

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



ANALISIS PENGARUH PENUKARAN FASA TERHADAP
KETIDAKSEIMBANGAN FASA PADA GARDU INDUK
BLIMBING DENGAN MENGGUNAKAN METODE
SIMULATED ANNEALLING

MILITARY
PERPUSTAKAAN
ITN MALANG

SKRIPSI

Disusun Oleh :
DIDIK PURNOMO
NIM. 01.12.128

MARET 2006

LEMBAR PERSEMBAHAN

Segala puji syukur kupanjatkan kepada Allah Ta'ala atas segala anugerah yang diberikan kepada hamba ini, khususnya anugerah atas terselesaikannya studiku di kampus ITN Malang yang tercinta. Semoga Allah Ta'ala menjadikan ilmuku bermanfaat baik terhadap masyarakat, nusa, bangsa, dan agama. Amiin

Buat kedua orang tuaku dan guru guruku digading aku haturkan beribu – ribu terimakasih, juga kepada kakak – kakakku, teman – temanku & sobat – sobatku semua serta tak lupa kuucapkan makasih buat mas ugro, Aroma, Ilmi, akhlis, wawan, yunus, eko, ileng atas semua dukungan dan bantuannya.

Skripsi ini kupersembahkan buat para pembaca semuanya khususnya kepada yang berminat dalam mempelajari masalah bidang kelistrikan.

Akhir kata, tiada kalimat yang dapat aku sampaikan untuk melukiskan kebahagiaan ini selain ungkapan rasa syukurku kepada Allah Ta'ala, kemudian juga rasa terimakasihku kepada semua pihak yang telah membantu memberikan bantuan baik moril, materiil maupun spirituil.

ABSTRAKSI

ANALISIS PENGARUH PENUKARAN FASA TERHADAP KETIDAK SEIMBANGAN FASA PADA GARDU INDUK BLIMBING DENGAN MENGUNAKAN METODE SIMULATED ANNEALLING

(Didik Purnomo, NIM 01.12.128, Teknik Elektro/ T.Energi Listrik)

(Dosen Pembimbing : IR. H. TAUFIK HIDAYAT, MT)

Kata Kunci : Optimalisasi Penyeimbangan Fasa, Penukaran Fasa, Kualitas Power, Simulated Annealing (Sa), Pengesetan Kandidat.

Dalam sistem distribusi, masalah ketidakseimbangan beban merupakan fenomena yang sudah lazim. Ketidakseimbangan feeder / penyulang tidak hanya meningkatkan rugi-rugi daya tetapi juga menyebabkan terjadinya beban lebih dan mempengaruhi kualitas sistem. Ketidakseimbangan feeder menggambarkan sebuah kondisi yang mana tegangan dari sumber tegangan 3 fasa, perbedaan fasanya tidak mencapai 120 derajat. Hal ini mempengaruhi motor-motor dan alat-alat lain yang bergantung pada keseimbangan sumber tegangan tiga fasa.

Keseimbangan fasa ini dimaksudkan untuk menjadikan beban seimbang pada tiap fasa dalam penyuplai. Penukaran fasa merupakan pendekatan langsung untuk menyeimbangkan fasa. Oleh karena itu, pengoptimalan terhadap penyeimbangan fasa saat ini mendapat banyak perhatian dalam bidang industri power. Pengaruh non linear seperti penurunan tegangan dan kerugian-kerugian daya membuat problemnya menjadi sulit untuk dipecahkan. Dengan menggunakan metode Simulated Annealing ini, merupakan metode yang efektif dalam memecahkan problem penyeimbangan fasa distribusi power.

DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN.....	i
ABSTRAKSI.....	ii
KATA PENGANTAR.....	iii
DAFTAR ISI.....	v
DAFTAR TABEL.....	viii
DAFTAR GAMBAR.....	ix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Pembahasan.....	2
1.4 Batasan Masalah.....	2
1.5 Metodologi Penelitian.....	3
1.6 Kontribusi Penelitian.....	3
1.7 Sistematika Pembahasan.....	4
BAB II. Sistem Distribusi Tenaga Listrik dan Teori Dasar Simulated Annealing.....	5
2.1 Sistem Distribusi.....	5
2.1.1 Sistem Distribusi Primer.....	6
2.1.2 Sistem Distribusi Sekunder.....	6
2.2 Struktur Jaringan Distribusi Tenaga Listrik.....	7
2.2.1 Sistem Jaringan Distribusi Radial.....	7
2.2.1.1 Sistem Radial Pohon.....	9
2.2.1.2 Sistem Radial dengan Tipe dan Switch Pemisah... ..	10
2.2.1.3 Sistem Radial dengan Pembagian Phase Area.....	11
2.2.1.4 Sistem Radial dengan Beban Terpusat.....	12
2.2.2 Struktur Jaringan Loop (Tertutup).....	12

2.2.3 Struktur Jaringan Mesh (tertutup)	13
2.3 Daya Sistem Distribusi.....	13
2.3.1 Daya Nyata.....	14
2.3.2 Daya Relatif.....	14
2.3.3 Daya Semu	14
2.3.4 Faktor Daya	15
2.4 Teori Dasar Simulated Annealing (SA)	16
2.4.1 Pengenalan Konsep Simulated Annealing	16
2.4.2 Kombinasi Optimasi.....	16
2.4.3 Analogi pada Pemanasan Fisik (Physical Annealing).....	17
2.4.4 Fungsi Obyektif.....	18
BAB III. Metode Penukaran Fasa Pada Beban Jaringan Tegangan Rendah....	22
3.1 Analisa Aliran Daya	22
3.1.1 Tujuan	22
3.1.2 Pendekatan Studi Aliran Daya	23
3.2 Klasifikasi Bus	23
3.3 Metode Penyelesaian Studi Aliran Daya.....	24
3.3.1 Metode Newton Kaphsan	25
3.3.2 Metode Fast Decoupled.....	34
3.4 Algoritma Fast Decoupled	32
3.5 Flowchart Fast Decoupled.....	34
3.6 Metode Penukaran Fasa Jaringan	36
BAB IV. Metode Penukaran Fasa Jaringan Yang Diaplikasikan Ke Dalam	
Program Komputer.....	38
4.1 Program Komputer Metode Penukaran Fasa Jaringan	38
4.2 Algoritma Program.....	38
4.3 Flow chart Program	40
4.4 Data Jaringan.....	41
4.4.1 Data Saluran	41
4.4.2 Data Pembebanan.....	44
4.5 Analisa Perhitungan	49

4.5.1 Perbedaan Kondisi Pembangkitan, Pembebanan, Rugi – rugi Daya, Efisiensi Daya Sebelum dan Setelah Penukaran Fasa	56
BAB V. Kesimpulan	61
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR TABEL

3.1	Skema Sambungan dari bermacam – macam tipe beban	37
4.1	Data Jenis Konduktor Pada Penyulang Glintung	41
4.2	Data Panjang Saluran Penyulang Glintung	42
4.3	Data Pembebanan Pada Tiap Bus Penyulang Glintung	44
4.4	Data Gardu Dan Persentase Pembebanan Penyulang Glintung Sebelum Penukaran Fasa.....	46
4.5	Data Gardu Dan Persentase Pembebanan Penyulang Glintung Sesudah Penukaran Fasa	47
4.6	Kondisi Awal Rugi – Rugi Daya Saluran Sebelum Penukaran Fasa	50
4.7	Hasil Perhitungan Aliran Daya Sebelum Penukaran Fasa	51
4.8	Hasil Perhitungan Aliran Daya Sesudah Penukaran Fasa	53
4.9	Kondisi Akhir Rugi – Rugi Daya Saluran Setelah Penukaran Fasa.....	53
4.10	Skema Penukaran Fasa Yang Optimum Dengan Metode SA	55
4.11	Perbandingan Rugi – Rugi Daya Saluran.....	59

DAFTAR GAMBAR

2.1 Jaringan Distribusi Tegangan Menengah (JTM), Jaringan Tegangan Rendah (JTR) Dan Sambungan Rumah Ke Pelanggan	5
2.2 Bentuk Sederhana Dari Sistem Distribusi Radial.....	7
2.3 Contoh Sistem Jaringan Distribusi Radial Pohon	9
2.4 Contoh Sistem Jaringan Distribusi Radial dengan Tie Dan Switch Pemisah	10
2.5 Contoh Sistem Jaringan Distribusi Radial dengan Phase Area	11
2.6 Contoh Sistem Jaringan Distribusi Radial dengan Beban Terpusat.....	12
2.7 Segitiga Daya	14
3.1 Sistem Distribusi Radial Sederhana	26
3.2 Flow Chart Fast Decoupled Three Phase	35
3.3 Type Koneksi Antara JTR Dengan Bebannya	37
4.1 Flow Chart Program	40
4.2 Tampilan Perhitungan Rugi – Rugi Daya Sebelum Penukaran Fasa	50
4.3 Tampilan Parameter SA Dan Skema Type Penukaran Fasa	52
4.4 Tampilan Perhitungan Rugi – Rugi Daya Saluran Setelah Penukaran Fasa.....	54
4.5 Tampilan Selisih Rugi – Rugi Saluran Sebelum Dan Setelah Optimasi..	55



BAB I

PENDAHULUAN

1.1 Latar Belakang

Dengan adanya permasalahan dalam industri kelistrikan yakni bertambah besarnya rugi rugi daya saluran distribusi, maka hal ini memaksa industri kelistrikan tersebut untuk memperbaiki kualitas listriknya. Salah satu cara untuk memperbaiki kualitas listriknya adalah dengan menyeimbangkan fasa. Penyeimbangan fasa ini dimaksudkan untuk menjadikan beban seimbang pada tiap fasa dalam penyuplai. Penukaran terhadap fasa merupakan suatu pendekatan langsung untuk menyeimbangkan fasa. Penyeimbangan fasa juga dapat meningkatkan efisiensi daya suatu sistem. Oleh karena itu, pengoptimalan terhadap penyeimbangan fasa saat ini mendapat banyak perhatian dalam bidang industri power, khususnya dibidang deregulasinya. Pengaruh non linier seperti penurunan tegangan dan kerugian-kerugian daya membuat problemnya menjadi sulit untuk dipecahkan. Dengan menggunakan metode *Simulated Annealing* ini adalah merupakan suatu metode yang efektif dalam memecahkan problem penyeimbangan fasa distribusi power.

1.2. Rumusan Masalah

Dalam sistem distribusi power, feeder-feeder yang tidak seimbang itu menimbulkan berbagai dampak negatif yaitu :

- a. Terjadi kelebihan beban
- b. Menambah kerugian-kerugian daya
- c. kualitas power pada sistem menjadi menurun

Berdasarkan rumusan masalah tersebut maka skripsi ini disusun dengan judul :

"Analisis Pengaruh Penukaran Fasa Terhadap Ketidakseimbangan Fasa pada Gardu Induk Blimbing Dengan Menggunakan Metode *Simulated Annealing*."

1.3. Tujuan Pembahasan

Adapun tujuan pembahasan skripsi ini adalah untuk mengatasi masalah ketidakseimbangan beban, yaitu dengan :

1. menyeimbangkan beban pada penyulang
2. Mengurangi rugi rugi daya saluran
3. meningkatkan efisiensi dayanya.

1.4. Batasan Masalah

Agar permasalahan mengarah sesuai dengan tujuan, maka pembahasan skripsi ini dibatasi hal-hal berikut:

- a. Hanya membahas masalah penukaran fasa dalam trafo distribusi yang dihubungkan pada penyulang primer (feeder) tetapi tidak membahas secara detail tentang pemasangan secara nyata penukaran fasanya.
 - b. tidak membahas aliran daya secara mendetail
 - c. beban tiga fasa tidak seimbang
 - d. data diambil dari PT. PLN (persero) distribusi Jatim cabang Malang di gardu Induk Blimbing
-

- e. Metode Penukaran fasa yang digunakan adalah SA (Simulated Annealing)
- f. Gangguan – gangguan jaringan seperti gangguan hubung singkat diabaikan.
- g. Rugi – rugi pada peralatan proteksi dan belitan trafo diabaikan

1.5. Metodologi Penelitian

Metode penelitian yang dipakai dalam penyusunan skripsi ini dilakukan dengan cara sebagai berikut:

a. Studi literatur

Studi ini diperlukan untuk mempelajari teori dan kajian-kajian ilmiah berhubungan dengan pembahasan masalah

b. Pengumpulan data

Bentuk data yang dipergunakan :

- data kuantitatif yaitu data yang dapat dihitung atau data yang berupa angka-angka
- data kualitatif yaitu data yang berbentuk diagram

c. Analisa Data

data-data yang diperoleh dianalisa dengan menggunakan perhitungan metode simulated annealing.

1.6. Kontribusi Penelitian

Dalam skripsi ini diharapkan dapat memberikan informasi mengenai optimalisasi penyelesaian persoalan penukaran fasa pada trafo distribusi yang dihubungkan pada feeder primer untuk menjaga keseimbangan beban fasa sistem di setiap feeder. Sehingga dengan metode SA dapat di jadikan acuan dan bahan

pertimbangan bagi instansi-instansi yang berkepentingan pada pemecahan masalah perencanaan dan operasi sistem tenaga listrik.

1.7. Sistematika Pembahasan

BAB I. Pendahuluan

Pada bab ini dibahas mengenai latar belakang, rumusan masalah, tujuan pembahasan, batasan masalah, metode pembahasan dan sistematika pembahasan serta kontribusi penelitian.

BAB II. Sistem Distribusi Tenaga Listrik dan Teori Dasar SA

Pada bab ini akan dibahas mengenai pengklarifikasian sistem distribusi, struktur jaringan distribusi, daya sistem, sistem per unit, rugi jaringan dan teori dasar SA.

BAB III. Metode Penukaran Fasa

Pada bab ini membahas analisis aliran daya dan penukaran fasa.

BAB IV. Metode Penukaran Fasa Jaringan Yang Diaplikasikan ke Dalam Komputer

Pada bab ini berisi tentang simulasi dan analisis yang terdiri dari prosedur simulasi, hasil simulasi dan analisis.

BAB V. Penutup

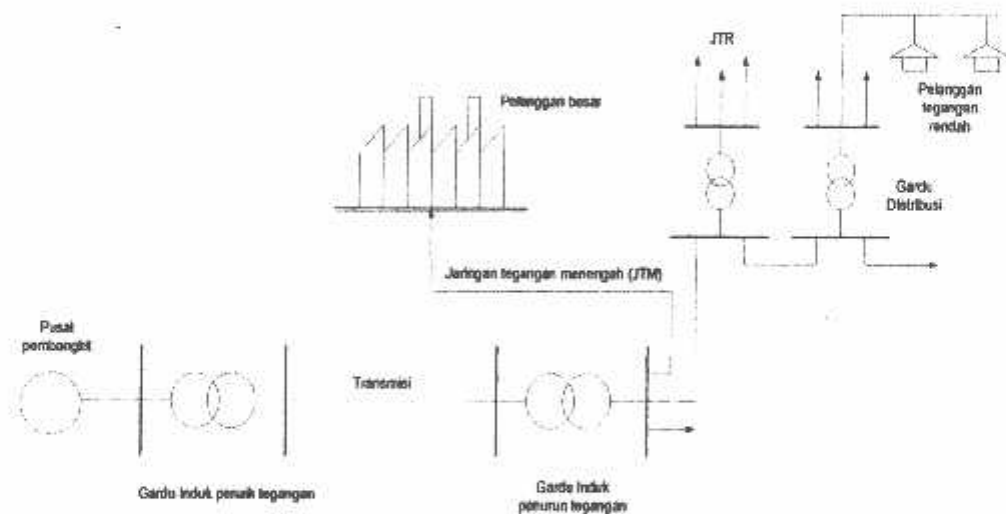
Bab ini berisi kesimpulan



BAB II
SISTEM DISTRIBUSI TENAGA LISTRIK DAN
TEORI DASAR SIMULATED ANNEALING

2.1. Sistem Distribusi^[1]

Jaringan setelah keluar dari G.I. biasa disebut jaringan distribusi. Setelah tenaga listrik disalurkan melalui jaringan distribusi primer kemudian tenaga listrik diturunkan tegangannya dalam gardu-gardu distribusi menjadi tegangan rendah, kemudian disalurkan melalui Jaringan Tegangan Rendah untuk selanjutnya disalurkan ke rumah-rumah pelanggan (konsumen) PLN melalui sambungan rumah seperti pada gambar 2.1.



Gambar 2-1. Jaringan Distribusi Tegangan Menengah (JTM), Jaringan Tegangan Rendah (JTR) dan Sambungan Rumah Ke Pelanggan^[1]

Sumber : Ir. Hasan Basri, 1996, "Sistem Tenaga Listrik", Balai Penerbit Dan Humas ISTN

Dalam pendistribusian tenaga listrik ke konsumen, tegangan yang digunakan bervariasi tergantung dari jenis konsumen yang membutuhkan. Untuk konsumen industri biasanya digunakan tegangan menengah 20 kV sedangkan untuk konsumen perumahan digunakan tegangan rendah 220/380 Volt, yang merupakan tegangan siap pakai untuk peralatan-peralatan rumah tangga. Dengan demikian sistem distribusi tenaga listrik dapat diklasifikasikan menjadi dua bagian sistem yaitu:

1. Sistem distribusi primer.
2. Sistem distribusi sekunder.

Pengklasifikasian sistem distribusi tenaga listrik menjadi dua ini berdasarkan tingkat tegangan distribusinya.

2.1.1. Sistem Distribusi Primer

Tingkat tegangan yang digunakan pada sistem distribusi primer adalah tegangan menengah 20 kV, oleh karena itu sistem distribusi ini sering disebut dengan sistem tegangan menengah.

2.1.2. Sistem Distribusi Sekunder

Tingkat tegangan yang digunakan pada sistem distribusi sekunder adalah tegangan rendah 127/220 Volt atau 220/380 Volt, oleh karena itu sistem distribusi ini sering disebut dengan sistem distribusi tegangan rendah.

Sistem jaringan yang digunakan untuk menyalurkan dan mendistribusikan tenaga listrik tersebut dapat menggunakan sistem satu fasa dengan dua kawat maupun sistem tiga fasa empat kawat.

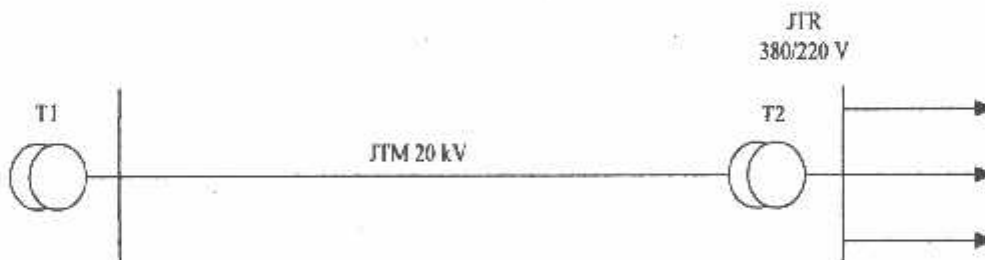
2.2. Struktur Jaringan Distribusi Tenaga Listrik^[1]

Ada beberapa bentuk jaringan yang umum dipergunakan untuk menyalurkan dan mendistribusikan tenaga listrik yaitu :

1. Sistem jaringan distribusi radial
2. Sistem jaringan distribusi rangkaian tertutup (loop)
3. Sistem jaringan distribusi mesh

2.2.1. Sistem Jaringan Distribusi Radial

Bentuk jaringan ini merupakan bentuk dasar yang paling sederhana dan paling banyak digunakan. Sistem ini dikatakan radial karena dari kenyataan bahwa jaringan ini ditarik secara radial dari GI ke pusat-pusat beban/konsumen yang dilayaninya. Sistem ini terdiri dari saluran utama (tunk line) dan saluran cabang (lateral) seperti pada gambar 2.2.



Gambar 2-2. Bentuk Sederhana Dari Sistem Distribusi Radial^[1]

Sumber : Ir. Hasan Basri, 1996, "Sistem Tenaga Listrik", Balai Penerbit Dan Humas ISTN

Pelayanan tenaga listrik untuk suatu daerah beban tertentu dilaksanakan dengan memasang transformator pada sembarang titik pada jaringan yang sedekat mungkin dengan daerah beban yang dilayani. Untuk daerah beban yang

menyimpang jauh dari saluran utama maupun saluran cabang, maka akan ditarik lagi saluran tambahan yang dicabangkan pada saluran tersebut.

Jaringan distribusi radial mempunyai keuntungan dan kelemahan, sebagai berikut;

- Keuntungan :

1. Bentuknya sederhana dibandingkan dengan bentuk yang lain.
2. Biaya investasinya relatif lebih murah, karena saluran menuju ke tiap beban hanya tersedia 1 (satu) jalur.
3. Mudah dikembangkan.

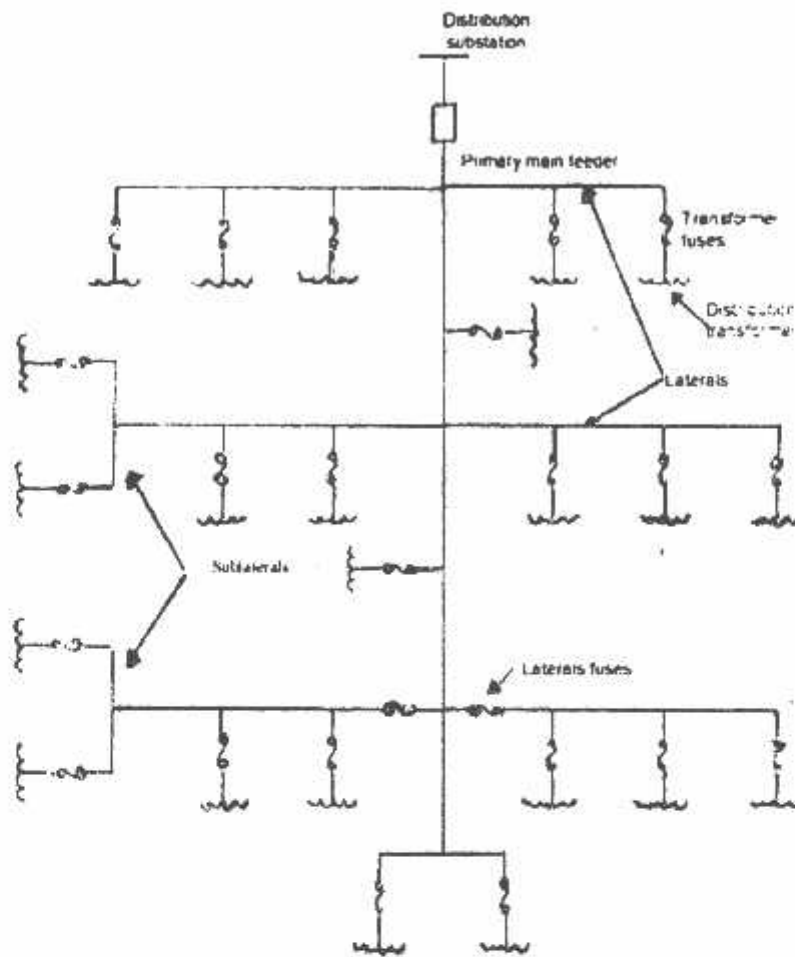
- Kelemahan :

1. Kualitas pelayanan (penyaluran daya) dibandingkan dengan tipe yang lain lebih jelek, sebab jatuh tegangan dan rugi daya relatif besar. Kerugian ini terjadi pada saluran.
2. Kontinuitas penyaluran daya tidak terjamin. Dengan hanya mengandalkan 1 (satu) saluran, maka bila terjadi gangguan pada saluran tersebut tidak dapat diharapkan supply melalui saluran lain.

Jaringan ini dalam perkembangannya mengalami beberapa modifikasi, sehingga dikenal beberapa jaringan distribusi radial, yaitu:

1. Struktur jaringan pohon
 2. Struktur jaringan dengan tie dan switch pemisah
 3. Struktur jaringan dengan pusat beban
 4. Struktur jaringan dengan pembagian daerah phasa (phasa area)
-

2.2.1.1. Sistem Radial Pohon^[2]



Gambar 2.3

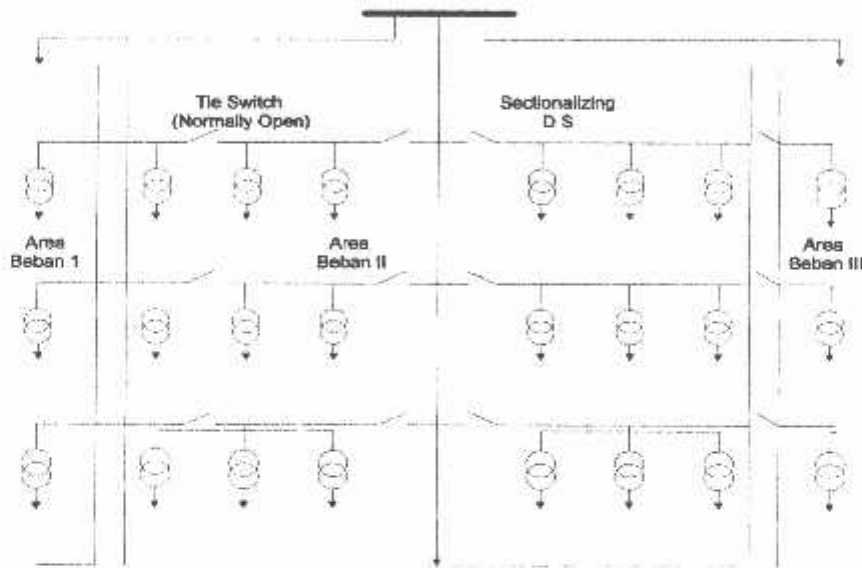
Contoh Sistem Jaringan Distribusi Radial^[2]

Sumber : Turan Gonen, "Electric Power Distribution System Engineering", University of Missouri at Columbia

Sistem jaringan pohon ini merupakan bentuk yang paling dasar dari sistem jaringan radial. Saluran Utama (main feeder) ditarik dari suatu Gardu Induk sesuai dengan kebutuhan kemudian dicabangkan melalui saluran cabang (lateral feeder), selanjutnya dicabangkan lagi melalui saluran anak cabang (sub lateral feeder).

Ukuran dari masing-masing saluran tergantung dari kerapatan arus yang ditanggung. Dari gambar 2.2 main feeder merupakan saluran yang dialiri arus terbesar, selanjutnya arus mengecil pada tiap cabang tergantung dari besarnya beban.

2.2.1.2. Sistem Radial dengan Tie dan Switch Pemisah^[2]



Gambar 2-4. Contoh Sistem Jaringan Distribusi Radial dengan Tie dan Switch Pemisah^[2]

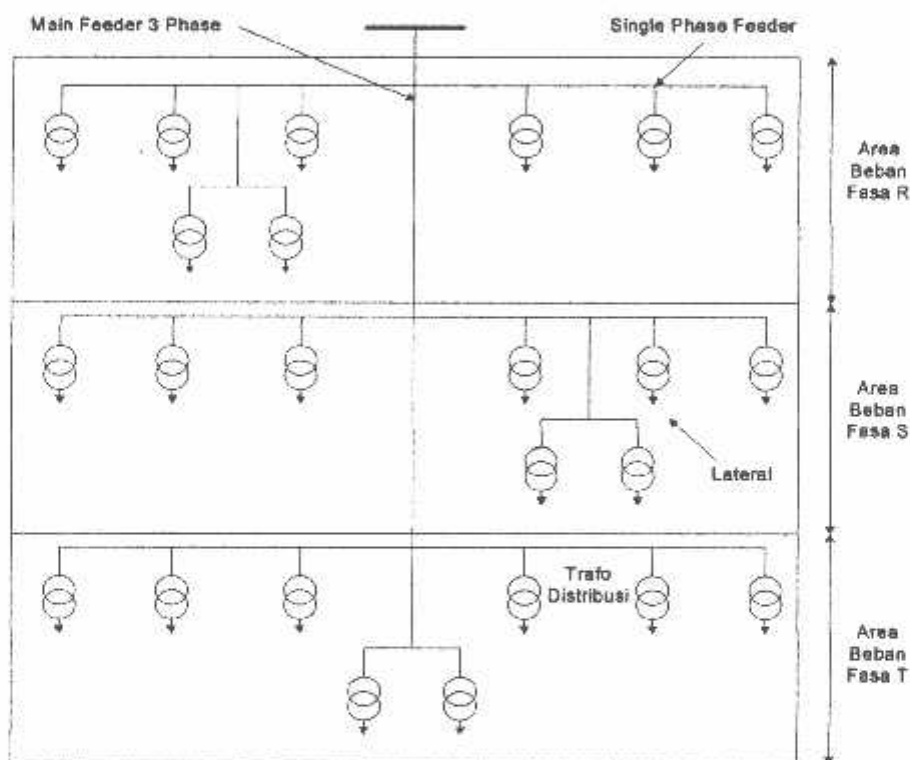
Sumber : Tarun Gonen, "Electric Power Distribution System Engineering", University of Missouri at Columbia

Sistem ini merupakan pengembangan dari sistem radial pohon, untuk meningkatkan keandalan sistem saat terjadinya gangguan maka feeder yang terganggu akan dilokalisir sedangkan area yang semula dilayani feeder tersebut

pelayanannya dialihkan pada feeder yang sehat atau yang tidak terganggu. Sistem radial dengan Tie dan Switch Pemisah dapat dilihat pada Gambar 2.4.

2.2.1.3. Sistem Radial dengan Pembagian Phase Area^[2]

Pada bentuk ini masing-masing fasa dari jaringan bertugas melayani daerah beban yang berlainan. Bentuk ini akan dapat menimbulkan kondisi sistem tiga fasa yang tidak seimbang (simetris), bila digunakan pada daerah beban yang baru dan belum mantap pembagian bebannya. Contoh dari sistem jaringan ini dapat dilihat pada gambar 2.5.

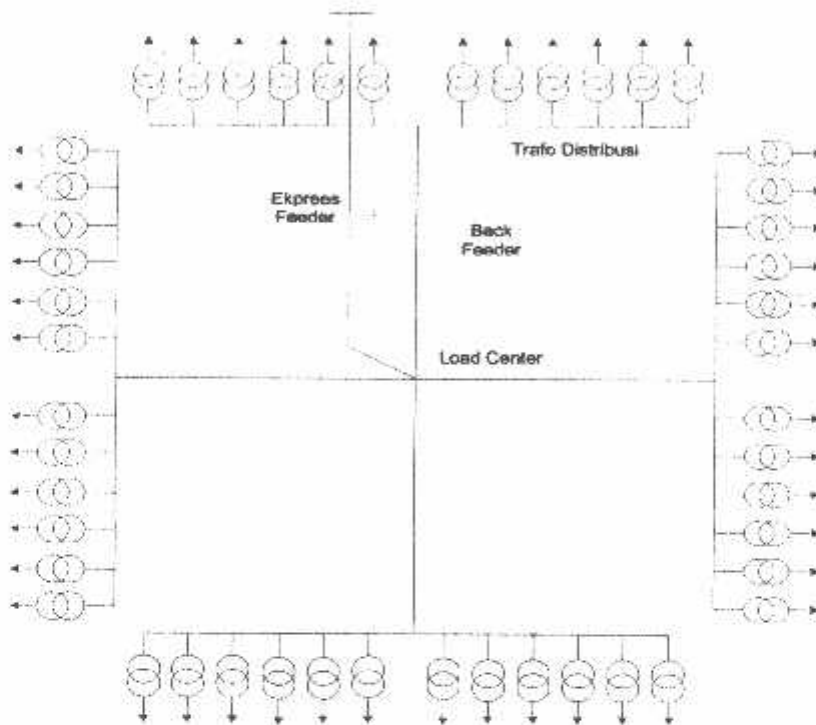


Gambar 2-5. Contoh Sistem Jaringan Distribusi Radial dengan Phase Area^[2]

Sumber : *Turan Gonen, "Electric Power Distribution System Engineering", University of Missouri at Columbia*

2.2.1.4. Sistem Radial dengan Beban Terpusat^[2]

Bentuk dari sistem ini mensuplai daya dengan menggunakan main feeder yang disebut *express feeder* langsung ke pusat beban, dan dari titik pusat beban ini disebut dengan menggunakan *back feeder* secara radial seperti terlihat pada gambar 2.6.



Gambar 2.6.
Contoh Sistem Jaringan Distribusi Radial dengan Beban Terpusat^[2]

Sumber : Terna Goen, "Electric Power Distribution System Engineering", University of Missouri at Columbia

2.2.2. Struktur Jaringan Loop (tertutup)^[1]

Struktur jaringan loop merupakan gabungan dari 2 (dua) sistem jaringan radial, dimana pada ujung kedua jaringan dipasang sebuah pemutus (CB) atau

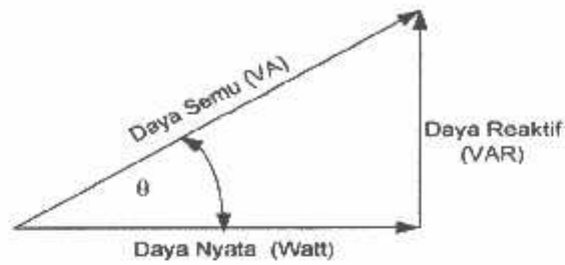
pemisah (DS). Pada saat terjadi gangguan dapat diisolir, maka pemisah atau pemutus ditutup sehingga aliran daya listrik ke bagian yang tidak terkena gangguan tidak berhenti. Dalam kondisi normal, struktur jaringan loop ini merupakan 2 (dua) struktur jaringan radial. Struktur jaringan ini mempunyai keandalan yang cukup, sehingga biaya pembangunannya relatif lebih mahal dibandingkan dengan biaya pembangunan struktur jaringan radial.

2.2.3. Struktur Jaringan Mesh¹¹

Struktur jaringan ini merupakan kombinasi antara sistem jaringan radial dengan sistem jaringan loop. Titik beban memiliki lebih banyak alternatif penyaluran, sehingga bila salah satu penyulang terganggu maka dengan segera dapat digantikan oleh penyulang yang lain. Dengan demikian kontinuitas penyaluran daya sangat terjamin.

2.3. Daya Sistem Distribusi¹¹

Daya merupakan banyaknya perubahan tenaga terhadap waktu dalam besaran tegangan dan arus. Daya dalam Watt yang diserap oleh suatu beban pada setiap saat adalah hasil kali jatuh tegangan sesaat diantara beban dalam Volt dengan arus sesaat yang mengalir dalam beban tersebut dalam Ampere.



Gambar 2-7. Segitiga Daya^[1]

Sumber : Ir. Hasan Basri, 1996, "Sistem Distribusi Tenaga Listrik," Balai Penerbit dan Humas ISTN

2.3.1. Daya Nyata (*Real Power*)^[1]

Daya nyata dinyatakan dalam persamaan :

$$P = |V| |I| \cos \varphi \quad (2.1)$$

Daya nyata untuk beban tiga fasa seimbang

$$P = \sqrt{3} |V_{jala-jala}| |I_{jala-jala}| \cos \varphi \quad (2.2)$$

2.3.2. Daya Reaktif (*Reactive Power*)^[1]

Daya reaktif dinyatakan dalam persamaan :

$$Q = |V| |I| \sin \varphi \quad (2.3)$$

Daya reaktif untuk beban tiga fasa seimbang :

$$Q = \sqrt{3} |V_{jala-jala}| |I_{jala-jala}| \sin \varphi \quad (2.4)$$

2.3.3. Daya Semu^[1]

Daya semu dinyatakan dalam persamaan :

$$S = |V| |I| \quad (2.5)$$

Daya semu untuk beban tiga fasa seimbang :

$$S = \sqrt{3} |V_{jala-jala}| |I_{jala-jala}| \quad (2.6)$$

Persamaan di atas berlaku tanpa memandang apakah beban dihubungkan secara segitiga atau bintang.

2.3.4. Faktor Daya^[1]

Faktor daya adalah perbandingan antara komponen daya KW terhadap KVA, seperti yang terlihat pada gambar 2.8.

Faktor daya :

$$\cos \varphi = \frac{kW}{kVA} \quad (2.7)$$

$$(kVA)^2 = (kW)^2 + (kVAR)^2 \quad (2.8)$$

$$kW = kVA \cos \varphi \quad (2.9)$$

$$kVAR = kVA \sin \varphi = kW \cdot \tan \varphi \quad (2.10)$$

$$\tan \varphi = \frac{kVAR}{kW} \quad (2.11)$$

Faktor daya akan mendahului (leading) apabila arus mendahului tegangan dan akan tertinggal (lagging) bila arus terbelakang terhadap tegangan

2.4 Teori Dasar Simulated Annealing (SA)^{[3][4]}

2.4.1 Pengenalan Konsep Simulated Annealing

Konsep Simulated Annealing atau SA pertama kali diperkenalkan pada dunia optimasi pada permulaan tahun 1980 oleh Kirkpatrick dan secara bebas dikembangkan oleh Cerny. SA adalah sebuah kombinasi algoritma optimasi berdasarkan metode probabilistic atau kemungkinan yang telah diterapkan dengan baik pada banyak area. Seperti pada jaringan kerja system saraf, dan penempatan kapasitor pada system daya.

2.4.2 Kombinasi Optimasi

Masalah kombinasi optimasi adalah sebuah masalah minimisasi atau maksimisasi yang terkait dengan penemuan solusi terbaik atau optimal dari kumpulan alternative kemungkinan. Penemuan solusi dapat digolongkan secara menyeluruh oleh ruang penelitian dan fungsi harga atau fungsi obyektif.

Ruang Penelitian S adalah sebuah kumpulan solusi kemungkinan terbatas atau tak terbatas yang dapat dihitung. Dan fungsi obyektif $f(S)$ mengelompokkan tiap titik pada ruang penelitian kedalam garis sesungguhnya, untuk memberikan pengukuran terhadap bagaimana sebuah solusi yang baik itu berhubungan dengan yang lainnya. Pada masalah minimisasi, solusi optimal yang diharapkan adalah x_{opt} , untuk :

$$f(x_{opt}) \leq f(x), \text{ untuk semua } x \in S \quad (2.7)$$

Masalah tersebut kemudian dapat dikelompokkan dengan mudah sebagai :

$$\text{Minimize } f(x)$$

$$X \in S \quad (2.8)$$

Solusi x_{opt} disebut sebagai global optimum dan nilai obyektif x_{opt} harga optimal disebut $f_{opt} = f(x_{opt})$. Kumpulan dari solusi optimal secara menyeluruh dikenal sebagai S_{opt} .

2.4.3 Analogi Pada Pemanasan Fisik atau (*Physical Annealing*)

Nama pemansan fisik berasal dari sebuah analogi antara pengkombinasian dan proses fisik pemanasan. Pada pemanasan fisik, zat padat didinginkan dengan sangat perlahan mulai dari temperature yang sangat tinggi sampai temperature yang sangat rendah agar dapat mencapai uraian energi internal minimum. Zat padat ini didinginkan secara perlahan supaya keseimbangan panas dicapai pada tiap temperature. Keseimbangan panas dapat digolongkan oleh distribusi

$$\text{Boltzman : } P_T = \{X = x\} = \frac{e^{-E_x / k_B T}}{\sum_{\text{all states } i} e^{-E_i / k_B T}} \quad (2.9)$$

Dimana X adalah variable acak yang menunjukkan posisi baru, E_x adalah energi dari posisi x, K_B adalah konstanta *Boltzman* dan T adalah temperature.

Evolusi dari posisi zat padat dalam tanur tinggi menuju keseimbangan panas dapat ditiru secara efisien oleh algoritma sederhana berdasarkan teknik *Monte Carlo* yang diusulkan oleh Metropolis tahun 1953. Pada algoritma metropolis mengambil posisi baru dan menggambarkan posisi tersebut yaitu x. Kemudian transisi dari posisi y menuju posisi x diterima dengan kemungkinan :

$$P_{accept}(x, y) = \begin{cases} 1 & , \text{jika } E_x - E_y \leq 0 \\ e^{-(E_x - E_y) / k_B T} & , \text{jika } E_x - E_y > 0 \end{cases} \quad (2.10)$$

Jika diterima, x menjadi solusi baru dan prosedurnya diulang. Aturan penerimaan ini dikenal dengan *Metropolis Criterion*.

2.4.4 Fungsi obyektif⁽⁴⁾

Untuk menyelidiki pengaruh pada penurunan tegangan atau rugi – rugi tegangan maka sebuah model aliran beban AC penuh harus dipertimbangkan dalam merumuskan penukaran fasa. SA mempunyai keunggulan dalam memecahkan masalah – masalah fungsi obyektif non linier. Untuk masalah – masalah penukaran fasa, maka fungsi nilainya pada setiap solusi kandidat dengan mudah dapat dievaluasi. Disini ditetapkan beberapa rumusan baru sebagai berikut, yaitu :

$$H_1 = \sum_i \xi(T_i^k - T_i^0) \quad (2.11)$$

dimana $\xi(T_i^k - T_i^0) = 0$, jika $(T_i^k = T_i^0)$

$$1, \text{ jika } (T_i^k \neq T_i^0) \quad (2.12)$$

Nilai penalty dari aliran-aliran beban:

$$H_2 = \sum_j w_j \cdot \sum_{ph} g\left(\frac{f_j, ph}{c_j}\right) \quad (2.13)$$

$$\text{dimana } g(x) = \begin{cases} 0 & , \text{ jika } x < 0,3 \\ e^{x-1} - e^{-0,7} & , \text{ jika } 0,3 \leq x < 1,0 \\ e^{2(x-1)} - e^{-0,7} & , \text{ jika } x \geq 1,0 \end{cases} \quad (2.14)$$

Index penyeimbangan system :

$$H = \delta.H_1 + \tau.H_2 \quad (2.15)$$

dimana:

$f_{j,ph}$ adalah aliran fasa pada cabang j , $ph = a, b, c$.

T_i^0 adalah skema pencabangan yang asli pada node i

T_i^k adalah skema pencabangan pada node i dalam kandidat hasil

C_j adalah kapasitas pada segmen jalur j

w_j adalah faktor pemberatan pada satu segmen j

δ adalah nilai pada tiap – tiap node penukaran fasa

τ adalah koefisien pada nilai penalty yang tak seimbang

H adalah indeks penyeimbangan sistem

Jumlah dari penukaran fasa, $H1$ adalah suatu indicator untuk nilai penukaran fasa. Fungsi penalty $H2$ adalah indicator untuk kontra benefit dari penyeimbangan fasa yaitu, semakin kecil penaltynya maka semakin seimbang penyuplainya. Hal ini mencakupi beberapa pertimbangan yang khusus, misalnya tidak adanya penalty pada penyuplai beban ringan. Penalty akan semakin banyak apabila arusnya mendekati kapasitasnya, khususnya pada keadaan beban lehih. Faktor pemberatan, ditentukan berdasarkan pada pentingnya tiap – tiap cabang indeks keseimbangan system. Untuk fungsi nilai penalty yang ditetapkan pada persamaan (2.13) jika aliran bebannya jauh dibawah 30 % dari kapasitasnya maka nilai penaltynya adalah nol. Tidak peduli walau ia belum seimbang. Nilai penalty akan cepat bertambah selama aliran bebannya bertambah. Oleh karena itu penyeimbangan fasanya lebih memfokuskan pada cabang – cabang yang berbeban berat dibandingkan cabang – cabang yang berbeban ringan. Ketika nilai penukaran fasa dan fungsi nilai penalty digunakan maka problem penukaran fasa

menjadi problem pemrograman bilangan yang non linier. Fungsi obyektifnya adalah sebuah fungsi non linier yang sulit dipecahkan oleh cara – cara yang lazim. Namun, SA sangat mudah menangani fungsi obyektif yang non linier. Karena kandidat pada penukaran fasa sangat mudah dihasilkan dan fungsi obyektifnya sangat mudah dievaluasi, maka SA sangat baik dalam memecahkan problem – problem penukaran fasa yang berskala besar.

SA yang digunakan untuk problem penukaran fasa memerlukan empat komponen dasar yaitu :

- **Konfigurasi** : Suatu model dimana suatu penukaran fasa yang legal adalah satu dan hanya satu beban yang ditentukan dalam lintasan fasa suatu node.
 - **Pergerakan** : Pada masing – masing node tak seimbang yang diperbolehkan paling banyak enam percabangan fasa. Masing – masing adalah suatu gerakan yang valid.
 - **Fungsi Nilai** : Fungsi nilai dapat menjadi aliran beban yang tak seimbang, indeks penyeimbangan dan keseluruhan fungsi nilai yang ditetapkan dalam persamaan (2.15). Dalam pembahasan berikut ini seluruh fungsi nilai akan dikurangi sehingga penyuplai menjadi seimbang.
 - **Jadwal Pendinginan**: Merupakan kunci dalam memastikan agar SA menjadi konvergen kepada solusi optimal. Ringkasnya jadwal pendinginan ditentukan dalam empat macam yaitu :
-

Temperatur awal, Temperatur final, Jumlah iterasi, dan Jumlah pendinginan.

Temperatur Awal ($T_{initial}$) : temperature awal ditentukan sedemikian rupa sehingga secara praktek semua penukaran fasanya dapat dibuat.

Temperatur Final (T_{final}) : temperature final ditentukan sedemikian rupa.

Jumlah iterasi (N) : Jumlah itersi ditentukan oleh waktu K dalam jumlah pilihan yang ada. Jumlah pilihan yang ada tergantung pada jumlah node didalam penyuplai dan jumlah fase pada tiap node.

Nilai Pendinginan ($r_{cooling}$) : Nilai pendinginan temperature berkisar antara 0,50 sampai 0,99. Hal ini merupakan fungsi dari temperature awal, temperature final, dan jumlah iterasi. Karena hanya satu kandidat yang digunakan pada tiap level temperature, maka nilai pendinginan ditentukan oleh :

$$R_{cooling} = \sqrt[N]{T_{final} / T_{initial}} \quad (2.16)$$

Misalnya : temperature awalnya = 450, temperature final = 0,1 dan jumlah iterasinya = 300, maka nilai pendinginannya = 0,9723

BAB III
METODE PENUKARAN FASA
PADA BEBAN JARINGAN TEGANGAN RENDAH

Dalam analisa metode penukaran fasa diperlukan suatu proses aliran daya untuk mengetahui tegangan pada bus beban dan rugi-rugi daya pada saluran. Oleh karena itu pembahasan selanjutnya sebelum membahas proses metode penukaran fasa adalah analisa aliran daya menggunakan Metode *Fast Decoupled*.

3.1. Analisa Aliran Daya^[5]

3.1.1. Tujuan

Dengan semakin kompleksnya problem di dalam sistem tenaga listrik, sebagai akibat dari meningkatnya permintaan konsumen, bertambahnya jumlah saluran transmisi dan distribusi, maka perlu adanya studi aliran daya dalam analisa sistem. Mempelajari aliran daya ini dilakukan untuk menentukan:

- Aliran daya aktif dan daya reaktif pada cabang-cabang rangkaian.
- Tidak adanya rangkaian yang mempunyai beban lebih dari tegangan busbar dalam batas-batas yang diterima.
- Pengaruh penambahan atau perubahan pada suatu sistem.
- Kondisi optimum pembebanan sistem.
- Kehilangan daya optimum sistem.

3.1.2. Pendekatan Studi Aliran Daya^[5]

Di dalam pengoperasian sistem tenaga listrik, parameter-parameter listrik yang perlu diperhatikan sehubungan dengan analisa aliran daya adalah besarnya magnitude tegangan $|V|$, sudut fasa tegangan θ , daya nyata P dan daya reaktif Q. Daya nyata P mempunyai ketergantungan yang kuat dengan sudut fasa tegangan θ dan daya reaktif Q mempunyai ketergantungan yang kuat dengan besarnya magnitude tegangan $|V|$. Bila P dan Q berubah, maka θ dan $|V|$ berubah pula demikian sebaliknya.

Di dalam analisa aliran daya terdapat 3 jenis variabel:

1. Variabel bebas, misalnya $|V|$ dan θ pada bus beban atau θ dan Q pada bus generator.
2. Variabel tidak bebas, misalnya P dan $|V|$ pada bus generator.
3. Variabel yang tidak dapat diatur, misalnya kebutuhan konsumen.

3.2. Klasifikasi Bus^[5]

Tujuan aliran daya pada sistem tenaga listrik adalah untuk menghitung besar (magnitude) tegangan $|V|$ dan beda fasa θ tegangan pada semua bus, sehingga dengan diketahuinya parameter-parameter tersebut akan dapat dihitung besar daya yang mengalir beserta rugi-ruginya. Pada setiap bus dari jaringan terdapat parameter sebagai berikut:

1. Daya nyata, dinyatakan dengan P satuan MW.
 2. Daya reaktif, dinyatakan dengan Q satuannya MVAR.
 3. Besar (magnitude) tegangan, dinyatakan dengan $|V|$ satuannya kV.
-

4. Beda fasa tegangan, dinyatakan dengan θ satuannya derajat.

1 dan 2 menyatakan daya yang dibangkitkan generator yang mengalir ke bus. Jika pada bus terdapat beban, maka daya tersebut menyatakan selisih antara daya yang dibangkitkan dengan daya beban.

Dalam aliran daya, pada setiap busnya perlu diketahui 2 parameter dari 4 parameter yang diperhitungkan. Dengan melihat kedua parameter tersebut dapat diketahui jenis busnya, yaitu :

1. Bus beban

Pada bus ini daya P dan daya Q diketahui, sementara magnitudo tegangan $|V|$ dan beda fasa θ dicari (dihitung).

2. Bus generator (rel tegangan konstan)

Pada bus ini daya nyata P dan magnitudo tegangan $|V|$ diketahui, sementara daya reaktif Q dan beda fasa θ dicari.

3. Bus berayun

Pada bus ini magnitudo tegangan $|V|$ dan beda fasa θ diketahui besarnya, sementara daya nyata P dan daya reaktif Q dicari. Sedangkan beda fasa θ tegangan berharga nol, karena bus ini fasor tegangan dari bus dipakai sebagai referensi.

3.3. Metode Penyelesaian Studi Aliran Daya^[5]

Dalam studi aliran daya, beberapa teknik yang umum digunakan untuk menyelesaikan persoalan-persoalan aliran daya tiga fasa adalah:

1. Metode *Newton Raphson*

2. Metode *Fast Decoupled*

Metode yang akan ditinjau disini adalah metode *Fast Decoupled Three Phase* yang penurunan matrik *Jacobianya* diturunkan dari metode *Newton Raphson*. Dengan asumsi dan pendekatan yang digunakan tersebut diharapkan metode *Fast Decoupled Three Phase* lebih efisien didalam perhitungan perbaikan (*up-date*).

3.3.1. Metode *Newton Raphson*^[5]

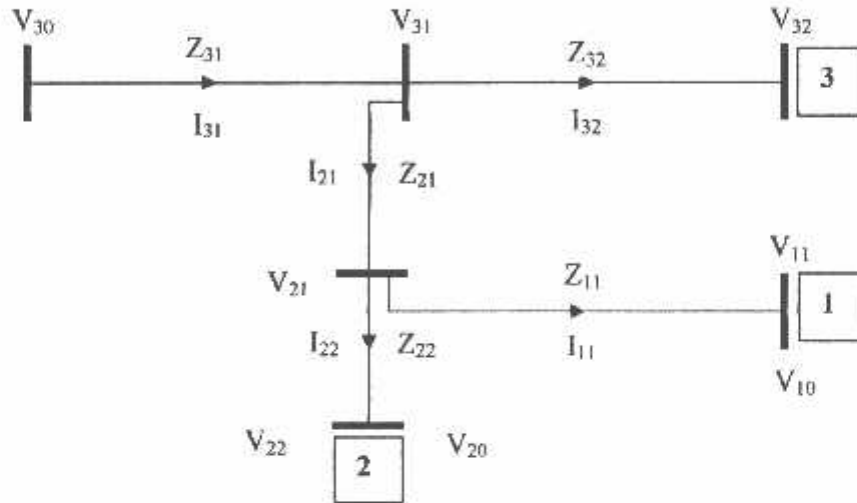
Persamaan aliran daya dapat diselesaikan dengan metode iterasi untuk mencari nilai nol dari persamaan non-linier. Prosedur metode *Newton Raphson* yang sudah dikenal luas untuk pemecahan non-linier adalah sebagai berikut:

1. Memilih perkiraan awal untuk x_0 .
2. Menetapkan $k = 0$
3. Menyelesaikan $F_k \approx F(x_k)$.
4. Jika $|F_k| \leq \text{toleransi}$, maka proses dihentikan
5. Perhitungan Jacobian, $J_k = \left. \frac{\partial F}{\partial x} \right|_{x_k}$
6. Menyelesaikan $J_k S_k = -F_k$.
7. $x_{k+1} = x_k + S_k$
8. $k = k + 1$, kembali ke langkah 3.

Sebagai sebuah ciri khas dalam metode *Newton Raphson*, mayoritas waktu perhitungan berada dalam langkah 5 yaitu pada perhitungan nilai-nilai *Jacobian* dan langkah 6 yaitu menyelesaikan langkah-langkah perbaikan. Dengan adanya pendekatan logis yang menghasilkan perhitungan yang lebih sedikit dalam kedua

langkah tersebut akan sangat mempengaruhi kecepatan perhitungan dengan hanya menambah jumlah iterasinya

Untuk lebih memudahkan pemahaman terhadap pendekatan-pendekatan yang dilakukan pada metode *Fast Decoupled* 3 fasa sistem distribusi radial yang dibahas disini maka akan diberikan suatu contoh sistem distribusi radial 3 buah lateral:



Gambar 3.1
Sistem Distribusi Radial Sederhana

Sistem distribusi radial di atas terdiri dari 3 (dga) buah lateral sehingga didapatkan 3 (tiga) buah persamaan aliaran daya:

$$F_{1(x)} = \bar{V}_{10} - \bar{V}_{21} = 0 \dots\dots\dots (3.1)$$

$$F_{2(x)} = \bar{V}_{20} - \bar{V}_{31} = 0 \dots\dots\dots (3.2)$$

$$F_{3(x)} = \bar{V}_{30} - \bar{V}_0 = 0 \dots\dots\dots (3.3)$$

Dimana:

\bar{V}_{10} = tegangan ujung awal dari cabang ke-1

\hat{V}_{20} = tegangan ujung awal dari cabang ke-2

\hat{V}_{30} = tegangan ujung awal dari cabang ke-3

\hat{V}_0 = tegangan source bus cabang ke-3 yang nilainya telah ditentukan

Sehingga besarnya nilai-nilai tegangan tersebut adalah :

$$\hat{V}_{10} = \hat{V}_{11} + Z_{11} (\frac{1}{2} Y_{11} \hat{V}_{11} + I_{11})$$

$$I_{11} = - \left[\frac{S_{L1}}{\hat{V}_{11}} \right]$$

$$\hat{V}_{21} = \hat{V}_{22} + Z_{22} (\frac{1}{2} Y_{22} \hat{V}_{22} + I_{22})$$

$$I_{22} = - \left[\frac{S_{L2}}{\hat{V}_{22}} \right]$$

$$\hat{V}_{20} = \hat{V}_{21} + Z_{21} (\frac{1}{2} Y_{21} \hat{V}_{21} + I_{21})$$

$$\begin{aligned} &= \hat{V}_2 + Z_{22} (\frac{1}{2} Y_{22} \hat{V}_{22} - \left[\frac{S_{L2}}{\hat{V}_{22}} \right]^*) + \frac{1}{2} Y_{21} Z_{21} (\hat{V}_{22} - Z_{22} (\frac{1}{2} Y_{22} \hat{V}_{22} - \left[\frac{S_{L2}}{\hat{V}_{22}} \right]^*)) \\ &\quad - Z_{21} \left[\frac{S_{L1}}{\hat{V}_{11}} \right]^* - Z_{21} \left[\frac{S_{L2}}{\hat{V}_{22}} \right]^* \end{aligned}$$

$$\hat{V}_{31} = \hat{V}_{32} + Z_{32} (\frac{1}{2} Y_{32} \hat{V}_{32} + I_{32})$$

$$I_{32} = - \left[\frac{S_{L3}}{\hat{V}_{32}} \right]^*$$

$$\hat{V}_{30} = \hat{V}_{31} + Z_{31} (\frac{1}{2} Y_{31} \hat{V}_{31} + I_{31})$$

$$\begin{aligned} \hat{V}_{30} &= \hat{V}_{32} + Z_{32} (\frac{1}{2} Y_{32} \hat{V}_{32} + \left[\frac{S_{L3}}{\hat{V}_{32}} \right]^*) + \frac{1}{2} Y_{31} Z_{31} (\frac{1}{2} Y_{32} \hat{V}_{32} + \left[\frac{S_{L3}}{\hat{V}_{32}} \right]^*) \\ &\quad + Z_{31} \left[\frac{S_{L3}}{\hat{V}_{32}} \right]^* + Z_{31} \left[\frac{S_{L2}}{\hat{V}_{22}} \right]^* + Z_{31} \left[\frac{S_{L1}}{\hat{V}_{11}} \right]^* \end{aligned}$$

Sistem *Jacobian* dari sistem pada gambar 3.1 di atas adalah :

$$J_k = \begin{bmatrix} \frac{\partial F_1}{\partial V_{11}} & \frac{\partial F_1}{\partial V_{22}} & \frac{\partial F_1}{\partial V_{32}} \\ \frac{\partial F_2}{\partial V_{11}} & \frac{\partial F_2}{\partial V_{22}} & \frac{\partial F_2}{\partial V_{32}} \\ \frac{\partial F_3}{\partial V_{11}} & \frac{\partial F_3}{\partial V_{22}} & \frac{\partial F_3}{\partial V_{32}} \end{bmatrix} \dots\dots\dots (3.4)$$

Dimana :

$$\frac{\partial F_1}{\partial V_{11}} = 1 + \frac{1}{2} Y_{11} V_{11} + Z_{11} \frac{S_{L1}^*}{|V_{11}|^2}$$

$$\frac{\partial F_1}{\partial V_{22}} = -(1 + \frac{1}{2} Y_{22} V_{22} + Z_{22} \frac{S_{L2}^*}{|V_{22}|^2})$$

$$\frac{\partial F_1}{\partial V_{32}} = 0$$

$$\frac{\partial F_2}{\partial V_{11}} = Z_{21} \frac{S_{L1}^*}{|V_{11}|^2}$$

$$\begin{aligned} \frac{\partial F_2}{\partial V_{22}} &= 1 + \frac{1}{2} Y_{22} V_{22} + Z_{22} \frac{S_{L2}^*}{|V_{22}|^2} + \frac{1}{2} Y_{21}^* V_{21} + \frac{1}{4} Y_{21} Y_{21} Z_{21} Z_{22} \\ &\quad + \frac{1}{2} Y_{21} Z_{21} Z_{22} \frac{S_{L2}^*}{|V_{22}|^2} \end{aligned}$$

$$\frac{\partial F_2}{\partial V_{32}} = -(1 + \frac{1}{2} Y_{32} V_{32} + Z_{32} \frac{S_{L2}^*}{|V_{32}|^2})$$

$$\frac{\partial F_3}{\partial V_{11}} = Z_{31} \frac{S_{L1}^*}{|V_{11}|^2}$$

$$\frac{\partial F_3}{\partial V_{22}} = Z_{31} \frac{S_{L2}^*}{|V_{22}|^2}$$

$$\begin{aligned} \frac{\partial F_3}{\partial V_{32}} = & 1 + \frac{1}{2} Y_{32} V_{32} + Z_{32} \frac{S_{L3}^*}{|V_{32}|^2} + \frac{1}{2} Y_{31} V_{31} + \frac{1}{4} Y_{31} Y_{31} Z_{31} Z_{32} \\ & + \frac{1}{2} Y_{31} Z_{31} Z_{32} \frac{S_{L3}^*}{|V_{32}|^2} \end{aligned}$$

3.3.2. Metode *Fast Decoupled*⁴⁵¹

Pada sistem distribusi nilai *line charging* diabaikan karena selain saluran distribusi pendek-pendek juga dalam perhitungan *pu* nilai *line charging* (Y) adalah kecil dibandingkan nilai tegangan (1 pu), sehingga komponen-komponen yang mengandung perkalian variabel Y pada turunan parsial diatas akan bernilai nol, sehingga selanjutnya *Jacobian* sistem gambar 3.1 diatas menjadi :

$$J_k = \begin{bmatrix} \left(1 + Z_{11} \frac{S_{L1}^*}{|V_{11}|^2}\right) - \left(1 + Z_{22} \frac{S_{L2}^*}{|V_{22}|^2}\right) & 0 \\ Z_{21} \frac{S_{L1}^*}{|V_{22}|^2} & \left(1 + Z_{22} \frac{S_{L2}^*}{|V_{22}|^2}\right) - \left(1 + Z_{32} \frac{S_{L3}^*}{|V_{32}|^2}\right) \\ Z_{31} \frac{S_{L1}^*}{|V_{11}|^2} & Z_{31} \frac{S_{L2}^*}{|V_{22}|^2} & \left(1 + Z_{32} \frac{S_{L3}^*}{|V_{32}|^2}\right) \end{bmatrix} \dots(3.5)$$

Karena blok-blok yang tidak nol yang berada dibawah diagonal memiliki $\frac{S_{Li}^*}{|V_i|^2}$ yang bernilai kecil serta dikalikan dengan $Z_i (\leq 10^{-4})$ yang nilainya kecil pula maka blok-blok ini akan bernilai mendekati nol, sehingga dapat kita abaikan. Dengan melakukan ini dapat dikatakan bahwa perbedaan tegangan untuk cabang ke-i hanya diakibatkan oleh tegangan ujung akhir dari cabang ke-i, sedangkan pengaruh tegangan dari ujung cabang-cabang yang lain diabaikan. Maka *Jacobiannya* sekarang menjadi:

$$J_k = \begin{bmatrix} \left(1 + Z_{11} \frac{S_{L1}^*}{|V_{11}|^2}\right) - \left(1 + Z_{22} \frac{S_{L2}^*}{|V_{22}|^2}\right) & 0 & 0 \\ 0 & \left(1 + Z_{22} \frac{S_{L2}^*}{|V_{22}|^2}\right) - \left(1 + Z_{32} \frac{S_{L3}^*}{|V_{32}|^2}\right) & 0 \\ 0 & 0 & \left(1 + Z_{32} \frac{S_{L3}^*}{|V_{32}|^2}\right) \end{bmatrix} \dots(3.6)$$

Pendekatan ini akan mempercepat proses langkah ke-6 dari metode Newton diatas dan matrik *Jacobian* sekarang menjadi lebih banyak kosongnya, sehingga langkah perbaikan (update step) pada langkah ke-6 tersebut dapat diselesaikan dengan mudah.

Pada algoritma decoupled evaluasi terhadap *Jacobian* pada langkah 5 dari metode Newton masih terlalu sulit. Oleh karena itu pendekatan lebih lanjut dengan memperhatikan nilai $Z_1 \frac{S_{L1}^*}{|V_1|^2}$ yang kecil sehingga blok yang nilainya mendekati 1 dapat diganti dengan nilai pasti yaitu "1" dan blok yang bernilai mendekati -1 diganti dengan nilai "1". Sehingga *Jacobiamnya* menjadi:

$$J_k = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \dots\dots\dots (3.7)$$

Dengan melakukan pendekatan ini maka *Jacobian* sekarang akan menjadi matrik konstan sehingga langkah ke-5 dari metode *Newton Raphson* telah dapat dihilangkan dan perhitungan pada langkah ke-6 akan menjadi sederhana. Sehingga kita tidak perlu lagi menyimpan *Jacobian* ini selama perhitungan. Maka dari itu kita dapat menyimpulkan nilai "1" terletak pada diagonal dan "-1" pada

kolom yang bersesuaian yang menunjukkan hubungan antara anak dan induk dalam suatu percabangan.

Maka langkah kc-6 dari metode *Newton Raphson* dapat disederhanakan menjadi substitusi terbalik:

$$J_{kx} = -F_k$$

$$\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} S_{11} \\ S_{22} \\ S_{32} \end{bmatrix} = - \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} \dots\dots\dots (3.8)$$

$$S_{32} = -F_2$$

$$S_{22} = -F_2 + S_{32}$$

$$S_{11} = -F_1 + S_{22}$$

Dari pembahasan diatas dapat disimpulkan bahwa metode *Fast Decoupled* merupakan suatu pendekatan atau penyederhanaan dari metode *Newton Raphson*, karena fakta dalam saluran distribusi bahwa elemen impedansi saluran Z_k dan admitansi *line charging* Y_k sangat kecil bila dibandingkan dengan nilai tegangan. cabang *Jacobian* dapat disederhanakan menjadi matrik identitas sehingga langkah perbaikan (*up date step*) dapat diselesaikan dengan cara sederhana yaitu substitusi terbalik.

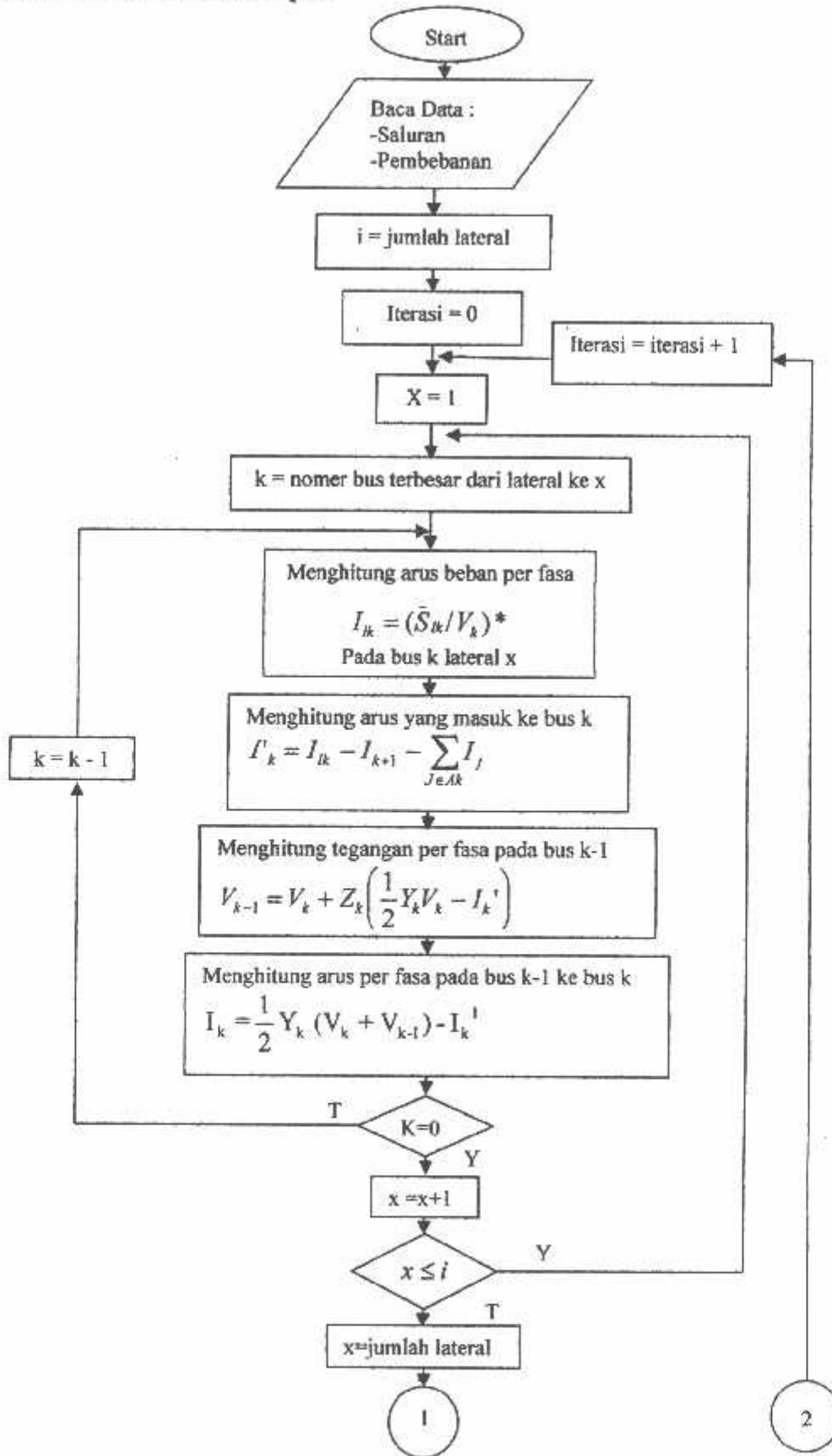
3.4. Algoritma

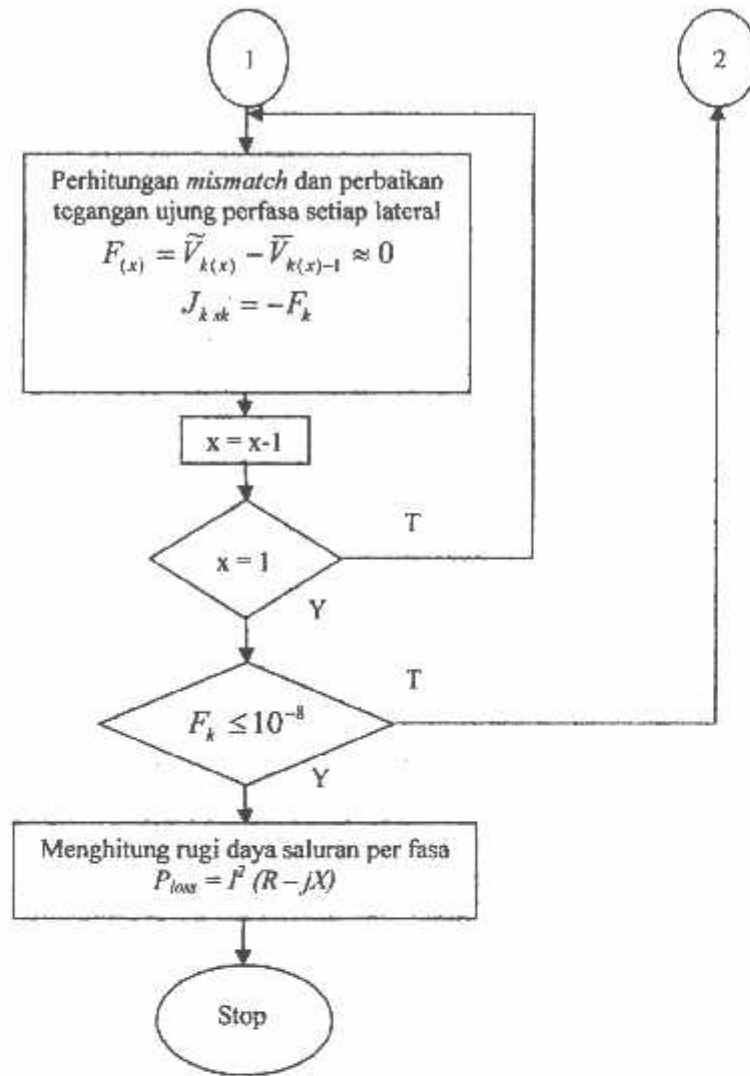
Analisa aliran daya dengan metode fast Decoupled Three Phase, mengikuti algoritma sebagai berikut:

1. Membaca data-data saluran seperti data impedansi dan admitansi line charging saluran, data pembebanan, tegangan sumber, tegangan asumsi awal ujung saluran, arus pada ujung lateral, data hubungan antar bus dan antar lateral.
 2. Perhitungan dimulai dari bus menurut urutan pembalikan pertama muncul (*reverse-breadth first*) yaitu dari bus terakhir pada lateral pertama.
 3. Menghitung arus injeksi per-fasa (I_{ja} , I_{jb} , I_{jk}) pada bus dari suatu cabang.
 4. Menghitung tegangan per-fasa (V_a , V_b , V_c) dari bus sebelumnya pada suatu cabang.
 5. Menghitung arus saluran per-fasa (I_a , I_b , I_c) dari suatu bus ke bus sebelumnya pada suatu cabang.
 6. Kembali ke langkah 3 untuk bus berikutnya sampai perhitungan dilakukan pada bus yang merupakan ujung awal dari suatu lateral.
 7. Kembali ke langkah 3 untuk lateral berikutnya sampai perhitungan dilakukan pada lateral utama.
 8. Menghitung perbedaan tegangan (*voltage mismatch*) pada setiap cabang dimulai dari lateral utama.
 9. Menghitung nilai perbaikan (S_k) dan mengoreksi nilai tegangan pada setiap akhir lateral.
 10. Jika perbedaan tegangan sudah memenuhi batas yang diijinkan/ toleransi maka perhitungan kembali ke langkah 3.
-

11. Jika perbedaan tegangan sudah memenuhi batas yang diijinkan maka tegangan dianggap konvergen dan menjadi output program.
 12. Menghitung rugi-rugi daya saluran per fasa dari total perulangan dan mencetak nilai-nilainya.
-

3.5. Flowchart Fast Decoupled





Gambar 3.2.
Flowchart Fast Decoupled Three Phase

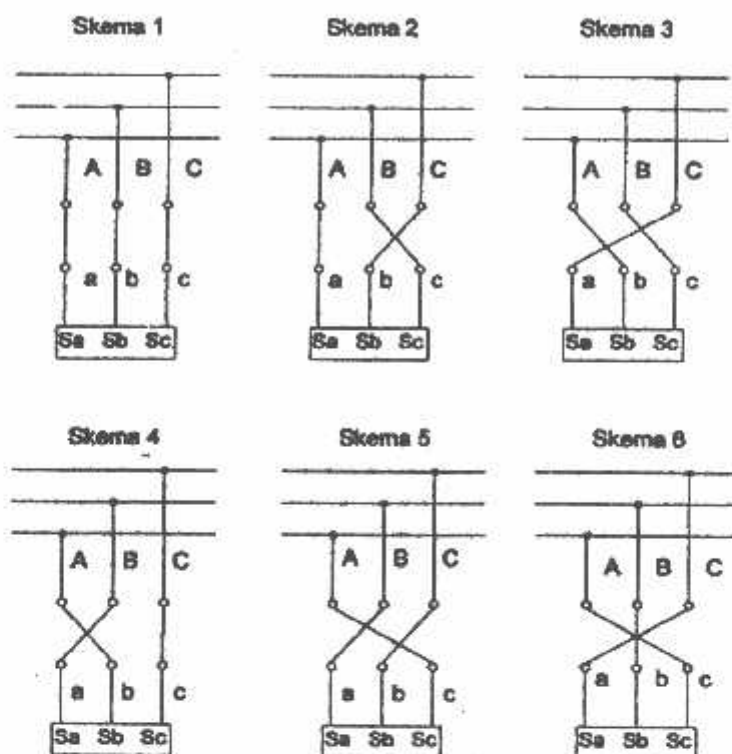
3.6. Metode Penukaran Fasa Jaringan

Proses mengatur penukaran fasa awal dari jaringan (*initial condition*) menjadi suatu penukaran fasa optimum akhir (*optimal condition*), sehingga dari konfigurasi yang terakhir ini diperoleh suatu beban fasa yang seimbang.. Salah satu tipe distribusi yang digunakan disini adalah model jaringan distribusi radial dengan melakukan penukaran fasa pada beban JTR. Karena beban tiga fasa pada JTM tidak seimbang, maka diperlukan penyeimbangan fasa. Penyebab ketidakseimbangan fasa pada JTM adalah karena beban fasa pada masing masing fasa pada JTR tidak seimbang. Penyeimbangan fasa ini tidak mungkin dilakukan pada sisi JTMnya karena adanya pertimbangan berbagai faktor penting seperti kerugian yang disebabkan adanya pemadaman pada JTM, urutan urutan fasa R,S,T yang telah ditetapkan dalam PUIL bahwa fasa R harus terletak dipinggir jalan dll. Hal inilah yang menjadi latar belakang dilakukannya penukaran fasa pada sisi JTRnya.

Untuk menyederhanakan prosedur optimasi, sebuah trafo distribusi dengan berbagai macam beban telah disederhanakan dengan menggunakan model integrasi (penggabungan) transformator distribusi dan beban trafo distribusi.

Biasanya, ukuran konduktor fasa pada saluran JTR adalah sama. Untuk trafo Distribusi, itu mempunyai empat buah konduktor yaitu konduktor fasa R,S,T dan Nol. Dalam memperbaiki sistem tegangan fasa dan arus ketidakseimbangan, sambungan antara beban JTRnya harus disusun secara optimal. Oleh karenanya, pada jenis beban ini ada beberapa petunjuk yang diperlukan untuk menghubungkan ke JTR, seperti yang ditunjukkan pada gambar 3.3. dibawah ini.

Gambar 3.3
Type Koneksi antara JTR dengan Bebannya^[6]



Tabel 3.1
Skema Sambungan dari bermacam-macam Tipe Beban^[6]

Tipe Nilai Beban	Tipe Beban Fasa	Skema Sambungan Valid	Tipe Sambungan
ABC	abc	1 to 6	1:a-A, b-B, c-C; 2:a-A, c-B, b-C; 3:a-C, b-A, c-B; 4:a-B, b-A, c-C; 5:a-B, b-C, c-A; 6: a-C, b-B, c-A;

BAB IV

METODE PENUKARAN FASA JARINGAN YANG DIAPLIKASIKAN KE DALAM PROGRAM KOMPUTER

4.1. Program Komputer Metode Penukaran Fasa Jaringan

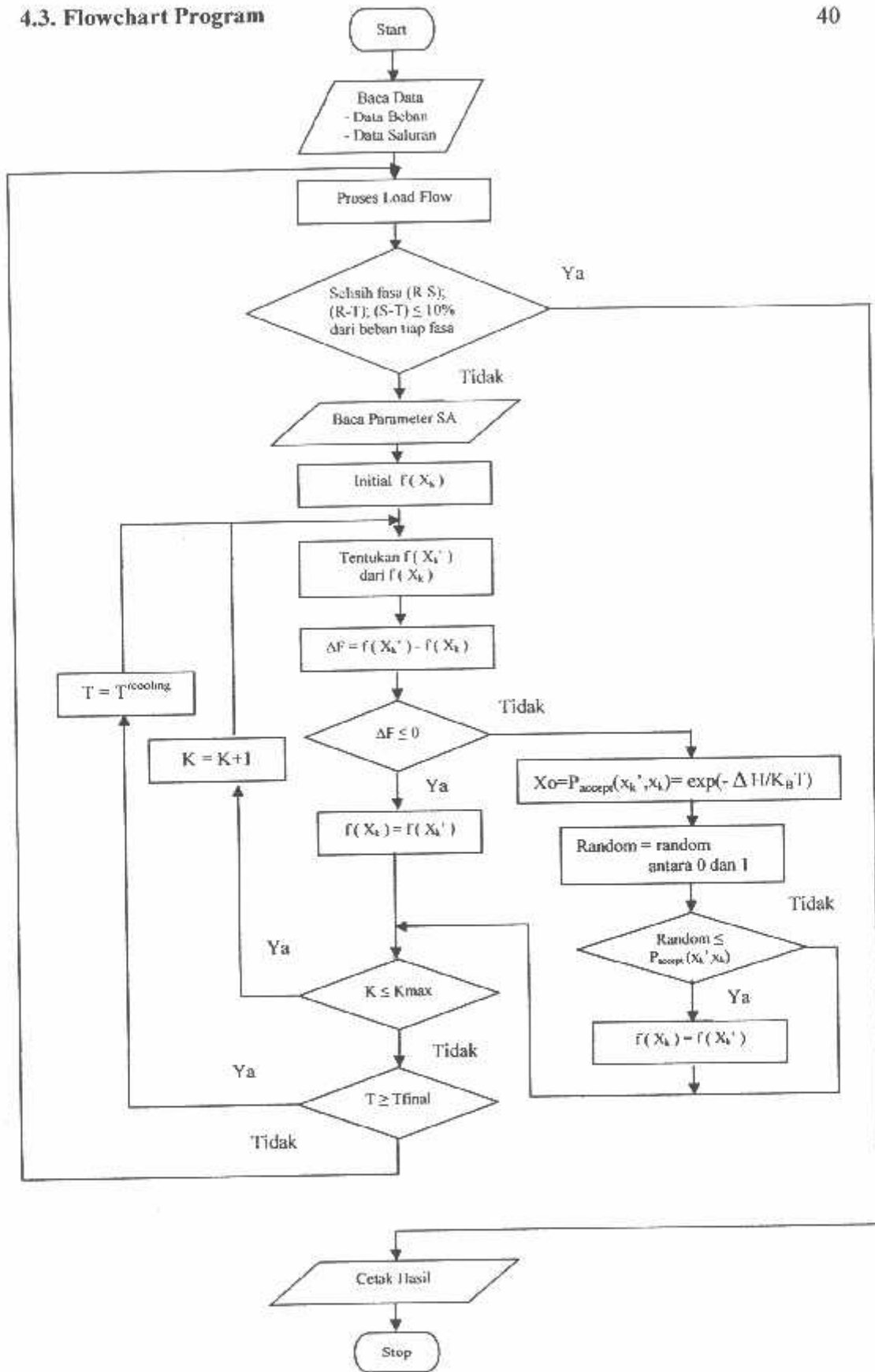
Untuk pemecahan masalah penukaran fasa jaringan digunakan bantuan program komputer. Program komputer sangat berguna untuk mempercepat proses perhitungan yang membutuhkan ketelitian yang tinggi dan sering melibatkan iterasi yang membutuhkan waktu yang lama bila di kerjakan manual.

Program komputer ini menggunakan bahasa pemrograman Borland Delphi versi 7.0 merupakan bahasa pemrograman terstruktur yang relatif mudah untuk dipelajari dan mudah penggunaannya.

4.2. Algoritma Program

Urutan langkah-langkah dalam program komputer yang digunakan ini dapat dilihat pada algoritma program berikut :

1. Masukkan inputan data beban dan saluran
2. Proses Load Flow
3. Jika selisih fasa $(R-S);(R-T);(S-T) \leq 10\%$ dari beban tiap fasa, maka proses berlanjut kelangkah 20.
4. Jika “ tidak” maka proses berlanjut kelangkah 5.
5. Masukkan parameter-parameter SA yang meliputi jumlah iterasi (N), konstanta pendingin($r_{cooling}$), konstanta x_o , jumlah pendinginan & panjang kromosom.



4.4. Data Jaringan

Dalam skripsi ini mengambil data dari Gardu Induk Blimbing Malang dimana *Out going* dari Gardu Induk tersebut merupakan jaringan distribusi primer 20 kV yang melayani 4 (empat) buah penyulang. Salah satu diantaranya adalah penyulang Glintung yang selanjutnya dijadikan sebagai data input dalam program perhitungan aliran daya.

Sebelum perhitungan dilakukan, terlebih dahulu ditetapkan dari gambar *single line diagram* yang ada, bahwa yang dijadikan sebagai *slack bus* adalah Busbar dari GI Blimbing, sedangkan node-node yang ada di sepanjang saluran (yang terdapat trafo distribusinya) dipandang sebagai *load bus*. Data yang ada, terlebih dahulu diubah ke dalam satuan per-unit (pu). Hal ini dilakukan untuk mempermudah proses perhitungan dimana nilai dasar yang di gunakan adalah :

1. Tegangan dasar = 20 kV
2. Daya dasar = 100 kVA

4.4.1. Data Saluran

Untuk menyalurkan tenaga listrik dari GI Blimbing sampai ke pelanggan pada penyulangan Glintung digunakan saluran jenis hantaran udara, dengan spesifikasi jenis konduktor yang digunakan adalah sebagai berikut :

Tabel 4.1
Data Jenis Konduktor Pada Penyulang Glintung

No	Jenis	Luas Penampang (mm ²)	Impedansi $Z = R + jX$ (Ω)	
			R	X
1	AAAC	150	0,2162	0,3305
2	AAAC	120	0,2688	0,3376

Untuk menentukan nilai R(resistansi) dan X (reaktansi) saluran dalam satuan ohm dapat dihitung dengan cara :

$$R = \text{panjang saluran (km)} \times R_{\text{konduktor}} (\Omega)$$

$$X = \text{panjang saluran (km)} \times X_{\text{konduktor}} (\Omega)$$

$$\text{Contoh : } R = 0.303 \times 0,2162 = 0,06551 (\Omega)$$

$$X = 0,303 \times 0,3305 = 0,10014 (\Omega)$$

Untuk mengubah nilai impedansi saluran ke dalam satuan perunit, maka sebelumnya harus ditentukan besarnya nilai impedansi dasar (Z_{dasar}) yang digunakan, yaitu :

$$Z_{\text{dasar}} = \frac{(\text{Tegangan dasar, kV}_{\text{LN}})^2}{\text{DasarMVA}_{10}} = \frac{20^2}{0,1} = 4000\Omega$$

Sehingga besarnya nilai impedansi dalam satuan perunit adalah :

$$Z_{\text{pu}} = \frac{Z\Omega}{4000}$$

$$Z_{\text{pu}} = \frac{0.06551 \Omega}{4000} = 1.63775 \times 10^{-5} \Omega$$

TABEL 4.2
DATA PANJANG SALURAN PENYULANG GLINTUNG

No Saluran	Dari Node	Ke Node	Panjang (m)	Impedansi (Ω)		Impedansi (pu)	
				R	X	R	X
1	0	1	303	0.06551	0.10014	1.63775E-05	4.09438E-09
2	1	2	412	0.08907	0.13617	2.22675E-05	5.56688E-09
3	2	3	19	0.00411	628	1.0275E-06	2.56875E-10
4	3	4	252	0.05448	0.08329	0.00001362	3.405E-09
5	4	5	148	0.03199	0.04891	7.9975E-06	1.99938E-09
6	5	6	266	0.05751	0.08791	1.43775E-05	3.59438E-09
7	6	7	63	0.01362	0.02082	0.000003405	8.5125E-10
8	7	8	150	0.03243	0.04958	8.1075E-06	2.02688E-09
9	8	9	303	0.06551	0.10014	1.63775E-05	4.09438E-09
10	9	10	86	0.02311	0.02903	5.7775E-06	1.44438E-09
11	10	11	202	0.05429	0.0682	1.35725E-05	3.39313E-09
12	11	12	324	0.08709	0.10938	2.17725E-05	5.44313E-09

13	12	13	217.5	0.05846	0.07343	0.000014615	3.65375E-09
14	13	14	86	0.02325	0.02903	5.8125E-06	1.45313E-09
15	14	15	146	0.03937	0.04928	9.8425E-06	2.46063E-09
16	15	16	410	0.11034	0.13842	0.000027585	6.89625E-09
17	16	17	96	0.0258	0.03241	0.00000645	1.6125E-09
18	17	18	328	0.08816	0.11073	0.00002204	5.51E-09
19	18	19	241	0.06478	0.08136	0.000016195	4.04875E-09
20	19	20	51	0.01371	0.01722	3.4275E-06	8.56875E-10
21	20	21	348	0.09354	0.11748	0.000023385	5.84625E-09
22	21	22	386	0.10375	0.13031	2.59375E-05	6.48438E-09
23	22	23	145	0.03897	0.04895	9.7425E-06	2.43563E-09
24	23	24	84	0.02257	0.02836	5.6425E-06	1.41063E-09
25	24	25	35	0.00941	0.01182	2.3525E-06	5.88125E-10
26	25	26	162	0.04435	0.05469	1.10875E-05	2.77188E-09
27	26	27	265	0.07123	0.08946	1.78075E-05	4.45188E-09
28	27	28	215	0.05779	0.07258	1.44475E-05	3.61188E-09
29	8	29	234	0.05059	0.07734	1.26475E-05	3.16188E-09
30	29	30	42	0.00908	0.01388	0.00000227	5.675E-10
31	30	31	201	0.04345	0.06643	1.08625E-05	2.71563E-09
32	31	32	205	0.04432	0.06775	0.00001108	2.77E-09
33	32	33	290	0.06269	0.09585	1.56725E-05	3.91813E-09
34	33	34	88	0.01903	0.02971	4.7575E-06	1.18938E-09
35	30	35	221	0.04778	0.07304	0.000011945	2.98625E-09
36	31	36	161	0.03481	0.05321	8.7025E-06	2.17563E-09
37	33	37	368	0.07956	0.12162	0.00001989	4.9725E-09
38	31	38	92	0.01989	0.03041	4.9725E-06	1.24313E-09
39	18	39	79	0.02124	0.02667	0.00000531	1.3275E-09
40	21	40	35	0.00941	0.01182	2.3525E-06	5.88125E-10
41	22	41	150	0.04032	0.05064	0.00001008	2.52E-09
42	24	42	140	0.03763	0.04726	9.4075E-06	2.35188E-09
43	23	43	208	0.05591	0.07022	1.39775E-05	3.49438E-09
44	43	44	154	0.04139	0.05199	1.03475E-05	2.58688E-09
45	44	45	195	0.05242	0.06583	0.000013105	3.27625E-09
46	45	46	120	0.03226	0.04051	0.000008065	2.01625E-09
47	46	47	119	0.03198	0.04017	0.000007995	1.99875E-09
48	43	48	37	0.00995	0.01249	2.4875E-06	6.21875E-10
49	48	49	296	0.07956	0.09993	0.00001989	4.9725E-09
50	49	50	85	0.02285	0.02869	5.7125E-06	1.42813E-09
51	49	51	37	0.00995	0.01249	2.4875E-06	6.21875E-10
52	49	52	254	0.06828	0.08575	0.00001707	4.2675E-09
53	2	53	98	0.02634	0.03308	0.000006585	1.64625E-09
54	6	54	75	0.01622	0.02479	0.000004055	1.01375E-09
55	54	55	146	0.03157	0.04825	7.8925E-06	1.97313E-09

4.4.2 Data Pembebanan

Data pembebanan diperoleh dari data gardu dan presentase pembebanan dari tiap-tiap bus dengan asumsi bahwa sistem berada pada kondisi normal dengan faktor daya sebesar 0,86 dan rugi-rugi pada trafo distribusi diabaikan. Contoh perhitungan pada bus no 2 sebagai berikut :

$$\begin{aligned} \text{Kapasitas Trafo} &= 200 \text{ kVA} & \text{Beban Gardu} &= 80 \% \\ \text{Faktor Daya} &= \cos \varphi = 0,86 & \sin \varphi &= 0,5102 \\ \text{Pembebanan} &= 200 \times 80 \% = 160 \text{ KVA} \\ \text{Beban Akif} &= 160 \times \cos \varphi = 160 \times 0,86 = 137,6000 \text{ kW} \\ \text{Beban Reaktif} &= 160 \times \sin \varphi = 160 \times 0,5102 = 81,6320 \text{ kVAR} \end{aligned}$$

Dengan cara yang sama, dapat dilanjutkan untuk bus-bus yang lain.

Tabel 4.3
Data Pembebanan Pada Tiap Bus Penyulang Glintang

No Bus	Kapasitas Trafo (kVA)	Beban Gardu (%)	Pembebanan (kVA)		Tipe Bus
			P(kW)	Q (kVAR)	
1	-	0	0.0000	0.0000	Slack
2	200	80	137.6000	81.6320	Load
3	-	0	0.0000	0.0000	Load
4	100	28	24.0800	14.2856	Load
5	160	41	56.4160	33.4691	Load
6	200	76	130.7200	77.5504	Load
7	100	72	61.9200	36.7344	Load
8	-	0	0.0000	0.0000	Load
9	200	73	125.5600	74.4892	Load
10	160	68	93.5680	55.5098	Load
11	250	55	118.2500	70.1725	Load
12	-	0	0.0000	0.0000	Load
13	100	87	74.8200	44.3874	Load
14	-	0	0.0000	0.0000	Load
15	160	27	37.1520	22.0406	Load
16	-	0	0.0000	0.0000	Load
17	250	95	204.2500	121.1725	Load
18	200	62	106.6400	63.2648	Load

19	250	46	98.9000	58.6730	Load
20	-	0	0.0000	0.0000	Load
21	160	72	99.0720	58.7750	Load
22	160	82	112.8320	66.9382	Load
23	250	108.29	232.8235	138.1239	Load
24	400	68.28	234.8832	139.3458	Load
25	250	48.84	105.0060	62.2954	Load
26	150	53	68.3700	40.5609	Load
27	290	48.89	84.0908	49.8874	Load
28	100	46	39.5600	23.4692	Load
29	160	40.01	55.0538	32.6610	Load
30	160	64.29	88.4630	52.4812	Load
31	160	44.38	61.0669	36.2283	Load
32	-	0	0.0000	0.0000	Load
33	100	19	16.3400	9.6938	Load
34	100	59	50.7400	30.1018	Load
35	160	116.85	160.7856	95.38699	Load
36	-	0	0.0000	0.0000	Load
37	100	38	32.6800	19.3876	Load
38	-	0	0.0000	0.0000	Load
39	100	44	37.8400	22.4488	Load
40	-	0	0.0000	0.0000	Load
41	-	0	0.0000	0.0000	Load
42	100	10.92	9.3912	5.5714	Load
43	150	66.85	86.2365	51.16031	Load
44	150	25	32.2500	19.1325	Load
45	100	52	44.7200	26.5304	Load
46	100	95.11	81.7946	48.52512	Load
47	-	0	0.0000	0.0000	Load
48	160	28.50	39.2160	23.2651	Load
49	160	35	48.1600	28.5712	Load
50	200	6.56	11.2832	6.6938	Load
51	-	0	0.0000	0.0000	Load
52	150	46	59.3400	35.2038	Load
53	150	16.88	21.7752	12.9183	Load
54	250	57.64	123.9260	73.5198	Load
55	100	31.5	27.0900	16.0713	Load
56	250	60.28	129.6020	76.8871	Load

Tabel 4.4
Data Gardu Dan Presentase Pembebanan Penyulang Glintang sebelum
penukaran fasa

No Node	No Gardu	Kapasitrs (kVA)	Beban Gardu (%)	Power Faktor	Pembebanan					
					R		S		T	
					P(kW)	Q (kVAR)	P(kW)	Q (kVAR)	P(kW)	Q (kVAR)
0	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
1	196	200	80	0.86	44.1648	26.153	31.0288	18.411	36.2452	21.551
2	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
3	575	160	20	0.86	8.9736	5.275	10.0276	5.950	15.2440	9.092
4	141	200	76	0.86	36.2184	21.439	43.8944	26.045	47.4080	28.173
5	93	100	85	0.86	19.1904	11.337	23.8392	14.145	28.6772	17.061
6	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
7	51	250	55	0.86	34.7048	20.541	39.1644	23.239	39.6508	23.571
8	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
9	148	250	71.10	0.86	48.1380	28.510	48.2460	28.627	53.0840	31.541
10	210	400	68.28	0.86	120.9800	71.724	157.4144	93.741	134.4400	79.805
11	16	250	48.84	0.86	33.9480	20.092	34.0560	20.208	37.0020	21.999
12	211	150	53	0.86	22.0284	13.020	24.2176	14.370	19.7848	11.786
13	40	250	50	0.86	29.5964	17.510	32.9208	19.534	41.5428	24.694
14	398	100	46	0.86	8.2168	4.826	13.4332	7.971	15.4332	9.204
15	153	160	40.01	0.86	16.5416	9.765	19.2984	11.451	19.2172	11.449
16	165	160	64.29	0.86	27.7044	16.388	43.5160	25.821	34.5424	20.541
17	107	160	44.38	0.86	2.9192	1.684	14.5684	8.644	20.1632	12.010
18	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
19	437	100	32.18	0.86	6.5140	3.816	9.4600	5.613	11.0816	6.622
20	142	160	20.13	0.86	3.6760	2.133	11.3520	6.736	11.4600	6.847
21	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
22	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
23	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
24	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
25	114	150	16.88	0.86	15.2172	8.979	11.3520	6.736	13.3520	7.969
26	173	250	57.64	0.86	35.8400	21.214	39.7320	23.576	48.3540	28.734
27	565	100	31.50	0.86	10.2980	6.061	3.7840	2.245	12.4060	7.408
28	166	250	60.28	0.86	41.5160	24.581	45.4080	26.943	42.6780	25.367
29	301	100	74	0.86	12.0008	7.071	20.6228	12.237	30.1908	17.959
30	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
31	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
32	112	250	47	0.86	24.6772	14.592	34.8128	20.657	31.1368	18.520
33	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
34	154	160	82	0.86	32.2452	19.081	38.4076	22.790	36.6236	21.775
35	221	160	27	0.86	7.6492	4.490	12.2980	7.297	16.3792	9.765
36	147	250	47	0.86	31.1100	18.408	32.3532	19.197	36.4344	21.663
37	509	160	69	0.86	30.7316	18.183	31.5964	18.748	36.0560	21.439
38	146	250	47	0.86	25.6232	15.153	31.5964	18.748	31.3260	18.632

39	762	100	19	0.86	5.3788	3.143	0.7568	0.449	9.7572	5.837
40	449	100	38	0.86	6.3248	3.704	11.9196	7.073	12.9736	7.745
41	860	100	44	0.86	4.2436	2.469	15.5144	9.206	16.0008	9.541
42	137	150	46	0.86	15.9740	9.428	18.9200	11.226	20.1632	12.010
43	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
44	529	100	10.92	0.86	3.5948	2.133	4.5408	2.694	7.7572	4.602
45	320	150	25.57	0.86	15.9740	9.428	22.704	13.472	17.1360	10.214
46	58	150	25	0.86	1.7840	1.010	12.4872	7.409	15.6224	9.316
47	156	100	52	0.86	15.0280	8.867	10.9736	6.511	17.1360	10.214
48	485	200	10.35	0.86	1.7840	1.010	9.4600	5.613	5.7840	3.480
49	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
50	59	200	6.56	0.86	2.2704	1.347	2.4596	1.459	6.0544	3.592
51	75	160	28.50	0.86	15.9740	9.428	22.1364	13.135	29.8124	17.735
52	599	160	35	0.86	0.3784	0.224	1.3244	0.786	4.7300	2.806
53	996	100	28	0.86	4.2436	2.469	8.8924	5.276	10.1356	6.061
54	536	200	73	0.86	39.8132	23.571	35.0020	20.769	24.7040	14.704
55	281	160	68	0.86	33.9480	20.092	25.7312	15.268	33.2180	19.755

Tabel 4.5
Data Gardu Dan Presentase Pembebanan Penyulang Glintang sesudah
penukaran fasa

No Node	No Gardu	Kapasitas (kVA)	Beban Gardu (%)	Power Faktor	Pembebanan					
					R		S		T	
					P(kW)	Q (kVAR)	P(kW)	Q (kVAR)	P(kW)	Q (kVAR)
0	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
1	196	200	80	0.86	31.0288	18.411	44.1648	26.153	36.2452	21.551
2	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
3	575	160	20	0.86	8.9736	5.275	10.0276	5.950	15.2440	9.092
4	141	200	76	0.86	36.2184	21.439	43.8944	26.045	47.4080	28.173
5	93	100	85	0.86	23.8392	14.145	19.1904	11.337	28.6772	17.061
6	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
7	51	250	55	0.86	34.7048	20.541	39.6508	23.571	39.1644	23.239
8	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
9	148	250	71.10	0.86	48.1360	28.510	53.0840	31.541	48.2460	28.627
10	210	400	68.28	0.86	120.9800	71.724	157.4144	93.741	134.4400	79.805
11	16	250	48.84	0.86	33.9480	20.092	34.0560	20.208	37.0020	21.999
12	211	150	53	0.86	22.0284	13.020	24.2176	14.370	19.7848	11.786
13	40	250	50	0.86	29.5964	17.510	32.9208	19.534	41.5428	24.694
14	398	100	46	0.86	8.2168	4.826	13.4332	7.971	15.4332	9.204
15	153	160	40.01	0.86	16.5416	9.765	19.2984	11.451	19.2172	11.449
16	165	160	64.29	0.86	27.7044	16.388	43.5160	25.821	34.5424	20.541
17	107	160	44.38	0.86	2.9192	1.684	14.5684	8.644	20.1632	12.010
18	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000

19	437	100	32.18	0.86	6.5140	3.816	9.4600	5.613	11.0816	6.622
20	142	160	20.13	0.86	3.6760	2.133	11.3520	6.736	11.4600	6.847
21	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
22	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
23	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
24	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
25	114	150	16.88	0.86	15.2172	8.979	11.3520	6.736	13.3520	7.969
26	173	250	57.64	0.86	35.8400	21.214	39.7320	23.576	48.3540	28.734
27	565	100	31.50	0.86	10.2980	6.061	3.7840	2.245	12.4060	7.408
28	166	250	60.28	0.86	41.5160	24.581	45.4080	26.943	42.6780	25.367
29	301	100	74	0.86	12.0008	7.071	20.6228	12.237	30.1908	17.959
30	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
31	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
32	112	250	47	0.86	24.6772	14.592	34.8128	20.657	31.1368	18.520
33	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
34	154	160	82	0.86	32.2452	19.081	38.4076	22.790	36.6236	21.775
35	221	160	27	0.86	7.6492	4.490	12.2980	7.297	16.3792	9.765
36	147	250	47	0.86	31.1100	18.408	32.3532	19.197	36.4344	21.663
37	509	160	69	0.86	30.7316	18.183	31.5964	18.748	36.0560	21.439
38	146	250	47	0.86	25.6232	15.153	31.5964	18.748	31.3260	18.632
39	762	100	19	0.86	5.3788	3.143	0.7568	0.449	9.7572	5.837
40	449	100	38	0.86	6.3248	3.704	11.9196	7.073	12.9736	7.745
41	860	100	44	0.86	4.2436	2.469	15.5144	9.208	16.0008	9.541
42	137	150	46	0.86	15.9740	9.428	18.9200	11.226	20.1632	12.010
43	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
44	529	100	10.92	0.86	3.5948	2.133	4.5408	2.694	7.7572	4.602
45	320	150	25.57	0.86	15.9740	9.428	22.704	13.472	17.1360	10.214
46	58	150	25	0.86	1.7840	1.010	12.4872	7.409	15.6224	9.316
47	156	100	52	0.86	15.0280	8.867	10.9736	6.511	17.1360	10.214
48	485	200	10.35	0.86	1.7840	1.010	9.4600	5.613	5.7840	3.480
49	-	-	0	0.86	0.0000	0.0000	0.0000	0.000	0.0000	0.0000
50	59	200	6.56	0.86	2.2704	1.347	2.4596	1.459	6.0544	3.592
51	75	160	28.50	0.86	15.9740	9.428	22.1364	13.135	29.8124	17.735
52	599	160	35	0.86	0.3784	0.224	1.3244	0.786	4.7300	2.806
53	996	100	28	0.86	4.2436	2.469	8.8924	5.276	10.1356	6.061
54	536	200	73	0.86	39.8132	23.571	35.0020	20.769	24.7040	14.704
55	281	160	68	0.86	33.9480	20.092	25.7312	15.268	33.2180	19.755

4.5. Analisa Perhitungan

Perhitungan penukaran fasa dalam trafo distribusi yang dihubungkan pada penyulang primer diawali dengan perhitungan aliran daya dengan menggunakan metode *Fast Decoupled Three Phase*. Studi aliran daya untuk mengetahui harga tegangan dan sudut fasa ditiap-tiap bus dan arus yang mengalir pada saluran antar bus. Setelah studi aliran daya dilakukan, barulah dilakukan perhitungan aliran daya dan rugi-rugi daya pada saluran sebelum dilakukan penukaran fasa. Langkah berikutnya, dilakukan penukaran fasa dengan menggunakan metode SA. Kemudian dilakukan perhitungan aliran daya dan rugi rugi daya saluran setelah penukaran fasa. Dari perhitungan aliran daya setelah penukaran fasa ini, dapat diketahui jumlah pembebanan pada tiap tiap fasa yang seimbang dan juga rugi rugi daya saluran yang lebih kecil dari rugi rugi daya sebelum penukaran fasa. Selanjutnya bandingkan kondisi jumlah pembangkitan, jumlah pembebanan, dan rugi rugi daya saluran serta efisiensi daya sebelum dan setelah penukaran fasa.

Untuk memudahkan perhitungan dan analisa pada sistem tenaga, biasanya dipakai harga-harga dalam per-satuan unit. Harga per-satuan unit adalah harga yang sebenarnya dibagi dengan harga dasar, harga dasar ini dapat dipilih sembarang. Harga yang dipilih pada studi ini adalah 20 kV dan 100 kVA sebagai harga tegangan dasar dan daya dasar. Mengingat bahwa pada jaringan tidak dilakukan pengukuran faktor daya maka perhitungan ini diambil harga faktor daya sebesar 0,86.

Tabel 4.6
Kondisi Awal Rugi-rugi Daya Saluran Sebelum Penukaran Fasa

Bus	R Lre kW	R Lim kVar	S Lre kW	S Lim kVar	T Lre kW	T Lim kVar
1	1.009	0.095	1.408	0.183	1.505	0.239
2	1.352	0.116	1.797	0.233	1.912	0.305
3	0.062	0.005	0.081	0.011	0.087	0.014
4	0.799	0.069	1.058	0.137	1.119	0.178
5	0.430	0.037	0.569	0.074	0.600	0.096
6	0.240	-0.592	0.336	-0.703	0.375	-0.691
7	0.046	-0.117	0.069	-0.145	0.078	-0.144
8	0.315	0.026	0.437	0.056	0.464	0.075
9	0.364	0.032	0.510	0.065	0.539	0.084
10	0.091	0.003	0.130	0.009	0.137	0.014
11	0.119	0.002	0.168	0.012	0.193	0.021
12	0.157	0.002	0.229	0.016	0.263	0.029
13	0.080	0.001	0.136	0.010	0.160	0.018
14	0.000	0.000	0.045	0.003	0.052	0.006

Tabel 4.6 selengkapnya dapat dilihat pada lampiran

Dari tabel 4.6 diatas ditunjukkan rugi rugi daya aktif dan reaktif tiap fasa pada tiap saluran setelah dilakukan proses aliran daya sebelum penukaran fasa,dengan jumlah salurannya ada 55 saluran. Yang dimaksud dengan BUS pada tabel diatas adalah saluran distribusinya.

Gambar 4.2
Tampilan Perhitungan Rugi-rugi Daya sebelum Penukaran Fasa

Summary Loadflow	kVA		
Jumlah Pembangkitan		[1064.929+ j 635.529]	[1134.305+ j 643.179]
Jumlah Pembebanan		[997.137+ j 560.032]	[1061.796+ j 630.028]
Jumlah Rugi-rugi		[5.728+ j 0.320]	[7.736+ j 0.023]
Iterasi	2		
Waktu Perhitungan	0.000	(jam : menit : detik)	

Dari gambar 4.2 diatas ditunjukkan jumlah pembangkitan, pembebanan dan rugi rugi daya tiap fasa dari 55 saluran setelah dilakukan proses aliran daya sebelum dilakukan penukaran fasa.

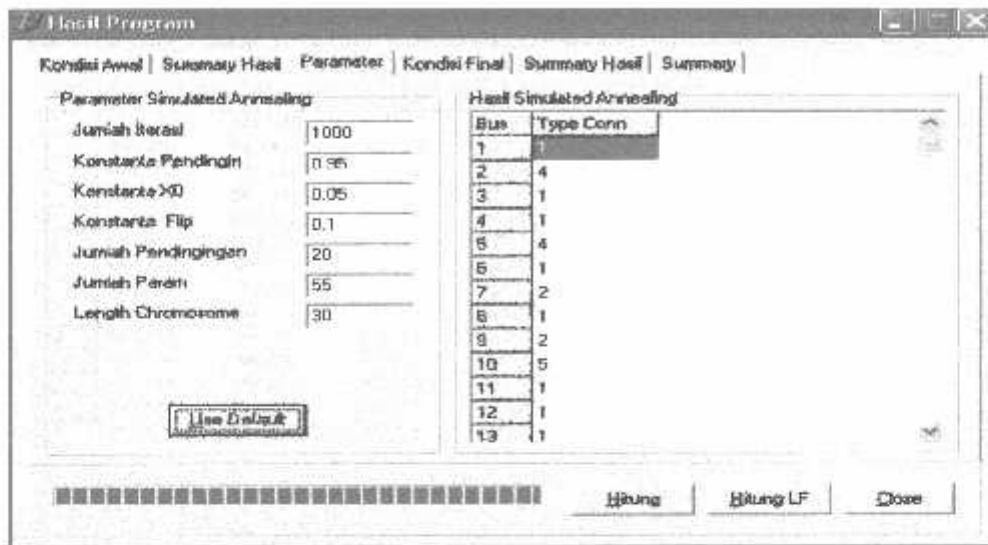
Tabel 4.7
Hasil Perhitungan Aliran Daya sebelum Penukaran Fasa

Bus	R Sre kW	R Sim kVar	S Sre kW	S Sim kVar	T Sre kW	T Sim kVar
1	899.475	564.054	1064.929	635.529	1134.305	649.179
2	854.831	535.632	1033.256	615.425	1097.355	625.624
3	850.565	532.869	1024.334	610.072	1087.188	620.915
4	841.308	525.844	1013.926	603.125	1071.532	610.767
5	804.937	505.001	969.828	576.544	1023.903	562.028
6	755.114	493.710	945.210	562.435	994.440	554.595
7	711.225	447.577	884.316	526.405	936.354	533.004
8	676.410	425.740	844.954	502.754	896.532	508.596
9	512.221	320.694	643.087	382.518	678.154	387.269
10	464.047	292.101	594.788	353.772	625.012	365.603
11	343.020	220.267	436.737	259.877	490.491	275.622
12	309.010	200.031	402.599	239.459	453.377	253.393
13	266.945	185.926	375.315	224.965	433.525	241.452
14	267.230	160.389	345.375	215.290	391.920	216.711

Tabel 4.7 selengkapnya dapat dilihat pada lampiran

Dari tabel 4.7 diatas ditunjukkan nilai daya aktif dan reaktif tiap tiap fasa pada tiap tiap saluran setelah dilakukan proses aliran daya sebelum penukaran fasa, dengan jumlah salurannya ada 55 saluran.

Gambar 4.3
Tampilan Parameter SA dan Skema type penukaran Fasa



Dari gambar 4.3 diatas ditunjukkan parameter SA (pada gambar sebelah kiri) dan skema type penukaran fasa (pada gambar sebelah kanan). Dengan parameter yang telah ditentukan diatas dan skema type penukaran fasa yang dihasilkan maka diperoleh nilai rugi rugi daya yang minimum apabila aliran dayanya dijalankan.

Dari tabel 4.9 diatas ditunjukkan rugi rugi daya aktif dan reaktif tiap fasa pada tiap saluran setelah dilakukan proses aliran daya setelah penukaran fasa,dengan jumlah salurannya ada 55 saluran.

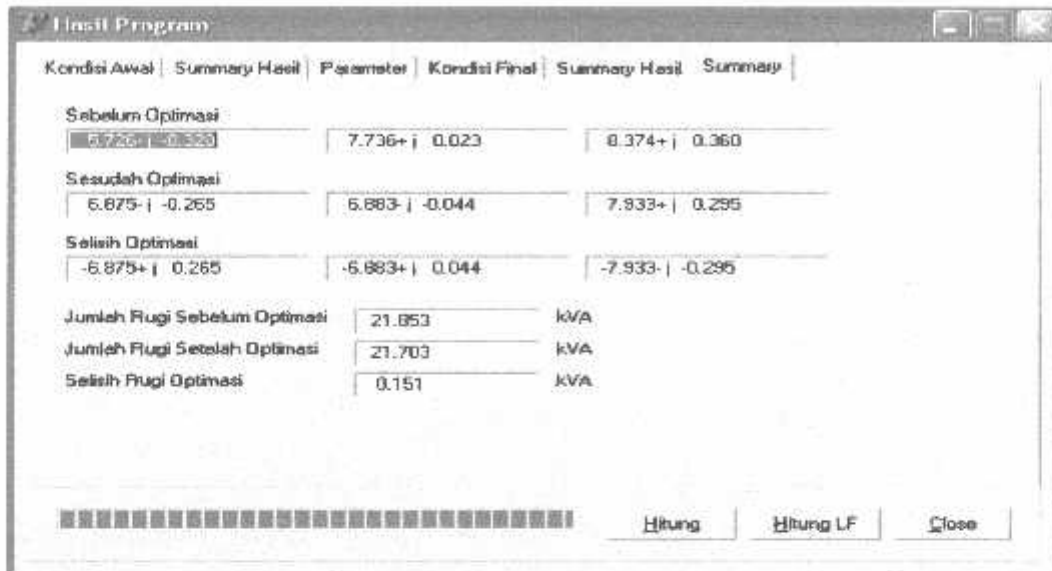
Gambar 4.4
Tampilan Perhitungan Rugi – rugi Daya Saluran Setelah Penukaran Fasa

Summary Loadflow			
Jumlah Pembangkitan	kVA	907.027+ 507.720	1097.689+ 631.800
Jumlah Pembebanan	kVA	385.089+ 592.833	1094.464+ 626.159
Jumlah Rugi-Rugi	kVA	6.875- -0.265	6.883- -0.044
Iterasi		2	
Waktu Perhitungan		0:0:54.297 (jam : menit : detik : mdesik)	

Buttons: Hitung, Hitung LF, Close

Dari gambar 4.4 diatas ditunjukkan jumlah pembangkitan, pembebanan dan rugi rugi daya tiap fasa dari 55 saluran setelah dilakukan proses aliran daya setelah dilakukan penukaran fasa.

Gambar 4.5
Tampilan Selisih Rugi – rugi Saluran Sebelum dan Setelah Optimasi



Dari gambar 4.5 diatas ditunjukkan jumlah rugi rugi daya sebelum dan setelah dilakukan penukaran fasa. Dapat diketahui pula selisih rugi rugi dayanya/pengurangan rugi rugi dayanya yakni 0,151 KVA.

Setelah di lakukan penukaran fasa terhadap beban fasa yang tak seimbang maka diperoleh suatu skema penukaran fasa yang optimum sebagai berikut :

Tabel 4.10
Skema penukaran fasa yang optimum dengan metode SA

BUS	TYPE PENUKARAN FASA	BUS	TYPE PENUKARAN FASA
1	1	29	4
2	4	30	1
3	1	31	1
4	1	32	1
5	4	33	1
6	1	34	1
7	2	35	1
8	1	36	1
9	2	37	4
10	5	38	1
11	1	39	1

12	1	40	1
13	1	41	1
14	1	42	1
15	4	43	1
16	1	44	1
17	1	45	4
18	1	46	4
19	1	47	1
20	3	48	4
21	1	49	1
22	5	50	1
23	1	51	3
24	2	52	1
25	1	53	2
26	3	54	1
27	1	55	1
28	1		

Beban dikatakan seimbang yakni apabila selisih beban pada masing-masing fasa feeder tidak lebih dari 5% - 10% dari beban fasanya.

4.5.1. Perbedaan kondisi pembangkitan, pembebanan, rugi-rugi daya, efisiensi daya sebelum dan setelah penukaran fasa

- Kondisi jumlah pembangkitan sebelum penukaran fasa (lihat gambar 4.2)

$$\text{fasa R} = 899,475 + j 564,054 = 1061,702 \text{ kVA}$$

$$\text{fasa S} = 1064,929 + j 635,529 = 1240,149 \text{ kVA}$$

$$\text{fasa T} = 1134,305 + j 649,179 = 1306,936 \text{ kVA}$$

$$\text{Total pembangkitan dari tiap fasa} = 3608,7 \text{ kVA}$$

- Kondisi jumlah pembangkitan setelah penukaran fasa (lihat gambar 4.4)

$$\text{fasa R} = 1013,167 + j 619,100 = 1187,347 \text{ kVA}$$

$$\text{fasa S} = 987,827 + j 597,728 = 1154,591 \text{ kVA}$$

$$\text{fasa T} = 1097,689 + j 631,800 = 1266,528 \text{ kVA}$$

$$\text{Total pembangkitan dari tiap fasa} = 3608,5 \text{ kVA}$$

2. Kondisi jumlah pembebanan sebelum penukaran fasa (lihat gambar 4.2)

$$\text{fasa R} = 897,137 + j 560,032 = 1057,587 \text{ kVA}$$

$$\text{fasa S} = 1061,796 + j 630,028 = 1234,644 \text{ kVA}$$

$$\text{fasa T} = 1130,898 + j 643,211 = 1294,616 \text{ kVA}$$

$$\text{Total pembebanan dari tiap-tiap fasa} = 3586,847 \text{ kVA}$$

Kondisi jumlah pembebanan setelah penukaran fasa (lihat gambar 4.4)

$$\text{fasa R} = 1010,277 + j 614,279 = 1182,370 \text{ kVA}$$

$$\text{fasa S} = 985,089 + j 592,833 = 1149,718 \text{ kVA}$$

$$\text{fasa T} = 1094,464 + j 626,158 = 1254,709 \text{ kVA}$$

$$\text{Total pembebanan dari tiap-tiap fasa} = 3586,797 \text{ kVA}$$

Sebelum dilakukan penukaran fasa, terjadi kelebihan beban yaitu:

- pada Fasa S, kelebihan bebannya =

$$\begin{aligned} & \text{Beban sebelum penukaran fasa} - \text{Beban setelah penukaran fasa} \\ & = 1234,644 \text{ kVA} - 1149,718 \text{ kVA} = 84,926 \text{ KVA} \end{aligned}$$

- pada Fasa T, kelebihan bebannya =

$$\begin{aligned} & \text{Beban sebelum penukaran fasa} - \text{Beban setelah penukaran fasa} \\ & = 1294,616 \text{ kVA} - 1254,709 \text{ kVA} = 39,907 \text{ KVA} \end{aligned}$$

❖ Dari kondisi jumlah pembebanan sebelum penukaran fasa dapat diketahui bahwa beban fasanya tidak seimbang, yaitu:

$$\text{fasa R} = 897,137 + j 560,032 = 1057,587 \text{ kVA}$$

$$\text{fasa S} = 1061,796 + j 630,028 = 1234,644 \text{ kVA}$$

$$\text{selisih beban fasa R-T} = 72,3 \text{ KVA}$$

$$\text{selisih beban fasa S-T} = 104,9 \text{ KVA}$$

Karena selisih fasa R-S, R-T, S-T \leq beban maksimum yang diijinkan maka tidak terjadi kelebihan beban sehingga beban fasanya menjadi seimbang.

3. Kondisi jumlah rugi-rugi daya sebelum penukaran fasa (lihat gambar 4.2)

$$\text{fasa R} = 5,726 - j -0,320 \text{ kVA}$$

$$\text{fasa S} = 7,736 + j 0,023 \text{ kVA}$$

$$\text{fasa T} = 8,374 + j 0,360 \text{ kVA}$$

$$\text{Total rugi-rugi daya dari tiap-tiap fasa} = 21,853 \text{ kVA}$$

Kondisi jumlah rugi-rugi daya setelah penukaran fasa (lihat gambar 4.4)

$$\text{fasa R} = 6,875 - j -0,265 \text{ kVA}$$

$$\text{fasa S} = 6,883 - j -0,044 \text{ kVA}$$

$$\text{fasa T} = 7,933 + j 0,295 \text{ kVA}$$

$$\text{Total rugi-rugi daya dari tiap-tiap fasa} = 21,703 \text{ kVA.}$$

Tabel 4.11
Perbandingan Rugi – rugi daya Saluran

Rugi – rugi Sebelum KVA	Rugi – rugi Sesudah KVA	Minimalisasi Rugi - rugi KVA
21,853	21,703	0,151

4. Efisiensi dari dayanya adalah

- a. Sebelum penukaran fasa:



BAB V

KESIMPULAN

5.1. Kesimpulan

1. Terjadi kelebihan beban sebelum dilakukan penukaran fasa, yaitu:

- pada Fasa S, kelebihan bebannya =
Beban sebelum penukaran fasa - Beban setelah penukaran fasa
= 1234,644 kVA - 1149,718 kVA = 84,926 KVA
- pada Fasa T, kelebihan bebannya =
Beban sebelum penukaran fasa - Beban setelah penukaran fasa
= 1294,616 kVA - 1254,709 kVA = 39,907 KVA

setelah dilakukan penukaran fasa, bebannya :

$$\begin{aligned} \text{fasa R} &= 1010,277 + j 14,279 &= 1182,370 \text{ kVA} \\ \text{fasa S} &= 985,089 + j 592,833 &= 1149,718 \text{ kVA} \\ \text{fasa T} &= 1094,464 + j 626,158 &= 1254,709 \text{ kVA} \end{aligned}$$

Karena selisih beban fasanya ($R - S = 32,6$; $R - T = 72,3$; $S - T = 104,9$) kVA \leq toleransi beban maksimal yang diijinkan (fasa R=118,2 ; fasa S= 114,9 ; fasa T= 125,5) kVA, maka dikatakan bahwa bebannya seimbang.

2. Total rugi-rugi daya dari tiap-tiap fasa sebelum dan sesudah penukaran fasa

berbeda yaitu : sebelum penukaran fasa = 21,853 kVA

setelah penukaran fasa = 21,703 kVA

sehingga pengurangan rugi-rugi dayanya = 0,151 kVA

3. Efisiensi dari dayanya adalah

a. Sebelum penukaran fasa = 99,394 %

b. Setelah penukaran fasa = 99,399 %

DAFTAR PUSTAKA

1. Hasan Basri, *Sistem Tenaga Listrik*, Balai Penerbit dan Humas ISTN, Jakarta, 1996.
 2. Turan Gonen, *Electric Power Distribution System Engineering*, University of Missouri at Columbia.
 3. Bradley J. Buchan and Casey Lambert, *Simulated Annealing Application*, Mechanical engineering Departement, University of Victoria, 18 November, 1999.
 4. Jinxiang. Zhu, Griff Bilbro and Mo Yuan Chow, *Phase Balancing Using simulated Annealing*, IEE transaction on powers systems, Vol. 14, No. 4, November 1999.
 5. Dedy, Widayanto, *Analisis Aliran Daya pada Jaringan Distribusi Primer Type Radial Tidak Seimbang Dengan Metode Fast Decoupled Three Phase di Gardu Induk Blimbing Malang*, Skripsi , Malang, 2002.
 6. Tsai – Hsiang, Chen, *Optimalisasi Phase Arrangement Of Distribution Transformers Connected To a Primary Feeder For System Unbalance Improvement And Loss Reduction Using a Genetic Algorithm*, IEEE Transactions On Power System, Vol 15, No 3, August, 2000.
 7. Diklat Manajemen Susuk Distribusi Tegangan Rendah PT. PLN (Persero) Distribusi Jawa timur area Pelayanan dan Jaringan Malang.
 8. J. Zhu, M. Y. Chow, and K. Hoffman, *Phase Swapping to balance a Radial feeder System*, Presented in IEEE Pes, Winter Meeting 1998.
 9. S Sivana Garaju, Dr. N Screeni Vasulu, Dr. M Wijaya Kumar, *Voltage Stability for Radial Distribution Networks*, November 1, 2002.
-

10. Ray D Zimmerman, Hsiao-Dong Chiang. *Fast Decoupled Power Flow For Unbalanced Radial Distribution Systems*, February 2, 1995
 11. J. Zhu, M. Y. Chow, and K. Hoffman, *Phase Balancing Optimazation for a large-scale feeder System*, Submitted to IEEE Transactions on Power Systems, 1997.
-

LAMPIRAN

❖ Hasil Perhitungan

❖ Single Line Penyulang Gantung

❖ Listing Program

Tampilan Tegangan Fasa R,S,T Hasil Perhitungan Aliran Daya Sebelum Penukaran Fasa

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Vrs pu	R Vrm pu	S Vrs pu	S Vrm pu	T Vrs pu	T Vrm pu	R Sre kW
1	0.99918	-0.00062	0.99910	-0.00071	0.99909	-0.00073	899.475
2	0.99811	-0.00143	0.99792	-0.00164	0.99790	-0.00169	854.831
3	0.99806	-0.00147	0.99787	-0.00168	0.99785	-0.00173	850.565
4	0.99742	-0.00195	0.99716	-0.00224	0.99713	-0.00231	841.308
5	0.99706	-0.00223	0.99676	-0.00255	0.99673	-0.00263	804.937
6	0.99651	-0.00182	0.99617	-0.00216	0.99615	-0.00227	785.114
7	0.99639	-0.00173	0.99605	-0.00208	0.99602	-0.00219	711.226
8	0.99607	-0.00197	0.99570	-0.00235	0.99567	-0.00248	676.410
9	0.99559	-0.00233	0.99516	-0.00278	0.99513	-0.00291	512.221
10	0.99546	-0.00243	0.99500	-0.00288	0.99497	-0.00302	464.047
11	0.99522	-0.00259	0.99473	-0.00307	0.99470	-0.00322	343.020
12	0.99486	-0.00282	0.99433	-0.00335	0.99428	-0.00352	309.010
13	0.99464	-0.00297	0.99408	-0.00353	0.99402	-0.00371	286.945
14	0.99456	-0.00303	0.99399	-0.00359	0.99393	-0.00377	257.338

Hitung Hitung LF Close

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Vrs pu	R Vrm pu	S Vrs pu	S Vrm pu	T Vrs pu	T Vrm pu	R Sre kW
14	0.99456	-0.00303	0.99399	-0.00359	0.99393	-0.00377	257.338
15	0.99443	-0.00311	0.99384	-0.00370	0.99378	-0.00388	249.103
16	0.99409	-0.00334	0.99345	-0.00397	0.99337	-0.00418	232.517
17	0.99402	-0.00339	0.99337	-0.00403	0.99329	-0.00424	204.804
18	0.99378	-0.00354	0.99311	-0.00421	0.99302	-0.00443	201.859
19	0.99361	-0.00365	0.99293	-0.00434	0.99282	-0.00457	196.461
20	0.99358	-0.00368	0.99289	-0.00437	0.99278	-0.00460	189.944
21	0.99335	-0.00383	0.99264	-0.00455	0.99252	-0.00479	186.244
22	0.99311	-0.00398	0.99237	-0.00473	0.99224	-0.00499	179.695
23	0.99302	-0.00404	0.99228	-0.00480	0.99215	-0.00506	175.643
24	0.99299	-0.00406	0.99224	-0.00482	0.99211	-0.00509	118.850
25	0.99298	-0.00407	0.99223	-0.00483	0.99210	-0.00509	102.875
26	0.99293	-0.00410	0.99219	-0.00486	0.99205	-0.00513	87.656
27	0.99289	-0.00413	0.99214	-0.00489	0.99201	-0.00516	63.815

Hitung Hitung LF Close

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Vre pu	R Vim pu	S Vre pu	S Vim pu	T Vre pu	T Vim pu	R Sre kW
27	0.99209	-0.00413	0.99214	-0.00409	0.99201	-0.00516	51.815
28	0.99206	-0.00415	0.99212	-0.00491	0.99198	-0.00518	41.516
29	0.99596	-0.00206	0.99556	-0.00246	0.99554	-0.00258	164.051
30	0.99594	-0.00207	0.99554	-0.00247	0.99552	-0.00260	152.048
31	0.99585	-0.00214	0.99545	-0.00255	0.99542	-0.00268	144.392
32	0.99580	-0.00218	0.99539	-0.00259	0.99537	-0.00272	87.656
33	0.99574	-0.00222	0.99533	-0.00264	0.99531	-0.00277	62.977
34	0.99573	-0.00223	0.99532	-0.00265	0.99530	-0.00278	32.245
35	0.99593	-0.00208	0.99554	-0.00248	0.99551	-0.00261	7.649
36	0.99584	-0.00215	0.99543	-0.00256	0.99541	-0.00269	31.110
37	0.99571	-0.00224	0.99530	-0.00267	0.99528	-0.00280	30.732
38	0.99584	-0.00214	0.99544	-0.00255	0.99542	-0.00268	25.623
39	0.99378	-0.00354	0.99311	-0.00421	0.99301	-0.00443	5.379
40	0.99335	-0.00383	0.99263	-0.00455	0.99252	-0.00479	6.325

Hitung | Hitung LF | Close

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Vre pu	R Vim pu	S Vre pu	S Vim pu	T Vre pu	T Vim pu	R Sre kW
40	0.99335	-0.00383	0.99263	-0.00455	0.99252	-0.00479	6.325
41	0.99311	-0.00399	0.99236	-0.00474	0.99224	-0.00500	4.244
42	0.99298	-0.00407	0.99224	-0.00483	0.99211	-0.00509	15.974
43	0.99298	-0.00407	0.99222	-0.00484	0.99209	-0.00510	56.789
44	0.99296	-0.00409	0.99220	-0.00485	0.99206	-0.00512	36.381
45	0.99293	-0.00410	0.99217	-0.00487	0.99204	-0.00514	32.786
46	0.99292	-0.00411	0.99216	-0.00488	0.99203	-0.00515	16.812
47	0.99292	-0.00411	0.99216	-0.00488	0.99202	-0.00515	15.028
48	0.99297	-0.00407	0.99222	-0.00484	0.99208	-0.00511	20.407
49	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	18.623
50	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	2.270
51	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	15.974
52	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	0.378
53	0.99211	-0.00413	0.99263	-0.00455	0.99252	-0.00479	4.244

Hitung | Hitung LF | Close

7 Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Vre pu	R Vim pu	S Vre pu	S Vim pu	T Vre pu	T Vim pu	R Sre kW
43	0.99298	-0.00407	0.99222	-0.00484	0.99209	-0.00510	56.789
44	0.99296	-0.00409	0.99220	-0.00485	0.99206	-0.00512	36.381
45	0.99293	-0.00410	0.99217	-0.00487	0.99204	-0.00514	32.786
46	0.99292	-0.00411	0.99216	-0.00488	0.99203	-0.00515	16.812
47	0.99292	-0.00411	0.99216	-0.00488	0.99202	-0.00515	15.028
48	0.99297	-0.00407	0.99222	-0.00484	0.99208	-0.00511	20.407
49	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	18.623
50	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	2.270
51	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	15.974
52	0.99295	-0.00409	0.99219	-0.00486	0.99205	-0.00513	0.379
53	0.99811	-0.00143	0.99792	-0.00164	0.99790	-0.00169	4.244
54	0.99649	-0.00183	0.99616	-0.00217	0.99614	-0.00228	73.761
55	0.99648	-0.00184	0.99615	-0.00218	0.99613	-0.00229	33.948

Hitung Hitung LF Close

Tabel 4.6
Kondisi awal Rugi rugi Daya Sebelum Penukaran Fasa

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Lre kW	R Lim kVar	S Lre kW	S Lim kVar	T Lre kW	T Lim kVar
1	1.089	0.095	1.408	0.183	1.505	0.239
2	1.352	0.116	1.797	0.233	1.912	0.305
3	0.062	0.005	0.081	0.011	0.087	0.014
4	0.799	0.069	1.058	0.137	1.119	0.178
5	0.430	0.037	0.569	0.074	0.600	0.096
6	0.230	-0.592	0.336	-0.703	0.375	-0.691
7	0.046	-0.117	0.069	-0.145	0.078	-0.144
8	0.315	0.026	0.437	0.056	0.464	0.075
9	0.364	0.032	0.510	0.065	0.539	0.084
10	0.091	0.003	0.130	0.009	0.137	0.014
11	0.119	0.002	0.168	0.012	0.193	0.021
12	0.157	0.002	0.229	0.016	0.263	0.029
13	0.092	0.001	0.136	0.010	0.160	0.018
14	0.030	0.000	0.045	0.003	0.052	0.006

Hitung Hitung LF Close

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Lre kW	R Lim kVar	S Lre kW	S Lim kVar	T Lre kW	T Lim kVar
14	0.030	0.000	0.045	0.003	0.052	0.006
15	0.047	0.000	0.070	0.005	0.081	0.009
16	0.116	-0.000	0.174	0.012	0.203	0.024
17	0.021	-0.000	0.031	0.002	0.038	0.005
18	0.069	-0.001	0.094	0.007	0.115	0.015
19	0.048	-0.001	0.068	0.005	0.080	0.010
20	0.010	-0.000	0.013	0.001	0.016	0.002
21	0.061	-0.001	0.083	0.005	0.099	0.013
22	0.063	-0.001	0.084	0.005	0.100	0.012
23	0.022	0.000	0.027	0.002	0.033	0.004
24	0.005	0.000	0.005	0.000	0.006	0.001
25	0.002	0.000	0.002	0.000	0.002	0.000
26	0.006	0.000	0.006	0.000	0.007	0.001
27	0.002	0.000	0.002	0.000	0.002	0.000

Hitung Hitung LF Close

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Sre kW	R Sim kVar	S Sre kW	S Sim kVar	T Sre kW	T Sim kVar	R Lre kW
15	249.103	164.520	331.914	197.355	376.492	207.433	0.047
16	232.517	154.648	312.545	165.744	357.188	195.799	0.116
17	204.804	138.240	269.016	159.895	322.629	175.223	0.021
18	201.059	136.492	254.409	151.165	302.415	163.108	0.069
19	196.461	132.069	253.625	150.653	292.624	158.432	0.048
20	189.944	128.245	244.159	145.028	281.535	151.796	0.010
21	186.244	124.827	232.773	138.215	270.033	146.093	0.061
22	179.895	119.829	220.820	131.066	257.016	139.491	0.063
23	175.643	116.104	205.294	121.895	241.001	131.153	0.022
24	118.850	76.463	119.201	70.738	136.959	75.343	0.005
25	102.875	65.797	100.280	59.510	116.795	64.563	0.002
26	87.656	55.575	88.926	52.769	103.440	57.821	0.006
27	51.815	33.120	49.193	29.191	55.085	30.313	0.003
28	41.516	25.821	45.408	26.943	42.678	24.137	0.002

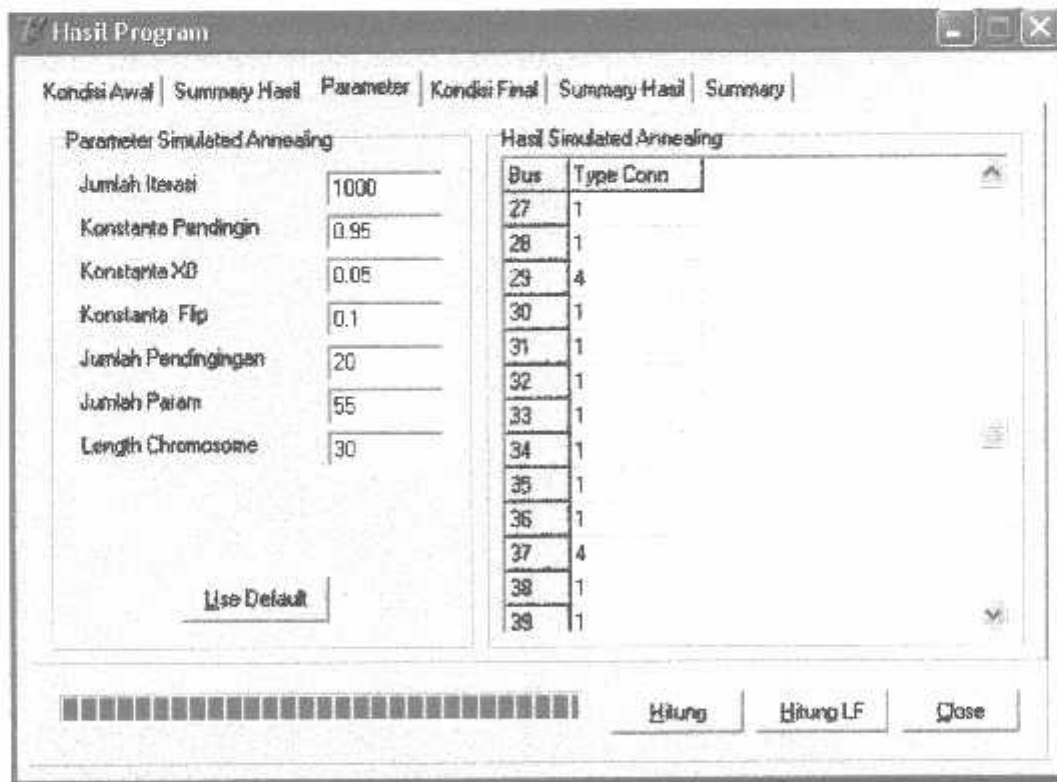
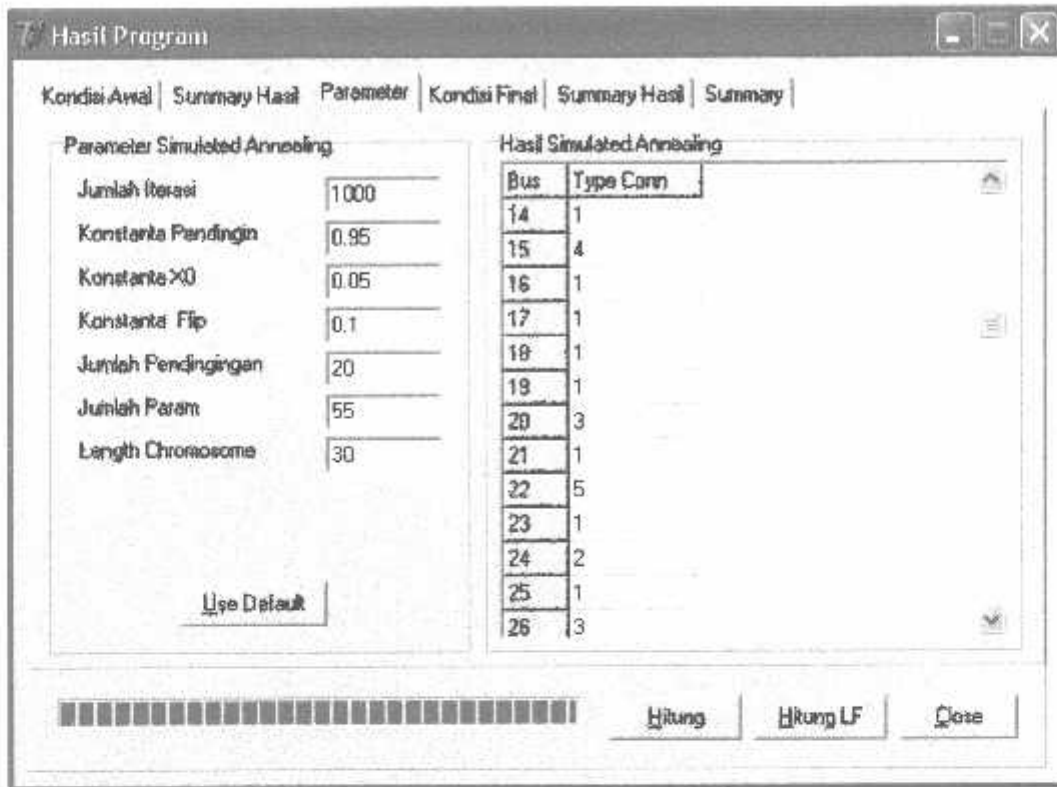
Hitung Hitung LF Close

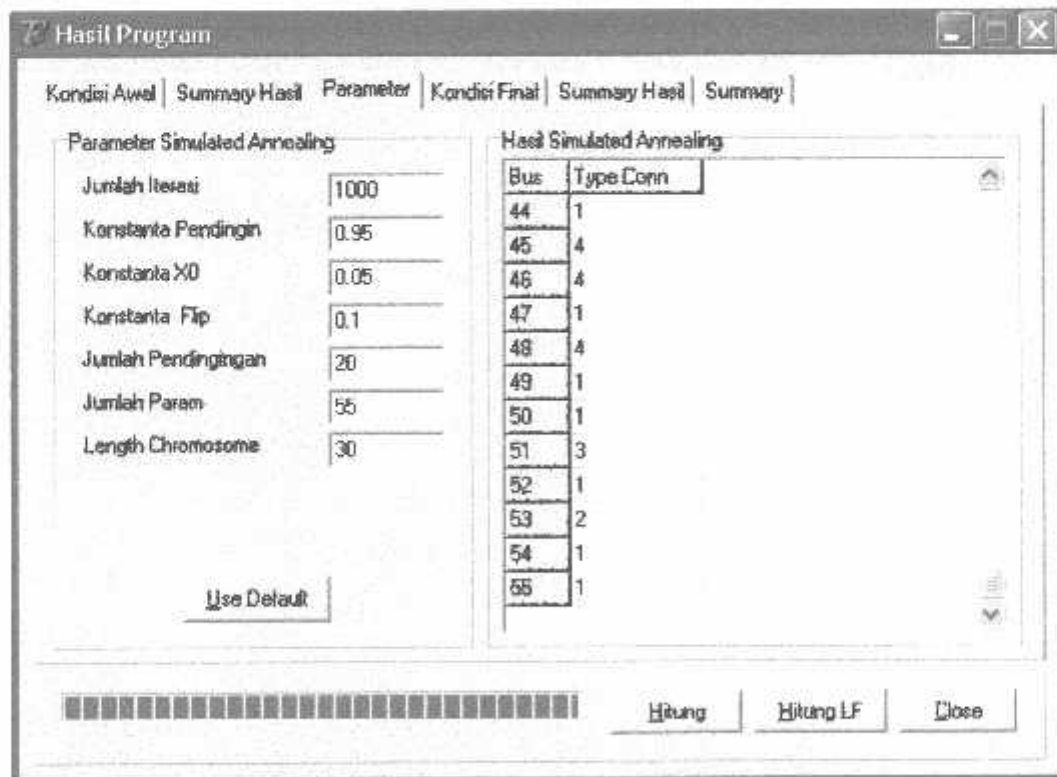
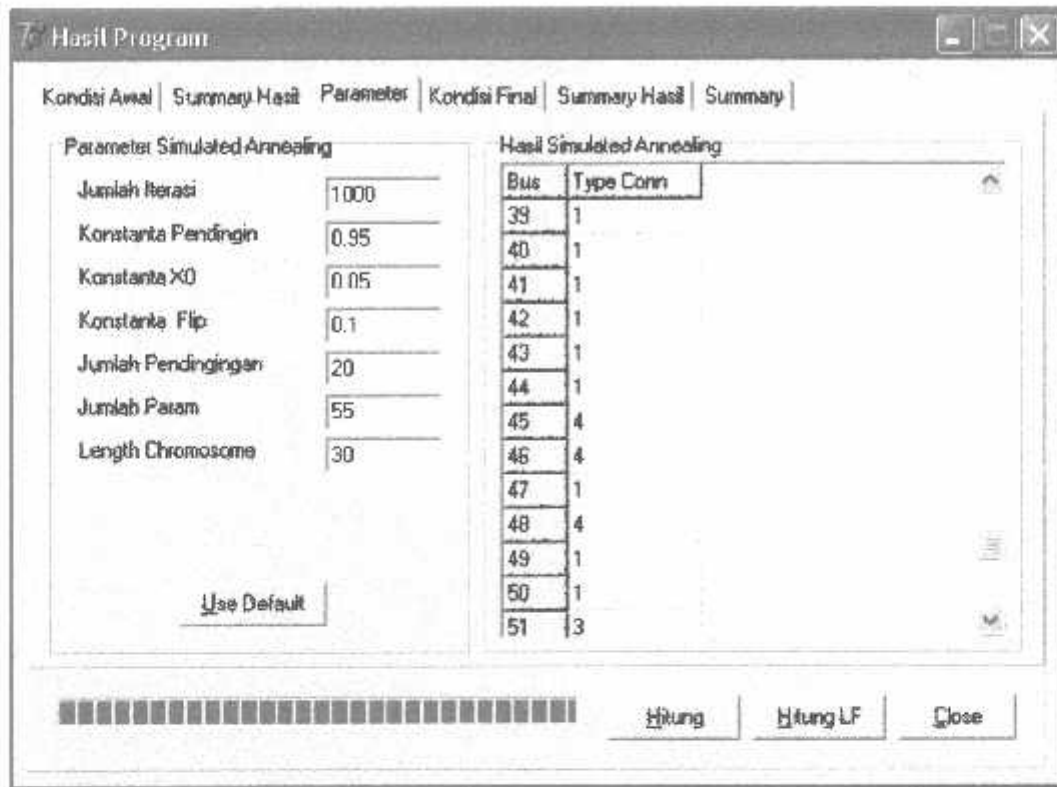
Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Sre kW	R Sim kVar	S Sre kW	S Sim kVar	T Sre kW	T Sim kVar	R Lre kW
28	41.516	25.821	45.408	26.943	42.678	24.137	0.002
29	164.051	105.676	201.710	119.719	218.164	121.178	0.029
30	152.048	97.365	181.085	107.477	187.971	104.445	0.004
31	144.392	91.622	168.778	100.157	171.584	95.891	0.019
32	87.656	55.577	104.825	62.202	103.819	58.049	0.007
33	62.977	39.743	70.005	41.540	72.680	40.754	0.005
34	32.245	20.320	38.408	22.790	36.624	20.544	0.000
35	7.649	5.726	12.298	7.297	16.379	8.532	0.000
36	31.110	19.646	32.353	19.197	36.434	20.432	0.001
37	30.732	19.422	31.596	18.748	36.056	20.208	0.001
38	25.623	16.391	31.596	18.748	31.326	17.401	0.000
39	5.379	4.378	0.757	0.449	9.757	4.603	0.000
40	6.325	4.940	11.920	7.073	12.974	6.511	0.000
41	4.344	2.705	15.514	9.205	16.001	8.308	0.000

Hitung Hitung LF Close





Tampilan Tegangan Fasa R,S,T Hasil Perhitungan Aliran Daya Setelah Penukaran Fasa

Hasil Program

Bus	R Vre pu	R Vim pu	S Vre pu	S Vim pu	T Vre pu	T Vim pu	R Sre kW
1	0.99915	-0.00065	0.99913	-0.00069	0.99911	-0.00072	1013.167
2	0.99803	-0.00149	0.99798	-0.00160	0.99793	-0.00166	968.420
3	0.99798	-0.00153	0.99793	-0.00164	0.99788	-0.00170	954.150
4	0.99730	-0.00204	0.99724	-0.00218	0.99718	-0.00227	954.830
5	0.99693	-0.00232	0.99685	-0.00249	0.99679	-0.00258	918.425
6	0.99637	-0.00190	0.99627	-0.00211	0.99621	-0.00222	893.851
7	0.99624	-0.00181	0.99615	-0.00203	0.99608	-0.00215	819.942
8	0.99591	-0.00206	0.99581	-0.00229	0.99573	-0.00242	785.097
9	0.99540	-0.00244	0.99529	-0.00270	0.99521	-0.00285	611.379
10	0.99525	-0.00254	0.99514	-0.00280	0.99506	-0.00296	563.192
11	0.99500	-0.00271	0.99488	-0.00299	0.99478	-0.00315	405.148
12	0.99462	-0.00296	0.99450	-0.00326	0.99438	-0.00344	371.118
13	0.99439	-0.00312	0.99426	-0.00343	0.99412	-0.00362	349.041
14	0.99430	-0.00318	0.99417	-0.00349	0.99403	-0.00369	319.428

Hitung Hitung LF Close

Hasil Program

Bus	R Vre pu	R Vim pu	S Vre pu	S Vim pu	T Vre pu	T Vim pu	R Sre kW
14	0.99430	-0.00318	0.99417	-0.00349	0.99403	-0.00369	319.428
15	0.99416	-0.00327	0.99403	-0.00359	0.99388	-0.00379	311.185
16	0.99379	-0.00352	0.99366	-0.00385	0.99348	-0.00408	297.823
17	0.99371	-0.00357	0.99359	-0.00390	0.99340	-0.00414	264.107
18	0.99346	-0.00374	0.99334	-0.00407	0.99313	-0.00433	251.148
19	0.99327	-0.00386	0.99317	-0.00420	0.99294	-0.00446	255.741
20	0.99323	-0.00388	0.99313	-0.00422	0.99291	-0.00449	249.221
21	0.99299	-0.00405	0.99289	-0.00439	0.99265	-0.00467	237.726
22	0.99272	-0.00422	0.99264	-0.00457	0.99239	-0.00487	231.365
23	0.99263	-0.00428	0.99255	-0.00463	0.99229	-0.00493	227.108
24	0.99259	-0.00430	0.99252	-0.00465	0.99226	-0.00496	131.365
25	0.99258	-0.00431	0.99251	-0.00466	0.99225	-0.00496	115.390
26	0.99253	-0.00434	0.99247	-0.00469	0.99220	-0.00500	100.170
27	0.99249	-0.00437	0.99242	-0.00472	0.99216	-0.00503	51.815

Hitung Hitung LF Close

Tabel 4.9
Kondisi Akhir Rugi rugi Daya Saluran Setelah Penukaran Fasa

Hasil Program

Bus	R Lre kW	R Lim kVar	S Lre kW	S Lim kVar	T Lre kW	T Lim kVar
1	1.268	0.124	1.277	0.160	1.435	0.223
2	1.586	0.154	1.625	0.203	1.821	0.284
3	0.072	0.007	0.074	0.009	0.083	0.013
4	0.940	0.082	0.955	0.118	1.066	0.167
5	0.509	0.049	0.511	0.063	0.570	0.090
6	0.270	-0.686	0.305	-0.641	0.356	-0.663
7	0.055	-0.138	0.062	-0.131	0.074	-0.138
8	0.380	0.037	0.388	0.047	0.438	0.069
9	0.457	0.046	0.445	0.054	0.499	0.076
10	0.116	0.005	0.111	0.007	0.128	0.012
11	0.146	0.006	0.146	0.009	0.184	0.018
12	0.197	0.008	0.196	0.012	0.249	0.025
13	0.117	0.004	0.116	0.007	0.151	0.015
14	0.079	0.001	0.077	0.002	0.089	0.005

Hitung Hitung LF Close

Hasil Program

Bus	R Lre kW	R Lim kVar	S Lre kW	S Lim kVar	T Lre kW	T Lim kVar
15	0.062	0.002	0.058	0.003	0.076	0.008
16	0.153	0.005	0.145	0.008	0.190	0.021
17	0.029	0.001	0.025	0.001	0.035	0.004
18	0.095	0.002	0.075	0.004	0.107	0.012
19	0.067	0.002	0.054	0.003	0.074	0.008
20	0.013	0.000	0.011	0.001	0.014	0.002
21	0.083	0.002	0.068	0.004	0.090	0.011
22	0.088	0.002	0.067	0.003	0.090	0.011
23	0.031	0.001	0.022	0.001	0.030	0.003
24	0.006	0.000	0.005	0.000	0.006	0.001
25	0.002	0.000	0.002	0.000	0.002	0.000
26	0.007	0.000	0.006	0.000	0.006	0.001
27	0.003	0.000	0.003	0.000	0.003	0.000
28	0.002	0.000	0.002	0.000	0.002	0.000

Hitung Hitung LF Close

7 Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Lre kW	R Lim kVar	S Lre kW	S Lim kVar	T Lre kW	T Lim kVar
29	0.031	0.003	0.037	0.004	0.042	0.007
30	0.004	0.000	0.006	0.001	0.006	0.001
31	0.019	0.002	0.023	0.003	0.023	0.004
32	0.007	0.001	0.009	0.001	0.009	0.002
33	0.005	0.000	0.006	0.001	0.006	0.001
34	0.000	0.000	0.001	0.000	0.000	0.000
35	0.000	-0.000	0.000	0.000	0.000	0.000
36	0.001	0.000	0.001	0.000	0.001	0.000
37	0.002	0.000	0.002	0.000	0.002	0.000
38	0.000	0.000	0.000	0.000	0.000	0.000
39	0.000	-0.000	0.000	0.000	0.000	0.000
40	0.000	-0.000	0.000	0.000	0.000	0.000
41	0.000	-0.000	0.000	0.000	0.000	0.000
42	0.000	-0.000	0.000	0.000	0.000	0.000

Hitung Hitung LF Close

7 Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Lre kW	R Lim kVar	S Lre kW	S Lim kVar	T Lre kW	T Lim kVar
43	0.007	0.000	0.004	0.000	0.007	0.001
44	0.002	0.000	0.001	0.000	0.002	0.000
45	0.002	0.000	0.001	0.000	0.002	0.000
46	0.000	-0.000	0.000	0.000	0.000	0.000
47	0.000	-0.000	0.000	0.000	0.000	0.000
48	0.000	0.000	0.000	-0.000	0.000	0.000
49	0.001	0.000	0.001	0.000	0.001	0.000
50	0.000	0.000	0.000	0.000	0.000	0.000
51	0.000	0.000	0.000	0.000	0.000	0.000
52	0.000	0.000	0.000	0.000	0.000	0.000
53	0.000	-0.000	0.000	0.000	0.000	0.000
54	0.002	0.000	0.001	0.000	0.001	0.000
55	0.001	0.000	0.000	0.000	0.001	0.000

Hitung Hitung LF Close

Kondisi Akhir Aliran Daya Setelah Penukaran Fasa

Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

Bus	R Sre kW	R Sim kVar	S Sre kW	S Sim kVar	T Sre kW	T Sim kVar	R Lre kW
1	1013.167	619.100	967.827	597.728	1097.689	631.800	1.268
2	968.420	591.464	956.226	577.782	1060.775	608.532	1.586
3	964.150	587.692	946.065	572.885	1051.852	603.177	0.072
4	954.830	581.539	935.702	566.034	1036.217	593.080	0.940
5	918.425	559.623	891.629	539.506	988.599	564.370	0.509
6	893.851	545.549	871.729	528.193	959.169	547.336	0.270
7	819.942	499.421	810.850	492.159	901.091	515.345	0.055
8	785.097	478.524	771.064	468.221	861.765	491.693	0.380
9	611.379	369.136	578.673	351.303	643.402	370.004	0.457
10	563.192	340.520	525.545	319.660	595.102	341.261	0.116
11	405.148	246.646	391.047	239.720	474.045	269.369	0.146
12	371.118	226.375	356.914	219.331	436.938	247.143	0.197
13	349.041	213.248	332.651	204.855	417.089	235.219	0.117
14	319.428	195.702	299.716	185.286	375.526	210.490	0.039

Hitung Hitung LF Close

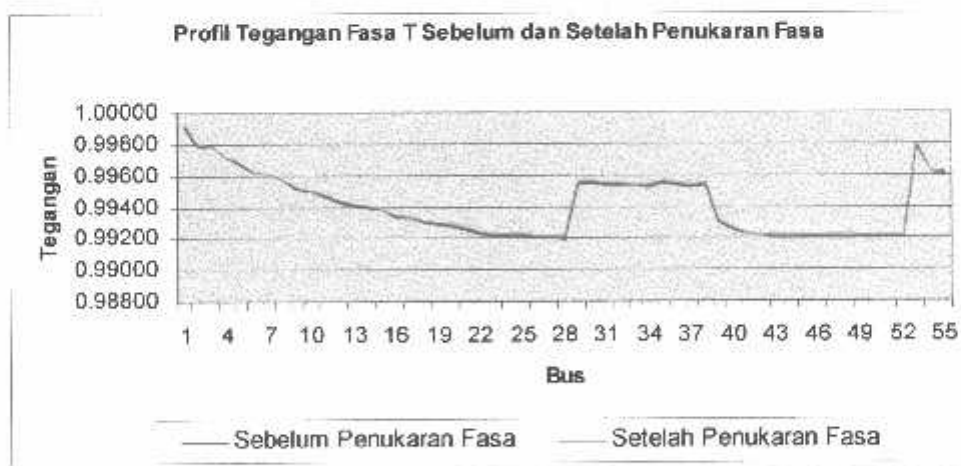
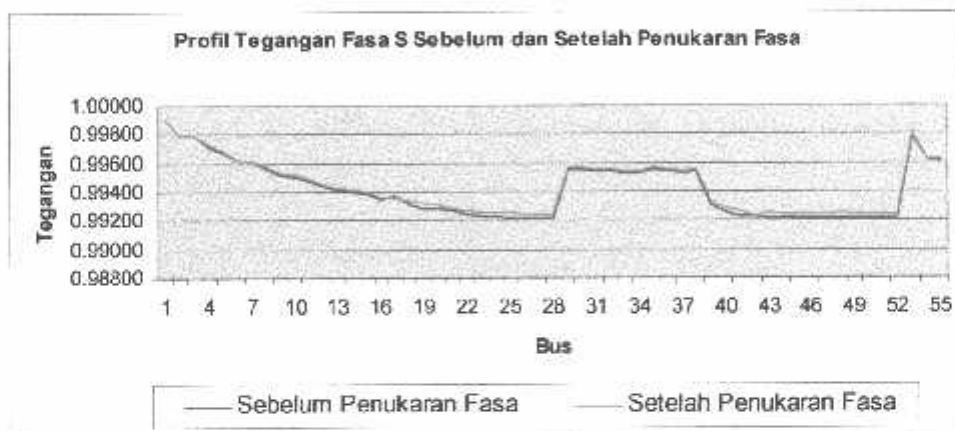
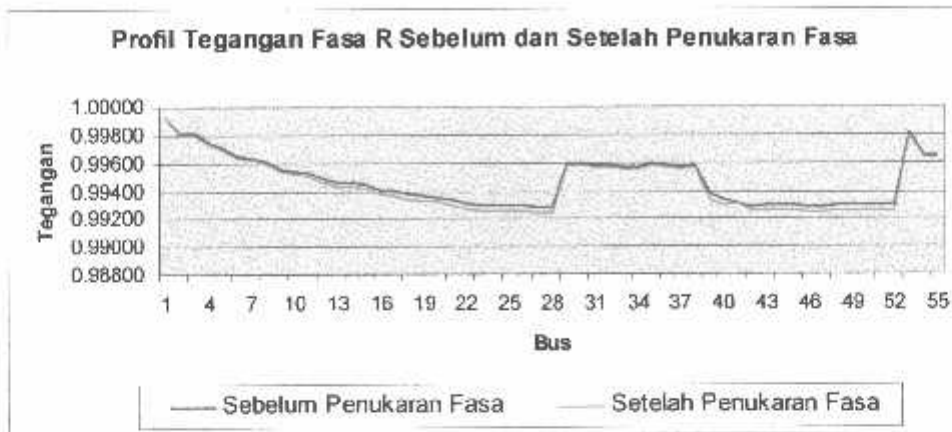
Hasil Program

Kondisi Awal | Summary Hasil | Parameter | Kondisi Final | Summary Hasil | Summary

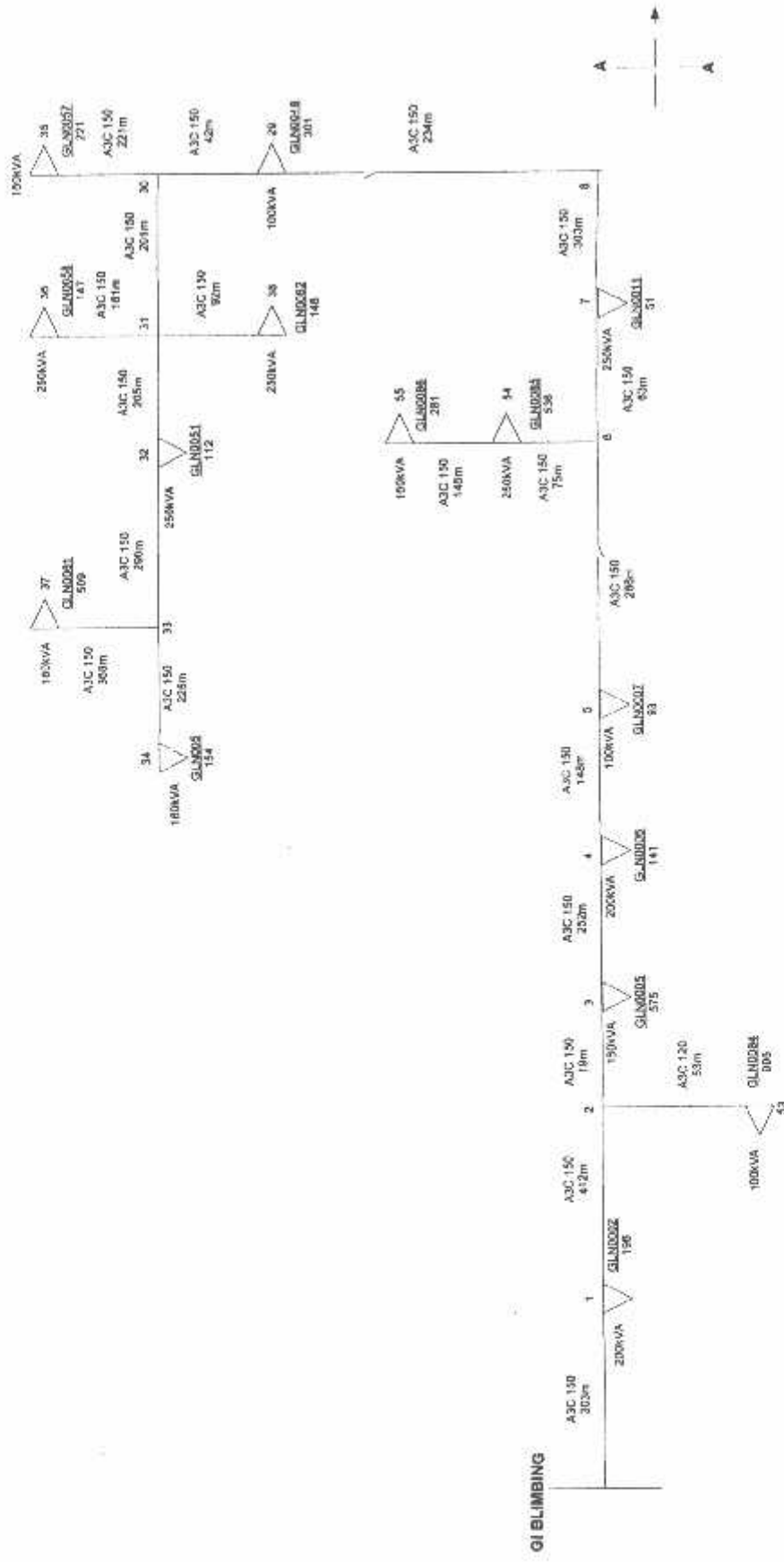
Bus	R Sre kW	R Sim kVar	S Sre kW	S Sim kVar	T Sre kW	T Sim kVar	R Lre kW
15	311.185	190.821	296.260	177.262	360.060	201.207	0.062
16	291.823	179.230	259.662	167.363	340.762	189.585	0.159
17	264.107	162.816	226.137	141.519	306.204	169.012	0.029
18	261.148	161.046	211.540	132.805	285.995	156.905	0.095
19	255.741	156.607	210.762	132.306	276.206	152.235	0.067
20	249.221	152.779	201.298	126.684	265.118	145.600	0.013
21	237.726	147.090	197.596	123.260	253.728	138.782	0.083
22	231.365	142.072	185.651	116.125	240.715	132.188	0.086
23	227.108	138.340	170.128	106.899	224.702	123.853	0.031
24	131.365	81.516	115.309	69.615	128.337	71.413	0.006
25	115.390	70.849	96.388	58.387	108.173	60.633	0.002
26	100.170	60.627	85.034	51.646	94.818	53.892	0.007
27	51.815	33.120	49.193	29.191	55.085	30.313	0.003
28	41.516	25.821	45.408	26.942	42.670	24.177	0.002

Hitung Hitung LF Close

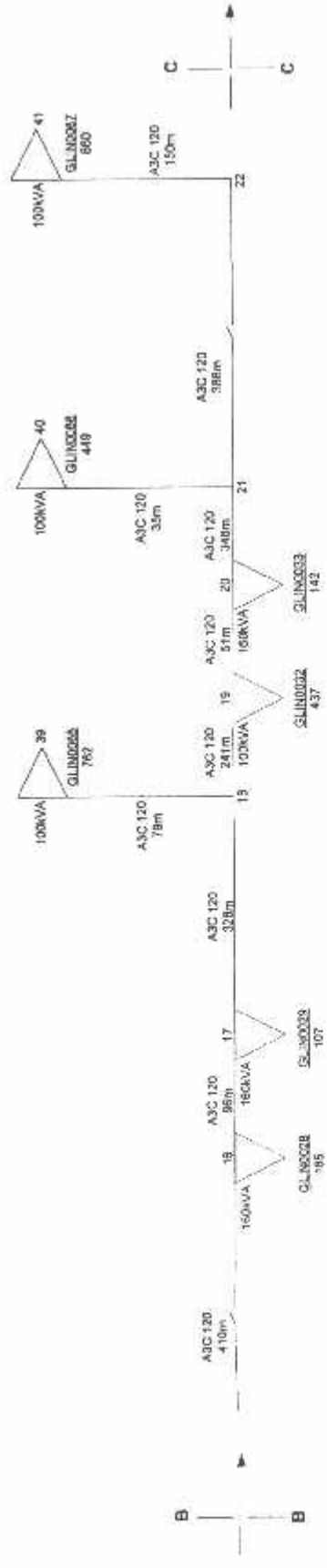
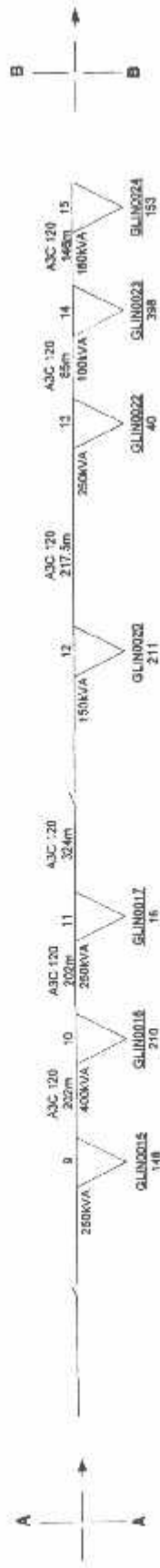
Profil Tegangan Sebelum dan Setelah Penukaran Fasa

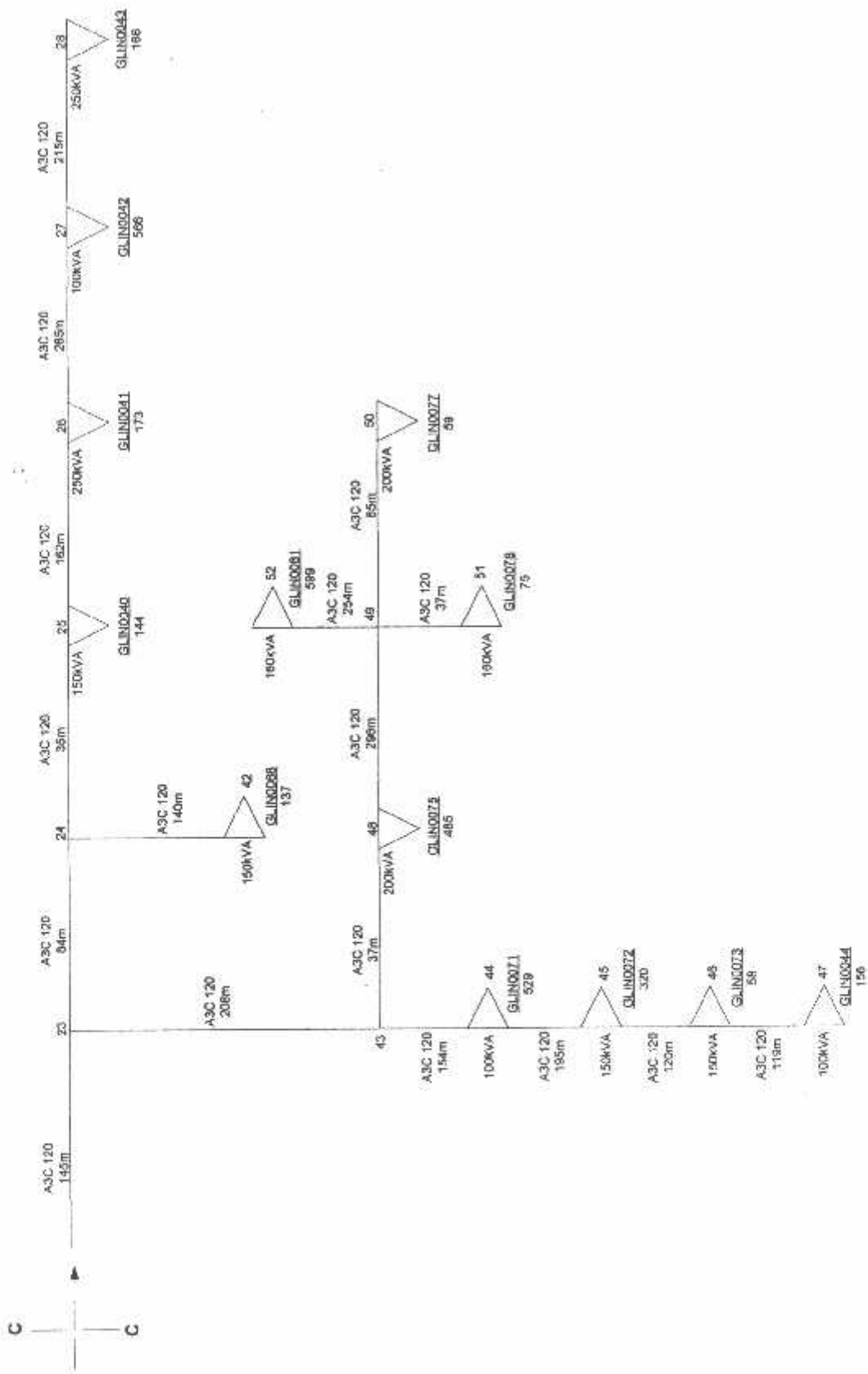


PENYULANG GLINTUNG



GAMBAR 4-3
SINGLE LINE DIAGRAM SISTEM 20 KV PENYULANG GLINTUNG





```

unit uFD3Phasa:

interface

uses Math,uUtils,SysUtils;

type
  TComplex=record
    re,im:double;
  end;

  TSaluran=record
    a,b,c:TComplex;
  end;

  CArr1a=Array of TComplex;
  CArr2a=Array of Array of TComplex;
  CArr3a=Array of Array of Array of TComplex;
  SArr=Array of TSaluran;

//unit Data
  TSaluran1=record
    No,Nbhind,LnAHead1,LnAHead2,LnAHead3:integer;
    Panjang,GMR,R:double;
    Zs:Array[1..4,1..4] of TComplex;
  end;

  TPhasa=record
    R,S,T:TComplex;
  end;

  TPhasaDouble=record
    R,S,T:double;
  end;

  TBus=record
    No:integer;
    Load:TPhasa;
  end;

  SArr=Array of TSaluran1;
  BArr=Array of TBus;
  PArr=Array of TPhasa;
  PArr2=Array of Array of TPhasa;

procedure HitungFD3Phasa(const rNbus:integer;
  const rVbase,rVKonst,rPbase,rPkonst:double;
  const rBus,rBranch:dArr2;
  var rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss:dArr2;
  var rIterasi:byte);overload;

procedure HitungFD3Phasa(var rV,rI,rSa,rSL,rSumGen,rSumLoad,
  rSumLoss:dArr2;
  var rIterasi:byte);overload;

function doHitung(const rChrom:dArr1):double;

```

```

procedure doHitungAkhir(const rChrom:dArr1;
  var rTypConn:iArr1;
  var rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss:dArr2;
  var rIterasi:byte);

var N:integer;
  terminal:iArr1;
  Bus:BArr;
  Vr,Ir,Sa,SL:PArr;
  Za:SArr;
  Pbase,Vbase,KonstP,KonstV,Zbase,Ibase:double;

implementation

function RealToStr(Num:double;Pecahan:byte):String;
var Hasil:String;
  le:byte;
begin
  le:=sizeof(Num);
  Str(Num:le:Pecahan,Hasil);
  Result:=Hasil;
end;

function StrToReal(Huruf:string):double;
var Temp:double;
  Code:integer;
begin
  val(Huruf,Temp,Code);
  Result:=Temp;
end;

function Complex(re,im:double):TComplex;
begin
  result.re:=re;
  result.im:=im;
end;

function CxRe(z:TComplex):double;
begin
  result:=z.re;
end;

function CxIm(Z:TComplex):double;
begin
  result:=z.im;
end;

function CxAbs(Z:TComplex):Double;
begin
  result:=sqrt(sqrt(Z.re)+sqrt(Z.im));
end;

function CxArg(Z:TComplex):Double;
var temp:double;
begin

```

```

temp:=0;
result:=0;
if Z.re<>0 then
  temp:=ArcTan(Abs(Z.im/Z.re))
else
  if Z.im<>0 then
    result:=Pi/2*Z.im/abs(Z.im);

if Z.im=0 then
  if Z.re>0 then
    result:=0
  else
    result:=Pi;

if Z.re> 0 then
  if Z.im> 0 then result:=temp
  else if z.im< 0 then
    result:=-temp;

if Z.re<0 then
  if Z.im>0 then
    result:=Pi-temp
  else if z.im<0 then
    result:=temp-Pi;

if ((Z.re=0) and (Z.im=0)) then result:= 0;
end;

function PolarToCxRad(r,theta:double):TComplex;
begin
  result.re:=r*cos(theta);
  result.im:=r*sin(theta);
end;

function PolarToCxDeg(r,theta:double):TComplex;
begin
  theta:=theta*Pi/180;
  result.re:=r*cos(theta);
  result.im:=r*sin(theta);
end;

function CxTambah(x,y:TComplex):TComplex;
begin
  result.re:=x.re+y.re;
  result.im:=x.im+y.im;
end;

function CxKurang(x,y:TComplex):TComplex;
begin
  result.re:=x.re-y.re;
  result.im:=x.im-y.im;
end;

function CxKali(x,y:TComplex):TComplex;
begin
  result.re:=x.re*y.re-x.im*y.im;

```

```

    result.im:=x.re*y.im+x.im*y.re;
end;

function CxBagi(x,y:TComplex):TComplex;
begin
    result.re:=(x.re*y.re+x.im*y.im)/(sqr(y.re)+sqr(y.im));
    result.im:=(x.im*y.re-x.re*y.im)/(sqr(y.re)+sqr(y.im));
end;

function CxPow(Z:TComplex;n:Integer):TComplex;
begin
    result.re:=Power(CxAbs(Z),n)*Cos(n*CxArg(Z));
    result.im:=Power(CxAbs(Z),n)*Sin(n*CxArg(Z));
end;

function CxSqrt(Z:TComplex):TComplex;
begin
    result.re:=Sqrt(CxAbs(Z))*Cos(0.5*CxArg(Z));
    result.im:=Sqrt(CxAbs(Z))*Sin(0.5*CxArg(Z));
end;

function CxExp(Z:TComplex):TComplex;
begin
    result.re:=Exp(Z.re)*Cos(Z.im);
    result.im:=Exp(Z.re)*Sin(Z.im);
end;

function CxLn(Z:TComplex):TComplex;
begin
    result.re:= Ln(CxAbs(Z));
    result.im:= CxArg(Z);
end;

function CxCos(Z:TComplex):TComplex;
var
    iZ,miZ,Temp:TComplex;
begin
    iZ.re:= -Z.im;
    iZ.im:= Z.re;
    miZ.re:= Z.im;
    miZ.im:= -Z.re;
    Temp := CxTambah(CxExp(iZ),CxExp(miZ));
    result.re:=Temp.re/2;
    result.im:=Temp.im/2;
end;

function cSin(Z:TComplex):TComplex;
var iZ,miZ,Temp,i:TComplex;
begin
    iZ.re:=-Z.im;
    iZ.im:=Z.re;
    miZ.re:=Z.im;
    miZ.im:=-Z.re;
    i.re:=0;
    i.im:=1;
    Temp:=CxKurang(CxExp(iZ),CxExp(miZ));

```

```

Temp.re:=Temp.re/2;
Temp.im:=Temp.im/2;
result:=CxBagi(Temp,l);
end;

function CxPangkat(z:TComplex;pangkat:double):TComplex;
var theta,sum:double;
begin
theta:=arctan(z.re/z.im);
sum:=exp((pangkat/2)*ln(sqrt(z.re)+sqrt(z.im)));
result.re:=sum*cos(pangkat*theta);
result.im:=sum*sin(pangkat*theta);
end;

function CxConjg(z:TComplex):TComplex;
begin
result.re:=z.re;
result.im:=-z.im;
end;

function CxNegatif(z:TComplex):TComplex;
begin
result.re:=-z.re;
result.im:=-z.im;
end;

Procedure CxArray1(N:byte;rea,ima:dArr1;
var Z:CArr1a);
var i:byte;
begin
for i:=1 to N do
begin
z[i]:=Complex(rea[i],ima[i]);
end;
end;

Procedure CxArray2(N:byte;rea,ima:dArr2;
var Z:CArr2a);
var i,j:byte;
begin
for i:=1 to N do
begin
for j:=1 to N do
begin
z[i,j]:=Complex(rea[i,j],ima[i,j]);
end;
end;
end;

Procedure CxArr2re(N:byte;rea:dArr2;
var z:CArr2a);
var i,j:byte;
begin
for i:=1 to N do
begin
for j:=1 to N do

```

```

begin
  z[i,j]:=Complex(rea[i,j],0);
end;
end;
end;

Procedure CxArr2im(N:byte;ima:dArr2;
  var z:CArr2a);
var i,j:byte;
begin
  for i:=1 to N do
  begin
    for j:=1 to N do
    begin
      z[i,j]:=Complex(0,ima[i,j]);
    end;
  end;
end;

function CxToStr(z:TComplex;len:byte):string;
begin
  if z.im<0 then
  begin
    result:=RealToStr(z.re,len)+' - j'+RealToStr(Abs(z.im),len);
  end
  else
  begin
    result:=RealToStr(z.re,len)+' + j'+RealToStr(z.im,len);
  end;
end;

//Unit Matrik
Procedure TransposeMatrik(N:integer;MA:dArr2;
  var MB:dArr2);
var i,j:integer;
begin
  SetLength(MB,N+1,N+1);
  for i:=1 to N do
  begin
    for j:=1 to N do
    begin
      MB[j,i]:=MA[i,j];
    end;
  end;
end;

Procedure InversMatrik(N:byte;MA:dArr2;
  var MB:dArr2);
var Konst,DE,AE:double;
  i,j,k:byte;
begin
  SetLength(MB,N+1,N+1);
  for i:=1 to N do
  begin
    for j:=1 to N do
    begin

```

```

    MB[i,j]:=MA[i,j];
  end;
end;
for i:=1 to N do
begin
  if MB[i,i]=0 then
  begin
    for k:=i+1 to N do
    begin
      if MB[k,i] <> 0 then
      begin
        for j:=1 to N do
        begin
          Konst:=MB[i,j];
          MB[i,j]:=MB[k,j];
          MB[k,j]:=Konst;
        end;
      end;
    end;
  end;
  DE:=MB[i,i];
  MB[i,i]:=1;
  for j:=1 to N do
  begin
    MB[i,j]:=MB[i,j]/DE;
  end;
  for k:=1 to N do
  begin
    if k <> i then
    begin
      AE:=MB[k,i];
      MB[k,i]:=0;
      for j:=1 to N do
      begin
        MB[k,j]:=MB[k,j]-AE*MB[i,j];
      end;
    end;
  end;
end;
end;

```

```

Procedure InversMatrikInf(N:byte;MA:iArr2;
    var MB:dArr2);
var MAA:dArr2;
    i,j:byte;
begin
  SetLength(MB,N+1,N+1);
  SetLength(MAA,N+1,N+1);
  for i:=1 to N do
  begin
    for j:=1 to N do
    begin
      MAA[i,j]:=MA[i,j];
    end;
  end;
  InversMatrik(N,MAA,MB);

```

```

end;

Procedure MatrikA(N:integer;Z:CArr2a;
  var B:dArr2);
var i,j,k:integer;
  A:dArr2;
  Zx:double;
begin
  SetLength(A,N+1,N+1);
  SetLength(B,N+1,N+1);
  for i:=0 to N do
  begin
    for j:=0 to N do
    begin
      for k:=i to N do
      begin
        Zx:=Cxim(Z[j,k]);
        if Zx<>0 then
        begin
          if k=i then
          begin
            A[i,k]:=-1;
          end
          else if j=i then
          begin
            A[i,k]:=1;
          end
          else
          begin
            A[i,k]:=0;
          end;
        end;
      end;
    end;
  end;
  InversMatrik(N,A,B);
  for i:=1 to N do
  begin
    for j:=1 to N do
    begin
      B[i,j]:=-B[i,j];
    end;
  end;
end;

Procedure CxKaliMatrikN1(N:integer;MA:CArr2a;MB:CArr1a;
  var MC:CArr1a);
var i,j:byte;
  temp:Tcomplex;
begin
  SetLength(MC,N+1);
  for i:=1 to N do
  begin
    MC[i]:=Complex(0,0);
    for j:=1 to N do
    begin

```

```

    temp:=CxKali(MA[i,j],MB[j]);
    MC[i]:=Cxtambah(MC[i],temp);
end;
end;
end;

Procedure CxKaliMatrikNN(N:byte;MA,MB:CArr2a;
var MC:CArr2a);
var i,j,k:byte;
    temp:TComplex;
begin
    SetLength(MC,N+1,N+1);
    for i:=1 to N do
    begin
        for j:=1 to N do
        begin
            MC[i,j]:=Complex(0,0);
            for k:=1 to N do
            begin
                temp:=CxKali(MA[i,k],MB[k,j]);
                MC[i,j]:=Cxtambah(MC[i,j],temp);
            end;
        end;
    end;
end;
end;

Function PhTambah(Var1,Var2:TPhasa):TPhasa;
begin
    Result.R:=Cxtambah(Var1.R,Var2.R);
    Result.S:=Cxtambah(Var1.S,Var2.S);
    Result.T:=Cxtambah(Var1.T,Var2.T);
end;

Function PhKali(Var1,Var2:TPhasa):TPhasa;
begin
    Result.R:=CxKali(Var1.R,Var2.R);
    Result.S:=CxKali(Var1.S,Var2.S);
    Result.T:=CxKali(Var1.T,Var2.T);
end;

Function PhBagi(Var1,Var2:TPhasa):TPhasa;
begin
    Result.R:=CxBagi(Var1.R,Var2.R);
    Result.S:=CxBagi(Var1.S,Var2.S);
    Result.T:=CxBagi(Var1.T,Var2.T);
end;

Function PhKurang(Var1,Var2:TPhasa):TPhasa;
begin
    Result.R:=CxKurang(Var1.R,Var2.R);
    Result.S:=CxKurang(Var1.S,Var2.S);
    Result.T:=CxKurang(Var1.T,Var2.T);
end;

Function PhAbs(Var1:TPhasa):TPhasaDouble;
begin

```

```

Result.R:=CxAbs(Var1.R);
Result.S:=CxAbs(Var1.S);
Result.T:=CxAbs(Var1.T);
end;

Function PhConj(Var1:TPhasa):TPhasa;
begin
Result.R:=CxConj(Var1.R);
Result.S:=CxConj(Var1.S);
Result.T:=CxConj(Var1.T);
end;

Procedure PhKaliMatrikN1(N:integer;MA:dArr2;MB:PArr;
var MC:PArr);
var i,j:integer;
MX:CArr2a;
MH,MS:CArr1a;
begin
SetLength(MX,N+1,N+1);
SetLength(MH,N+1);
SetLength(MS,N+1);
SetLength(MC,N+1);
for i:=1 to N do
begin
for j:=1 to N do
begin
MX[i,j]:=Complex(MA[i,j],0);
end;
end;
for i:=1 to N do
begin
MS[i]:=MB[i].R;
end;
CxKaliMatrikN1(N,MX,MS,MH);
for i:=1 to N do
begin
MC[i].R:=MH[i];
end;
for i:=1 to N do
begin
MS[i]:=MB[i].S;
end;
CxKaliMatrikN1(N,MX,MS,MH);
for i:=1 to N do
begin
MC[i].S:=MH[i];
end;
for i:=1 to N do
begin
MS[i]:=MB[i].T;
end;
CxKaliMatrikN1(N,MX,MS,MH);
for i:=1 to N do
begin
MC[i].T:=MH[i];
end;
end;

```

```

end;

Procedure ImpedansiMatrik(N:integer;D:dArr2;
  var Za:SArr);
var i,j,k:integer;
  tmpRe,tmpIm:double;
  temp1:TComplex;
begin
  //modified form of Carson's Equation
  for k:=1 to N do
  begin
    for i:=1 to 4 do
    begin
      for j:=1 to 4 do
      begin
        if i=j then
        begin
          tmpRe:=Za[k].R+0.0953;
          tmpIm:=0.12134*(ln(1/Za[k].GMR)+7.934);
          Za[k].Zs[i,j]:=Complex(tmpRe,tmpIm);
        end
        else
        begin
          tmpRe:=0.0953;
          tmpIm:=0.12134*(ln(1/D[i,j])+7.934);
          Za[k].Zs[i,j]:=Complex(tmpRe,tmpIm);
        end;
      end;
    end;
  end;
  //This matrix is reduced 3X3 by using Kron reduction
  for k:=1 to N do
  begin
    for l:=1 to 3 do
    begin
      for j:=1 to 3 do
      begin
        temp1:=CxKali(Za[k].Zs[i,4],Za[k].Zs[4,j]);
        temp1:=CxBagi(temp1,Za[k].Zs[4,4]);
        Za[k].Zs[i,j]:=CxKurang(Za[k].Zs[i,j],temp1);
      end;
    end;
  end;
end;

Procedure JarakKawat(var D:dArr2);
var ab,bc,an,bn,nn:double;
begin
  ab:=0.9;
  bc:=0.75;
  an:=0.45;
  bn:=0.45;
  nn:=1;
  SetLength(D,5,5);
  D[2,1]:=ab;
  D[1,2]:=D[2,1];

```

```

D[3,1]:=ab+bc;
D[1,3]:=D[3,1];
D[4,1]:=sqrt(sqrt(an)+sqrt(nn));
D[1,4]:=D[4,1];
D[3,2]:=bc;
D[2,3]:=D[3,2];
D[4,2]:=sqrt(sqrt(bn)+sqrt(nn));
D[2,4]:=D[4,2];
D[4,3]:=sqrt(sqrt(bn+bc)+sqrt(nn));
D[3,4]:=D[4,3];
end;

Procedure ImpedansiSaluranKm(N:integer;
var Za:SArr);
var i,j,k:integer;
    mile,ft:double;
    temp:TComplex;
    D:dArr2;
begin
    mile:=1609;
    ft:=0.3048;
    JarakKawat(D);
    for i:=1 to N do
    begin
        Za[i].GMR:=Za[i].GMR/ft;
        Za[i].R:=Za[i].R*mile;
    end;
    for i:=1 to 4 do
    begin
        for j:=1 to 4 do
        begin
            D[i,j]:=D[i,j]/ft;
        end;
    end;
    ImpedansiMatrik(N,D,Za);
    for k:=1 to N do
    begin
        temp:=Complex(Za[k].Panjang/mile,0);
        for i:=1 to 3 do
        begin
            for j:=1 to 3 do
            begin
                Za[k].Zs[i,j]:=CxKali(Za[k].Zs[i,j],temp);
            end;
        end;
    end;
end;

//Unit Loadflow FastDecoupled 3 Phasa
Procedure MatrikAm(N:integer,Za:SArr;
var Am:iArr2);
var i,j,k,ia,ja:integer;
    Zs:iArr2;
begin
    SetLength(Am,N+1,N+1);
    SetLength(Zs,N+1,N+1);

```

```

    if Am[i,j]<>0 then
    begin
        s:=s+1;
    end;
end;
if s=0 then
begin
    Trm[sa]:=i;
    sa:=sa+1;
end
else if s>1 then
begin
    CbTrm[sb]:=i;
    sb:=sb+1;
end;
end
else
begin
    Trm[sa]:=i;
end;
end;
Nt:=sa;
Nc:=sb-1;
end;

```

```

Procedure CarTerminal(N,Nt,Node:integer;Am:iArr2;Trm:iArr1;
    var endNode:integer);

```

```

var j,k,s:integer;
begin
    for j:=Node to N do
    begin
        s:=0;
        for k:=j+1 to N do
        begin
            if Am[j,k]=1 then
            begin
                s:=s+1;
                break;
            end;
        end;
        if s=0 then
        begin
            for k:=1 to Nt do
            begin
                if j=Trm[k] then
                begin
                    endNode:=j;
                end;
            end;
            break;
        end;
    end;
end;

```

```

Procedure IndexTerminal(Node,Nt:integer;Trm:iArr1;
    var idxTerm:integer);

```

```

var i:integer;
begin
  for i:=1 to Nt do
  begin
    if Node=Trm[i] then
    begin
      idxTerm:=Nt-i+1;
    end;
  end;
end;

Procedure MatrikJaqobian(N,Nt,Nc:integer;Am:iArr2;Trm,CbTrm:iArr1;
  var MxJaq:iArr2);
var i,j,ia,ix,jx,ja:integer;
begin
  SetLength(MxJaq,Nt+1,Nt+1);
  for i:=Nc downto 1 do
  begin
    ia:=CbTrm[i];
    CariTerminal(N,Nt,ia,Am,Trm,ja);
    IndexTerminal(ja,Nt,Trm,ix);
    for j:=N downto ia do
    begin
      if Am[ia,j]=1 then
      begin
        CariTerminal(N,Nt,j,Am,Trm,ja);
        IndexTerminal(ja,Nt,Trm,jx);
        if {x<>}x then
        begin
          MxJaq[jx,ix]:=-1;
        end;
      end;
    end;
  end;
  for i:=1 to Nt do
  begin
    MxJaq[i,i]:=1;
  end;
end;

Function DropTegangan(Node:integer;Za:SArr;Ir:PArr):TPhasa;
var i,j:integer;
  B,C:CArr1a;
  A:CArr2a;
begin
  SetLength(A,4,4);
  SetLength(B,4);
  SetLength(C,4);
  for i:=1 to 3 do
  begin
    for j:=1 to 3 do
    begin
      A[i,j]:=Za[Node].Zs[i,j];
    end;
  end;
  B[1]:=Ir[Node].R;

```

```

B[2]:=Irf(Node).S;
B[3]:=Irf(Node).T;
CxKaliMatrikN1(3,A,B,C);
Result.R:=C[1];
Result.S:=C[2];
Result.T:=C[3];
end;

```

```

Procedure HitungTegNode(N:integer;Am:iArr2;Za:SArr;Bus:BArr;
var Vr,Ir:PArr);
var i,j:integer;
dV,dl:TPhasa;
begin
for i:=N downto 1 do
begin
Vr[i]:=Vr[i];
Irf[i].R:=Complex(0,0);
Irf[i].S:=Complex(0,0);
Irf[i].T:=Complex(0,0);
if i<>N then
begin
for j:=i+1 to N do
begin
if Am[i,j]=1 then
begin
dV:=DropTegangan(j,Za,Ir);
Vr[i]:=PhTambah(Vr[j],dV);
break;
end;
end;
for j:=i+1 to N do
begin
if Am[i,j]=1 then
begin
Irf[i]:=PhTambah(Irf[i],Irf[j]);
end;
end;
end;
dl:=PhConj(Phbagi(Bus[i].Load,Vr[i]));
Irf[i]:=PhTambah(Irf[i],dl);
end;
end;
end;

```

```

Procedure HitungMxFs(N:integer;Nt,Nc:byte;Am:iArr2;Trr,CbTrr:iArr1;
var Vr,Ir:PArr;
var MxFs:PArr);
var i,j,ia,ix,jx,ja:integer;
dV,Vs:TPhasa;
begin
SetLength(MxFs,Nt+1);
Vs.R:=Complex(1,0);
Vs.S:=Complex(1,0);
Vs.T:=Complex(1,0);
for i:=Nc downto 1 do
begin
ia:=CbTrr[i];

```

```

CariTerminal(N,Nt,ia,Am,Trm,ja);
IndexTerminal(ja,Nt,Trm,ix);
for j:=N downto ia do
begin
  if Am[ia,j]=1 then
  begin
    CariTerminal(N,Nt,j,Am,Trm,ja);
    IndexTerminal(ja,Nt,Trm,jx);
    if ix<>jx then
    begin
      dV:=DropTegangan(j,Za,Ir);
      dV:=PhTambah(Vr[j],dV);
      MxFs[jx]:=PhKurang(dV,Vr[ia]);
    end;
  end;
end;
end;
dV:=DropTegangan(1,Za,Ir);
dV:=PhTambah(Vr[1],dV);
MxFs[Nt]:=PhKurang(dV,Vs);
for i:=1 to Nt do
begin
  MxFs[i].R:=Complex(CxRe(MxFs[i].R)*-1,CxIm(MxFs[i].R)*-1);
  MxFs[i].S:=Complex(CxRe(MxFs[i].S)*-1,CxIm(MxFs[i].S)*-1);
  MxFs[i].T:=Complex(CxRe(MxFs[i].T)*-1,CxIm(MxFs[i].T)*-1);
end;
end;

Procedure CekKonvergen(Nt:byte;MxFs:PArr;tol:double;
  var Konvergen:Boolean);
var i:integer;
  err:TPhasaDouble;
begin
  Konvergen:=true;
  for i:=1 to Nt do
  begin
    err:=PhAbs(MxFs[i]);
    if err.R>tol then
    begin
      Konvergen:=false;
    end
    else if err.S>tol then
    begin
      Konvergen:=false;
    end
    else if err.T>tol then
    begin
      Konvergen:=false;
    end;
  end;
end;

Procedure UpdateTeg(Nt:byte;InvJaq;dArr2;MxFs:PArr;Trm:iArr1;
  var Vr:PArr);
var i,ia:byte;
  dVr:PArr;

```

```

begin
  SetLength(dVr,Nt+1);
  PhKaliMatrikN1(Nt,InvJaq,MxFs,dVr);
  for i:=1 to Nt do
  begin
    ia:=Trm[Nt-i+1];
    Vr[ia]:=PhTambah(Vr[ia],dVr[i]);
  end;
end;

Procedure AliranDaya(N:integer;Vr,Ir:PArr;
  var Sa:PArr);
var i:integer;
begin
  SetLength(Sa,N+1);
  for i:=1 to N do
  begin
    Sa[i]:=PhKali(Vr[i],PhConj(Ir[i]));
  end;
end;

Procedure RugiSaluran(N:integer,Za:SArr,Ir:PArr;
  var SL:PArr);
var i:integer;
    temp:TPhasa;
begin
  SetLength(SL,N+1);
  for i:=1 to N do
  begin
    temp:=DropTegangan(i,Za,Ir);
    SL[i]:=PhKali(temp,Ir[i]);
  end;
end;

Procedure FastDLFlow(N:integer;Za:SArr;Bus:BArr;
  var iterasi:byte;
  var Vr,Ir,Sa,SL:PArr);
var Nc,Nt:byte;
    tol:double;
    Am,MxJaq,iArr2;
    InvJaq,dArr2;
    Trm,CbTrm:iArr1;
    MxFs:PArr;
    Konvergen:boolean;
    Label Mulai;
begin
  SetLength(Ir,N+1);
  tol:=0.000001;
  iterasi:=0;
  MatrikAm(N,Za,Am);
  IdentifikasiNode(N,Am,Trm,CbTrm,Nt,Nc);
  MatrikJacobian(N,Nt,Nc,Am,Trm,CbTrm,MxJaq);
  InversMatrikInt(Nt,MxJaq,InvJaq);
Mulai:
  HitungTegNode(N,Am,Za,Bus,Vr,Ir);
  HitungMxFs(N,Nt,Nc,Am,Trm,CbTrm,Vr,Ir,MxFs);

```

```

CekKonvergen(Nt,MxFs,tol,Konvergen);
if Konvergen=false then
begin
  UpdateTeg(Nt,InvJaq,MxFs,Trm,Vr);
  iterasi:=iterasi+1;
  goto Mulai;
end;
AliranDaya(N,Vr,Ir,Sa);
RugiSaluran(N,Za,Ir,SL);
end;

procedure DecodeData(const rNbus:integer;
  const rVbase,rVKonst,rPbase,rPKonst:double;
  const rBus,rBranch:dArr2);
var Nterm,i:integer;
begin
  N:=rNbus;
  Vbase:=rVbase;
  KonstV:=rVKonst;
  Pbase:=rPbase;
  KonstP:=rPKonst;
  Zbase:=sqr(Vbase*KonstV)/(Pbase*KonstP);
  lbase:=Vbase*KonstV/Zbase;
  SetLength(Bus,N+1);
  SetLength(Vr,N+1);
  SetLength(Za,N+1);
  for i:=1 to N do
  begin
    Bus[i].No:=round(rBus[i,1]);
    Bus[i].Load.R:=complex(rBus[i,2]/Pbase,rBus[i,3]/Pbase);
    Bus[i].Load.S:=complex(rBus[i,4]/Pbase,rBus[i,5]/Pbase);
    Bus[i].Load.T:=complex(rBus[i,6]/Pbase,rBus[i,7]/Pbase);
  end;
  for i:=0 to N do
  begin
    Vr[i].R:=complex(1,0);
    Vr[i].S:=complex(1,0);
    Vr[i].T:=complex(1,0);
  end;
  for i:=1 to N do
  begin
    With Za[i] do
    begin
      No:=round(rBranch[i,1]);
      Nbhind:=round(rBranch[i,2]);
      LnAHead1:=round(rBranch[i,3]);
      LnAHead2:=round(rBranch[i,4]);
      LnAHead3:=round(rBranch[i,5]);
      Panjang:=rBranch[i,6];
      GMR:=rBranch[i,7]*0.001;
      R:=rBranch[i,8]*0.001;
    end;
  end;
  Nterm:=0;
  for i:=1 to N do
  begin

```

```

With Za[i] do
begin
  if (LnAHead1=0) and (LnAHead2=0) and (LnAHead3=0) then
  begin
    Inc(Nterm);
    terminal[Nterm]:=i;
  end;
end;
end;
end;

```

```

procedure DataSkripsi;
var i:integer;
begin
  N:=55;
  Vbase:=20;KonstV:=1000;
  Pbase:=100;KonstP:=1000;
  Zbase:=sqrt(Vbase*KonstV)/(Pbase*KonstP);
  Ibase:=-Vbase*KonstV/Zbase;
  SetLength(Bus,N+1);
  SetLength(Vr,N+1);
  SetLength(Za,N+1);
  SetLength(terminal,17+1);
  Bus[1].No:=1;
  Bus[1].Load.R:=complex(44.1648/Pbase,26.153/Pbase);
  Bus[1].Load.S:=complex(31.0288/Pbase,18.411/Pbase);
  Bus[1].Load.T:=complex(36.2452/Pbase,21.551/Pbase);
  Bus[2].No:=2;
  Bus[2].Load.R:=complex(0.0000/Pbase, 0.0000/Pbase);
  Bus[2].Load.S:=complex(0.0000/Pbase, 0.0000/Pbase);
  Bus[2].Load.T:=complex(0.0000/Pbase, 0.0000/Pbase);
  Bus[3].No:=3;
  Bus[3].Load.R:=complex(8.9736/Pbase,5.275/Pbase);
  Bus[3].Load.S:=complex(10.0276/Pbase,5.950/Pbase);
  Bus[3].Load.T:=complex(15.2440/Pbase,9.092/Pbase);
  Bus[4].No:=4;
  Bus[4].Load.R:=complex(36.2184/Pbase,21.439/Pbase);
  Bus[4].Load.S:=complex(43.8944/Pbase,26.045/Pbase);
  Bus[4].Load.T:=complex(47.4080/Pbase,28.173/Pbase);
  Bus[5].No:=5;
  Bus[5].Load.R:=complex(19.1904/Pbase,11.337/Pbase);
  Bus[5].Load.S:=complex(23.8392/Pbase,14.145/Pbase);
  Bus[5].Load.T:=complex(28.6772/Pbase,17.061/Pbase);
  Bus[6].No:=6;
  Bus[6].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
  Bus[6].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
  Bus[6].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
  Bus[7].No:=7;
  Bus[7].Load.R:=complex(34.7048/Pbase,20.541/Pbase);
  Bus[7].Load.S:=complex(39.1644/Pbase,23.239/Pbase);
  Bus[7].Load.T:=complex(39.6508/Pbase,23.571/Pbase);
  Bus[8].No:=8;
  Bus[8].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
  Bus[8].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
  Bus[8].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
  Bus[9].No:=9;
  Bus[9].Load.R:=complex(48.1380/Pbase,28.510/Pbase);
  Bus[9].Load.S:=complex(48.2460/Pbase,28.627/Pbase);

```

```
Bus[9].Load.T:=complex(53.0840/Pbase,31.541/Pbase);
Bus[10].No:=10;
Bus[10].Load.R:=complex(120.9800/Pbase,71.724/Pbase);
Bus[10].Load.S:=complex(157.9820/Pbase,93.741/Pbase);
Bus[10].Load.T:=complex(134.4400/Pbase,79.805/Pbase);
Bus[11].No:=11;
Bus[11].Load.R:=complex(33.9480/Pbase,20.092/Pbase);
Bus[11].Load.S:=complex(34.0560/Pbase,20.208/Pbase);
Bus[11].Load.T:=complex(37.0020/Pbase,21.999/Pbase);
Bus[12].No:=12;
Bus[12].Load.R:=complex(22.0284/Pbase,13.020/Pbase);
Bus[12].Load.S:=complex(24.2176/Pbase,14.370/Pbase);
Bus[12].Load.T:=complex(19.7848/Pbase,11.786/Pbase);
Bus[13].No:=13;
Bus[13].Load.R:=complex(29.5964/Pbase,17.510/Pbase);
Bus[13].Load.S:=complex(32.9208/Pbase,19.534/Pbase);
Bus[13].Load.T:=complex(41.5428/Pbase,24.694/Pbase);
Bus[14].No:=14;
Bus[14].Load.R:=complex(8.2168/Pbase,4.826/Pbase);
Bus[14].Load.S:=complex(13.4332/Pbase,7.971/Pbase);
Bus[14].Load.T:=complex(15.4332/Pbase,9.204/Pbase);
Bus[15].No:=15;
Bus[15].Load.R:=complex(16.5416/Pbase,9.765/Pbase);
Bus[15].Load.S:=complex(19.2984/Pbase,11.451/Pbase);
Bus[15].Load.T:=complex(19.2172/Pbase,11.449/Pbase);
Bus[16].No:=16;
Bus[16].Load.R:=complex(27.7044/Pbase,16.388/Pbase);
Bus[16].Load.S:=complex(43.5160/Pbase,25.821/Pbase);
Bus[16].Load.T:=complex(34.5424/Pbase,20.541/Pbase);
Bus[17].No:=17;
Bus[17].Load.R:=complex(2.9192/Pbase,1.684/Pbase);
Bus[17].Load.S:=complex(14.5684/Pbase,8.644/Pbase);
Bus[17].Load.T:=complex(20.1632/Pbase,12.010/Pbase);
Bus[18].No:=18;
Bus[18].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[18].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[18].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[19].No:=19;
Bus[19].Load.R:=complex(6.5140/Pbase,3.816/Pbase);
Bus[19].Load.S:=complex(9.4600/Pbase,5.613/Pbase);
Bus[19].Load.T:=complex(11.0816/Pbase,6.622/Pbase);
Bus[20].No:=20;
Bus[20].Load.R:=complex(3.6760/Pbase,2.133/Pbase);
Bus[20].Load.S:=complex(11.3520/Pbase,6.736/Pbase);
Bus[20].Load.T:=complex(11.4600/Pbase,6.847/Pbase);
Bus[21].No:=21;
Bus[21].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[21].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[21].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[22].No:=22;
Bus[22].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[22].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[22].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[23].No:=23;
Bus[23].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[23].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[23].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[24].No:=24;
```

```

Bus[24].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[24].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[24].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[25].No:=25;
Bus[25].Load.R:=complex(15.2172/Pbase,8.979/Pbase);
Bus[25].Load.S:=complex(11.3520/Pbase,6.736/Pbase);
Bus[25].Load.T:=complex(13.3520/Pbase,7.969/Pbase);
Bus[26].No:=26;
Bus[26].Load.R:=complex(35.8400/Pbase,21.214/Pbase);
Bus[26].Load.S:=complex(39.7320/Pbase,23.576/Pbase);
Bus[26].Load.T:=complex(48.3540/Pbase,28.734/Pbase);
Bus[27].No:=27;
Bus[27].Load.R:=complex(10.2980/Pbase,6.061/Pbase);
Bus[27].Load.S:=complex(3.7840/Pbase,2.245/Pbase);
Bus[27].Load.T:=complex(12.4060/Pbase,7.408/Pbase);
Bus[28].No:=28;
Bus[28].Load.R:=complex(41.5160/Pbase,24.581/Pbase);
Bus[28].Load.S:=complex(45.4080/Pbase,26.943/Pbase);
Bus[28].Load.T:=complex(42.6780/Pbase,25.367/Pbase);
Bus[29].No:=29;
Bus[29].Load.R:=complex(12.0008/Pbase,7.071/Pbase);
Bus[29].Load.S:=complex(20.6228/Pbase,12.237/Pbase);
Bus[29].Load.T:=complex(30.1908/Pbase,17.959/Pbase);
Bus[30].No:=30;
Bus[30].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);//3
Bus[30].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);//3 5
Bus[30].Load.T:=complex(0.0000/Pbase,0.0000/Pbase)//1 0
Bus[31].No:=31;
Bus[31].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);//3 8
Bus[31].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);//9 2
Bus[31].Load.T:=complex(0.0000/Pbase,0.0000/Pbase)//1 9
Bus[32].No:=32;
Bus[32].Load.R:=complex(24.6772/Pbase,14.592/Pbase);
Bus[32].Load.S:=complex(34.8182/Pbase,20.657/Pbase);
Bus[32].Load.T:=complex(31.1368/Pbase,18.520/Pbase);
Bus[33].No:=33;
Bus[33].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[33].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[33].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[34].No:=34;
Bus[34].Load.R:=complex(32.2452/Pbase,19.081/Pbase); // 2 5
Bus[34].Load.S:=complex(38.4076/Pbase,22.790/Pbase); // 8 7
Bus[34].Load.T:=complex(36.6236/Pbase,21.775/Pbase); // 0 8
Bus[35].No:=35;
Bus[35].Load.R:=complex(7.6492/Pbase,4.490/Pbase);
Bus[35].Load.S:=complex(12.2980/Pbase,7.297/Pbase);
Bus[35].Load.T:=complex(16.3792/Pbase,9.765/Pbase);
Bus[36].No:=36;
Bus[36].Load.R:=complex(31.1100/Pbase,18.408/Pbase);
Bus[36].Load.S:=complex(32.3532/Pbase,19.197/Pbase);
Bus[36].Load.T:=complex(36.4344/Pbase,21.663/Pbase);
Bus[37].No:=37;
Bus[37].Load.R:=complex(30.7316/Pbase,18.183/Pbase);
Bus[37].Load.S:=complex(31.5964/Pbase,18.748/Pbase);
Bus[37].Load.T:=complex(36.0560/Pbase,21.439/Pbase);
Bus[38].No:=38;
Bus[38].Load.R:=complex(25.6232/Pbase,15.153/Pbase);
Bus[38].Load.S:=complex(31.5964/Pbase,18.748/Pbase);

```

```

Bus[38].Load.T:=complex(31.3260/Pbase,18.632/Pbase);
Bus[39].No:=39;
Bus[39].Load.R:=complex(5.3788/Pbase,3.143/Pbase);
Bus[39].Load.S:=complex(0.7568/Pbase,0.449/Pbase);
Bus[39].Load.T:=complex(9.7572/Pbase,5.837/Pbase);
Bus[40].No:=40;
Bus[40].Load.R:=complex(6.3248/Pbase,3.704/Pbase);
Bus[40].Load.S:=complex(11.9196/Pbase,7.073/Pbase);
Bus[40].Load.T:=complex(12.9736/Pbase,7.745/Pbase);
Bus[41].No:=41;
Bus[41].Load.R:=complex(4.2436/Pbase,2.469/Pbase);
Bus[41].Load.S:=complex(15.5144/Pbase,9.206/Pbase);
Bus[41].Load.T:=complex(16.0008/Pbase,9.541/Pbase);
Bus[42].No:=42;
Bus[42].Load.R:=complex(15.9740/Pbase,9.428/Pbase);
Bus[42].Load.S:=complex(18.9200/Pbase,11.226/Pbase);
Bus[42].Load.T:=complex(20.1632/Pbase,12.010/Pbase);
Bus[43].No:=43;
Bus[43].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[43].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[43].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[44].No:=44;
Bus[44].Load.R:=complex(3.5948/Pbase,2.133/Pbase); // 3 6
Bus[44].Load.S:=complex(4.5408/Pbase,2.694/Pbase); // 2 7
Bus[44].Load.T:=complex(7.7572/Pbase,4.602/Pbase); // 0 0
Bus[45].No:=45;
Bus[45].Load.R:=complex(15.9740/Pbase,9.428/Pbase);
Bus[45].Load.S:=complex(22.7040/Pbase,13.472/Pbase);
Bus[45].Load.T:=complex(17.1360/Pbase,10.214/Pbase);
Bus[46].No:=46;
Bus[46].Load.R:=complex(1.7840/Pbase,1.010/Pbase);
Bus[46].Load.S:=complex(12.4872/Pbase,7.409/Pbase);
Bus[46].Load.T:=complex(15.6224/Pbase,9.316/Pbase);
Bus[47].No:=47;
Bus[47].Load.R:=complex(15.0280/Pbase,8.867/Pbase);
Bus[47].Load.S:=complex(10.9736/Pbase,6.511/Pbase);
Bus[47].Load.T:=complex(17.1360/Pbase,10.214/Pbase);
Bus[48].No:=48;
Bus[48].Load.R:=complex(1.7840/Pbase,1.010/Pbase); // 5 5
Bus[48].Load.S:=complex(9.4600/Pbase,5.613/Pbase); // 0 5
Bus[48].Load.T:=complex(5.7840/Pbase,3.480/Pbase); // 7 2
Bus[49].No:=49;
Bus[49].Load.R:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[49].Load.S:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[49].Load.T:=complex(0.0000/Pbase,0.0000/Pbase);
Bus[50].No:=50;
Bus[50].Load.R:=complex(2.2704/Pbase,1.347/Pbase);
Bus[50].Load.S:=complex(2.4596/Pbase,1.459/Pbase);
Bus[50].Load.T:=complex(6.0544/Pbase,3.592/Pbase);
Bus[51].No:=51;
Bus[51].Load.R:=complex(15.9740/Pbase,9.428/Pbase);
Bus[51].Load.S:=complex(22.1364/Pbase,13.135/Pbase);
Bus[51].Load.T:=complex(29.8124/Pbase,17.735/Pbase);
Bus[52].No:=52;
Bus[52].Load.R:=complex(0.3784/Pbase,0.224/Pbase);
Bus[52].Load.S:=complex(1.3244/Pbase,0.786/Pbase);
Bus[52].Load.T:=complex(4.7300/Pbase,2.806/Pbase);
Bus[53].No:=53;

```

```

Bus[53].Load.R:=complex(4.2436/Pbase,2.469/Pbase);
Bus[53].Load.S:=complex(8.8924/Pbase,5.276/Pbase);
Bus[53].Load.T:=complex(10.1356/Pbase,6.061/Pbase);
Bus[54].No:=54;
Bus[54].Load.R:=complex(39.8132/Pbase,23.571/Pbase);
Bus[54].Load.S:=complex(35.0020/Pbase,20.769/Pbase);
Bus[54].Load.T:=complex(24.7040/Pbase,14.704/Pbase);
Bus[55].No:=55;
Bus[55].Load.R:=complex(33.9480/Pbase,20.092/Pbase);
Bus[55].Load.S:=complex(25.7312/Pbase,15.268/Pbase);
Bus[55].Load.T:=complex(33.2180/Pbase,19.755/Pbase);
for i:=0 to N do
begin
  Vr[i].R:=complex(1,0);
  Vr[i].S:=complex(1,0);
  Vr[i].T:=complex(1,0);
end;
With Za[1] do
begin
  No:=1;
  Nhind:=0;
  LnAHead1:=2;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=303;
  GMR:=5.2365*0.001;
  R:=0.2162*0.001;
end;
With Za[2] do
begin
  No:=2;
  Nhind:=1;
  LnAHead1:=3;
  LnAHead2:=53;
  LnAHead3:=0;
  Panjang:=412;

```



```
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[3] do
begin
No:=3;
Nbhind:=2;
LnAHead1:=4;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=19;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[4] do
begin
No:=4;
Nbhind:=3;
LnAHead1:=5;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=252;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[5] do
begin
No:=5;
Nbhind:=4;
LnAHead1:=6;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=148;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[6] do
begin
No:=6;
Nbhind:=5;
LnAHead1:=7;
LnAHead2:=54;
LnAHead3:=0;
Panjang:=266;
GMR:=5.2365;
R:=0.2162*0.001;
end;
With Za[7] do
begin
No:=7;
Nbhind:=6;
LnAHead1:=8;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=63;
GMR:=5.2365;
```

```
R:=0.2162*0.001;
end;
With Za[8] do
begin
No:=8;
Nbhind:=7;
LnAHead1:=9;
LnAHead2:=29;
LnAHead3:=0;
Panjang:=150;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[9] do
begin
No:=9;
Nbhind:=8;
LnAHead1:=10;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=303;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[10] do
begin
No:=10;
Nbhind:=9;
LnAHead1:=11;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=86;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[11] do
begin
No:=11;
Nbhind:=10;
LnAHead1:=12;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=202;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[12] do
begin
No:=12;
Nbhind:=11;
LnAHead1:=13;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=324;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
```

```
end;
With Za[13] do
begin
  No:=13;
  Nbhind:=12;
  LnAHead1:=14;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=217.5;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[14] do
begin
  No:=14;
  Nbhind:=13;
  LnAHead1:=15;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=86;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[15] do
begin
  No:=15;
  Nbhind:=14;
  LnAHead1:=16;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=146;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[16] do
begin
  No:=16;
  Nbhind:=15;
  LnAHead1:=17;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=410;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[17] do
begin
  No:=17;
  Nbhind:=16;
  LnAHead1:=18;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=96;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
```

```
With Za[18] do
begin
  No:=18;
  Nbhind:=17;
  LnAHead1:=19;
  LnAHead2:=39;
  LnAHead3:=0;
  Panjang:=328;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[19] do
begin
  No:=19;
  Nbhind:=18;
  LnAHead1:=20;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=241;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[20] do
begin
  No:=20;
  Nbhind:=19;
  LnAHead1:=21;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=51;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[21] do
begin
  No:=21;
  Nbhind:=20;
  LnAHead1:=22;
  LnAHead2:=40;
  LnAHead3:=0;
  Panjang:=348;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[22] do
begin
  No:=22;
  Nbhind:=21;
  LnAHead1:=23;
  LnAHead2:=41;
  LnAHead3:=0;
  Panjang:=386;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[23] do
```

```
begin
  No:=23;
  Nbhind:=22;
  LnAHead1:=24;
  LnAHead2:=43;
  LnAHead3:=0;
  Panjang:=145;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[24] do
begin
  No:=24;
  Nbhind:=23;
  LnAHead1:=25;
  LnAHead2:=42;
  LnAHead3:=0;
  Panjang:=84;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[25] do
begin
  No:=25;
  Nbhind:=24;
  LnAHead1:=26;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=35;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[26] do
begin
  No:=26;
  Nbhind:=25;
  LnAHead1:=27;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=162;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[27] do
begin
  No:=27;
  Nbhind:=26;
  LnAHead1:=28;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=265;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[28] do
begin
```

```
No:=28;
Nbhind:=27;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=215;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[29] do
begin
No:=29;
Nbhind:=8;
LnAHead1:=30;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=234;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[30] do
begin
No:=30;
Nbhind:=29;
LnAHead1:=31;
LnAHead2:=35;
LnAHead3:=0;
Panjang:=42;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[31] do
begin
No:=31;
Nbhind:=30;
LnAHead1:=32;
LnAHead2:=36;
LnAHead3:=38;
Panjang:=201;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[32] do
begin
No:=32;
Nbhind:=31;
LnAHead1:=33;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=205;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[33] do
begin
No:=33;
```

```
Nbhind:=32;
LnAHead1:=34;
LnAHead2:=37;
LnAHead3:=0;
Panjang:=290;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[34] do
begin
No:=34;
Nbhind:=33;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=88;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[35] do
begin
No:=35;
Nbhind:=30;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=221;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[36] do
begin
No:=36;
Nbhind:=31;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=161;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[37] do
begin
No:=37;
Nbhind:=33;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=368;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[38] do
begin
No:=38;
Nbhind:=31;
```

```
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=92;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[39] do
begin
  No:=39;
  Nbhind:=18;
  LnAHead1:=0;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=79;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[40] do
begin
  No:=40;
  Nbhind:=21;
  LnAHead1:=0;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=35;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[41] do
begin
  No:=41;
  Nbhind:=22;
  LnAHead1:=0;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=150;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[42] do
begin
  No:=42;
  Nbhind:=24;
  LnAHead1:=0;
  LnAHead2:=0;
  LnAHead3:=0;
  Panjang:=140;
  GMR:=4.6837*0.001;
  R:=0.2688*0.001;
end;
With Za[43] do
begin
  No:=43;
  Nbhind:=23;
  LnAHead1:=44;
```

```
LnAHead2:=48;
LnAHead3:=0;
Panjang:=208;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[44] do
begin
No:=44;
Nbhind:=43;
LnAHead1:=45;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=154;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[45] do
begin
No:=45;
Nbhind:=44;
LnAHead1:=46;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=195;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[46] do
begin
No:=46;
Nbhind:=45;
LnAHead1:=47;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=120;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[47] do
begin
No:=47;
Nbhind:=46;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=119;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[48] do
begin
No:=48;
Nbhind:=43;
LnAHead1:=49;
LnAHead2:=0;
```

```
LnAHead3:=0;
Panjang:=37;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[49] do
begin
No:=49;
Nbhind:=48;
LnAHead1:=50;
LnAHead2:=51;
LnAHead3:=52;
Panjang:=296;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[50] do
begin
No:=50;
Nbhind:=49;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=85;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[51] do
begin
No:=51;
Nbhind:=49;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=37;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[52] do
begin
No:=52;
Nbhind:=49;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=259;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[53] do
begin
No:=53;
Nbhind:=2;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
```

```

Panjang:=98;
GMR:=4.6837*0.001;
R:=0.2688*0.001;
end;
With Za[54] do
begin
No:=54;
Nbhind:=6;
LnAHead1:=55;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=75;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
With Za[55] do
begin
No:=55;
Nbhind:=54;
LnAHead1:=0;
LnAHead2:=0;
LnAHead3:=0;
Panjang:=146;
GMR:=5.2365*0.001;
R:=0.2162*0.001;
end;
terminal[1]:=28;
terminal[2]:=42;
terminal[3]:=52;
terminal[4]:=51;
terminal[5]:=50;
terminal[6]:=47;
terminal[7]:=41;
terminal[8]:=40;
terminal[9]:=39;
terminal[10]:=34;
terminal[11]:=37;
terminal[12]:=38;
terminal[13]:=36;
terminal[14]:=35;
terminal[15]:=55;
terminal[16]:=53;
end;

procedure EncodeData(var rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss:dArr2);
var i:integer;
begin
SetLength(rV,N+1,7);
SetLength(rI,N+1,7);
SetLength(rSa,N+1,7);
SetLength(rSL,N+1,7);
SetLength(rSumGen,4,3);
SetLength(rSumLoad,4,3);
SetLength(rSumLoss,4,3);
rSumGen[1,1]:=Sa[1].R.re;rSumGen[1,2]:=Sa[1].R.im;
rSumGen[2,1]:=Sa[1].S.re;rSumGen[2,2]:=Sa[1].S.im;

```

```

rSumGen[3,1]:=Sa[1].T.re;rSumGen[3,2]:=Sa[1].T.im;
rSumLoss[1,1]:=0;rSumLoss[1,2]:=0;
rSumLoss[2,1]:=0;rSumLoss[2,2]:=0;
rSumLoss[3,1]:=0;rSumLoss[3,2]:=0;
rSumLoad[1,1]:=0;rSumLoad[1,2]:=0;
rSumLoad[2,1]:=0;rSumLoad[2,2]:=0;
rSumLoad[3,1]:=0;rSumLoad[3,2]:=0;
for i:=1 to N do
begin
  rV[i,1]:=Vr[i].R.re;
  rV[i,2]:=Vr[i].R.im;
  rV[i,3]:=Vr[i].S.re;
  rV[i,4]:=Vr[i].S.im;
  rV[i,5]:=Vr[i].T.re;
  rV[i,6]:=Vr[i].T.im;
  rI[i,1]:=r[i].R.re;
  rI[i,2]:=r[i].R.im;
  rI[i,3]:=r[i].S.re;
  rI[i,4]:=r[i].S.im;
  rI[i,5]:=r[i].T.re;
  rI[i,6]:=r[i].T.im;
  rSa[i,1]:=Sa[i].R.re;
  rSa[i,2]:=Sa[i].R.im;
  rSa[i,3]:=Sa[i].S.re;
  rSa[i,4]:=Sa[i].S.im;
  rSa[i,5]:=Sa[i].T.re;
  rSa[i,6]:=Sa[i].T.im;
  rSL[i,1]:=SL[i].R.re;
  rSL[i,2]:=SL[i].R.im;
  rSL[i,3]:=SL[i].S.re;
  rSL[i,4]:=SL[i].S.im;
  rSL[i,5]:=SL[i].T.re;
  rSL[i,6]:=SL[i].T.im;
  rSumLoss[1,1]:=rSumLoss[1,1]+rSL[i,1];
  rSumLoss[1,2]:=rSumLoss[1,2]+rSL[i,2];
  rSumLoss[2,1]:=rSumLoss[2,1]+rSL[i,3];
  rSumLoss[2,2]:=rSumLoss[2,2]+rSL[i,4];
  rSumLoss[3,1]:=rSumLoss[3,1]+rSL[i,5];
  rSumLoss[3,2]:=rSumLoss[3,2]+rSL[i,6];
  rSumLoad[1,1]:=rSumLoad[1,1]+Bus[i].Load.R.re;
  rSumLoad[1,2]:=rSumLoad[1,2]+Bus[i].Load.R.im;
  rSumLoad[2,1]:=rSumLoad[2,1]+Bus[i].Load.S.re;
  rSumLoad[2,2]:=rSumLoad[2,2]+Bus[i].Load.S.im;
  rSumLoad[3,1]:=rSumLoad[3,1]+Bus[i].Load.T.re;
  rSumLoad[3,2]:=rSumLoad[3,2]+Bus[i].Load.T.im;
end;
end;

procedure HitungFD3Phasa(const rNbus:integer;
  const rVbase,rVKonst,rPbase,rPKonst:double;
  const rBus,rBranch:dArr2;
  var rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss:dArr2;
  var riterasi:byte);
var i,j,k:integer;
  ib,Pb:TComplex;
begin

```

```

DecodeData(rNbus,rVbase,rVkonst,rPbase,rPkonst,rBus,rBranch);
ImpedansiSaluranKm(N,Za);
lb:=complex(lbase,0);
Pb:=complex(Pbase,0);
for i:=1 to N do
begin
  With Za[i] do
  begin
    for j:=1 to 3 do
    begin
      for k:=1 to 3 do
      begin
        Zs[j,k]:=complex(CxRe(Zs[j,k])/Zbase,CxIm(Zs[j,k])/Zbase);
      end;
    end;
  end;
end;
FastDLFlow(N,Za,Bus,rIterasi,Vr,Ir,Sa,SL);
EncodeData(rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss);
end;

procedure HitungFD3Phasa(var rV,rI,rSa,rSL,rSumGen,rSumLoad,
  rSumLoss:dArr2;
  var rIterasi:byte);
var i,j,k:integer;
  lb,Pb:TComplex;
begin
  DataSkripsi;
  ImpedansiSaluranKm(N,Za);
  lb:=complex(lbase,0);
  Pb:=complex(Pbase,0);
  for i:=1 to N do
  begin
    With Za[i] do
    begin
      for j:=1 to 3 do
      begin
        for k:=1 to 3 do
        begin
          Zs[j,k]:=complex(CxRe(Zs[j,k])/Zbase,CxIm(Zs[j,k])/Zbase);
        end;
      end;
    end;
  end;
  FastDLFlow(N,Za,Bus,rIterasi,Vr,Ir,Sa,SL);
  EncodeData(rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss);
end;

function PowerUnbalance(var rBus:BArr):double;
var i:integer;
  sum,avg:double;

function getAbs(var rRe,rIm:double):double;
begin
  result:=sqrt(sqr(rRe)+sqr(rIm));
end;

```

```

begin
  result:=0;
  for i:=1 to N do
  begin
    sum:=0;
    sum:=sum+getAbs(rBus[i].Load.R.re,rBus[i].Load.R.im);
    sum:=sum+getAbs(rBus[i].Load.S.re,rBus[i].Load.S.im);
    sum:=sum+getAbs(rBus[i].Load.T.re,rBus[i].Load.T.im);
    avg:=sum/3;
    result:=result+abs(getAbs(rBus[i].Load.R.re,rBus[i].Load.R.im)-avg);
    result:=result+abs(getAbs(rBus[i].Load.S.re,rBus[i].Load.S.im)-avg);
    result:=result+abs(getAbs(rBus[i].Load.T.re,rBus[i].Load.T.im)-avg);
  end;
end;

function LineLoss(var rSumLoss:dArr2):double;
begin
  result:=rSumLoss[1,1]+rSumLoss[2,1]+rSumLoss[3,1];
end;

function AvgVoltDrop(var rV:PArr):double;
var i:integer;
    sum:double;

function getAbs(const rRe,rIm:double):double;
begin
  result:=sqrt(sqr(rRe)+sqr(rIm));
end;

begin
  result:=0;
  for i:=1 to N do
  begin
    sum:=0;
    sum:=sum+abs((1.000-getAbs(rV[i].R.re,rV[i].R.im))/1.000);
    sum:=sum+abs((1.000-getAbs(rV[i].S.re,rV[i].S.im))/1.000);
    sum:=sum+abs((1.000-getAbs(rV[i].T.re,rV[i].T.im))/1.000);
    result:=result+sum/3;
  end;
end;

function getTypConn(const rChrom:dArr1):iArr1;
var i:integer;

begin
  SetLength(result,N+1);
  for i:=1 to N do
  begin
    result[i]:=round(GetBatasToReal(rChrom[i],1.6));
  end;
end;

function doHitung(const rChrom:dArr1):double;
var i,j,k:integer;
    typConn:iArr1;
    iter:byte;

```

```

tmpBus:BArr;
ib,Pb:TComplex;
rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss:dArr2;
begin
DataSkripsi;
SetLength(tmpBus,N+1);
for i:=1 to N do
begin
tmpBus[i].No:=Bus[i].No;
tmpBus[i].Load.R.re:=Bus[i].Load.R.re;
tmpBus[i].Load.R.im:=Bus[i].Load.R.im;
tmpBus[i].Load.S.re:=Bus[i].Load.S.re;
tmpBus[i].Load.S.im:=Bus[i].Load.S.im;
tmpBus[i].Load.T.re:=Bus[i].Load.T.re;
tmpBus[i].Load.T.im:=Bus[i].Load.T.im;
end;
typConn:=getTypConn(rChrom);
for i:=1 to N do
begin
if typConn[i]=1 then
begin
Bus[i].Load.R.re:=tmpBus[i].Load.R.re;
Bus[i].Load.R.im:=tmpBus[i].Load.R.im;
Bus[i].Load.S.re:=tmpBus[i].Load.S.re;
Bus[i].Load.S.im:=tmpBus[i].Load.S.im;
Bus[i].Load.T.re:=tmpBus[i].Load.T.re;
Bus[i].Load.T.im:=tmpBus[i].Load.T.im;
end
else if typConn[i]=2 then
begin
Bus[i].Load.R.re:=tmpBus[i].Load.R.re;
Bus[i].Load.R.im:=tmpBus[i].Load.R.im;
Bus[i].Load.S.re:=tmpBus[i].Load.T.re;
Bus[i].Load.S.im:=tmpBus[i].Load.T.im;
Bus[i].Load.T.re:=tmpBus[i].Load.S.re;
Bus[i].Load.T.im:=tmpBus[i].Load.S.im;
end
else if typConn[i]=3 then
begin
Bus[i].Load.R.re:=tmpBus[i].Load.T.re;
Bus[i].Load.R.im:=tmpBus[i].Load.T.im;
Bus[i].Load.S.re:=tmpBus[i].Load.R.re;
Bus[i].Load.S.im:=tmpBus[i].Load.R.im;
Bus[i].Load.T.re:=tmpBus[i].Load.S.re;
Bus[i].Load.T.im:=tmpBus[i].Load.S.im;
end
else if typConn[i]=4 then
begin
Bus[i].Load.R.re:=tmpBus[i].Load.S.re;
Bus[i].Load.R.im:=tmpBus[i].Load.S.im;
Bus[i].Load.S.re:=tmpBus[i].Load.R.re;
Bus[i].Load.S.im:=tmpBus[i].Load.R.im;
Bus[i].Load.T.re:=tmpBus[i].Load.T.re;
Bus[i].Load.T.im:=tmpBus[i].Load.T.im;
end
else if typConn[i]=5 then

```

```

oegin
  Bus[i].Load.R.re:=tmpBus[i].Load.S.re;
  Bus[i].Load.R.im:=tmpBus[i].Load.S.im;
  Bus[i].Load.S.re:=tmpBus[i].Load.T.re;
  Bus[i].Load.S.im:=tmpBus[i].Load.T.im;
  Bus[i].Load.T.re:=tmpBus[i].Load.R.re;
  Bus[i].Load.T.im:=tmpBus[i].Load.R.im;
end
else if typConn[i]=6 then
begin
  Bus[i].Load.R.re:=tmpBus[i].Load.T.re;
  Bus[i].Load.R.im:=tmpBus[i].Load.T.im;
  Bus[i].Load.S.re:=tmpBus[i].Load.S.re;
  Bus[i].Load.S.im:=tmpBus[i].Load.S.im;
  Bus[i].Load.T.re:=tmpBus[i].Load.R.re;
  Bus[i].Load.T.im:=tmpBus[i].Load.R.im;
end
end;
ImpedansiSaluranKm(N,Za);
Ib:=complex(Ibase,0);
Pb:=complex(Pbase,0);
for i:=1 to N do
begin
  With Za[i] do
  begin
    for j:=1 to 3 do
    begin
      for k:=1 to 3 do
      begin
        Zs[j,k]:=complex(CxRe(Zs[j,k])/Zbase,CxIm(Zs[j,k])/Zbase);
      end;
    end;
  end;
end;
FastDLFlow(N,Za,Bus,Iter,Vr,Ir,Sa,SL);
EncodeData(rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss);
result:=PowerUnbalance(Bus)+LineLoss(rSumLoss)+avgVoltDrop(Vr);
for i:=1 to N do
begin
  Bus[i].Load.R.re:=tmpBus[i].Load.R.re;
  Bus[i].Load.R.im:=tmpBus[i].Load.R.im;
  Bus[i].Load.S.re:=tmpBus[i].Load.S.re;
  Bus[i].Load.S.im:=tmpBus[i].Load.S.im;
  Bus[i].Load.T.re:=tmpBus[i].Load.T.re;
  Bus[i].Load.T.im:=tmpBus[i].Load.T.im;
end;
end;

procedure doHitungAkhir(const rChrom:dArr1;
  var rTypConn:iArr1;
  var rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss:dArr2;
  var rIterasi:byte);
var i,j,k:integer;
    tmpBus:BArr;
    Ib,Pb:TComplex;
begin

```



```

DataSkripsi;
SetLength(tmpBus,N+1);
for i:=1 to N do
begin
  tmpBus[i].No:=Bus[i].No;
  tmpBus[i].Load.R.re:=Bus[i].Load.R.re;
  tmpBus[i].Load.R.im:=Bus[i].Load.R.im;
  tmpBus[i].Load.S.re:=Bus[i].Load.S.re;
  tmpBus[i].Load.S.im:=Bus[i].Load.S.im;
  tmpBus[i].Load.T.re:=Bus[i].Load.T.re;
  tmpBus[i].Load.T.im:=Bus[i].Load.T.im;
end;
rTypConn:=getTypConn(rChrom);
for i:=1 to N do
begin
  if rTypConn[i]=1 then
  begin
    Bus[i].Load.R.re:=tmpBus[i].Load.R.re;
    Bus[i].Load.R.im:=tmpBus[i].Load.R.im;
    Bus[i].Load.S.re:=tmpBus[i].Load.S.re;
    Bus[i].Load.S.im:=tmpBus[i].Load.S.im;
    Bus[i].Load.T.re:=tmpBus[i].Load.T.re;
    Bus[i].Load.T.im:=tmpBus[i].Load.T.im;
  end
  else if rTypConn[i]=2 then
  begin
    Bus[i].Load.R.re:=tmpBus[i].Load.R.re;
    Bus[i].Load.R.im:=tmpBus[i].Load.R.im;
    Bus[i].Load.S.re:=tmpBus[i].Load.T.re;
    Bus[i].Load.S.im:=tmpBus[i].Load.T.im;
    Bus[i].Load.T.re:=tmpBus[i].Load.S.re;
    Bus[i].Load.T.im:=tmpBus[i].Load.S.im;
  end
  else if rTypConn[i]=3 then
  begin
    Bus[i].Load.R.re:=tmpBus[i].Load.T.re;
    Bus[i].Load.R.im:=tmpBus[i].Load.T.im;
    Bus[i].Load.S.re:=tmpBus[i].Load.R.re;
    Bus[i].Load.S.im:=tmpBus[i].Load.R.im;
    Bus[i].Load.T.re:=tmpBus[i].Load.S.re;
    Bus[i].Load.T.im:=tmpBus[i].Load.S.im;
  end
  else if rTypConn[i]=4 then
  begin
    Bus[i].Load.R.re:=tmpBus[i].Load.S.re;
    Bus[i].Load.R.im:=tmpBus[i].Load.S.im;
    Bus[i].Load.S.re:=tmpBus[i].Load.R.re;
    Bus[i].Load.S.im:=tmpBus[i].Load.R.im;
    Bus[i].Load.T.re:=tmpBus[i].Load.T.re;
    Bus[i].Load.T.im:=tmpBus[i].Load.T.im;
  end
  else if rTypConn[i]=5 then
  begin
    Bus[i].Load.R.re:=tmpBus[i].Load.S.re;
    Bus[i].Load.R.im:=tmpBus[i].Load.S.im;
    Bus[i].Load.S.re:=tmpBus[i].Load.T.re;

```

```

Bus[i].Load.S.im:=tmpBus[i].Load.T.im;
Bus[i].Load.T.re:=tmpBus[i].Load.R.re;
Bus[i].Load.T.im:=tmpBus[i].Load.R.im;
end
else if rTypConn[i]=6 then
begin
Bus[i].Load.R.re:=tmpBus[i].Load.T.re;
Bus[i].Load.R.im:=tmpBus[i].Load.T.im;
Bus[i].Load.S.re:=tmpBus[i].Load.S.re;
Bus[i].Load.S.im:=tmpBus[i].Load.S.im;
Bus[i].Load.T.re:=tmpBus[i].Load.R.re;
Bus[i].Load.T.im:=tmpBus[i].Load.R.im;
end
end;
ImpedansiSaluranKm(N,Za);
Ib:=complex(Ibase,0);
Pb:=complex(Pbase,0);
for i:=1 to N do
begin
With Za[i] do
begin
for j:=1 to 3 do
begin
for k:=1 to 3 do
begin
Zs[j].k:=complex(CxRe(Zs[j].k)/Zbase,CxIm(Zs[j].k)/Zbase);
end;
end;
end;
end;
end;
FastDLFlow(N,Za,Bus,rIterasi,Vr,Ir,Sa,SL);
EncodeData(rV,rI,rSa,rSL,rSumGen,rSumLoad,rSumLoss);
for i:=1 to N do
begin
Bus[i].Load.R.re:=tmpBus[i].Load.R.re;
Bus[i].Load.R.im:=tmpBus[i].Load.R.im;
Bus[i].Load.S.re:=tmpBus[i].Load.S.re;
Bus[i].Load.S.im:=tmpBus[i].Load.S.im;
Bus[i].Load.T.re:=tmpBus[i].Load.T.re;
Bus[i].Load.T.im:=tmpBus[i].Load.T.im;
end;
end;
end.

```

```

unit uSA;

interface

uses uUtils,uFitness,uGenetic,uRandom,uHasilGA;

type
TSA=class
private
  FNgen,FNjam,Filterasi,FNCool:integer;
  Fr,FTheta,FX0,FPflip:double;
  FBest,FOpt,FCek:TIndiBin2;
  FRandom:TRandomu;
  function getRandomChrom(const rPflip:double):bArr2;
  procedure InitSA;
  function UbahSopt(const rScek:TIndiBin2):TIndiBin2;
  procedure doHitung;
  function GetChromHasil:bArr2;
public
  constructor Create;overload;
  constructor Create(const rNgen,rNjam,rIterasi,rNCool:integer;
    const rR,rX0,rPflip:double);overload;
  destructor Destroy;override;
  property Ngen:integer read FNgen write FNgen;
  property Njam:integer read FNjam write FNjam;
  property Iterasi:integer read Filterasi write Filterasi;
  property NCool:integer read FNCool write FNCool;
  property R:double read FR write FR;
  property Theta:double read FTheta write FTheta;
  property X0:double read FX0 write FX0;
  property Pflip:double read FPflip write FPflip;
  property ChromHasil:bArr2 read GetChromHasil;
end;

implementation

//constructor
constructor TSA.Create;
begin
  inherited Create;
  FNgen:=0;
  FNjam:=0;
  Filterasi:=100;
  FNCool:=10;
  FR:=0.95;
  FX0:=0.95;
  FPflip:=0.3;
  FRandom:=TRandomu.Create;
end;

constructor TSA.Create(const rNgen,rNjam,rIterasi,rNCool:integer;
  const rR,rX0,rPflip:double);
begin
  inherited Create;
  FNgen:=rNgen;
  FNjam:=rNjam;

```

```

Filterasi:=rIterasi;
FNCool:=rNCool;
Fr:=rR;
FX0:=rX0;
FPflip:=rPflip;
end;

//data processing
function TSA.UbahSopt(const rScek:TindiBin2):TindiBin2;
var i,j:integer;
begin
  SetLength(result.chrom,FNgen,FNjam);
  for i:=0 to FNgen-1 do
    begin
      for j:=0 to FNjam-1 do
        begin
          result.chrom[i,j]:=rScek.chrom[i,j];
        end;
      end;
    result.fitness:=rScek.fitness;
  end;

function TSA.getRandomChrom(const rPflip:double):bArr2;
var i,j:integer;
begin
  SetLength(result,FNgen,FNjam);
  for i:=0 to FNgen-1 do
    begin
      for j:=0 to FNjam-1 do
        begin
          result[i,j]:=FRandom.NextBoolean(rPflip);
        end;
      end;
    end;
  end;

procedure TSA.InitSA;
var i,X1:integer;
    ran,dC:double;
begin
  X1:=0;
  dC:=0;
  FOpt.chrom:=getRandomChrom(FPflip);
  FOpt.fitness:=CalcFitness(FOpt.chrom);
  for i:=1 to Filterasi do
    begin
      FCek.chrom:=getRandomChrom(FPflip);
      FCek.fitness:=CalcFitness(FOpt.chrom);
      if FCek.fitness<FOpt.fitness then
        begin
          Fopt:=UbahSopt(FCek);
        end
      else
        begin
          ran:=random;
          if FX0>ran then
            begin

```

```

    X1:=X1+1;
    dC:=dC+(FCek.fitness-FOpt.fitness);
    end;
    end;
    end;
    dC:=dC/X1;
    FTheta:=abs(dC/ln(1/X0-1));
    end;

procedure TSA.doHitung;
var i,k:integer;
    ran,dF,PdF,the:double;
begin
    InitSA;
    the:=FTheta;
    frmHasil.pbGen.StepBy(1);
    FOpt.chrom:=getRandomChrom(FPflip);
    FOpt.fitness:=CalcFitness(FOpt.chrom);
    FBest:=UbahSopt(FOpt);
    for k:=1 to FNCool do
    begin
        for l:=1 to Fiterasi do
        begin
            FCek.chrom:=getRandomChrom(FPflip);
            FCek.fitness:=CalcFitness(FOpt.chrom);
            if FCek.fitness<FBest.fitness then
            begin
                FBest:=UbahSopt(FCek);
            end;
            if FCek.fitness<FOpt.fitness then
            begin
                FOpt:=UbahSopt(FCek);
            end
            else
            begin
                dF:=FCek.fitness-FOpt.fitness;
                ran:=random;
                PdF:=1/(1+exp(dF/the));
                if ran<PdF then
                begin
                    FOpt:=UbahSopt(FCek);
                end;
            end;
        end;
        the:=pangkat(r,k-1)*Ftheta;
        frmHasil.pbGen.StepBy(1);
    end;
end;

//data output
function TSA.GetChromHasil:bArr2;
var i,j:integer;
begin
    SetLength(result,FNgen,FNjam);
    doHitung;
    for i:=0 to FNgen-1 do

```

```
begin
  for j:=0 to FNjam-1 do
    begin
      result[i,j]:=FBest.chrom[i,j];
    end;
  end;
end;
```

```
//destructor
destructor TSA.Destroy;
begin
  try
    FRandom.Free;
  finally
    inherited Destroy;
  end;
end;

end.
```



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Nomor : ITN- 608 /III.TA/2/05
Lampiran : -
Perihal : Survey

Malang, 14 Desember 2005

Kepada : Yth. Pimpinan
PT. PLN (Persero)
UB Distribusi Jatim Area Pelayan Malang
Jl. Jend. Basuki Rahmad No. 100
Di - Malang

Bersama ini dengan hormat kami mohon kebijaksanaan Saudara agar Mahasiswa kami dari Fakultas Teknologi Industri Jurusan Teknik Elektro S-1 Konsentrasi Teknik Energi Listrik dapat diijinkan untuk melaksanakan survey pada perusahaan yang saudara pimpin untuk mendapatkan data - data guna penyusunan Skripsi

Mahasiswa tersebut Adalah :

Didik Purnomo Nim. 01.12.128

Adapun lamanya Survey adalah : 30 Hari

Demikian agar maklum dan atas perhatian serta bantuannya kami ucapkan terima kasih.


DEKAN
Fakultas Teknologi Industri
H. Mochtar Asroni, MSME
Nip. Y.1018100036

PT. PLN (PERSERO)

DISTRIBUSI JAWA TIMUR

AREA PELAYANAN & JARINGAN MALANG

JL. BASUKI RAHMAD NO. 100 MALANG 651111



Certificate No.: **QSC**
00475

cs : 0341 - 326034 (Hunting)
: 211

Facsimile : 0341 - 362046

Nomor : 1444/330/APJ-MLG/ 2005
Surat Sdr.No : ITN-608/III.TA/2/05
Lampiran : -
Perihal : Survey

Malang ,23 Desember 2005

Kepada
INSTITUT TEKNOLOGI NASIONAL
JL. BENDUNGAN SIGURA GURA
NO 2
MALANG

Menjawab surat Saudara nomor ITN-608/III.TA/2/05 Tanggal 14 Desember 2005 perihal tersebut diatas, dengan ini kami beritahukan bahwa pada prinsipnya kami tidak keberatan / mengizinkan mahasiswa saudara atas nama :

Didik Purnomo

Nim 01.12.128

Untuk melaksanakan Survey di PT. PLN (Persero) Area Pelayanan dan Jaringan Malang mulai tanggal 26 Desember 2005 sampai dengan 06 Januari 2006 dengan catatan PLN hanya memberikan data yang **TIDAK BERSIFAT RAHASIA**.

Sebelumnya diminta agar mahasiswa tersebut mengisi surat pernyataan, dilengkapi dengan pas photo ukuran 3 X 4 Cm di SDM & Administrasi PT. PLN (Persero) Area Pelayanan dan Jaringan Malang.

Demikian agar menjadikan maklum.



Tembusan :
Manajer Unit Jaringan
Supervisor Terkait

LAMPIRAN

- ❖ Bimbingan Skripsi**
 - ❖ Formulir Bimbingan Skripsi**
 - ❖ Lembar Bimbingan Skripsi**
 - ❖ Berita Acara Ujian Skripsi**
- Fakultas Teknologi Industri**
-



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

PERSETUJUAN PERBAIKAN SKRIPSI

Dari hasil ujian skripsi jurusan Teknik Elektro jenjang strata satu (S-1) yang diselenggarakan pada :

Hari : Rabu
Tanggal : 22 Maret 2006

Telah dilakukan perbaikan Skripsi oleh :

1. Nama : Didik Purnomo
2. NIM : 01.12.128
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik (S-1)
5. Judul Skripsi : ANALISIS PENGARUH PENUKARAN FASA TERHADAP KETIDAKSEIMBANGAN FASA PADA GARDU INDUK BLIMBING DENGAN MENGGUNAKAN METODE SIMULATED ANNEALING

6. Perbaikan Meliputi:

No	Materi Perbaikan	Paraf Penguji
1.	Sebaiknya dicek juga profile tegangan sebelum dan sesudah di tukar fasanya	
2.	Flowchart loadflow dibuat satu saja, jadi di looping	

Anggota Penguji

29/03/06

(Ir. I Made Wartana, MT)
Penguji I

(Ir. M. Abdul Hamid, MT)
Penguji II

Dosen Pembimbing

(Ir. Taufik Hidayat, MT)
NIP : 1018700151

FORMULIR BIMBINGAN SKRIPSI

Nama : DIDIK PURNOMO
 Nim : 01.12.128
 Masa Bimbingan : 20 Desember 2005 s/d 20 Juni 2006
 Judul Skripsi : ANALISIS PENGARUH PENUKARAN FASA TERHADAP KETIDAKSEIMBANGAN FASA PADA GI BLIMBING DENGAN MENGGUNAKAN METODE SIMULATED ANNEALLING

No.	Tanggal	Uraian	Parap Pembimbing
1.	20/01 '06	Selesaikan bab IV (analisisnya)	Ah
2.	24/01 '06	- lengkapi data 2 y komputer	Ah
3.	26/01 '06	- paham resistansi/fahapan dan diagram alir	Ah
4.	17/02 '06	- tentukan parameter impedansi, unguis, daya, p, dan q yg tayar sebelum & setelah p-ur	Ah
5.	28/02 '06	- Beri keterangan pd tabel & gambar pd Bab IV, Beri contoh perhitungan utk data impedansi Eksteren	Ah
6.	8/03 '06	lengkapi bab III	Ah
7.	10/03 '06	lengkapi bab II	Ah
8.	12/03 '06	lengkapi flowchartnya (bab III)	Ah
9.	15/03 '06	ace mengkritik uraian	Ah
10.			

Malang, 200
 Dosen Pembimbing,

(Signature)
Dr. TAUFIK HIDAYAT, MT

Form.S-4b




LEMBAR BIMBINGAN SKRIPSI

1. Nama : DIDIK PURNOMO
2. NIM : 01.12.128
3. Jurusan : TEKNIK ELEKTRO
4. Konsentrasi : TEKNIK ENERGI LISTRIK
5. Program Pendidikan : STRATA SATU (S-1)
6. Judul Skripsi : ANALISIS PENGARUH PENUKARAN
FASATERHADAP
KETIDAKSEIMBANGAN FASA PADA
GARDU INDUK BLIMBING DENGAN
MENGUNAKAN METODE
SIMULATED ANNEALING


7. Tanggal Mengajukan Skripsi : 20 DESEMBER 2005
8. Tanggal Menyelesaikan Skripsi : 8 MARET 2006
9. Dosen Pembimbing : Ir. H. Taufik Hidayat, MT
10. Telah Dievaluasi dengan nilai : 85 (delapan puluh lima)

Mengetahui,
Ketua Jurusan Teknik Elektro


Ir. F. YUDI LIMPRAPTONO, MT
Nip. 103 9500 274

Malang, Maret 2006

Disetujui
Dosen Pembimbing


Ir. H. TAUFIK HIDAYAT, MT
Nip. 101 8700 151



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

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

1. Nama Mahasiswa : DIDIK PURNOMO
2. N.I.M. : 01.12.128
3. Jurusan : TEKNIK ELEKTRO
4. Konsentrasi : TEKNIK ENERGI LISTRIK (S-1)
5. Judul Skripsi : ANALISIS PENGARUH PENUKARAN FASA
TERHADAP KETIDAKSEIMBANGAN FASA PADA
GARDU INDUK BLIMBING DENGAN
MENGUNAKAN METODE SIMULATED
ANNEALING

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

Hari : Rabu
Tanggal : 22 Maret 2006
Dengan Nilai : 81,55 (A) *g*



(Ir. Mochtar Asroni, MSME)
Ketua

Panitia Ujian Skripsi

(Ir. F. Yudi Limpraptono, MT)
Sekretaris

Anggota Penguji

(Ir. I Made Wartana, MT)
Penguji I

(Ir. M. Abdul Hamid, MT)
Penguji II