

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



**REKONFIGURASI JARINGAN UNTUK MEREDUKSI
RUGI-RUGI SISTEM DISTRIBUSI MENGGUNAKAN
METODE ARTIFICIAL NEURAL NETWORK**

SKRIPSI

**Disusun oleh :
DIANA RACHMANNINDIRA
NIM : 99.12.041**



OKTOBER 2005

LEMBAR PERSETUJUAN

**REKONFIGURASI JARINGAN UNTUK MEREDUKSI RUGI-RUGI
SISTEM DISTRIBUSI MENGGUNAKAN METODE *ARTIFICIAL
NEURAL NETWORK* DI GI BLIMBING**

SKRIPSI

*Disusun Guna Melengkapi dan Memenuhi Syarat-Syarat
Guna Mencapai Gelar Sarjana Teknik Elektro S-1*

Disusun Oleh :

**DIANA RACHMANNINDIRA
NIM. 99.12.041**



**Mengetahui,
Ketua Jurusan Teknik Elektro S-1**

**(Ir. F. Yudi Limpraptono, MT)
NIP. Y. 10 9500274**

**Diperiksa dan Disetujui
Dosen Pembimbing**

**(Ir. I Made Wartana, MT)
NIP. 131 991 182**

20
12 '05

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

ABSTRAKSI

REKONFIGURASI JARINGAN UNTUK MEREDUKSI RUGI- RUGI SISTEM DISTRIBUSI MENGGUNAKAN METODE *ARTIFICIAL NEURAL NETWORK*

(Diana Rachmannindira, 99.12.041, Teknik Elektro Energi Listrik)

(Dosen Pembimbing : Ir. I Made Wartana, MT)

Kata Kunci : Rekonfigurasi Jaringan Distribusi, Reduksi Rugi Daya, *Artificial Neural Network*.

Sistem distribusi tenaga listrik adalah sarana penyampaian tenaga listrik dari sumber ke pusat beban/konsumen. Keberhasilan dalam sistem distribusi dapat terjamin bila dipenuhinya beberapa persyaratan antara lain mengenai kontinuitas pelayanan dan fleksibilitas terhadap pertumbuhan beban. Kelemahan sistem distribusi yang hampir semuanya menggunakan sistem radial yakni sulitnya memenuhi aspek teknis. Hal ini karena lokasi beban dan variasi dari kerapatan beban yang menyebabkan tingginya rugi-rugi daya dari sistem distribusi tersebut dan masalah dalam profil tegangan, terutama pada ujung penyulang.

Rekonfigurasi jaringan distribusi radial merupakan salah satu solusi untuk mereduksi rugi-rugi daya tersebut serta memperbaiki profil tegangan. Banyak metode rekonfigurasi yang telah dikembangkan. Skripsi ini memberi metode alternatif rekonfigurasi menggunakan metode *Artificial Neural Network*. Dari analisa rekonfigurasi tersebut didapatkan hasil perbandingan rugi daya dimana jaringan distribusi sebelum rekonfigurasi sebesar $P_{Loss} = 0.228$ MW dan $Q_{Loss} = 0.349$ MVAR sedangkan setelah rekonfigurasi sebesar $P_{Loss} = 0.152$ MW dan $Q_{Loss} = 0.232$ MVAR, sehingga dapat direduksi sebesar 33.33% sekaligus profil tegangan dapat diperbaiki sesuai batas-batas yang diijinkan.

KATA PENGANTAR

Dengan Rahmat Tuhan YME serta mengucapkan syukur kehadirat-Nya atas karunia yang telah dilimpahkan kepada penulis sehingga dapat menyelesaikan skripsi yang berjudul **“REKONFIGURASI JARINGAN UNTUK MEREDUKSI RUGI-RUGI SISTEM DISTRIBUSI MENGGUNAKAN METODE *ARTIFICIAL NEURAL NETWORK*”**.

Skripsi ini bertujuan untuk memenuhi kurikulum akademik yang harus ditempuh oleh setiap mahasiswa Institut Teknologi Nasional dalam menempuh jenjang pendidikan Strata Satu pada jurusan Teknik Elektro Konsentrasi Teknik Energi Listrik. Diharapkan dengan penulisan skripsi ini dapat memperdalam dan sekaligus melatih mahasiswa agar dapat menerapkan ilmu yang didapat pada waktu kuliah dengan kondisi yang sesungguhnya.

Penulis menyadari bahwa skripsi ini jauh dari sempurna, oleh karena itu saran dan kritik yang bersifat membangun sangat diharapkan.

Atas tersusunnya skripsi ini tidak lupa penulis mengucapkan terima kasih kepada :

1. Dr. Ir. Abraham Lomi, MSEE, selaku Rektor ITN Malang.
2. Ir. F. Yudi Limpratono, MT, selaku Ketua Jurusan Teknik Elektro ITN Malang.
3. Ir. I Made Wartana, MT, selaku Dosen Pembimbing.

Akhirnya penyusun mohon maaf yang sebesar-besarnya kepada semua pihak apabila ada kesalahan dan kekurangan selama penyusunan skripsi ini.

Malang, Oktober 2005

Penyusun

DAFTAR ISI

| | |
|--|------|
| HALAMAN JUDUL | i |
| LEMBAR PERSETUJUAN | ii |
| ABSTRAK | iii |
| KATA PENGANTAR | iv |
| DAFTAR ISI | v |
| DAFTAR GAMBAR | viii |
| DAFTAR GRAFIK | ix |
| DAFTAR TABEL | x |
| BAB I PENDAHULUAN | |
| 1.1. Latar Belakang | 1 |
| 1.2. Rumusan Masalah | 2 |
| 1.3. Tujuan Pembahasan | 2 |
| 1.4. Batasan Masalah | 3 |
| 1.5. Metode Pembahasan | 3 |
| 1.6. Sistematika Penulisan | 4 |
| 1.7. Relevansi | 5 |
| BAB II SISTEM DISTRIBUSI TENAGA LISTRIK | |
| 2.1. Tinjauan Umum | 6 |
| 2.2. Sistem Distribusi | 7 |
| 2.2.1. Struktur Jaringan Radial | 8 |
| 2.2.2. Struktur Jaringan Loop | 10 |

| | |
|--|----|
| 2.2.3. Struktur Jaringan Mesh..... | 11 |
| 2.3. Macam-macam Struktur Jaringan Distribusi Radial..... | 12 |
| 2.4. Daya Sistem Distribusi..... | 12 |
| 2.4.1. Daya Nyata (<i>Real Power</i>)..... | 12 |
| 2.4.2. Daya Reaktif (<i>Reaktif Power</i>)..... | 13 |
| 2.4.3. Daya Kompleks (<i>Compleks Power</i>)..... | 13 |
| 2.5. Beban Sistem Distribusi..... | 13 |
| 2.6. Rugi-rugi Jaringan..... | 14 |
| | |
| BAB III METODE REKONFIGURASI JARINGAN DISTRIBUSI UNTUK MEREDUKSI RUGI DAYA MENGGUNAKAN <i>ARTIFICIAL NEURAL NETWORK</i> | |
| 3.1. Analisa Aliran Daya..... | 16 |
| 3.1.1. Tujuan..... | 16 |
| 3.1.2. Klasifikasi Bus..... | 17 |
| 3.1.3. Pembentukan Matriks admitansi Simpul..... | 18 |
| 3.1.4. Penyelesaian Aliran Daya..... | 19 |
| 3.1.5. Metode <i>Newton Raphson</i> | 21 |
| 3.2. Metode Rekonfigurasi..... | 26 |
| 3.3. Metode <i>Artificial Neural Network</i> | 27 |
| 3.4. Fungsi Aktivasi..... | 30 |
| 3.5. Proses Pembelajaran..... | 31 |
| 3.6. <i>Backpropagation</i> | 32 |
| 3.7. Algoritma Program Rekonfigurasi Jaringan Menggunakan Metode <i>Artificial Neural Network</i> | 38 |

BAB IV HASIL DAN ANALISIS HASIL

| | |
|--|----|
| 4.1. Program Komputer Untuk Rekonfigurasi Jaringan | 39 |
| 4.2. Flowchart Rekonfigurasi Jaringan Menggunakan <i>Artificial Neural Network</i> | 40 |
| 4.3. Flowchart Pembelajaran <i>Backpropagation</i> | 41 |
| 4.4. Data Jaringan | 42 |
| 4.3. Analisa Perhitungan Sebelum dan Sesudah Rekonfigurasi | 45 |
| 4.3.1. Analisa Perhitungan Sebelum Rekonfigurasi | 46 |
| 4.3.2. Analisa Perhitungan Sesudah Rekonfigurasi | 47 |
| 4.4. Tampilan Program | 49 |

BAB V KESIMPULAN DAN SARAN

| | |
|-----------------------|----|
| 5.1. Kesimpulan | 56 |
| 5.2. Saran | 57 |

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar :

| | | |
|------|---|----|
| 2-1 | Diagram Satu garis Sistem Tenaga Listrik..... | 7 |
| 2-2 | Diagram satu garis JTM, JTR dan Sambungan ke Pelanggan..... | 8 |
| 2-3 | Struktur jaringan Radial..... | 10 |
| 2-4 | Struktur Jaringan <i>Loop</i> | 11 |
| 2-5 | Struktur Jaringan <i>Mesh</i> | 12 |
| 2-6 | Rugi-rugi Pada Bagian Sistem Tenaga..... | 14 |
| 3-1 | Diagram skematik tipe Switch (a) NC dan (b) NO | 27 |
| 3-2 | Susunan Syaraf Manusia | 28 |
| 3-3 | Struktur Neuron Jaringan Syaraf..... | 29 |
| 3-4 | Arsitektur Jaringan <i>Backpropagation</i> | 33 |
| 4-1 | Flowchart Rekonfigurasi Jaringan Menggunakan <i>ANN</i> | 40 |
| 4-2 | Flowchart Pembelajaran <i>Backpropagation</i> | 41 |
| 4-3 | Konfigurasi Jaringan Sebelum Rekonfigurasi..... | 44 |
| 4-4 | Konfigurasi Jaringan Mesh..... | 45 |
| 4-5 | Konfigurasi Jaringan Radial Setelah Rekonfigurasi..... | 48 |
| 4-6 | Tampilan Menu Utama Program..... | 50 |
| 4-7 | Tampilan Data Sebelum Rekonfigurasi | 51 |
| 4-8 | Tampilan Hasil Perhitungan Sebelum Rekonfigurasi | 52 |
| 4-9 | Tampilan Posisi <i>Switch</i> | 52 |
| 4-10 | Tampilan Hasil Perhitungan Sesudah Rekonfigurasi | 53 |

DAFTAR GRAFIK

Grafik :

| | |
|--|----|
| 4-1 Profil Tegangan Sebelum dan Sesudah Rekonfigurasi..... | 54 |
| 4-2 Perbandingan Pembangkitan, Pembebanan dan Rugi-rugi..... | 55 |

DAFTAR TABEL

Tabel :

| | |
|--|----|
| 4-1 Data Jaringan Sistem Distribusi di GI Blimbing..... | 43 |
| 4-2 Hasil Perhitungan Aliran Daya Sebelum Rekonfigurasi..... | 46 |
| 4-3 Hasil Perhitungan Aliran Daya Setelah Rekonfigurasi..... | 48 |
| 4-4 Perbandingan Rugi-rugi Saluran..... | 49 |

BAB I

PENDAHULUAN

1.1. Latar Belakang

Kemajuan teknologi yang cepat memberikan dampak yang cukup besar pada kebutuhan dan pola konsumsi masyarakat akan energi listrik. Keadaan tersebut berpengaruh pada pola dan sistem penyediaan energi listrik, terutama pada sisi penyediaan tenaga dan pengaturan pendistribusian energi listrik agar didapat suatu sistem yang baik.

Berangkat dari hal diatas, produsen energi listrik harus benar-benar memperhatikan dan mengusahakan penyaluran energi listrik yang baik kepada konsumen, mulai dari sistem transmisi tenaga sampai dengan sistem distribusi. Dalam hal ini yang perlu ditekankan disini adalah sistem distribusi, karena sistem distribusi berhubungan langsung dengan pemakai atau konsumen energi listrik.

Sistem distribusi merupakan sistem penyaluran energi listrik yang paling dekat dengan konsumen. Rugi-rugi daya dalam sistem distribusi merupakan masalah yang klasik dan selalu dicari cara untuk mereduksi rugi-rugi daya dengan mengatur ulang konfigurasi (rekonfigurasi) penyulang-penyulang dengan suatu metode pengaturan tertentu melalui pengaturan membuka dan menutup pemutus daya (*switch*) pada penyulang, sehingga didapatkan rugi-rugi daya yang seminimal mungkin.

Metode rekonfigurasi pada skripsi ini berdasarkan metode *Artificial Neural Network (ANN)* sedangkan metode aliran dayanya menggunakan *Newton Raphson*.

1.2. Rumusan Masalah

Berdasarkan latar belakang tersebut di atas, maka diperlukan upaya untuk mereduksi rugi-rugi daya pada sistem distribusi. Pada sistem distribusi skala besar dan kompleks akibat bertambahnya beban pada jaringan distribusi akan timbul masalah di dalam menentukan penyulang mana yang akan dipakai untuk menyalurkan energi listrik secara radial dengan cepat dan mereduksi rugi-rugi daya pada jaringan, perlu adanya metode alternatif untuk pengaturan status dari *Normally Open Switch* dan *Normally Closed Switch* pada penyulang.

Dengan berdasarkan gambaran masalah di atas, maka skripsi ini diberi judul :

REKONFIGURASI JARINGAN UNTUK MEREDUKSI RUGI-RUGI
SISTEM DISTRIBUSI MENGGUNAKAN METODE *ARTIFICIAL*
NEURAL NETWORK DI GI BLIMBING

1.3. Tujuan Pembahasan

Mengaplikasikan metode *Artificial Neural Network* dalam rekonfigurasi jaringan distribusi radial untuk :

- Memilih operasi *switch* yang tepat.
 - Mereduksi rugi-rugi daya.
 - Memperbaiki profil tegangan.
-

1.4. Batasan Masalah

Dalam skripsi ini akan dilakukan analisa tentang pengaturan ulang (rekonfigurasi) jaringan sistem distribusi.

Adapun batasan masalahnya adalah sebagai berikut :

- Beban diasumsikan sebagai beban tiga fasa seimbang.
- Hanya membahas rugi-rugi daya pada saluran.
- Tidak membahas koordinasi pengaman dan penyebab terjadinya gangguan
- Tidak membahas cara pengontrolan *switch* yang dilakukan.
- Tidak membahas masalah ekonomis.
- Pengambilan data dilakukan pada Trafo II Gardu Induk Blimbing.

1.5. Metode Pembahasan

Metode yang digunakan dalam pembahasan dilaksanakan dengan langkah-langkah sebagai berikut :

- Kajian literatur

Yaitu kajian pustaka untuk mempelajari teori-teori yang terkait melalui literatur yang ada, yang berhubungan dengan permasalahan.

- Pengumpulan Data

Bentuk data yang digunakan adalah :

- Data Kuantitatif, yaitu data yang dapat dihitung atau data yang berbentuk angka yaitu berupa data saluran dan data pembebanan.
 - Data kualitatif, yaitu data yang berbentuk diagram. Dalam hal ini *single line diagram* penyulang.
-

- Simulasi dan Pembahasan Masalah

Analisa perhitungan tegangan, sudut fasa tegangan, rugi-rugi daya dan aliran daya tiap saluran menggunakan metode *Newton Raphson*, sedangkan untuk menentukan rekonfigurasi jaringan sistem distribusi menggunakan metode *Artificial Neural Network* dengan pemrograman komputer.

- Pengambilan kesimpulan dan hasil analisis.

1.6. Sistematika Pembahasan

Untuk memudahkan pembahasan, maka skripsi ini akan dibagi menjadi lima bab dengan sistematika sebagai berikut :

BAB I PENDAHULUAN

Bab ini meliputi latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, sistematika pembahasan dan relevansi.

BAB II SISTEM DISTRIBUSI TENAGA LISTRIK

Bab ini menjelaskan tentang sistem distribusi tenaga listrik, klasifikasi sistem saluran daya listrik, penjelasan tentang jaringan distribusi radial dan karakteristik beban.

BAB III METODE REKONFIGURASI JARINGAN UNTUK MEREDUKSI RUGI-RUGI DAYA MENGGUNAKAN *ARTIFICIAL NEURAL NETWORK*.

Bab ini akan membahas pengenalan awal aliran daya dan rekonfigurasi jaringan dalam upaya mengurangi kerugian daya pada jaringan distribusi radial.

BAB IV HASIL DAN ANALISIS HASIL

Bab ini memuat pembahasan mengenai permasalahan rekonfigurasi jaringan distribusi dengan metode *Artificial Neural Network* dan juga perbandingan hasil perhitungan antara sebelum dan sesudah rekonfigurasi.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.7. Relevansi

Dari analisa yang dilakukan maka sangat diharapkan dapat memberikan informasi mengenai penyelesaian rekonfigurasi jaringan distribusi radial untuk mereduksi rugi-rugi daya pada jaringan. Dimana pada skripsi ini digunakan metode *Artificial Neural Network* dan diharapkan dengan metode penyelesaian tersebut dapat memperoleh hasil yang paling optimal sehingga akhirnya dapat diplikasikan pada instansi-instansi yang terkait sehingga penyaluran tenaga listrik menjadi lebih baik.

BAB II

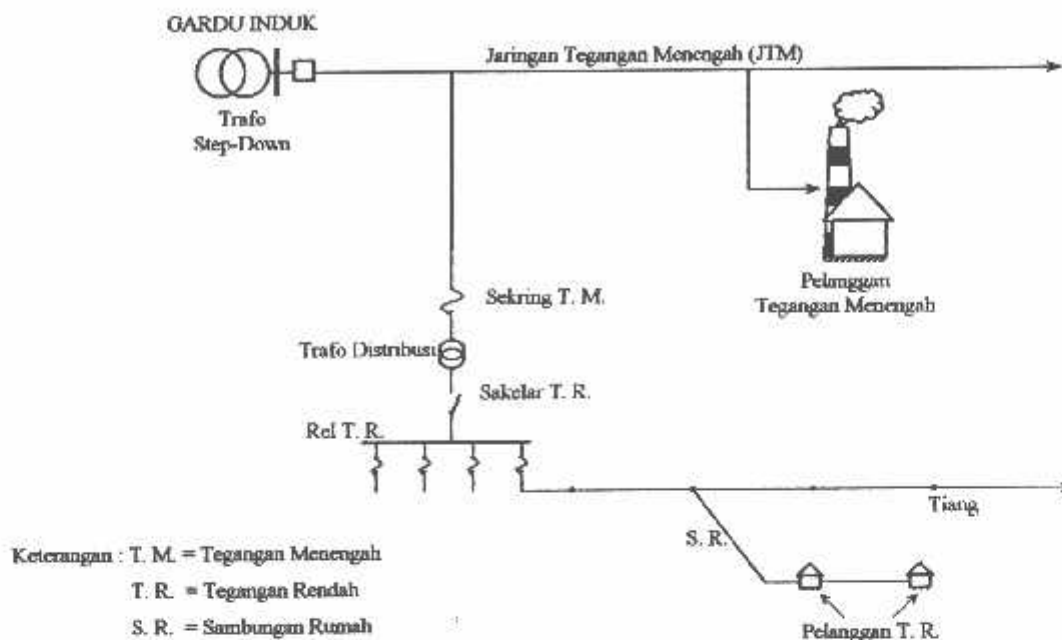
SISTEM DISTRIBUSI TENAGA LISTRIK

2.1. Tinjauan Umum

Sistem tenaga listrik merupakan suatu sistem terpadu yang terbentuk oleh hubungan-hubungan peralatan dan komponen-komponen listrik. Sistem tenaga listrik ini mempunyai peranan utama untuk menyalurkan energi listrik yang dibangkitkan oleh generator ke konsumen yang membutuhkan energi listrik tersebut.

Secara garis besar suatu sistem tenaga listrik dapat dikelompokkan menjadi 3 bagian sub sistem, yaitu :

1. Sistem pembangkitan, berperan sebagai sumber daya tenaga listrik dan disebut juga sebagai Produsen energi.
2. Sistem Transmisi atau Penyaluran, berfungsi sebagai penyalur daya listrik secara besar-besaran dari pembangkit ke bagian sistem distribusi atau konsumen.
3. Sistem Distribusi dan Beban, berperan sebagai distributor energi ke konsumen yang membutuhkan energi.



Gambar 2-2 Jaringan Tegangan Menengah, jaringan Tegangan Rendah dan Sambungan ke Pelanggan.

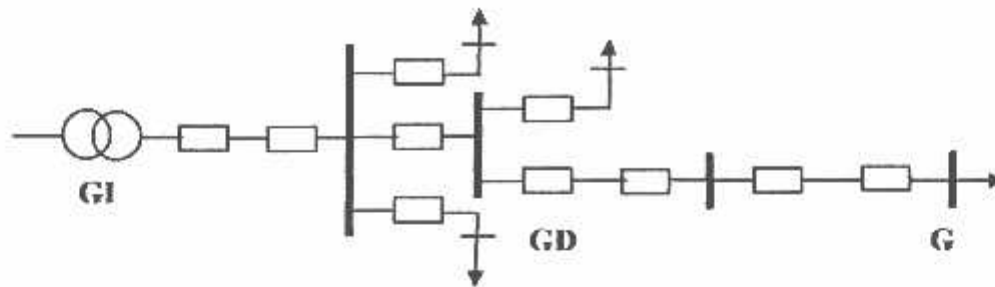
Sumber : Djiteng Marsudi, "Operasi Sistem Tenaga Listrik", Balai Penerbit dan Humas ISTN, 1990.

Secara garis besar jaringan distribusi diklasifikasikan menjadi :

1. Struktur Jaringan Radial
2. Struktur Jaringan Loop (lingkaran)
3. Struktur Jaringan Grid atau Mesh (anyaman)

2.2.1. Struktur Jaringan Radial

Struktur jaringan radial merupakan struktur jaringan yang paling sederhana dan paling murah biaya pembangunannya. Struktur jaringan ini dalam menyalurkan energi listrik, keandalannya kurang. Suatu gangguan pada penyulang dapat mengakibatkan gangguan dalam penyaluran energi listrik ke konsumen yang berada di belakang titik gangguan.

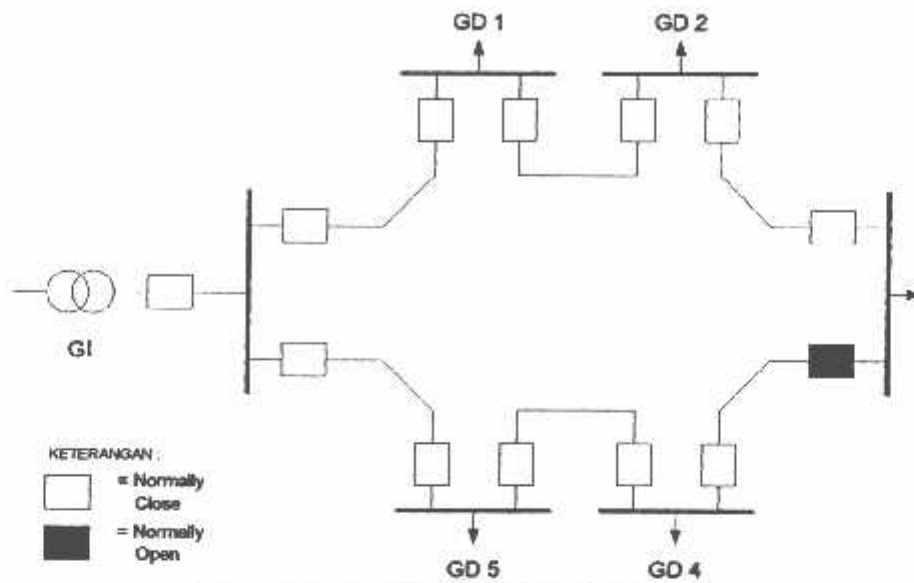


Gambar 2-3 Struktur Jaringan Radial

Sumber : Djiteng Marsudi, "Operasi Sistem Tenaga Listrik", Balai Penerbit dan Humas ISTN, 1990.

2.2.2. Struktur Jaringan Loop

Struktur jaringan loop merupakan gabungan dari 2 struktur jaringan radial, dimana pada ujung kedua jaringan, dipasang sebuah pemutus atau pemisah. Pada saat terjadi gangguan, setelah gangguan dapat diisolir, maka pemutus atau pemisah ditutup, sehingga aliran daya listrik kebagian yang tidak terkena gangguan tidak berhenti. Dalam kondisi normal, jaringan ini merupakan dua struktur jaringan radial. Struktur jaringan ini mempunyai keandalan yang cukup, sehingga biaya pembangunannya lebih mahal dibandingkan dengan biaya struktur jaringan radial.

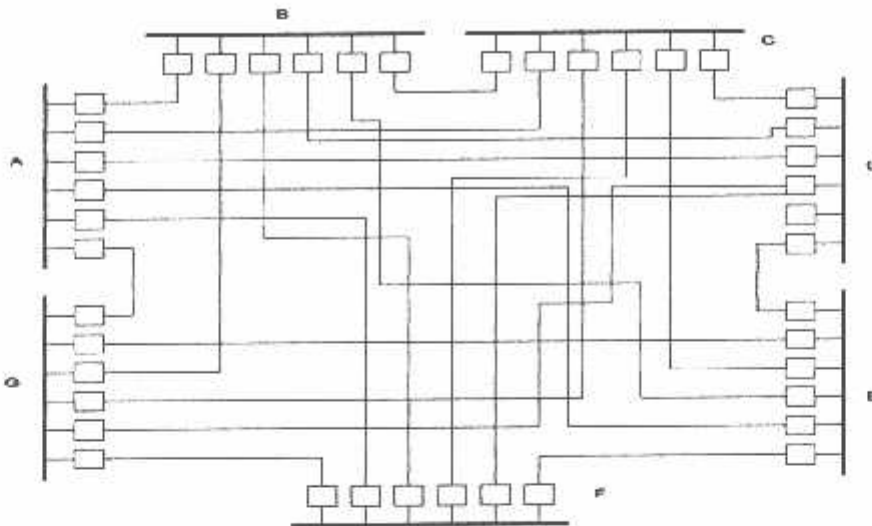


Gambar 2-4 Struktur Jaringan Loop

Sumber : Djiteng Marsudi, "Operasi Sistem Tenaga Listrik", Balai Penerbit dan Humas ISTN, 1990.

2.2.3. Struktur Jaringan Mesh

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



Gambar 2-5 Struktur Jaringan Mesh

Sumber : Djiteng Marsudi, "Operasi Sistem Tenaga Listrik", Balai Penerbit dan Humas ISTN, 1990.

2.3. Macam-macam Struktur Jaringan Distribusi Radial

Tipe jaringan ini merupakan bentuk dasar, susunan maupun kebutuhan alat-alat penunjangnya paling sedikit dan paling sederhana. Tipe ini paling banyak digunakan untuk melayani konsumen terutama beban-beban dari satu titik.

Salurannya dicabang-cabang menuju ketitik-titik beban dan antara titik sumber dengan titik beban hanya ada satu pilihan, dengan demikian bila salah satu saluran cabang mengalami gangguan maka seluruh beban yang ada disaluran tersebut akan mengalami pemadaman total.

2.4. Daya Sistem Distribusi

Daya merupakan banyaknya perubahan tenaga terhadap waktu dalam besaran tegangan dan arus.

2.4.1. Daya Nyata (Real Power)

Daya nyata dinyatakan dalam persamaan :

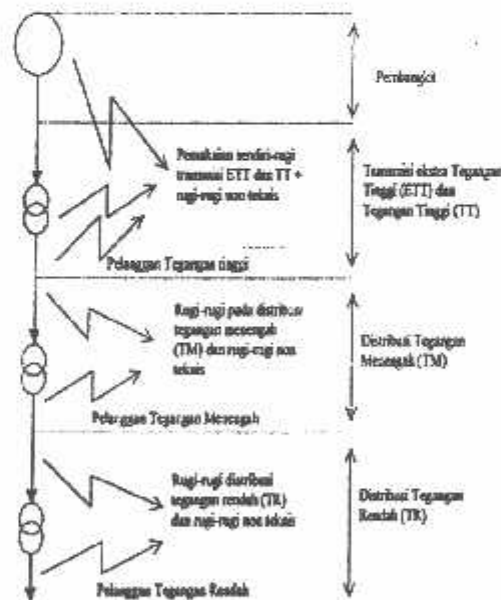
$$P = |V| |I| \cos \varphi \dots\dots\dots (2.1)$$

3. Beban Industri

Beban industri diklasifikasikan menjadi 2, yaitu skala besar dan skala kecil.

2.6. Rugi-rugi Jaringan

Dalam proses penyaluran energi listrik ke pelanggan terjadi rugi-rugi teknis (*losses*), yaitu rugi-rugi daya dan rugi energi, dimulai dari pembangkit, transmisi dan distribusi. Rugi-rugi teknis adalah pada pengantar saluran, adanya tahanan dari pengantar yang dialiri arus, timbulah rugi-rugi teknis (I^2R) pada jaringan tersebut. Pada Gambar 2-6 rugi teknis tersebut terdapat pada saluran transformator. Rugi teknis dari jaringan tenaga listrik tergantung dari macam pembebanan pada saluran tersebut (beban merata, terpusat).



Gambar 2-6 Rugi-rugi Pada Bagian Sistem Tenaga

Sumber : Djiteng Marsudi, "Operasi Sistem Tenaga Listrik", Balai Penerbit dan Humas ISTN, 1990.

Keterangan gambar :



Dari permasalahan tersebut, maka diperlukan suatu upaya untuk mengurangi kerugian daya yang terjadi. Dalam prakteknya, sebenarnya ada 2 macam cara upaya untuk mengurangi/mereduksi rugi-rugi daya yang terjadi pada sistem distribusi tenaga listrik, yaitu :

1. Pemasangan kapasitor.
2. Rekonfigurasi jaringan distribusi.

Untuk selanjutnya pada skripsi ini akan dibahas mengenai upaya untuk mereduksi kerugian yang terjadi pada saluran dengan rekonfigurasi jaringan distribusi yaitu dengan memanfaatkan status dari *NO Switch/NC Switch*.

BAB III

METODE REKONFIGURASI JARINGAN DISTRIBUSI UNTUK MEREDUKSI RUGI DAYA MENGGUNAKAN *ARTIFICIAL NEURAL NETWORK*

Dalam menganalisa metode rekonfigurasi 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 rekonfigurasi adalah analisa aliran daya menggunakan Metode *Newton Raphson*.

3.1. Analisa Aliran Daya

3.1.1. Tujuan

Dalam analisa aliran daya dilakukan perhitungan tegangan, arus, daya nyata dan daya reaktif yang terdapat pada berbagai titik dalam suatu jala-jala listrik pada keadaan pengopersian normal untuk sekarang dan akan datang.

Tujuan dari aliran daya :

1. Untuk menentukan daya nyata dan daya reaktif.
2. Untuk mengetahui apakah semua peralatan pada sistem memenuhi batas-batas yang telah ditetapkan untuk operasi penyaluran daya.
3. Untuk mengetahui kondisi awal pada perencanaan sistem yang baru.
4. Untuk mengetahui daya yang mengalir di tiap saluran jaringan tenaga listrik.

Dalam analisa aliran daya terdapat 3 jenis variabel, yaitu :

1. Variabel bebas, misalnya $|V|$ dan δ pada bus beban atau δ 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.

3.1.2. Klasifikasi Bus

Tujuan aliran daya pada sistem tenaga listrik adalah untuk menghitung besar tegangan $|V|$ dan sudut fasa tegangan δ pada semua bus, sehingga dengan diketahui 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 satuannya MW.
2. Daya reaktif, dinyatakan dengan Q satuannya MVAR.
3. Besar magnitude tegangan, dinyatakan dengan $|V|$ satuannya kV.
4. Sudut fasa tegangan, dinyatakan dengan δ satuannya derajat.

Untuk 1 dan 2 menyatakan daya yang dibangkitkan oleh generator yang mengalir ke bus. Jika pada bus terdapat beban, maka daya tersebut menyatakan selisih antara daya yang dibangkitkan dengan 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 nyata P dan daya reaktif Q diketahui, sementara magnitude tegangan $|V|$ dan sudut fasa δ dicari (dihitung).

2. Bus generator

Pada bus ini daya nyata P dan magnitudo tegangan $|V|$ diketahui besarnya, sementara daya reaktif Q dan sudut fasa δ dihitung.

3. Bus berayun

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

3.1.3. Pembentukan Matriks Admitansi Simpul

Suatu rangkaian yang mempunyai n simpul (termasuk simpul referensi) dapat ditulis dalam bentuk persamaan arus :

$$I_1 = Y_{11}V_1 + Y_{12}V_2 + \dots + Y_{1n}V_n$$

$$I_2 = Y_{21}V_1 + Y_{22}V_2 + \dots + Y_{2n}V_n$$

$$I_n = Y_{n1}V_1 + Y_{n2}V_2 + \dots + Y_{nn}V_n$$

atau
$$I_p = \sum_{q=1}^n Y_{pq}V_{pq} \dots \dots \dots (3.1)$$

dimana : P = 1, 2, 3, ..., n. n adalah jumlah simpul yang ada dalam sistem.

I_p = arus yang masuk ke simpul P.

V_q = tegangan simpul q terhadap referensi.

Y_{pq} = admitansi antara simpul p dan q.

Persamaan diatas dapat ditulis dalam bentuk matrik sebagai berikut :

$$I_{bus} = Y_{bus} V_{bus} \dots \dots \dots (3.2)$$

dimana : I_{bus} = matrik kolom vektor arus simpul.

Y_{bus} = matrik kolom vektor admitansi simpul.

V_{bus} = matrik kolom vektor tegangan simpul.

Elemen dari Y_{bus} matrik adalah arus yang mengalir pada simpul jaringan dimana terdapat unit sumber tegangan atau simpul referensi.

Untuk elemen diagonal Y_{pp} , unit sumber tegangan dipasang pada simpul terhadap referensi dan diukur arus yang melalui elemen tersebut, dimana semua simpul yang lain dihubung singkat, sedangkan elemen bukan diagonal Y_{pq} diukur arus pada simpul p dimana unit sumber tegangan dipasang pada simpul q, sedangkan simpul yang lain dihubung singkat, jika p dan q tidak dihubung secara langsung maka arus yang mengalir akan sama dengan nol atau Y_{pq} sama dengan nol.

3.1.4. Penyelesaian Aliran Daya

Untuk penyelesaian masalah aliran daya diperlukan suatu persamaan daya aktif dan daya reaktif yang dinyatakan dalam besaran kompleks. Dimana untuk jaringan sistem tenaga listrik pada tiap-tiap bus dapat dituliskan sebagai berikut :

$$I_1 = Y_{11}V_1 + Y_{12}V_2 + \dots + Y_{1n}V_n$$

$$I_2 = Y_{21}V_1 + Y_{22}V_2 + \dots + Y_{2n}V_n$$

$$I_p = Y_{p1}V_1 + Y_{p2}V_2 + \dots + Y_{pn}V_n$$

Dimana persamaan tersebut dapat dituliskan menjadi bentuk umum :

$$I_p = \sum_{q=1}^n Y_{pq} V_{pq} \dots \dots \dots (3.3)$$

dimana : P = 1, 2, 3, ..., n, n adalah jumlah simpul yang ada dalam sistem.

I_p = arus dalam bentuk kompleks yang masuk bus p.

V_q = tegangan bus q dalam bentuk kompleks terhadap referensi.

Y_{pq} = admitansi antara simpul p dan q.

Dalam bentuk matrik persamaan diatas dapat ditulis sebagai berikut :

$$\begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_n \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} & \dots & Y_{1n} \\ Y_{21} & Y_{22} & \dots & Y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ Y_{n1} & Y_{n2} & \dots & Y_{nn} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{bmatrix}$$

Atau dapat juga ditulis : $I_{bus} = Y_{bus} V_{bus}$ (3.4)

Dimana : I_{bus} = matrik kolom vektor arus bus.

Y_{bus} = matrik kolom vektor admitansi bus.

V_{bus} = matrik kolom vektor tegangan bus.

Persamaan daya nyata dan daya reaktif yang masuk ke bus adalah :

$$P_p - jQ_p = V_p I_p$$
 (3.5)

Dengan mensubstitusikan persamaan (3.3), ke persamaan (3.5) maka diperoleh persamaan sebagai berikut :

$$P_p - jQ_p = V_p \sum_{q=1}^n Y_{pq} V_{pq}$$
 (3.6)

$$V_p = e_p - jF_p = |V_p| \angle -\delta_p$$

$$V_q = e_q - jF_q = |V_q| \angle -\delta_q$$

$$Y_{pq} = G_{pq} - jB_{pq} = |Y_{pq}| \angle -\theta_q$$

Dengan demikian persamaan ini dapat dinyatakan dalam bentuk rectangular sebagai berikut :

$$P_p - jQ_p = (e_p - jF_p) \sum_{q=1}^n (G_{pq} - jB_{pq}) (e_q + jF_q)$$
 (3.7)

Dimana : $p = 1, 2, 3, \dots, n$.

Persamaan terakhir menyatakan dua buah persamaan aliran daya pada sistem dengan n bus, dimana tiap-tiap bus digolongkan dalam $4n$ variabel dan $2n$ variabel yang lain telah ditentukan terlebih dahulu.

Variabel dari bus berayun telah ditentukan terlebih dahulu sehingga persamaan untuk bus berayun ini dapat dihilangkan dan dengan demikian hanya dibutuhkan $(n-1)$ buah persamaan untuk studi aliran daya. Bentuk persamaan dua terakhir +1 menyatakan hubungan matematis yang diperlukan untuk analisa aliran daya dengan komputer.

Selain itu persamaan diatas hanya berlaku untuk sistem satu fasa dan untuk sistem tiga fasa yang seimbang, dengan satuan yang dipakai adalah per unit, sebab sistem per unit tidak akan terpengaruh oleh perbedaan tegangan pada sistem jaringan yang sesungguhnya.

3.1.5. Metode *Newton Raphson*

Metode *Newton Raphson* dibentuk berdasarkan matrik admitansi simpul (Y_{bus}), sehingga merupakan suatu prosedur yang langsung dan sederhana.

Pada admitansi simpul elemen diagonalnya (Y_{pp}) adalah jumlah admitansi dari semua elemen-elemen jaringan yang terhubung dengan simpul p tersebut. Untuk elemen bukan diagonal (Y_{pq}) adalah sama dengan negatif admitansi dari elemen jaringan yang menghubungkan simpul p dan simpul q .

Karena pada jaringan sistem tenaga listrik tidak semua simpul saling berhubungan satu dengan yang lainnya, maka Y_{bus} akan membentuk matrik yang terdiri dari elemen-elemen yang mempunyai nilai tidak sama dengan nol (diantara

simpul-simpul tersebut tidak mempunyai hubungan saluran) dan elemen-elemen yang bernilai sama dengan nol (diantara simpul-simpul tersebut tidak mempunyai hubungan saluran). Secara matematis persamaan aliran daya metode *Newton Raphson* dapat menggunakan koordinat rectangular, koordinat polar atau bentuk hibrid (gabungan antara bentuk kompleks dengan bentuk polar). Dalam pembahasan ini digunakan bentuk koordinat polar.

Hubungan antara arus simpul I_p dengan tegangan simpul V_q pada suatu jaringan dengan n simpul dapat dituliskan :

$$I_p = \sum_{q=1}^n Y_{pq} V_{pq} \dots\dots\dots (3.8)$$

Injeksi daya pada simpul p adalah :

$$S_p = P_p - jQ_p = V_p^* I_p \dots\dots\dots (3.9)$$

$$= V_p^* \sum_{q=1}^n Y_{pq} V_{pq} \dots\dots\dots (3.10)$$

Dalam penyelesaian aliran daya dengan *Newton Raphson* bentuk persamaan aliran daya yang dipilih adalah polar, dimana tegangan dinyatakan dalam bentuk polar, yaitu :

$$V_p^* = |V_p| e^{-j\delta_p}$$

$$V_q = |V_q| e^{j\delta_q}$$

$$Y_{pq} = |Y_{pq}| e^{j\delta_{pq}}$$

Maka persamaan (3.6) dapat ditulis :

$$P_p - jQ_p = \sum_{q=1}^n |V_p| |V_q| |Y_{pq}| e^{j(\delta_p - \delta_q + \theta_{pq})} \dots\dots\dots (3.11)$$

Dengan memisahkan bagian riil dan bagian imajiner maka diperoleh :

$$P_p = \sum_{q=1}^n |V_p V_q Y_{pq}| \cos(\delta_p - \delta_q + \theta_{pq}) \dots\dots\dots (3.12)$$

$$Q_p = \sum_{q=1}^n |V_p V_q Y_{pq}| \sin(\delta_p - \delta_q + \theta_{pq}) \dots\dots\dots (3.13)$$

Kedua persamaan diatas akan menghasilkan suatu kesimpulan persamaan serentak (simultan) yang tidak linier untuk setiap simpul sistem tenaga listrik. Untuk mengetahui magnitude tegangan (V) dan sudut fasa (δ) disetiap simpul dapat diselesaikan dengan menggunakan persamaan (3.12) dan (3.13) yang dilinierkan dengan metode *Newton Raphson* seperti persamaan berikut :

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |V| \end{bmatrix} \dots\dots\dots (3.14)$$

Dimana :

ΔP = selisih injeksi bersih daya nyata dengan penjumlahan aliran daya nyata tiap saluran yang menghubungkan simpul dengan V yang didapat dari perhitungan iterasi ke-k.

ΔQ = selisih injeksi bersih daya reaktif dengan penjumlahan aliran daya reaktif tiap saluran yang menghubungkan simpul dengan V yang didapat dari perhitungan iterasi ke-k.

$\Delta \delta$ = vektor koreksi sudut fasa tegangan.

$\Delta |V|$ = vektor koreksi magnitude tegangan .

J_1, J_2, J_3, J_4 merupakan elemen *off diagonal* dan *diagonal* di sub matrik *Jacobian* yang dibentuk dengan mendefinisikan persamaan (3.12) dan (3.13), dimana :

$$J_1 = \frac{\partial P_p}{\partial \delta_q} \qquad J_2 = \frac{\partial P_p}{\partial |V_q|} \qquad J_3 = \frac{\partial Q_p}{\partial \delta_q} \qquad J_4 = \frac{\partial Q_p}{\partial |V_q|}$$

Adapun rumus dari elemen matrik *Jacobian* adalah :

Untuk J_1 :

$$\frac{\partial P_p}{\partial \delta_q} = |V_p V_q Y_{pq}| \sin(\delta_p - \delta_q + \theta_{pq}) \quad q \neq p$$

$$\frac{\partial P_p}{\partial \delta_p} = -\sum_{q=1}^n |V_p V_q Y_{pq}| \sin(\delta_p - \delta_q + \theta_{pq}) \quad q \neq p$$

Untuk J_2 :

$$\frac{\partial P_p}{\partial |V_p|} = |V_p Y_{pq}| \cos(\delta_p - \delta_q + \theta_{pq}) \quad q \neq p$$

$$\frac{\partial P_p}{\partial |V_p|} = 2|V_p Y_{pq}| \cos \theta_{pq} + \sum_{q=1}^n |V_q - Y_{pq}| \cos(\delta_p - \delta_q + \theta_{pq})$$

Untuk J_3 :

$$\frac{\partial Q_p}{\partial \delta_q} = -|V_p V_q Y_{pq}| \cos(\delta_p - \delta_q + \theta_{pq}) \quad q \neq p$$

$$\frac{\partial Q_p}{\partial \delta_p} = -\sum_{q=1}^n |V_p V_q Y_{pq}| \cos(\delta_p - \delta_q + \theta_{pq}) \quad q \neq p$$

Untuk J_4 :

$$\frac{\partial Q_p}{\partial |V_q|} = |V_p Y_{pq}| \sin(\delta_p - \delta_q + \theta_{pq}) \quad q \neq p$$

$$\frac{\partial Q_p}{\partial |V_p|} = 2|V_p Y_{pq}| \sin \theta_{pq} + \sum_{q=1}^n |V_q - Y_{pq}| \sin(\delta_p - \delta_q + \theta_{pq})$$

Untuk menghitung selisih daya, maka mula-mula ditentukan nilai awal tegangan simpul dan sudut fasanya. Kemudian daya nyata dan daya reaktif dihitung dengan menggunakan persamaan (3.12) dan (3.13). selisih antara daya

yang telah ditentukan dengan daya hasil perhitungan ini merupakan perubahan daya yang terjadi pada simpulnya.

$$\Delta P_p = P_{p, \text{specified}} - P_{p, \text{calculated}} \dots\dots\dots (3.15)$$

$$\Delta Q_p = Q_{p, \text{specified}} - Q_{p, \text{calculated}} \dots\dots\dots (3.16)$$

Magnitude tegangan $|V|$ dan sudut fasa δ_p yang diasumsikan serta selisih daya yang dihitung (ΔP_p dan ΔQ_p) digunakan untuk memperoleh elemen-elemen matrik *Jacobian*.

Persamaan (3.14) diselesaikan untuk menghitung vektor koreksi magnitude tegangan ($\Delta |V|$) dan sudut fasa tegangan ($\Delta \delta$) yang baru. Sehingga diperoleh harga magnitude tegangan dan sudut fasa tegangan yang baru yaitu :

$$|V|^{k+1} = |V|^k + \Delta |V|^k$$

$$\delta^{k+1} = \delta^k + \Delta \delta^k$$

Proses perhitungan akan berulang sampai selisih daya nyata dan daya reaktif antara yang dijadwalkan dengan yang dihitung, yaitu ΔP dan ΔQ untuk semua simpul mendekati nilai toleransi atau proses perhitungan iterasi mencapai konvergen.

Prosedur Aliran Daya *Newton Raphson*

1. Tentukan nilai P_p (ditetapkan) dan Q_p (ditetapkan) yang mengalir ke dalam sistem pada setiap rel untuk nilai yang ditentukan atau diperkirakan dari tegangan nyata dan tegangan reaktif untuk iterasi pertama atau tegangan yang ditentukan paling akhir untuk iterasi berikutnya.
2. Hitung ΔP dan ΔQ pada setiap rel.

3. Hitung nilai-nilai untuk *Jacobian* dengan menggunakan nilai-nilai perkiraan atau yang ditentukan dari tegangan nyata dan tegangan reaktif dalam persamaan untuk konsumen parsial yang ditentukan.
4. Balikkan *Jacobian* itu dan hitung koneksi magnitudo tegangan $\Delta(|V|)$ dan sudut fasa ($\Delta\delta_p$) pada setiap rel.
5. Hitung nilai baru dari $|V_p|$ dan δ_p dengan menambahkan $\Delta|V_i|$ dan $\Delta\delta_i$ pada nilai sebelumnya.
6. Kembali kelangkah 1 dan ulangi proses ini dengan menggunakan nilai untuk tegangan nyata dan tegangan reaktif yang ditentukan paling akhir sehingga nilai ΔP dan ΔQ lebih kecil atau sama dengan nilai toleransi yang telah ditetapkan.

3.2. Metode Rekonfigurasi Jaringan

Proses menata konfigurasi awal dari jaringan (*initial condition*) menjadi suatu konfigurasi optimum akhir (*optimal condition*), sehingga dari konfigurasi yang terakhir ini diperoleh kerugian daya sistem distribusi yang paling kecil (aliran daya yang paling optimum).

Meskipun jaringan distribusi dioperasikan menggunakan sistem radial, namun besarnya sistem tersebut dikombinasikan ke dalam beberapa tipe. Tipe yang paling umum adalah model jaringan distribusi radial dengan *Tie* dan *Switch* pemisah.

Rekonfigurasi jaringan distribusi dapat dilakukan dengan mengatur kondisi dari *Switch-switch* ini dalam keadaan *ON* atau *OFF*. *Switch* yang terdapat dalam jaringan distribusi terdapat 2 macam :

1. *Normally closed switch (NC switch)* yang dalam keadaan operasi normal, posisi kontakannya selalu tertutup (*ON*).
2. *Normally open switch (NO switch)* yang dalam keadaan operasi normal, posisi kontakannya selalu terbuka (*OFF*).



Gambar 3-1 Diagram Skematik Tipe (a) *NC switch* dan (b) *NO switch*.

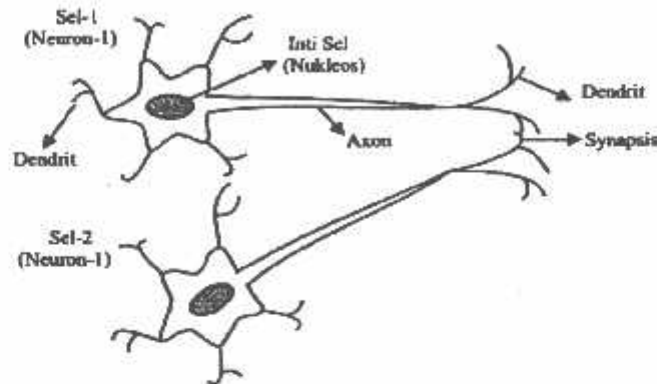
Sumber : Djiteng Marsudi, "Operasi Sistem Tenaga Listrik", Balai Penerbit dan Humas ISTN, 1990.

Jadi jelaslah bahwa sebenarnya rekonfigurasi jaringan dalam sistem distribusi tenaga listrik adalah operasi mengganti struktur topologi dari jaringan distribusi dengan mengubah status *open/closed* dari *NO/NC switch* sehingga dari proses ini bisa didapatkan aliran daya yang optimal pada sistem distribusi yang menyebabkan kerugian daya jadi minimum dengan proses akhir sistem kembali radial.

3.3. Metode *Artificial Neural Network*^[5]

Artificial Neural Network adalah merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan ini digunakan karena

jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah perhitungan selama proses belajar.



Gambar 3-2 Susunan syaraf manusia

Sumber : Artificial Intelligent, Sri Kusuma Dewi, Hal. 207

Susunan sel syaraf manusia, setiap sel syaraf akan memiliki 1 inti sel, sel ini nanti yang akan bertugas untuk melakukan pemrosesan informasi yang datang akan diterima oleh dendrit. Selain menerima informasi, dendrit juga menyertai axon sebagai keluaran pemrosesan suatu informasi. Informasi hasil olahan akan menjadi masukan bagi neuron yang lain yang mana antara dendrit kedua sel tersebut dipertemukan dengan synapsis. Informasi yang dikirimkan antar neuron ini berupa rangsangan yang dilewatkan melalui dendrit. Informasi yang datang akan diterima oleh dendrit akan dijumlahkan dan akan dikirim melalui axon ke dendrit akhir yang bersentuhan dengan dendrit dari neuron lain. Informasi ini akan diterima oleh neuron lain jika memenuhi batasan tertentu, yang sering dikenal dengan nama nilai ambang (*threshold*). Pada kasus ini, neuron tersebut dikatakan aktivasi. Hubungan antara neuron terjadi secara adaptif artinya struktur hubungan

tersebut terjadi secara dinamis. Otak manusia selalu memiliki kemampuan untuk belajar dengan melakukan adaptasi. Seperti halnya otak manusia jaringan syaraf juga terdiri dari beberapa neuron, dan terdapat hubungan antara neuron-neuron tersebut. Neuron-neuron tersebut akan mentransformasikan informasi yang diterima melalui sambungan, keluarnya melalui neuron-neuron yang lain. Pada jaringan syaraf, hubungan ini diketahui dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut.



Gambar 3-3 Struktur Neuron Jaringan Syaraf.

Sumber : Artificial intelligent, Sri kusuma Dewi , hal 210

Neuron buatan ini bekerja dengan cara yang sama pula dengan neuron biologis. Informasi tersebut (disebut dengan input) akan dikirim ke neuron dengan bobot kedatangan tertentu. Input ini akan diproses oleh fungsi perambatan yang akan menjumlahkan nilai-nilai bobot yang akan datang. Hasil penjumlahan ini kemudian akan dibandingkan dengan suatu hasil ambang (*threshold*) tertentu melalui fungsi aktivasi setiap neuron. Apabila input melewati suatu nilai ambang tertentu, maka neuron akan diaktifkan, tetapi kalau tidak maka neuron tersebut tidak akan diaktifkan. Apabila neuron tersebut diaktifkan, maka neuron tersebut

akan mengirimkan output melalui bobot-bobot outputnya ke semua neuron yang berhubungan dengannya. Demikian seterusnya.

Pada *Artificial Neural Network* neuron-neuron akan dikumpulkan dalam lapisan (*layer*) yang disebut dengan lapisan neuron (*neuron layers*). Biasanya neuron-neuron pada satu lapisan akan dihubungkan dengan lapisan sebelum dan sesudahnya (kecuali lapisan input dan lapisan output). Informasi yang diberikan pada jaringan syaraf akan dirambatkan lapisan ke lapisan, mulai dari lapisan input sampai lapisan output melalui lapisan yang lainnya, yang sering disebut dengan nama lapisan tersembunyi (*hidden layer*).

3.4. Fungsi Aktivasi

Ada beberapa fungsi aktivasi yang sering digunakan dalam *Artificial Neural Network*, antara lain :

- a. fungsi Undak Biner (*Hard Limit*).
 - b. Fungsi Undak Biner (*Threshold*).
 - c. Fungsi Bipolar (*Symetric Hard Limit*).
 - d. Fungsi Bipolar (*Threshold*).
 - e. Fungsi Linier (*Identitas*).
 - f. Fungsi Saturating Linier.
 - g. Fungsi Symetric Saturating Linier.
-

h. Fungsi Sigmoid Biner.

i. Fungsi Sigmoid Bipolar.

Fungsi yang digunakan adalah fungsi sigmoid biner. Fungsi sigmoid biner dirumuskan sebagai :

$$f(x) = \frac{1}{1 + e^{-\sigma x}} \dots\dots\dots(3.13)$$

dengan $f'(x) = \sigma f(x) [1 - f(x)]$

σ = adalah parameter kecuraman fungsi.

$f(x)$ = x (untuk semua x, merupakan fungsi aktivasi untuk semua input unit).

Fungsi sigmoid biner berbentuk kurva s. Fungsi ini memiliki nilai pada interval 0 sampai 1.

3.5. Proses Pembelajaran

Pada jaringan syaraf akan mencoba untuk mensimulasikan kemampuan otak manusia untuk belajar. *Artificial Neural Network* juga tersusun atas neuron-neuron dan dendrit. Tidak seperti model biologis, jaringan syaraf memiliki struktur yang tidak dapat dirubah, dibangun oleh sejumlah neuron dan memiliki nilai tertentu yang menunjukkan seberapa besar koneksi antar neuron (yang dikenal dengan nama bobot). Perubahan yang terjadi selama proses pembelajaran adalah perubahan bobot. Nilai bobot akan bertambah jika informasi yang diberikan oleh neuron yang bersangkutan tersampaikan, sebaliknya jika informasi

tidak tersampaikan oleh suatu neuron ke neuron lain, maka nilai bobot akan diubah secara dinamis hingga mencapai suatu nilai yang seimbang. Apabila nilai ini telah tercapai mengindikasikan bahwa tiap-tiap input telah berhubungan dengan output yang diharapkan. Ada 2 tipe pembelajaran :

- a. Pembelajaran terawasi (*Supervised Learning*).
- b. Pembelajaran Tidak Terawasi (*Unsupervised Learning*).

Pembelajaran yang digunakan adalah pembelajaran terawasi. Metode ini disebut terawasi jika output yang diharapkan telah diketahui sebelumnya. Pada proses pembelajaran ini, satu pola input akan diberikan satu neuron pada lapisan input. Pola ini akan dirambatkan disepanjang lapisan output. Lapisan output ini akan membangkitkan pola output yang nantinya akan dicocokkan dengan pola output targetnya. Apabila terjadi perbedaan antara pola output hasil pembelajaran dengan pola output target, maka disini akan muncul *error*. Apabila nilai *error* ini masih cukup besar, mengindikasikan bahwa masih perlu dilakukan lebih banyak pembelajaran lagi. Ada beberapa metode dalam proses pembelajaran terawasi, diantaranya *Delta Rule*, *Backpropagation* atau *Generalized Delta Rule* dan *Counter Propagation*. Yang digunakan dalam pembelajaran penelitian ini adalah *Backpropagation*.

3.6. *Backpropagation*.^[5]

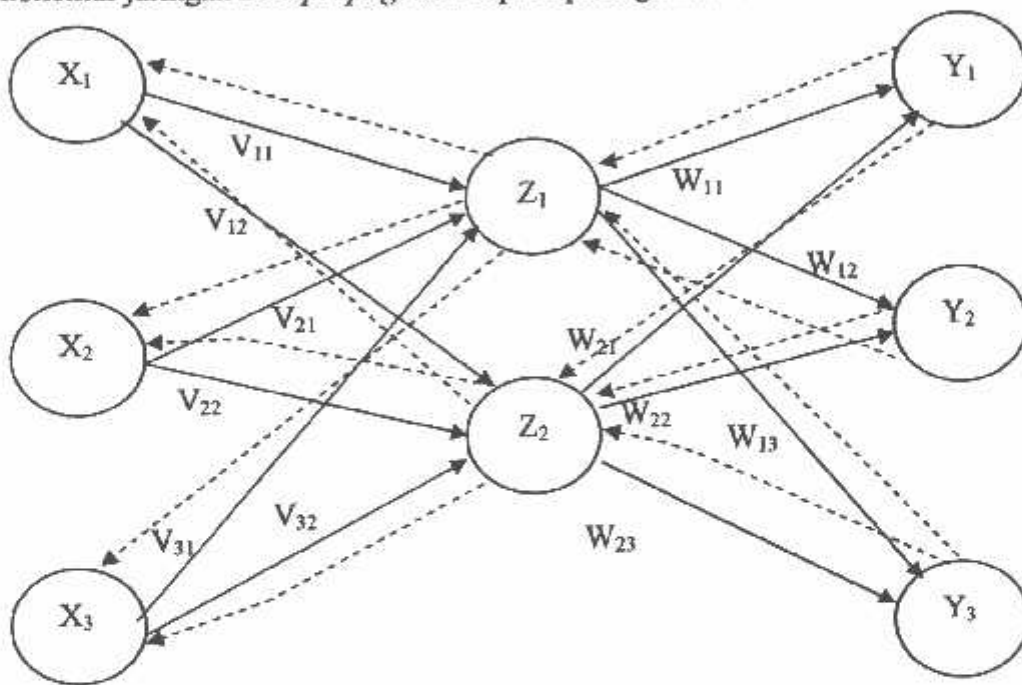
Algoritma *Backpropagation* menggunakan *error* output untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap perambatan maju (*forward propagation*) harus dikerjakan dahulu. Pada

saat perambatan maju, neuron-neuron diaktifkan dengan menggunakan fungsi aktivasi sigmoid, yaitu:

$$f(x) = \frac{1}{1 + e^{-x}} \dots\dots\dots(3.14)$$

$f(x)$ = fungsi aktivasi semua input unit.

Arsitektur jaringan *Backpropagation* seperti pada gambar berikut:



Gambar 3-4 Arsitektur Jaringan *Backpropagation*

Sumber : *Artifisial intelligent*, Sri kusuma Dewi , hal 236

a. Pemrosesan Data Input ^[4]

Pelatihan *artificial neural network* akan lebih efektif dan efisien apabila data-data input berada dalam satu *range* tertentu. Menyajikan data mentah kedalam *artificial neural network* dapat membuat *neuron* mengalami saturasi dan gagal

melakukan pembelajaran (*learning*). Data-data P, Q, dan V akan dinormalisasi agar berada pada *range* yang dipilih yaitu 0-1 dengan persamaan berikut :

$$X_i(\text{norm}) = \frac{X_i - X_{\min}}{X_{\max} - X_{\min}} \dots\dots\dots (3.15)$$

b. Pelatihan (training) ^[4]

Pelatihan metode *backpropagation* terdiri atas 3 langkah.

- Data yang dimasukkan ke *input* jaringan (*feed forward*).
- Perhitungan dan *propagasi* balik *error* dari *error* yang bersangkutan
- Pembaharuan bobot dan bias.

Algoritma untuk melatih ANN *Backpropagation* terdiri dari langkah-langkah sebagai berikut :

- Inisialisasi bobot (ambil bobot awal dengan nilai random yang cukup kecil)
- Kerjakan langkah-langkah berikut selama kondisi berhenti bernilai *FALSE*
 1. Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan :

Feedforward

- a. Tiap-tiap input (X_i , $i= 1,2,3,\dots,n$) menerima sinyal x_1 dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (*Hidden layer*).

X_i = unit input ke- i untuk unit input, sinyal yang masuk dan keluar pada suatu unit yang dilambangkan dengan variabel yang sama yaitu x_i .

- b. Tiap-tiap unit tersembunyi (Z_j , $J = 1, 2, 3, \dots, p$) menjumlahkan sinyal-sinyal input berbobot :

$$Z_in_j = Vo_j + \sum_{i=1}^n x_i v_{ij} \dots\dots\dots(3.16)$$

Z = hidden unit (unit tersembunyi)

Vo_j = Bias unit hidden unit ke-j.

V_{ij} = bobot antara unit input ke-I dan hidden unit ke-j

V = bobot antara X dan Z.

W = bobot antara Z dan Y.

gunakan fungsi aktivasi untuk menghitung sinyal outputnya :

$$Z_j = f(z_in_j) \dots\dots\dots(3.17)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit output).

- c. Tiap-tiap unit output (Y_k , $k = 1, 2, 3, \dots, m$) menjumlahkan sinyal-sinyal input berbobot :

$$y_in_j = Wo_k + \sum_{i=1}^p Z_i W_{jk} \dots\dots\dots(3.18)$$

Wo_k = Bias unit output ke-k.

W_{jk} = bobot antar hidden unit ke-j dan unit output ke-k.

gunakan fungsi aktivasi untuk menghitung sinyal outputnya :

$$y_k = f(y_in_k) \dots\dots\dots(3.19)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit output).

y_k = Unit input ke-k. Sinyal input ke Y_k dilambangkan y_{in_k} .

Sinyal output (aktivasi) untuk Y_k dilambangkan dengan y_k .

y_k = Sinyal output untuk Y_k .

Backpropagation

- d. Tiap-tiap unit output (Y_k , $k = 1,2,3,\dots,m$) menerima target pola yang berhubungan dengan pola input pembelajaran, hitung informasi *error*nya : $\delta_k = (t_k - y_k)f'(y_{in_k})\dots\dots\dots(3.20)$

t = Data training untuk output (target / desired output).

t_k = telah diskalakan menurut fungsi aktivasi yang dipakai.

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai W_{jk}):

$$\Delta W_{jk} = \alpha \delta_k Z_j \dots\dots\dots(3.21)$$

α = *learning rate* yaitu parameter yang mengontrol perubahan bobot selama pelatihan. Jika *learning* besar, jaringan semakin cepat belajar, tetapi hasilnya kurang akurat. *Learning rate* biasanya dipilih antara 0 dan 1.

δ_k = faktor koreksi *error* untuk bobot W_{jk} .

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai W_{0k}):

$$\Delta W_{0k} = \alpha \delta_k \dots\dots\dots(3.22)$$

kiriman δ_k ini ke unit-unit yang ada dilapisan dibawahnya.

- c. Tiap-tiap unit tersembunyi (Z_j , $j = 1,2,3,\dots,p$) menjumlahkan delta inputnya (dari unit-unit yang berada pada lapisan diatasnya):

$$\delta_{in_j} = \sum_{k=1}^m \delta_k W_{jk} \dots\dots\dots(3.23)$$

kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error* :

$$\delta_j = \delta_{in_j} f'(Z_{in_j}) \dots\dots\dots(3.24)$$

δ_j = faktor koreksi *error* untuk bobot v_{ij} .

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai V_{ij}):

$$\Delta V_{jk} = \alpha \delta_j x_i \dots\dots\dots(3.25)$$

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai V_{oj}) :

$$\Delta V_{oj} = \alpha \delta_j \dots\dots\dots(3.26)$$

- f. Tiap-tiap unit output ($Y_k, k = 1,2,3,\dots,m$) memperbaiki bias dan bobotnya ($j = 0,1,2,3,\dots,p$)

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk} \dots\dots\dots(3.27)$$

Tiap-tiap unit tersembunyi ($Z_j, j = 1,2,3,\dots,p$) memperbaiki bias dan bobotnya ($i = 0,1,2,\dots,n$):

$$V_{ij}(\text{baru}) = V_{ij}(\text{lama}) + \Delta V_{ij} \dots\dots\dots(3.28)$$

2. Tes kondisi berhenti

3.6. Algoritma Program Rekonfigurasi Jaringan Menggunakan Metode *Artificial Neural Network*

1. Membaca data historis berupa data saluran dan data pembebanan.
 2. Setelah membentuk data historis dilanjutkan dengan membentuk input dan target dengan loadflow.
 3. Setelah membentuk input dan target kemudian menjalankan *ANN*.
 4. Bobot yang dihasilkan setelah menjalankan *ANN* kemudian disimpan.
 5. Setelah bobot disimpan, apakah $0.95 < V_{bus} < 1.05$ pu, jika ya maka proses berhenti, jika tidak maka melakukan rekonfigurasi.
 6. Selanjutnya membaca input dan bobot *ANN*.
 7. Setelah membaca bobot *ANN* dilanjutkan dengan menjalankan *ANN* sehingga didapatkan hasil.
 8. Setelah didapatkan hasil kemudian melakukan loadflow akhir selanjutnya kembali melakukan proses pada langkah 5.
-

BAB IV

HASIL DAN ANALISIS HASIL

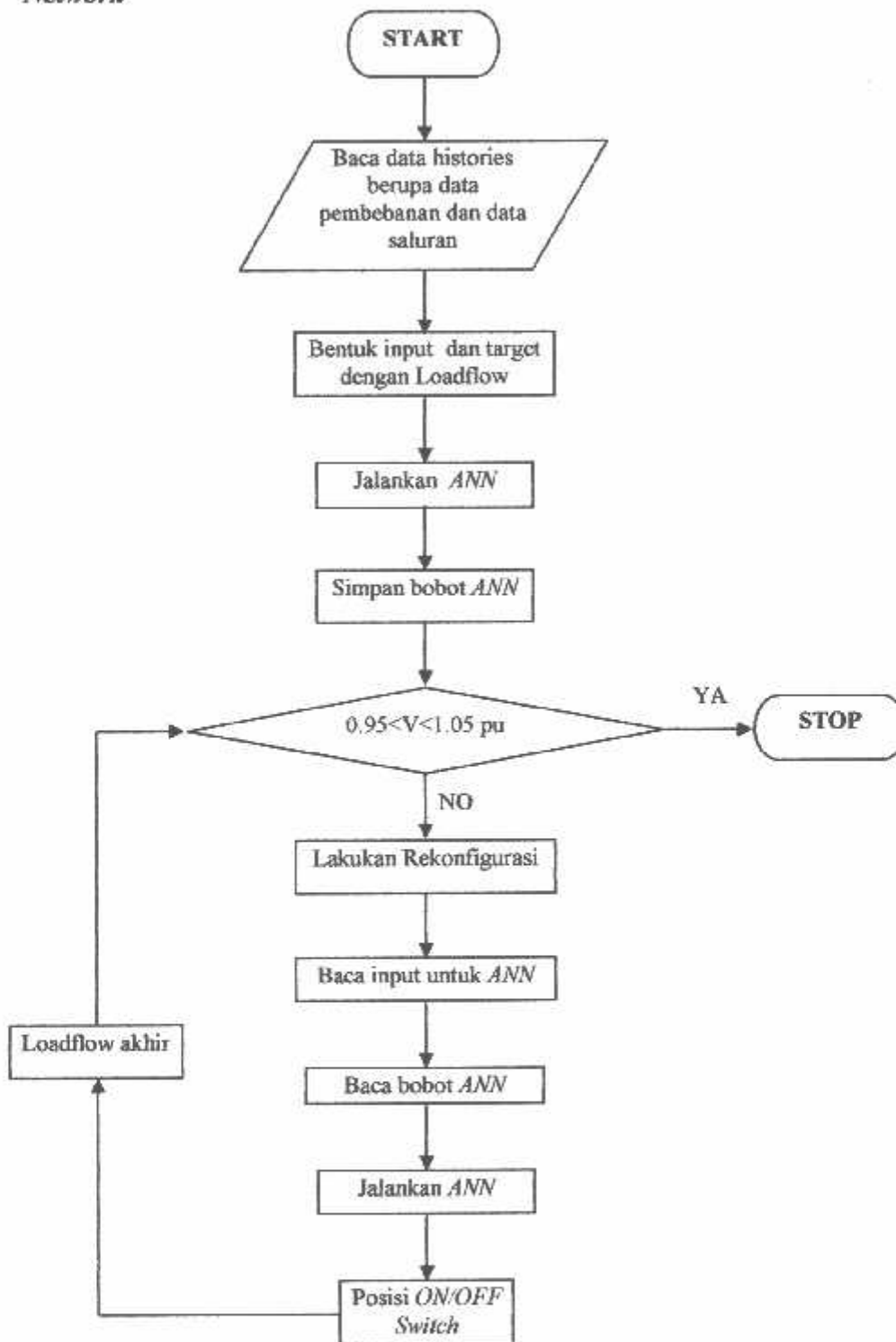
4.1. Program Komputer Untuk Rekonfigurasi Jaringan

Analisa aliran daya pada sistem jaringan distribusi radial pada dasarnya digunakan untuk memecahkan permasalahan (mengetahui parameter-parameter yang belum diketahui berdasarkan parameter-parameter yang telah diketahui) pada masing-masing bus dan cabang dalam sistem jaringan distribusi.

Dalam pemecahan masalah rekonfigurasi jaringan distribusi digunakan bantuan program komputer. Program komputer ini sangat berguna untuk mempercepat proses perhitungan aliran daya yang membutuhkan ketelitian tinggi dan sering melibatkan iterasi yang membutuhkan waktu yang lama bila dikerjakan secara manual.

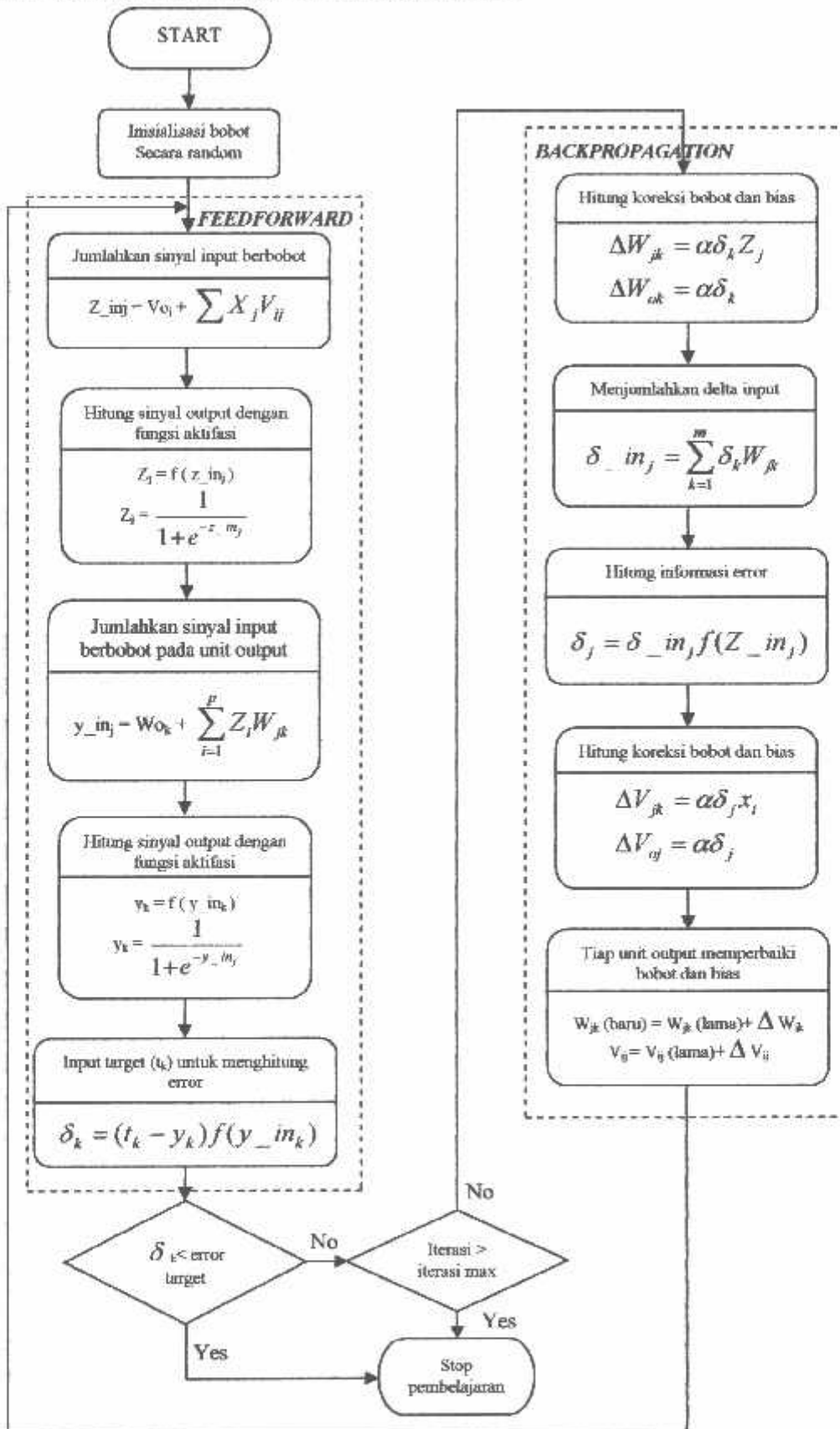
Program komputer ini menggunakan pemrograman Borland Delphi versi 7.0 merupakan bahasa pemrograman yang terstruktur dan mudah diaplikasikan.

4.2. Flowchart Rekonfigurasi Jaringan Menggunakan *Artificial Neural Network*



Gambar 4-1 Flowchart Rekonfigurasi Jaringan Menggunakan *Artificial Neural Network*

4.3. Flowchart Pembelajaran *Backpropagation*



Gambar 4-2 Flowchart Pembelajaran *Backpropagation*

4.4. Data Jaringan

Data yang digunakan dalam skripsi ini adalah penyulang-penyulang yang keluar dari Trafo II GI Blimbing Malang. Alasan dari pemilihan data tersebut karena output dari Trafo II GI Blimbing ini mensuplai daerah yang kepadatan bebannya bervariasi.

Data teknis dari Trafo II GI Blimbing adalah sebagai berikut :

| | |
|-----------------------|--|
| Tegangan dasar | : 20 kV |
| Daya dasar | : 30 MVA |
| Penghantar SUTM 20 kV | : AAAC 150 mm |
| Impedansi penghantar | : $0,2162 + j 0,3305 \Omega/\text{km}$ |
| $\cos \varphi$ | : 0,8 |
| $\sin \varphi$ | : 0,6 |

Data beban untuk masing-masing penyulang dapat dilihat pada lampiran.

Dari lampiran tersebut, contoh perhitungan daya P(MW) dan Q(MVAR) dari bus 2 penyulang Mawar dapat dihitung sebagai berikut :

$$P = 1784,55 \cdot \cos \varphi = 1,4276 \text{ MW}$$

$$Q = 1784,55 \cdot \sin \varphi = 1,0707 \text{ MVAR}$$

Dengan cara perhitungan yang sama, maka beban untuk masing-masing bus akan diperoleh hasil seperti pada Tabel 4-1.

Tabel 4-1

DATA JARINGAN SISTEM DISTRIBUSI DI GI BLIMBING

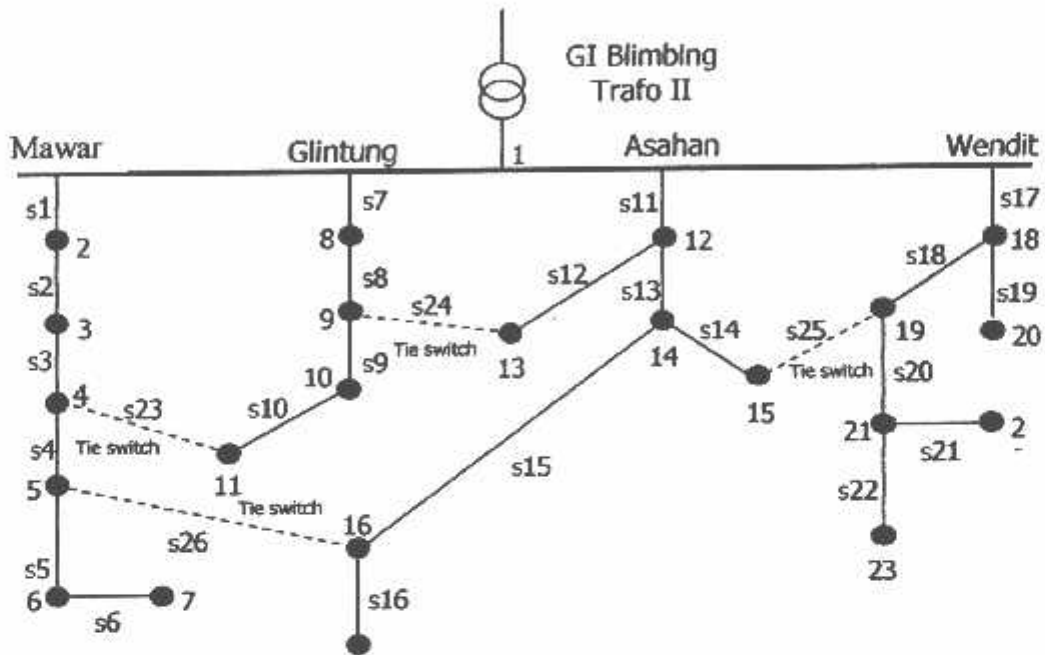
| BUS PANGKAL | BUS UJUNG | P BUS UJUNG (MW) | Q BUS UJUNG (MVAR) | JARAK (KM) | R (Ω) | X (Ω) |
|-------------|-----------|------------------|--------------------|------------|----------------|----------------|
| 1 | 2 | 1.4276 | 1.0707 | 5.4090 | 1.2142 | 1.8650 |
| 2 | 3 | 0.7655 | 0.5741 | 3.0140 | 0.8832 | 1.3501 |
| 3 | 4 | 1.4842 | 1.1132 | 4.9150 | 1.0896 | 0.6657 |
| 4 | 5 | 0.8594 | 0.6446 | 3.7820 | 0.8425 | 0.2880 |
| 5 | 6 | 0.1661 | 0.1245 | 0.9940 | 0.1464 | 0.2238 |
| 6 | 7 | 0.6190 | 0.6463 | 2.7380 | 0.3628 | 0.5546 |
| 1 | 8 | 0.3342 | 0.2506 | 1.4790 | 0.2986 | 0.4546 |
| 8 | 9 | 0.7353 | 0.5515 | 1.2240 | 0.2581 | 0.3946 |
| 9 | 10 | 0.2871 | 0.2153 | 1.7060 | 0.2197 | 0.3358 |
| 10 | 11 | 0.4398 | 0.3299 | 1.4920 | 0.3226 | 0.4931 |
| 1 | 12 | 1.4908 | 1.1181 | 2.0590 | 0.5349 | 0.8177 |
| 12 | 13 | 1.4538 | 1.0903 | 3.4130 | 0.7444 | 1.1379 |
| 12 | 14 | 0.1972 | 0.1479 | 1.6460 | 0.3559 | 0.5440 |
| 14 | 15 | 0.6540 | 0.4905 | 1.0390 | 0.2558 | 0.3910 |
| 14 | 16 | 1.4906 | 1.1179 | 2.5820 | 0.5935 | 0.9072 |
| 16 | 17 | 0.0450 | 0.0338 | 0.1310 | 0.0283 | 0.0433 |
| 1 | 18 | 0.1324 | 0.0993 | 1.8170 | 0.3853 | 0.5890 |
| 18 | 19 | 0.3921 | 0.2941 | 2.6270 | 0.6553 | 1.0017 |
| 18 | 20 | 2.8164 | 2.1123 | 3.0090 | 1.6516 | 2.5247 |
| 19 | 21 | 0.1776 | 0.1332 | 0.5290 | 0.3349 | 0.5119 |
| 21 | 22 | 0.6146 | 0.4609 | 1.4760 | 0.3191 | 0.4878 |
| 21 | 23 | 0.9675 | 0.7256 | 4.6820 | 1.0318 | 1.5497 |
| 4 | 11 | - | - | 0.2070 | 0.0448 | 0.0684 |
| 9 | 13 | - | - | 0.1730 | 0.0374 | 0.0572 |
| 15 | 19 | - | - | 1.1140 | 0.2408 | 0.3682 |
| 5 | 16 | - | - | 0.2620 | 0.0566 | 0.0866 |

Sumber : PT. PLN (PERSERO) Distribusi Jatim Cabang Malang

Output Trafo II GI Blimbing terdiri dari :

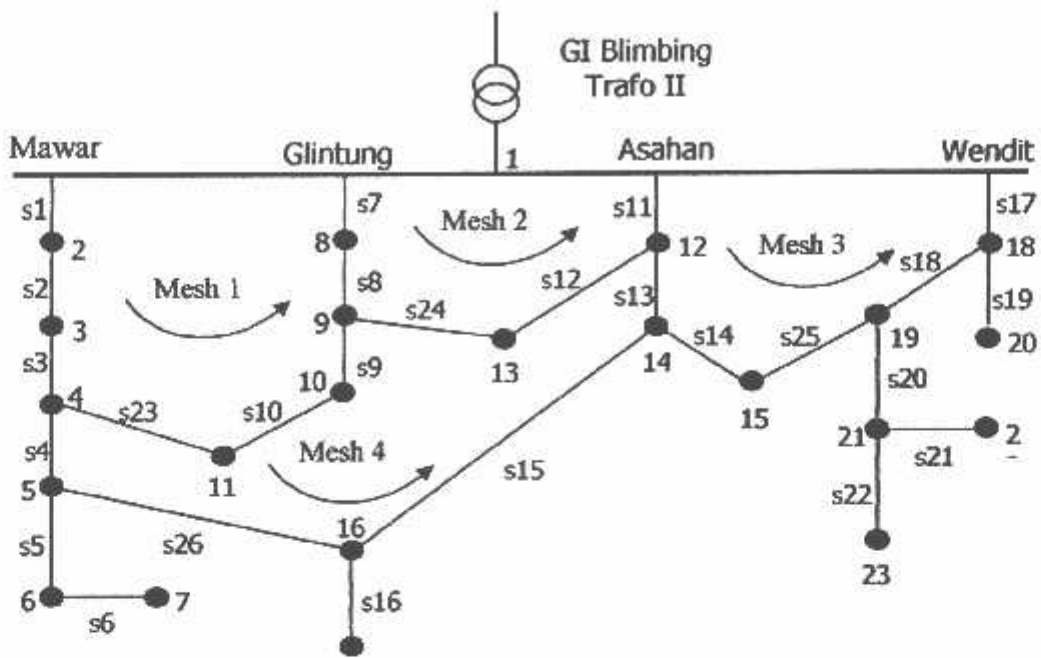
1. Penyulang Mawar (M).
2. Penyulang Glintung (G).
3. Penyulang Asahan (A).
4. Penyulang Wendit (W).

Konfigurasi awal (*Initial condition*) penyulang pada Trafo II GI Blimbing Malang dapat dilihat pada Gambar 4-1, dimana pada gambar dapat dilihat posisi *switch* (*NO switch*) sebelum rekonfigurasi yaitu *switch* 4-11, *switch* 5-16, *switch* 9-13 dan *switch* 15-19.



Gambar 4-1 Konfigurasi Jaringan Sebelum Rekonfigurasi

Dalam menganalisa rekonfigurasi jaringan semua *tie switch* (*normally open*) ditutup sehingga jaringan radial seperti pada Gambar 4-1 menjadi jaringan mesh seperti pada Gambar 4-2. Dari penutupan *tie switch* terdapat empat mesh seperti gambar berikut :



Gambar 4-2 Konfigurasi Jaringan Mesh

4.3. Analisa Perhitungan Sebelum dan Sesudah Rekonfigurasi

Dari data pada Tabel 4-1 dan 4-2 (data jaringan distribusi) dimasukkan kedalam software rekonfigurasi (program komputer), maka dapat diketahui hasil perhitungan sebelum dan sesudah rekonfigurasi secara cepat dan efisien.

4.3.1. Analisa Perhitungan Sebelum Rekonfigurasi

Tabel 4-2

HASIL PERHITUNGAN ALIRAN DAYA SEBELUM REKONFIGURASI

| BUS | Tegangan | | Daya Pembangkitan | | Beban | |
|-----|----------|-------------|-------------------|-------|--------|--------|
| | Abs (pu) | Sudut (rad) | MW | MVAR | MW | MVAR |
| 1 | 1.00000 | 0.0000 | 12.379 | 9.565 | 0.000 | 0.000 |
| 2 | 0.97390 | -0.50073 | - | - | 1.4276 | 1.0707 |
| 3 | 0.95984 | -0.76904 | - | - | 0.7655 | 0.5741 |
| 4 | 0.94573 | -1.03859 | - | - | 1.4842 | 1.1132 |
| 5 | 0.93977 | -1.14231 | - | - | 0.8594 | 0.6446 |
| 6 | 0.93923 | -1.14941 | - | - | 0.1661 | 0.1245 |
| 7 | 0.93815 | -1.16170 | - | - | 0.6190 | 0.6463 |
| 8 | 0.99805 | -0.04052 | - | - | 0.3342 | 0.2506 |
| 9 | 0.99667 | -0.06914 | - | - | 0.7353 | 0.5515 |
| 10 | 0.99609 | -0.08126 | - | - | 0.2871 | 0.2153 |
| 11 | 0.99557 | -0.09205 | - | - | 0.4398 | 0.3299 |
| 12 | 0.98965 | -0.21418 | - | - | 1.4908 | 1.1181 |
| 13 | 0.98572 | -0.29663 | - | - | 1.4538 | 0.0903 |
| 14 | 0.98656 | -0.27883 | - | - | 0.1972 | 0.1479 |
| 15 | 0.98595 | -0.29160 | - | - | 0.6540 | 0.4905 |
| 16 | 0.98324 | -0.34864 | - | - | 1.4906 | 1.1179 |
| 17 | 0.98324 | -0.34874 | - | - | 0.0450 | 0.0338 |
| 18 | 0.99227 | -0.15805 | - | - | 0.1324 | 0.0993 |
| 19 | 0.98676 | -0.27295 | - | - | 0.3921 | 0.2941 |
| 20 | 0.97391 | -0.54212 | - | - | 2.8164 | 2.1123 |
| 21 | 0.98445 | -0.32135 | - | - | 0.1776 | 0.1332 |
| 22 | 0.98369 | -0.33752 | - | - | 0.6146 | 0.4609 |
| 23 | 0.98061 | -0.40241 | - | - | 0.9675 | 0.7256 |

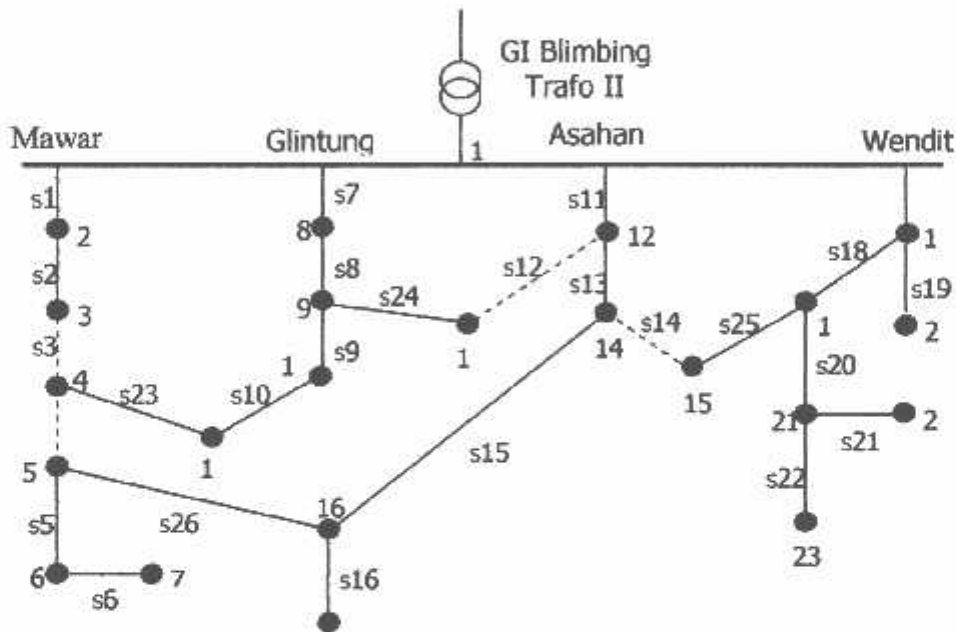
Dari hasil perhitungan aliran daya sebelum rekonfigurasi jaringan pada Tabel 4-2

diatas dapat diketahui :

| | | |
|--------------------------------|-------------|-------------|
| Jumlah Total Pembangkitan | : 18.033 MW | 13,900 MVAR |
| Jumlah Total Pembebanan | : 17,550 MW | 13,163 MVAR |
| Jumlah Total Rugi-rugi Saluran | : 0,228 MW | 0,349 MVAR |

4.3.2. Analisa Perhitungan Sesudah Rekonfigurasi

Setelah dilakukan rekonfigurasi akan didapatkan kondisi *NO switch* yang terbuka yaitu *switch 3-4*, *switch 4-5*, *switch 12-13*, *switch 14-15*. Pembukaan *switch 3-4* (*tie switch* pada *s3*) harus diikuti oleh penutupan *switch 4-11* (*tie switch* pada *s23*) dan seterusnya, sehingga jaringan mesh yang telah ada akan berubah menjadi jaringan radial kembali, dapat dilihat pada Gambar 4-3



Gambar 4-3 Konfigurasi Jaringan Radial Setelah Rekonfigurasi

Tabel 4-3

HASIL PERHITUNGAN ALIRAN DAYA SETELAH REKONFIGURASI

| BUS | Tegangan | | Daya Pembangkitan | | Beban | |
|-----|----------|-------------|-------------------|-------|--------|--------|
| | Abs (pu) | Sudut (rad) | MW | MVAR | MW | MVAR |
| 1 | 1.00000 | 0.00000 | 12.302 | 9.448 | 0.000 | 0.000 |
| 2 | 0.98984 | -0.21714 | - | - | 1.4276 | 1.0707 |
| 3 | 0.98728 | -0.27082 | - | - | 0.7655 | 0.5741 |
| 4 | 0.98618 | -0.28747 | - | - | 1.4842 | 1.1132 |
| 5 | 0.97768 | -0.41968 | - | - | 0.8594 | 0.6446 |
| 6 | 0.97717 | -0.42624 | - | - | 0.1661 | 0.1245 |
| 7 | 0.97613 | -0.43761 | - | - | 0.6190 | 0.6463 |
| 8 | 0.99478 | -0.10764 | - | - | 0.3342 | 0.2506 |
| 9 | 0.99059 | -0.19501 | - | - | 0.7353 | 0.5515 |
| 10 | 0.98877 | -0.23313 | - | - | 0.2871 | 0.2153 |
| 11 | 0.98643 | -0.28218 | - | - | 0.4398 | 0.3299 |
| 12 | 0.99009 | -0.18861 | - | - | 1.4908 | 1.1181 |
| 13 | 0.99039 | -0.19913 | - | - | 1.4538 | 0.0903 |
| 14 | 0.98542 | -0.27551 | - | - | 0.1972 | 0.1479 |
| 15 | 0.98368 | -0.33712 | - | - | 0.6540 | 0.4905 |
| 16 | 0.97806 | -0.41319 | - | - | 1.4906 | 1.1179 |
| 17 | 0.98706 | -0.41329 | - | - | 0.0450 | 0.0338 |
| 18 | 0.99134 | -0.17708 | - | - | 0.1324 | 0.0993 |
| 19 | 0.98425 | -0.32503 | - | - | 0.3921 | 0.2941 |
| 20 | 0.97296 | -0.56189 | - | - | 2.8164 | 2.1123 |
| 21 | 0.98194 | -0.37368 | - | - | 0.1776 | 0.1332 |
| 22 | 0.98118 | -0.38992 | - | - | 0.6146 | 0.4609 |
| 23 | 0.97809 | -0.45515 | - | - | 0.9675 | 0.7256 |

Dari hasil perhitungan aliran daya setelah rekonfigurasi jaringan pada Tabel 4-3 diatas dapat diketahui :

| | | |
|--------------------------------|-------------|-------------|
| Jumlah Total Pembangkitan | : 18.033 MW | 13,900 MVAR |
| Jumlah Total Pembebanan | : 17,550 MW | 13,163 MVAR |
| Jumlah Total Rugi-rugi Saluran | : 0,152 MW | 0,232 MVAR |

Hasil perhitungan aliran daya sebelum dan sesudah rekonfigurasi pada Tabel 4-3 maka didapatkan perbandingan hasil perhitungan untuk mengurangi rugi-rugi daya pada saluran, untuk lebih jelasnya lihat perbandingannya pada Tabel 4-4

Tabel 4-4
PERBANDINGAN RUGI-RUGI SALURAN

| Rugi-rugi sebelum rekonfigurasi | | Rugi-rugi sesudah rekonfigurasi | | Reduksi rugi-rugi | |
|---------------------------------|-------|---------------------------------|-------|-------------------|-------|
| MW | MVAR | MW | MVAR | MW | MVAR |
| 0.228 | 0.349 | 0.152 | 0.232 | 0.076 | 0.117 |

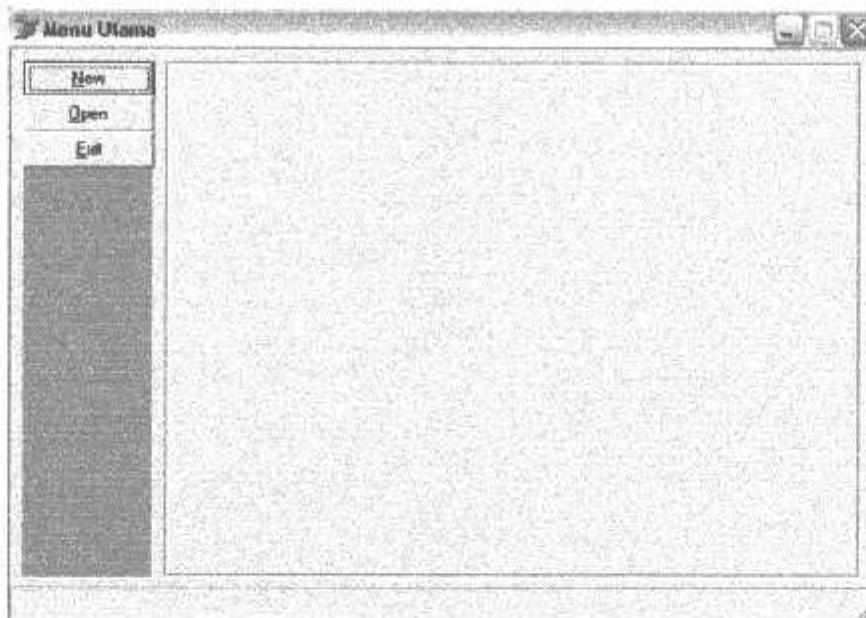
Jika dilihat dalam prosentase setelah dilakukan rekonfigurasi jaringan maka reduksi rugi-rugi saluran sebesar 33,33%.

4.4. Tampilan Program

Untuk penyelesaian dalam memperkecil rugi-rugi daya pada jaringan distribusi digunakan bantuan komputer yaitu program Borland Delphi versi 7.0, dengan menggunakan bantuan program komputer ini maka hasil dari perhitungan aliran daya dapat diketahui lebih cepat dan akurat, lain halnya bila dilakukan secara manual yang membutuhkan waktu lebih lama dan keakuratannya kurang terjamin karena membutuhkan ketelitian yang lebih.

Hasil program dapat ditampilkan sebagai berikut :

1. Tampilan utama adalah menampilkan menu utama dari program mereduksi rugi-tugi daya.

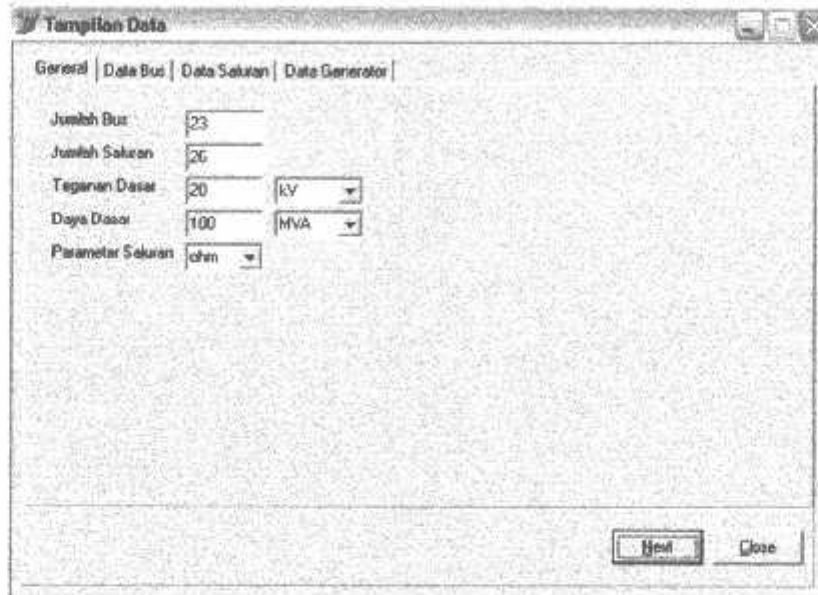


Gambar 4-4 Tampilan Menu Utama Program

- Setelah itu klik open, kemudian pilih data GI Blimbing.

| Bus | abV (pu) | aδV (deg) | Pg (MW) | Qg (MVAR) | PL (MW) | QL (MVAR) | Saps (pu) | Type Bus |
|-----|----------|-----------|---------|-----------|---------|-----------|-----------|----------|
| 1 | 1.00000 | 0.00000 | 12.379 | 9.565 | 0.000 | 0.000 | 0.000 | 1 |
| 2 | 0.97390 | -0.50073 | 0.000 | 0.000 | 1.025 | 0.745 | 0.000 | 3 |
| 3 | 0.95984 | -0.76904 | 0.000 | 0.000 | 0.533 | 0.400 | 0.000 | 3 |
| 4 | 0.94573 | -1.03859 | 0.000 | 0.000 | 1.033 | 0.775 | 0.000 | 3 |
| 5 | 0.93977 | -1.14231 | 0.000 | 0.000 | 0.598 | 0.449 | 0.000 | 3 |
| 6 | 0.93823 | -1.14941 | 0.000 | 0.000 | 0.116 | 0.067 | 0.000 | 3 |
| 7 | 0.93815 | -1.16170 | 0.000 | 0.000 | 0.431 | 0.450 | 0.000 | 3 |
| 8 | 0.99805 | -0.04052 | 0.000 | 0.000 | 0.226 | 0.170 | 0.000 | 3 |
| 9 | 0.99667 | -0.06914 | 0.000 | 0.000 | 0.497 | 0.373 | 0.000 | 3 |
| 10 | 0.99609 | -0.08126 | 0.000 | 0.000 | 0.194 | 0.146 | 0.000 | 3 |
| 11 | 0.99557 | -0.09205 | 0.000 | 0.000 | 0.297 | 0.223 | 0.000 | 3 |
| 12 | 0.98965 | 0.21418 | 0.000 | 0.000 | 0.994 | 0.745 | 0.000 | 3 |
| 13 | 0.98672 | -0.29663 | 0.000 | 0.000 | 0.969 | 0.727 | 0.000 | 3 |
| 14 | 0.98656 | -0.27883 | 0.000 | 0.000 | 0.131 | 0.099 | 0.000 | 3 |
| 15 | 0.98695 | -0.29160 | 0.000 | 0.000 | 0.436 | 0.327 | 0.000 | 3 |

At the bottom of the window, there are three buttons: 'Create/Corr', 'Hitung', and 'Close'.



Gambar 4-5 Tampilan Data Sebelum Rekonfigurasi

3. Kemudian klik *next* dan klik hitung, akan terlihat hasil perhitungan sebelum rekonfigurasi, untuk melihat besar aliran daya sebelum rekonfigurasi klik aliran daya.

| Bus | abV (pu) | eselV (deg) | Pg (MW) | Qg (MVAR) | PL (MW) | QL (MVAR) | Supte (pu) | Type Bus |
|-----|----------|-------------|---------|-----------|---------|-----------|------------|----------|
| 1 | 1.00000 | 0.00000 | 12.329 | 9.565 | 0.000 | 0.000 | 0.000 | 1 |
| 2 | 0.97390 | -0.50073 | 0.000 | 0.000 | 1.025 | 0.745 | 0.000 | 3 |
| 3 | 0.95984 | -0.76904 | 0.000 | 0.000 | 0.533 | 0.400 | 0.000 | 3 |
| 4 | 0.94573 | -1.03859 | 0.000 | 0.000 | 1.033 | 0.775 | 0.000 | 3 |
| 5 | 0.93177 | -1.14231 | 0.000 | 0.000 | 0.598 | 0.449 | 0.000 | 3 |
| 6 | 0.91823 | -1.14941 | 0.000 | 0.000 | 0.116 | 0.087 | 0.000 | 3 |
| 7 | 0.93815 | -1.16170 | 0.000 | 0.000 | 0.431 | 0.450 | 0.000 | 3 |
| 8 | 0.99805 | -0.04052 | 0.000 | 0.000 | 0.226 | 0.170 | 0.000 | 3 |
| 9 | 0.99667 | -0.06914 | 0.000 | 0.000 | 0.497 | 0.373 | 0.000 | 3 |
| 10 | 0.99609 | -0.08126 | 0.000 | 0.000 | 0.194 | 0.145 | 0.000 | 3 |
| 11 | 0.99557 | -0.09205 | 0.000 | 0.000 | 0.297 | 0.223 | 0.000 | 3 |
| 12 | 0.98965 | -0.21418 | 0.000 | 0.000 | 0.994 | 0.745 | 0.000 | 3 |
| 13 | 0.98572 | -0.29663 | 0.000 | 0.000 | 0.969 | 0.727 | 0.000 | 3 |
| 14 | 0.98656 | -0.27883 | 0.000 | 0.000 | 0.131 | 0.089 | 0.000 | 3 |
| 15 | 0.98595 | -0.29160 | 0.000 | 0.000 | 0.436 | 0.327 | 0.000 | 3 |

| No | Dari | Ke | P (MW) | Q (MVAR) | Asumsi SAJ | Asumsi SAJ | Dari | Ke | P (MW) | Q |
|----|------|----|--------|----------|------------|------------|------|----|--------|---|
| 1 | 1 | 2 | 3.865 | 3.104 | 193.243 | 195.191 | 2 | 1 | -3.790 | |
| 2 | 2 | 3 | 2.769 | 2.251 | 141.168 | 116.791 | 3 | 2 | -2.740 | |
| 3 | 3 | 4 | 2.209 | 1.809 | 113.733 | 95.779 | 4 | 3 | -2.185 | |
| 4 | 4 | 5 | 1.152 | 0.987 | 59.943 | 53.824 | 5 | 4 | 1.147 | |
| 5 | 5 | 6 | 0.548 | 0.539 | 28.577 | 29.261 | 6 | 5 | 0.548 | |
| 6 | 6 | 7 | 0.432 | 0.452 | 22.503 | 24.512 | 7 | 6 | -0.432 | |
| 7 | 7 | 8 | 1.218 | 0.916 | 60.887 | 45.788 | 8 | 7 | -1.216 | |
| 8 | 8 | 9 | 0.390 | 0.744 | 49.571 | 37.269 | 9 | 8 | -0.989 | |
| 9 | 9 | 10 | 0.492 | 0.369 | 24.651 | 18.581 | 10 | 9 | -0.492 | |
| 10 | 10 | 11 | 0.298 | 0.223 | 14.918 | 11.229 | 11 | 10 | -0.297 | |
| 11 | 11 | 12 | 3.589 | 2.720 | 179.446 | 135.996 | 12 | 11 | -3.562 | |
| 12 | 12 | 13 | 0.972 | 0.731 | 48.955 | 37.118 | 13 | 12 | -0.969 | |
| 13 | 13 | 14 | 1.587 | 1.203 | 80.455 | 61.073 | 14 | 13 | -1.583 | |
| 14 | 14 | 15 | 0.436 | 0.327 | 22.018 | 16.695 | 15 | 14 | -0.436 | |

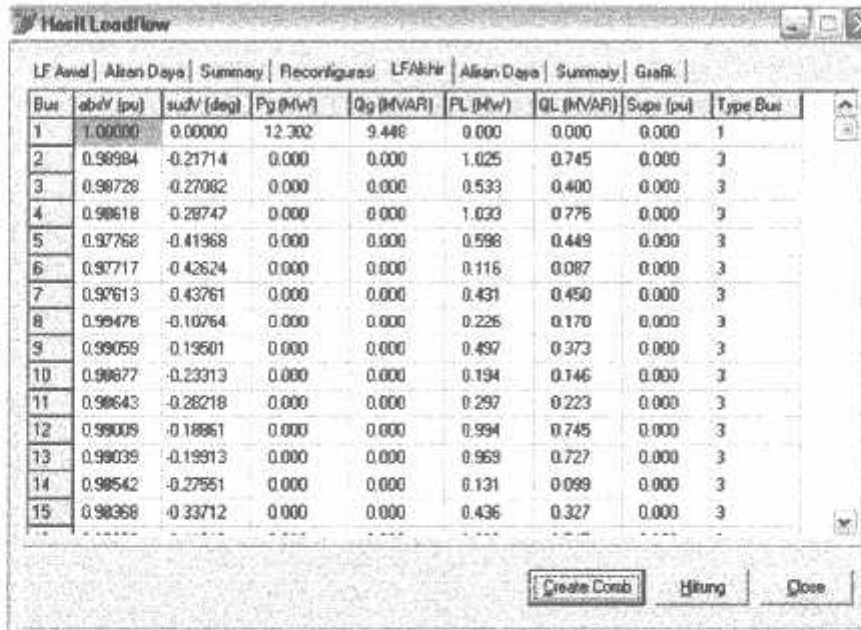
Gambar 4-6 Tampilan Hasil Perhitungan Sebelum Rekonfigurasi

- Selanjutnya klik rekonfigurasi dan klik use default, kemudian klik create comb sehingga tampil posisi switch rekonfigurasi Artificial Neural Network.

| No | Asal | Tujuan |
|----|------|--------|
| 1 | 2 | 4 |
| 2 | 4 | 5 |
| 3 | 12 | 13 |
| 4 | 14 | 15 |

Gambar 4-7 Tampilan Posisi Switch

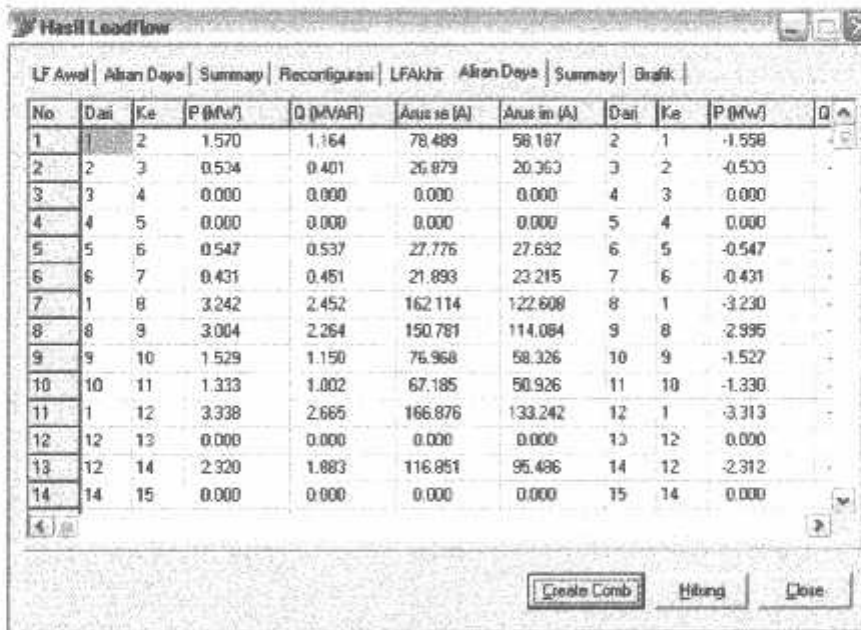
5. Kemudian klik loadflow dan klik aliran daya untuk melihat hasil perhitungan setelah rekonfigurasi.



The screenshot shows a window titled "Hasil Loadflow" with a menu bar containing "LF Awal", "Aliran Daya", "Summary", "Rekonfigurasi", "LFAKhr", "Aliran Daya", "Summary", and "Grafik". The main area contains a table with the following data:

| Bus | absV (pu) | sudV (deg) | Pg (MW) | Qg (MVAR) | PL (MW) | QL (MVAR) | Supr (pu) | Type Bus |
|-----|-----------|------------|---------|-----------|---------|-----------|-----------|----------|
| 1 | 1.00000 | 0.00000 | 12.302 | 9.448 | 0.000 | 0.000 | 0.000 | 1 |
| 2 | 0.98994 | -0.21714 | 0.000 | 0.000 | 1.025 | 0.745 | 0.000 | 3 |
| 3 | 0.98728 | -0.27082 | 0.000 | 0.000 | 0.533 | 0.400 | 0.000 | 3 |
| 4 | 0.98618 | -0.28747 | 0.000 | 0.000 | 1.033 | 0.776 | 0.000 | 3 |
| 5 | 0.97768 | -0.41968 | 0.000 | 0.000 | 0.598 | 0.449 | 0.000 | 3 |
| 6 | 0.97717 | -0.42624 | 0.000 | 0.000 | 0.116 | 0.087 | 0.000 | 3 |
| 7 | 0.97613 | -0.43761 | 0.000 | 0.000 | 0.431 | 0.450 | 0.000 | 3 |
| 8 | 0.99478 | -0.10764 | 0.000 | 0.000 | 0.226 | 0.170 | 0.000 | 3 |
| 9 | 0.99059 | -0.13601 | 0.000 | 0.000 | 0.497 | 0.373 | 0.000 | 3 |
| 10 | 0.98677 | -0.23313 | 0.000 | 0.000 | 0.194 | 0.146 | 0.000 | 3 |
| 11 | 0.98643 | -0.28218 | 0.000 | 0.000 | 0.297 | 0.223 | 0.000 | 3 |
| 12 | 0.99009 | -0.18861 | 0.000 | 0.000 | 0.994 | 0.745 | 0.000 | 3 |
| 13 | 0.99039 | -0.19913 | 0.000 | 0.000 | 0.969 | 0.727 | 0.000 | 3 |
| 14 | 0.98542 | -0.27551 | 0.000 | 0.000 | 0.131 | 0.099 | 0.000 | 3 |
| 15 | 0.98368 | -0.33712 | 0.000 | 0.000 | 0.436 | 0.327 | 0.000 | 3 |

At the bottom of the window are buttons for "Create Comb", "Hitung", and "Close".



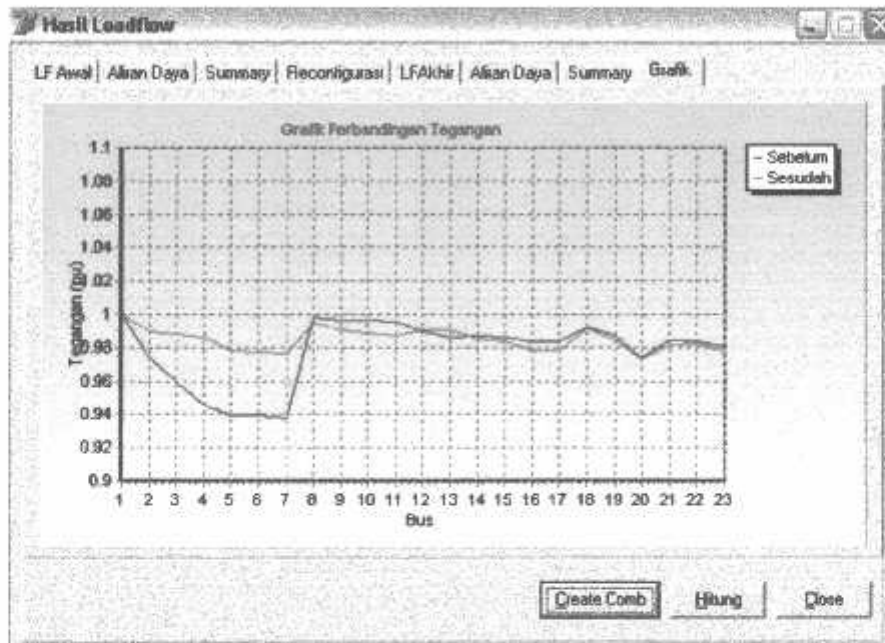
The screenshot shows a window titled "Hasil Loadflow" with a menu bar containing "LF Awal", "Aliran Daya", "Summary", "Rekonfigurasi", "LFAKhr", "Aliran Daya", "Summary", and "Grafik". The main area contains a table with the following data:

| No | Dari | Ke | P (MW) | Q (MVAR) | Anus se (A) | Anus in (A) | Dari | Ke | P (MW) | Q |
|----|------|----|--------|----------|-------------|-------------|------|----|--------|---|
| 1 | 1 | 2 | 1.570 | 1.164 | 78.489 | 58.187 | 2 | 1 | -1.558 | |
| 2 | 2 | 3 | 0.524 | 0.401 | 26.879 | 20.363 | 3 | 2 | -0.533 | |
| 3 | 3 | 4 | 0.000 | 0.000 | 0.000 | 0.000 | 4 | 3 | 0.000 | |
| 4 | 4 | 5 | 0.000 | 0.000 | 0.000 | 0.000 | 5 | 4 | 0.000 | |
| 5 | 5 | 6 | 0.547 | 0.537 | 27.776 | 27.632 | 6 | 5 | -0.547 | |
| 6 | 6 | 7 | 0.431 | 0.461 | 21.893 | 23.215 | 7 | 6 | -0.431 | |
| 7 | 1 | 8 | 3.242 | 2.452 | 162.114 | 122.608 | 8 | 1 | -3.230 | |
| 8 | 8 | 9 | 3.004 | 2.264 | 150.781 | 114.084 | 9 | 8 | -2.995 | |
| 9 | 9 | 10 | 1.529 | 1.150 | 76.968 | 58.326 | 10 | 9 | -1.527 | |
| 10 | 10 | 11 | 1.333 | 1.002 | 67.185 | 50.926 | 11 | 10 | -1.330 | |
| 11 | 1 | 12 | 3.338 | 2.665 | 166.876 | 133.242 | 12 | 1 | -3.313 | |
| 12 | 12 | 13 | 0.000 | 0.000 | 0.000 | 0.000 | 13 | 12 | 0.000 | |
| 13 | 12 | 14 | 2.320 | 1.883 | 116.851 | 95.486 | 14 | 12 | -2.312 | |
| 14 | 14 | 15 | 0.000 | 0.000 | 0.000 | 0.000 | 15 | 14 | 0.000 | |

At the bottom of the window are buttons for "Create Comb", "Hitung", and "Close".

Gambar 4-8 Tampilan Hasil Perhitungan Sesudah Rekonfigurasi

Grafik 4-1
PERBANDINGAN TEGANGAN SEBELUM DAN SESUDAH
REKONFIGURASI

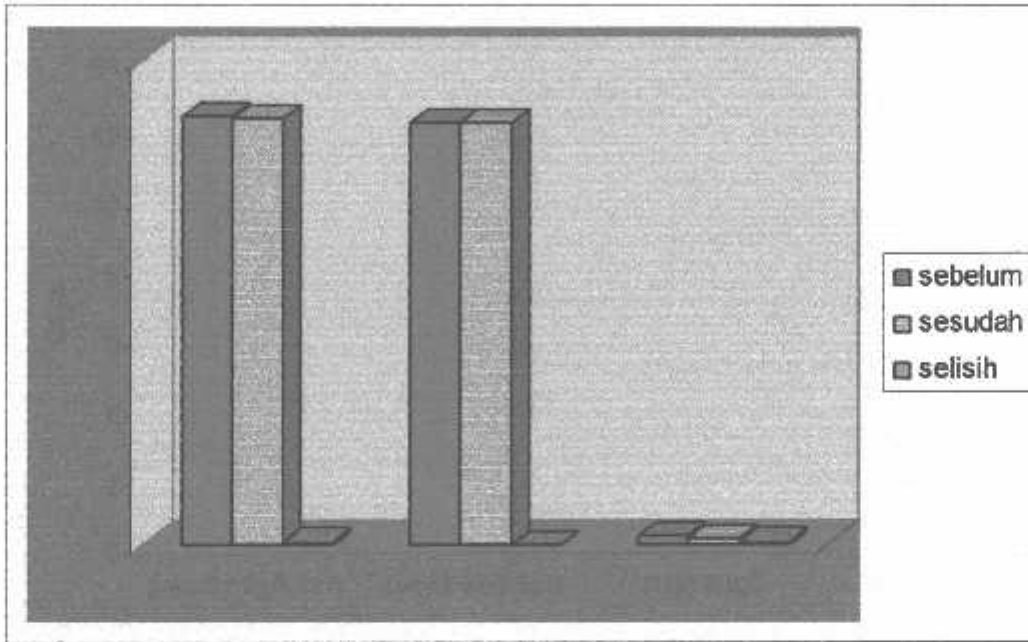


Keterangan :

- Bus 1 : Sebagai Slack Bus
- Bus 2-7 : Penyulang Mawar
- Bus 8-11 : Penyulang Glintung
- Bus 12-17 : Penyulang Asahan
- Bus 18-23 : Penyulang Wendit

Grafik 4-2

PERBANDINGAN PEMBANGKITAN, PEMBEBANAN DAN RUGI-RUGI



BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah dilakukan analisa mengenai metode rekonfigurasi jaringan, maka dapat diambil kesimpulan sebagai berikut :

1. Metode *Artificial Neural Network* mampu melakukan rekonfigurasi pada penyulang distribusi, yaitu dengan mengubah status *switch* 4-11, *switch* 5-16, *switch* 9-13 dan *switch* 15-19 yang semula *NO* sebelum rekonfigurasi menjadi *NC* setelah direkonfigurasi, sedangkan *switch* 3-4, *switch* 4-5, *switch* 12-13 dan *switch* 14-15 yang semula *NC* menjadi *NO*.
2. Sebelum dilakukan rekonfigurasi terjadi pelanggaran tegangan pada bus 4, 5, 6 dan 7 sebesar 0.94 pu dari ketentuan batas tegangan $0.95 < V_{bus} < 1.05$ pu, sedangkan setelah direkonfigurasi pada bus tegangan-tegangan tersebut meningkat menjadi 0.97 pu.
3. Disamping itu setelah rekonfigurasi rugi-rugi daya dapat direduksi sebesar 33.33% perinciannya sebagai berikut :

| | | |
|---|------------|------------|
| Rugi-rugi saluran sebelum direkonfigurasi | : 0.228 MW | 0.349 MVAR |
| Rugi-rugi saluran setelah rekonfigurasi | : 0.152 MW | 0.232 MVAR |
| Reduksi rugi-rugi saluran | : 0.076 MW | 0.117 MVAR |

5.2. Saran

Dari beberapa argumentasi di atas dapat bahwa pemanfaatan *Artificial Neural Network* dalam merekonfigurasi jaringan distribusi memungkinkan untuk diaplikasikan pada sistem distribusi di GI Blimbing Malang, meskipun masih diperlukan perbaikan-perbaikan untuk menyempurnakan metode ini, terutama dalam mengolah data karena dibutuhkan banyak data dan ketelitian tinggi.

DAFTAR PUSTAKA

- [1] Hoyong Kim, Yunseok Ko, Kyung-Hee Jung, "*Artificial Neural-Network Based Feeder Reconfiguration for Loss Reduction in Distribution Systems*", IEEE Trans. Power Delivery, vol. 8, No. 3, July 1993.
 - [2] Hasan Basri, "*Sistem Distribusi Tenaga Listrik*", Balai Penerbit & Humas ISTN, Jakarta, 1996.
 - [3] Stevenson, W.D., "*Analisis Sistem Tenaga Listrik*", Edisi Keempat, Erlangga, Jakarta, 1982.
 - [4] Djiteng Marsudi, Ir, "*Operasi Sistem Tenaga Listrik*", ISTN, Jakarta, 1990.
 - [5] Sri Kusuma Dewi, "*Artificial Intelligent*", Graha Ilmu, Yogyakarta, 2003.
 - [6] A S Pabla, "*Sistem Distribusi Daya Listrik*", Erlangga, Jakarta, 1986.
 - [7] S. Civanlar, J. J. Grainger, H. Yin, S. S. H. Lee, "*Distribution Feeder Reconfiguration for Loss Reduction*", IEEE Trans. Power Delivery, Vol. 3, No. 3, July 1988.
 - [8] A. Augugliaro, L. Dusonchet, M. G. Giuseppe Ippolito, E. Riva Sanseverino, "*Minimum Losses Reconfiguration of MV Distribution Network Through Lokal Control of Tie-Switch*", IEEE Trans. Power Delivery, vol. 18, No. 3, July 2003.
-




LEMBAR BIMBINGAN SKRIPSI

1. Nama : **DIANA RACHMANNINDIRA**
2. N.I.M. : **99.12.041**
3. Jurusan : **Teknik Elektro S-1**
4. Konsentrasi : **Teknik Energi Listrik**
5. Judul Skripsi : **REKONFIGURASI JARINGAN
UNTUK MEREDUKSI RUGI-RUGI
SISTEM DISTRIBUSI
MENGUNAKAN METODE
ARTIFICIAL NEURAL NETWORK DI
GILIMBING**
6. Tanggal Mengajukan Skripsi : **23 Mei 2005**
7. Tanggal Menyelesaikan Skripsi : **7 Oktober 2005**
8. Dosen Pembimbing : **Ir. I Made Wartana, MT**
9. Telah dievaluasi dengan nilai : **80,0 (Delapan Puluh Koma Nol) 80**

**Mengetahui,
Ketua Jurusan Teknik Elektro S-1**


**(Ir. F. Yudi Limpraptono, MT)
NIP. Y. 163 9500 274**

**Diperiksa dan disetujui,
Dosen Pembimbing**


**(Ir. I Made Wartana, MT)
NIP. 131 991 182**



PERSETUJUAN PERBAIKAN SKRIPSI

Dari hasil ujian Skripsi Jurusan Teknik Elektro jenjang Strata satu (S-1) yang diselenggarakan pada :

Hari : **Jumat**
Tanggal : **7 Oktober 2005**

Telah dilakukan perbaikan skripsi oleh :

1. Nama : **DIANA RACHMANNINDIRA**
2. N.L.M. : **99.12.041**
3. Jurusan : **Teknik Elektro S-1**
4. Konsentrasi : **Teknik Energi Listrik**
5. Judul Skripsi : **REKONFIGURASI JARINGAN
UNTUK MEREDUKSI RUGI-RUGI
SISTEM DISTRIBUSI
MENGUNAKAN METODE
ARTIFICIAL NEURAL NETWORK DI
GI BLIMBING**

Perbaikan meliputi :

| No. | Materi Perbaikan | Keterangan |
|-----|--|------------|
| 1. | Gambar 4-1 jika dicocokkan dengan data tidak sesuai | |
| 2. | Gambar 4-7 (hal.52) satuan dari rugi-rugi tidak sesuai | |
| 3. | Gambar 4-7 (hal. 52) tulisan gambar tidak ada | |

Disetujui/Diperiksa

(Ir. Taufik Hidayat, MT)
Penguji Pertama

(Irrine Eudi, ST, MT)
Penguji Kedua

Mengetahui/menyetujui
Dosen Pembimbing

(Ir. I Made Wartana, MT)
NIP. 131 991 182



**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

1. Nama : DIANA RACHMANNINDIRA
2. N.I.M. : 99.12.041
3. Jurusan : Teknik Elektro S-1
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : REKONFIGURASI JARINGAN
UNTUK MEREDUKSI RUGI-RUGI
SISTEM DISTRIBUSI
MENGUNAKAN METODE
ARTIFICIAL NEURAL NETWORK DI
GI BLIMBING

Dipertahankan Dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1)

6. Hari : 23 Mei 2005
7. Tanggal : 7 Oktober 2005
8. Dengan Nilai : 75,65 (Tujuh Puluh Lima Koma Enam
Puluh Lima) *8*



(Ir. Mochtar Asroni, MSME)
Ketua

Panitia Ujian Skripsi

(Ir. F. Yudi Limpraptono, MT)
Sekertaris

Anggota Penguji

(Ir. Taufik Hidayat, MT)
Penguji Pertama

(Irrine Budi, ST, MT)
Penguji Kedua

**Data Beban Tiap Penyulang
Pada Trafo II GI Blimbing Malang**

**Data Beban Tiap Seksi
Penyulang Mawar**

| Seksi | No. Trafo | Nominal (kVA) | Persen Beban (%) | Beban Trafo (kVA) |
|--------|-----------|---------------|------------------|-------------------|
| 1 | 186 | 150 | 74,3 | 111,45 |
| | 270 | 150 | 71,4 | 107,1 |
| | 369 | 150 | 74,3 | 111,45 |
| | 370 | 150 | 74,3 | 111,45 |
| | 371 | 150 | 74,3 | 111,45 |
| | 424 | 250 | 73,4 | 183,51 |
| | 425 | 100 | 30,8 | 30,82 |
| | 455 | 250 | 61,7 | 154,31 |
| | 542 | 150 | 79,6 | 89,4 |
| | 588 | 160 | 53,6 | 85,8 |
| | 605 | 100 | 48,9 | 48,99 |
| | 744 | 160 | 86,2 | 137,94 |
| | 764 | 160 | 67,6 | 108,23 |
| | 775 | 250 | 82,7 | 206,9 |
| | 779 | 250 | 74,3 | 185,75 |
| Jumlah | | | | 1748,55 |
| 2 | 193 | 200 | 43,5 | 87,1 |
| | 253 | 150 | 76,8 | 115,22 |
| | 253A | 200 | 73,8 | 147,6 |
| | 284 | 150 | 68,2 | 102,3 |
| | 290 | 150 | 69,8 | 104,65 |
| | 585 | 200 | 64,8 | 129,5 |
| | 615 | 160 | 74,3 | 118,88 |
| | 737 | 250 | 51,5 | 128,82 |
| | 920 | 160 | 14,2 | 22,77 |
| Jumlah | | | | 956,84 |
| 3 | 32 | 100 | 105 | 105 |
| | 110 | 150 | 74,3 | 111,45 |
| | 133 | 150 | 74,3 | 111,45 |
| | 230 | 160 | 74,3 | 118,88 |
| | 231 | 160 | 81,3 | 130 |
| | 305 | 160 | 80,6 | 129 |
| | 305A | 200 | 36 | 72 |
| | 335 | 75 | 74,3 | 55,73 |
| | 358 | 150 | 74,3 | 111,45 |
| | 368 | 150 | 114 | 171 |
| | 390 | 75 | 97,5 | 73,13 |
| | 409 | 150 | 74 | 111 |
| | 409A | 160 | 74,3 | 118,8 |
| | 543 | 150 | 86 | 129 |
| | 584 | 160 | 86,9 | 139 |
| 741 | 160 | 58,8 | 94 | |
| 762 | 100 | 74,3 | 74,3 | |
| Jumlah | | | | 1855,27 |
| 4 | 15 | 250 | 60,3 | 150,7 |
| | 21 | 200 | 90 | 180 |

| | | | | |
|---|--------|-----|------|--------|
| | 36 | 250 | 74 | 185 |
| | 38 | 100 | 76,5 | 76,5 |
| | 235 | 200 | 65,5 | 131 |
| | 244 | 250 | 50 | 125 |
| | 315 | 150 | 96,7 | 145 |
| | 706 | 50 | 13,5 | 6,75 |
| | 729 | 100 | 74,3 | 74,3 |
| | Jumlah | | | 1074,5 |
| 5 | 22 | 150 | 74,3 | 111,45 |
| | 240 | 160 | 60,1 | 96,13 |
| | Jumlah | | | 207,58 |
| 6 | 53 | 150 | 83,4 | 125,09 |
| | 122 | 150 | 90,8 | 136,18 |
| | 265 | 150 | 45,3 | 67,98 |
| | 361 | 160 | 74,3 | 118,88 |
| | 412 | 100 | 74,3 | 74,3 |
| | 570 | 100 | 89,3 | 89,32 |
| | 777 | 250 | 64,8 | 162 |
| | Jumlah | | | 773,75 |

Sumber: PT. PLN (PERSERO) DISTRIBUTI
JATIM C. ARANG M. LANG

Data Beban Tiap Seksi Penyulang Glinting

| Seksi | No. Trafo | Nominal (kVA) | Persen Beban (%) | Beban Trafo (kVA) | |
|-------|-----------|---------------|------------------|-------------------|--------|
| 1 | 93 | 160 | 72,3 | 72,3 | |
| | 141 | 200 | 76,5 | 153 | |
| | 196 | 160 | 79,4 | 127 | |
| | 575 | 160 | 40,9 | 65,4 | |
| | Jumlah | | | | 417,7 |
| 2 | 16 | 250 | 48,8 | 122,1 | |
| | 51 | 250 | 54,8 | 137 | |
| | 148 | 250 | 108,4 | 271 | |
| | 210 | 250 | 65,6 | 164 | |
| | 281 | 160 | 68,1 | 109 | |
| | 536 | 160 | 72,5 | 116 | |
| | Jumlah | | | | 919,1 |
| 3 | 40 | 200 | 49 | 98 | |
| | 153 | 160 | 40 | 64,02 | |
| | 211 | 150 | 65,6 | 98,4 | |
| | 398 | 150 | 65,6 | 98,4 | |
| | Jumlah | | | | 358,82 |
| 4 | 107 | 100 | 11,1 | 11,38 | |
| | 14 | 160 | 116,9 | 187 | |
| | 165 | 200 | 64,3 | 128,57 | |
| | 437 | 100 | 58,6 | 58,6 | |
| | 449 | 100 | 65,6 | 65,6 | |
| | 762 | 100 | 65,6 | 65,6 | |
| | Jumlah | | | | 549,70 |

Sumber: PT. PLN (PERSERO) DISTRIBUTI
JATIM C. ARANG M. LANG

**Data Beban Tiap Seksi
Penyulang Asahan**

| Seksi | No. Trafo | Nominal (kVA) | Persen Beban (%) | Beban Trafo (kVA) |
|-------|-----------|---------------|------------------|-------------------|
| 1 | 39 | 200 | 66,5 | 133 |
| | 88 | 250 | 75 | 187,5 |
| | 124 | 200 | 73,4 | 146,7 |
| | 125 | 160 | 24,4 | 39 |
| | 128 | 100 | 41,9 | 41,9 |
| | 267 | 160 | 75 | 120 |
| | 445 | 160 | 75 | 120 |
| | 495 | 200 | 5 | 10 |
| | 621 | 160 | 12,6 | 20,1 |
| | 622 | 100 | 75 | 75 |
| | 623 | 50 | 75 | 37,5 |
| | 654 | 160 | 23 | 36,9 |
| | 660 | 160 | 6,8 | 10,9 |
| | 668 | 250 | 75 | 187,5 |
| | 730 | 100 | 75 | 75 |
| 757 | 630 | 75 | 472,5 | |
| 865 | 200 | 75 | 150 | |
| | Jumlah | | | 1863,5 |
| 2 | 372 | 250 | 49,6 | 124 |
| | 373 | 160 | 40,5 | 64,8 |
| | 434 | 160 | 75 | 120 |
| | 694 | 160 | 10,4 | 16,6 |
| | 697 | 200 | 74,2 | 148,3 |
| | 726 | 160 | 10,6 | 17 |
| | 734 | 160 | 9,6 | 15,3 |
| | 735 | 160 | 39,8 | 63,6 |
| | 738 | 200 | 53,2 | 106,3 |
| | 767 | 100 | 75 | 75 |
| | 843 | 630 | 75 | 472,5 |
| | 875 | 160 | 38,3 | 61,3 |
| | 899 | 250 | 75 | 187,5 |
| | 913 | 100 | 75 | 75 |
| | 924 | 200 | 75 | 150 |
| 960 | 160 | 75 | 120 | |
| | Jumlah | | | 1817,2 |
| 3 | 285 | 25 | 4,8 | 1,2 |
| | 286 | 160 | 97,7 | 156,2 |
| | 582 | 160 | 13 | 20,8 |
| | 752 | 200 | 34,2 | 68,3 |
| | Jumlah | | | 246,5 |
| 4 | 26A | 100 | 75 | 75 |
| | 354 | 150 | 80 | 120 |
| | 667 | 200 | 75 | 180 |
| | 815 | 630 | 75 | 472,5 |
| | Jumlah | | | 817,5 |
| 5 | 2 | 160 | 75 | 120 |
| | 3 | 200 | 81,5 | 163 |
| | 234 | 100 | 79,8 | 79,88 |
| | 279 | 160 | 28,5 | 45,9 |
| | 314 | 150 | 75 | 112,5 |

| | | | | |
|---|-----|--------|------|---------|
| | 264 | 150 | 75 | 112,5 |
| | 384 | 75 | 75 | 56,25 |
| | 528 | 630 | 75 | 472,5 |
| | 624 | 550 | 75 | 412,5 |
| | 818 | 200 | 46,8 | 93,5 |
| | 906 | 160 | 75 | 120 |
| | 910 | 100 | 75 | 75 |
| | | Jumlah | | 1863,22 |
| 6 | 80A | 75 | 75 | 56,25 |
| | | Jumlah | | 56,25 |

Sumber: PT PLN (PERSERO) DISKRIBUSI
JATIMBARANG MALANG

Data Beban Tiap Seksi Penyulang Wendit

| Seksi | No. Trafo | Nominal (kVA) | Persen Beban (%) | Beban Trafo (kVA) |
|-------|-----------|------------------|------------------|-------------------|
| 1 | 22 | 200 | 82,8 | 165,5 |
| | | Jumlah | | 165,5 |
| 2 | 215 | 160 | 71,4 | 114,3 |
| | 338 | 250 | 78,7 | 196,8 |
| | 577 | 200 | 89,5 | 179 |
| | | Jumlah | | 490,1 |
| 3 | 28 | 100 | 92,4 | 94,4 |
| | 203 | 100 | 60,9 | 60,9 |
| | 204 | 160 | 63,7 | 101,9 |
| | 222 | 103 ⁰ | 74,4 | 766,32 |
| | 293 | 100 | 82 | 82 |
| | 502 | 250 | 9,1 | 22,8 |
| | 503 | 160 | 74,4 | 119,04 |
| | 608 | 150 | 66,7 | 100 |
| | 649 | 250 | 81,1 | 202,7 |
| | 650 | 200 | 81,5 | 163 |
| | 655 | 100 | 105,8 | 105,8 |
| | 659 | 800 | 74,4 | 595,2 |
| | 676 | 200 | 74,4 | 148,8 |
| | 755 | 630 | 77,3 | 468,72 |
| | 773 | 250 | 67,7 | 169,2 |
| | 828 | 100 | 10 | 10 |
| | 840 | 160 | 83,1 | 133 |
| | 966 | 160 | 5,9 | 9,5 |
| 967 | 160 | 0,6 | 1 | |
| 968 | 160 | 30,8 | 49,2 | |
| 991 | 160 | 74,4 | 119,2 | |
| | | Jumlah | 3520,52 | |
| 4 | 378 | 160 | 64,4 | 103 |
| | 727 | 160 | 74,4 | 119,04 |
| | | Jumlah | | 222,04 |
| 5 | 185 | 150 | 73,4 | 110,1 |
| | 254 | 150 | 74,4 | 111,6 |
| | 428 | 100 | 77,8 | 77,8 |
| | 496 | 630 | 74,4 | 468,72 |
| | | Jumlah | 768,22 | |

| | | | | |
|--------|-----|-----|--------|-------|
| 6 | 537 | 250 | 53,1 | 130,2 |
| | 681 | 150 | 58,1 | 87,2 |
| | 699 | 100 | 74,4 | 74,4 |
| | 705 | 50 | 74,4 | 37,2 |
| | 714 | 160 | 61,1 | 97,7 |
| | 715 | 160 | 76,4 | 122,2 |
| | 763 | 160 | 75,1 | 120,1 |
| | 768 | 160 | 25,5 | 40,8 |
| | 769 | 160 | 43,4 | 69,4 |
| | 822 | 200 | 44,7 | 89,3 |
| | 844 | 200 | 23,9 | 47,8 |
| | 854 | 100 | 43,1 | 43,1 |
| | 864 | 200 | 27,3 | 54,6 |
| | 923 | 160 | 36,4 | 58,3 |
| | 955 | 160 | 28,4 | 45,5 |
| | 956 | 160 | 28,6 | 45,8 |
| | 974 | 160 | 28,6 | 45,8 |
| Jumlah | | | 1209,4 | |

Sumber: PT. PLN (PERSERO) INSTITUSI
 LAPANGAN MELANG

Pengelompokan Switch

| Penyulang | Seksi | Switch | Jumlah beban (kVA) |
|-----------|-------|--------|--------------------|
| Mawar | 1 | 1 | 1748,55 |
| | 2 | 2 | 956,84 |
| | 3 | 3 | 1855,27 |
| | 4 | 4 | 1074,5 |
| | 5 | 5 | 207,55 |
| | 6 | 6 | 773,75 |
| Glantung | 1 | 7 | 417,7 |
| | 2 | 8 | 919,1 |
| | 3 | 9 | 358,82 |
| | 4 | 10 | 549,75 |
| Asahan | 1 | 11 | 1863,5 |
| | 2 | 12 | 1817,2 |
| | 3 | 13 | 246,5 |
| | 4 | 14 | 817,5 |
| | 5 | 15 | 1863,22 |
| | 6 | 16 | 56,25 |
| Wendit | 1 | 17 | 165,5 |
| | 2 | 18 | 400,1 |
| | 3 | 19 | 3520,52 |
| | 4 | 20 | 222,04 |
| | 5 | 21 | 768,22 |
| | 6 | 22 | 1209,4 |

Data Jaringan Distribusi GI Blimbing Malang

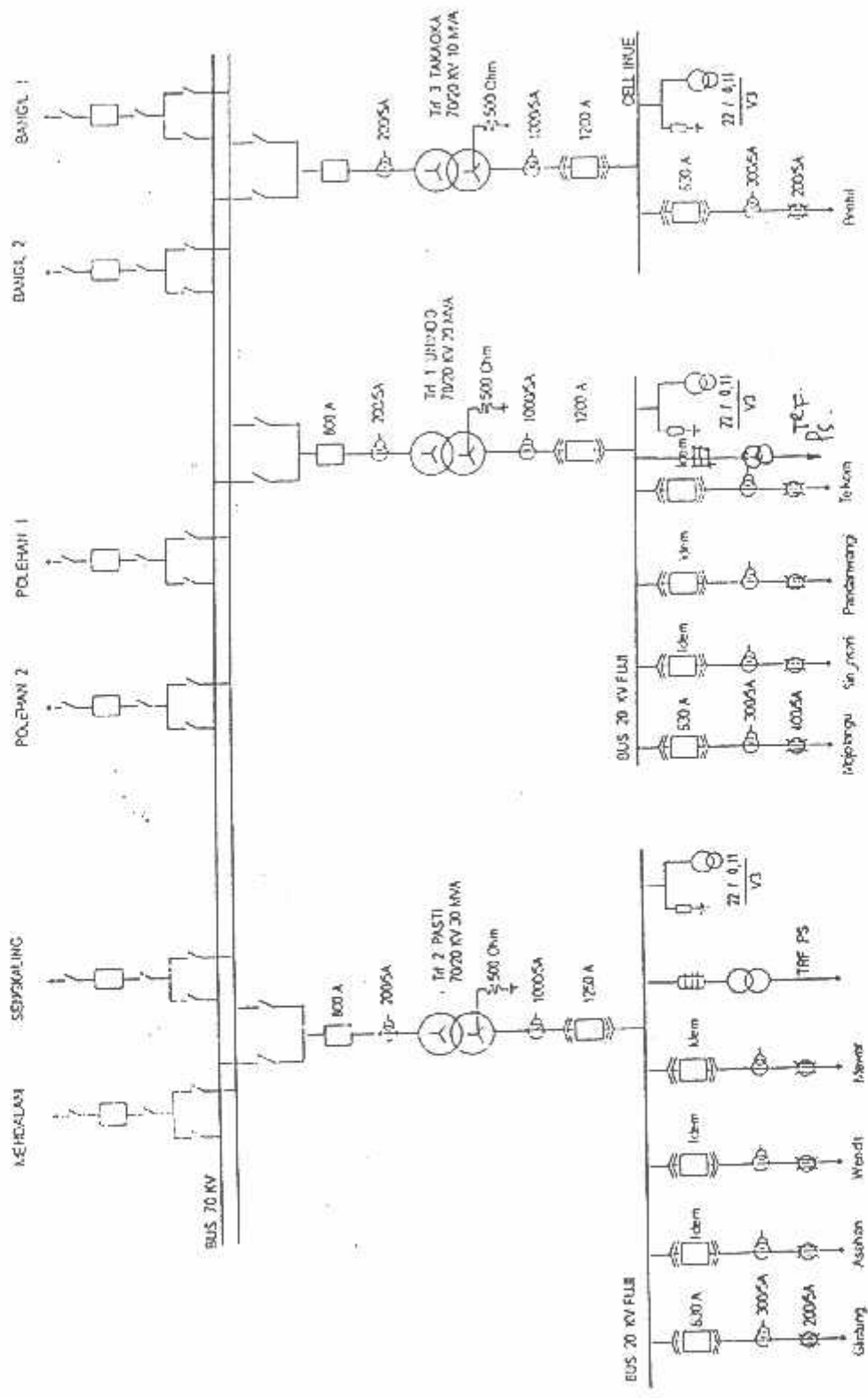
| Bus Pangkal | Bus Ujung | P Bus Ujung (MW) | Q Bus Ujung (MVAR) | Jarak (km) | Resistansi R (ohm) | Reaktansi X (ohm) |
|-------------|-----------|------------------|--------------------|------------|--------------------|-------------------|
| 1 | 2 | 1,4276 | 1,0707 | 5,616 | 1,2142 | 1,8560 |
| 2 | 3 | 0,7655 | 0,5741 | 4,085 | 0,8832 | 1,3501 |
| 3 | 4 | 1,4842 | 1,1132 | 5,040 | 1,0896 | 1,6657 |
| 4 | 5 | 0,8594 | 0,6446 | 3,897 | 0,8425 | 1,2880 |
| 5 | 6 | 0,1661 | 0,1245 | 0,677 | 0,1464 | 0,2238 |
| 6 | 7 | 0,6190 | 0,4643 | 1,678 | 0,3862 | 0,5546 |
| 1 | 8 | 0,3342 | 0,3506 | 1,381 | 0,2986 | 0,4564 |
| 8 | 9 | 0,7353 | 0,5515 | 1,194 | 0,2581 | 0,3946 |
| 9 | 10 | 0,2871 | 0,2153 | 1,016 | 0,2197 | 0,3358 |
| 10 | 11 | 0,4398 | 0,3299 | 1,492 | 0,3226 | 0,4931 |
| 1 | 12 | 1,4908 | 1,1181 | 2,474 | 0,5349 | 0,8177 |
| 12 | 13 | 1,4539 | 1,0903 | 3,443 | 0,7444 | 1,1379 |
| 12 | 14 | 0,1972 | 0,1479 | 1,646 | 0,3559 | 0,5540 |
| 14 | 15 | 0,6540 | 0,4905 | 1,183 | 0,2558 | 0,3910 |
| 14 | 16 | 1,4906 | 1,1179 | 2,745 | 0,5935 | 0,9072 |
| 16 | 17 | 0,0450 | 0,0338 | 0,131 | 0,0283 | 0,0433 |
| 1 | 18 | 0,1324 | 0,0993 | 1,782 | 0,3853 | 0,5890 |
| 18 | 19 | 0,3921 | 0,2941 | 3,031 | 0,6553 | 1,0017 |
| 18 | 20 | 2,8164 | 2,1123 | 7,639 | 1,6516 | 2,5247 |
| 19 | 21 | 0,1776 | 0,1332 | 1,549 | 0,3349 | 0,5119 |
| 21 | 22 | 0,6146 | 0,4609 | 1,476 | 0,3191 | 0,4878 |
| 21 | 23 | 0,9675 | 0,7256 | 4,689 | 1,0138 | 1,5497 |
| 4 | 11 | - | - | 0,207 | 0,0488 | 0,0684 |
| 9 | 13 | - | - | 0,173 | 0,0374 | 0,0572 |
| 15 | 19 | - | - | 1,114 | 0,2408 | 0,3682 |
| 5 | 16 | - | - | 0,262 | 0,0566 | 0,0866 |

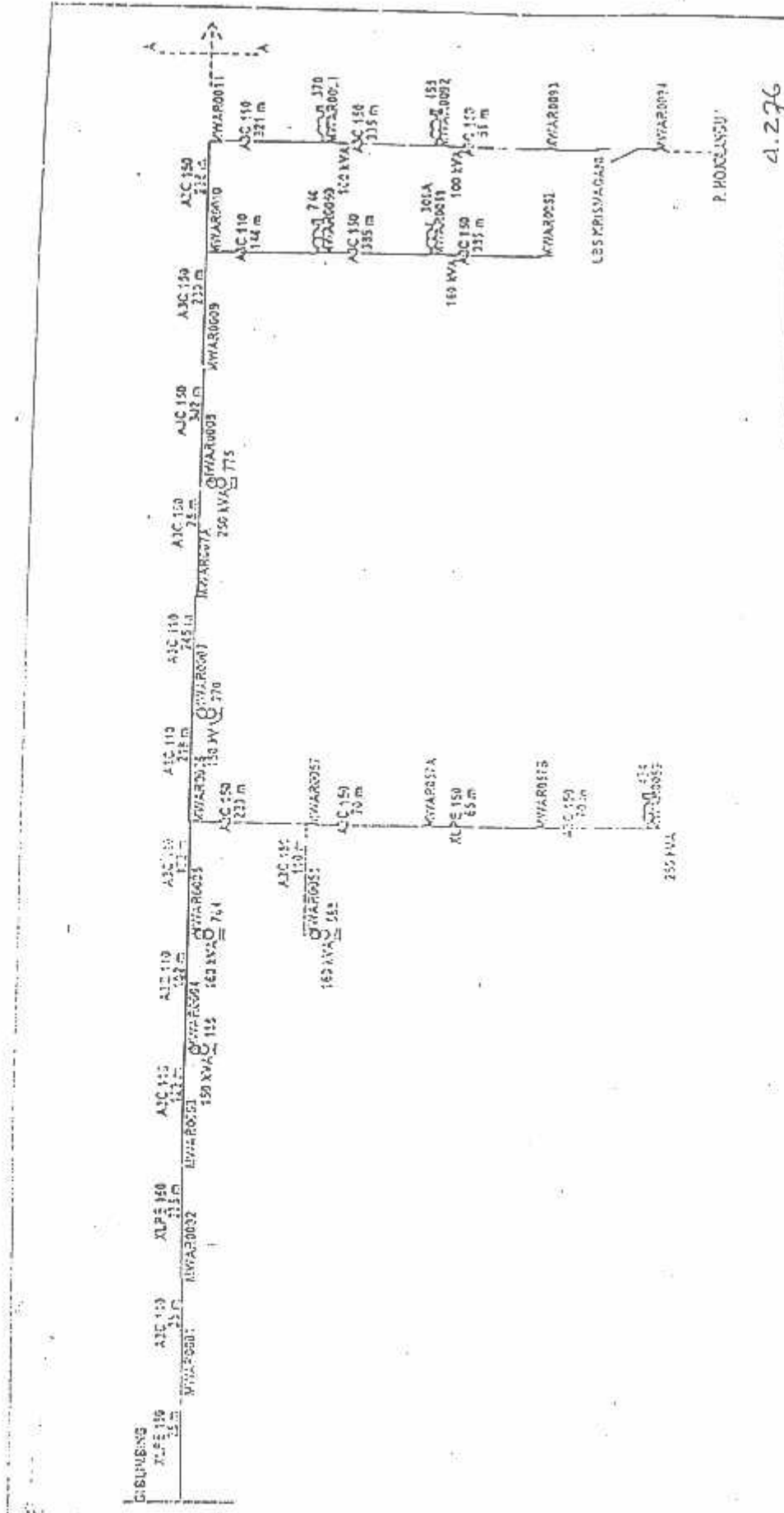
Sumber PT PLN (PERSERO) DISTRIBUSI
JATIM CABANG MALANG

Data Penyulang Sistem 20 kV GI Blimbing Malang


| Penampang Nominal (mm ²) | Jari-jari (mm) | Urut | GMR (mm) | Impedansi Urutan Posistif (Ω/km) | Impedansi Urutan Nol (Ω/km) |
|--------------------------------------|----------------|------|----------|----------------------------------|-----------------------------|
| 16 | 2,2563 | 7 | 1,6380 | 2,0161 + j0,4036 | 2,1641 + j1,6911 |
| 25 | 2,8203 | 7 | 2,0475 | 1,2903 + j0,3895 | 1,4384 + j1,6770 |
| 35 | 3,3371 | 7 | 2,4227 | 0,9217 + j0,3790 | 1,0697 + j1,6665 |
| 50 | 3,9886 | 7 | 2,8957 | 0,6452 + j0,3678 | 0,7932 + j1,6653 |
| 70 | 4,7193 | 7 | 3,4262 | 0,4608 + j0,3572 | 0,6088 + j1,6447 |
| 95 | 5,4979 | 19 | 4,1674 | 0,3396 + j0,3449 | 0,4876 + j1,6324 |
| 120 | 6,1791 | 19 | 4,6837 | 0,2688 + j0,3376 | 0,4168 + j1,6251 |
| 150 | 6,9084 | 19 | 5,2365 | 0,2162 + j0,3305 | 0,3631 + j1,6180 |
| 185 | 7,6722 | 19 | 5,8155 | 0,1744 + j0,3239 | 0,3224 + j1,6114 |
| 240 | 8,7386 | 19 | 6,6238 | 0,1344 + j0,3158 | 0,2824 + j1,6033 |

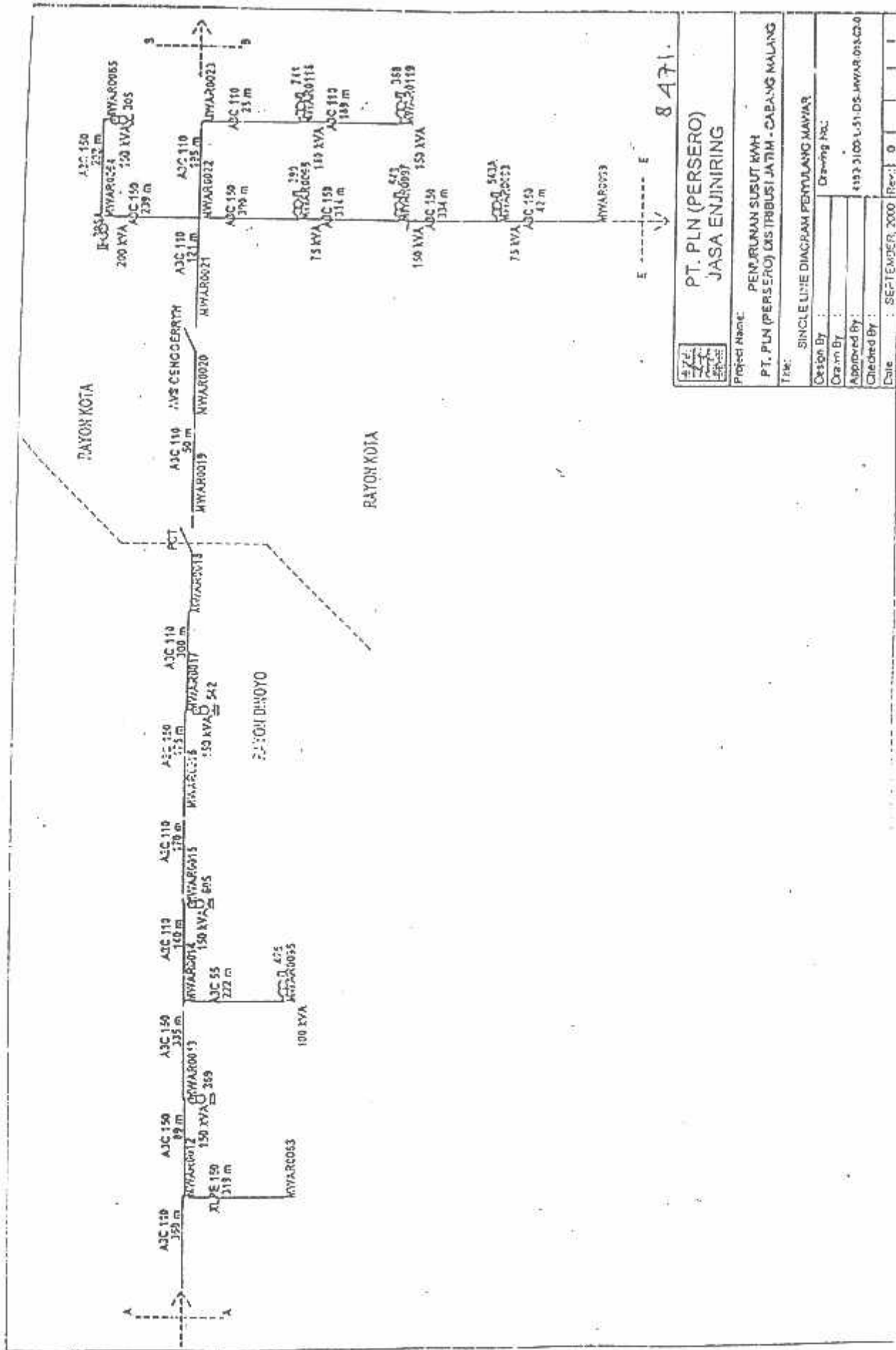
Sumber PT PLN (PERSERO) DISTRIBUSI
JATIM CABANG MALANG





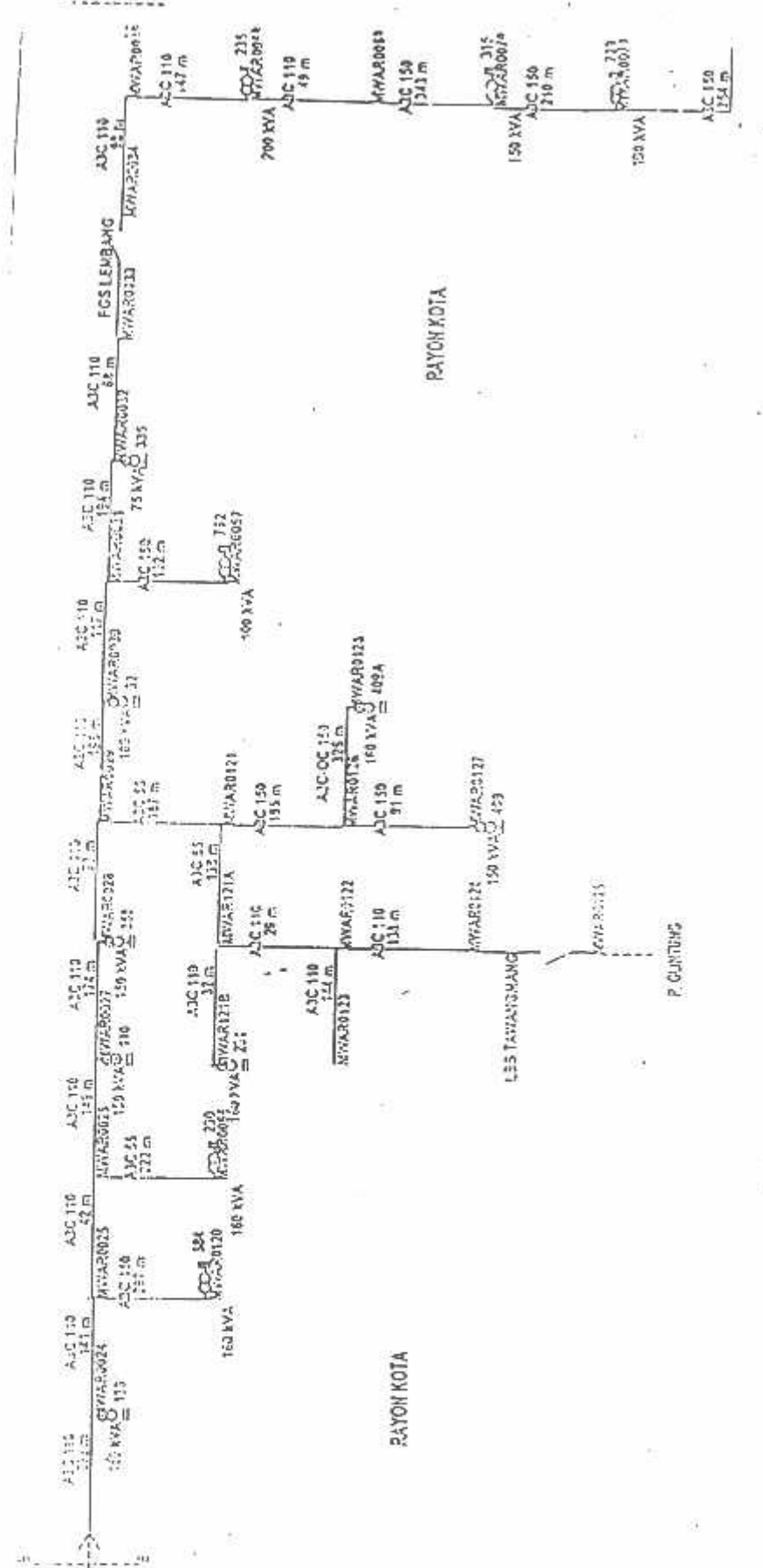
4.276

| | |
|---|---------------------------------|
|  | |
| Project Nam : PT. PLN (PERSERO) PENYULANG SUSUT KWH PT. PLN (PERSERO) DISTRIBUSI JA TIM - CABANG HALANG | |
| Title : SINGLE LINE DIAGRAM PENYULANG ANAYAR | |
| Design By : | Drawing No.: |
| Drawn By : | |
| Approved By : | 416231701.011.D5.00142.003.01.0 |
| Checked By : | |
| Date : | SEPTEMBER 2010 Rev. 0 |




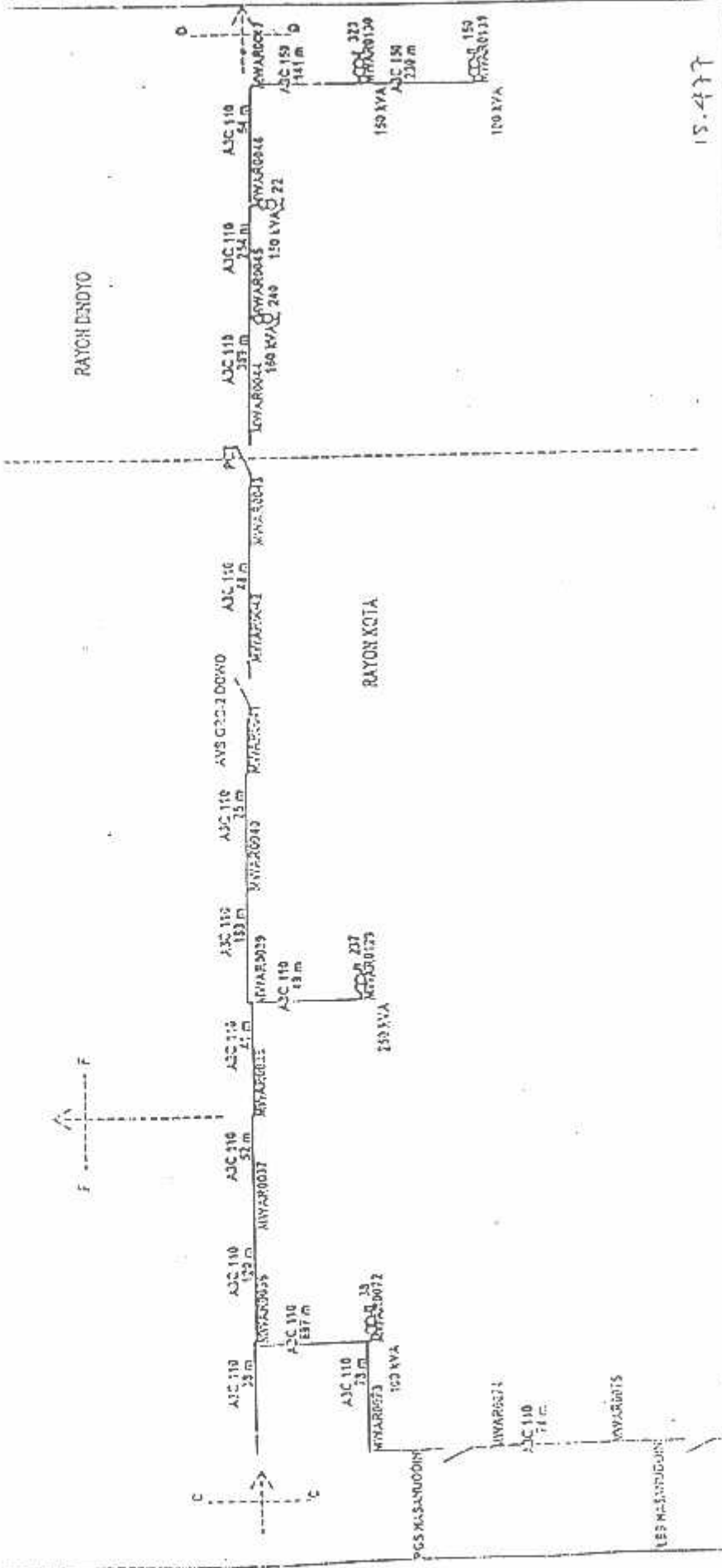
**PT. PLN (PERSERO)
JASA ENJINIRING**

| | |
|---|----------------------------------|
| Project Name: | |
| PENJURUAN SUSUT KWH PT. PLN (PERSERO) DIS TRIBUTSI JATIM - CABANG MALANG | |
| Title: | |
| SINGLE LINE DIAGRAM PENYULANG MAYAR | |
| Design By : | Drawing No.: |
| Drawn By : | |
| Approved By : | 4193.0100.1.51-DS-MAYAR 014.02.0 |
| Checked By : | |
| Date : | SEPTEMBER 2000 [Rev:] 0 |



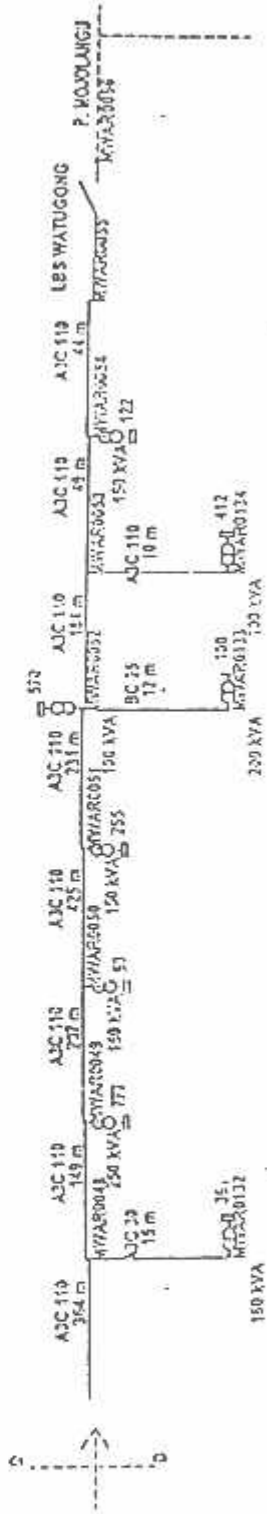
12.573

| | |
|--|-----------------------------|
|  | |
| PT. PLN (PERSERO): JASA ENJINIRING | |
| Project Name: PENUNJAN SUSUT RUMAH PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG | |
| Title: SINGLE LINE DIAGRAM PENYULANG MAYAR | |
| Design By : | Drawing No.: |
| Drawn By : | |
| Approved By : | 4100-3100-1-11-05-MW/P-1111 |
| Checked By : | |
| Date : | SEPTEMBER, 2020 |
| Rev: | 0 |



| | |
|---|---------------|
|  | |
| PT. PLN (PERSERO) JASA ENJINIRING | |
| Project Name: PEMURNAN SUSUT MW1 PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG | |
| Title: SINGLE LINE DIAGRAM PENYULANG MAWAR | |
| Design By : | Drawing No.: |
| Check By : | Approved By : |
| Checked By : | Date : |
| : SEPTEMBER, 2000 | Rev: 0 |

15.477



17.167



PT. PLN (PERSERO)
JASA ENJINIRING

Project Name: PENURUNAN SUSUT KVH
PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG HALANG

Task: SINGLE LINE DIAGRAM PERPLANAN JAWAB

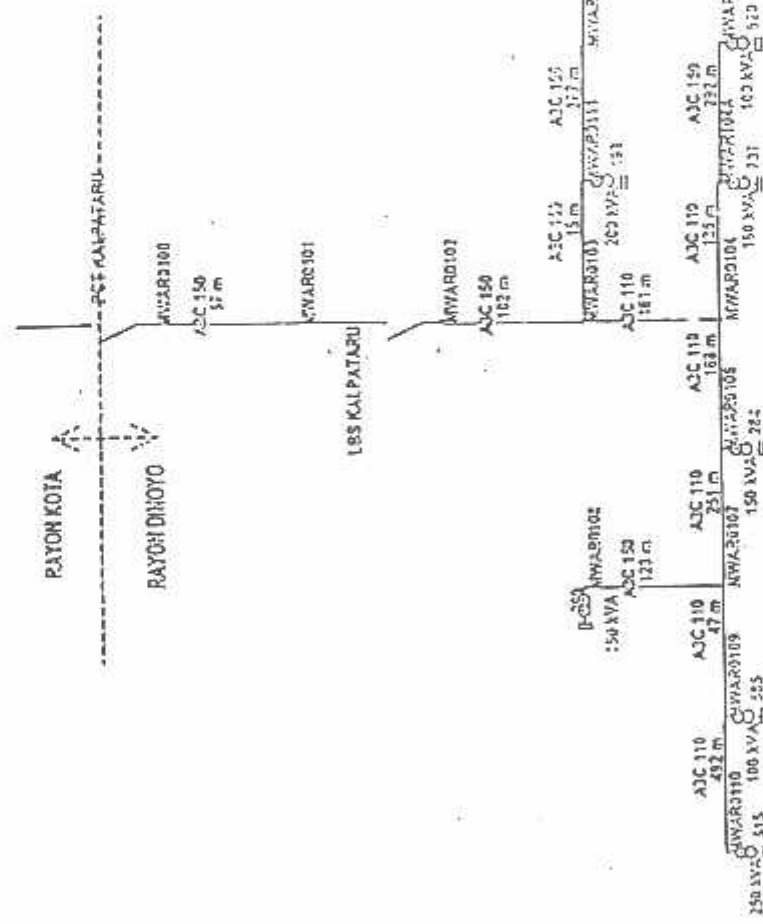
Design By: Dwi Wng Hal

Drawn By:

Approved By: #102.3105.1-31-05-KOVAR-BIB 25.0

Checked By:

Date: SEPTEMBER 2000 REV: 0

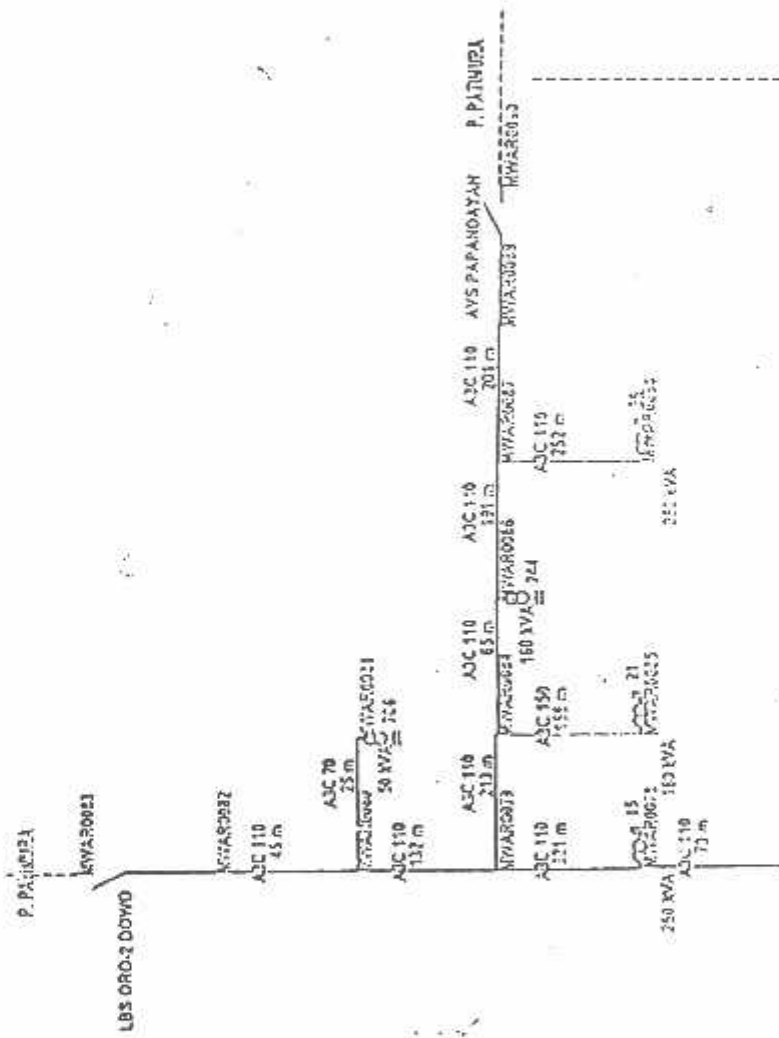


21. 309




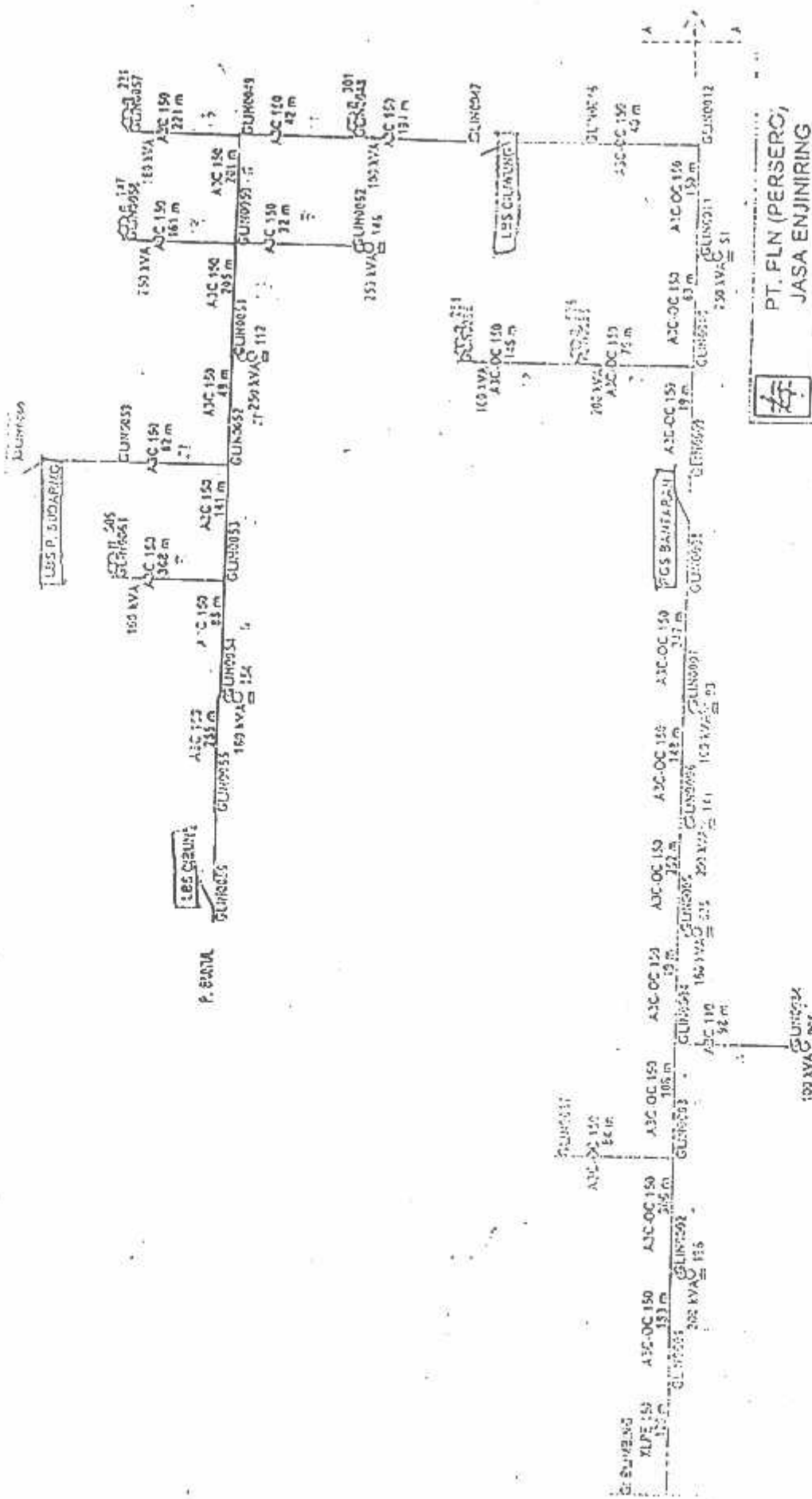
PT. PLN (PERSERO)
JASA ENJINIRING

| | | |
|------------------------------------|-------------------------------|--|
| Project Name: | | PEMUNIAN SUSUT KWH |
| Project Name: | | PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG |
| Title: | | |
| SINGLE LINE DIAGRAM PEMULANG MAWAR | | |
| Design By : | Drawing No. : | |
| Drawn By : | | |
| Approved By : | K19231001-31-05-WWAR-015-05-0 | |
| Checked By : | | |
| Date : | SEPTEMBER 2000 | Rev. : 0 |



PAYONGDINYO

| | |
|--|---------------------------------|
|  | |
| Project Name: PERMULAN SUSUT KWH PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG | |
| Title: SINGLE LINE DIAGRAM PERMULAN HAWAR 2.2.903 | |
| Design By : | Drawing No.: |
| Drawn By : | |
| Approved By : | 4102-3100-L-51-09-MHAR-015-07-0 |
| Checked By : | |
| Date : | SEPTEMBER, 2000 Rev. 0 |



**PT. PLN (PERSERO)
JASA ENJINERING**

Project Name: **PEMBANGUNAN SALURAN TENAGA LISTRIK (PT. PLN (PERSERO) DISTRIBUTUSI JATIM - CARANG MAJAPITAH**

Scale: **1:400**

Single Line Diagram Pelelutan Guntung

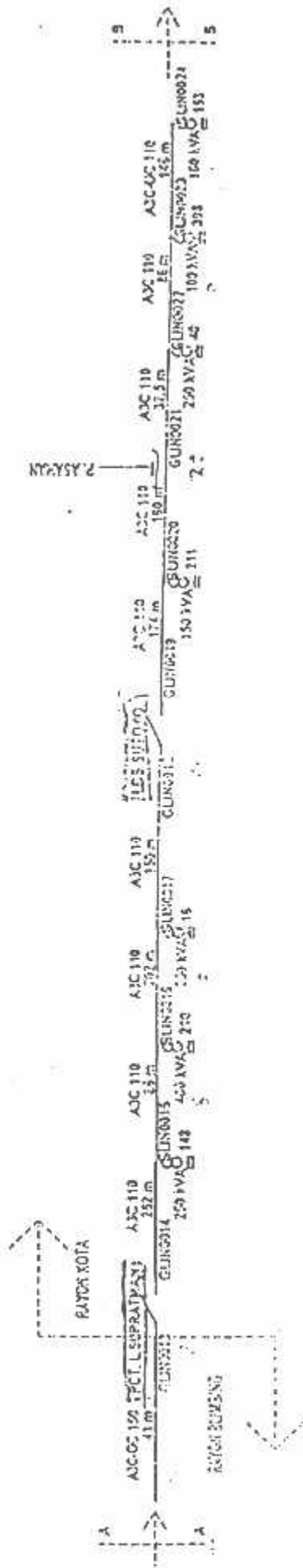
Drawn By: _____

Checked By: _____

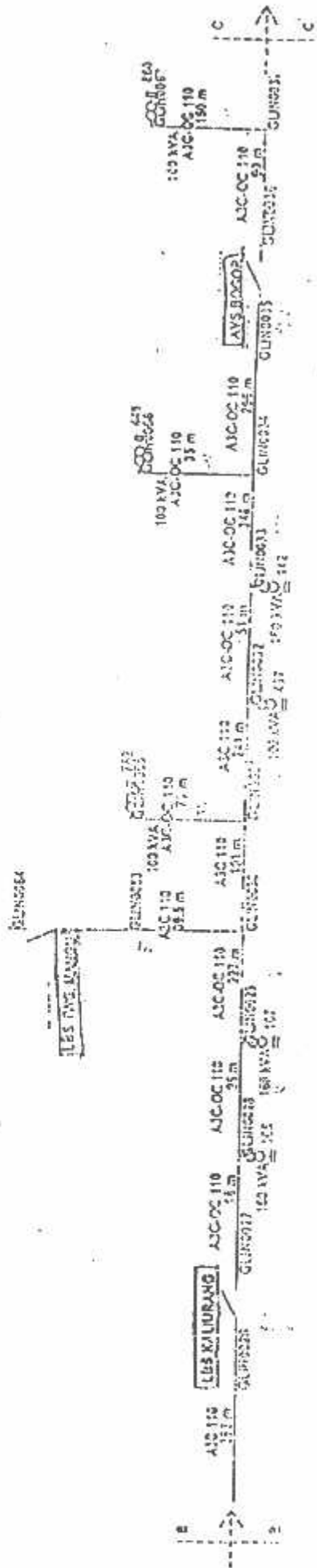
Approved By: _____

Project No: **4102/3120/1-S1-05-01/04/01/19**

Date: **10 SEPTEMBER 2009**



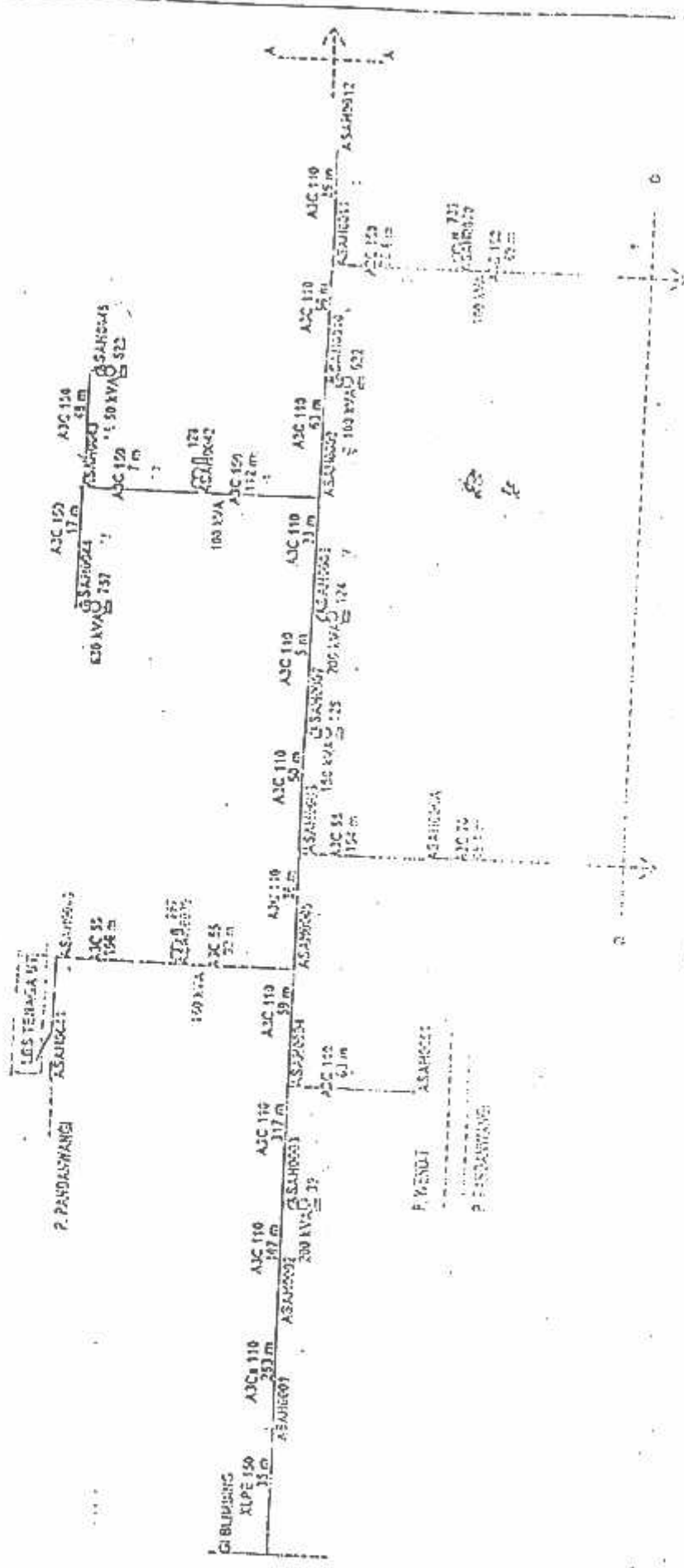
| | |
|---|----------------------|
| PT. PLN (PERSERO) JASA ENJINIRING | |
| Project Name: PENURUAN SUDUT WAH PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG | |
| Job: SINGLE LINE DIAGRAM PERMULANGAN GURUPU | |
| Design By: Drawing No: | Checked By: Date: |
| Drawn By: Approved By: | Checked By: Date: |
| Date: | Rev: |



PT. PLN (PERSERO) JASA ENJINIRING

| | |
|---------------|--|
| Project Name: | PENURUAN SUSUT KWH PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG KALIJATI |
| Title: | SINGLE LINE DIAGRAM PENYUSUTAN GUNUNG |
| Design By: | Cahyani |
| Drawn By: | |
| Approved By: | |
| Checked By: | |
| Date: | SEPTEMBER 2009 |

Rev: 0

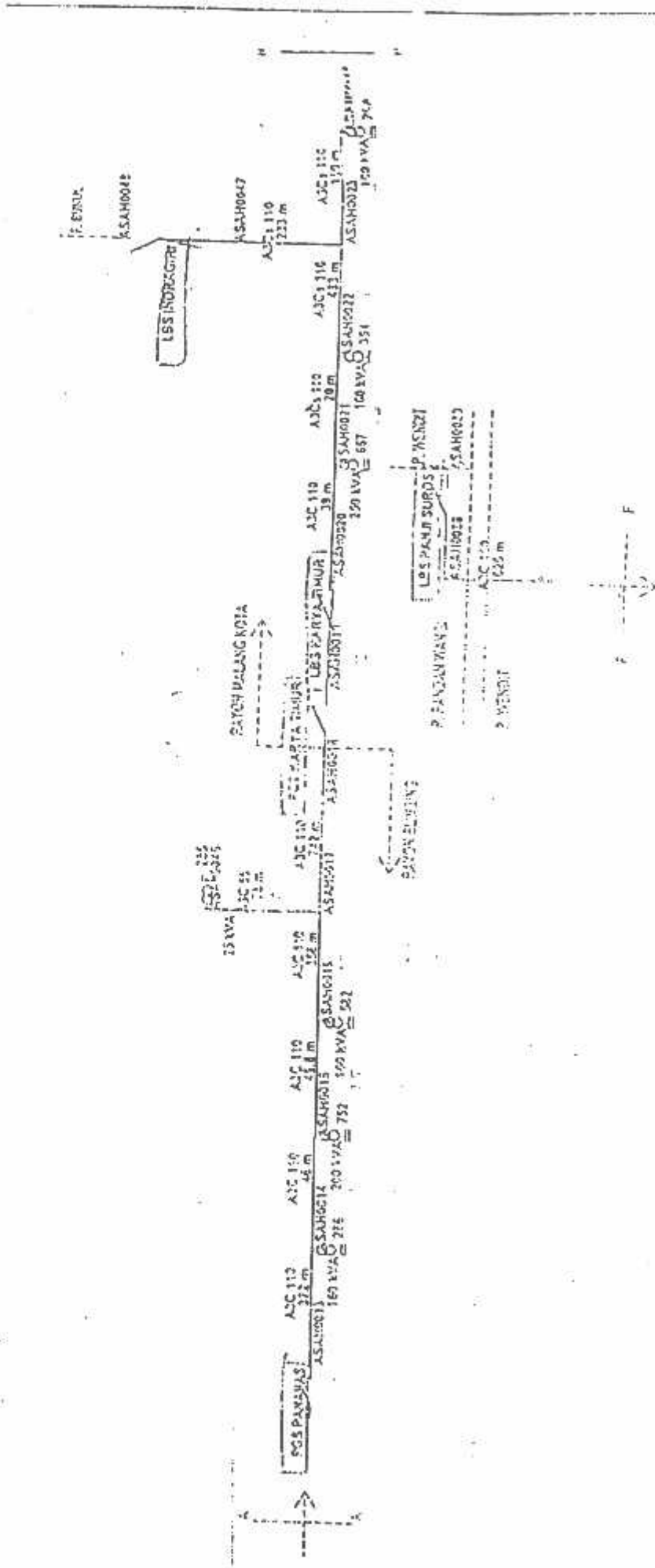


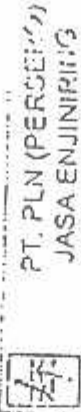
**PT. PLN (PERSERO)
JASA ENJINIRING**

PROJEK Nama: **PERAWATAN SUBSTANSI**
 PT. PLN (PERSERO) DISTRIBUTASI TENAGA LISTRIK DI DAERAH MALANG

Disusun By: **...**
 Diperiksa By: **...**
 Disetujui By: **...**
 Tanggal: **...**

4002 3000 431100 4000013 014





 PT. PLN (PERSERO)

 JASA ENJINIRING

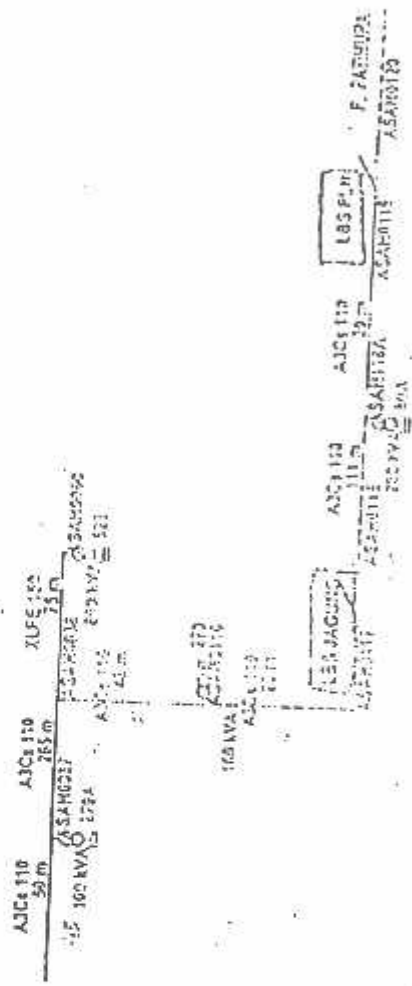
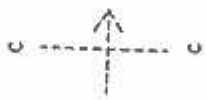
Project Name:

PENURUNAN SUSUT 33KV

 PT. PLN (PERSERO) DISTRIK JATIM

SINGLE LIFE DIAGRAM PERMITS (S.L.P.)

| | |
|--------------|----------------|
| Design By | |
| Submitted By | |
| Approved By | |
| Checked By | |
| Date | SEPTEMBER 2000 |



**PT. PLN (PERSERO)
JASA ENJINERING**

Project Name

PERUMAHAN SUGUT XXII
PT. PLN (PERSERO) DISTRIKUSI JATIM - CADANGAN MALANG

Site

CIRCLE LINE DIACAPAI PEYULANG ASAHAN

Designed By

Engineering

Approved By

1102-2100-05-105-ASAHAN/11/12

Checked By

Supervisor, PLN

Date

0

0

0

0

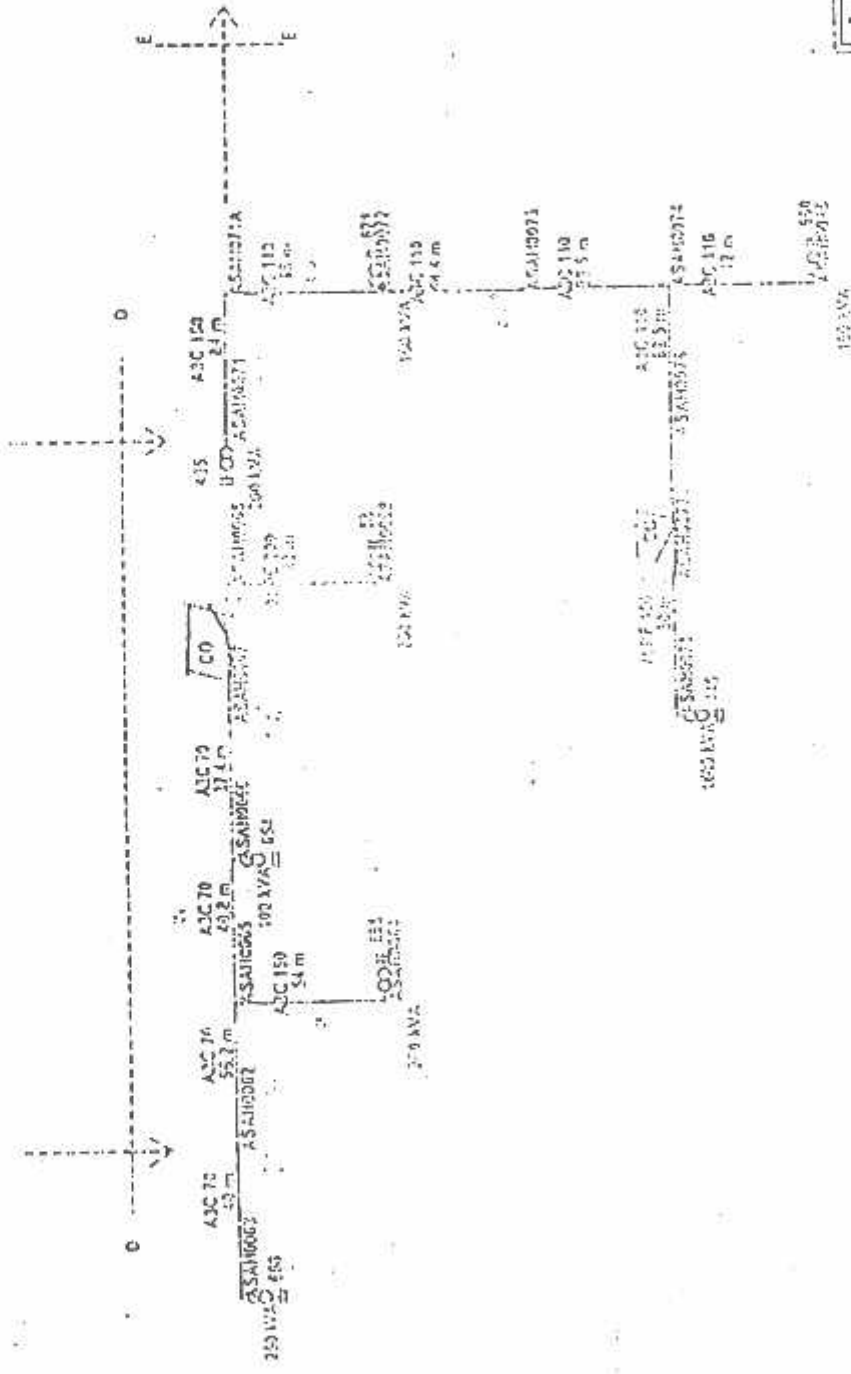
0


0

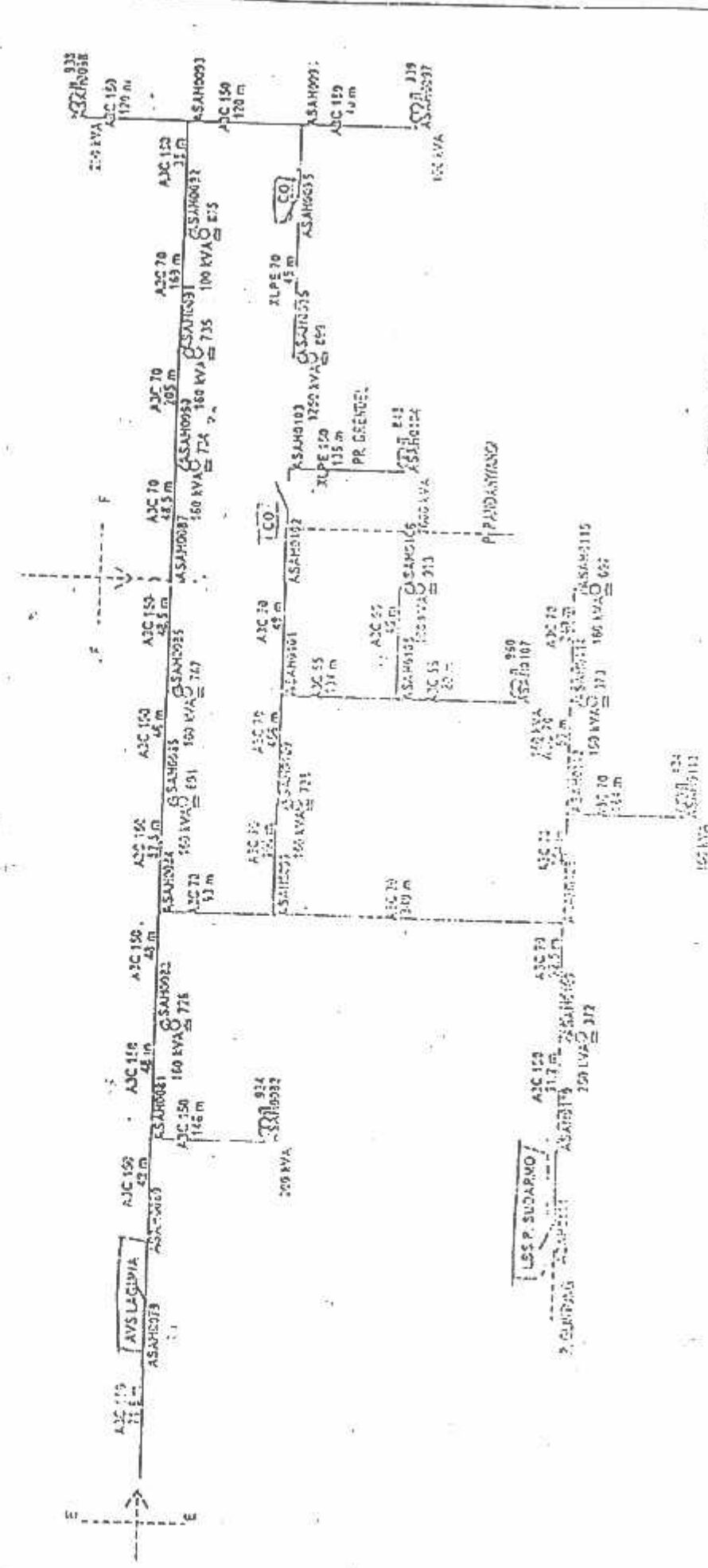
0


0

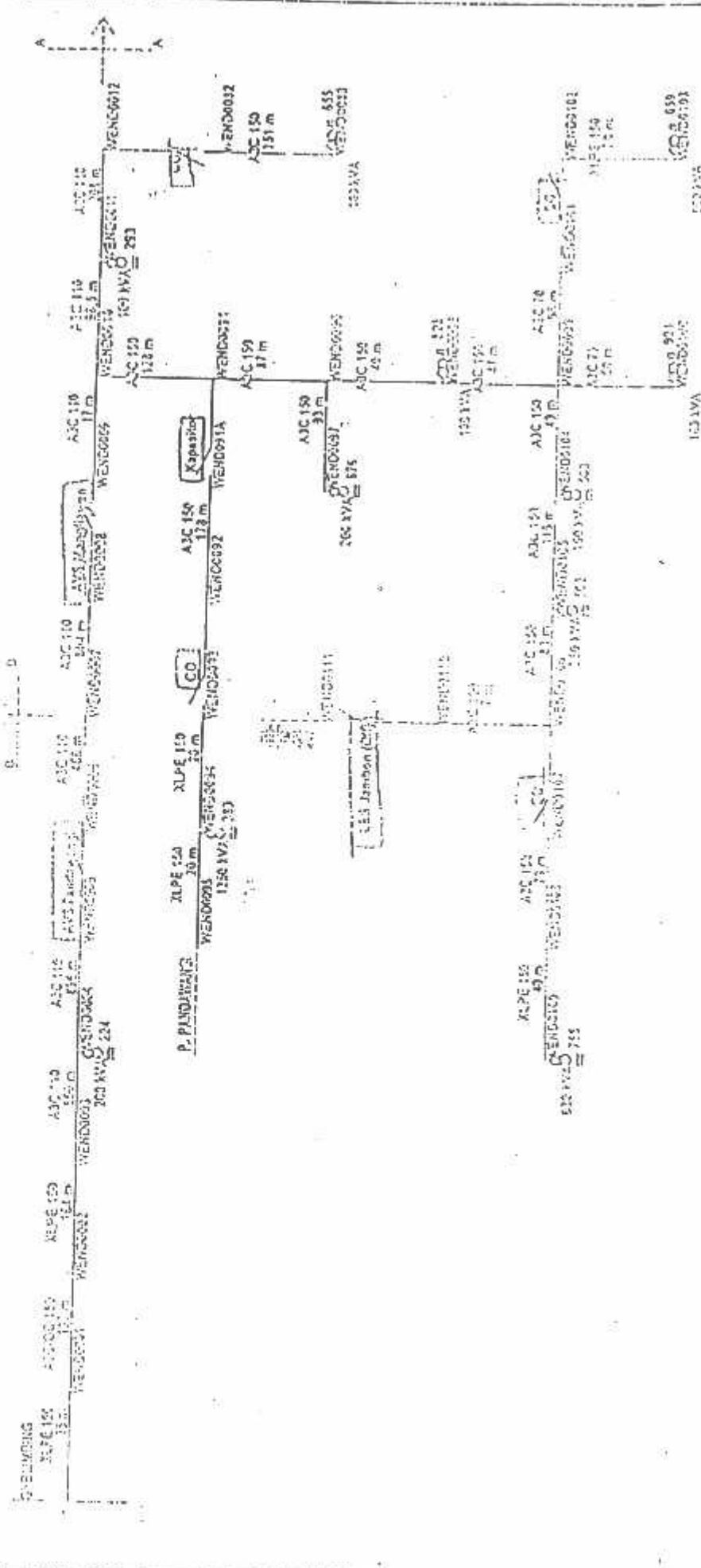
0




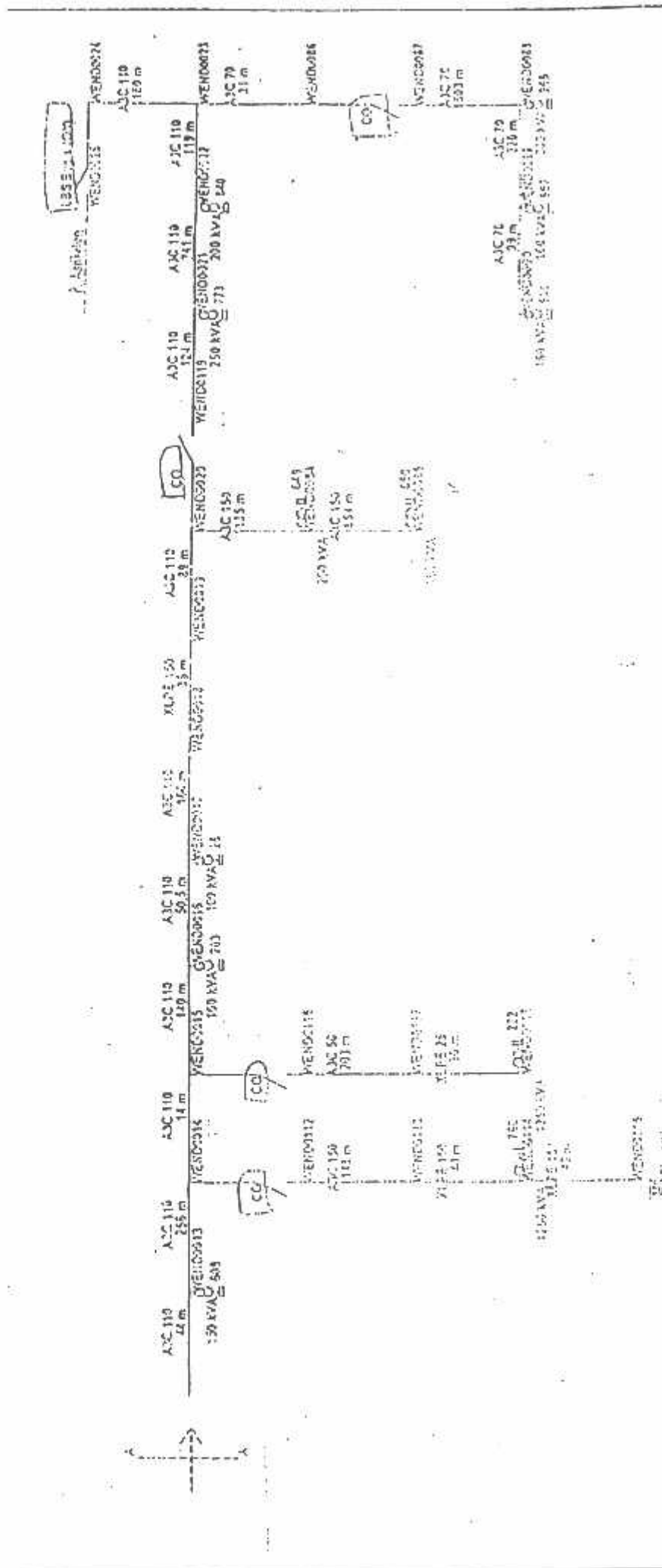
| | |
|---|---|
|  | |
| Project Name | PENURUNAN SUSUT WVM PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALA |
| Task | SINGLE LINE DIAGRAM PENYULANG A-SAKAN |
| Drawn By | Dewi W. S. |
| Approved By | |
| Checked By | |
| Date | SEPTEMBER 2008 (Rev. 0) |



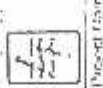
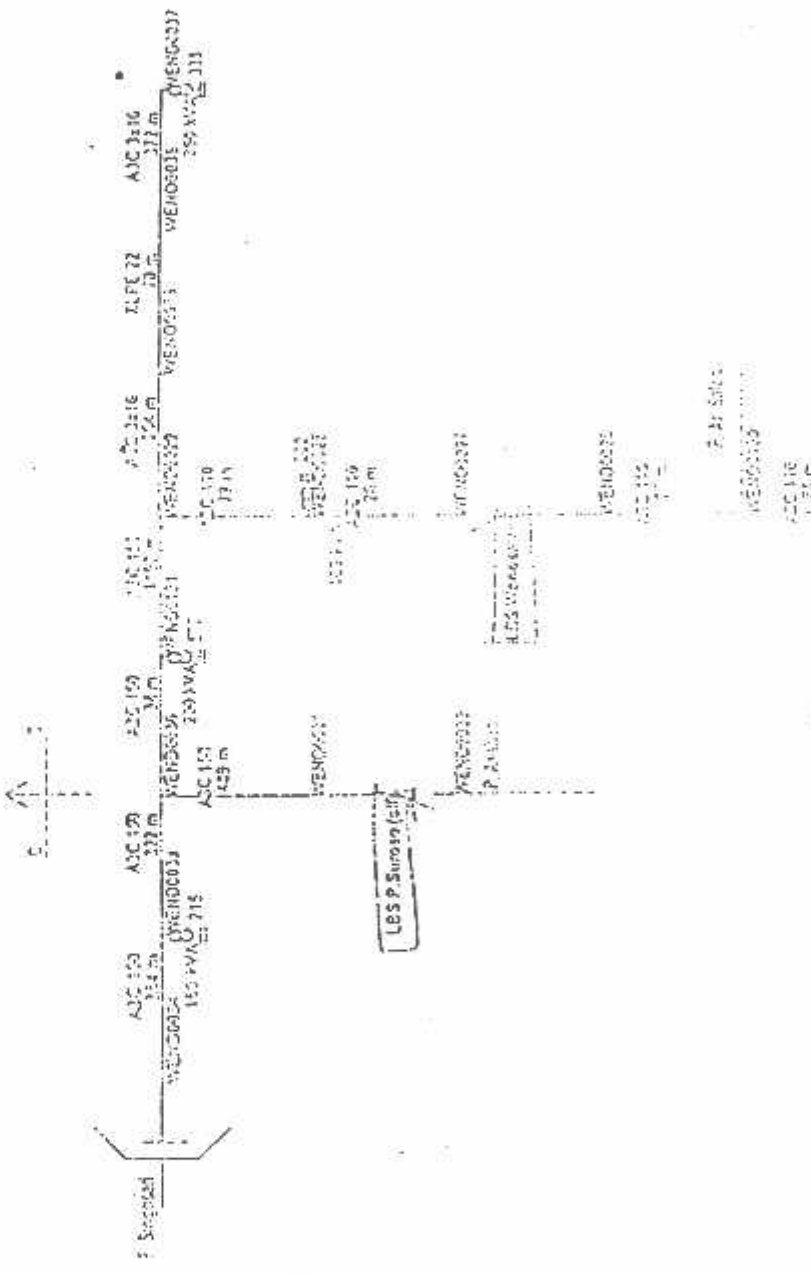
| | | | |
|---|--|---|--|
|  | | Project Name: | |
| | | PERUBAHAN SUDUT WSK PT. PLN (PERSERO) DISTRIKUSI JATIM - CABANG TALANG | |
| Date: | | Drawing No: | |
| Checked By: | | Drawing By: | |
| Approved By: | | Date: | |
| Date: | | Rev: | |
| SEPTEMBER 2000 | | 6 | |



| | |
|--|-------------|
|  PT. PLN (PERSEROG) JASA ENJINIRING | |
| Project Name: | |
| PENURUNAN SUSUT KWH PT. PLN (PERSEROG) DISTRIBUSI JATIM - CABANG MALANG | |
| 187 | |
| Drawn By: | ... |
| Checked By: | ... |
| Date: | SEP 19 1999 |

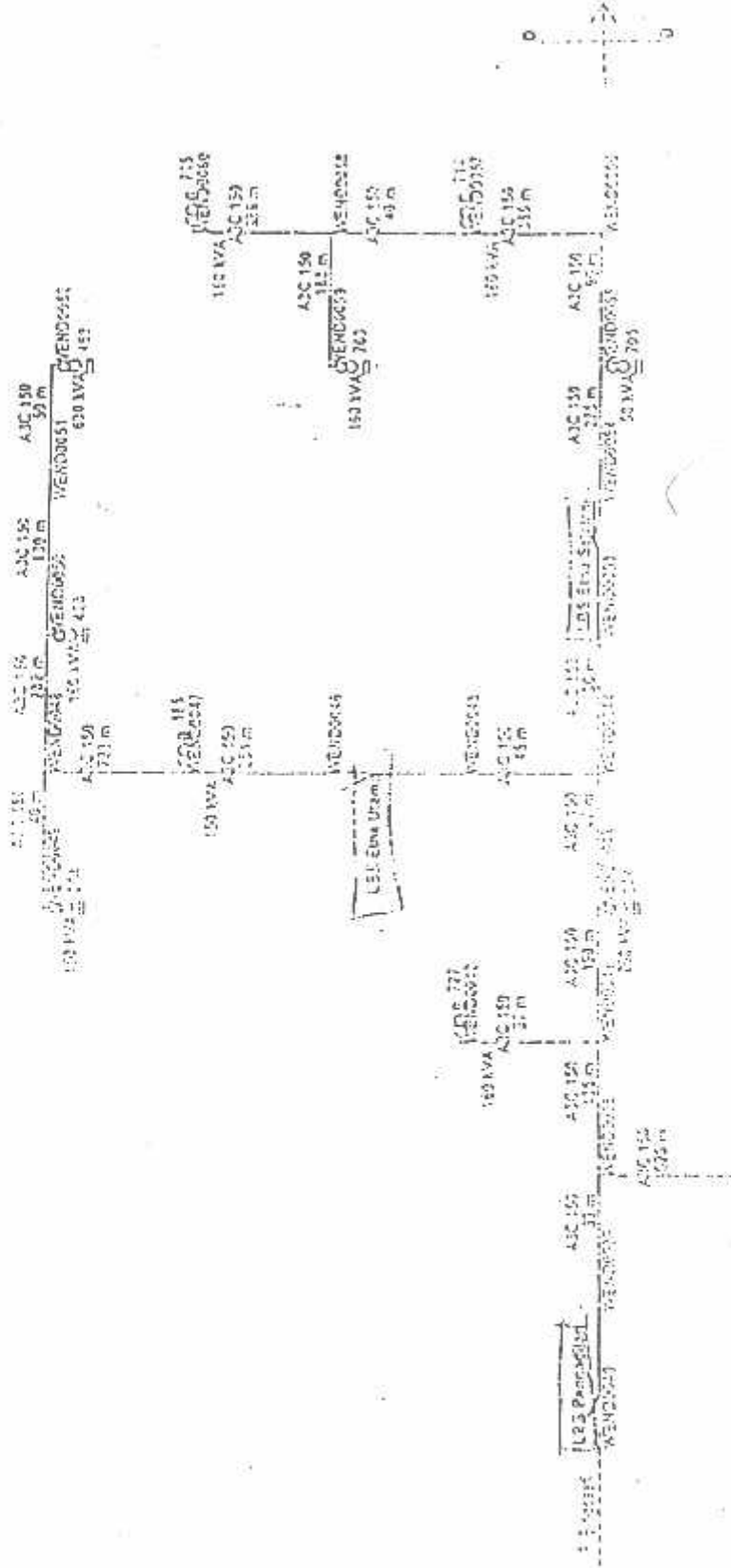


| | |
|---|----------------|
| PT. PLN (PERSERO) JASA ENJINIRING | |
| Project Name: PENURUTAN SUDUT KEM PT. PLN (PERSERO), DISTRIKSI JATIM - CABANG TIMANG | |
| Title: SKEMA DISTRIBUSI PENYULUNG PERBATAS | |
| Desain By: | 02-10-2009 |
| Dimensi By: | |
| Approved By: | |
| Checked By: | |
| Date: | SEPTEMBER 2009 |



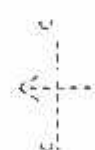
PT. PLN (PERSERO)
JASA ENJINIRING

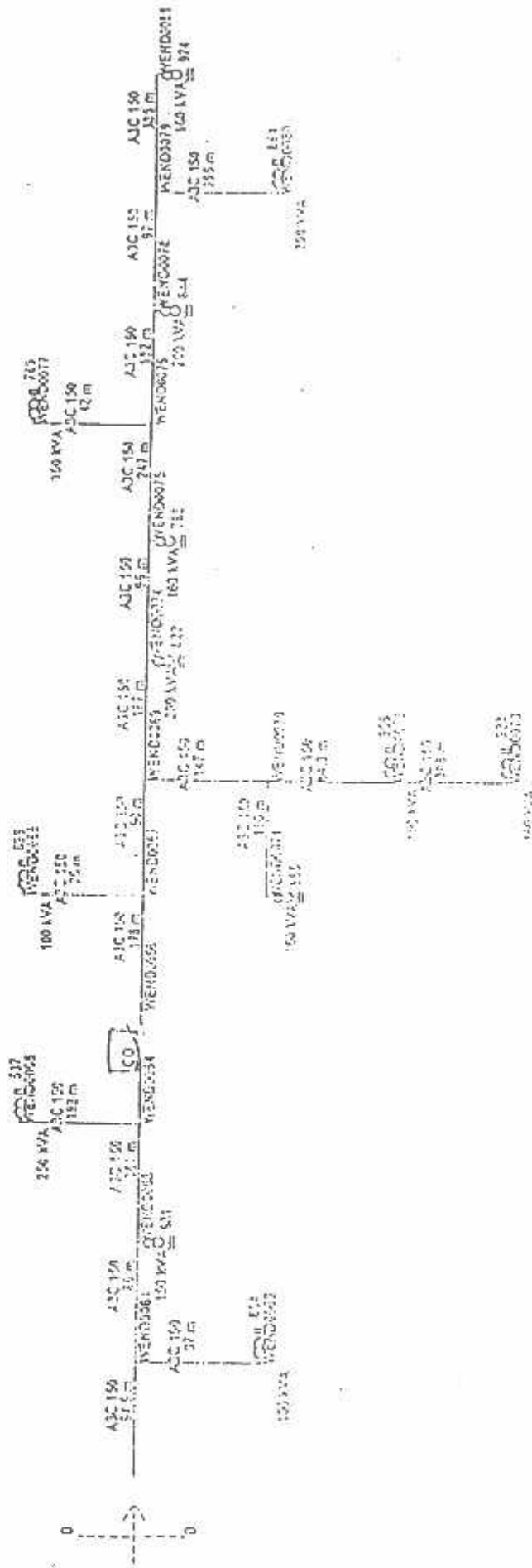
| | | |
|---------------|--|-----------------------------------|
| Project Name: | PERURAHAN SUSUT NEM | |
| | PT. PLN (PERSERO) DISTRIBUSI JAWA TIMUR - CALASAN 100 kV | |
| | SINGLE LINE DIAGRAM REVISI ATURAN NEM | |
| Created By: | Dharmilly | Drawing No: |
| Approved By: | | 4182 0100 01 01 20 10 01 00000001 |
| Checked By: | | |
| Date: | 01/10/2018 | Page: 0 / 0 |



**PT. PLN (PERSERO)
JASA ENGINEERING**

| | |
|--------------|---|
| Project Name | PERUBAHAN SUGUT IYAH PT. PLN (PERSERO) DIS TRIGUNA JATIM - GRAMPEL IIA 107 |
| Task | SINGLE LINE DIAGRAM PENYULANG VEHT 11 |
| Designed By | Dr. Ing. I. B. |
| Checked By | |
| Approved By | |
| Created By | |
| Date | SEPTEMBER 2000 (Rev. 01) |





| | |
|--|---|
| | |
| PT. PLN (PERSERO) JASA ENJINIRING | |
| Project Name: | PERUPAHAN SUBSTANSI WPM |
| | PT. PLN (PERSERO) DISTRIK PUSAT TIM - CABANG MALANG |
| Client: | SISTEM LINE PROGRAM PERUMAHAN RUMAH |
| Location: | Malang |
| Revision: | 1-01 |
| Scale: | 1:1000 |
| Date: | 10/01/2020 |
| Sheet: | 01 |


```

unit uMenu;

interface

uses
  Windows, Messages, Sysutils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, ComCtrls, StdCtrls, ExtCtrls;

type
  TfrmMenu = class(TForm)
    Panel1: TPanel;
    btnNew: TButton;
    btnOpen: TButton;
    btnExit: TButton;
    StatusBar1: TStatusBar;
    Panel2: TPanel;
    OpenDialog1: TOpenDialog;
    procedure btnExitClick(Sender: TObject);
    procedure btnNewClick(Sender: TObject);
    procedure btnOpenClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmMenu: TfrmMenu;

implementation

uses uInputLFChild,uComplex, uUtils, uLoadflow;

{$R *.dfm}

procedure TfrmMenu.btnExitClick(Sender: TObject);
begin
  Application.Terminate;
end;

procedure TfrmMenu.btnNewClick(Sender: TObject);
begin
  try
    if frmInput=nil then
      begin
        frmInput:=TfrmInputLFChild.Create(Application);
        end;
    frmInput.Caption:='Input Data';
    frmInput.btnNext.Caption:='&Save';
    frmInput.ShowModal;
  finally
    frmInput.Free;
  end;
end;

procedure TfrmMenu.btnOpenClick(Sender: TObject);
var NamaFile,Nama:string;
    output:TextFile;
    i,j,Typ,dari,ke,Nbus,Nsa1,Param,Ngen,NCable:integer;
    Cap,absV,sudV,Pg,Qg,PL,QL,CapSa1,Pmin,Pmax,Harga,Length:double;
    R,X,Lc,Tr,Tu,Su,vKonst,pKonst,Pbase,vbase:double;
begin
  try
    if OpenDialog1.Execute then
      begin

```

```

NamaFile:=OpenDialog1.FileName;
AssignFile(output,NamaFile);
Reset(output);
Readln(output,Nbus);
Readln(output,Nsal);
Readln(output,Vbase);
Readln(output,VKonst);
Readln(output,Pbase);
Readln(output,PKonst);
Readln(output,param);
gParamLF.Vbase:=Vbase;
gParamLF.VKonst:=VKonst;
gParamLF.Pbase:=Pbase;
gParamLF.PKonst:=PKonst;
if Param=1 then
begin
gParamLF.ParamBranch:=pbPu;
end
else if Param=2 then
begin
gParamLF.ParamBranch:=pbOhm;
end;
gParamLF.MaxIterasi:=15;
gParamLF.Toleransi:=0.0001;
try
frmInput:=TfrmInputLFChild.Create(Application);
frmInput.edtNbus.Text:=IntToStr(Nbus);
frmInput.edtNsal.Text:=IntToStr(Nsal);
frmInput.edtVbase.Text:=FloatToStr(Vbase);
if VKonst=1 then
begin
frmInput.cmbVKonst.Text:='V';
end
else if VKonst=1000 then
begin
frmInput.cmbVKonst.Text:='kv';
end
else if VKonst=1000000 then
begin
frmInput.cmbVKonst.Text:='MV';
end;
frmInput.edtPbase.Text:=FloatToStr(Pbase);
if PKonst=1 then
begin
frmInput.cmbPKonst.Text:='VA';
frmInput.fgBus.Cells[3,0]:='Pg (W)';
frmInput.fgBus.Cells[4,0]:='Qg (VAR)';
frmInput.fgBus.Cells[5,0]:='PL (W)';
frmInput.fgBus.Cells[6,0]:='QL (VAR)';
frmInput.fgBranch.Cells[9,0]:='Kap (VA)';
frmInput.fgBranch.Cells[10,0]:='P (W)';
frmInput.fgBranch.Cells[11,0]:='Q (VAR)';
frmInput.fgBranch.Cells[16,0]:='P (W)';
frmInput.fgBranch.Cells[17,0]:='Q (VAR)';
end
else if PKonst=1000 then
begin
frmInput.cmbPKonst.Text:='kVA';
frmInput.fgBus.Cells[3,0]:='Pg (kW)';
frmInput.fgBus.Cells[4,0]:='Qg (kVAR)';
frmInput.fgBus.Cells[5,0]:='PL (kW)';
frmInput.fgBus.Cells[6,0]:='QL (kVAR)';
frmInput.fgBranch.Cells[9,0]:='Kap (kVA)';
frmInput.fgBranch.Cells[10,0]:='P (kW)';
frmInput.fgBranch.Cells[11,0]:='Q (kVAR)';
frmInput.fgBranch.Cells[16,0]:='P (kW)';

```

```

    frmInput.fgBranch.Cells[17,0]:='Q (kVAR)';
end
else if PKonst=1000000 then
begin
    frmInput.cmbPKonst.Text:='MVA';
    frmInput.fgBus.Cells[3,0]:='Pg (MW)';
    frmInput.fgBus.Cells[4,0]:='Qg (MVAR)';
    frmInput.fgBus.Cells[5,0]:='PL (MW)';
    frmInput.fgBus.Cells[6,0]:='QL (MVAR)';
    frmInput.fgBranch.Cells[9,0]:='kap (MVA)';
    frmInput.fgBranch.Cells[10,0]:='P (MW)';
    frmInput.fgBranch.Cells[11,0]:='Q (MVAR)';
    frmInput.fgBranch.Cells[16,0]:='P (MW)';
    frmInput.fgBranch.Cells[17,0]:='Q (MVAR)';
end;
if param=1 then
begin
    frmInput.cmbParam.Text:='pu';
    frmInput.fgBranch.Cells[3,0]:='R (pu)';
    frmInput.fgBranch.Cells[4,0]:='X (pu)';
    frmInput.fgBranch.Cells[5,0]:='Lc (pu)';
    frmInput.fgBus.Cells[7,0]:='Cap (pu)';
end
else if param=2 then
begin
    frmInput.cmbParam.Text:='ohm';
    frmInput.fgBranch.Cells[3,0]:='R (ohm)';
    frmInput.fgBranch.Cells[4,0]:='X (ohm)';
    frmInput.fgBranch.Cells[5,0]:='Lc (ohm)';
    frmInput.fgBus.Cells[7,0]:='Cap (ohm)';
end;
SetLength(gBus,Nbus);
for i:=0 to Nbus-1 do
begin
    ReadLn(output,absv,sudv,Pg,Qg,PL,QL,Cap,Typ);
    gBus[i].absv:=absv;
    gBus[i].sudv:=sudv;
    gBus[i].Pgen:=Pg;
    gBus[i].Qgen:=Qg;
    gBus[i].PL:=PL;
    gBus[i].QL:=QL;
    gBus[i].Cap:=Cap;
    gBus[i].typeBus:=Typ;
    frmInput.fgBus.Cells[0,i+1]:=IntToStr(i+1);
    frmInput.fgBus.Cells[1,i+1]:=FloatToStr(absv);
    frmInput.fgBus.Cells[2,i+1]:=FloatToStr(sudv);
    frmInput.fgBus.Cells[3,i+1]:=FloatToStr(Pg);
    frmInput.fgBus.Cells[4,i+1]:=FloatToStr(Qg);
    frmInput.fgBus.Cells[5,i+1]:=FloatToStr(PL);
    frmInput.fgBus.Cells[6,i+1]:=FloatToStr(QL);
    frmInput.fgBus.Cells[7,i+1]:=FloatToStr(Cap);
    frmInput.fgBus.Cells[8,i+1]:=IntToStr(typ);
end;
SetLength(gBranch,Nsa1);
for i:=0 to Nsa1-1 do
begin
    ReadLn(output,dari,ke,R,X,Lc,Tr,Tu,Su,Capsa1);
    gBranch[i].dari:=dari;
    gBranch[i].ke:=ke;
    gBranch[i].R:=R;
    gBranch[i].X:=X;
    gBranch[i].Lc:=Lc;
    gBranch[i].Tr:=Tr;
    gBranch[i].Tu:=Tu;
    gBranch[i].Su:=Su;
    gBranch[i].kapsa1:=Capsa1;
end;

```

```

frmInput.fgBranch.Cells[0, i+1] := IntToStr(i+1);
frmInput.fgBranch.Cells[1, i+1] := IntToStr(dari);
frmInput.fgBranch.Cells[2, i+1] := IntToStr(ke);
frmInput.fgBranch.Cells[3, i+1] := FloatToStr(R);
frmInput.fgBranch.Cells[4, i+1] := FloatToStr(X);
frmInput.fgBranch.Cells[5, i+1] := FloatToStr(Lc);
frmInput.fgBranch.Cells[6, i+1] := FloatToStr(Tr);
frmInput.fgBranch.Cells[7, i+1] := FloatToStr(Tu);
frmInput.fgBranch.Cells[8, i+1] := FloatToStr(Su);
frmInput.fgBranch.Cells[9, i+1] := FloatToStr(CapSal);
end;
Readln(output, Ngen);
if Ngen <> 0 then
begin
frmInput.fgGen.RowCount := Ngen+1;
SetLength(gGenLF, Ngen);
for i := 0 to Ngen-1 do
begin
Readln(output, dari, R, X, Lc, Tr, Tu, Su, CapSal, Pmin, Pmax);
gGenLF[i].bus := dari;
gGenLF[i].Qmin := R;
gGenLF[i].Qmax := X;
gGenLF[i].a2 := Lc;
gGenLF[i].a1 := Tr;
gGenLF[i].a0 := Tu;
gGenLF[i].FixCost := Su;
gGenLF[i].VarCost := CapSal;
gGenLF[i].Pmin := Pmin;
gGenLF[i].Pmax := Pmax;
frmInput.fgGen.Cells[0, i+1] := IntToStr(i+1);
frmInput.fgGen.Cells[1, i+1] := IntToStr(gGenLF[i].bus);
frmInput.fgGen.Cells[2, i+1] := RealToStr(gGenLF[i].Qmin, 2);
frmInput.fgGen.Cells[3, i+1] := RealToStr(gGenLF[i].Qmax, 2);
frmInput.fgGen.Cells[4, i+1] := RealToStr(gGenLF[i].a2, 5);
frmInput.fgGen.Cells[5, i+1] := RealToStr(gGenLF[i].a1, 5);
frmInput.fgGen.Cells[6, i+1] := RealToStr(gGenLF[i].a0, 5);

frmInput.fgGen.Cells[7, i+1] := RealToStr(gGenLF[i].FixCost, 2);

frmInput.fgGen.Cells[8, i+1] := RealToStr(gGenLF[i].VarCost, 2);
frmInput.fgGen.Cells[9, i+1] := RealToStr(gGenLF[i].Pmin, 2);

frmInput.fgGen.Cells[10, i+1] := RealToStr(gGenLF[i].Pmax, 2);
end;
end
else
begin
frmInput.fgGen.RowCount := 2;
end;
CloseFile(output);
frmInput.Caption := 'Tampilan Data';
frmInput.btnNext.Caption := '&Next';
frmInput.ShowModal;
finally
frmInput.Free;
end;
end;
except
MessageDlg('File Corrupt atau Error
Program!', mtWarning, [mbOK], 0);
end;
end;
end.

```

```

unit uInputLFchild;

interface

uses uInputLF,Sysutils,uHasil,Forms;

type
  TfrmInputLFchild=class(TfrmInputLF)
  protected
    procedure ShowHasil;override;
  end;

var frmInput:TfrmInputLFchild;

implementation

uses uLoadflow;

procedure TfrmInputLFchild.ShowHasil;
begin
  try
    if frmHasil=nil then
      begin
        frmHasil:=TfrmHasil.Create(Application);
      end;
    frmHasil.fgBus.Cells[0,0]:='Bus';
    frmHasil.fgBus.Cells[1,0]:='absV (pu)';
    frmHasil.fgBus.Cells[2,0]:='sudV (deg)';
    frmHasil.fgBus.Cells[7,0]:='SupS (pu)';
    frmHasil.fgBus.Cells[8,0]:='Type Bus';
    frmHasil.fgBranch.Cells[0,0]:='No';
    frmHasil.fgBranch.Cells[1,0]:='Dari';
    frmHasil.fgBranch.Cells[2,0]:='ke';
    frmHasil.fgBranch.Cells[5,0]:='Arus re (A)';
    frmHasil.fgBranch.Cells[6,0]:='Arus im (A)';
    frmHasil.fgBranch.Cells[7,0]:='Dari';
    frmHasil.fgBranch.Cells[8,0]:='ke';
    frmHasil.fgBranch.Cells[11,0]:='Arus re (A)';
    frmHasil.fgBranch.Cells[12,0]:='Arus im (A)';
    frmHasil.fgBus2.Cells[0,0]:='Bus';
    frmHasil.fgBus2.Cells[1,0]:='absV (pu)';
    frmHasil.fgBus2.Cells[2,0]:='sudV (deg)';
    frmHasil.fgBus2.Cells[7,0]:='SupS (pu)';
    frmHasil.fgBus2.Cells[8,0]:='Type Bus';
    frmHasil.fgBranch2.Cells[0,0]:='No';
    frmHasil.fgBranch2.Cells[1,0]:='Dari';
    frmHasil.fgBranch2.Cells[2,0]:='ke';
    frmHasil.fgBranch2.Cells[5,0]:='Arus re (A)';
    frmHasil.fgBranch2.Cells[6,0]:='Arus im (A)';
    frmHasil.fgBranch2.Cells[7,0]:='Dari';
    frmHasil.fgBranch2.Cells[8,0]:='ke';
    frmHasil.fgBranch2.Cells[11,0]:='Arus re (A)';
    frmHasil.fgBranch2.Cells[12,0]:='Arus im (A)';
    if gParamLF.PKonst=1 then
      begin
        frmHasil.fgBus.Cells[3,0]:='Pg (W)';
        frmHasil.fgBus.Cells[4,0]:='Qg (VAR)';
        frmHasil.fgBus.Cells[5,0]:='PL (W)';
        frmHasil.fgBus.Cells[6,0]:='QL (VAR)';
        frmHasil.fgBranch.Cells[3,0]:='P (W)';
        frmHasil.fgBranch.Cells[4,0]:='Q (VAR)';
        frmHasil.fgBranch.Cells[9,0]:='P (W)';
        frmHasil.fgBranch.Cells[10,0]:='Q (VAR)';
        frmHasil.lblGen.Caption:='VA';
        frmHasil.lblLoad.Caption:='VA';
        frmHasil.lblLoss.Caption:='VA';
      end;
  end;
end;

```

```

frmHasi1.fgBus2.Cells[3,0]:='Pg (W)';
frmHasi1.fgBus2.Cells[4,0]:='Qg (VAR)';
frmHasi1.fgBus2.Cells[5,0]:='PL (W)';
frmHasi1.fgBus2.Cells[6,0]:='QL (VAR)';
frmHasi1.fgBranch2.Cells[3,0]:='P (W)';
frmHasi1.fgBranch2.Cells[4,0]:='Q (VAR)';
frmHasi1.fgBranch2.Cells[9,0]:='P (W)';
frmHasi1.fgBranch2.Cells[10,0]:='Q (VAR)';
frmHasi1.lblGen2.Caption:='VA';
frmHasi1.lblLoad2.Caption:='VA';
frmHasi1.lblLoss2.Caption:='VA';
end
else if gParamLF.PKonst=1000 then
begin
frmHasi1.fgBus.Cells[3,0]:='Pg (kW)';
frmHasi1.fgBus.Cells[4,0]:='Qg (kVAR)';
frmHasi1.fgBus.Cells[5,0]:='PL (kW)';
frmHasi1.fgBus.Cells[6,0]:='QL (kVAR)';
frmHasi1.fgBranch.Cells[3,0]:='P (kW)';
frmHasi1.fgBranch.Cells[4,0]:='Q (kVAR)';
frmHasi1.fgBranch.Cells[9,0]:='P (kW)';
frmHasi1.fgBranch.Cells[10,0]:='Q (kVAR)';
frmHasi1.lblGen.Caption:='kVA';
frmHasi1.lblLoad.Caption:='kVA';
frmHasi1.lblLoss.Caption:='kVA';
frmHasi1.fgBus2.Cells[3,0]:='Pg (kW)';
frmHasi1.fgBus2.Cells[4,0]:='Qg (kVAR)';
frmHasi1.fgBus2.Cells[5,0]:='PL (kW)';
frmHasi1.fgBus2.Cells[6,0]:='QL (kVAR)';
frmHasi1.fgBranch2.Cells[3,0]:='P (kW)';
frmHasi1.fgBranch2.Cells[4,0]:='Q (kVAR)';
frmHasi1.fgBranch2.Cells[9,0]:='P (kW)';
frmHasi1.fgBranch2.Cells[10,0]:='Q (kVAR)';
frmHasi1.lblGen2.Caption:='kVA';
frmHasi1.lblLoad2.Caption:='kVA';
frmHasi1.lblLoss2.Caption:='kVA';
end
else if gParamLF.PKonst=1000000 then
begin
frmHasi1.fgBus.Cells[3,0]:='Pg (MW)';
frmHasi1.fgBus.Cells[4,0]:='Qg (MVAR)';
frmHasi1.fgBus.Cells[5,0]:='PL (MW)';
frmHasi1.fgBus.Cells[6,0]:='QL (MVAR)';
frmHasi1.fgBranch.Cells[3,0]:='P (MW)';
frmHasi1.fgBranch.Cells[4,0]:='Q (MVAR)';
frmHasi1.fgBranch.Cells[9,0]:='P (MW)';
frmHasi1.fgBranch.Cells[10,0]:='Q (MVAR)';
frmHasi1.lblGen.Caption:='MVA';
frmHasi1.lblLoad.Caption:='MVA';
frmHasi1.lblLoss.Caption:='MVA';
frmHasi1.fgBus2.Cells[3,0]:='Pg (MW)';
frmHasi1.fgBus2.Cells[4,0]:='Qg (MVAR)';
frmHasi1.fgBus2.Cells[5,0]:='PL (MW)';
frmHasi1.fgBus2.Cells[6,0]:='QL (MVAR)';
frmHasi1.fgBranch2.Cells[3,0]:='P (MW)';
frmHasi1.fgBranch2.Cells[4,0]:='Q (MVAR)';
frmHasi1.fgBranch2.Cells[9,0]:='P (MW)';
frmHasi1.fgBranch2.Cells[10,0]:='Q (MVAR)';
frmHasi1.lblGen2.Caption:='MVA';
frmHasi1.lblLoad2.Caption:='MVA';
frmHasi1.lblLoss2.Caption:='MVA';
end;
frmHasi1.fgBus.RowCount:=StrToInt(edtNbus.Text)+1;
frmHasi1.fgBranch.RowCount:=StrToInt(edtNsal.Text)+1;
frmHasi1.fgBus2.RowCount:=StrToInt(edtNbus.Text)+1;
frmHasi1.fgBranch2.RowCount:=StrToInt(edtNsal.Text)+1;

```

```
    frmHasil.ShowModal;  
finally  
    frmHasil.Free;  
end;  
end;  
end.
```

```

unit uFitness;

interface

uses uUtils, uComplex, uLoadflow, uNewtonRaphson, uRecursive,
    Sysutils, Classes, uHasil, uMatrix;

type
    TIndividu=record
        chrom:iArr1;
        fitness:double;
    end;

    TPopulasi=array of TIndividu;

    TFitness=class
    private
        FNbus, FNsa1, FNLoop, FNFault:integer;
        FKp, FCostSwith:double;
        Fvb:TBatas;
        FV, FSg, FSL:CxArr1;
        FCap:dArr1;
        FTypBus:iArr1;
        FZ, FTP:CxArr2;
        FLC, FTr:dArr2;
        FSwith:TSwithArr1;
        FBestIndi:TIndividu;
        function getIndividu(const rIndi:TIndividu):TIndividu;
        function CekStopRecomb(const rData:iArr1):boolean;
        function isNumeric(const rStr:Char):boolean;
        function isSameData(const rStr1,rStr2:string):boolean;
        function isSameDataList(var rList:TStringList;
            const rStr:string):boolean;
        function isContainData(const rNo:integer;
            const rData:iArr1):boolean;
        function CalcVPinalty(const rV:CxArr1):integer;
        function CalcFitness(const rData:iArr1):double;
        function getChrom:iArr1;
    public
        constructor Create(const rNLoop,rNFault:integer;
            const rKp,rCostSwith:double);
        procedure CreateCombinasi(var rList:TStringList;
            var rCount:integer);
        function GetSwith(const rData:iArr1):TSwithArr1;
        function DecodeStrToData(const rStr:string):iArr1;
        function DecodeDataToStr(const rData:iArr1):string;
        property BestChrom:iArr1 read getChrom;
        property NFault:integer read FNFault write FNFault;
    end;

implementation

constructor TFitness.Create(const rNLoop,rNFault:integer;
    const rKp,rCostSwith:double);
begin
    inherited Create;
    FNLoop:=rNLoop;
    FNFault:=rNFault;
    FKp:=rKp;
    FCostSwith:=rCostSwith;
    DecodeCommDataToLFData(gBus, FNbus, FNsa1, FV, FSg, FSL, FCap, FTypBus,

```

```

    gBranch, FZ, FTP, FLC, FTR);
    FSwith:=InitSwith(FZ);
    Fvb.min:=0.95;
    Fvb.max:=1.05;
end;

function TFitness.getIndividu(const rIndi:TIndividu):TIndividu;
var i:integer;
begin
    SetLength(result.chrom, FNLoop);
    for i:=0 to FNLoop-1 do
        begin
            result.chrom[i]:=rIndi.chrom[i];
        end;
    result.fitness:=rIndi.fitness;
end;

function TFitness.GetSwith(const rData:iArr1):TSwithArr1;
var i, NData:integer;
begin
    NData:=high(rData)+1;
    SetLength(result, NData);
    for i:=0 to NData-1 do
        begin
            if (rData[i]<0) or (rData[i]>=FNsa1) then
                begin
                    raise Exception.Create('Melebihi Indeks Matrik!');
                end;
            result[i].dari:=FSwith[rData[i]].dari;
            result[i].ke:=FSwith[rData[i]].ke;
        end;
    end;
end;

function TFitness.CekStopRecomb(const rData:iArr1):boolean;
var i:integer;
begin
    result:=true;
    for i:=0 to FNLoop-1 do
        begin
            if rData[i] <> (FNsa1-1) then
                begin
                    result:=false;
                    break;
                end;
        end;
    end;
end;

function TFitness.isNumeric(const rStr:Char):boolean;
begin
    result:=false;
    if rStr in ['1', '2', '3', '4', '5', '6', '7', '8', '9', '0'] then
        begin
            result:=true;
        end;
end;

function TFitness.DecodeStrToData(const rStr:string):iArr1;
var i, sa:integer;
    st, str:string;
begin
    str:=trim(rStr);

```

```

SetLength(result,4);
sa:=0;
for i:=1 to length(str)+1 do
begin
  if isNumeric(str[i])=true then
  begin
    st:=st+str[i];
  end
  else
  begin
    result[sa]:=StrToInt(st);
    inc(sa);
    st:='';
  end;
end;
end;

function TFitness.DecodeDataToStr(const rData:iArr1):string;
var i,NData:integer;
begin
  NData:=high(rData)+1;
  result:='';
  for i:=0 to NData-1 do
  begin
    result:=result+IntToStr(rData[i])+' ';
  end;
end;

function TFitness.isSameData(const rStr1,rStr2:string):boolean;
var Data1,Data2:iArr1;
    i,j,Count:integer;
begin
  Data1:=DecodeStrToData(rStr1);
  Data2:=DecodeStrToData(rStr2);
  result:=false;
  Count:=0;
  for i:=0 to high(Data1) do
  begin
    for j:=0 to high(Data2) do
    begin
      if Data1[i]=Data2[j] then
      begin
        inc(Count);
        break;
      end;
    end;
  end;
  if Count=high(Data1)+1 then result:=true;
end;

function TFitness.isSameDataList(var rList:TStringList;
const rstr:string):boolean;
var i:integer;
begin
  result:=false;
  for i:=0 to rList.Count-1 do
  begin
    if isSameData(rstr,rList.Strings[i])=true then
    begin
      result:=true;
      break;
    end;
  end;
end;

```

```

    end;
end;
end;

function TFitness.isContainData(const rNo:integer;
    const rData:iArr1):boolean;
var i,NData:integer;
begin
    result:=false;
    NData:=high(rData)+1;
    for i:=0 to NData-1 do
    begin
        if rNo=rData[i] then
        begin
            result:=true;
            break;
        end;
    end;
end;

function TFitness.CalcVPinalty(const rv:CxArr1):integer;
var i:integer;
begin
    result:=0;
    for i:=0 to FNbus-1 do
    begin
        if rv[i].real>1.05 then inc(result);
        if rv[i].real<0.95 then inc(result);
    end;
end;

function TFitness.CalcFitness(const rData:iArr1):double;
var Vpin:integer;
    V,Sg:CxArr1;
    Zb,Alir,Arus:CxArr2;
    sw:TswithArr1;
begin
    sw:=GetSwith(rData);
    Zb:=BuatZBaru(FZ,sw);
    V:=CopyMatrix(FV);
    Sg:=CopyMatrix(FSg);
    NewtonRaphson(gParamLF,V,Sg,FSL,FCap,FTypBus,
    Zb,FTp,Alir,Arus,FLc,FTr);
    Vpin:=CalcVPinalty(FV);
    result:=168*10*1000+1000*FNbus;
    if gParamLF.Iterasi<gParamLF.MaxIterasi then
    begin
        result:=FKp*gParamLF.SumLoss.real*1000+1000*vpin;
    end;
end;

procedure TFitness.CreateCombinasi(var rList:TStringList;
    var rCount:integer);
var i:integer;
    CekRadial,CekCombinasi:boolean;
    str:string;
    sw:TswithArr1;
    Zb:CxArr2;
    Data:iArr1;
    CekIndi:TIndividu;
begin

```

```

SetLength(Data, FNLoop);
for i:=0 to FNLoop-1 do
begin
  Data[i]:=0;
end;
rCount:=0;
rList.Clear;
repeat
  GenCombinasi(FNSal, Data, CekCombinasi);
  if CekCombinasi=true then
  begin
    sw:=GetSwith(Data);
    Zb:=BuatZBaru(FZ, sw);
    CekRadial:=TesJaringan(Zb);
    if (CekCombinasi=true) and (CekRadial=true) and
      (isContainData(FNFault, Data)=true) then
    begin
      str:=DecodeDataToStr(Data);
      if isSameDataList(rList, str)=false then
      begin
        if rCount=0 then
        begin
          FBestIndi.chrom:=CopyMatrix(Data);
          FBestIndi.fitness:=CalcFitness(Data);
        end
        else
        begin
          CekIndi.chrom:=CopyMatrix(Data);
          CekIndi.fitness:=CalcFitness(Data);
          if CekIndi.fitness<FBestIndi.fitness then
          begin
            FBestIndi:=getIndividu(CekIndi);
          end;
        end;
        rList.Add(str);
        inc(rCount);
      end;
    end;
  end;
  frmHasil.pbCount.StepBy(1);
until CekStopRecomb(Data)=true;
end;

function TFitness.getChrom:iArr1;
var i:integer;
begin
  SetLength(result, FNLoop);
  for i:=0 to FNLoop-1 do
  begin
    result[i]:=FBestIndi.chrom[i];
  end;
end;

end.

```

```
unit uAbout;
interface
uses
  Windows, Messages, Sysutils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs;
type
  TfrmAbout = class(TForm)
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  frmAbout: TfrmAbout;
implementation
{$R *.dfm}
end.
```

```
unit uAbout;
interface
uses
  windows, Messages, Sysutils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs;
type
  TfrmAbout = class(TForm)
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  frmAbout: TfrmAbout;
implementation
{$R *.dfm}
end.
```

```
unit uAbout;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs;

type
  TfrmAbout = class(TForm)
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmAbout: TfrmAbout;

implementation

{$R *.dfm}

end.
```

```

unit uRecursive;
interface
uses uUtils, uComplex, uLoadflow;

function InitSwitH(const rZb:CxArr2):TSwitHArr1;
function BuatZBaru(const rZ:CxArr2;
  const rSwitH:TSwitHArr1):CxArr2; overload;
function BuatZBaru(const rZ:CxArr2;
  const rChrom:bArr1):CxArr2; overload;
function TesJaringan(const rZ:CxArr2):boolean;
procedure GenCombinasi(const rNData:integer;
  var rData:iArr1;
  var rCek:boolean);

var gZrec:CxArr2;
    gSw:TSwitHArr1;

implementation

var bus:bArr1;
    Njar:integer;
    findLoop:boolean;

function InitSwitH(const rZb:CxArr2):TSwitHArr1;
var i,j,sa,Nbus:integer;
begin
  Nbus:=high(rZb)+1;
  sa:=0;
  for i:=0 to Nbus-1 do
  begin
    for j:=0 to Nbus-1 do
    begin
      if rZb[i,j].imag<>0 then
      begin
        inc(sa);
      end;
    end;
  end;
  SetLength(result,sa);
  sa:=0;
  for i:=0 to Nbus-1 do
  begin
    for j:=0 to Nbus-1 do
    begin
      if rZb[i,j].imag<>0 then
      begin
        result[sa].dari:=i;
        result[sa].ke:=j;
        inc(sa);
      end;
    end;
  end;
end;

function BuatZBaru(const rZ:CxArr2;
  const rSwitH:TSwitHArr1):CxArr2;
var i,j,NSwitH,Nbus:integer;
begin
  Nbus:=high(rZ)+1;
  SetLength(result,Nbus,Nbus);
  for i:=0 to Nbus-1 do
  begin
    for j:=0 to Nbus-1 do
    begin

```

```

        result[i,j]:=Cplx(rz[i,j]);
    end;
end;
NSwith:=high(rSwith)+1;
for i:=0 to NSwith-1 do
begin
    result[rSwith[i].dari,rSwith[i].ke].real:=0;
    result[rSwith[i].dari,rSwith[i].ke].imag:=0;
end;
end;
end;

```

```

function BuatZBaru(const rz:CxArr2;
    const rChrom:bArr1):CxArr2;
var i,j,Nbus,sa:integer;
begin
    Nbus:=high(rz)+1;
    SetLength(result,Nbus,Nbus);
    sa:=0;
    for i:=0 to Nbus-1 do
    begin
        for j:=0 to Nbus-1 do
        begin
            if rz[i,j].imag<>0 then
            begin
                if rChrom[sa]=true then
                begin
                    result[i,j]:=Cplx(0,0);
                end
                else
                begin
                    result[i,j]:=Cplx(rz[i,j]);
                end;
                inc(sa);
            end;
        end;
    end;
end;
end;

```

```

function BuatMatrikJalur(const rz:CxArr2):iArr2;
var i,j,Nbus:integer;
begin
    Nbus:=high(rz)+1;
    SetLength(result,Nbus,Nbus);
    for i:=0 to Nbus-1 do
    begin
        for j:=0 to Nbus-1 do
        begin
            result[i,j]:=0;
        end;
    end;
    for i:=0 to Nbus-1 do
    begin
        for j:=0 to Nbus-1 do
        begin
            if rz[i,j].imag<>0 then
            begin
                result[i,j]:=1;
                result[j,i]:=1;
            end;
        end;
    end;
end;
end;

```

```

function TesJaringan(const rz:CxArr2):boolean;
var i:integer;
    Jalur:iArr2;

```

```

procedure doRecursive(dari,ke:integer);
var i:integer;
begin
  if findLoop=true then
  begin
    exit;
  end;
  if bus[ke]=false then
  begin
    bus[ke]:=true;
  end
  else
  begin
    findLoop:=true;
    exit;
  end;
  for i:=0 to Njar-1 do
  begin
    if Jalur[ke,i]=1 then
    begin
      if i>dari then
      begin
        doRecursive(ke,i);
        if findLoop=true then
        begin
          exit;
        end;
      end;
    end;
  end;
end;

begin
  Jalur:=BuatMatrikJalur(rz);
  Njar:=high(Jalur)+1;
  SetLength(bus,Njar);
  for i:=0 to Njar-1 do
  begin
    bus[i]:=false;
  end;
  findLoop:=false;
  result:=true;
  doRecursive(0,0);
  if findLoop=true then
  begin
    result:=false;
    exit;
  end;
  for i:=0 to Njar-1 do
  begin
    if bus[i]=false then
    begin
      result:=false;
      exit;
    end;
  end;
end;

function CekCombinasi(const rData:iArr1):boolean;
var i,j:integer;
begin
  result:=true;
  for i:=0 to high(rData)-1 do
  begin
    for j:=i+1 to high(rData) do

```

```

begin
  if rData[i]=rData[j] then
    begin
      result:=false;
      break;
    end;
  end;
end;
end;

procedure GenCombinasi(const rNData:integer;
  var rData:iArr1;
  var rCek:boolean);
var i,NData,Ncom:integer;
begin
  NData:=rNData-1;
  Ncom:=high(rData)+1;
  for i:=Ncom-1 downto 0 do
    begin
      inc(rData[i]);
      if rData[i]<=NData then
        begin
          break;
        end
      else
        begin
          rData[i]:=0;
        end;
      end;
    rCek:=CekCombinasi(rData);
  end;
end.

```

```

unit uHasil;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, ExtCtrls, TeEngine, Series, TeeProcs, Chart,
  Grids,
  ComCtrls;

type
  TfrmHasil = class(TForm)
    TabSheet2: TTabSheet;
    fgBus: TStringGrid;
    TabSheet3: TTabSheet;
    fgBranch: TStringGrid;
    TabSheet4: TTabSheet;
    GroupBox6: TGroupBox;
    Label8: TLabel;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Label12: TLabel;
    lblGen: TLabel;
    lblLoad: TLabel;
    lblLoss: TLabel;
    edtSumGen: TEdit;
    edtSumLoad: TEdit;
    edtSumLoss: TEdit;
    edtIterasi: TEdit;
    edtTime: TEdit;
    TabSheet9: TTabSheet;
    Panel1: TPanel;
    btnClose: TButton;
    btnHitung: TButton;
    TabSheet1: TTabSheet;
    TabSheet5: TTabSheet;
    TabSheet6: TTabSheet;
    TabSheet7: TTabSheet;
    Chart1: TChart;
    Series1: TLineSeries;
    Series2: TLineSeries;
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    lblGen2: TLabel;
    lblLoad2: TLabel;
    lblLoss2: TLabel;
    edtSumGen2: TEdit;
    edtSumLoad2: TEdit;
    edtSumLoss2: TEdit;
    edtIterasi2: TEdit;
    edtTime2: TEdit;
    fgBranch2: TStringGrid;
    fgBus2: TStringGrid;
    Edit1: TEdit;
    PageControl1: TPageControl;
    GroupBox2: TGroupBox;
    Label6: TLabel;
    edtNLoop: TEdit;
    btnUseDefault: TButton;
    GroupBox3: TGroupBox;
  end;

```

```

Label16: TLabel;
Label17: TLabel;
edtSebelum: TEdit;
edtSesudah: TEdit;
btnCreate: TButton;
lblLossP: TLabel;
lblLossN: TLabel;
GroupBox5: TGroupBox;
fgOS: TStringGrid;
pbCount: TProgressBar;
Label21: TLabel;
Label18: TLabel;
lbxComb: TListBox;
cmbFault: TComboBox;
Label22: TLabel;
procedure btnCloseClick(Sender: TObject);
procedure btnHitungClick(Sender: TObject);
procedure btnCreateClick(Sender: TObject);
procedure btnUseDefaultClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure edtNLoopChange(Sender: TObject);
private
  { Private declarations }
  function isDataOnList(const rStr:string;
    const rItems:TStrings):integer;
public
  { Public declarations }
end;

var
  frmHasil: TfrmHasil;

implementation

uses uUtils, uLoadflow, uNewtonRaphson, uComplex, uTopology,
  uRecursive, uMatrix, uFitness;

{$R *.dfm}

var Nsa1:integer;
    CostSebelum, CostSesudah:double;

function TfrmHasil.isDataOnList(const rStr:string;
  const rItems:TStrings):integer;
var i:integer;
begin
  result:=-1;
  for i:=0 to rItems.Count-1 do
  begin
    if rStr=rItems[i] then
    begin
      result:=i;
      break;
    end;
  end;
end;

procedure TfrmHasil.btnCloseClick(Sender: TObject);
begin
  Close;
end;

procedure TfrmHasil.btnHitungClick(Sender: TObject);
var i, ia, ja, Nbus:integer;
    mulai, selesai, selang:TDateTime;
    jam, menit, detik, mdetik:word;

```

```

V, Sg, SL: CxArr1;
Cap: dArr1;
TypBus: iArr1;
Z, Tp, Alir, Arus: CxArr2;
Lc, Tr: dArr2;
sw: TSwithArr1;
begin
DecodeCommDataToLFData(gBus, Nbus, Nsal, V, Sg, SL, Cap, TypBus, gBranch,
Z, Tp, Lc, Tr);
gzrec:=CopyMatrix(Z);
Z[3,10]:=Cmplx(0,0);
Z[8,12]:=Cmplx(0,0);
Z[14,18]:=Cmplx(0,0);
Z[4,15]:=Cmplx(0,0);
btnUseDefault.Enabled:=true;
mulai:=time;
NewtonRaphson(gParamLF, V, Sg, SL, Cap, TypBus, Z, Tp, Alir, Arus, Lc, Tr);
selesai:=time;
selang:=selesai-mulai;
Series1.Clear;
for i:=0 to Nbus-1 do
begin
fgBus.Cells[0,i+1]:=IntToStr(i+1);
fgBus.Cells[1,i+1]:=RealToStr(V[i].real,5);
fgBus.Cells[2,i+1]:=RealToStr(V[i].imag*
ANGLE_OF_DEGREES/ANGLE_OF_RADIAN,5);
fgBus.Cells[3,i+1]:=RealToStr(Sg[i].real,3);
fgBus.Cells[4,i+1]:=RealToStr(Sg[i].imag,3);
fgBus.Cells[5,i+1]:=RealToStr(SL[i].real,3);
fgBus.Cells[6,i+1]:=RealToStr(SL[i].imag,3);
fgBus.Cells[7,i+1]:=RealToStr(Cap[i],3);
fgBus.Cells[8,i+1]:=IntToStr(TypBus[i]);
Series1.Add(V[i].real,IntToStr(i+1));
end;
for i:=0 to high(gBranch) do
begin
ia:=gBranch[i].dari-1;
ja:=gBranch[i].ke-1;
fgBranch.Cells[0,i+1]:=IntToStr(i+1);
fgBranch.Cells[1,i+1]:=IntToStr(ia+1);
fgBranch.Cells[2,i+1]:=IntToStr(ja+1);
fgBranch.Cells[3,i+1]:=RealToStr(Alir[ia,ja].real,3);
fgBranch.Cells[4,i+1]:=RealToStr(Alir[ia,ja].imag,3);
fgBranch.Cells[5,i+1]:=RealToStr(Arus[ia,ja].real,3);
fgBranch.Cells[6,i+1]:=RealToStr(Arus[ia,ja].imag,3);
fgBranch.Cells[7,i+1]:=IntToStr(ja+1);
fgBranch.Cells[8,i+1]:=IntToStr(ia+1);
fgBranch.Cells[9,i+1]:=RealToStr(Alir[ja,ia].real,3);
fgBranch.Cells[10,i+1]:=RealToStr(Alir[ja,ia].imag,3);
fgBranch.Cells[11,i+1]:=RealToStr(Arus[ja,ia].real,3);
fgBranch.Cells[12,i+1]:=RealToStr(Arus[ja,ia].imag,3);
end;
edtSumGen.Text:=toStringJ(gParamLF.SumGen,3);
edtSumLoad.Text:=toStringJ(gParamLF.SumLoad,3);
edtSumLoss.Text:=toStringJ(gParamLF.SumLoss,3);
edtIterasi.Text:=IntToStr(gParamLF.Iterasi);
DecodeTime(selang,jam,menit,detik,mdetik);
edtTime.Text:=IntToStr(jam)+' '+IntToStr(menit)+' ':' '+
IntToStr(detik)+' ':' '+IntToStr(mdetik);
CostSebelum:=gParamLF.SumLoss.real;
sw:=InitSwith(gzrec);
for i:=0 to high(sw) do
begin
cmbFault.Items.Add(IntToStr(sw[i].dari+1)+' - '+
IntToStr(sw[i].ke+1));
end;
end;

```

```

end;

procedure TfrmHasil.btnCreateClick(Sender: TObject);
var i, Nbus, Nsal, ia, ja, Nfault: integer;
    List: TStringList;
    fit: TFitness;
    Count, NLoop: integer;
    sw: TSwWithArr1;
    V, Sg, SL: CXArr1;
    Cap: dArr1;
    TypBus, Data: iArr1;
    Zb, Tp, Alir, Arus: CXArr2;
    Lc, Tr: dArr2;
    mulai, selesai, selang: TDateTime;
    jam, menit, detik, mdetik: word;
begin
    lbxComb.Clear;
    DecodeCommDataToLFData(gBus, Nbus, Nsal, V, Sg, SL, Cap, TypBus, gBranch,
    Zb, Tp, Lc, Tr);
    NLoop:=StrToInt(edtNLoop.Text);
    pbCount.Max:=round(pangkat(Nsal, NLoop)-1);
    Nfault:=isDataOnList(cmbFault.Text, cmbFault.Items);
    fit:=TFitness.Create(NLoop, Nfault, 169, 1);
    List:=TStringList.Create;
    fit.CreateCombinasi(List, Count);
    Data:=fit.BestChrom;
    lbxComb.Items:=List;
    Edit1.Text:=IntToStr(Count);
    sw:=fit.GetSwWith(Data);
    Zb:=BuatZBaru(gZrec, sw);
    List.Free;
    fit.Free;
    mulai:=time;
    NewtonRaphson(gParamLF, V, Sg, SL, Cap, TypBus, Zb, Tp, Alir, Arus, Lc, Tr);
    selesai:=time;
    selang:=selesai-mulai;
    Series2.Clear;
    for i:=0 to Nbus-1 do
    begin
        fgBus2.Cells[0, i+1]:=IntToStr(i+1);
        fgBus2.Cells[1, i+1]:=RealToStr(V[i].real, 5);
        fgBus2.Cells[2, i+1]:=RealToStr(V[i].imag*
            ANGLE_OF_DEGREES/ANGLE_OF_RADIAN, 5);
        fgBus2.Cells[3, i+1]:=RealToStr(Sg[i].real, 3);
        fgBus2.Cells[4, i+1]:=RealToStr(Sg[i].imag, 3);
        fgBus2.Cells[5, i+1]:=RealToStr(SL[i].real, 3);
        fgBus2.Cells[6, i+1]:=RealToStr(SL[i].imag, 3);
        fgBus2.Cells[7, i+1]:=RealToStr(Cap[i], 3);
        fgBus2.Cells[8, i+1]:=IntToStr(TypBus[i]);
        Series2.Add(V[i].real, IntToStr(i+1));
    end;
    for i:=0 to high(gBranch) do
    begin
        ia:=gBranch[i].dari-1;
        ja:=gBranch[i].ke-1;
        fgBranch2.Cells[0, i+1]:=IntToStr(i+1);
        fgBranch2.Cells[1, i+1]:=IntToStr(ia+1);
        fgBranch2.Cells[2, i+1]:=IntToStr(ja+1);
        fgBranch2.Cells[3, i+1]:=RealToStr(Alir[ia, ja].real, 3);
        fgBranch2.Cells[4, i+1]:=RealToStr(Alir[ia, ja].imag, 3);
        fgBranch2.Cells[5, i+1]:=RealToStr(Arus[ia, ja].real, 3);
        fgBranch2.Cells[6, i+1]:=RealToStr(Arus[ia, ja].imag, 3);
        fgBranch2.Cells[7, i+1]:=IntToStr(ja+1);
        fgBranch2.Cells[8, i+1]:=IntToStr(ia+1);
        fgBranch2.Cells[9, i+1]:=RealToStr(Alir[ja, ia].real, 3);
        fgBranch2.Cells[10, i+1]:=RealToStr(Alir[ja, ia].imag, 3);
    end;
end;

```

```

        fgBranch2.Cells[11, i+1] := RealToStr(Arus[ja, ia].real, 3);
        fgBranch2.Cells[12, i+1] := RealToStr(Arus[ja, ia].imag, 3);
    end;
    edtSumGen2.Text := toStringJ(gParamLF.SumGen, 3);
    edtSumLoad2.Text := toStringJ(gParamLF.SumLoad, 3);
    edtSumLoss2.Text := toStringJ(gParamLF.SumLoss, 3);
    edtIterasi2.Text := IntToStr(gParamLF.Iterasi);
    DecodeTime(selang, jam, menit, detik, mdetik);
    edtTime2.Text := IntToStr(jam) + ':' + IntToStr(menit) + ':' +
        IntToStr(detik) + ':' + IntToStr(mdetik);
    CostSesudah := gParamLF.SumLoss, real;
    edtSebelum.Text := FormatFloat('#,##0.000', CostSebelum);
    edtSesudah.Text := FormatFloat('#,##0.000', CostSesudah);
    for i:=0 to 3 do
    begin
        fgOS.Cells[0, i+1] := IntToStr(i+1);
        fgOS.Cells[1, i+1] := IntToStr(sw[i].dari+1);
        fgOS.Cells[2, i+1] := IntToStr(sw[i].ke+1);
    end;
end;

procedure TfrmHasil.btnUseDefaultClick(Sender: TObject);
begin
    edtNLoop.Text := '4';
    fgOS.RowCount := StrToInt(edtNLoop.Text)+1;
    btnCreate.Enabled := true;
end;

procedure TfrmHasil.FormCreate(Sender: TObject);
begin
    fgOS.Cells[0, 0] := 'No';
    fgOS.Cells[1, 0] := 'Asal';
    fgOS.Cells[2, 0] := 'Tujuan';
end;

procedure TfrmHasil.edtNLoopChange(Sender: TObject);
begin
    if edtNLoop.Text = '' then
    begin
        fgOS.RowCount := 2;
    end
    else
    begin
        fgOS.RowCount := StrToInt(edtNLoop.Text)+1;
    end;
end;

end.

```

Skripsi Ini Kuperssembahkan Kepada

ALLAH SWT.

Terima kasih karena telah memberikan rahmat, berkah dan segala kemudahannya kepadaku

Manda Tercinta (Ibu RR. Surya Prakhargiani)

Terima kasih atas doa dan dukungannya serta kesabarannya dalam menunggu Adik menyelesaikan kuliah.

Ayahanda Tercinta (Bapak Faturachman Surya Haminata, SE, MBA)

Terima kasih atas dukungan moral dan materinya, tanpa dukungan papa mungkin Adik ga bisa lulus.

Kakakku Tersayang (Yusrof Azizah Rachmanidita, ST)

Akhirnya Adik bisa lulus & bisa nyusul mba juga ban, makasih atas supportnya, abis ini kita sama-sama berjuang untuk sekolahnya Dinas ya.

Adikku Tersayang (Arief Rachman Andhika)

Makasih atas kepercayaannya yang bikin mba harus cepet lulus, sekarang Dinas bisa sekolah lagi, maaf ga di sekolahnya jadi tertunda, Love u much.

Kakandaku Terkasih (Akhmadi)

Mas. makasih banyak ya atas dukungan moral dan materinya serta kesabarannya dalam menunggu dan mensupport Adik untuk terus semangat menyelesaikan skripsi. Love & Miss u

UCAPAN TERIMA KASIH KEPADA

- © Bapak I Made Wartana, MT. yang telah membimbing saya hingga dapat menyelesaikan skripsi ini
- © Ibu Ida Barita, SH. sebagai dosen wali yang slalu menasehai dan memberi semangat.
- © Mas Jayeng & Mba Eta yang slalu memberikan bantuan dan mempermudah smuanya.
- © Pak Taufik & Bu Imne yang telah menguji dan meluluskan saya.
- © semua dosen-dosen yang tidak dapat disebutkan yang telah membimbing dan membantu dalam perkuliahan
- © keluarga besar Bapak R.C. Soetjipto yang telah memberikan perhatian, support & tumpangan rumah.
- © Eyang Putri RA. Soemamilah yang selalu berdoa untuk kesuksesan semua cucunya.
- © keluarga Bapak Nur Bambang Tri Raharjo yang selalu memberi semangat dan ngerewetin supaya cepat lulus.
- © keluarga Bapak Sapri (Krebet) & keluarga Bapak Abu Sopian (Keparjari) atas dukungan dan doanya.

- © Sahabat-sahabat terbaikku (Soleh, Gandhi, DJ, Tono) makasih atas support, doa dan nasehatnya, moga kalian jadi orang sukses
- © Keluarga Pohanin C334 (Ari, Mas Mico, Ruslan, Rani, Agus, Daeng, De'Aris, Dayat, Ibad, Doni & Arsa) yang telah membantu menyelesaikan skripsi dan supportnya
- © Keluarga GS 313 (Mien, Yunes, Imda, Mani Sakatami, Ama, Ika & Mba ve) Akhirnya aku ikut wisuda juga, makasih doa dan semangatnya
- © Para Mas-masku di Malang (Roni, Sigit, Opik & Joe '96), (Musni, Andik, Endik, Josua, Kiwi & semua yg ditoko IdeoElektrik) makasih bantuan dan antar jemputnya
- © Semua angkatan '99 yang lulus bareng (Giring, Mbah Edi, Rahmat blimbing, Farid Singasari, Aris Bapak, Bima, Samsul, Doni Probolingga, Doni Coki, Amin, Bambang Ngep, Dwi, Dani Brem, Agus, Teguh, Aris Slorok & yang lain yang belum disebut) Selamat ya moga cepat dapat kerja & jadi orang sukses, AMIN.
- © Teman-teman angkatan '99 yg belum lulus, tetap semangat ya.
- © Serta semua yang ga bisa disebutin disini satu persatu, terima kasih sebanyak-banyaknya atas bantuannya selama ini.