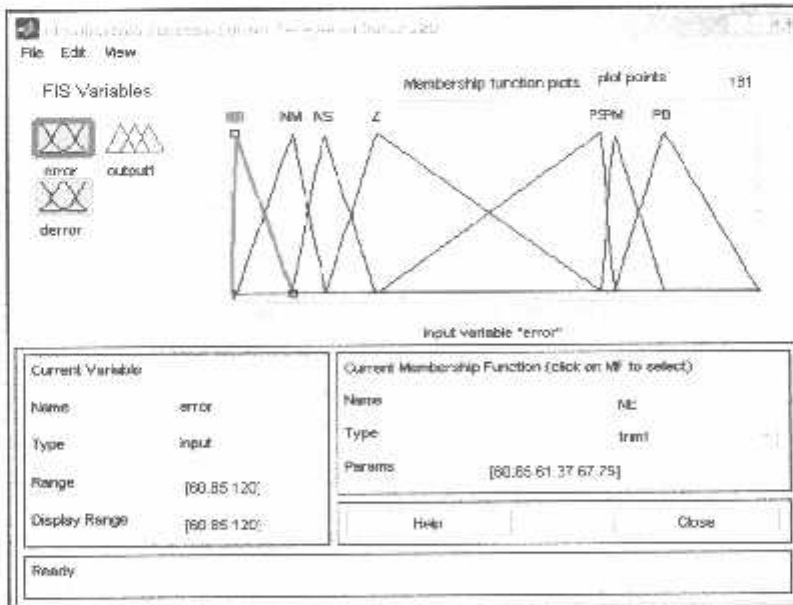
Untuk *FIS Editor fuzzy* pada kecepatan sudut putar 120 rad/s dapat dilihat dalam gambar 3-8 :



Gambar 3-8: FIS Editor pada kecepatan sudut putar 120 rad/s

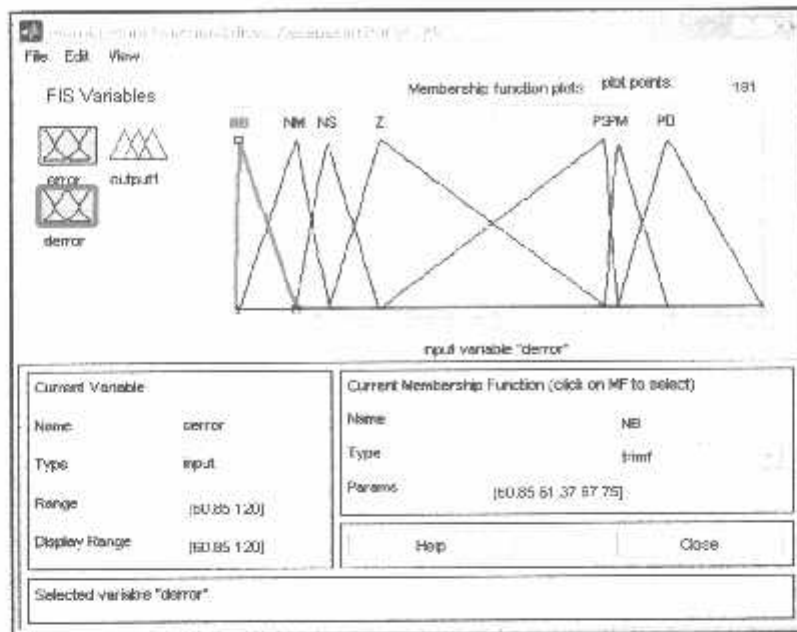
Sumber : Perancangan

Sedangkan untuk Fungsi keanggotaan (*membership function*) *error* (*E*) dan perubahan *error* (*dE*) untuk kecepatan sudut putar 120 rad/s ditunjukkan pada gambar 3-9 dan gambar 3-10:



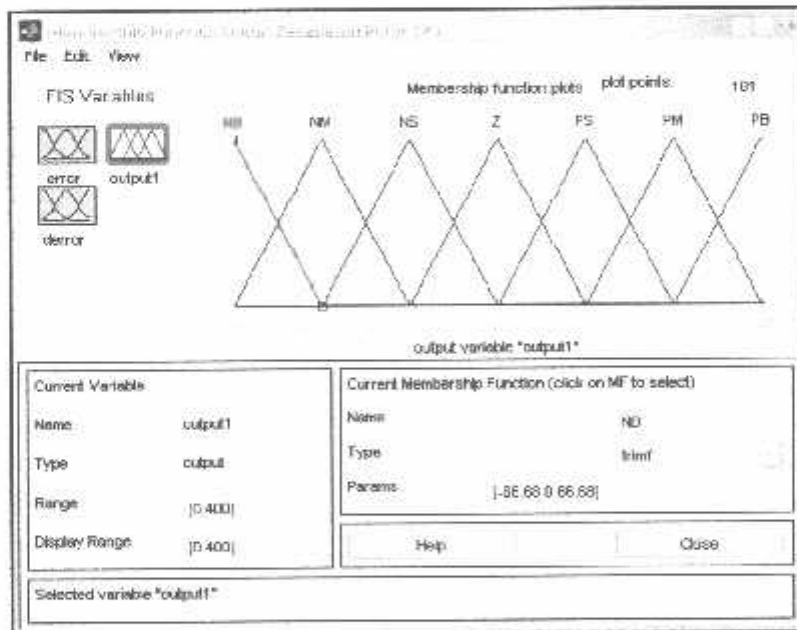
Gambar 3-9: Membership Error pada kecepatan sudut putar 120 rad/s

Sumber: Perancangan



Gambar 3-10: *Membership Delta Error* pada kecepatan sudut putar 120 rad/s
Sumber: Perancangan

Dan untuk *membership output fuzzy* pada kecepatan sudut putar 120 rad/s dapat dilihat dalam gambar 3-11 :



Gambar 3-8: *Membership Output* pada kecepatan sudur putar 120 rad/s
Sumber: Perancangan

Dari proses *Fuzzyifikasi* pada kecepatan sudut putar 120 rad/s didapatkan input nilai tengah fungsi keanggotaan (*membership function*) *error (E)* dan *perubahan error (dE)* sebagai berikut :

Tabel 3.6. Cuplikan input nilai tengah fungsi keanggotaan pada kecepatan sudut putar 120 rad/s

No	MEMBERSHIP FUZZY	Nilai Tengah
1	NB (<i>Negative Big</i>)	61.37
2	NM (<i>Negative Medium</i>)	67.75
3	NS (<i>Negative Small</i>)	71.4
4	Z (<i>Zero</i>)	77.2
5	PS (<i>Positive Small</i>)	102.4
6	PM (<i>Posiyive Medium</i>)	103.7
7	PB (<i>Positive Big</i>)	109.4

Sumber: Perancangan

3.4.1.2 PENENTUAN ATURAN (*RULE EVALUATION*)

Penentuan komposisi aturan merupakan tahap selanjutnya setelah penentuan input nilai tengah fungsi keanggotaan (*membership function*) *error (E)* dan *perubahan error (dE)* serta label himpunan *fuzzy* input dan output.

Penyusunan basis aturan menghasilkan tabel pandang (*look up table*) yang disebut dengan tabel Mac Vicar Whelan dengan kaidah atur 7 variabel *error (E)* dan 7 variabel perubahan error (*dE*) sehingga dihasilkan 49 basis aturan.

Dimana semua keanggotaan dari sinyal *error* (E) bertemu dengan semua keanggotaan sinyal *perubahan error* (dE) dan begitu juga sebaliknya, semua keanggotaan *perubahan error* (dE) bertemu dengan semua keanggotaan sinyal *error* (E).

Secara tabel dapat dituliskan :

Tabel 3.7. Aturan *fuzzy* berdasarkan *Mac Vicar Whelan*

E							
NB	NB	NB	NB	NB	NB	NB	PS
NM	NB	NM	NM	NM	NS	Z	PM
NS	NB	NM	NM	Z	PM	PM	PM
Z	NM	NM	NM	Z	PM	PM	PM
PS	NM	NM	NS	PS	PM	PM	PB
PM	NM	Z	PS	PM	PM	PB	PB
PB	NS	PB	PB	PB	PB	PB	PB

Sumber: Perancangan

Tabel Aturan dengan kaidah atur 7 variabel *error* (E) dan 7 variabel perubahan error (dE) dan menghasilkan 49 basis aturan diatas yang berdasarkan aturan *Mac Vicar Whelan* dapat dijabarkan dengan pernyataan linguistik sebagai berikut :

- R_1 : Jika E adalah NB dan dE adalah NB maka U adalah NB
- R_2 : Jika E adalah NB dan dE adalah NM maka U adalah NB
- R_3 : Jika E adalah NB dan dE adalah NS maka U adalah NB
- R_4 : Jika E adalah NB dan dE adalah Z maka U adalah NB
- R_5 : Jika E adalah NB dan dE adalah PS maka U adalah NB
- R_6 : Jika E adalah NB dan dE adalah PM maka U adalah NB
- R_7 : Jika E adalah NB dan dE adalah PB maka U adalah Z.

- R₈ : Jika E adalah NM dan dE adalah NB maka U adalah NB
- R₉ : Jika E adalah NM dan dE adalah NM maka U adalah NM
- R₁₀ : Jika E adalah NM dan dE adalah NS maka U adalah NM
- R₁₁ : Jika E adalah NM dan dE adalah Z maka U adalah NM
- R₁₂ : Jika E adalah NM dan dE adalah PS maka U adalah NS
- R₁₃ : Jika E adalah NM dan dE adalah PM maka U adalah Z
- R₁₄ : Jika E adalah NM dan dE adalah PB maka U adalah PS
- R₁₅ : Jika E adalah NS dan dE adalah NB maka U adalah NB
- R₁₆ : Jika E adalah NS dan dE adalah NB maka U adalah NB
- R₁₇ : Jika E adalah NS dan dE adalah NM maka U adalah NM
- R₁₈ : Jika E adalah NM dan dE adalah Z maka U adalah NM
- R₁₉ : Jika E adalah Z dan dE adalah PS maka U adalah Z
- R₂₀ : Jika E adalah Z dan dE adalah PM maka U adalah PM
- R₂₁ : Jika E adalah Z dan dE adalah PB maka U adalah PM
- R₂₂ : Jika E adalah Z dan dE adalah NB maka U adalah NB
- R₂₃ : Jika E adalah Z dan dE adalah NM maka U adalah NM
- R₂₄ : Jika E adalah Z dan dE adalah NK maka U adalah NS
- R₂₅ : Jika E adalah Z dan dE adalah SN maka U adalah Z
- R₂₆ : Jika E adalah SN dan dE adalah PK maka U adalah NS
- R₂₇ : Jika E adalah Z dan dE adalah PS maka U adalah PM
- R₂₈ : Jika E adalah Z dan dE adalah PB maka U adalah PB
- R₂₉ : Jika E adalah PS dan dE adalah NB maka U adalah NM
- R₃₀ : Jika E adalah PS dan dE adalah NS maka U adalah NS
- R₃₁ : Jika E adalah PS dan dE adalah NK maka U adalah Z
- R₃₂ : Jika E adalah PS dan dE adalah SN maka U adalah PS
- R₃₃ : Jika E adalah PS dan dE adalah PK maka U adalah PM
- R₃₄ : Jika E adalah PS dan dE adalah PS maka U adalah PB
- R₃₅ : Jika E adalah PS dan dE adalah PB maka U adalah PB
-

- R₃₆ : Jika E adalah PS dan dE adalah NB maka U adalah NK
 R₃₇ : Jika E adalah PS dan dE adalah NS maka U adalah SN
 R₃₈ : Jika E adalah PS dan dE adalah NK maka U adalah PK
 R₃₉ : Jika E adalah PS dan dE adalah SN maka U adalah PS
 R₄₀ : Jika E adalah PS dan dE adalah PK maka U adalah PB
 R₄₁ : Jika E adalah PS dan dE adalah PS maka U adalah PB
 R₄₂ : Jika E adalah PS dan dE adalah PB maka U adalah PB
 R₄₃ : Jika E adalah PB dan dE adalah NB maka U adalah SN
 R₄₄ : Jika E adalah PB dan dE adalah NS maka U adalah PK
 R₄₅ : Jika E adalah PB dan dE adalah NK maka U adalah PS
 R₄₆ : Jika E adalah PB dan dE adalah SN maka U adalah PB
 R₄₇ : Jika E adalah PB dan dE adalah PK maka U adalah PB
 R₄₈ : Jika E adalah PB dan dE adalah PS maka U adalah PB
 R₄₉ : Jika E adalah PB dan dE adalah PB maka U adalah PB

3.4.1.3 DEFUZZYFIKASI

Defuzzifikasi merupakan proses pemetaan himpunan *fuzzy* ke himpunan *crisp*. Proses ini merupakan kebalikan dari proses fuzzifikasi berfungsi untuk mengubah keluaran linguistik ke dalam keluaran *crisp*.

Proses ini berfungsi untuk menentukan suatu nilai *crisp* output. Prosesnya adalah sebagai berikut: suatu nilai *fuzzy* output yang berasal dari *rule evaluation* diambil kemudian dimasukkan ke dalam suatu *membership function* output.

Metode yang digunakan pada skripsi ini adalah dengan menggunakan metode *centroid*.

BAB IV

SIMULASI DAN ANALISA HASIL

Simulasi pengontrolan Motor Induksi 3 Fasa dengan *fuzzy* pada jaringan syaraf tiruan kohonen ini, menggunakan software Matlab 7.0 dengan memanfaatkan fasilitas *simulink matlab environment*.

Simulasi pengendalian motor induksi 3 fasa ini meliputi beberapa pengujian yang bertujuan untuk mengetahui performansi dari kontroler yang dirancang. Pengujian yang dilakukan adalah sebagai berikut :

4.1 Simulasi Dengan Menggunakan Fuzzy berdasarkan pembelajaran Jaringan

Kohonen :

1. Pengujian pada kecepatan 100 rad/s dengan torsi beban 0 Nm
2. Pengujian pada kecepatan 120 rad/s dengan torsi beban 0 Nm
3. Pengujian pada kecepatan 100 rad/s dengan torsi beban 50 Nm
4. Pengujian pada kecepatan 120 rad/s dengan torsi beban 50 Nm

4.2 Simulasi Dengan Menggunakan PID

1. Pengujian pada kecepatan 100 rad/s dengan torsi beban 0 Nm
 2. Pengujian pada kecepatan 120 rad/s dengan torsi beban 0 Nm
 3. Pengujian pada kecepatan 100 rad/s dengan torsi beban 50 Nm
 4. Pengujian pada kecepatan 120 rad/s dengan torsi beban 50 Nm
-

Parameter motor Induksi 3 fasa :

Tipe rotor = squirrel cage

Daya Motor (P_n) = $50 \cdot 746$ VA

V_{rms} = 460 volt

Frekuensi = 60 Hz

R rotor = 0.0087 ohm

Induktansi rotor = $0.8 \cdot 10^{-3}$ H

R rotor = 0.2298 ohm

Induktansi rotor = $0.8 \cdot 10^{-3}$ H

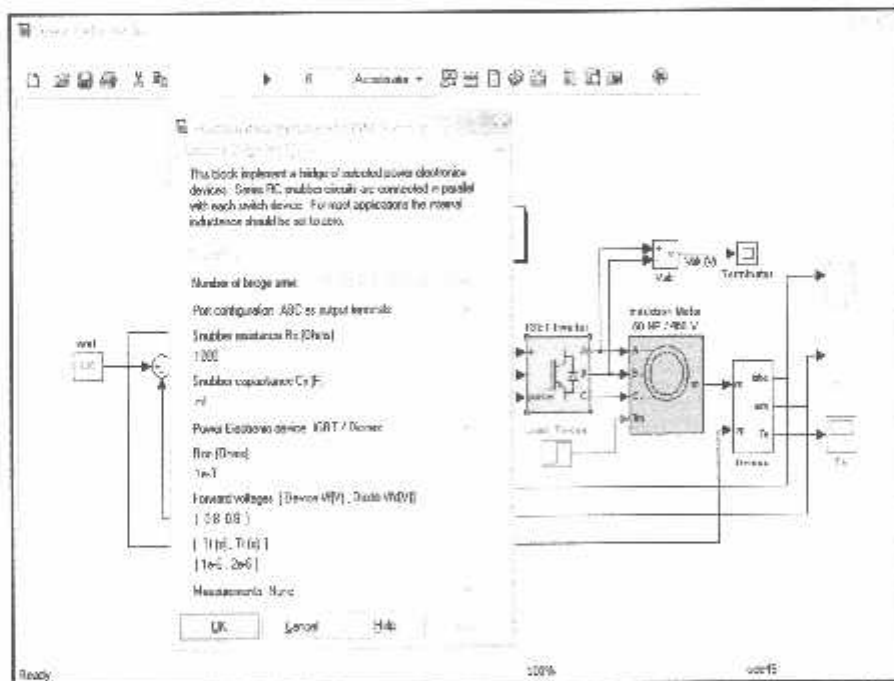
Induktansi bersama = $34.7 \cdot 10^{-3}$

Inersia = $1.662 \text{ kg} \cdot \text{m}^2$

Faktor gesekan = 0.1 Nms

Banyaknya kutub = 2 pasang

Begitu pula untuk parameter dari IGBT inverter 3 fasa yang diperoleh berdasarkan *default* program *Matlab 7.0*. Parameter-parameter tersebut dapat dilihat sebagai berikut



Gambar 4-3: Teraan Parameter IGBT Inverter 3 fasa untuk *fuzzy*
Sumber : Simulasi Matlab 7.0

Parameter IGBT Inverter 3 fasa :

Jumlah jembatan = 3 buah

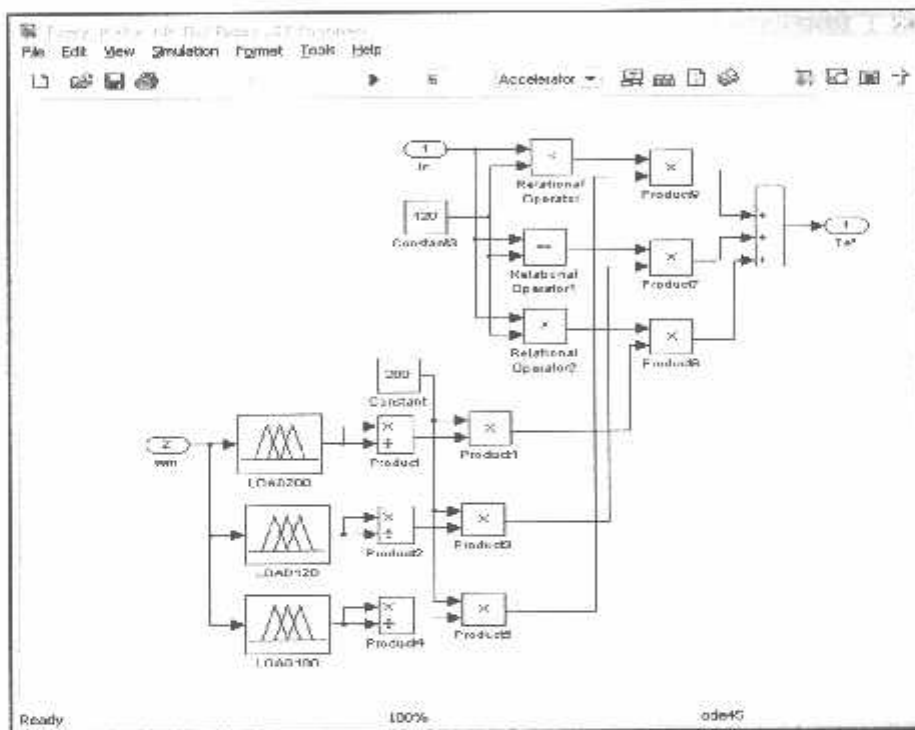
Resistansi snubber = 1000 ohms

Kapasitansi snubber = inf

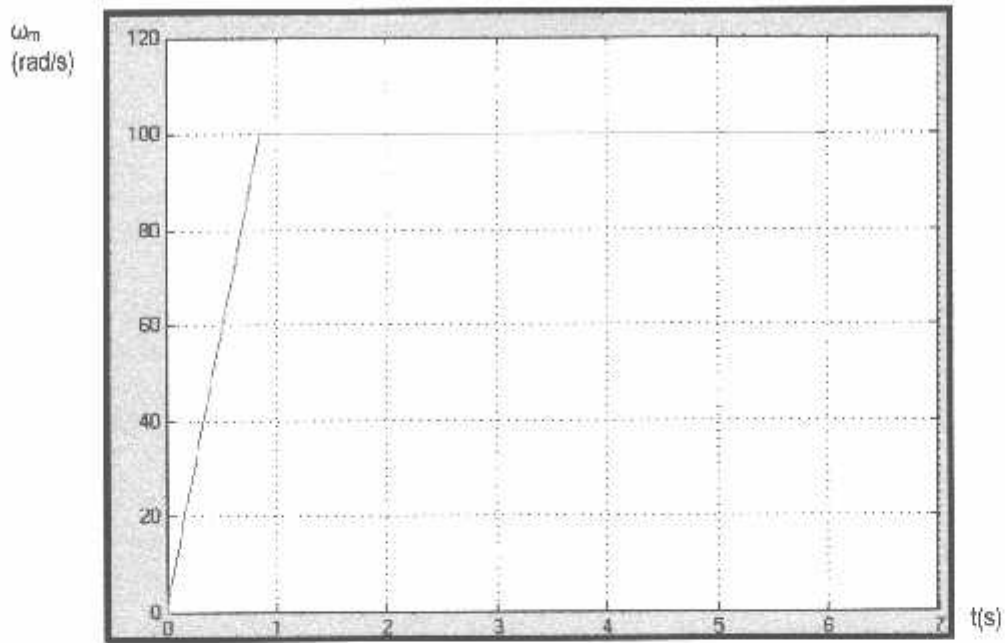
Nama device = IGBT / Dioda

$R_{on} = 1e-3$ ohm

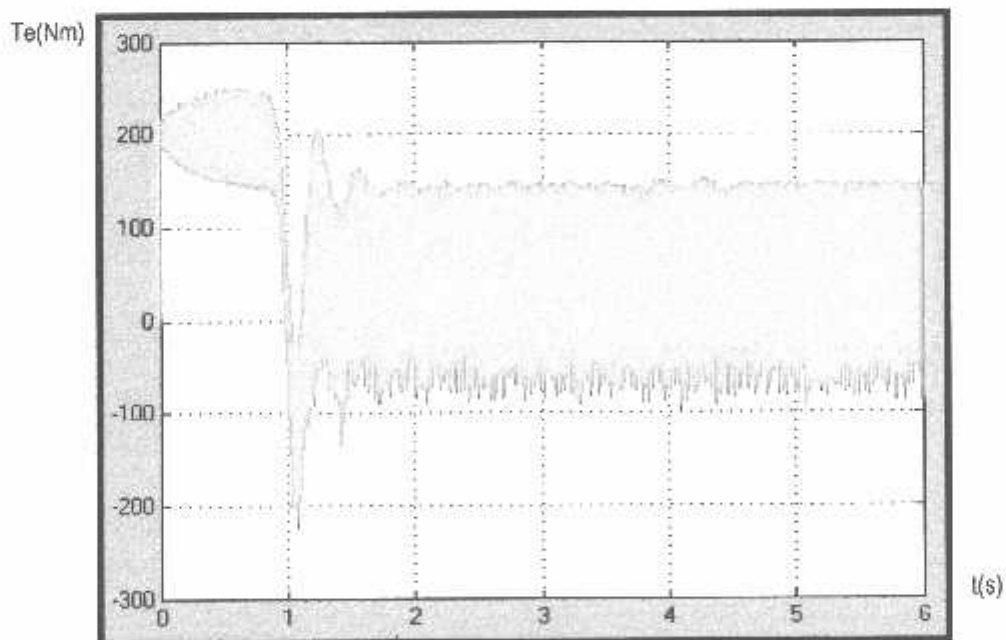
Dan dibawah ini adalah gambar diagram vektor kontrol *fuzzy* pada jaringan syaraf tiruan kohonen :



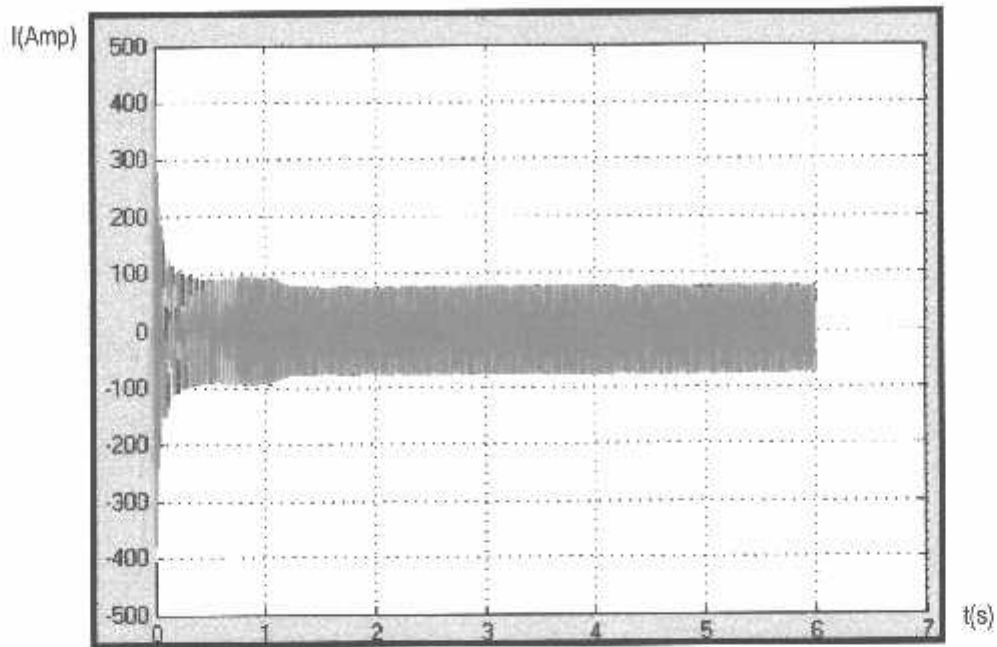
Gambar 4-4: Blok Diagram Vektor Control Fuzzy
Sumber : Simulasi Matlab 7.0

PENGUJIAN :**4.1.1 Pengujian pada kecepatan sudut putar 100 rad/s dengan torsi beban 0 Nm**

Gambar 4-1 : Grafik kecepatan MI dengan setting point 100 rad/s, beban nol
Sumber : Simulasi Matlab 7.0

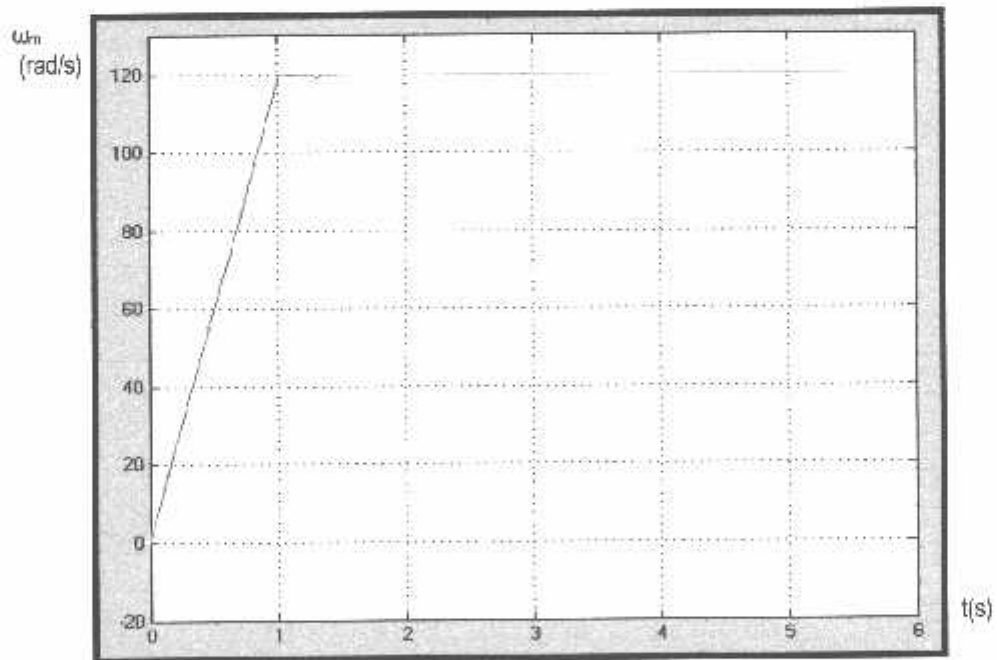


Gambar 4-2: Grafik torsi MI dengan setting point 100 rad/s, beban nol
Sumber : Simulasi Matlab 7.0

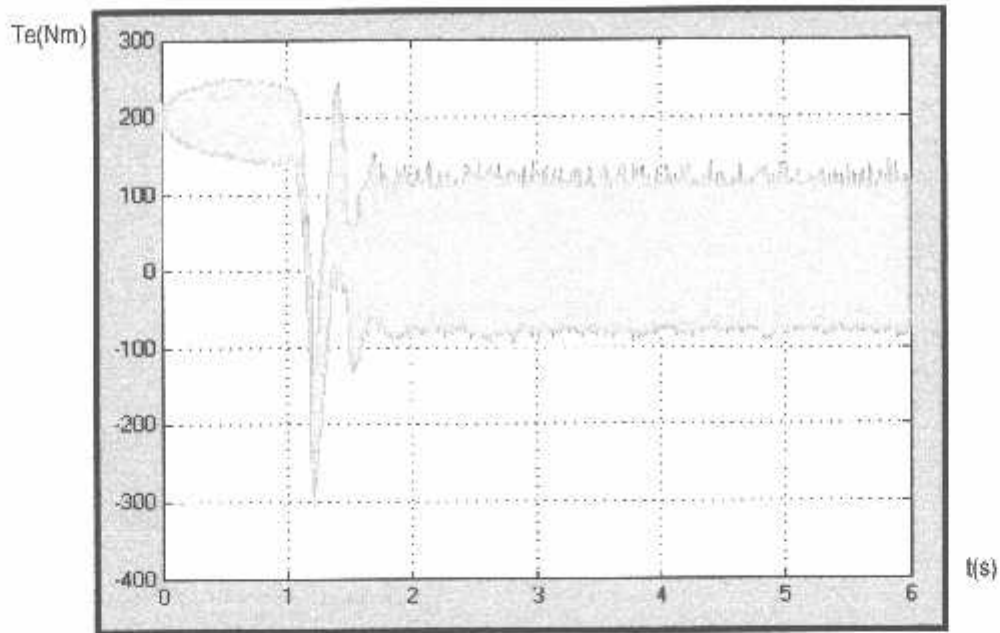


Gambar 4-3: Grafik Arus MI dengan setting point 100 rad/s, beban nol
Sumber : Simulasi Matlab 7.0

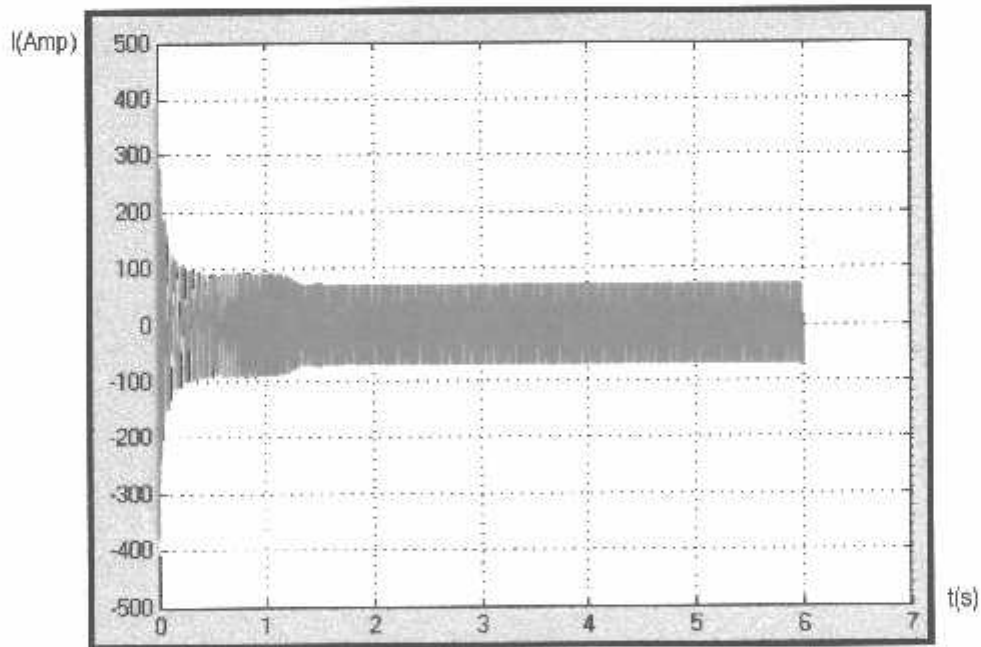
4.1.2 Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 0 Nm



Gambar 4-4: Grafik kecepatan MI dengan setting point 120 rad/s, beban nol
Sumber : Simulasi Matlab 7.0

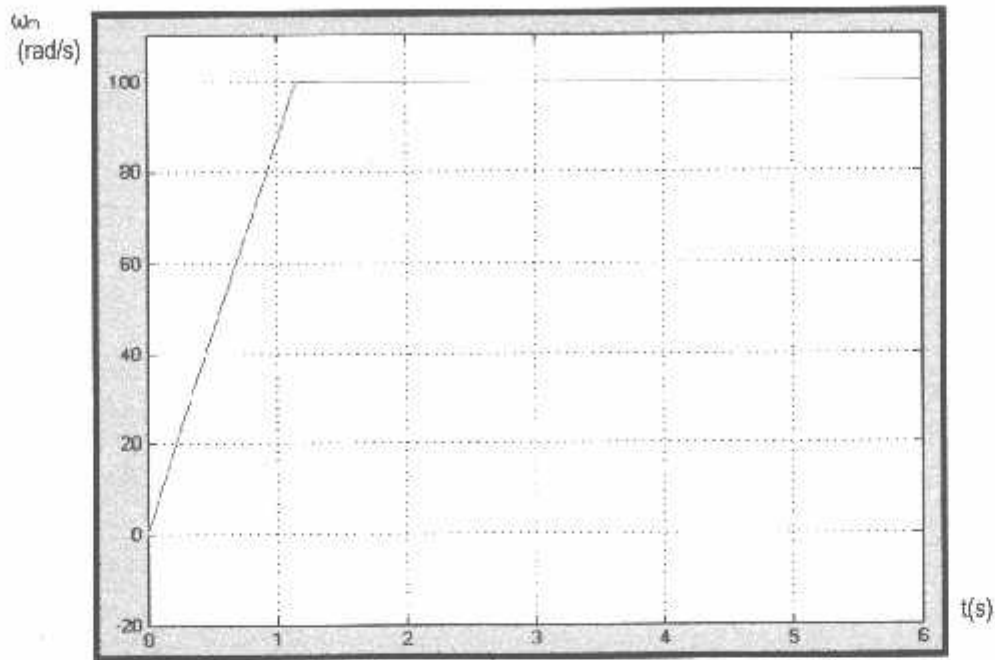


Gambar 4-5: Grafik torsi MI dengan setting point 120 rad/s, beban nol
Sumber : Simulasi Matlab 7.0

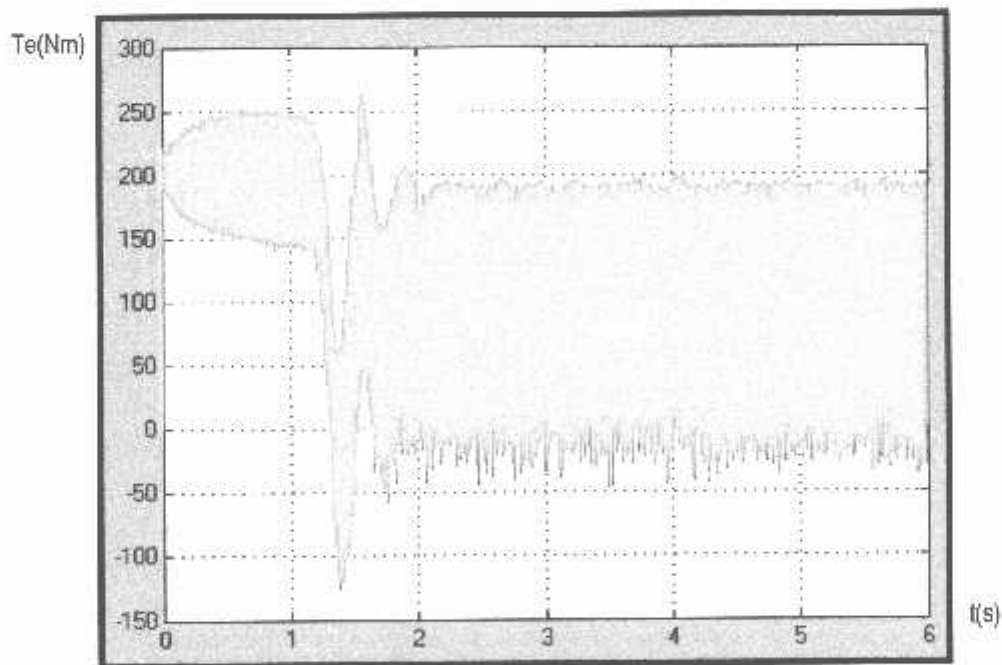


Gambar 4-6: Grafik Arus MI dengan setting point 120 rad/s, beban nol
Sumber : Simulasi Matlab 7.0

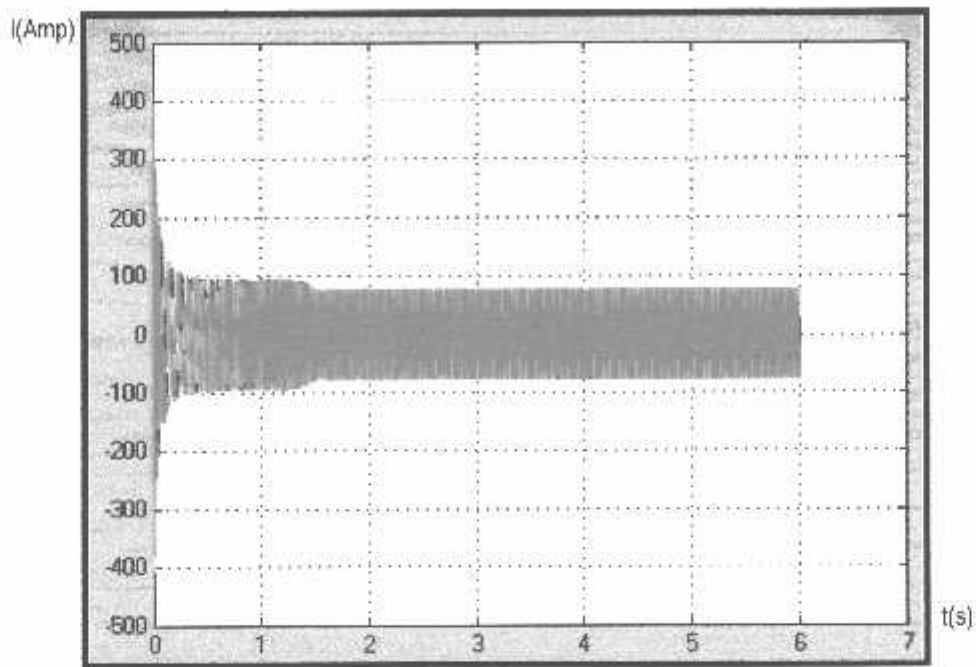
4.1.3 Pengujian pada kecepatan sudut putar 100 rad/s dengan torsi beban 50 Nm



Gambar 4-7: Grafik kecepatan MI dengan setting point 100 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0

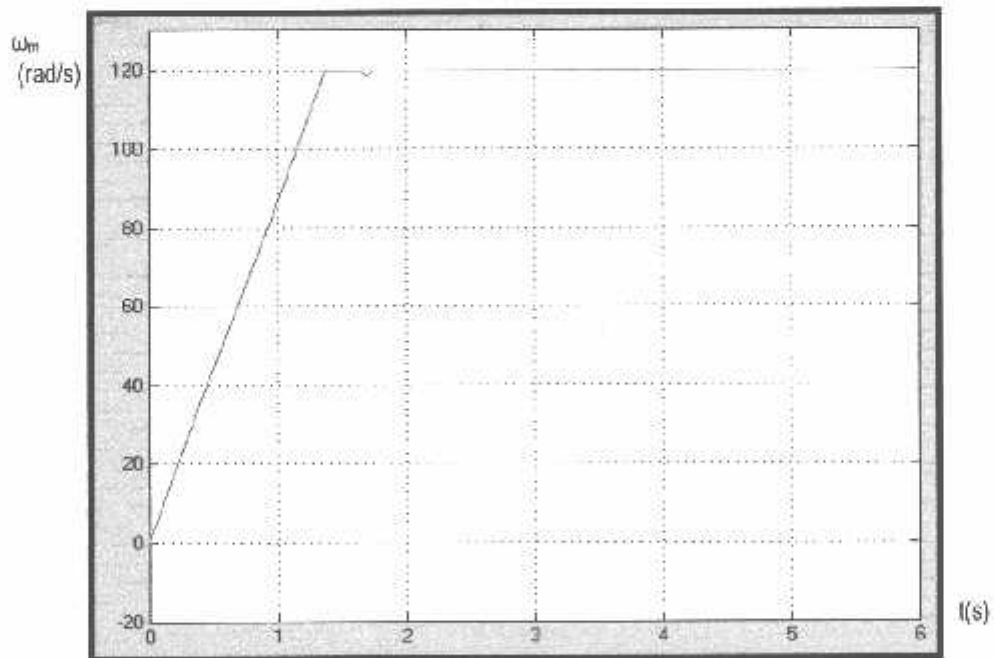


Gambar 4-8: Grafik torsi MI dengan setting point 100 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0

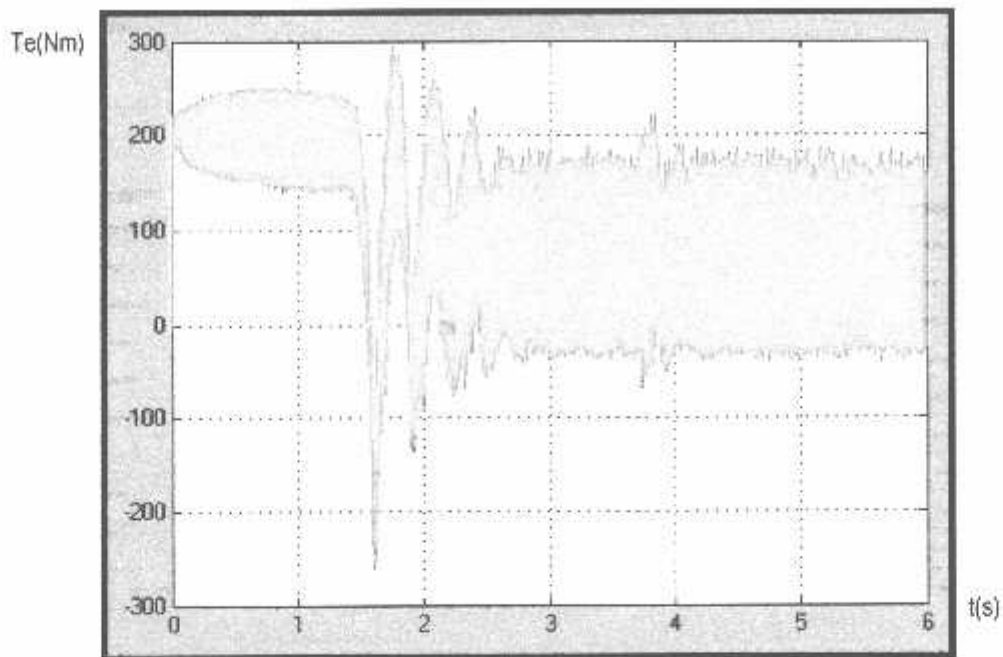


Gambar 4-9 : Grafik Arus MI dengan setting point 100 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0

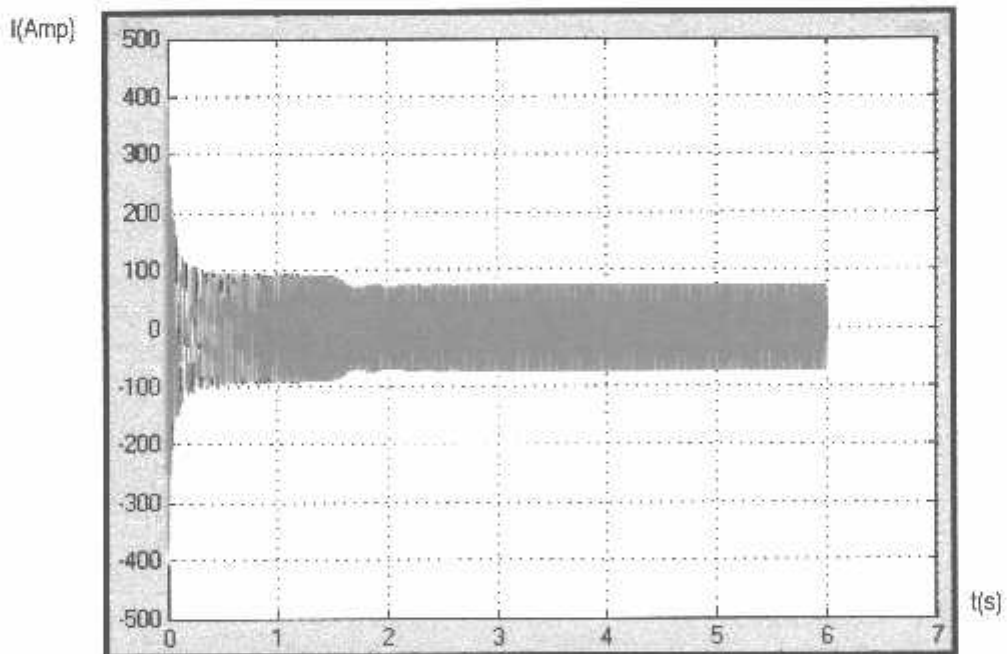
4.1.4 Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 50 Nm



Gambar 4-10: Grafik kecepatan MI dengan setting point 120 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0



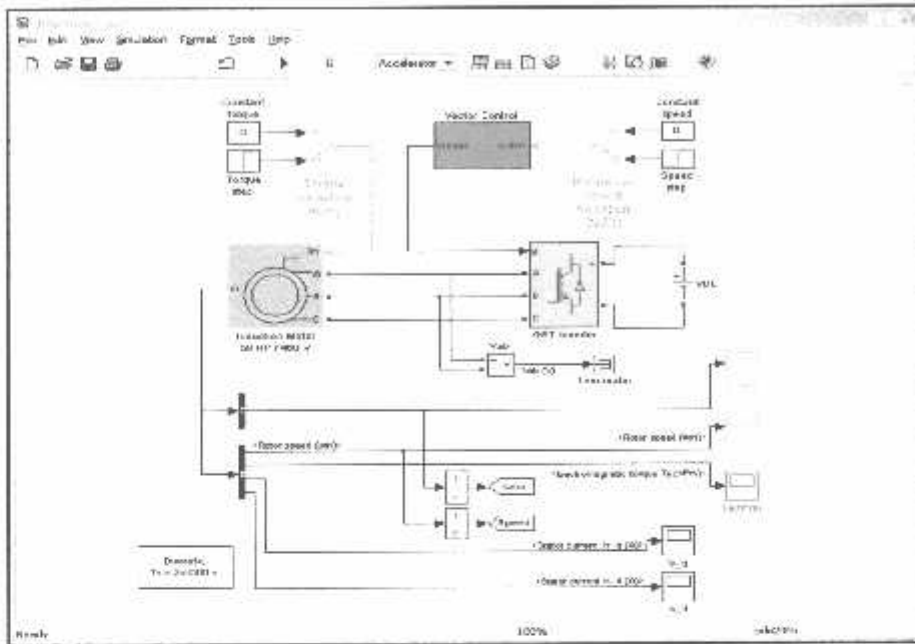
Gambar 4-11: Grafik torsi MI dengan setting point 120 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0



Gambar 4-12: Grafik Arus MI dengan setting point 120 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0

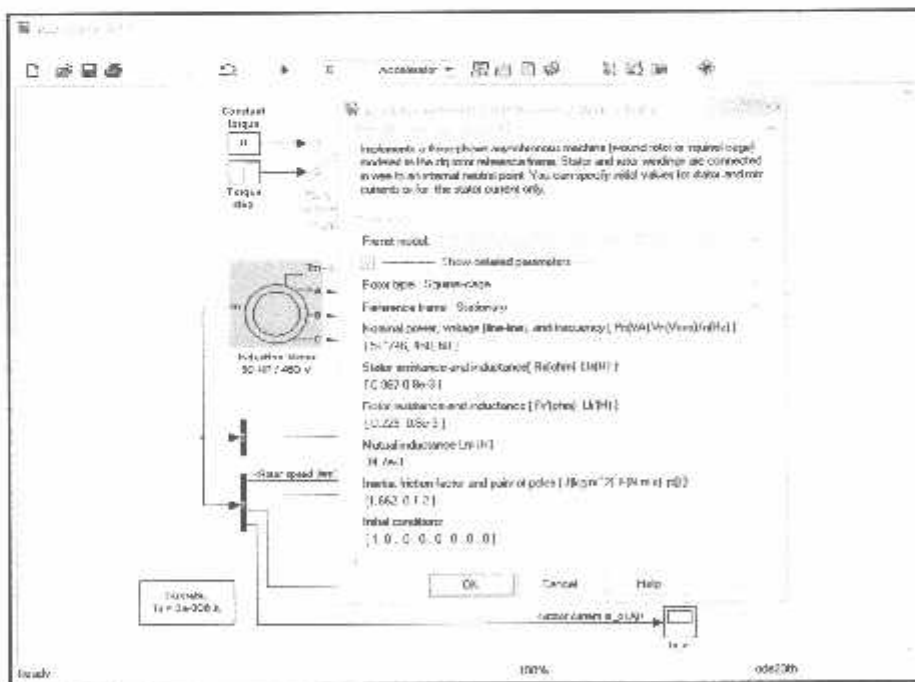
4.2 Simulasi Dengan Menggunakan PID

Pemodelan Sistem



Gambar 4-5 : Pemodelan Sistem Dengan Menggunakan PID
Sumber : Simulasi Matlab 7.0

Parameter untuk motor induksi 3 fasa diperoleh berdasarkan *default* dari program *Matlab 7.0*.. Parameter-parameter tersebut dapat dilihat sebagai berikut :



Gambar 4-6 : Teraan Parameter Motor Induksi 3 Fasa untuk PID
Sumber : Simulasi Matlab 7.0

Parameter motor Induksi 3 fasa :

Tipe rotor = squirrel cage

Daya Motor (P_n) = 50×746 VA

V_{rms} = 460 volt

Frekuensi = 60 Hz

R rotor = 0.0087 ohm

Induktansi rotor = 0.8×10^{-3} H

R rotor = 0.2298 ohm

Induktansi rotor = 0.8×10^{-3} H

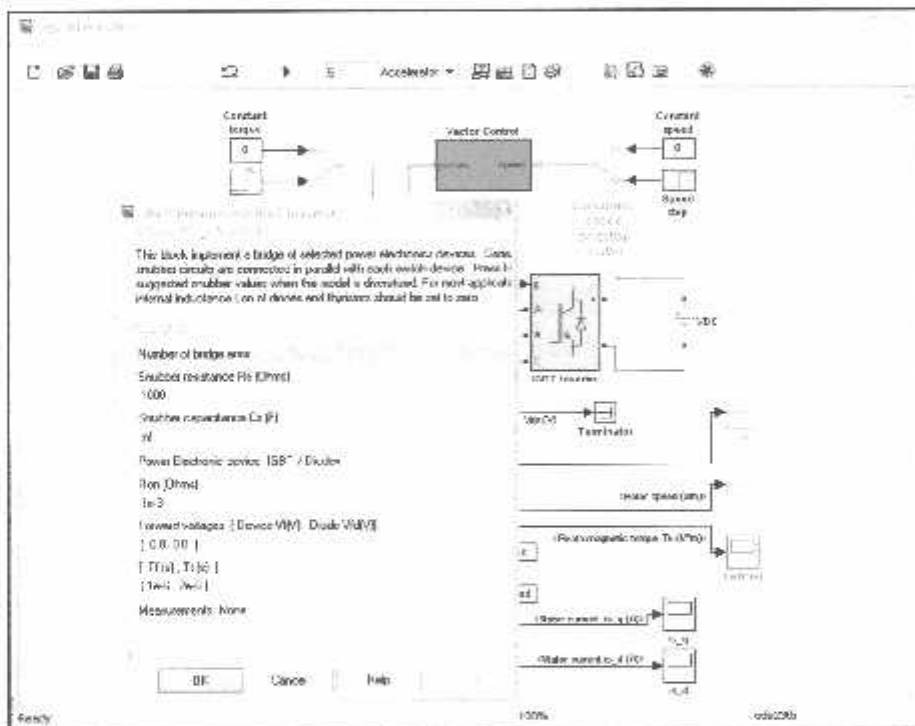
Induktansi bersama = 34.7×10^{-3}

Inersia = 1.662 kg.m^2

Faktor gesekan = 0.1 Nms

Banyaknya kutub = 2 pasang

Begitu pula untuk parameter dari IGBT inverter 3 fasa yang diperoleh berdasarkan *default* program *Matlab 7.0*. Parameter-parameter tersebut dapat dilihat sebagai berikut



Gambar 4-7: Teraan Parameter IGBT Inverter 3 fasa untuk PID
Sumber : Simulasi Matlab 7.0

Parameter IGBT Inverter 3 fasa :

Jumlah jembatan = 3 buah

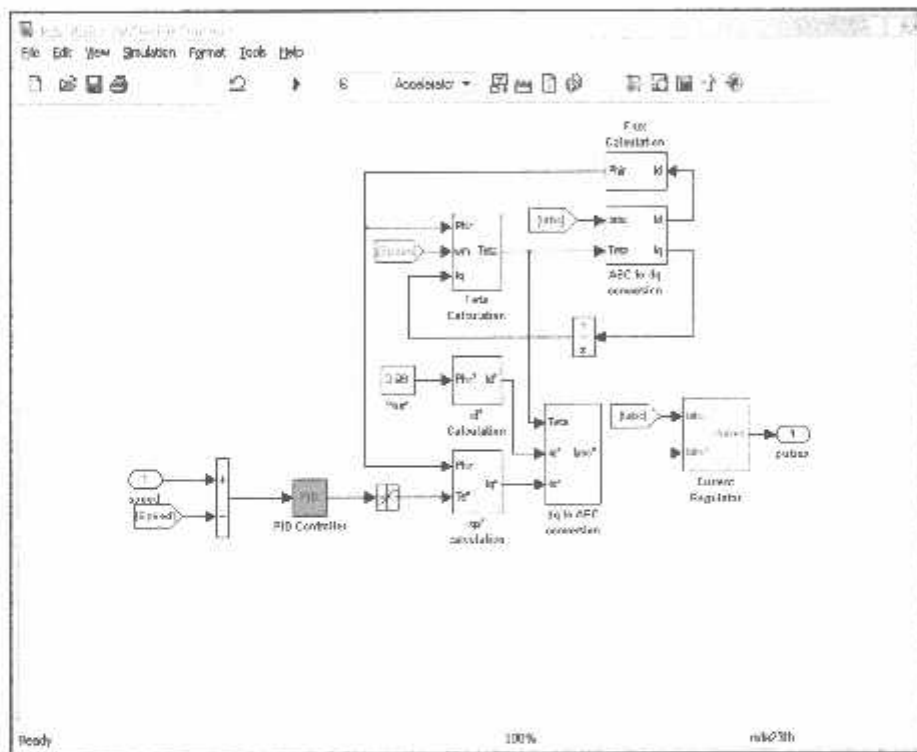
Resistansi snubber = 1000 ohms

Kapasitansi snubber = inf

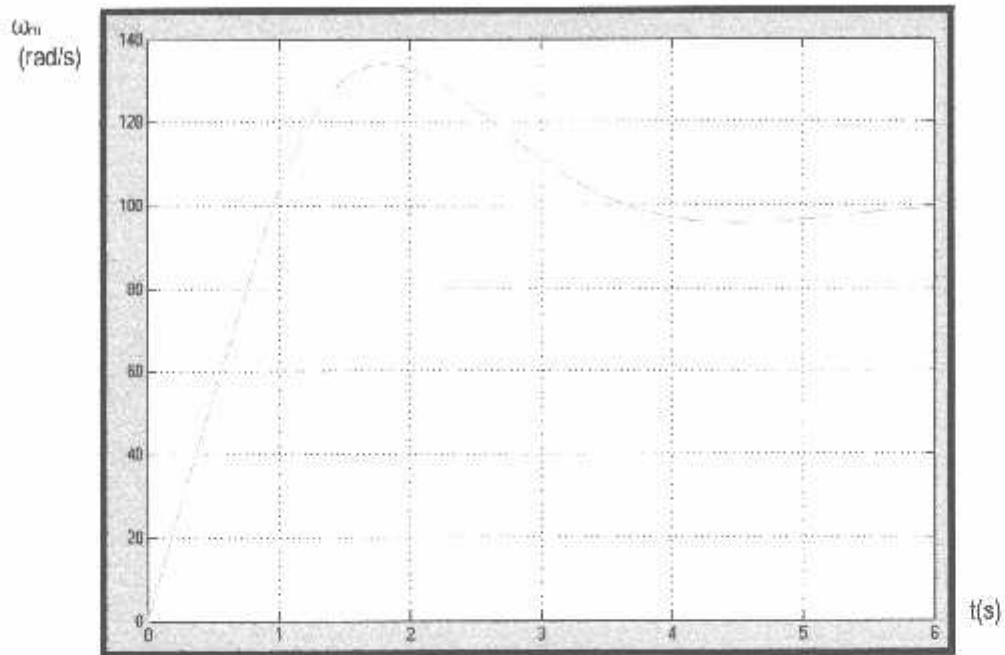
Nama device = IGBT / Dioda

R on = 1e-3 ohm

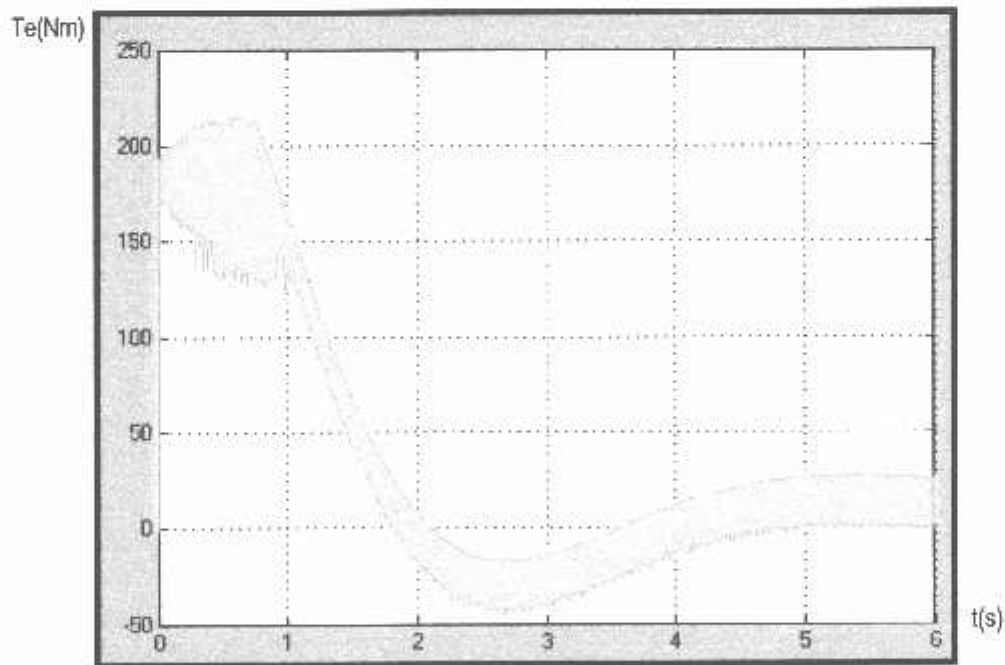
Dalam blok vector control pada model sistem pengaturan diatas, untuk PID digunakan nilai *default* dari program *Matlab 7.0* dengan **Kp** =10, **Ki** =13 dan **Kd** = 5. Adapun blok vector kontrolnya sebagai berikut:



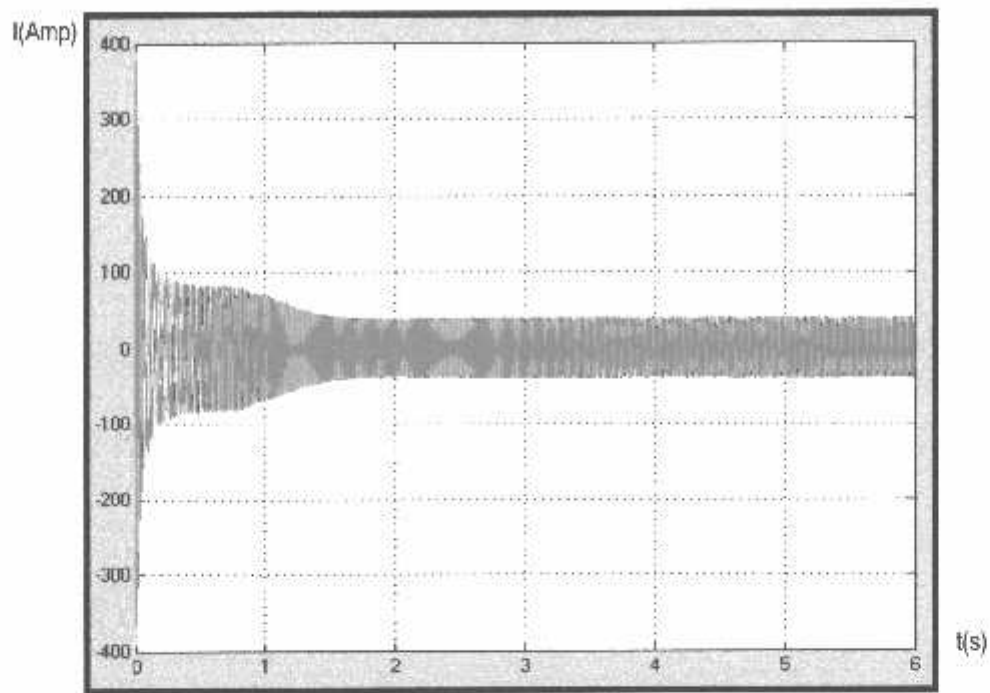
Gambar 4-8: Diagram Blok Vektor Control PID
Sumber : Simulasi Matlab 7.0

Pengujian :**4.2.1 Pengujian pada kecepatan sudut putar 100 rad/s dengan torsi beban 0 Nm**

Gambar 4-13: Grafik kecepatan PID dengan setting point 100 rad/s, beban 0 Nm
Sumber : Simulasi Matlab 7.0

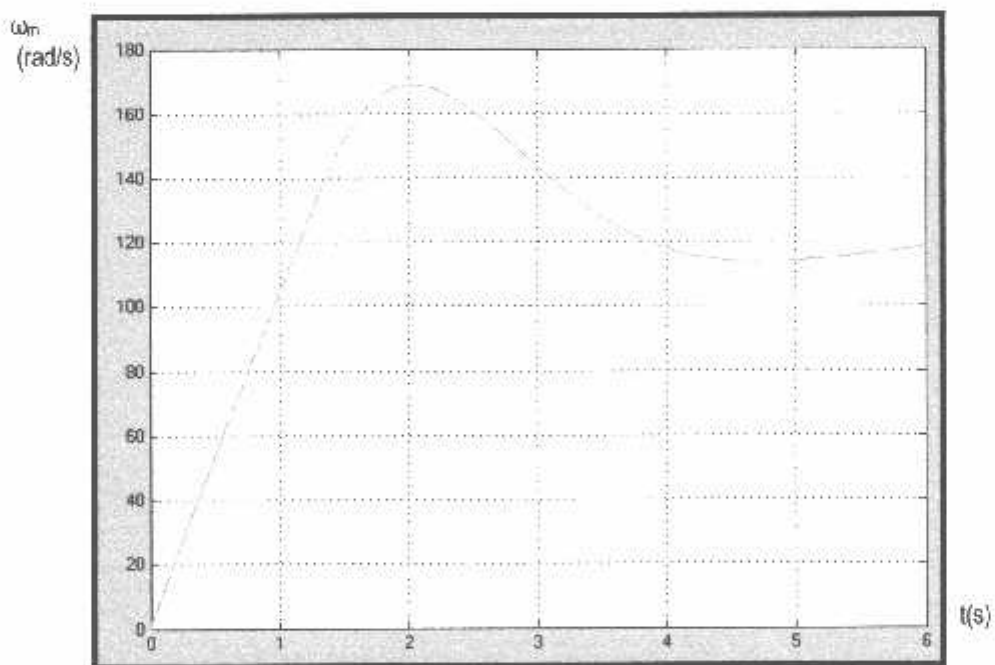


Gambar 4-14: Grafik torsi PID dengan setting point 100 rad/s, beban 0 Nm
Sumber : Simulasi Matlab 7.0

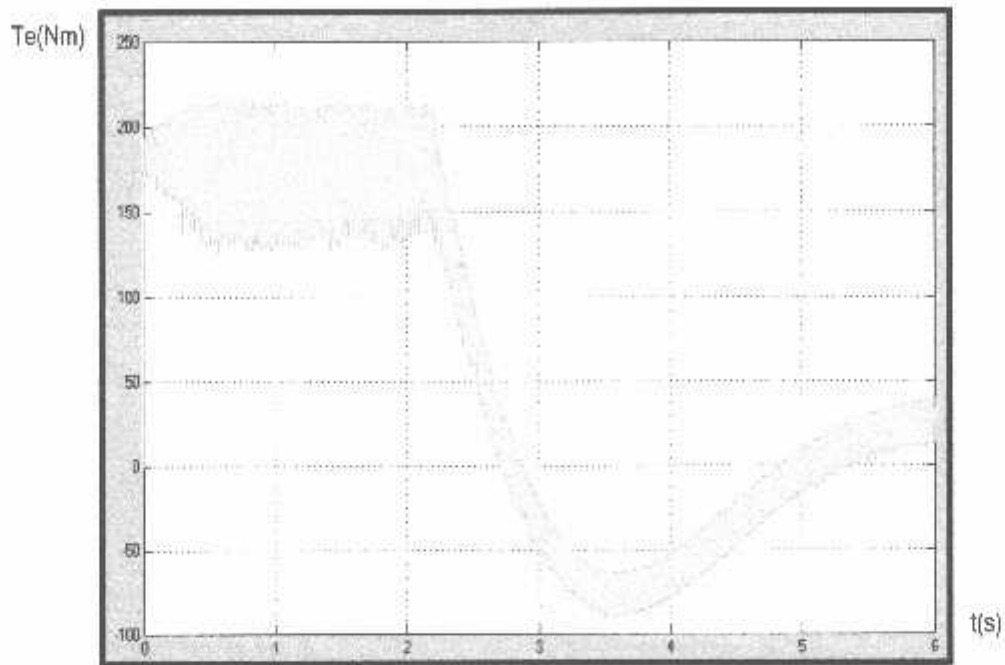


Gambar 4-15: Grafik Arus PID dengan setting point 100 rad/s, beban 0 Nm
Sumber : Simulasi Matlab 7.0

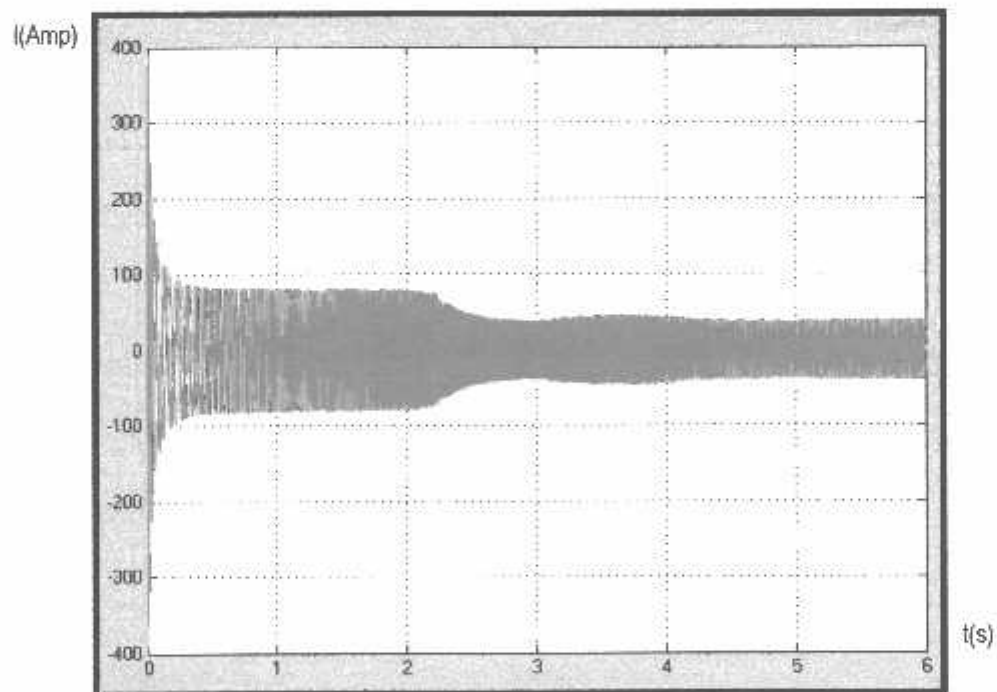
4.2.2 Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 0 Nm



Gambar 4-16: Grafik Kecepatan PID dengan setting point 120 rad/s, beban 0 Nm
Sumber : Simulasi Matlab 7.0

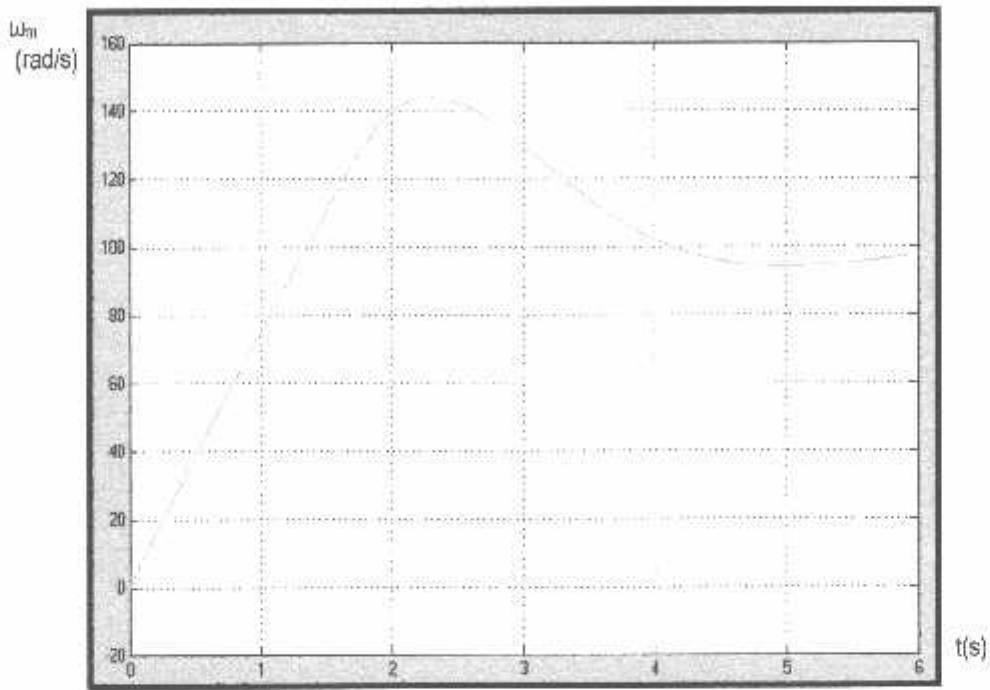


Gambar 4-17: Grafik Torsi PID dengan setting point 120 rad/s, beban Nol
Sumber : Simulasi Matlab 7.0



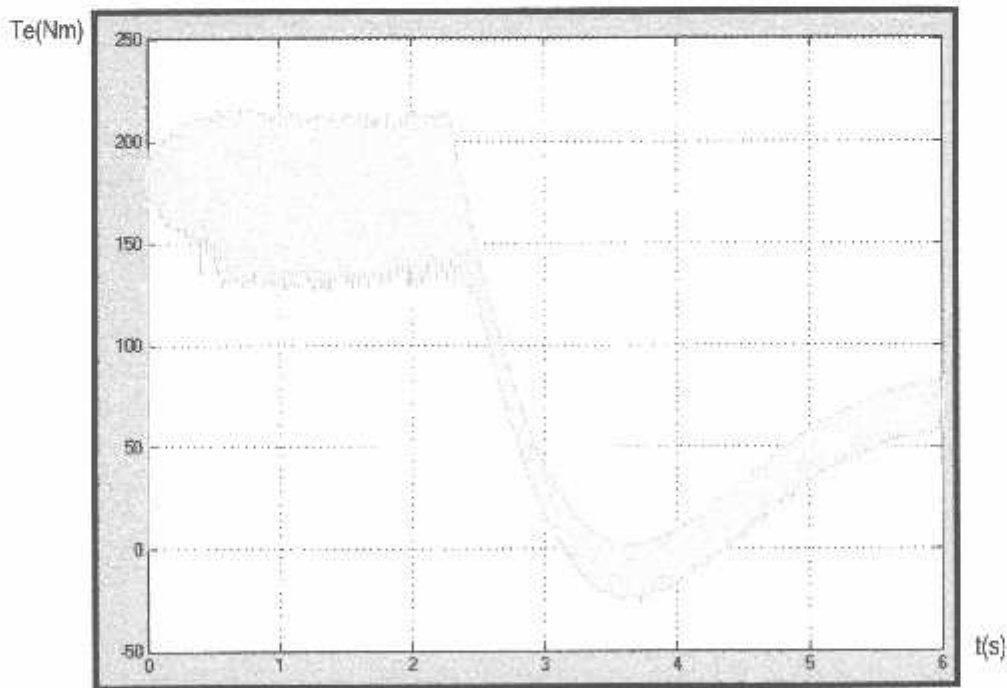
Gambar 4-18: Grafik Arus PID dengan setting point 120 rad/s, beban Nol
Sumber : Simulasi Matlab 7.0

4.2.3 Pengujian pada kecepatan sudut putar 100 rad/s dengan torsi beban 50 Nm



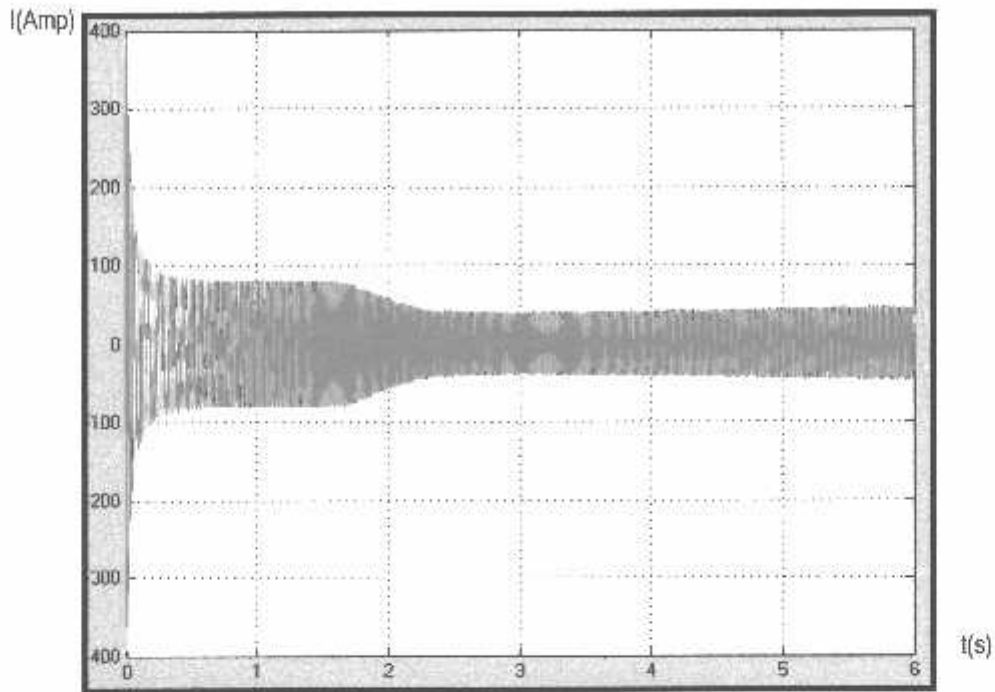
Gambar 4-19: Grafik Kecepatan PID dengan setting point 100 rad/s, beban 50 Nm

Sumber : Simulasi Matlab 7.0



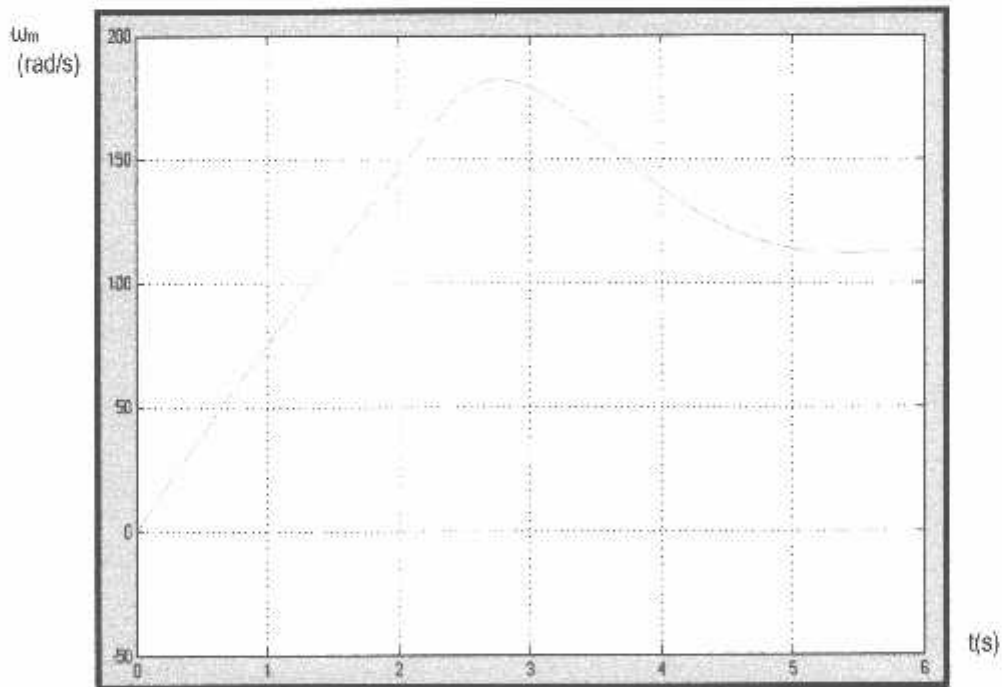
Gambar 4-20: Grafik Torsi PID dengan setting point 100 rad/s, beban 50 Nm

Sumber : Simulasi Matlab 7.0

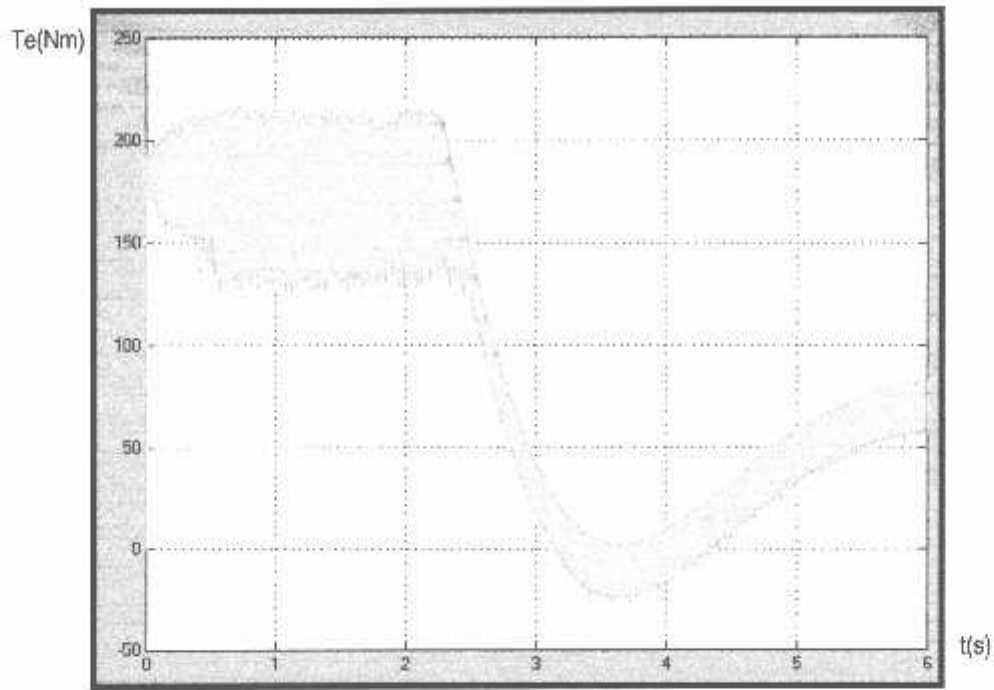


Gambar 4-21 : Grafik Arus PID dengan setting point 100 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0

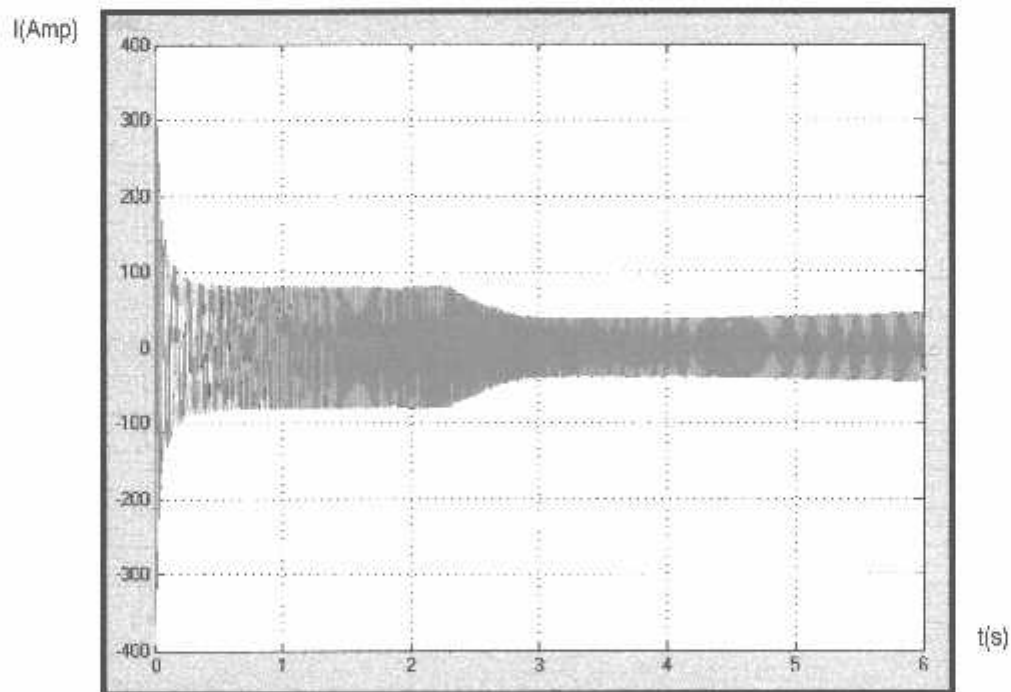
4.2.4 Pengujian pada kecepatan sudut putar 120 rad/s dengan torsi beban 50 Nm



Gambar 4-22: Grafik Kecepatan PID dengan setting point 120 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0



Gambar 4-23 : Grafik Torsi PID dengan setting point 120 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0

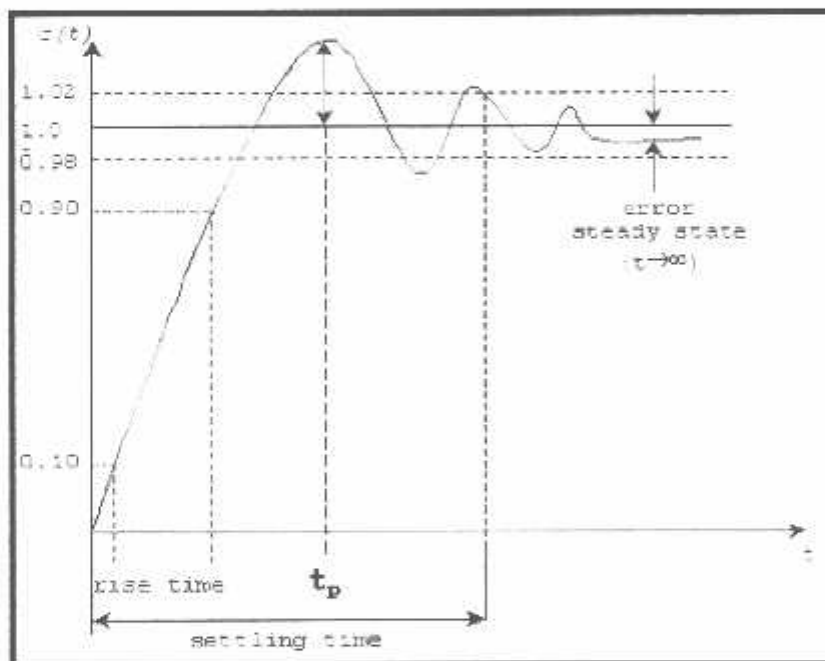


Gambar 4-24 : Grafik Arus PID dengan setting point 120 rad/s, beban 50 Nm
Sumber : Simulasi Matlab 7.0

4.3 Analisis Performansi Sistem

Untuk mengetahui performansi sistem, maka diperlukan adanya suatu kriteria dalam penentuannya. Performansi (unjuk kerja) sistem yang baik merupakan tujuan dari pengontrolan kecepatan sudut putar motor induksi 3 fasa ini. Dalam penelitian ini, sistem dinyatakan memiliki performansi yang baik jika memenuhi kriteria performansi yang ditetapkan yaitu *settling time*, *rise time* dan *error steady state*.

Pendefinisian kriteria performansi berdasarkan grafik respon sistem diperlihatkan pada gambar 4-25. Harga 1 pada gambar 4-25 menunjukkan nilai *setpoint* atau harga akhir yang ingin dicapai oleh sistem.



Gambar 4-25: Kriteria Performansi Sistem

Sumber : Ogata, Jilid 1, 1996

Berikut ini penjelasan dari beberapa komponen kriteria performansi yang akan digunakan adalah sebagai berikut :

- *Error Steady State*
 - Nilai selisih antara nilai set point dengan nilai *actual plant* pada kondisi *steady state*.
 - *Rise Time* atau Waktu Naik (t_r)
 - Waktu yang dibutuhkan oleh tanggapan untuk naik dari 10% sampai 90%.
 - *Settling Time* atau Waktu Turun (t_s)
 - Waktu yang diperlukan kurva respon untuk mencapai dan menetap dalam daerah sekitar harga akhir yang ukurannya dengan prosentase mutlak dari harga akhir (biasanya 2% sampai 5%).
-

Dari beberapa pengujian yang dilakukan, dapat dilihat perbandingannya simulasi antara FLC dengan PID. Secara tabel dapat dituliskan sebagai berikut:

Tabel 4.1 Tabel respon kecepatan motor 100 rad/s dengan beban 0 Nm

Performansi Sistem	FLC	PID
Settling Time (det)	0,817	4,25
Rise Time (det)	0,763	0,8
Error Steady State (%)	0,03	0,05

Sumber : Simulasi Matlab 7.0

Tabel 4.2 Tabel respon kecepatan motor 120 rad/s dengan beban 0 Nm

Performansi Sistem	FLC	PID
Settling Time (det)	1,012	5,45
Rise Time (det)	0,912	1,7
Error Steady State (%)	0,025	0,04

Sumber : Simulasi Matlab 7.0

Tabel 4.3 Tabel respon kecepatan motor 100 rad/s dengan Torsi beban 50 Nm

Performansi Sistem	FLC	PID
Settling Time (det)	1,2	5,0
Rise Time (det)	1,03	2,1
Error Steady State (%)	0,211	1,25

Sumber : Simulasi Matlab 7.0

Tabel 4.4 Tabel respon kecepatan motor 120 rad/s dengan Torsi beban 50 Nm

Performansi Sistem	FLC	PID
Settling Time (det)	1,44	5,4
Rise Time (det)	1,36	2,2
Error Steady State (%)	0,2	0,08

Sumber : Simulasi Matlab 7.0

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

- 1) Dari simulasi Pengaturan kecepatan sudut putar Motor Induksi 3 Fasa dengan menggunakan FLC berdasarkan pembelajaran jaringan kohonen didapatkan performansi sistem sebagai berikut :
 - Pada saat keadaan motor dalam keadaan tak berbeban/torsi beban adalah nol, untuk kecepatan 100 rad/s, *settling time* (t_s) sebesar 0.817 detik, waktu naik / *rise time* sebesar 0.763 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 0.03 %.
 - Pada saat keadaan motor dalam keadaan tak berbeban/torsi beban adalah nol, untuk kecepatan 120 rad/s, *settling time* (t_s) sebesar 1.012 detik, waktu naik / *rise time* (t_r) sebesar 0.912 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 0.025 %.
 - Pada saat motor dalam keadaan berbeban yaitu pada kecepatan 100 rad/s dengan beban 50 Nm, didapatkan performansi sistem sebagai berikut, *settling time* (t_s) sebesar 1.2 detik, waktu naik / *rise time* (t_r) sebesar 1.03 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 0.211 %
 - Pada saat motor diputar dengan kecepatan 120 rad/s dengan beban 50 Nm, *settling time* (t_s) sebesar 1.44 detik, waktu naik / *rise time* (t_r) sebesar 1.366 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 0.2 %
-

- Pengaturan kecepatan sudut putar motor induksi 3 fasa dengan metode *fuzzy* berdasarkan pembelajaran jaringan kohonen ini dapat beroperasi dengan baik dengan kesalahan pada keadaan mantap pada saat diputar dengan kecepatan 100rad/s dan 120 rad/s, pada saat berbeban dan tanpa beban, di bawah 2% dan waktu menetap yang cepat pula.

2) Dari simulasi Pengaturan kecepatan sudut putar Motor Induksi 3 Fasa dengan menggunakan **PID** didapatkan performansi sistem sebagai berikut :

- Pada saat keadaan motor dalam keadaan tak berbeban/torsi beban adalah nol, untuk kecepatan 100 rad/s, *settling time* (t_s) sebesar 4.25 detik, waktu naik / *rise time* (t_r) sebesar 0.8 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 0.05 %.
- Pada saat keadaan motor dalam keadaan tak berbeban/torsi beban adalah nol, Untuk kecepatan 120 rad/s, *settling time* (t_s) sebesar 5.45 detik, waktu naik / *rise time* (t_r) sebesar 1.7 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 0.04 % .
- Pada saat motor dalam keadaan berbeban yaitu pada kecepatan 100 rad/s dengan beban 50 Nm, didapatkan performansi sistem sebagai berikut, *settling time* (t_s) sebesar 5 detik, waktu naik / *rise time* (t_r) sebesar 2.1 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 1.25 % .
- Pada saat motor diputar dengan kecepatan 120 rad/s dengan beban 50 Nm, *settling time* (t_s) sebesar 5.45 detik, waktu naik / *rise time* sebesar 2.25 detik, dan kesalahan keadaan mantap / *error steady state* (e_{ss}) sebesar 0.08 % .

- Pengaturan kecepatan sudut putar motor induksi 3 fasa dengan menggunakan PID ini dapat beroperasi dengan baik dengan kesalahan pada keadaan mantap pada saat diputar dengan kecepatan 100rad/s dan 120 rad/s, pada saat berbeban dan tanpa beban, di bawah 2% dan waktu menetap yang cepat pula.
- 3) Dari beberapa pengujian yang dilakukan baik pada saat kondisi berbeban dan tanpa beban, dapat dilihat perbandingan performansi sistemnya, bahwa Pengaturan dengan menggunakan *Fuzzy* berdasarkan pembelajaran jaringan kohonen **lebih baik** daripada Pengaturan dengan menggunakan **PID**, hal ini dilihat dari :
- Sistem yang menggunakan FLC lebih cepat mencapai keadaan tunak (*steady state*) dari pada sistem yang menggunakan PID controller, yaitu sistem yang menggunakan FLC hanya membutuhkan waktu kurang 1 detik untuk mencapai *steady state* sedangkan yang menggunakan membutuhkan waktu lebih dari 6 detik untuk mencapai *steady state*.
 - Sistem yang menggunakan FLC memiliki nilai *Settling Time* yang lebih kecil dibandingkan yang menggunakan PID Controller, yaitu sistem dengan FLC *Settling Time* nya 1,2 detik sedangkan dengan PID *Settling Time* nya 5,0 detik
 - Sistem yang menggunakan PID controller *steady state error*nya sebesar 0.21 sedangkan dengan FLC sebesar 0.125.
-

5.2 Saran

Berbagai macam metode *fuzzy* banyak diterapkan dalam beberapa rangkaian pengontrolan, salah satunya dalam kontrol motor induksi 3 fasa dengan metode *fuzzy* kohonen, dan secara simulasi, metode ini telah mampu dan mempunyai performansi yang cukup baik. Untuk pengembangan lebih lanjut dapat dilakukan beberapa hal diantaranya :

1. Perlu adanya penelitian lebih lanjut tentang *Fuzzy Logic Controller (FLC)* terutama dalam penggabungannya dengan jaringan syaraf tiruan baik itu *supervised* ataupun *unsupervised learnig*.
 2. Studi lebih lanjut tentang pemetaan untuk fungsi keanggotan *fuzzy* baik masukan ataupun keluaran, basis aturan dan penggunaan metode defuzzifikasi yang berbeda.
 3. Untuk pengaplikasiannya di lapangan, dapat dilakukan pembuatan kontroler ini yang dapat menghubungkan program simulasi dengan alat secara riil sehingga hasilnya akan lebih tampak.
 4. Dapat dilakukan penelitian untuk mengaplikasikan kontroler ini pada motor induksi yang lain ataupun sistem lain yang memiliki dinamika sistem yang lebih kompleks.
-

DAFTAR PUSTAKA

Jurnal Seminar Nasional Ketenagalistrikan 2005, " *Analisa Pembelajaran Mandiri Untuk Fuzzy Logic Control Pada Pengaturan Motor Induksi* " Fakultas Teknik Elektro-Universitas Diponegoro.

Jurnal IES 2004, " *Pengaturan Kecepatan Motor Induksi Motor Induksi Eksitasi Pwm Sinusoida Menggunakan Fuzzy Logic Control* ", Politeknik elektronika Negeri Surabaya - ITS.

Jurnal IES 2004, " *Kendali Kecepatan Putar Motor Induksi Tiga Fasa Berbasis Logika Fuzzy Dengan Mikrokontroler AT89C52* ", Politeknik elektronika Negeri Surabaya - ITS.

Drs.Jong Jek Siang,M.Sc. " *Jaringan Syaraf Tiruan & Pemogramannya Menggunakan MATLAB* ",Penerbit ANDI.

Ir.Yusuf Ismail Nahkoda " *Diktat Mata Kuliah Logika Fuzzy* ",Teknik Elektro ITN Malang.

Ir.Yusuf Ismail Nahkoda " *Diktat Mata Kuliah Jaringan Syaraf Tiruan* ",Teknik Elektro ITN Malang.

Sri Kusumadewi, " *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB & EXCEL LINK* ",Penerbit GRAHA ILMU.

Sri Kusumadewi , " *Aplikasi LOGIKA FUZZY Untuk Pendukung Keputusan* ".Penerbit GRAHA ILMU.

Sri Kusumadewi, " *Analisis & Desain Sistem Fuzzy Menggunakan Tool Box Matlab* ", Penerbit GRAHA ILMU.

SonKuswadi, " *KENDALI CERDAS (INTELLIGENT CONTROL)* " , EEPIS Press Surabaya Indonesia,September 2000.

Drs.Yon Rijono, " *Dasar Teknik Tenaga Listrik Edisi Revisi* ", Penerbit Andi Yogyakarta.

Zuhal, " *Dasar Teknik Tenaga Listrik* ", Penerbit ITB Bandung.

LAMPIRAN 1



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : RIZKY ALAM PURYANTO
NIM : 02.12.075
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi :

**SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC CONTROL (FLC)
PADA PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA
DENGAN MENGGUNAKAN PERANGKAT LUNAK MATLAB 7.0**

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Kamis
Tanggal : 21 September 2006
Dengan Nilai : 85.2 (A)



Ir. Mochtar Asroni, MSME
Ketua

Panitia Ujian Skripsi



Ir. F. Yudi Limpraptono, MT
Sekretaris

Anggota Penguji



Ir.H. Taufik Hidayat, MT
Penguji Pertama



Ir. Choirul Saleh, MT
Penguji Kedua



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK

LEMBAR BIMBINGAN SKRIPSI

Nama Mahasiswa : RIZKY ALAM PURYANTO
NIM : 02.12.075
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi :

SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC CONTROL (FLC)
PADA PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA

INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura-gura No. 2
MALANG

No. 007862

UNTUK MAHASISWA

SURAT PUAS TUGAS / PRAKTIKUM

Nomor : ITN 05/SKN/XLVI/SPT/2006

Menyatakan bahwa mahasiswa :

Nama : Rizky Alam Puryanto NIM : 02.12.075
Jurusan : Teknik Elektro ST/S1
Angkatan : 2002 (Dua Ribu Dua)
Telah menyelesaikan tugas : "Simulasi Pembelajaran Mandiri Fuzzy Logic Control Pada pengaturan motor induksi 3 ϕ "
Semester : IX (Sembilan)
Dengan Hasil : 70

Malang, 19 Sept 2006

Dosen Asisten ybs.

(Signature)



**LEMBAR PENGAJUAN JUDUL SKRIPSI
JURUSAN TEKNIK ELEKTRO S-1**

Konsentrasi : Teknik Energi Listrik / ~~Teknik Elektronika~~

1	Nama Mahasiswa : Rizky Alam Puryanto	Nim : 02.12.075
2	Waktu pengajuan	Tanggal : 16 Bulan : Juni Tahun : 2006
3	Spesifikasi judul (berilah tanda silang)	
	a. Sistem Tenaga Elektrik b. Energi & Konversi Energi c. Tegangan Tinggi & Pengukuran <input checked="" type="checkbox"/> d. Sistem Kendali Industri	e. Elektronika & Komponen f. Elektronika Digital & Komputer g. Elektronika Komunikasi h. lainnya
4	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen *) : Ir. Oedho Pudi Pd	Mengetahui, Ketua Jurusan. Ir. F. Yudi Limpraptono, MT Nip. Y. 1039500274
5	Judul yang diajukan mahasiswa :	" Simulasi Pembelajaran Mandiri Fuzzy Logie Control (FLC) Pada Pengaturan Kecepatan Motor Induksi Tiga Fasa dengan menggunakan Perangkat Lunak Matlab 7.0 "
6	Perubahan Judul yang disetujui Dosen sesuai materi bidang ilmu
7	Catatan :	Disetujui, 16 Juni 2006... Dosen Oedho Pudi

Perhatian :

1. Formulir Pengajuan ini harap dikembalikan kepada jurusan paling lambat satu minggu setelah disetujui kelompok dosen keahlian dengan dilampirkan proposal skripsi beserta persyaratan skripsi sesuai form S-1
2. Keterangan : *) coret yang tidak perlu
**) dilingkari a, b, c, atau g sesuai bidang keahlian



Malang, 19 Juni 2006

Lampiran : Satu Lembar
Perihal : **Kesediaan Sebagai
Dosen Pembimbing**

Kepada : Yth. Bapak **Ir. Widodo Pudji M, MT**
Dosen Jurusan Elektro/ T.Energi Listrik
Institut Teknologi Nasional Malang
di-
Malang

Yang bertandatangan di bawah ini :

Nama : Rizky Alam Puryanto
Nim : 02.12.075
Semester : VIII
Jurusan : TEKNIK ELEKTRO
Konsentrasi : ENERGI LISTRIK S-1

Dengan ini mengajukan permohonan, kiranya Bapak bersedia menjadi Dosen Pembimbing Utama, untuk penyusunan Skripsi dengan judul :

**SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC
CONTROL (FLC) PADA PENGATURAN KECEPATAN
KECEPATAN MOTOR INDUKSI TIGA FASA DENGAN
MENGUNAKAN PERANGKAT LUNAK MATLAB 7.0**

Seperti proposal terlampir.


Adapun Tugas tersebut sebagai salah satu syarat untuk mendapatkan gelar Sarjana Teknik pada jurusan Teknik Elektro.


Demikian permohonan kami, atas kesediaan Bapak kami ucapkan terimakasih.

Mengetahui,
Ketua Jurusan
Teknik Elektro

Malang, 19 Juni 2006

Pemohon,


Ir.F.Yudi Limpraptono,MT
Nip.P.1039500274


Rizky Alam Puryanto
Nim : 02.12.075

Form S-3a



PERNYATAAN KESEDIAAN SEBAGAI DOSEN PEMBIMBING SKRIPSI

Sesuai dengan Permohonan Mahasiswa :

Nama : Rizky Alam Puryanto
Nim : 02.12.075
Semester : VIII
Jurusan : TEKNIK ELEKTRO
Konsentrasi : ENERGI LISTRIK S-1

Dengan ini menyatakan **bersedia** / ~~tidak bersedia~~ *) menjadi Dosen Pembimbing Utama / ~~Pendamping~~ *) , untuk penyusunan Skripsi Mahasiswa tersebut dengan judul :

SIMULASI PEMBELAJARAN MANDIRI FUZZY LOGIC CONTROL (FLC) PADA PENGATURAN KECEPATAN KECEPATAN MOTOR INDUKSI TIGA FASA DENGAN MENGGUNAKAN PERANGKAT LUNAK MATLAB 7.0

Demikian pernyataan ini kami buat untuk dapat dipergunakan sebagaimana mestinya.

Malang, 19 Juni 2006
Yang Membuat Pernyataan,

Ir. Widodo Pudji M, MT
Nip. P. 1028700171

Catatan :

1. Setelah disetujui agar formulir ini diserahkan mahasiswa/I yang bersangkutan kepada jurusan untuk diproses lebih lanjut.
2. *) Coret yang tidak perlu



FORMULIR BIMBINGAN SKRIPSI

Nama : Rizky Alam Puryanto
 Nim : 02.12.075
 Masa Bimbingan : 20 Juli 2006 s/d 20 Januari 2007
 Judul Skripsi : **Simulasi Pembelajaran Mandiri Fuzzy Logic Control (FLC) Pada Pengaturan Kecepatan Motor Induksi Tiga Fasa Dengan Menggunakan Perangkat Lunak Matlab 7.0**

NO	Tanggal	Uraian	Paraf Pembimbing
1.	25/07 '06	Bimbingan Bab I dan Bab II (Acc)	
2.	26/07 '06	Bimbingan Bab III dan Bab IV	
3.	26/07 '06	Revisi Bab III dan Bab IV (Penulisan kalimat, parameter, Bandingkan dan permasi Pw)	
4.	28/07 '06	Bimbingan Bab III, IV dan V	
5.	28/07 '06	Revisi Bab III, IV dan V (Penulisan kalimat, parameter, Keterangan gambar)	
6.	30/07 '06	Bimbingan Bab III, IV dan V	
7.	30/07 '06	Revisi Bab III, IV dan V (Penulisan kalimat, parameter grafik)	
8.	3/07 '06	Bimbingan Bab III, IV dan V (Acc)	
9.			
10.			

Malang, 3 Agustus 2006
 Dosen pembimbing

Ir. Widodo Pudji M, MT
 NIP.P 1028700171

Form. S-4b

Perhatian:

- Keterangan: *) Coret yang tidak perlu
 **) diilingkari a, b, c,atau g sesuai bidang keahlian

LAMPIRAN 2

DATA TRAINING

Kecepatan 100 rad/s beban nol

Time (t)	D_Error (rad/s)	Error (rad/s)
0	100	100
0.1	-0.00066	94.24
0.2	-0.00057	88.785
0.3	-0.00036	83.735
0.4	-0.00058	79.055
0.5	-0.00041	74.706
0.6	-0.00023	70.669
0.7	-0.00024	66.912
0.8	-0.00037	63.407
0.9	-0.00031	60.139
1	-0.00012	57.09
1.1	-0.00013	54.251
1.2	-0.00026	51.612
1.3	-0.00046	49.158
1.4	-0.00015	46.851
1.5	-0.0001	44.713
1.6	-0.00014	42.71
1.7	-0.00028	40.846
1.8	-8.56E-06	39.107
1.9	4.43E-05	37.487
2	2.81E-05	35.976
2.1	4.83E-05	34.565
2.2	-0.00033	33.245
2.3	9.55E-05	32.018
2.4	-0.00014	30.867
2.5	4.67E-05	29.793
2.6	-9.86E-05	28.799
2.7	6.83E-05	27.875
2.8	-0.00027	27.004
2.9	-0.00029	26.216
3	0.000176	25.454
3.1	0.000114	24.75
3.2	-0.00021	24.095
3.3	9.52E-05	23.474
3.4	-6.38E-05	22.894
3.5	-0.00011	22.339
3.6	-0.00028	21.819
3.7	-2.73E-05	21.303
3.8	-0.00023	20.793
3.9	-5.02E-06	20.344
4	7.63E-05	19.965
4.1	-0.00023	19.602
4.2	-0.00016	19.263
4.3	-0.00023	18.916
4.4	0.000151	18.558

4.5	6.91E-05	18.181
4.6	-0.0002	17.794
4.7	1.03E-05	17.471
4.8	-0.00012	17.117
4.9	0.000178	16.896
5	-0.00015	16.661
5.1	-0.00016	16.438
5.2	-6.01E-05	16.239
5.3	4.50E-05	15.897
5.4	9.39E-05	15.714
5.5	-8.36E-05	15.569
5.6	-0.00017	15.406
5.7	6.38E-05	15.261
5.8	-0.0002	15.024
5.9	-0.00016	14.807
6	0.000234	14.574
6.1	-0.00016	14.39
6.2	-4.08E-05	14.272
6.3	0.000155	14.112
6.4	-7.63E-05	13.932
6.5	0.000187	13.738
6.6	0.000139	13.548
6.7	0.000266	13.411
6.8	-9.49E-05	13.261
6.9	-0.00021	13.091
7	0.000116	12.888
7.1	-0.00017	12.763
7.2	0.000235	12.59
7.3	-2.78E-05	12.414
7.4	-0.0002	12.149
7.5	-6.65E-05	11.996
7.6	-2.00E-05	11.814
7.7	7.26E-05	11.688
7.8	7.81E-06	11.606
7.9	0.000268	11.506
8	-0.00013	11.394
8.1	0.000111	11.186
8.2	0.000206	10.999
8.3	0.000232	10.813
8.4	-0.00018	10.693
8.5	-0.00015	10.623
8.6	-2.99E-05	10.46
8.7	-0.00021	10.355
8.8	0.000228	10.332
8.9	0.000141	10.246
9	-8.87E-05	10.12
9.1	-6.17E-05	9.9414
9.2	0.000268	9.8087
9.3	3.19E-05	9.7561

9.4	-7.02E-05	9.6541
9.5	7.78E-05	9.5594
9.6	0.000158	9.5661
9.7	-3.69E-05	9.404
9.8	-0.00027	9.3547
9.9	-0.00012	9.3155
10	-0.00014	9.2268

DATA TRAINING

Kecepatan 100 rad/s beban 50 Nm

Time (t)	D. Error (rad/s)	Error (rad/s)
0	100	100
0.1	-0.0003	97.151
0.2	-0.00039	94.414
0.3	-0.00011	91.898
0.4	-0.00021	89.567
0.5	-3.56E-05	87.402
0.6	-0.00012	85.401
0.7	-0.00015	83.545
0.8	-0.00033	81.815
0.9	-0.00032	80.199
1	1.00E-05	78.689
1.1	-0.00034	77.281
1.2	1.33E-05	75.975
1.3	7.03E-05	74.764
1.4	0.000119	73.638
1.5	-0.00025	72.584
1.6	-0.00025	71.602
1.7	-0.00027	70.691
1.8	-0.00027	69.847
1.9	8.72E-05	69.058
2	0.000144	68.328
2.1	-0.0003	67.637
2.2	5.15E-06	66.997
2.3	0.000136	66.397
2.4	-0.00023	65.839
2.5	-1.61E-05	65.326
2.6	-0.00024	64.844
2.7	-6.07E-05	64.392
2.8	0.000153	63.97
2.9	0.000186	63.571
3	-0.00023	63.195
3.1	0.000146	62.856
3.2	-0.00011	62.535
3.3	0.000131	62.23
3.4	0.000199	61.949
3.5	3.40E-05	61.695
3.6	0.000218	61.46
3.7	5.81E-06	61.234
3.8	-4.06E-06	61.027
3.9	-6.05E-05	60.835
4	-0.00017	60.659
4.1	-0.00018	60.488
4.2	0.000178	60.327
4.3	-0.00024	60.179
4.4	-2.38E-05	60.037

4.5	-0.00012	59.912
4.6	-0.00019	59.796
4.7	-0.00023	59.679
4.8	-0.00018	59.574
4.9	-0.00021	59.483
5	-7.05E-05	59.399
5.1	-0.0001	59.314
5.2	-0.00013	59.235
5.3	0.000156	59.16
5.4	-0.00021	59.096
5.5	-0.0001	59.035
5.6	-4.15E-05	58.976
5.7	0.000245	58.929
5.8	-0.00013	58.878
5.9	-0.00022	58.831
6	5.03E-05	58.787
6.1	-8.87E-05	58.748
6.2	0.00019	58.711
6.3	0.000224	58.685
6.4	-0.0002	58.646
6.5	2.18E-05	58.616
6.6	-0.00025	58.586
6.7	1.74E-05	58.561
6.8	-7.83E-05	58.539
6.9	-0.00014	58.513
7	-9.46E-06	58.491
7.1	-1.61E-05	58.478
7.2	-0.00011	58.455
7.3	-6.23E-05	58.441
7.4	-0.00023	58.42
7.5	5.95E-05	58.404
7.6	-2.45E-05	58.386
7.7	0.000113	58.376
7.8	0.000185	58.361
7.9	0.000242	58.345
8	-0.00017	58.338
8.1	0.000172	58.332
8.2	0.000212	58.324
8.3	-9.50E-05	58.316
8.4	-2.20E-06	58.306
8.5	-2.03E-05	58.295
8.6	0.000156	58.288
8.7	-0.00021	58.277
8.8	0.000133	58.27
8.9	-0.00021	58.263
9	-0.00011	58.258
9.1	-0.00022	58.253
9.2	-1.07E-05	58.256
9.3	-4.13E-05	58.255

9.4	-0.00021	58.255
9.5	-0.00021	58.252
9.6	0.000127	58.248
9.7	-0.0002	58.245
9.8	0.00024	58.243
9.9	0.000103	58.234
10	-3.01E-05	58.233

DATA TRAINING

Kecepatan 120 rad/s beban Nol

Time (t)	D_Error (rad/s)	Error (rad/s)
0	120	120
0.1	-0.03121	113.48
0.2	-0.00245	107.48
0.3	0.004438	102.11
0.4	-0.03106	98.03
0.5	0.023633	94.557
0.6	-0.00605	92.145
0.7	0.0097	89.874
0.8	-0.01859	86.894
0.9	0.015924	84.624
1	-0.01654	82.987
1.1	0.003953	81.602
1.2	0.004263	80.092
1.3	-0.01343	79.323
1.4	-0.02462	78.808
1.5	0.018812	78.285
1.6	-0.01822	77.623
1.7	0.004711	76.687
1.8	0.018288	76.389
1.9	-0.00491	75.536
2	-0.00377	75.251
2.1	0.000839	75.016
2.2	-0.02219	74.761
2.3	0.023599	74.441
2.4	-0.0266	74.18
2.5	0.010818	74.124
2.6	-0.02761	73.734
2.7	-2.20E-07	73.664
2.8	-0.00166	73.421
2.9	-0.00356	72.974
3	0.030048	73.028
3.1	0.01287	73.093
3.2	-0.01194	73.474
3.3	0.012083	73.181
3.4	0.022151	73.048
3.5	0.015608	73.101
3.6	-0.00062	73.146
3.7	-0.01744	73.486
3.8	0.012065	72.963
3.9	0.005634	73.089
4	0.005775	73.116
4.1	-0.00583	73.363
4.2	0.015501	73.133
4.3	-0.0008	72.75

4.4	0.019483	72.363
4.5	0.016947	72.179
4.6	-0.01153	72.167
4.7	0.003147	71.698
4.8	-0.02062	71.651
4.9	0.004661	71.654
5	-0.00031	71.411
5.1	-0.00459	71.343
5.2	0.007379	71.488
5.3	-0.02448	71.36
5.4	-0.01563	71.125
5.5	-0.0189	70.905
5.6	-0.02457	71.047
5.7	0.006496	71.023
5.8	-0.00409	70.905
5.9	0.023667	71.163
6	-0.01737	71.319
6.1	0.021666	71.376
6.2	0.002917	71.444
6.3	0.030342	71.227
6.4	-0.01734	70.806
6.5	0.017895	70.985
6.6	-0.01408	70.777
6.7	0.000719	70.822
6.8	-0.0243	70.534
6.9	0.007928	70.532
7	-0.01003	70.644
7.1	0.005589	70.946
7.2	0.006637	71.162
7.3	-0.01213	70.893
7.4	0.014412	70.955
7.5	0.02504	70.858
7.6	0.017153	70.649
7.7	-0.00644	70.085
7.8	-0.00528	69.876
7.9	0.024706	69.88
8	-0.00214	69.9
8.1	-0.02703	69.85
8.2	0.017788	69.843
8.3	0.002659	69.886
8.4	-0.0125	69.843
8.5	-0.00457	69.819
8.6	0.009082	69.88
8.7	-0.02234	69.828
8.8	0.001536	69.838
8.9	0.017238	69.922
9	-0.0057	69.865
9.1	-0.00519	69.825
9.2	0.024704	69.866

9.3	-0.00214	69.896
9.4	-0.02703	69.849
9.5	0.017791	69.843
9.6	0.002659	69.886
9.7	-0.0125	69.843
9.8	-0.00457	69.819
9.9	0.009082	69.88
10	-0.02234	69.828

DATA TRAINING

Kecepatan 120 rad/s beban 50 Nm

Time (t)	D_Error (rad/s)	Error (rad/s)
0	120	120
0.1	-0.00037	116
0.2	-0.00052	112.17
0.3	-0.0002	108.65
0.4	-0.00048	105.39
0.5	-0.0004	102.35
0.6	-0.0003	99.529
0.7	-0.00012	96.897
0.8	-0.0004	94.459
0.9	-0.0001	92.175
1	-0.0004	90.061
1.1	-0.00024	88.088
1.2	-8.07E-05	86.252
1.3	2.75E-05	84.533
1.4	-0.00034	82.923
1.5	1.55E-05	81.427
1.6	-0.00019	80.029
1.7	-0.00014	78.735
1.8	-0.00027	77.523
1.9	-0.00034	76.392
2	-9.95E-05	75.344
2.1	-0.00033	74.366
2.2	-7.98E-05	73.457
2.3	-0.0003	72.607
2.4	-0.00037	71.817
2.5	-0.00027	71.085
2.6	3.73E-05	70.402
2.7	-0.00029	69.768
2.8	-0.00013	69.166
2.9	-0.00029	68.598
3	8.06E-05	68.073
3.1	-0.00014	67.582
3.2	-8.95E-05	67.129
3.3	-5.77E-05	66.704
3.4	2.97E-05	66.315
3.5	0.000144	65.953
3.6	-0.00016	65.609
3.7	0.000174	65.281
3.8	-0.00025	64.979
3.9	8.65E-05	64.692
4	-0.00022	64.433
4.1	-2.45E-05	64.188
4.2	-3.50E-05	63.952
4.3	-0.00022	63.74

4.4	0.000173	63.556
4.5	0.000142	63.388
4.6	-5.54E-05	63.227
4.7	0.000111	63.093
4.8	0.000199	62.947
4.9	0.000227	62.812
5	5.10E-05	62.671
5.1	-0.00018	62.549
5.2	0.000183	62.433
5.3	0.000193	62.313
5.4	0.000217	62.216
5.5	-0.00013	62.115
5.6	-0.00018	62.022
5.7	-9.20E-05	61.933
5.8	0.000181	61.844
5.9	0.000129	61.771
6	-0.00024	61.695
6.1	0.000101	61.634
6.2	0.000155	61.564
6.3	-0.00019	61.511
6.4	-0.00018	61.453
6.5	0.000232	61.397
6.6	-0.0002	61.36
6.7	-0.0002	61.321
6.8	-0.00011	61.282
6.9	-1.94E-05	61.242
7	-0.00018	61.21
7.1	9.62E-05	61.174
7.2	0.000122	61.136
7.3	5.56E-06	61.11
7.4	0.000166	61.083
7.5	-0.00017	61.058
7.6	7.04E-05	61.041
7.7	-0.00024	61.026
7.8	-5.57E-05	61.007
7.9	-0.0001	60.996
8	4.58E-06	60.978
8.1	0.000176	60.962
8.2	3.65E-05	60.959
8.3	2.30E-05	60.95
8.4	-0.00016	60.95
8.5	0.000229	60.951
8.6	0.000176	60.958
8.7	-0.00019	60.967
8.8	0.000161	60.961
8.9	-0.00015	60.965
9	-0.00025	60.966
9.1	0.000167	60.971
9.2	0.00012	60.969

9.3	-0.0002	60.951
9.4	-1.67E-05	60.936
9.5	3.45E-05	60.911
9.6	-0.00025	60.898
9.7	-2.67E-06	60.889
9.8	-6.12E-05	60.872
9.9	-0.00014	60.86
10	4.06E-05	60.85

PROGRAM
KOHONEN 100
RAD/S

error - [100	14.272	63.97
94.24	14.112	63.571
88.785	13.932	63.195
83.735	13.738	62.856
79.055	13.548	62.535
74.706	13.411	62.23
70.669	13.261	61.949
66.912	13.091	61.695
63.407	12.888	61.46
60.139	12.763	61.234
57.09	12.59	61.027
54.251	12.414	60.835
51.612	12.149	60.659
49.158	11.996	60.488
46.851	11.814	60.327
44.713	11.688	60.179
42.71	11.606	60.037
40.846	11.506	59.912
39.107	11.394	59.796
37.487	11.186	59.679
35.976	10.999	59.574
34.565	10.813	59.483
33.245	10.693	59.399
32.018	10.623	59.314
30.867	10.46	59.235
29.793	10.355	59.16
28.799	10.332	59.096
27.875	10.246	59.035
27.004	10.12	58.976
26.216	9.9414	58.929
25.454	9.8087	58.878
24.75	9.7561	58.831
24.095	9.6541	58.787
23.474	9.5594	58.748
22.894	9.5661	58.711
22.339	9.404	58.685
21.819	9.3547	58.646
21.303	9.3155	58.616
20.793	9.2268	58.586
20.344	100	58.561
19.965	97.151	58.539
19.602	94.414	58.513
19.263	91.898	58.491
18.916	89.567	58.478
18.558	87.402	58.455
18.181	85.401	58.441
17.794	83.545	58.42
17.471	81.815	58.404
17.117	80.199	58.386
16.896	78.689	58.376
16.661	77.281	58.361
16.438	75.975	58.345
16.239	74.764	58.338
15.897	73.638	58.332
15.714	72.584	58.324
15.569	71.602	58.316
15.406	70.691	58.306
15.261	69.847	58.295
15.024	69.058	58.288
14.807	68.328	58.277
14.574	67.637	58.27
14.39	66.997	58.263
	66.397	58.258
	65.839	58.253
	65.326	58.256
	64.844	58.255
	64.392	58.255

58.252	100.46	-9.86E-05
58.248	100.46	6.83E-05
58.245	100.46	-0.0002724
58.243	100.46	-0.00028517
58.234	100.46	0.00017564
58.233	100.46	0.00011437
100	100.46	-0.00020651
100.06	100.46	9.52E-05
100.03	100.46	-6.38E-05
100.05	100.46	-0.00011112
100.07	100.46	-0.00027832
100.1	100.46	-2.73E-05
100.12	100.46	-0.00022822
100.14	100.47	-5.02E-06
100.16	100.46	7.63E-05
100.18	100.46	-0.00022657
100.2	100.47	-0.00016011
100.21	100.47	-0.00023083
100.23	100.47	0.00015091
100.25	100.47	6.91E-05
100.27	100.47	-0.00020188
100.28	100.47	1.03E-05
100.28	100.47	-0.0001165
100.3	100.47	0.00017776
100.31	100.47	-0.00014978
100.32	100.47	-0.00015849
100.32	100.47	-6.01E-05
100.32	100.47	4.50E-05
100.33	100.46	9.39E-05
100.34	100.46	-8.36E-05
100.35	100.47	-0.00016889
100.35	100.46	6.38E-05
100.36	100.46	-0.00020075
100.37	100.46	-0.00015531
100.37	100.46	0.00023446
100.38	100.46	-0.00016335
100.38	100.46	-4.08E-05
100.38	100.46	0.00015456
100.38	100.46	-7.63E-05
100.39	100.47	0.00018739
100.39];	0.00013886
100.39	derror=[100	0.0002655
100.4	-0.000663	-9.49E-05
100.4	-0.00057333	-0.00020852
100.4	-0.00036028	0.00011565
100.41	-0.00057586	-0.00017453
100.41	-0.00040824	0.00023521
100.4	-0.00023191	-2.78E-05
100.41	-0.00024433	-0.0002009
100.42	-0.00037034	-6.65E-05
100.43	-0.00031135	-2.00E-05
100.43	-0.00011983	7.26E-05
100.43	-0.00013132	7.81E-06
100.43	-0.00025884	0.00026819
100.44	-0.00045932	-0.00013044
100.44	-0.00014812	0.00011053
100.44	-0.0001035	0.00020617
100.44	-0.00013889	0.00023169
100.44	-0.0002796	-0.00018378
100.44	-8.56E-06	-0.00014504
100.45	4.43E-05	-2.99E-05
100.45	2.81E-05	-0.00021144
100.45	4.83E-05	0.00022769
100.45	-0.00033387	0.00014084
100.46	9.55E-05	-8.87E-05
100.46	-0.00013827	-6.17E-05
100.46	4.67E-05	0.0002682

3.19E-05	-0.00022438	-0.00017521
-7.02E-05	5.03E-05	2.91E-05
7.78E-05	-8.87E-05	-0.00021334
0.00015829	0.00018995	3.30E-05
-3.69E-05	0.00022356	-0.00013487
-0.00027432	-0.00019825	-0.00025098
-0.00011932	2.18E-05	-8.00E-05
-0.00014237	-0.00024651	-0.00025417
100	1.74E-05	0.00015779
-0.00030221	-7.83E-05	-0.0001335
-0.0003929	-0.00014401	0.00021705
-0.00011486	-9.46E-06	-0.00017668
-0.00020775	-1.61E-05	0.00012788
-3.56E-05	-0.00010911	0.00017874
-0.0001195	-6.23E-05	0.00019892
-0.00015293	-0.00022683	8.64E-05
-0.00032941	5.95E-05	0.00018349
-0.00031787	-2.45E-05	0.00013531
1.00E-05	0.00011287	0.00020087
-0.00034301	0.0001854	3.56E-05
1.33E-05	0.00024157	-5.67E-06
7.03E-05	-0.00016547	0.00010522
0.00011934	0.0001715	
-0.00024662	0.00021162	
-0.00025465	-9.50E-05	
-0.00027495	-2.20E-06	
-0.00026601	-2.03E-05	
8.72E-05	0.00015563	
0.00014444	-0.00020822	
-0.00030408	0.00013274	
5.15E-06	-0.00020699	
0.00013595	-0.00011326	
-0.00023145	-0.00022377	
-1.61E-05	-1.07E-05	
-0.00023506	-4.13E-05	
-6.07E-05	-0.0002098	
0.0001534	-0.00020924	
0.00018606	0.00012723	
-0.00023238	-0.0002046	
0.00014589	0.00024049	
-0.00011105	0.0001033	
0.00013065	-3.01E-05	
0.00019914	100	
3.40E-05	8.32E-05	
0.00021758	1.74E-05	
5.81E-06	-5.00E-05	
-4.06E-06	0.00012798	
-6.05E-05	0.00010266	
-0.00017436	-9.62E-05	
-0.00018338	0.00016783	
0.00017818	0.00010621	
-0.00023501	-3.35E-05	
-2.38E-05	0.00015959	
-0.00011767	-7.75E-05	
-0.00018942	-0.00010946	
-0.00022684	0.00019151	
-0.00017571	-0.00019735	
-0.00021129	-0.0002116	
-7.05E-05	8.92E-05	
-0.00010418	0.00016086	
-0.00013476	0.0002011	
0.00015639	0.00018509	
-0.00021018	0.00026762	
-0.00010321	0.00023607	
-4.15E-05	0.00020053	
0.00024489	-0.0001867	
-0.00012636	-0.00017457	


```

-4.63E-05
0.00017656
-0.00016001
9.75E-05
-0.00010448
0.00017411
-2.17E-06
0.0001085
-0.00012274
0.00012182
7.06E-05
-0.00010516
-1.24E-05
7.70E-05
5.68E-05
-0.00013548
0.00025319
0.00010488
5.33E-05
0.00023257
9.20E-05
0.00024638
-3.11E-05
0.00016912
-0.00021786
0.000199
-7.30E-06
0.0001959
-0.00011747
0.00024701
-0.0001985
5.38E-05
-4.81E-05
-0.00015068
-0.00016772
0.00016606
0.00010686
0.00011157
-8.72E-06
-1.92E-05
0.00013224
0.00017356
-0.00019346
0.00019787
-0.00012129
0.00020838
0.0001798
0.00012061
6.86E-06
5.12E-05
9.05E-05
-0.00016094
0.00015183
0.00019439
];

x=transpose(error);
y=transpose(derror);
p=[x;y];%vektor input data

%pembangunan jaringan syaraf
BanyakKlas=7;%membership function input fuzzy
net=newnet([-200 200;-200 200],BanyakKlas);

%bobot input, lapisan, dan bias
bobot_input = net.IW{1,1};
bobot_bias_input = net.b{1,1};

%pelatihan jaringan
net.trainParam.epochs = 500;
net.trainParam.goal=0.001;
net = train(net,p);

%simulasi jaringan
Ft=sim(net,p);
F=vec2ind(Ft);

%grafik hasil simulasi jaringan
r=rand(1,BanyakKlas);
g=rand(1,BanyakKlas);
b=rand(1,BanyakKlas);
for i=1:size(p,2),
    for j=1:BanyakKlas,
        if F(i)==j,
            plot(p(1,i),p(2,i),'marker','*', 'color',[r(j) g(j) b(j)]);
            hold on;
        plot(net.IW{1,1}(j,1),net.IW{1,1}(j,2),'marker','o','color',[r(j) g(j) b(j)]);
        text(net.IW{1,1}(j,1)+.1,net.IW{1,1}(j,2)+.1,int2str(j));
    end;
end;
end;
end;

```

PROGRAM
KOHONEN 120
RAD/S

error = 1120

113.48	71.36	82.923
107.48	71.125	81.427
102.11	70.905	80.029
98.03	71.047	78.735
94.557	71.023	77.523
92.145	70.905	76.392
89.874	71.163	75.344
86.894	71.319	74.366
84.624	71.376	73.457
82.987	71.444	72.607
81.602	71.227	71.817
80.092	70.806	71.085
79.323	70.985	70.402
78.808	70.777	69.768
78.285	70.822	69.166
77.623	70.534	68.598
76.687	70.532	68.073
76.389	70.644	67.582
75.536	70.946	67.129
75.251	71.162	66.704
75.016	70.893	66.315
74.761	70.955	65.953
74.441	70.856	65.609
74.18	70.649	65.281
74.124	70.085	64.979
73.734	69.876	64.692
73.664	69.88	64.433
73.621	69.9	64.188
73.421	69.85	63.952
73.421	69.843	63.74
73.028	69.886	63.556
73.028	69.843	63.388
73.093	69.819	63.227
73.474	69.88	63.093
73.181	69.828	62.947
73.048	69.838	62.812
73.101	69.922	62.671
73.146	69.865	62.549
73.486	69.825	62.433
72.963	69.866	62.313
73.089	69.896	62.216
73.116	69.849	62.115
73.363	69.843	62.022
73.133	69.886	61.933
72.75	69.843	61.844
72.363	69.819	61.771
72.179	69.88	61.695
72.167	69.828	61.634
71.698	120	61.564
71.651	116	61.511
71.654	112.17	61.453
71.411	108.65	61.397
71.343	105.39	61.36
71.488	102.35	61.321
	99.529	61.282
	96.897	61.242
	94.459	61.21
	92.175	61.174
	90.061	61.136
	88.088	61.11
	86.252	61.083
	84.533	61.058

61.041	104.77	103.59
61.026	104.69	103.59
61.007	104.62];
60.996	104.55	derror-[120
60.978	104.48	-0.031212
60.962	104.43	-0.0024474
60.959	104.37	0.0044384
60.95	104.32	-0.031057
60.95	104.27	0.023633
60.951	104.22	-0.0060533
60.958	104.17	0.0097001
60.967	104.13	-0.01859
60.961	104.09	0.015924
60.965	104.06	-0.016541
60.966	104.03	0.0039532
60.971	103.99	0.0042625
60.969	103.97	-0.01343
60.951	103.94	-0.024619
60.936	103.92	0.018812
60.911	103.9	-0.018223
60.898	103.88	0.0047111
60.889	103.86	0.018288
60.872	103.84	-0.0049141
60.86	103.83	-0.0037676
60.85	103.81	0.00083895
120	103.8	-0.022189
118.91	103.78	0.023599
117.8	103.77	-0.026604
116.81	103.76	0.010818
115.89	103.75	-0.027606
115.03	103.74	-2.20E-07
114.23	103.73	-0.0016649
113.49	103.72	-0.00356
112.81	103.71	0.030048
112.17	103.7	0.01287
111.57	103.69	-0.011935
111.02	103.69	0.012083
110.51	103.68	0.022151
110.03	103.67	0.015608
109.58	103.67	-0.00061581
109.16	103.66	-0.017442
108.77	103.65	0.012065
108.41	103.65	0.005634
108.08	103.64	0.0057749
107.77	103.64	-0.0058299
107.48	103.63	0.015501
107.21	103.63	-0.0007992
106.96	103.63	0.019483
106.73	103.62	0.016947
106.52	103.62	-0.011526
106.31	103.61	0.0031465
106.13	103.61	-0.020616
105.96	103.61	0.0046614
105.79	103.6	-0.00031256
105.65	103.6	-0.0045871
105.5	103.6	0.0073787
105.37	103.6	-0.024481
105.26	103.6	-0.015631
105.14	103.6	-0.018895
105.03	103.6	-0.02457
104.94	103.6	0.0064956
104.85	103.6	-0.004086

0.023667	-9.95E-05	3.65E-05
-0.017372	-0.00032624	2.30E-05
0.021666	-7.98E-05	-0.00016172
0.0029174	-0.00029886	0.00022949
0.030342	-0.00036727	0.00017624
-0.017343	-0.00026708	-0.00019362
0.017895	3.73E-05	0.00016091
-0.014084	-0.00028562	-0.0001519
0.00071927	-0.00012958	-0.0002499
-0.024299	-0.00028577	0.00016667
0.007928	8.06E-05	0.00012005
-0.010026	-0.00013571	-0.00019919
0.0055887	-8.95E-05	-1.67E-05
0.0066371	-5.77E-05	3.45E-05
-0.012125	2.97E-05	-0.00024746
0.014412	0.00014447	-2.67E-06
0.02504	-0.0001576	-6.12E-05
0.017153	0.00017369	-0.00014465
-0.0064406	-0.00024645	4.06E-05
-0.0052778	6.65E-05	120
0.024706	-0.00022337	-2.82E-05
-0.0021444	-2.45E-05	1.76E-05
-0.027032	-3.50E-05	0.0001167
0.017788	-0.00021649	-0.00013538
0.0026586	0.00017262	-0.00028136
-0.012499	0.00014222	0.00014762
-0.0045746	-5.54E-05	-5.07E-05
0.0090821	0.00011069	-0.00025692
-0.022337	0.00019902	0.00017671
0.0015363	0.00022745	0.00015248
0.017238	5.10E-05	-0.00027294
-0.0057027	-0.00018378	0.00012249
-0.0051854	0.00018251	-0.00011697
0.024704	0.00019282	-9.91E-05
-0.0021395	0.00021685	7.65E-05
-0.027028	-0.00012798	4.89E-06
0.017791	-0.00017516	-0.00019065
0.0026587	-9.20E-05	0.00013686
-0.012499	0.00018125	-0.0002266
-0.0045747	0.00012896	-0.00013802
0.0090821	-0.00023767	0.00017565
-0.022337	0.00010093	-0.00019929
120	0.0001549	-0.0002637
-0.0003721	-0.00019234	-0.00015858
-0.00051889	-0.00017514	-0.00019044
-0.00020006	0.00023155	0.00013721
-0.0004762	-0.00020012	0.00017268
-0.00039674	-0.00020439	-0.00023625
-0.00030114	-0.00011098	0.00012948
-0.00011988	-1.94E-05	0.00014191
-0.00039701	-0.00017809	-1.63E-05
-0.00010024	9.62E-05	0.00022156
-0.00040355	0.00012205	-0.00022332
-0.00024033	5.56E-06	0.00021147
-8.07E-05	0.00016632	-9.89E-05
2.75E-05	-0.0001747	5.83E-05
-0.00032699	7.04E-05	0.00021667
1.55E-05	-0.00023818	0.00017242
-0.00019114	-5.57E-05	-1.63E-05
-0.00014367	-0.00010481	-9.78E-05
-0.00027411	4.58E-06	2.37E-05
-0.00033731	0.00017603	0.000209

```

0.00011314
-3.05E-06
-0.00024852
3.67E-05
6.18E-05
5.46E-05
-0.0001916
0.00018702
-8.03E-05
0.00018146
-5.22E-05
7.79E-05
0.00016445
0.00012703
0.00021407
-0.00010122
-0.0001933
-0.00015737
-0.00022914
0.00020752
0.00020023
-0.00019238
-5.66E-05
0.00018042
0.00022467
-3.26E-05
-0.00010631
0.00015085
-8.87E-05
-0.00017027
0.00011879
0.00019208
0.00019584
-4.06E-06
-3.60E-05
-5.83E-07
7.45E-05
-0.00014939
5.61E-05
-0.00013662
-0.00024147
5.64E-05
0.00010516
0.00016409
-0.00016875
0.00023024
-7.11E-05
-0.00022958
5.62E-05
0.00019045
0.00018093
0.00022038
-0.00021662
0.0002553
0.00018122
-0.00021118
-0.00021871
-0.00022873
];

```

```

x=transpose(error);
y=transpose(derror);
p=[x;y];%vektor input data

%pembangunan jaringan syaraf
BanyakKlas=7;%membership function input fuzzy
nct=newc([-200 200;-200 200],BanyakKlas);

%bobot input, lapisan, dan bias
bobot_input = nct.IW{1,1}
bobot_bias_input = net.b{1,1}

%pelatihan jaringan
net.trainParam.epochs = 500;
nct.trainParam.goal=0.001;
net=train(net,p);

%simulasi jaringan
Ft=sim(nct,p);
F=vec2ind(Ft)

%grafik hasil simulasi jaringan
r=rand(1,BanyakKlas);
g=rand(1,BanyakKlas);
b=rand(1,BanyakKlas);
for i = 1:size(p,2)
    for j=1:BanyakKlas
        if F(i)==j,
            plot(p(1,i),p(2,i),'marker','*', 'color',[r(j) g(j) b(j)]);
            hold on;
        plot(net.IW{1,1}(j,1),nct.IW{1,1}(j,2),'marker','o','color',[r(j) g(j) b(j)]);
        text(nct.IW{1,1}(j,1)+.1,nct.IW{1,1}(j,2)+.1,int2str(j));
    end;
end;
end;
end;

```