

SKRIPSI

PENGEMBANGAN APLIKASI PEMBELAJARAN SIMULASI RANGKAIAN LISTRIK



Disusun Oleh :

RAHMAWIJAYA

05.12.553



**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
FEBRUARI 2010**

LEMBAR PERSETUJUAN
PENGEMBANGAN APLIKASI PEMBELAJARAN
SIMULASI RANGKAIAN LISTRIK
SKRIPSI

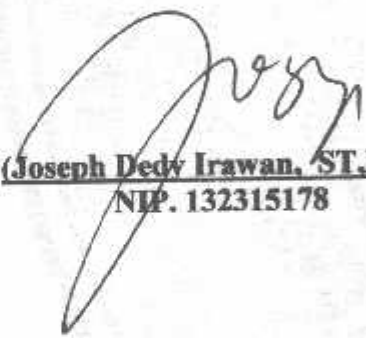
*Disusun dan Diajukan Untuk Melengkapi dan Memenuhi Syarat
Guna Mencapai Gelar Sarjana Teknik*

Disusun oleh :

RAHMAWIJAYA

NIM: 05.12.553

**Diperiksa dan Disetujui,
Dosen Pembimbing**


(Joseph Dedy Irawan, ST,MT)
NIP. 132315178



**Mengetahui,
Ketua Jurusan T. Elektro S-1**


Ir. F. Yudi Limpraptono, MT
NIP.Y.1039500274

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2010

A B S T R A K S I

PENGEMBANGAN APLIKASI PEMBELAJARAN SIMULASI RANGKAIAN LISTRIK

(RAHMAWIJAYA, Nim 05.12.553, Teknik Elektro/T.KOMP. & INFORMATIKA S-1)
(Dosen Pembimbing I : Joseph Dedy Irawan, ST. MT.)

Kata Kunci : aplikasi pembelajaran, rangkaian listrik, kegiatan belajar mengajar.

Perkembangan dunia teknologi komputer saat ini sangat pesat perkembangannya dan sangat luas aplikasinya, tidak hanya di monopoli pada saat awal perkembangannya yaitu oleh pihak militer saja tapi sudah merambah hingga dunia pendidikan. Begitupun juga dalam hal pembelajaran, dengan merekayasa perangkat lunak sebuah materi yang sulit akan lebih mudah dipaparkan dan jauh lebih menarik perhatian daripada menggunakan metode konvensional yaitu menjabarkan materi di papan tulis.

Pada makalah ini, dibuatlah suatu konsep perancangan dan implementasi aplikasi pembelajaran sebagai sarana pendukung dalam hal kegiatan belajar mengajar pada materi rangkaian listrik. Aplikasi pembelajaran ini berbasis Visual Basic 6.0, diharapkan dengan adanya sarana pendukung ini kegiatan proses belajar mengajar agar menjadi lebih optimal dalam hal transfer ilmu dan nilai dari pendidik ke pendengarnya. Aplikasi pembelajaran ini memiliki beberapa kelebihan yaitu pengguna/user dapat dengan mudah menjelaskan materi beserta simulasi dan menghitung besar kuat arus, hambatan dan tegangan dalam rangkaian listrik.

KATA PENGANTAR

Dengan mengucapkan syukur kehadirat Allah SWT yang dengan segala Kasih dan Anugerah – Nya, telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul : **“PENGEMBANGAN APLIKASI PEMBELAJARAN SIMULASI RANGKAIAN LISTRIK“**.

Pembuatan skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata – 1 di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis banyak mendapat bantuan baik moril maupun materil, saran dan dorongan semangat dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE., selaku Rektor ITN Malang.
2. Bapak Ir. Sidik Noertjahyono, MT., selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. F. Yudi Limpraptono, MT., selaku Ketua Jurusan Teknik Elektro S - 1 ITN Malang.
4. Bapak Joseph Dedy Irawan, ST. MT., selaku Dosen Pembimbing I.
5. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan.

Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua. Amiin.

Malang, Februari 2010

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAK.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	ix
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan dan Manfaat	3
1.5. Metodologi Penelitian	3
1.6. Sistematika Penulisan.....	4
BAB II LANDASAN TEORI	5
2.1. Rangkaian Listrik	5
2.1.1 Rangkaian Listrik Terbuka	5
2.1.2. Rangkaian Listrik Tertutup	6
2.1.2.1 Rangkaian Seri	7
2.1.2.2 Rangkaian Paralel	8
2.2. Metode Pengukuran Arus Listrik	10

2.2.1. Hukum Rangkaian	11
2.2.2. Hukum Arus Kirchoff	11
2.2.3. Hukum Tegangan Kirchoff	13
2.3. Microsoft Visual Basic 6.0	13
2.3.1. Pengertian Visual Basic	13
2.3.2. Antar muka Visual Basic 6.0	15
2.3.3. Kelebihan dan Kekurangan Visual Basic 6.0	17
2.3.4. Istilah-istilah Dalam Visual Basic	18
2.4. Macromedia Flash MX	20
2.4.1. Bagian Utama Dalam Flash MX	21
BAB III PERANCANGAN SISTEM.....	23
3.1. Perancangan Modul	24
3.2. Flowchat	25
3.2.1 Flowchat Aplikasi Utama	25
3.2.2 Flowchat Penghitungan Itotal	26
3.2.3 Flowchat Penghitungan VIR Setiap Elemen	26
3.3 Pembahasan	27
3.4 Perancangan Tampilan	27
3.4.1. Form Menu Utama	28
3.4.2 Form Penghitungan	29
3.4.3 Form Petunjuk	30
3.4.4. Form Profil Penulis	31
3.4.5 Form Desain Rangkaian	32

3.4.6. Form Teori Arus	33
3.4.7 Form Teori Hambatan	34
3.4.8. Form Teori Tegangan	35
BAB IV IMPLEMENTASI DAN PENGUJIAN HASIL.....	36
4.1. Implementasi Sistem	36
4.2 Pengujian Hasil	36
4.2.1 Form Splash Screen	37
4.2.2 Halaman Utama	38
4.2.3 Halaman Teori Arus	39
4.2.4. Halaman Teori Tegangan	40
4.2.5. Halaman Teori Hambatan	41
4.2.6. Halaman Desain Rangkaian	42
4.2.7. Halaman Penghitungan	50
4.2.8. Halaman Petunjuk	55
4.2.9. Halaman Profil	55
BAB V PENUTUP.....	56
5.1. Kesimpulan.....	56
5.2. Saran.....	56
DAFTAR PUSTAKA	57
LAMPIRAN – LAMPIRAN	

DAFTAR GAMBAR

BAB II LANDASAN TEORI

Gambar 2.1(A). Contoh Rangkaian Listrik Terbuka	6
Gambar 2.1.(B) Contoh Rangkaian Listrik Terbuka Dengan Bola Lampu	6
Gambar 2.2. Gambar Rangkaian Listrik Tertutup.....	6
Gambar 2.3. Contoh Rangkaian Seri	7
Gambar 2.4. Contoh Rangkaian Paralel	9
Gambar 2.5. Model Rangkaian Untuk Hubungan Tersolder	12
Gambar 2.6. Antarmuka Visual Basic 6.0	15
Gambar 2.7. Komponen Standar Toolbox VB.....	16
Gambar 2.8 Antarmuka Macromedia Flash MX	20
Gambar 2.9 Toolbox	21

BAB III PERANCANGAN DAN PEMBUATAN SISTEM

Gambar 3.1. Perancangan Modul Aplikasi	24
Gambar 3.2. Desain Halaman Utama	28
Gambar 3.3. Desain Halaman Penghitungan	29
Gambar 3.4. Desain Halaman Petunjuk	30
Gambar 3.5. Desain Profil	31
Gambar 3.6. Desain Halaman Rangkaian	32
Gambar 3.7. Desain Halaman Teori Arus	33
Gambar 3.8. Desain Halaman Teori Hambatan	34
Gambar 3.9. Desain Halaman Teori Tegangan	35

BAB IV IMPLEMENTASI DAN PENGUJIAN

Gambar 4.1. Splash Screen	37
Gambar 4.2. Halaman Utama	38
Gambar 4.3. Halaman Teori Arus	39
Gambar 4.4. Halaman Teori Tegangan	40
Gambar 4.5. Halaman Teori Hambatan	41
Gambar 4.6. Halaman Desain Rangkaian	42
Gambar 4.7. Halaman Penghitungan	50
Gambar 4.8. Halaman Petunjuk	55
Gambar 4.9. Halaman Profil	55



BAB I

PENDAHULUAN

1.1 Latar Belakang

Rangkaian listrik adalah suatu bentuk rangkaian yang disusun oleh beberapa komponen (elemen) listrik yang terhubung satu sama lain. Secara umum, rangkaian listrik terbagi atas 2 macam, yaitu rangkaian listrik terbuka dan rangkaian listrik tertutup. Rangkaian listrik terbuka merupakan suatu bentuk rangkaian listrik dimana arus tidak dapat mengalir dalam rangkaian karena ada bagian dalam rangkaian yang tidak terhubung atau dihubungkan dengan komponen pemutus arus seperti saklar. Sedangkan rangkaian listrik tertutup merupakan rangkaian listrik yang terdapat aliran arus di dalamnya. Besar kuat arus dan tegangan dari masing – masing elemen dalam rangkaian listrik tertutup dapat dihitung dengan cara mencari total hambatan (hambatan pengganti) dari elemen – elemen dalam rangkaian. Setelah didapat besar hambatan total rangkaian, maka dapat dicari kuat arus di masing-masing elemen. Setelah mendapatkan kuat arus di masing-masing elemen, kita dapat menghitung besar tegangan dengan mengalikan kuat arus dan besar hambatan.

Penulis memilih topik perangkat lunak pembelajaran rangkaian listrik tertutup ini karena penulis ingin merancang suatu perangkat lunak yang mampu melakukan penghitungan kuat arus, hambatan dan tegangan dalam suatu rangkaian listrik tertutup secara otomatis dan memuat teori tentang arus, hambatan dan tegangan pada rangkaian listrik tertutup dengan simulasinya.

Berdasarkan uraian di atas, penulis bermaksud untuk merancang suatu perangkat lunak aplikasi pembelajaran yang mampu untuk memuat teori tentang arus, hambatan dan tegangan dengan masing-masing simulasi dan menghitung besar

kuat arus elemen, hambatan pengganti dan tegangan elemen dalam rangkaian listrik tertutup.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang ada, maka yang menjadi permasalahan adalah :

1. Merancang sebuah form untuk menyusun rangkaian listrik.
2. Membuat suatu perangkat lunak yang mampu menerangkan dan menghitung besar kuat arus elemen, hambatan pengganti dan tegangan elemen serta mensimulasikan arus dalam rangkaian listrik dengan menggunakan bahasa pemrograman *Visual Basic 6.0 dan Macromedia Flash MX 2004*.

1.3 Batasan Masalah

Untuk mempermudah dalam mencapai tujuan dari penulisan skripsi, terdapat pembatasan masalah sebagai berikut :

1. *Input* dibatasi hanya pada *keyboard* dan *mouse* sedangkan *Output* perangkat lunak pada layar monitor.
2. Hanya menghitung R_{total} , I_{total} , dan serta tegangan arus hambatan ($V-I-R$) saja.
3. Rangkaian listrik yang disusun harus berbentuk persegi atau persegi panjang (sederhana).
4. Komponen yang digunakan dalam rangkaian listrik hanya tegangan (catu daya) dan hambatan (*resistor*) yang independen.
5. Jumlah hambatan (*resistor*) dalam rangkaian dibatasi maksimal 10 buah.
6. Rangkaian listrik hanya mempunyai 1 buah sumber tegangan (catu daya) dan hanya memakai sumber tegangan DC.
7. Program akan menerangkan teori tentang rangkaian listrik tertutup.

8. Program akan menunjukkan cara penghitungan hambatan pengganti, kuat arus dan tegangan secara tahap demi tahap dan simulasi pada setiap teorinya.

1.4 Tujuan dan Manfaat

Tujuan dan manfaat dari perancangan perangkat lunak, antara lain :

1. Membantu mempersentasikan dan penghitungan kuat arus, tegangan dan hambatan pengganti serta mensimulasikan arus dalam rangkaian listrik.
2. Sebagai fasilitas pendukung dalam proses belajar – mengajar.

1.5 Metodologi Penelitian

Langkah – langkah pembuatan perangkat lunak ini antara lain :

- a. Mencari dan mengumpulkan data – data yang diperlukan untuk pembuatan perangkat lunak baik dari pengetahuan dan pengalaman penulis maupun dari sumber – sumber lainnya seperti *internet* dan buku – buku komputer.
- b. Membaca dan mempelajari buku – buku elektronika yang berhubungan dengan rangkaian listrik.
- c. Mempelajari cara penghitungan kuat arus, tegangan dan hambatan pengganti elemen dalam rangkaian listrik tertutup.
- d. Merancang suatu perangkat lunak yang mampu menerangkan teori dan menghitung besar kuat arus, tegangan dan hambatan pengganti serta mensimulasikan arus dalam rangkaian listrik dengan menggunakan bahasa pemrograman *Visual Basic 6.0* dan *Macromedia Flash MX 2004*.

1.6 Sistematika Penulisan

BAB I PENDAHULUAN

Memberikan gambaran umum tentang isi skripsi yang meliputi: latar belakang, rumusan masalah, tujuan dan manfaat penulisan, pembatasan masalah, metodologi penyelesaian masalah, dan sistematika penulisan.

BAB II LANDASAN TEORI

Berisi rangkaian listrik, metode pengukuran arus listrik, hukum rangkaian, visual basic, macromedia flash mx.

BAB III PERANCANGAN DAN PEMBUATAN SISTEM

Berisi analisa kebutuhan, pembahasan, perancangan tampilan.

BAB IV IMPLEMENTASI DAN PENGUJIAN HASIL

Berisi penjelasan analisa setelah program didajalankan.

BAB V KESIMPULAN DAN SARAN

Berisi kesimpulan dan saran dari pembuatan aplikasi pembelajaran.

DAFTAR PUSTAKA

LAMPIRAN



BAB II

LANDASAN TEORI

2.1 Rangkaian Listrik

Rangkaian listrik adalah suatu bentuk rangkaian yang disusun oleh beberapa komponen (elemen) listrik yang terhubung satu sama lain. Komponen (elemen) listrik tersebut dapat berupa resistor (sebagai hambatan), sumber tegangan (catu daya), lampu, saklar, transistor, IC, dan komponen – komponen listrik lainnya. Perilaku suatu rangkaian dapat diungkapkan sepenuhnya dalam besaran satu dimensi, yang berkaitan dengan kedudukannya sepanjang lintasan yang menyusun rangkaian. Pada sebuah rangkaian listrik, hal – hal yang menjadi pusat perhatian adalah tegangan dan arus di berbagai titik sepanjang rangkaian. Pada rangkaian yang tegangan dan arusnya konstan (tidak berubah dengan waktu), arus dibatasi oleh hambatannya.

Secara umum, rangkaian listrik terbagi atas 2 macam, yaitu

1. Rangkaian listrik terbuka.
2. Rangkaian listrik tertutup.

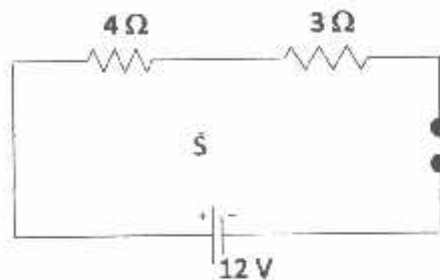
2.1.1 Rangkaian Listrik Terbuka

Rangkaian listrik terbuka merupakan suatu bentuk rangkaian listrik dimana arus tidak dapat mengalir dalam rangkaian karena ada bagian dalam rangkaian yang tidak terhubung atau dihubungkan dengan komponen pemutus arus seperti saklar.

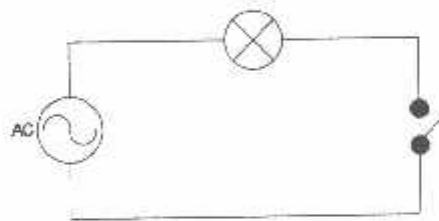
Komponen – komponen listrik dalam rangkaian listrik terbuka tidak dialiri oleh arus walaupun rangkaian listrik tersebut dihubungkan ke sumber tegangan (catu daya). Untuk lebih jelasnya, perhatikan gambar 2.1. Sebuah bola lampu dihubungkan pada sumber tegangan dengan sebuah saklar. Ketika saklar

dimatikan (diputuskan), bola lampu tersebut tidak akan menyala walaupun sumber tegangan aktif. Sedangkan ketika saklar dihidupkan, bola lampu tersebut akan menyala, karena arus dapat mengalir melalui saklar menuju ke bola lampu. Pada keadaan ini, rangkaian menjadi suatu rangkaian listrik tertutup.

Contoh rangkaian listrik terbuka dapat dilihat pada gambar di bawah ini.



Gambar 2.1 (A) Contoh rangkaian listrik terbuka

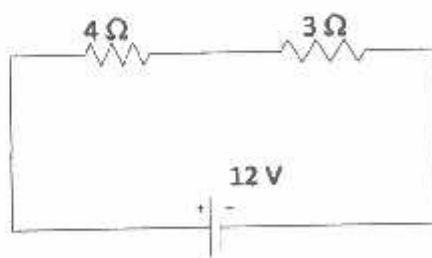


Gambar 2.1 (B) Contoh rangkaian listrik terbuka dengan bola lampu

2.1.2 Rangkaian Listrik Tertutup

Rangkaian listrik tertutup merupakan rangkaian listrik yang dapat dialiri oleh arus listrik. Titik – titik di sepanjang rangkaian listrik tertutup terdapat tegangan dan arus. Besar kuat arus dan tegangan dari masing – masing elemen berbeda – beda, tergantung pada hambatan dari elemen tersebut.

Contoh rangkaian listrik tertutup dapat dilihat pada gambar 2.2 berikut ini.



Gambar 2.2 Contoh rangkaian listrik tertutup

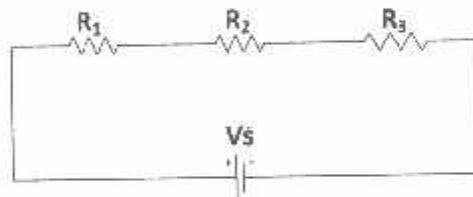
Secara umum, terdapat dua macam cara untuk menghubungkan elemen-elemen listrik satu sama lain dalam rangkaian listrik, yaitu

1. Menghubungkannya secara seri. Rangkaian ini sering disebut sebagai rangkaian seri.
2. Menghubungkannya secara paralel. Rangkaian ini sering disebut sebagai rangkaian paralel.

2.1.2.1 Rangkaian Seri

Pada rangkaian seri, arus yang mengalir melalui setiap komponen sama. Tegangan pada masing – masing hambatan (resistor) bergantung kepada harga resistor tersebut. Jumlah tegangan total dari hambatan (resistor) pada rangkaian sama dengan tegangan catu daya (sumber tegangan).

Contoh rangkaian seri dapat dilihat pada gambar di bawah ini :



Gambar 2.3 Contoh rangkaian seri

Tegangan pada resistor R₁, R₂ dan R₃ berbeda – beda dan bergantung pada besarnya hambatan dari masing – masing resistor. Jumlah tegangan dari semua resistor sama dengan tegangan sumber (V_s), sehingga dapat dirumuskan seperti berikut,

$$V_s = V_1 + V_2 + V_3$$

dimana : V_s = besar tegangan catu daya (sumber tegangan).

V₁ = besar tegangan pada resistor R₁.

V₂ = besar tegangan pada resistor R₂.

$V_3 =$ besar tegangan pada resistor R_3 .

Sedangkan besar kuat arus untuk setiap resistor adalah sama, dan dapat dirumuskan seperti berikut,

$$I = I_1 = I_2 = I_3$$

dimana : $I =$ besarnya kuat arus yang mengalir dalam rangkaian.

$I_1 =$ besar kuat arus pada resistor R_1 .

$I_2 =$ besar kuat arus pada resistor R_2 .

$I_3 =$ besar kuat arus pada resistor R_3 .

Sedangkan besarnya hambatan total dalam rangkaian adalah sama dengan jumlah hambatan dari resistor yang terdapat dalam rangkaian. Rumus perhitungan untuk mencari hambatan total dalam rangkaian dapat dirumuskan seperti berikut,

$$R_t = R_1 + R_2 + R_3$$

dimana : $R_t =$ besar hambatan total dalam rangkaian.

$R_1 =$ besar hambatan dari resistor R_1 .

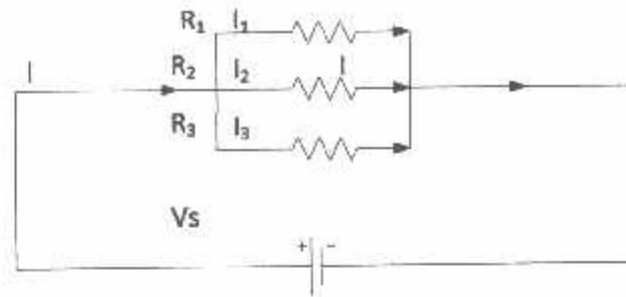
$R_2 =$ besar hambatan dari resistor R_2 .

$R_3 =$ besar hambatan dari resistor R_3 .

2.1.2.2 Rangkaian Paralel

Pada rangkaian paralel, arus yang mengalir melalui setiap komponen bergantung pada besarnya hambatan dari masing – masing resistor. Jumlah total arus dari semua hambatan (resistor) pada rangkaian adalah sama dengan besar kuat arus yang mengalir pada rangkaian. Sedangkan tegangan pada semua hambatan (resistor) pada rangkaian adalah sama besar.

Contoh rangkaian paralel dapat dilihat pada gambar di bawah ini :



Gambar 2.4 Contoh rangkaian paralel

Besar kuat arus pada setiap resistor R_1 , R_2 dan R_3 adalah tidak sama tetapi bergantung pada besarnya hambatan dari masing – masing resistor. Jumlah total arus dari semua resistor sama dengan besar kuat arus yang mengalir dalam rangkaian, sehingga dapat dirumuskan seperti berikut,

$$I = I_1 + I_2 + I_3$$

dimana : I = besar kuat arus yang mengalir dalam rangkaian.

I_1 = besar kuat arus yang mengalir pada resistor R_1 .

I_2 = besar kuat arus yang mengalir pada resistor R_2 .

I_3 = besar kuat arus yang mengalir pada resistor R_3 .

Sedangkan besar tegangan untuk setiap resistor adalah sama besar dan sama dengan tegangan catu daya (sumber tegangan), sehingga dapat dirumuskan seperti berikut,

$$V_s = V_1 = V_2 = V_3$$

dimana : V_s = besar tegangan catu daya (sumber tegangan).

V_1 = besar tegangan pada resistor R_1 .

V_2 = besar tegangan pada resistor R_2 .

V_3 = besar tegangan pada resistor R_3 .

Sedangkan besarnya hambatan total dalam rangkaian adalah sama dengan jumlah dari kebalikan besar hambatan yang terdapat dalam rangkaian. Rumus

perhitungan untuk mencari hambatan total dalam rangkaian dapat dirumuskan seperti berikut,

$$\frac{1}{R_t} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}$$

dimana : R_t = besar hambatan total dalam rangkaian.

R_1 = besar hambatan dari resistor R_1 .

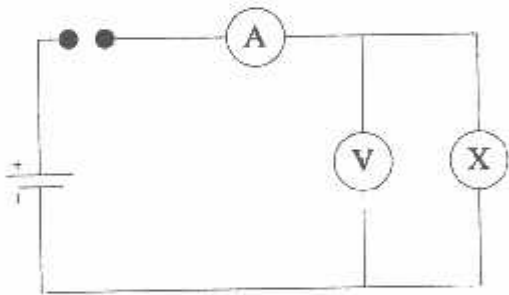
R_2 = besar hambatan dari resistor R_2 .

R_3 = besar hambatan dari resistor R_3 .

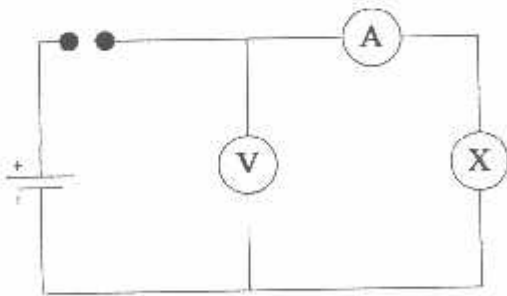
2.2 Metode Pengukuran Arus Listrik

Dalam ilmu elektronika, terdapat dua metode sederhana untuk mengukur arus yang mengalir di dalam elemen listrik. Metode ini tergantung pada susunan voltmeter dan amperemeter.

1. Cara Pertama



2. Cara Kedua



Keterangan :

V = Voltmeter, berfungsi untuk mengukur besar tegangan dari elemen listrik.

A = Amperemeter, berfungsi untuk mengukur besar kuat arus dari elemen listrik.

X = elemen listrik (seperti resistor, lampu pijar, dioda semi konduktor, dan sebagainya).

R_A = Hambatan dalam dari Amperemeter.

R_V = Hambatan dalam dari Voltmeter.

Ketetapan pengukuran elemen listrik (X) sangat bergantung pada hambatan dalam dari alat ukur R_A dan R_V . Untuk cara pertama, agar pengukuran elemen (X) terukur dengan tepat maka besar hambatan R_A harus lebih kecil daripada R_V sedangkan untuk cara kedua, besar hambatan R_A harus lebih besar daripada R_V .

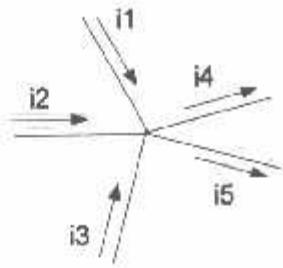
2.2.1 Hukum Rangkaian

Landasan teori rangkaian telah diletakkan kira – kira 150 tahun lalu oleh Gustav Kirchhoff, seorang profesor universitas di Jerman, yang percobaan – percobaan cermatnya telah menghasilkan hukum – hukum yang disebut dengan Hukum Kirchhoff. Suatu simpul adalah titik tempat dua atau lebih cabang bertemu, dan suatu *loop* adalah lintasan tertutup yang dibentuk dengan menghubungkan cabang – cabang.

2.2.2 Hukum Arus Kirchhoff

Untuk mengulangi percobaan Kirchhoff di laboratorium, dapat diatur pengukuran arus pada sejumlah konduktor atau kawat yang disolder bersama

Dalam keadaan apapun, akan terjadi suatu kondisi bahwa jumlah arus masuk pada setiap saat sama dengan jumlah arus keluar dari simpul (titik bersama) tersebut. Suatu model rangkaian digunakan untuk mewakili hubungan yang sebenarnya. Contoh model rangkaian dapat dilihat pada gambar berikut :



Gambar 2.5 Model Rangkaian untuk hubungan tersolder

Anak panah menunjukkan arah acuan untuk arus positif jika arus didefinisikan berdasarkan gerak muatan positif. Besaran “ i_1 ” menyatakan besar kuat arus dan tanda aljabarnya berdasarkan arah acuan tersebut. Jika i_1 bernilai “+ 5 A”, pengaruhnya seperti muatan positif yang bergerak menuju simpul dengan kelajuan 5 C/det. Jika $i_1 = +5A$ mengalir dalam suatu konduktor logam di mana muatan dialirkan dalam bentuk elektron negatif, elektron tersebut sebenarnya bergerak menjauhi simpul tetapi pengaruhnya sama. Jika i_2 bernilai “- 3 A”, muatan positif seperti bergerak menjauhi simpul. Dalam praktek arah aliran arus mudah untuk ditentukan, jika amperemeter “skalanya menaik” arus mengalir memasuki terminal bertanda + dan keluar meter pada terminal -.

Dengan hukum Kirchhoff maka dapat dinyatakan bahwa jumlah aljabar arus masuk suatu simpul pada setiap saat adalah nol. Atau dengan perkataan lain dapat dinyatakan bahwa jumlah arus masuk suatu simpul adalah sama dengan jumlah arus keluar suatu simpul. Pernyataan di atas dapat dirumuskan seperti berikut,

$$\begin{aligned} \sum i &= 0 = i_1 + i_2 + i_3 - i_4 - i_5 \\ i_1 + i_2 + i_3 &= i_4 + i_5 \\ i \text{ masuk} &= i \text{ keluar} \end{aligned}$$

2.2.3 Hukum Tegangan Kirchhoff

Hukum arus pada mulanya dirumuskan berdasar data eksperimen. Hasil yang sama dapat diperoleh dengan asas kekekalan muatan dan definisi arus. Hukum tegangan Kirchhoff juga didasarkan pada eksperimen, namun hasil yang sama dapat diperoleh dengan hukum kekekalan energi dan definisi tegangan.

Untuk mengulang pengamatan Kirchhoff terhadap tegangan, dapat dirangkai suatu rangkaian listrik dan diatur pengukuran tegangan pada berbagai komponen yang membentuk lintasan tertutup.

Berdasarkan hukum tegangan Kirchhoff, jumlah aljabar tegangan dalam suatu simpul setiap saat adalah nol. Atau dengan perkataan lain dapat dinyatakan bahwa jumlah tegangan masuk suatu simpul adalah sama dengan jumlah tegangan keluar suatu simpul. Pernyataan di atas dapat dirumuskan seperti berikut,

$$\begin{aligned}\sum V &= 0 = V_1 + V_2 + V_3 - V_4 - V_5 \\ V_1 + V_2 + V_3 &= V_4 + V_5 \\ V \text{ masuk} &= V \text{ keluar}\end{aligned}$$

2.3 Microsoft Visual Basic 6.0

2.3.1 Pengertian Visual Basic

Visual Basic berasal dari kata visual dan basic. Kata “*visual*” mengacu pada cara yang digunakan untuk membuat Graphical User Interface (GUI). Dengan cara ini kita tidak banyak menuliskan instruksi pemrograman dalam kode – kode baris, tetapi dengan cara melakukan drag dan drop obyek – obyek yang akan digunakan (Wahana Komputer, 2003 : 1).

Kata “ *basic* “ merupakan bagian bahasa *BASIC* (*Beginners All Purpose Symbolic Instruction Code*), yaitu sebuah bahasa pemrograman yang dalam sejarahnya sudah banyak digunakan oleh para programmer untuk menyusun aplikasi (Wahana Komputer, 2003 : 2).

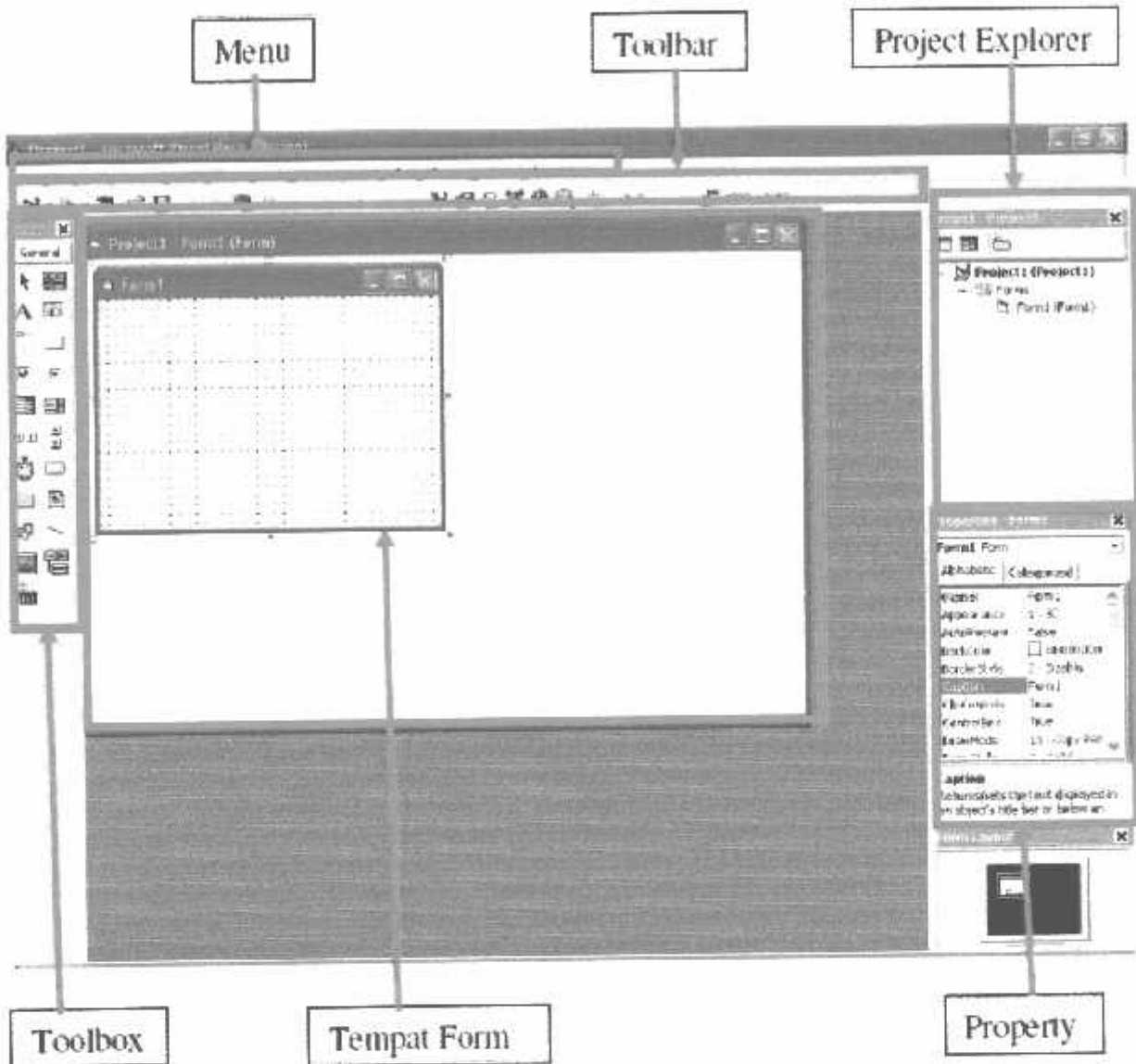
Jadi, visual basic merupakan satu development tool untuk membangun aplikasi dalam lingkungan windows yang dikembangkan dari bahasa BASIC. Dalam pengembangan aplikasi, Visual Basic menggunakan pendekatan visual untuk merancang *user interface* dalam bentuk *form*.

Dalam lingkungan windows, *user interface* sangat memegang peranan penting, karena dalam aplikasi yang dibuat, user senantiasa berinteraksi dengan *user interface* tanpa menyadari bahwa di belakangnya berjalan intruksi-intruksi program yang mendukung tampilan dan proses yang dilakukan.

Pada pemrograman visual, pengembangan aplikasi dimulai dengan membentuk *user interface*, kemudian mengatur property dari objek-objek yang digunakan dalam user interface, kemudian dilakukan penulisan kode program untuk menangani kejadian-kejadian (event). Tahap pengembangan aplikasi demikian dikenal dengan istilah pengembangan aplikasi dengan *Bottom-Up*.

2.3.2 Antarmuka Visual Basic 6.0

Antar muka Visual Basic 6.0, berisi menu, toolbar, toolbox, *form*, project explorer dan property seperti terlihat pada gambar 2.1. berikut:

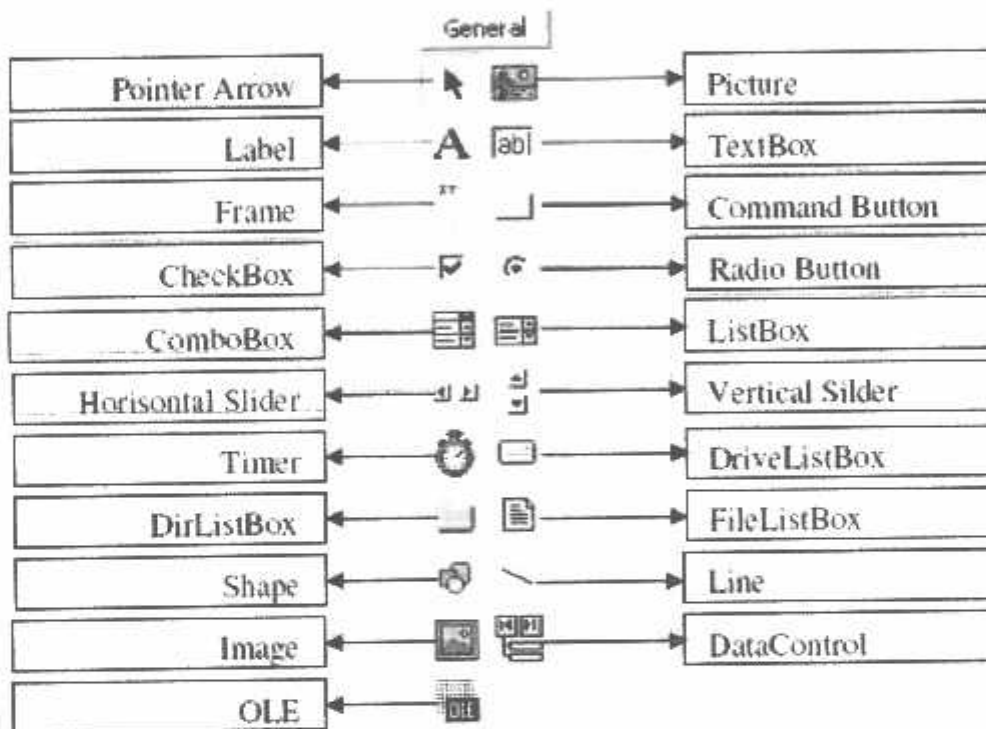


Gambar 2.6 Antarmuka Visual Basic 6.0

Pembuatan program aplikasi menggunakan Visual Basic dilakukan dengan membuat tampilan aplikasi pada *form*, kemudian diberi *script* program di dalam komponen-komponen yang diperlukan. *Form* disusun oleh komponen-komponen yang berada di *Toolbox*, dan setiap komponen yang dipakai harus diatur propertinya lewat jendela *Property*. Menu pada dasarnya adalah operasional

standar di dalam sistem operasi windows, seperti membuat *form* baru, membuat project baru, membuka project dan menyimpan project. Di samping itu terdapat fasilitas-fasilitas pemakaian visual basic pada menu. Untuk lebih jelasnya Visual Basic menyediakan bantuan yang sangat lengkap dan detail dalam MSDN.

Toolbox berisi komponen-komponen yang bisa digunakan oleh suatu project aktif, artinya isi komponen dalam toolbox sangat tergantung pada jenis project yang dibangun. Komponen standar dalam toolbox dapat dilihat pada gambar 2.2 berikut ini :



Gambar 2.7 Komponen Standar dalam *Toolbox* VB

Objek

Sering disebut entity, yakni sesuatu yang bisa dibedakan dengan lainnya. Pada dasarnya seluruh benda di dunia ini bisa dikatakan sebagai objek, contoh : mobil, computer, radio, telepon, dan lain-lain.

Dalam Visual Basic, objek-objek yang dimaksud adalah control. Jenis-jenis control antara lain: *Label*, *Text Box*, *Combo Box*, *List Box*, dan masih banyak lainnya.

Properti

Sering disebut atribut, adalah cirri-ciri yang menggambarkan suatu objek. Misalkan disebut objek mobil jika mempunyai ban, spion, rem dan lainnya.

Event

Suatu kejadian yang menimpa objek. Bagaimana jika mobil didorong, ditabrak, dicat, dan sebagainya.

Metode

Kemampuan yang dimiliki suatu objek. Contohnya jika mobil berbelok, mundur dan maju.

2.3.3 Kelebihan dan Kekurangan Visual Basic

Kelebihan visual basic dibanding dengan bahasa pemrograman yang lain adalah visual basic mampu menambahkan sendiri sebagian kode program secara otomatis kedalam program. Jadi tidak seperti pada bahasa pemrograman yang lain dimana kita harus menuliskan kode program untuk segala sesuatunya. Selain itu, visual basic juga mempunyai banyak sarana untuk membangun program aplikasi berbasis windows dengan cepat dan efisien. Visual basic juga dapat digunakan untuk membuat program aplikasi yang sederhana maupun yang kompleks, database, dan DHTML.

Selain itu, visual basic sudah mengenal beberapa jenis operator matematik yang dapat digunakan untuk pengoperasian aritmatik seperti penjumlahan, pengurangan, pembagian, dan perkalian.

Dalam hal tampilan program, visual basic mampu membuat file yang berekstensi “ *jpg* “ untuk dijadikan sebagai *background*. Hal tersebut akan membuat tampilan program yang dibuat dengan menggunakan visual basic akan menjadi lebih menarik.

Visual basic khususnya visual basic 6.0 memiliki fasilitas “ *Package and Deployment wizard* “ yang akan mempermudah proses pendistribusian program yang sudah jadi. Sehingga aplikasi yang dibuat dengan menggunakan visual basic dapat diubah atau *dicompile* menjadi suatu program yang dapat *diinstall* seperti program – program aplikasi komputer yang lain dan dapat berjalan tanpa adanya program visual basic.

Kekurangan yang terdapat pada visual basic adalah, visual basic hanya dapat bekerja didalam *operating system* berbasis *windows* (*under windows*).

2.3.4 Istilah – Istilah Dalam Visual Basic

a. Form

Form merupakan “ *window* “ atau sebuah lembar kerja yang akan menjadi tampilan program yang merupakan tempat pengguna program berinteraksi dengan program.

b. Project

Project adalah sekumpulan file yang terorganisir dan digunakan untuk membangun sebuah aplikasi. File dapat berupa *form* beserta perintah – perintah yang terdapat didalam *form*.

c. Syntax

Syntax merupakan sekumpulan perintah yang berupa kode – kode dalam bahasa pemrograman.

d. Kontrol intrinsik

Kontrol intrinsik adalah kontrol – kontrol dasar yang digunakan untuk menyusun suatu aplikasi. Didalam visual basic, kontrol intrinsik terkumpul menjadi satu dalam sebuah “ Tool Box “.

e. Statement

Statement dapat diartikan sebagai sebuah pernyataan yang mewakili suatu perintah. Contohnya adalah statement error yang merupakan pernyataan bila terjadi error pada program maka program akan diperlakukan seperti pada ketentuan yang dibuat programmer. Apakah program akan dihentikan dan meminta untuk diperbaiki atau program akan ditutup.

f. Deklarasi

Deklarasi merupakan suatu perintah yang berfungsi sebagai pen jembatan antara variabel yang ditulis programmer dengan visual basic sehingga variabel tersebut dapat dibaca oleh visual basic.

g. Variabel

Variabel merupakan suatu besaran yang nilainya dapat berubah – ubah selama terjadinya proses pada program. Penulisan nama variabel tidak boleh menggunakan spasi.

h. Eksekusi

Eksekusi merupakan proses menjalankan program.

i. Kompilasi

Kompilasi merupakan proses menerjemahkan bahasa pemrograman yang ditulis oleh programmer menjadi bahasa mesin sehingga dapat dibaca oleh “ mesin “. Dalam hal ini kata “ mesin “ mengacu pada visual basic.

j. Windows *form* designer

Merupakan jendela atau sebuah windows yang merupakan tempat mendesain tampilan program.

k. Windows code editor

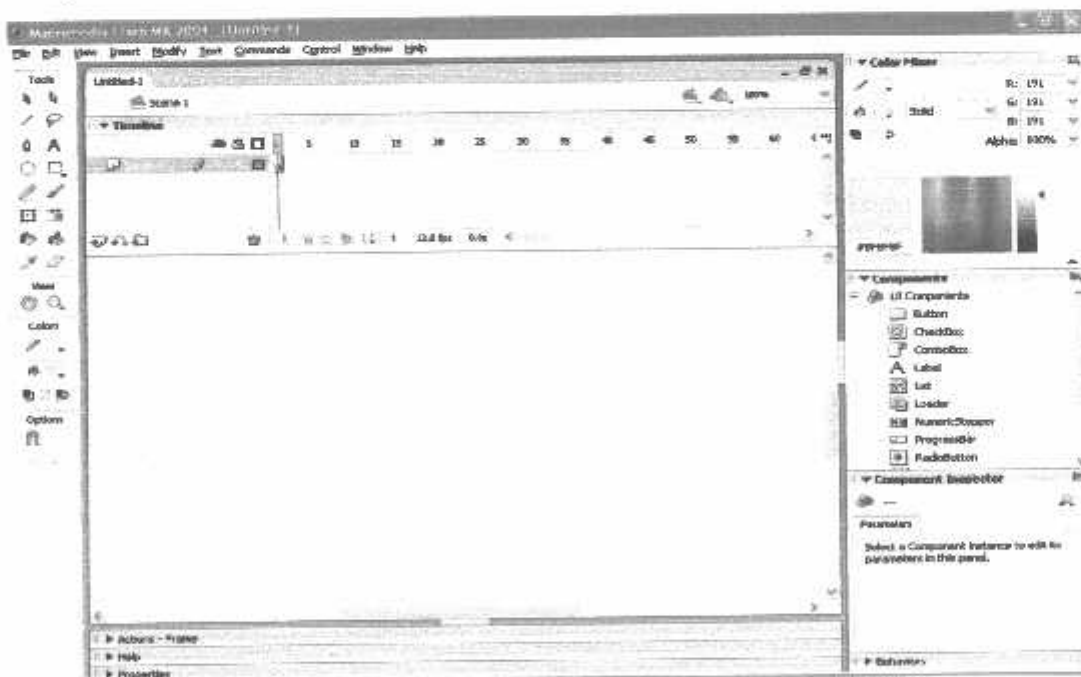
Merupakan jendela atau sebuah windows yang merupakan tempat menuliskan perintah – perintah dalam kode bahasa pemrograman.

l. Properti kontrol intrinsik

Properti kontrol intrinsik dapat diartikan sebagai ciri atau identitas dari kontrol intrinsik. Diantaranya yaitu berisi tentang nama kontrol intrinsik, warna latar belakang, jenis font, dan lain sebagainya.

2.4 Macromedia Flash MX

Macromedia flash adalah sebuah tool yang dapat digunakan untuk membuat berbagai animasi, persentasi, game bahkan perangkat ajar. Selain itu, Flash MX dapat juga digunakan juga sebagai tool untuk mendesain web dan berbagai aplikasi multimedia lainnya.



Gambar 2.8 Antarmuka Macromedia Flash MX

2.4.1 Bagian Utama Dalam Flash MX

1. Toolbox

Toolbox merupakan perangkat utama dalam menggambar dalam Flash, toolbox terdiri dari empat bagian utama, yaitu *Tool*, *Colors*, *View* dan *Options*.

Tools selain terdiri dari perangkat utama menggambar dan untuk menuliskan text, juga terdapat perangkat untuk memilih objek yang akan dimodifikasi. *View* menyediakan perangkat *hand* yang digunakan untuk memindahkan stage dan perintah *zoom* yang digunakan untuk memperbesar dan memperkecil gambar. *Colors* menyediakan perangkat untuk mewarnai dan mengubah bentuk garis, mewarnai isi objek dan texts. Bagian akhir toolbox, yaitu *Options* yang memperlihatkan seting-seting optional untuk tool yang sedang aktif. Jika toolbox tidak terlihat dilayar, Anda bisa menampilkannya dengan menggunakan menu Window dan pilih Tools.



Gambar 2.9 Toolbox

2. Timeline

Timeline merupakan tool utama untuk membuat animasi dan objek-objek interaktif. Timeline memiliki tiga bagian utama, yaitu *Scene*, *Layer*, *Frame*.

- Scene merupakan subbagian dari Timeline.
- Layer merupakan bagian dari Scene. Kita bisa menambah dan mengurangi layer sesuai dengan keperluan kita.
- Frame merupakan bagian dari layer, setiap layer memiliki banyak frame. Frame juga merupakan kecepatan animasi, maksudnya lama dari animasi tersebut berjalan. Pada setiap layer dan frame inilah kita akan membuat animasi.

3. Stage

Stage adalah lembar kerja dimana Anda membuat semua elemen *Movie Flash*.

Jika diinginkan Anda bisa mengubah ukuran dan warna latar belakang dari Stage.

4. Panel

Panel digunakan untuk menentukan atribut-atribut yang akan digunakan oleh tool-tool dalam toolbox untuk membuat atau mengubah objek pada stage.

5. Symbol

Dalam pembuatan animasi, bisa memilah-milahnya menjadi bagian yang kecil untuk memudahkan pembuatannya dan menghindari penggunaan layer dan frame yang banyak sehingga dapat membuat bingung, untuk memilah-milah animasi bisa menggunakan *symbol*. Pada pembuatan symbol terdapat tiga pilihan yaitu :

1. Movie Clip, untuk membuat animasi dalam bentuk *Movie*.
2. Button, digunakan apabila kita membuat sebuah tombol dengan efek-efek animasi tertentu.
3. Graphic, apabila hanya membuat sebuah grafik atau image tanpa animasi.



BAB III

PERENCANGAN DAN PEMBUATAN SISTEM

Aplikasi yang direncanakan ini berguna sebagai perangkat ajar dalam hal materi rangkaian listrik tertutup yang meliputi arus, hambatan dan tegangan. Sehingga memberikan sebuah suasana yang berbeda ketika seorang pengajar/pendidik ketika menerangkan materi yang bersangkutan.

Aplikasi menampilkan menu utama, di mana pengguna/*user* dapat memilih menu yang tersedia. Materi (teori) yang ditampilkan, dapat langsung dipilih dengan setiap halaman materi terdiri dari beberapa halaman lagi. Setiap pilihan teori, di dalamnya tersedia deskripsi mengenai materi subyeknya dan animasinya.

Dalam merancang aplikasi ini terlebih dahulu dilakukan pembuatan desain proses dan desain antarmuka aplikasi. Desain proses berguna untuk mengintegrasikan semua proses yang terjadi dalam aplikasi yang akan dibuat. Sedangkan perancangan antarmuka berfungsi sebagai antarmuka interaksi antara pengguna dengan sistem aplikasi yang dibuat, sehingga pengguna/*user* dapat mengoperasikan aplikasi yang dibuat.

3.1 Perancangan Modul

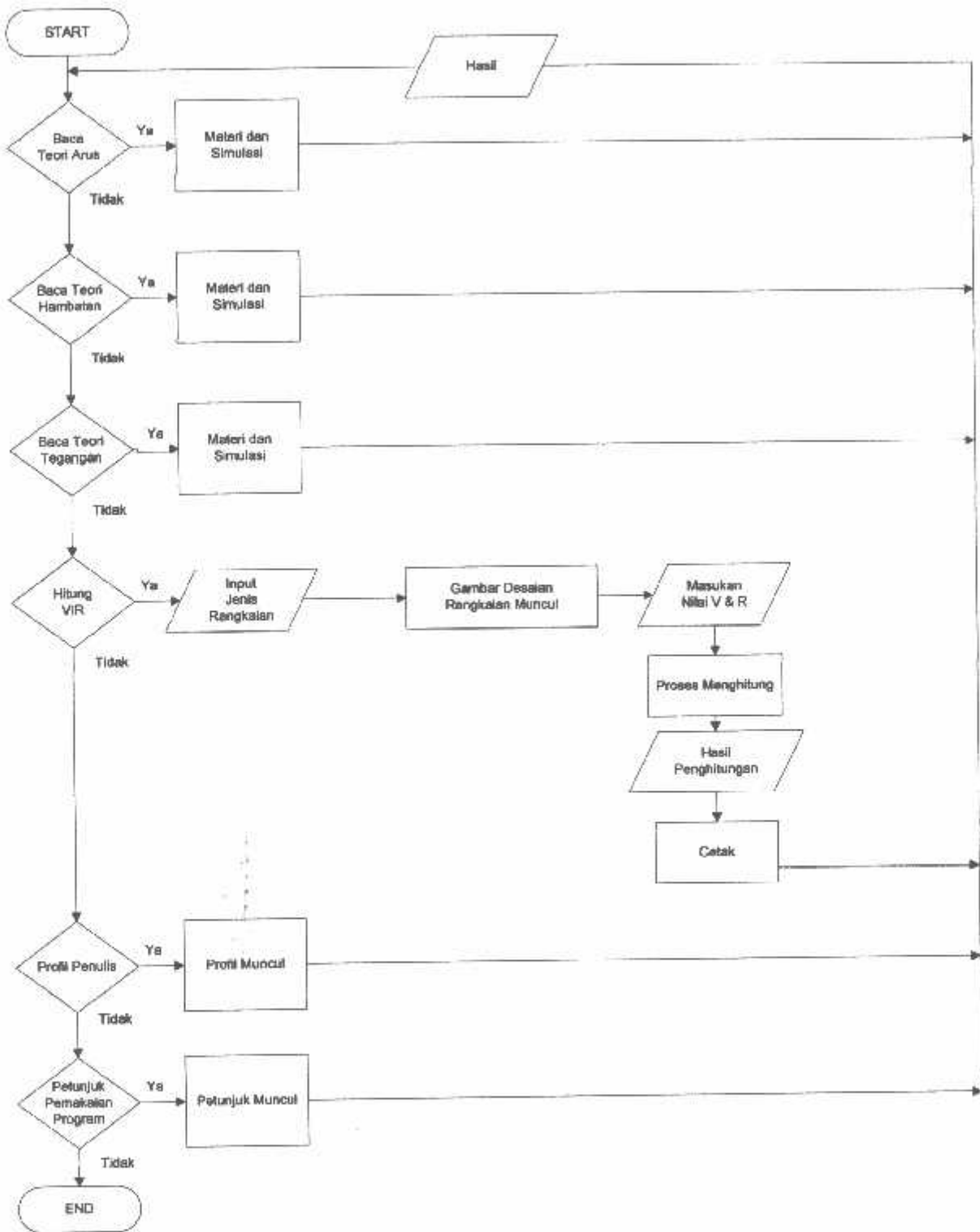
Sistem yang akan dibuat pada perancangan ini adalah sebuah aplikasi yang dapat memaparkan dan menghitung serta men-simulasikan gambar rangkaian sesuai dengan input-an mengenai rangkaian listrik tertutup. Dimana penghitungan tersebut berupa kalkulasi besaran R_{total} dan I_{total} serta tegangan arus hambatan(V-I-R) dalam setiap rangkaian listrik tertutup.

Gambar 3.1 Perancangan Modul Aplikasi

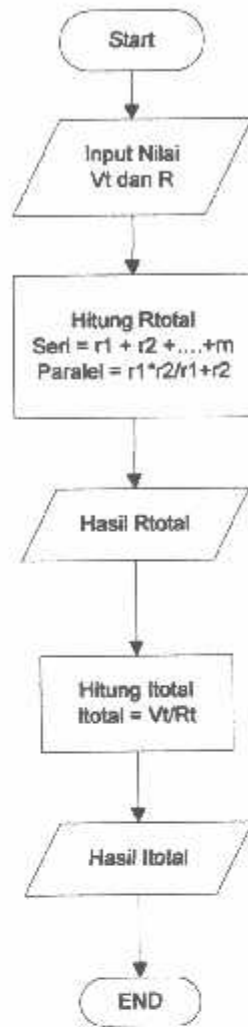


3.2 Flowchat

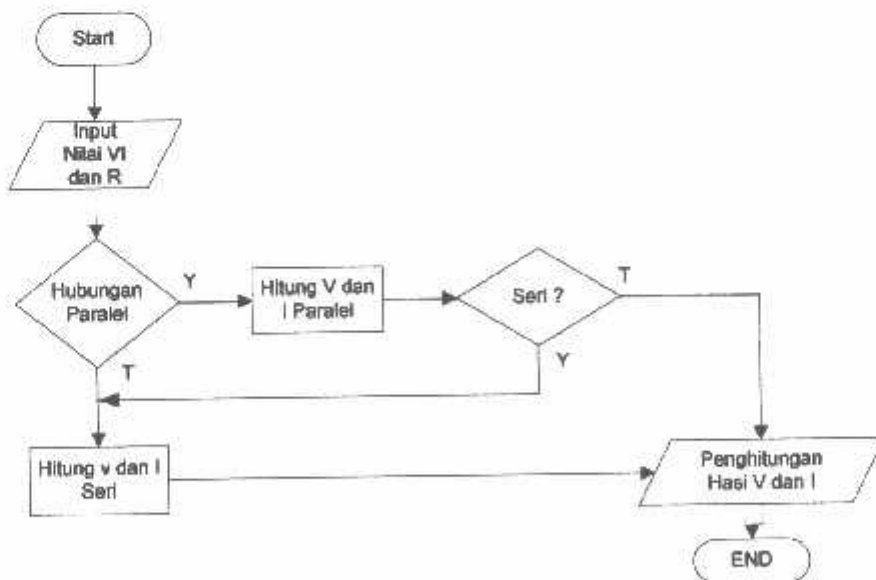
3.2.1 Flowchart Aplikasi Utama



3.2.2 Flowchart Penghitungan Itotal



3.2.3 Flowchart Penghitungan VIR Setiap Elemen



3.3 Pembahasan

Proses kerja dari perangkat lunak penghitungan besar kuat arus, hambatan dan tegangan elemen serta simulasi dalam rangkaian tertutup ini terbagi menjadi 2 bagian besar yaitu :

1. Proses penggambaran rangkaian listrik tertutup.
2. Proses penghitungan besar kuat arus, hambatan dan tegangan setiap elemen dalam rangkaian.

Proses penggambaran rangkaian listrik tertutup ini akan dimulai dengan melakukan pengecekan terhadap *input* struktur rangkaian hambatan. Proses pengecekan ini menggunakan tanda sebagai berikut :

- a. Tanda '→' yang mewakili hubungan seri.
- b. Tanda '⇨' yang mewakili hubungan paralel.

3.4 Perancangan Tampilan

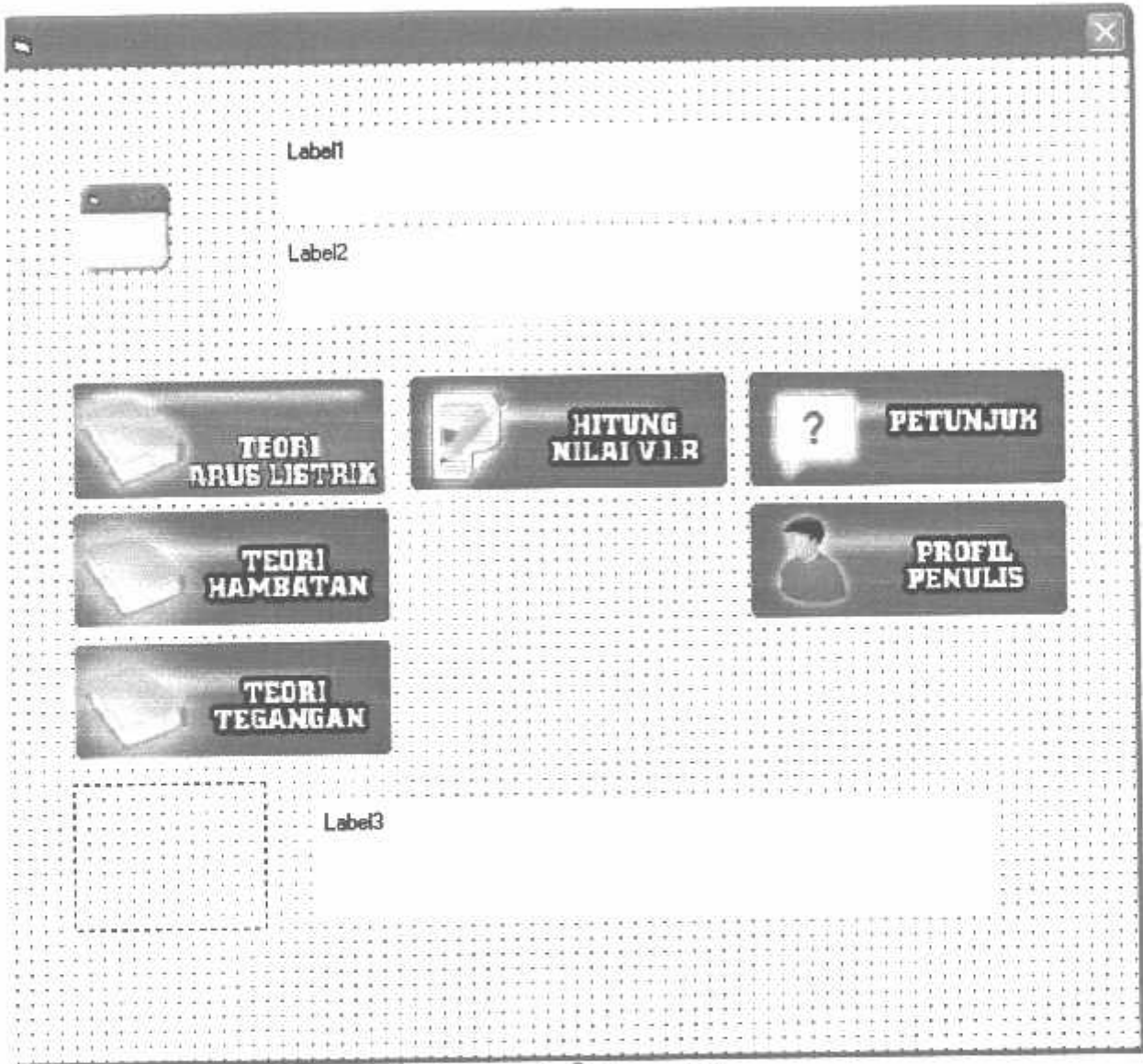
Perangkat lunak penghitungan besar kuat arus, hambatan dan tegangan elemen serta simulasi dalam rangkaian tertutup ini dirancang dengan menggunakan bahasa pemrograman Microsoft Visual Basic 6.0 dan Macromedia Flash MX .

Dalam proses perancangan, penulis membuat beberapa *Form* antara lain :

1. *Form* Menu Utama.
2. *Form* Perhitungan.
3. *Form* Petunjuk.
4. *Form* ProfilPenulis.
5. *Form* Rangkaian.
6. *Form* TeoriArus.

7. *Form* TeoriHambatan.
8. *Form* TeoriTegangan.
9. *Form* Petunjuk
10. *Form* Profil

3.4.1 *Form* Menu Utama

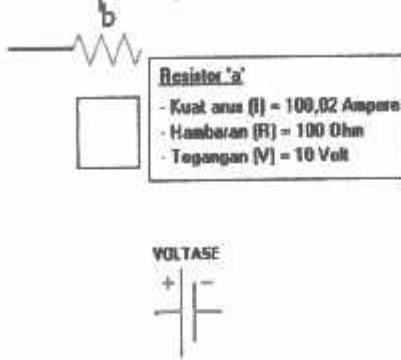


Gambar 3.2 Desain Halaman Utama

Ini adalah desain halaman utama dalam aplikasi ini, pada *Form* ini tersedia beberapa pilihan menu, *image*, *VbSkinner*, dan beberapa buah *Label*.

3.4.2 Form Penghitungan

GAMBAR RANGKAIAN HAMBATAN : 

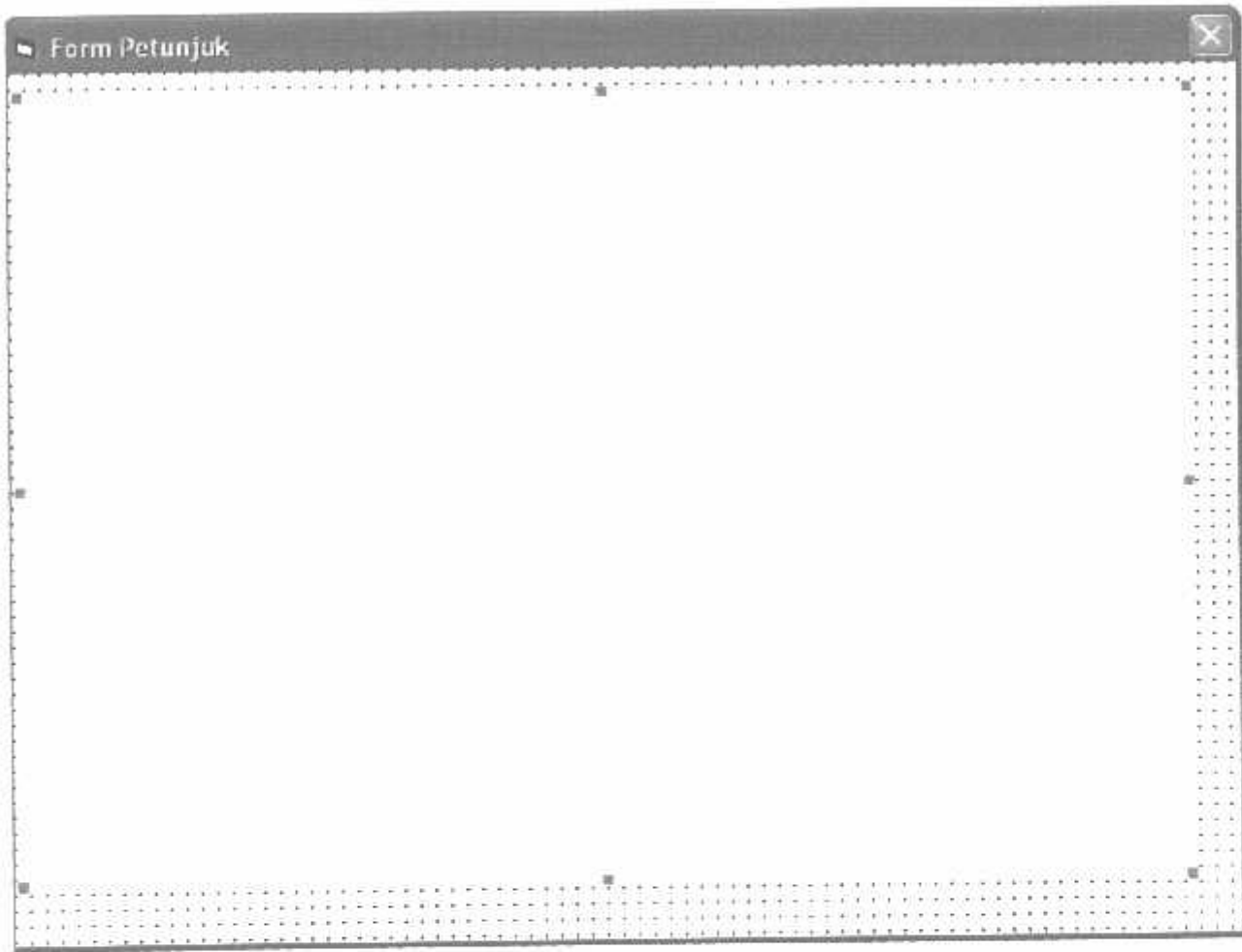


Input tegangan total (V-total) :	0 volt	Perhitungan kuat arus total (I-total) :	Kalkulasi	Cetak	V-I-R untuk setiap elemen :	Kalkulasi	Cetak

Gambar 3.3 Desain Halaman Penghitungan

Form ini berfungsi untuk menginputkan nilai V dan besar resistor. Terdiri atas beberapa buah *label*, *picture box*, *table*, *text box*, *line*, *image* dan *button*.

3.4.3 Form Petunjuk



Gambar 3.4 Desain Halaman Petunjuk

Pada *Form* ini, hanya terdiri dari komponen *rich text box*, yang mana *Form* ini akan menampilkan petunjuk tentang penggunaan pemakaian aplikasi ini.

3.4.4 Form Profil Penulis



The image shows a screenshot of a web form titled "PENGEMBANGAN APLIKASI PEMBELAJARAN SIMULASI RANGKAIAN LISTRIK". The form has a light gray background with a fine grid pattern. At the top, the title is centered in bold black text. Below the title is the logo of Institut Teknologi Nasional Malang, which is a shield-shaped emblem with a central figure and the text "INSTITUT TEKNOLOGI NASIONAL" and "MALANG". Underneath the logo, the author's name "NAMA : RAHMAWIJAYA" and student ID "NIM : 0512553" are listed. The author's affiliation "TEKNIK KOMPUTER & INFORMATIKA" and "TEKNIK JURUSAN ELEKTRO S1" are also present, along with "ITN MALANG" and the year "2010".

**PENGEMBANGAN APLIKASI PEMBELAJARAN SIMULASI
RANGKAIAN LISTRIK**

**INSTITUT TEKNOLOGI NASIONAL
MALANG**

**NAMA : RAHMAWIJAYA
NIM : 0512553**

**TEKNIK KOMPUTER & INFORMATIKA
TEKNIK JURUSAN ELEKTRO S1
ITN MALANG
2010**

Gambar 3.5 Desain profil

Form ini akan menunjukkan nama aplikasi dan profil dari penulis. Terdiri dari banyak label dan image.

3.4.5 Form Desain Rangkaian

Rangkaian Hambatan

Input struktur rangkaian hambatan :

- Simbol - simbol untuk resistor : a, b, c, d, e, f, g t, u, v, w, x, y, z

- Jenis hubungan antar resistor :

→	rangkaian seri
⇒	rangkaian paralel

- Pemakaian tanda kurung

{	Buka kurung
}	Tutup Kurung

Gambar Rangkaian Listrik

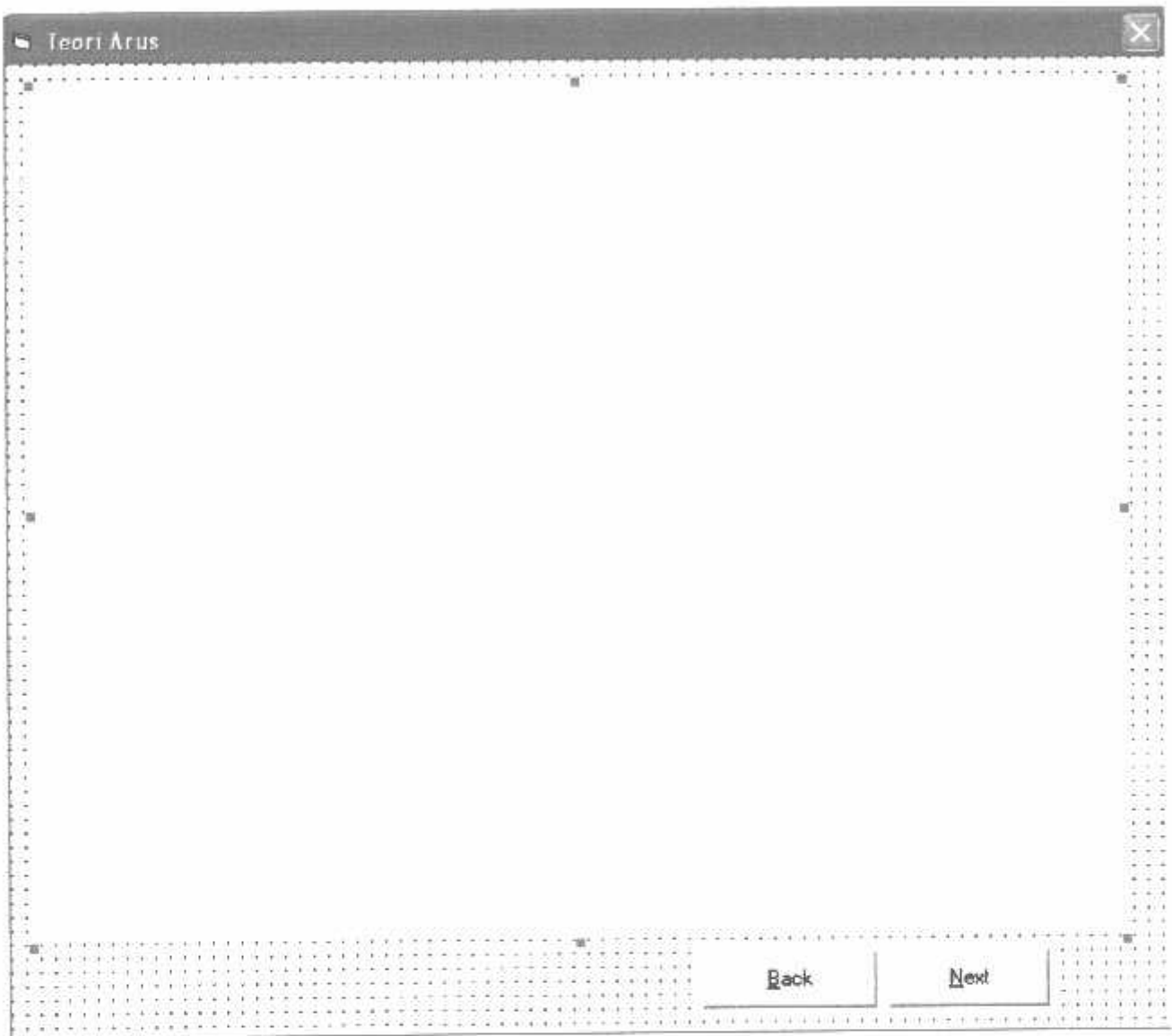
Contoh rangkaian hambatan dari : a

The diagram shows a circuit with a DC power source at the bottom. The circuit branches into two parallel paths. The upper path contains two resistors, labeled 'a' and 'b', connected in series. The lower path contains a single resistor labeled 'c'. Both paths recombine and return to the power source.

Gambar 3.6 Desain Halaman Rangkaian

Form ini berfungsi untuk mendesain halaman rangkaian. Terdiri dari banyak label, image, line, button serta rich text box.

3.4.6 *Form* Teori Arus



Gambar 3.7 Desain Halaman Teori Arus

Pada *Form* ini, akan menampilkan teori tentang arus. *Form* ini terdiri atas satu *rich text box* dan dua buah *button*.

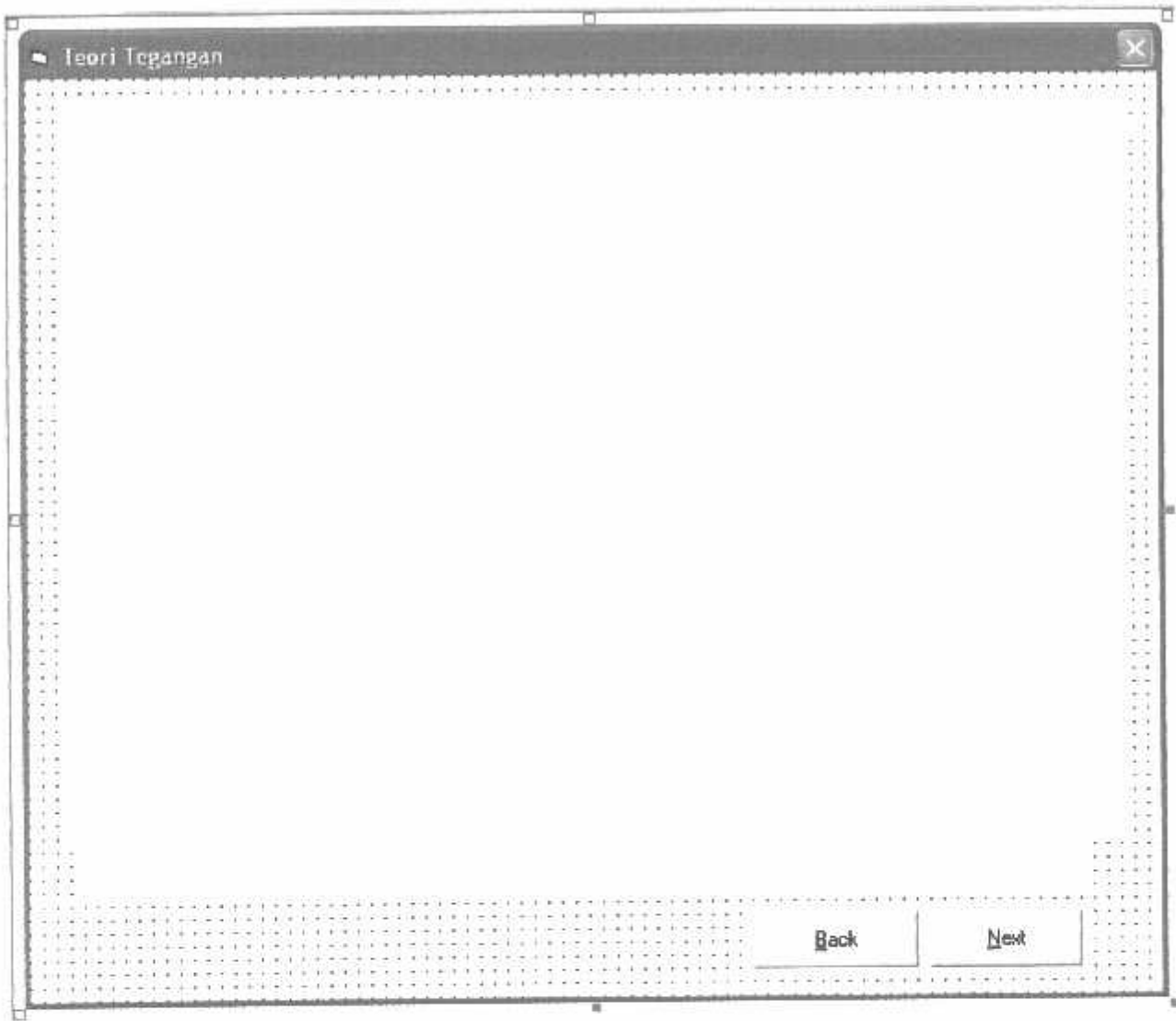
3.4.7 Form Teori Hambatan



Gambar 3.8 Desain Halaman Teori Hambatan

Pada *Form* ini, akan menampilkan teori tentang hambatan. *Form* ini terdiri atas satu rich *text box* dan dua buah *button*.

3.4.8 *Form* Teori Tegangan



Gambar 3.9 Desain Halaman Teori Tegangan

Pada *Form* ini, akan menampilkan teori tentang tegangan. *Form* ini terdiri atas satu *rich text box* dan dua buah *button*.



BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Tahap implementasi pengembangan perangkat lunak merupakan proses pengubahan spesifikasi sistem menjadi sistem yang dapat dijalankan. Tahap ini merupakan lanjutan dari proses perancangan, yaitu proses pemrograman perangkat lunak sesuai dengan spesifikasi dan desain sistem.

Aplikasi pembelajaran ini menggunakan Visual Basic 6.0. Sedangkan untuk simulasinya menggunakan Macromedia Flash MX yang kemudian nantinya *file* Flash yang sudah jadi (ekstensi.swf) akan dimasukkan ke dalam *form* VB.

Ada beberapa tampilan/*form* pada aplikasi ini. Diantaranya adalah tampilan halaman utama, halaman teori, halaman desain rangkaian, halaman penghitungan serta halaman petunjuk dan profil.

4.2. Pengujian Hasil

Pengembangan aplikasi pembelajaran ini merupakan aplikasi untuk perangkat bantu dalam melakukan proses kegiatan proses belajar mengajar mengenai materi rangkaian listrik tertutup. Pengoperasian aplikasi berupa memilih menu-menu yang telah disediakan yaitu berupa menu teori arus, tegangan dan hambatan serta menu penghitungan besar hambatan total dan kuat arus total serta tegangan dan kuat arus pada masing – masing elemen resistor yang terdapat di dalam rangkaian.

4.2.1. *Splash Screen*



Gambar 4.1 Splash Screen

Form ini adalah komponen tambahan dari Visual Basic, ketika program dijalankan, maka *splash screen* akan terlebih dahulu muncul. Komponen tambahan ini dapat di download di www.visual-basic.com.ar hanya saja penulis di sini memakai versi *trial*.

Dengan memakai komponen tambahan ini, penulis dapat dengan mudah medesain halaman sebuah *form* pada VB.

4.2.2. Halaman Utama



Gambar 4.2 Halaman Utama

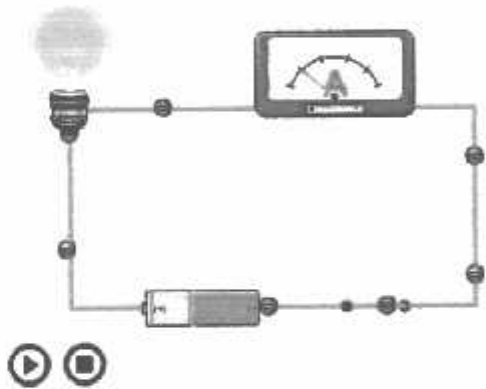
Setelah *splash screen* tadi dilewati, maka halaman utama dari aplikasi ini akan muncul. Di halaman utama ini pengguna dapat dengan langsung memilih menu yang tertera dengan cara meng-*klik* menu yang diinginkan.

4.2.3. Halaman Teori Arus

ARUS

Arus merupakan perubahan kecepatan muatan terhadap waktu atau muatan yang mengalir dalam satuan waktu dengan symbol I (dari kata Perancis : *intensita*), dengan kata lain arus adalah muatan yang bergerak.

Muatan adalah menyatakan atom terdiri dari partikel inti (proton bermuatan $+$ dan neutron bersifat netral) yang dikelilingi oleh muatan electron ($-$), normalnya atom bermuatan netral. Muatan terdiri dari dua jenis yaitu muatan positif dan negative. Arah arus searah dengan arah muatan positif atau berlawanan dengan arah aliran electron.



Previous Next

Gambar 4.3 Halaman Teori Arus

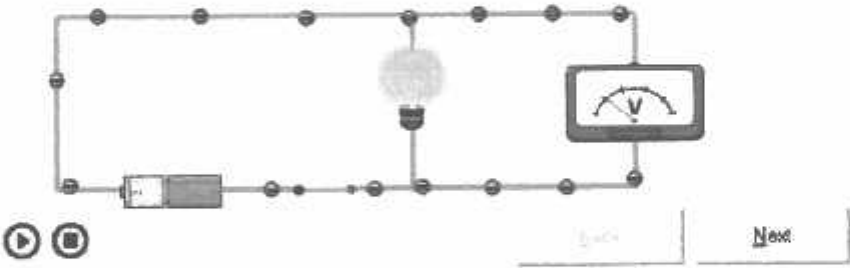
Halaman ini akan memuat materi dasar dari arus listrik beserta simulasi arus yang mengalir dalam sebuah rangkaian listrik tertutup.

4.2.4. Halaman Teori Tegangan

Tegangan

Tegangan adalah beda potensial antara dua titik yang mempunyai perbedaan jumlah muatan. Tegangan didefinisikan sebagai energi per satuan muatan listrik. Hukum Kirchoff adalah salah satu hukum yang sangat berhubungan dengan rangkaian listrik. Ada dua hukum Kirchoff, yaitu tentang arus (*Kirchoff Current Law*) dan tentang tegangan (*Kirchoff Voltage Law*).

- HUKUM KIRCHOFF I : jumlah arus menuju suatu titik cabang sama dengan jumlah arus yang meninggalkannya.
$$\sum I_{masuk} = \sum I_{keluar}$$
- HUKUM KIRCHOFF II : dalam rangkaian tertutup, jumlah aljabar GGL (\mathcal{E}) dan jumlah penurunan potensial sama dengan nol.
$$\sum E = \sum IR = 0$$



Gambar 4.4 Halaman Teori Tegangan.

Di halaman ini, akan memuat teori dasar tegangan beserta simulasinya, untuk menjalankan simulasi terlebih dahulu dengan meng-*klik* tombol *play*.

4.2.5. Halaman Teori Hambatan

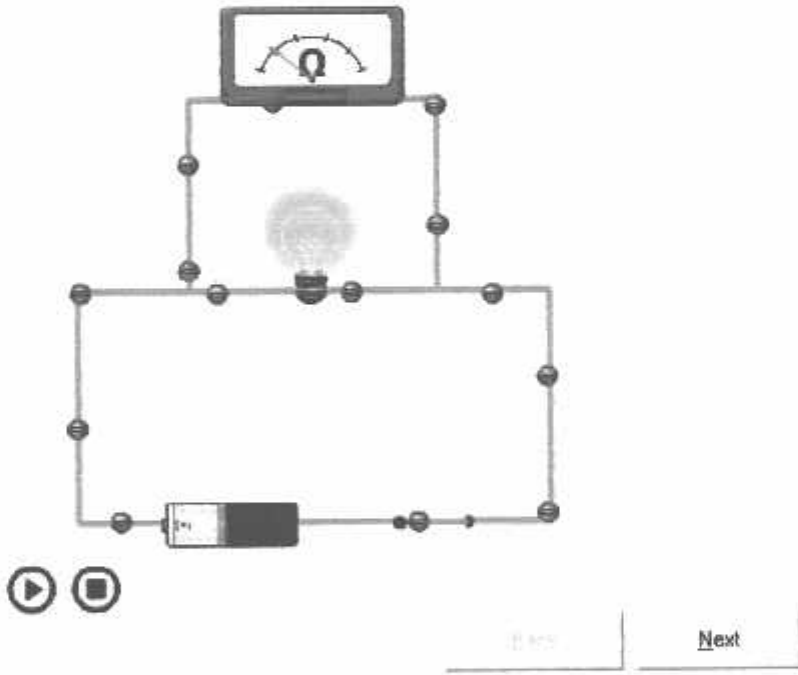
Teori Hambatan

Hambatan

Apabila terjadi beda potensial antara kedua ujung dari suatu konduktor, maka akan menyebarkan muatan listrik pada konduktor tersebut yang menyebabkan terjadinya arus listrik pada konduktor tersebut. Besarnya arus yang mengalir ini akan sebanding dengan beda potensial (tegangan) pada konduktor tersebut. Perbandingan antara besarnya beda potensial (V) dengan arus (I) yang mengalir, maka akan menunjukkan suatu besaran tertentu yang disebut dengan Konstanta. Nilai konstanta ini dinamakan dengan resistansi atau tahanan, yang diberi notasi R dalam satuan ohm (Ω), yang diambil dari nama George Simon Ohm (1787 - 1845) menyatakan : *"Tahanan satu ohm adalah besarnya resistor atau hambatan yang menyebabkan mengalirnya arus listrik sebesar satu ampere apabila pada kedua ujung resistor tersebut dihubungkan dengan sumber tegangan sebesar satu volt"*, dalam bentuk persamaan :

$$V=IR \text{ atau } R=V/I \text{ atau } I=V/R$$

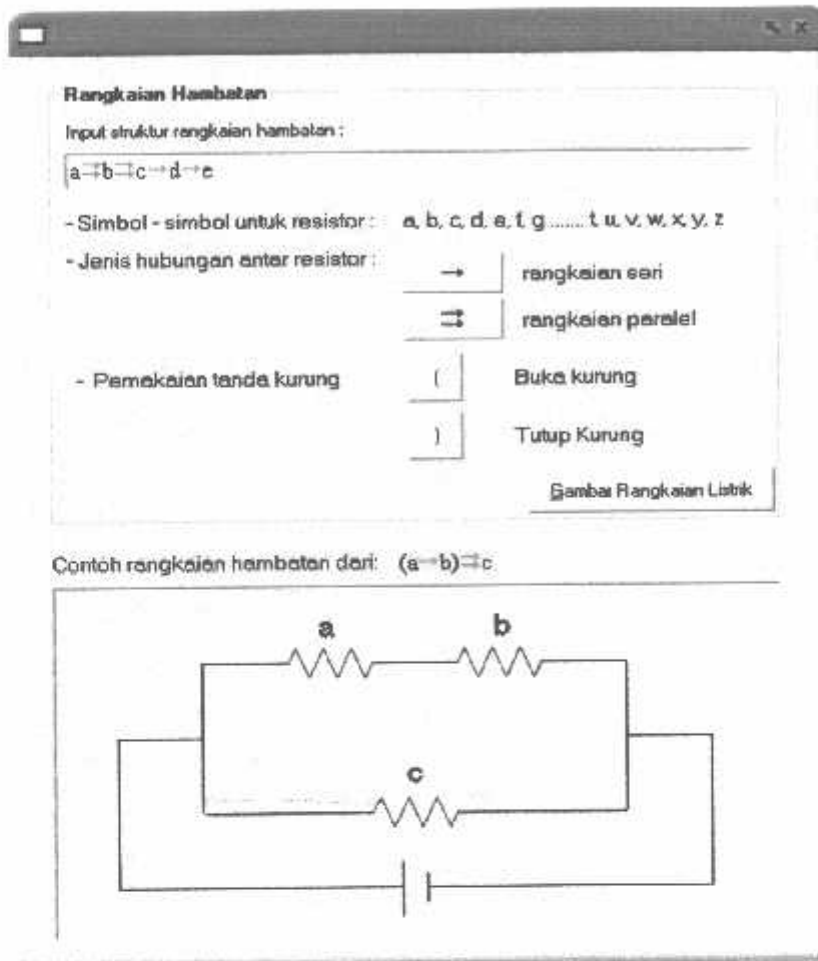
Untuk selanjutnya persamaan di atas dikenal dengan "Hukum Ohm" yang merupakan konsep dasar dalam teknik listrik yang menyatakan hubungan antara tegangan, arus dan tahanan.



Gambar 4.5 Halaman Teori Hambatan

Di halaman ini, akan memuat teori dasar hambatan beserta simulasinya, untuk menjalankan simulasi terlebih dahulu dengan meng-klik tombol *play*.

4.2.6. Halaman Desain Rangkaian



Gambar 4.6 Halaman Desain Rangkaian

Pada halaman ini, pengguna dapat mendesain sebuah rangkaian dengan memasukan maksimal 10 buah resistor. Symbol resistor itu sendiri berupa alphabet, setiap satu symbol resistor harus dipisahkan dengan jenis rangkaian baik itu seri maupun paralel. Untuk memilih jenis rangkaian ini sudah disediakan dua buah *button* yang terdapat pada *form*.

- **Algoritma Penggambaran Rangkaian Seri**

```
Private Sub GambarRangkaian()
```

```
    Dim Object As Object
    Dim A As Long
    Dim B As Long
    Dim ArrTag() As String
    Dim PosKiriX1 As Long
    Dim PosKiriY1 As Long
    Dim PosKananX1 As Long
    Dim PosKananY1 As Long
    Dim PosKiriX2 As Long
    Dim PosKiriY2 As Long
    Dim PosKananX2 As Long
    Dim PosKananY2 As Long
    Dim PosX As Long
    Dim PosY As Long
    Dim MinY As Long
    Dim MaxY As Long
    Dim TempTag As String
    Dim Temp1 As Long
    Dim Temp2 As Long
```

```
    ReDim Resistor(0)
```

```
    nResistor = 0
```

```
    'Gambar rangkaian
```

```
    For A = 1 To UBound(Rangkaian)
```

```
        With Rangkaian(A)
```

```
            'RANGKAIAN SERI
```

```
            'gambar ditampilkan sesuai dengan banyaknya R yang dimasukkan dalam RTB
```

```
            'dimana untuk rangkaian seri penambahan R selalu kekanan
```

```
            'Cari Obj Resistor-1 & simpan posisi
```

```
                For Each Object In Me
```

```
                    If Object.Tag <> "" Then
```

```
                        If TypeName(Object) = "Image" Then
```

```
                            'Bagi ke array tag
```

```
                            ArrTag = Split(Object.Tag, ",")
```

```
                            If ArrTag(0) = ".Resistor1" Then
```

```
                                PosKiriX1 = ConvX(Val(ArrTag(1)), False)
```

```
                                PosKiriY1 = ConvY(Val(ArrTag(2)), False)
```

```
                                PosKananX1 = ConvX(Val(ArrTag(3)), False)
```

```
                                PosKananY1 = ConvY(Val(ArrTag(4)), False)
```

```
                                Exit For
```

```
                            End If
```

```
                        End If
```

```
                    End If
```

```
                Next
```

```
            'Cari Obj Resistor-2 & simpan posisi
```

```
                For Each Object In Me
```

```
                    If Object.Tag <> "" Then
```

```
                        If TypeName(Object) = "Image" Then
```

```
                            'Bagi ke array tag
```

```

    ArrTag = Split(Object.Tag, ",")
    If ArrTag(0) = .Resistor2 Then
        PosKiriX2 = ConvX(Val(ArrTag(1)), False)
        PosKiriY2 = ConvY(Val(ArrTag(2)), False)
        PosKananX2 = ConvX(Val(ArrTag(3)), False)
        PosKananY2 = ConvY(Val(ArrTag(4)), False)
        Exit For
    End If
End If
End If
End If
Next

'Geser Resistor-2 di samping Resistor-1
PosX = PosKananX1 - PosKiriX2 - 1
PosY = PosKananY1 - PosKiriY2
TempTag = .Nama & "," & ConvX(PosKiriX1) & "," & ConvY(PosKiriY1) & _
    "," & ConvX(PosKananX2 + PosX) & "," & ConvY(PosKananY2 + PosY)
PosKananX2 = PosKananX2 + PosX
PosKananY2 = PosKananY2 + PosY
For Each Object In Me
    If Object.Tag <> "" Then
        'Bagi ke array tag
        ArrTag = Split(Object.Tag, ",")
        If ArrTag(0) = .Resistor1 Then
            'Resistor-1
            Object.Tag = TempTag
        ElseIf ArrTag(0) = .Resistor2 Then
            'Resistor-2
            If TypeName(Object) = "Image" Then
                Object.Left = Object.Left + ConvX(PosX)
                Object.Top = Object.Top + ConvY(PosY)
            ElseIf TypeName(Object) = "Line" Then
                Object.X1 = Object.X1 + ConvX(PosX)
                Object.X2 = Object.X2 + ConvX(PosX)
                Object.Y1 = Object.Y1 + ConvY(PosY)
                Object.Y2 = Object.Y2 + ConvY(PosY)
            End If
            Object.Tag = TempTag
        End If
    End If
End If
Next

'Cari yang paling atas -> atau y paling kecil
'jika terdapat dibawah 1, maka geser ke bawah
MinY = -9999
For Each Object In Me
    If Object.Tag <> "" Then
        If TypeName(Object) = "Image" Then
            'Bagi ke array tag
            ArrTag = Split(Object.Tag, ",")
            If ArrTag(0) = .Nama Then
                If MinY = -9999 Or (ConvRY(Object.Top, False) < MinY) Then
                    MinY = ConvRY(Object.Top, False)
                End If
            End If
        End If
    End If
End If

```

```

        End If
    End If
Next

PosY = 1 - MinY
If PosY < 0 Then
    TempTag = .Nama & "," & ConvX(PosKiriX1) & "," & ConvY(PosKiriY1 +
PosY) & _
        "," & ConvX(PosKananX2) & "," & ConvY(PosKananY2 + PosY)
    For Each Object In Me
        If Object.Tag <> "" Then
            'Bagi ke array tag
            ArrTag = Split(Object.Tag, ",")
            If ArrTag(0) = .Nama Then
                If TypeName(Object) = "Image" Then
                    Object.Top = Object.Top + ConvY(PosY)
                ElseIf TypeName(Object) = "Line" Then
                    Object.Y1 = Object.Y1 + ConvY(PosY)
                    Object.Y2 = Object.Y2 + ConvY(PosY)
                End If
                Object.Tag = TempTag
            End If
        End If
    End If
Next
End If

```

```
Elseif .Hubungan = "P" Then
```

- **Algoritma Penggambaran Rangkaian Paralel**

'Y Maksimum dari resistor - 1

```

MaxY = 0
For Each Object In Me
    If Object.Tag <> "" Then
        If TypeName(Object) = "Image" Then
            'Bagi ke array tag
            ArrTag = Split(Object.Tag, ",")
            If ArrTag(0) = .Resistor1 Then
                If (ConvRY(Object.Top, False) > MaxY) Then
                    MaxY = ConvRY(Object.Top, False)
                End If
            End If
        End If
    End If
End If
Next

```

'Y Minimum dari resistor - 2

```

MinY = -.9999
For Each Object In Me
    If Object.Tag <> "" Then
        If TypeName(Object) = "Image" Then
            'Bagi ke array tag
            ArrTag = Split(Object.Tag, ",")
            If ArrTag(0) = .Resistor2 Then

```

```

        If MinY = -9999 Or (ConvRY(Object.Top, False) < MinY) Then
            MinY = ConvRY(Object.Top, False)
        End If
    End If
End If
End If
Next

'Garis horizontal-1
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
    .X1 = PosKiriX2: .X2 = PosKiriX2 + PosX
    .Y1 = PosKiriY2: .Y2 = PosKiriY2
    .Visible = True
    .Tag = TempTag
End With

'Garis horizontal-2
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
    .X1 = PosKananX2 + PosX: .X2 = PosKananX2 + (2 * PosX)
    .Y1 = PosKananY2: .Y2 = PosKananY2
    .Visible = True
    .Tag = TempTag
End With
PosKananX2 = PosKananX2 + Temp2
Else
    'PosX < 0 - R2 lebih panjang
    'Tambah panjang R1

    TempTag = .Resistor1 & _
        "," & PosKiriX1 & _
        "," & PosKiriY1 & _
        "," & PosKananX1 + Abs(PosX) & _
        "," & PosKananY1
    Temp2 = Abs(PosX)
    PosX = Abs(PosX) / 2

    'Geser sebesar posX
    For Each Object In Me
        If Object.Tag <> "" Then
            'Bagi ke array tag
            ArrTag = Split(Object.Tag, ",")
            If ArrTag(0) = .Resistor1 Then
                If TypeName(Object) = "Image" Then
                    Object.Left = Object.Left + PosX
                    Object.ZOrder 0
                ElseIf TypeName(Object) = "Line" Then
                    Object.X1 = Object.X1 + PosX
                    Object.X2 = Object.X2 + PosX
                End If
                Object.Tag = TempTag
            End If
        End If
    End If

```

```

    End If
Next

'Garis horizontal-1
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
    .X1 = PosKiriX1: .X2 = PosKiriX1 + PosX
    .Y1 = PosKiriY1: .Y2 = PosKiriY1
    .Visible = True
    .Tag = TempTag
End With

'Garis horizontal-2
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
    .X1 = PosKananX1 + PosX: .X2 = PosKananX1 + (2 * PosX)
    .Y1 = PosKananY1: .Y2 = PosKananY1
    .Visible = True
    .Tag = TempTag
End With
PosKananX1 = PosKananX1 + Temp2
End If
End If

'Selish-Y
Temp2 = ConvY((PosKiriY2 - PosKiriY1) \ 2, False)

'Temporary Tag
TempTag = .Nama & _
    "," & ConvX(1) & _
    "," & PosKiriY1 + ConvY(Temp2) & _
    "," & PosKananX1 + ConvX(1) & _
    "," & PosKananY1 + ConvY(Temp2)

'Garis vertikal-1
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
    .X1 = PosKiriX1: .X2 = PosKiriX2
    .Y1 = PosKiriY1: .Y2 = PosKiriY2
    .Visible = True
    .Tag = TempTag
End With

'Garis vertikal-2
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
    .X1 = PosKananX1: .X2 = PosKananX2
    .Y1 = PosKananY1: .Y2 = PosKananY2
    .Visible = True
    .Tag = TempTag
End With

```

```

'Garis horizontal - 1
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
.X1 = ConvX(1): .X2 = ConvX(2)
.Y1 = PosKiriY1 + ConvY(Temp2)
.Y2 = PosKiriY1 + ConvY(Temp2)
.Visible = True
.Tag = TempTag
End With

'Garis horizontal - 2
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
.X1 = PosKananX1: .X2 = PosKananX1 + ConvX(1)
.Y1 = PosKiriY1 + ConvY(Temp2)
.Y2 = PosKiriY1 + ConvY(Temp2)
.Visible = True
.Tag = TempTag
End With

'Susun posisi label
For A = 1 To lblR.UBound
lblR(A).Top = imgR(A).Top - 120
lblR(A).Left = imgR(A).Left + 315
lblR(A).ZOrder 0
Next A

'Gambar voltase
TempTag = imgR(1).Tag
ArrTag = Split(TempTag, ",")
PosKiriX1 = Val(ArrTag(1))
PosKiriY1 = Val(ArrTag(2))
PosKananX1 = Val(ArrTag(3))
PosKananY1 = Val(ArrTag(4))

'Garis vertikal-1
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
.BorderColor = &H80&
.X1 = PosKiriX1: .X2 = PosKiriX1
.Y1 = PosKiriY1: .Y2 = MaxY
.Visible = True
.Tag = "VOLT"
End With

'Garis vertikal-1
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
.BorderColor = &H80&
.X1 = PosKananX1: .X2 = PosKananX1

```

```

.Y1 = PosKananY1: .Y2 = MaxY
.Visible = True
.Tag = "VOLT"
End With

'Garis horizontal-1
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
.BorderColor = &H80&
.X1 = PosKiriX1: .X2 = PosKananX1
.Y1 = MaxY: .Y2 = MaxY
.Visible = True
.Tag = "VOLT"
End With

imgV.Top = MaxY - 450
imgV.Left = nSkalaX + (PosKananX1 - PosKiriX1 - imgV.Width) \ 2

lblV.Top = imgV.Top - 225
lblV.Left = imgV.Left
End Sub

```

4.2.7. Halaman Penghitungan

GAMBAR RANGKAIAN HAMBATAN : a → b → c → d → e

Input tegangan total (V-total) : 100 volt Pembagian kuat arus total (I-total) : V-I-R untuk setiap elemen :

No	Symbol Resistor	Besarnya (ohm)
1	c	8
2	d	12
3	e	15
4	a	10
5	b	14

BESAR R-total dan I-total

1) R(c) dan R(d) merupakan rangkaian seri, sehingga perhitungan R(cd) adalah :

$$R(cd) = R(c) + R(d) = 8 + 12$$

R(abode) = R(ab) - PARALEL - R(cde)
 Karena merupakan rangkaian paralel, maka :
 $V(ab) = V(cde) = V(abode) = 100 \text{ Volt}$

Gambar 4.7 Halaman Penghitungan

Setelah mendesain rangkaian pada halaman sebelumnya, maka gambar rangkaian akan muncul. Sebelum melakukan penghitungan besar hambatan total dan kuat arus total serta tegangan dan kuat arus pada masing – masing elemen resistor yang terdapat di dalam rangkaian terlebih dahulu pengguna memasukkan nilai Vtotal dan besaran untuk masing-masing resistor.

Untuk contohnya penulis mendesain rangkaian seperti ini : a → b → c → d → e, maka gambar rangkaian yang muncul ada pada gambar di atas dengan hasil penghitungan sebagai berikut :

BESAR R-total dan I-total

1) R(c) dan R(d) merupakan rangkaian seri, sehingga perhitungan R(cd) adalah :

$$R(cd) = R(c) + R(d) = 8 + 12$$

$$R(cd) = 20 \text{ ohm}$$

2) R(cd) dan R(e) merupakan rangkaian seri, sehingga perhitungan R(cde) adalah :

$$R(cde) = R(cd) + R(e) \\ = 20 + 15$$

$$R(cde) = 35 \text{ ohm}$$

3) R(a) dan R(b) merupakan rangkaian paralel, sehingga perhitungan R(ab) adalah :

$$R(ab) = (R(a) * R(b)) / (R(a) + R(b)) \\ = (10 * 14) / (10 + 14)$$

$$R(ab) = 5.83 \text{ ohm}$$

4) R(ab) dan R(cde) merupakan rangkaian paralel, sehingga perhitungan R(abcde) adalah :

$$R(abcde) = (R(ab) * R(cde)) / (R(ab) + R(cde)) \\ = (5.83 * 35) / (5.83 + 35)$$

$$R(abcde) = 5 \text{ ohm}$$

$$R(\text{total}) = R(abcde) = 5 \text{ Ohm}$$

$$V(\text{total}) = 100 \text{ Volt}$$

$$\text{sehingga } I(\text{total}) = V(\text{total}) / R(\text{total}) \\ = 100 / 5$$

$$I(\text{total}) = 20 \text{ Ampere}$$

V-I-R untuk setiap elemen

$$R(abcde) = R(ab)\text{-PARALEL-}R(cde)$$

Karena merupakan rangkaian paralel, maka :

$$V(ab) = V(cde) = V(abcde) = 100 \text{ Volt}$$

$$I(ab) = V(ab) / R(ab)$$

$$= 100 / 5.83$$

$$= 17.15 \text{ Ampere}$$

$$I(cde) = V(cde) / R(cde)$$

$$= 100 / 35$$

$$= 2.86 \text{ Ampere}$$

$$R(ab) = R(a)\text{-PARALEL-}R(b)$$

Karena merupakan rangkaian paralel, maka :

$$V(a) = V(b) = V(ab) = 100 \text{ Volt}$$

$$I(a) = V(a) / R(a)$$

$$= 100 / 10$$

$$= 10 \text{ Ampere}$$

$$I(b) = V(b) / R(b)$$

$$= 100 / 14$$

$$= 7.14 \text{ Ampere}$$

$$R(cde) = R(cd)\text{-SERI-}R(e)$$

Karena merupakan rangkaian seri, maka :

$$V(cd) = (R(cd) / R(cde)) * V(cde)$$

$$= (20 / 35) * 100$$

$$= 57.14 \text{ Volt}$$

$$V(e) = (R(e) / R(cde)) * V(cde)$$

$$= (15 / 35) * 100$$

$$= 42.86 \text{ Volt}$$

$$I(cd) = I(e) = I(cde) = 2.86 \text{ Ampere}$$

$$R(cd) = R(c)\text{-SERI-}R(d)$$

Karena merupakan rangkaian seri, maka :

$$V(c) = (R(c) / R(cd)) * V(cd)$$

$$= (8 / 20) * 57.14$$

$$= 22.86 \text{ Volt}$$

$$V(d) = (R(d) / R(cd)) * V(cd)$$

$$= (12 / 20) * 57.14$$

$$= 34.28 \text{ Volt}$$

$$I(c) = I(d) = I(cd) = 2.86 \text{ Ampere}$$

Algoritma Penghitungan Itotal :

```

For A = 1 To UBound(Rangkaian)
  With Rangkaian(A)
    If .Hubungan = "S" Then
      'Rangkaian seri

      'Perhitungan seri
      .NilaiR = AmbilNilai(.Resistor1, "R") + AmbilNilai(.Resistor2, "R")

    ElseIf .Hubungan = "P" Then
      'Rangkaian paralel

      'Perhitungan paralel
      .NilaiR = Bagi((AmbilNilai(.Resistor1, "R") * AmbilNilai(.Resistor2,
        "R")), _
        (AmbilNilai(.Resistor1, "R") + AmbilNilai(.Resistor2, "R")))

    Else
      .NilaiR = AmbilNilai(.Resistor1, "R")
    End If
  End With
Next A

```

```

nRTotal = Rangkaian(UBound(Rangkaian)).NilaiR
nVTotal = Val(txtVolt)
nITotal = Bagi(nVTotal, nRTotal)

With Rangkaian(UBound(Rangkaian))
  .NilaiV = nVTotal
  .NilaiI = nITotal
End With

```

Algoritma Penghitungan VIR Setiap Elemen :

Algoritma ini akan menghitung nilai VIR setiap elemen dengan dua cara yaitu menghitung jenis rangkaian resistor secara seri dan parallel.

Untuk yang pertama kali dihitung adalah rangkaian resistor jenis seri (*kondisi if .Hubungan = "S" then*) dan apabila tidak ada jenis rangkaian resistor parallel, maka kondisi kedua *ElseIf .Hubungan = "P"* Then tidak akan dijalankan. Tetapi jika ada, maka kondisi *else if* akan dilanjutkan.

```

For A = UBound(Rangkaian) To 1 Step -1
  With Rangkaian(A)
    If .Hubungan = "S" Then
      //Proses Hitung untuk rangkaian seri//
      R1 = AmbilNilai(.Resistor1, "R")
      R2 = AmbilNilai(.Resistor2, "R")

      V1 = Bagi(R1 * .NilaiV, .NilaiR)
      V2 = Bagi(R2 * .NilaiV, .NilaiR)

      I1 = .NilaiI
      I2 = .NilaiI

      'V untuk resistor / rangkaian - I
      Temp1 = AmbilKet(.Resistor1)
      If IsTVar(.Resistor1) Then
        'Rangkaian
        Rangkaian(Mid(.Resistor1, 3, Len(.Resistor1) - 3)).NilaiV = V1
        Rangkaian(Mid(.Resistor1, 3, Len(.Resistor1) - 3)).NilaiI = I1
      Else
        'Resistor
        Indeks = AmbilIndeksR(.Resistor1)

        Resistor(Indeks).NilaiV = V1
        Resistor(Indeks).NilaiI = I1
      End If
    End If
  End With
Next A

```

```

'V untuk resistor / rangkaian - II
Temp2 = AmbilKet(.Resistor2)
If IsTVar(.Resistor2) Then
    'Rangkaian
    Rangkaian(Mid(.Resistor2, 3, Len(.Resistor2) - 3)).NilaiV = V2
    Rangkaian(Mid(.Resistor2, 3, Len(.Resistor2) - 3)).NilaiI = I2
Else
    'Resistor
    Indeks = AmbilIndeksR(.Resistor2)

    Resistor(Indeks).NilaiV = V2
    Resistor(Indeks).NilaiI = I2
End If

ElseIf .Hubungan = "P" Then
    'Rangkaian paralel

    R1 = AmbilNilai(.Resistor1, "R")
    R2 = AmbilNilai(.Resistor2, "R")

    V1 = .NilaiV
    V2 = .NilaiV

    I1 = Bagi(V1, R1)
    I2 = Bagi(V2, R2)

    'V1=V2
    Temp1 = AmbilKet(.Resistor1)
    Temp2 = AmbilKet(.Resistor2)

    'I untuk resistor / rangkaian - I
    If IsTVar(.Resistor1) Then
        'Rangkaian
        Rangkaian(Mid(.Resistor1, 3, Len(.Resistor1) - 3)).NilaiV = V1
        Rangkaian(Mid(.Resistor1, 3, Len(.Resistor1) - 3)).NilaiI = I1
    Else
        'Resistor
        Indeks = AmbilIndeksR(.Resistor1)

        Resistor(Indeks).NilaiV = V1
        Resistor(Indeks).NilaiI = I1
    End If

    'I untuk resistor / rangkaian - II
    If IsTVar(.Resistor2) Then
        'Rangkaian
        Rangkaian(Mid(.Resistor2, 3, Len(.Resistor2) - 3)).NilaiV = V2
        Rangkaian(Mid(.Resistor2, 3, Len(.Resistor2) - 3)).NilaiI = I2
    Else
        'Resistor
        Indeks = AmbilIndeksR(.Resistor2)

        Resistor(Indeks).NilaiV = V2
        Resistor(Indeks).NilaiI = I2
    End If

Else

```

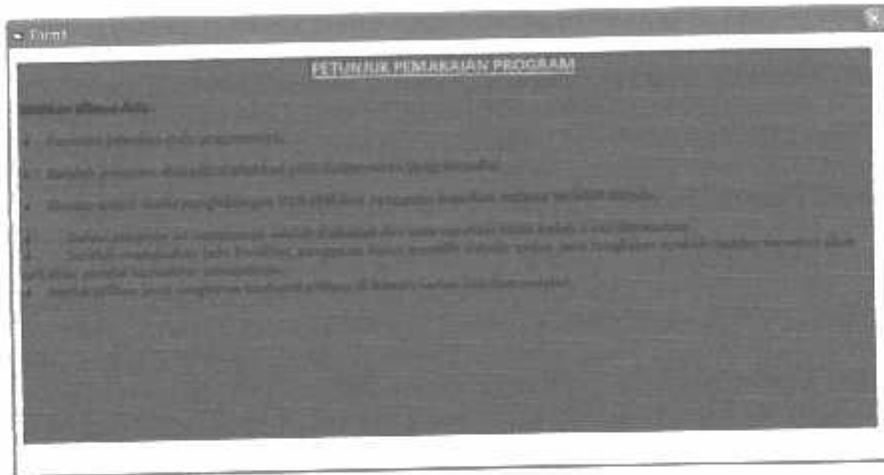
```
'=  
Indeks = AmbilIndeksR(.Resistor1)
```

```
Resistor(Indeks).NilaiV = .NilaiV  
Resistor(Indeks).NilaiI = .NilaiI
```

```
End If
```

```
End With
```

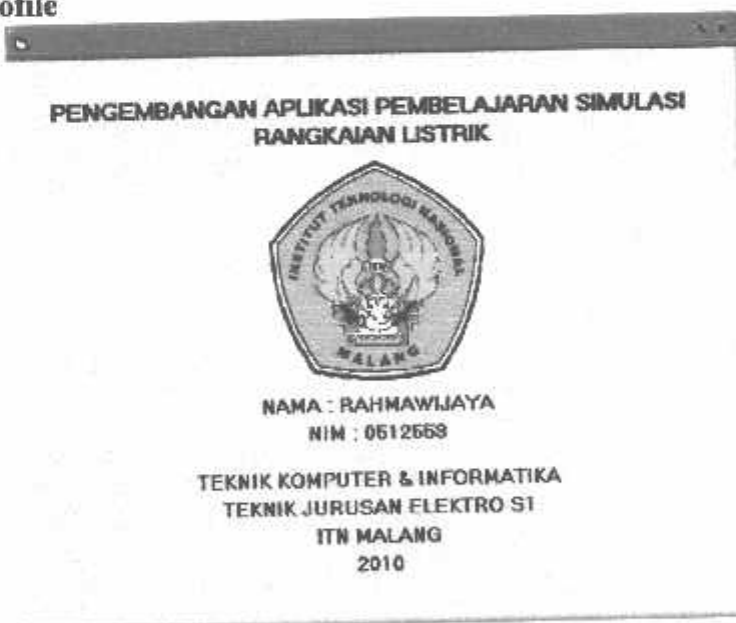
4.2.8. Halaman petunjuk



Gambar 4.8 Halaman Petunjuk

Di halaman ini akan memuat petunjuk-petunjuk bagaimana cara menggunakan aplikasi pembelajaran ini.

4.2.9. Halaman Profile



Gambar 4.9 Halaman Profil

Halaman ini memuat judul dan profil penulis serta nama institusi.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah selesai merancang aplikasi pembelajaran ini, maka penulis menarik kesimpulan sebagai berikut :

1. Aplikasi pembelajaran mampu melakukan perhitungan besar tegangan dan kuat arus untuk setiap elemen resistor (hambatan).
2. Aplikasi pembelajaran mampu menjabarkan teori dan simulasi rangkaian listrik.
3. Aplikasi pembelajaran menyediakan fasilitas 'Cetak' untuk mencetak hasil perhitungan.
4. Jumlah resistor yang dapat di-*input* adalah 10 buah.
5. Bentuk rangkaian dapat dirancang sesuai dengan input struktur rangkaian.

5.2 Saran

Penulis ingin memberikan beberapa saran yang mungkin dapat digunakan dalam pengembangan lebih lanjut pada perancangan Aplikasi pembelajaran ini, yaitu :

1. Aplikasi pembelajaran dapat dikembangkan dengan menyediakan beberapa soal (pertanyaan) untuk dijawab.
2. Aplikasi pembelajaran dapat dikembangkan dengan menyediakan lebih dari satu buah sumber tegangan atau menggunakan sumber tegangan AC.
3. Aplikasi pembelajaran dapat dikembangkan dengan menyediakan animasi yang dinamis.



DAFTAR PUSTAKA

- LPKBM MADCOMS – MADIUN.** *Panduan Perograman dan Referensi Kamus Visual Basic 6.0.* Penerbit ANDI, Madiun 2005.
- BUNAFIT KOMPUTER.** *25 Aplikasi Windows Populer dengan Visual Basic 6.0.* Elex Media Komputindo, Jakarta 2008.
- HASBALLAH, FAJRILLAH.** *Aplikasi Game dan Multimedia dengan VB.* Elex Media Komputindo, Jakarta 2009.
- ZEEMBRY.** *Menggambar dan Animasi Karakter dengan Flash MX 2004.* Elex Media Komputindo, Jakarta 2004.
- PRIYANTO HIDAYATULLAH, M, AMARULLAH AKBAR, ZAKY RAHIM.** *Making Educational Animation Using Flash.* Penerbit Informatika, Bandung 2008.
- SURYATMO. F.** *Dasar-dasar Teknik Listrik Cetakan Ketiga.* Bina Adikasara, Jakarta 2002.
- WILLIAM H. HAYT, JR, JACK E. KEMMERLY, STEVEN M. DURBIN.** *Rangkaian Listrik Jilid I Edisi Keenam.* Penerbit Erlangga, Jakarta 2005.
- MILITON GUSSOW, WILLIAM T. SMITH.** *Dasar-dasar Teknik Listrik Schaum's easy outlines.* Penerbit Erlangga, Jakarta 2002.
- Tempat nongkrong Belajar vb 6.0 www.amirmahmud2008.blogspot.com.
- Yanis, artikel Flash www.IlmuKomputer.com
- <http://digilib.stikom.edu>.
- <http://www.e-dukasi.net>.
- <http://nic.unud.ac.id/~dayu/EC/Lecture1.pdf>.



LAMPIRAN





INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK KOMPUTER & INFORMATIKA S-1
Jl. Raya Karanglo KM. 2 Malang

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : RAHMAWIJAYA
NIM : 05.12.553
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer & Informatika
Judul Skripsi : Pengembangan Aplikasi Pembelajaran Simulasi Rangkaian Listrik.

Dipertahankan di hadapan majelis penguji Skripsi jenjang Strata satu (S-1) pada :

Hari : Senin
Tanggal : 08 Februari 2010
Dengan Nilai : 79,00 (B+) *2/4*



Ketua Majelis Penguji

(Ir. H. Sidik Noertjahjono, MT)

NIP.Y. 1028700163

PANITIA UJIAN SKRIPSI

Sekretaris Majelis Penguji

(Ir. F. Yudi Limpraptono, MT)

NIP.Y. 1039500274

ANGGOTA PENGUJI

Dosen Penguji I

Dr. Eng. Aryugnto Soetedjo, ST. MT

NIP.Y. 1030800417

Dosen Penguji II

(Sotvohadi, ST)

NIP.1039700309



FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer & Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Rahmawijaya
NIM : 05 12 553
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer & Informatika
Masa Bimbingan : 03 Desember 2009 s/d 03 Mei 2010.
Judul Skripsi : Pengembangan Aplikasi Pembelajaran Simulasi Rangkaian Listrik.

Penguji/Tanggal	Uraian	Paraf
Penguji I 08 Februari 2010	Penjelasan dan pemahaman algoritma untuk menggambarkan dan menghitung VIR di laporan belum ada.	
	Flowchart program	

Dosen Penguji,
Penguji I

(Dr. Eng. Aryuanto Soetedjo, ST, MT)
NIP. 1030800417

Mengetahui,
Dosen Pembimbing

(Joseph Dedy Irawan ST, MT)
NIP. 132315178



FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer & Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Rahmawijaya
NIM : 05 12 553
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer & Informatika
Masa Bimbingan : 03 Desember 2009 s/d 03 Mei 2010.
Judul Skripsi : Pengembangan Aplikasi Pembelajaran Simulasi Rangkaian Listrik.

Penguji/Tanggal	Uraian	Paraf
Penguji II 08 Februari 2010	Tambahkan button tanda kurung terbuka dan kurung tertutup pada design rangkaian.	
	Analisa dan algoritma dari, misalnya : $a \rightarrow b / a \Rightarrow b$	
	Flowchart penghitungan VIR dan I Total	

Dosen Penguji,
Penguji II

(Soriyohadi, ST)
NIP.Y.1039700309

Mengetahui,
Dosen Pembimbing

(Joseph Dedy Irawan ST, MT)
NIP.132315178



FORMULIR BIMBINGAN SKRIPSI

Nama : RAHMAWIJAYA
Nim : 05.12.553
Masa Bimbingan : 03 Desember 2009 s/d 03 Mei 2010
Judul Skripsi : PENGEMBANGAN APLIKASI PEMBELAJARAN
SIMULASI RANGKAIAN LISTRIK

No.	Tanggal	Uraian	Paraf Pembimbing
1.	25/01/2010	Kumpul BAB I, II dan III	
2.	25/01/2010	Revisi Pendahuluan, Latar Belakang (BAB I). Revisi Flowchat (keterangan Ya/Tidak)	
3.	25/01/2010	ACC BAB I,II,III	
4.	25/01/2010	Kumpul BAB IV, V, Daftar Pustaka, Daftar Isi.	
5.	25/01/2010	Kumpul Lembar Persetujuan.	
6.	25/01/2010	Revisi Daftar Pustaka (salah pengetikan)	
7.	25/01/2010	ACC BAB IV, V, Daftar Pustaka.	
8.	25/01/2010	Tanda tangan lembar persetujuan.	
9.	25/01/2010	ACC Kompre.	

Malang,
Dosen Pembimbing,

JOSEPH/DEDY IRAWAN, ST. MT.
NIP. 132315178

Form S-4b

- **Form Menu Utama**

```
Private Sub Button_Click(Index As Integer)
```

```
  Select Case Index
```

```
  Case 0
```

```
    Me.Hide
```

```
    FrmTeoriArus.Show
```

```
  Case 1
```

```
    Me.Hide
```

```
    FrmTeoriHambatan.Show
```

```
  Case 2
```

```
    Me.Hide
```

```
    FrmTeoriTegangan.Show
```

```
  Case 3
```

```
    Me.Hide
```

```
    frmPerhitungan.Show
```

```
  Case 4
```

```
    Me.Hide
```

```
    FrmPetunjuk.Show
```

```
  Case 5
```

```
    Me.Hide
```

```
    FrmProfilPenulis.Show
```

```
  End Select
```

```
End Sub
```

- **Form Desain Rangkaian**

```
Option Explicit
```

```
Private bClick As Boolean
```

```
Private nPosisi As Integer
```

```
Private Sub buttonbuka_Click()
```

```
  bClick = True
```

```
  nPosisi = RTB.SelStart
```

```
  RTB.Text = Left(RTB.Text, nPosisi) & "(" &  
  Right(RTB.Text, Len(RTB.Text) - nPosisi)
```

```
End Sub
```

```
Private Sub buttontutup_Click()
```

```
  bClick = True
```

```
  nPosisi = RTB.SelStart
```

```
  RTB.Text = Left(RTB.Text, nPosisi) & ")" &  
  Right(RTB.Text, Len(RTB.Text) - nPosisi)
```

```
End Sub
```

```
Private Sub cmdParalel_Click()
```

```
  bClick = True
```

```
  nPosisi = RTB.SelStart  
  RTB.Text = Left(RTB.Text, nPosisi) & "I" &  
  Right(RTB.Text, Len(RTB.Text) - nPosisi)  
End Sub
```

```
Private Sub cmdSeri_Click()
```

```
  bClick = True
```

```
  nPosisi = RTB.SelStart  
  RTB.Text = Left(RTB.Text, nPosisi) &  
  Chr(34) & Right(RTB.Text, Len(RTB.Text) -  
  nPosisi)  
End Sub
```

```
Private Sub Form_Load()
```

```
  RTBExmp = "(a" & Chr(34) & "b)Ic"  
  Call RefreshText(RTBExmp, False)
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
  FrmMenuUtama.Show
```

```
End Sub
```

```
Private Sub RTB_Change()
```

```
  Call RefreshText(RTB, bClick)
```

```
End Sub
```

```
Private Sub RTB_GotFocus()
```

```
  bClick = False
```

```
End Sub
```

```
Private Sub RTB_KeyPress(KeyAscii As  
Integer)
```

```
  If Chr(KeyAscii) <> "(" And Chr(KeyAscii)  
<> ")" And KeyAscii <> vbKeyBack And  
  (Chr(KeyAscii) Like "[a-z]" = False) Then  
    KeyAscii = 0
```

```
  End If
```

```
End Sub
```

```
Private Sub cmdGambar_Click()
```

```
  Dim A As Integer
```

```
  Dim A1 As Integer
```

```
  Dim cVar1 As String
```

```
  Dim cVar2 As String
```

```
  Dim Juml_Rangkaian As Byte
```

```
  On Error GoTo errHandler
```

```
  'Variabel sementara
```



```

cVar1 = HilangkanSpasi(RTB.Text)

'cVar1 kosong
If Len(cVar1) = 0 Then
    MsgBox "Rangkaian hambatan kosong !",
vbCritical
    Exit Sub
End If
Juml_Rangkaian = 0
Dim i As Long
For i = 1 To Len(RTB.Text)
    If Asc(Mid(RTB.Text, i, 1)) = 34 Or
Mid(RTB.Text, i, 1) = "I" Then
        Juml_Rangkaian = Juml_Rangkaian + 1
    End If
Next
If Juml_Rangkaian > 9 Then
    MsgBox "hambatan harus <=10 !",
vbCritical
    Exit Sub
End If
'Periksa symbol
For A = 1 To Len(cVar1)
    cVar2 = Mid(cVar1, A, 1)
    If (cVar2 Like "[a-z]" Or _
        InStr(1, "()|" & Chr(34), cVar2) <> 0) =
False Then
        MsgBox "Struktur rangkaian tidak benar
!" & Chr(13) & _
            "Symbol "" & cVar2 & "" tidak
dikenal !", vbCritical
        Exit Sub
    End If
Next A

'Struktur tidak benar
If Periksa(cVar1) = False Then
    MsgBox "Struktur rangkaian tidak benar !",
vbCritical
    Exit Sub
End If

'Periksa apakah ada symbol resistor yang
sama
For A = 1 To Len(cVar1)
    If InStr(1, "()|" & Chr(34), Mid(cVar1, A,
1)) = 0 Then
        For A1 = A + 1 To Len(cVar1)
            If Mid(cVar1, A, 1) = Mid(cVar1, A1,
1) Then

```

```

        MsgBox "Terdapat penulisan
symbol "" & Mid(cVar1, A, 1) & "" lebih dari 1
kali !", vbCritical
        Exit Sub
    End If
Next A1
End If
Next A

'Rangkaian Hambatan
strRangkaian = cVar1

'Bagi rangkaian
Call BagiRangkaianH(strRangkaian,
Rangkaian)
strRangkaian = cVar1

Me.Hide
FrmRangkaian.Show
Exit Sub
errHandler:
    MsgBox Err.Description, vbCritical
End Sub

```

```

Private Function HilangkanSpasi(pcTeks As
String) As String
    Dim A As Integer
    Dim cVar1 As String
    Dim cVar2 As String

    For A = 1 To Len(pcTeks)
        cVar1 = Mid(pcTeks, A, 1)
        If cVar1 <> " " Then
            cVar2 = cVar2 & cVar1
        End If
    Next A

    HilangkanSpasi = cVar2
End Function

```

```

Private Function Periksa(pcTeks As String) As
Boolean
    Dim A As Integer
    Dim cVar1 As String
    Dim cVar2 As String
    Dim cNow As String
    Dim cLast As String
    Dim nT As Double

```

```

Periksa = True

'Periksa kurung '(', ')'
For A = 1 To Len(pcTeks)
  cVar1 = Mid(pcTeks, A, 1)
  If cVar1 = "(" Then nT = nT + 1
  If cVar1 = ")" Then nT = nT - 1
Next A
If nT <> 0 Then
  Periksa = False
  Exit Function
End If

'Periksa struktur
cVar1 = ""; cVar2 = ""
For A = 1 To Len(pcTeks)
  Temp
  cVar1 = Mid(pcTeks, A, 1)
  If cVar1 Like "[a-z]" Then
    cNow = "symbol"
  ElseIf cVar1 = "I" Or cVar1 = Chr(34)
Then
    cNow = "rangkaian"
  Else
    cNow = cVar1
  End If

  'Struktur
  If cNow = "rangkaian" Then
    If cLast = "" Or cLast = "rangkaian" Or
cVar2 = "(" Then
      Periksa = False
      Exit Function
    End If
  ElseIf cNow = "symbol" Then
    If cLast = "symbol" Or cVar2 = ")" Then
      Periksa = False
      Exit Function
    End If
  ElseIf cVar1 = "(" Then
    If cLast = "symbol" Or cVar2 = ")" Then
      Periksa = False
      Exit Function
    End If
  ElseIf cVar1 = ")" Then
    If cLast = "rangkaian" Or cVar2 = "("
Then
      Periksa = False
      Exit Function
    End If

    End If
  End If

  'Simpan var akhir
  cVar2 = cVar1
  cLast = cNow
Next A
End Function

'Bagi rangkaian listrik - header
Private Sub BagiRangkaianH(pcTeks As String,
pcRangkaian() As RL)
  Dim nP1 As Integer
  Dim nP2 As Integer
  Dim nLP1 As Integer
  Dim cLangkah As String

  nRangkaian = 0
  ReDim pcRangkaian(nRangkaian)

  While IsTVar(pcTeks) = False

    nP2 = InStr(1, pcTeks, ")")

    If nP2 > 0 Then

      'Ada pemisah
      nP1 = 1
      While nP1 > 0 And nP1 < nP2
        nLP1 = nP1 'Last Value
        nP1 = InStr(nP1 + 1, pcTeks, "(")
      Wend
      cLangkah = Mid(pcTeks, nLP1 + 1, nP2
- nLP1 - 1)

      'Rangkaian Detail
      Call BagiRangkaianD(cLangkah,
pcRangkaian)
      pcTeks = Left(pcTeks, nLP1 - 1) &
cLangkah & Right(pcTeks, Len(pcTeks) - nP2)

    Else

      'Tidak ada pemisah
      cLangkah = pcTeks

      'Rangkaian Detail
      Call BagiRangkaianD(cLangkah,
pcRangkaian)
      pcTeks = cLangkah
    End If
  End While
End Sub

```

```

End If
Wend
End Sub

'Bagi rangkaian listrik - detail
Private Sub BagiRangkaianD(pcLangkah As String, pcRangkaian1() As RL)

    Dim nP1 As Integer
    Dim nP2 As Integer
    Dim nP3 As Integer
    Dim cOperasi As String

    'Loop hingga pcLangkah diubah menjadi Temp.Var
    While IsTVar(pcLangkah) = False
        nRangkaian = nRangkaian + 1
        ReDim Preserve pcRangkaian1(nRangkaian)

        If InStr(1, pcLangkah, Chr(34)) > 0 Then
            'Rangkaian seri

            nP1 = InStr(1, pcLangkah, Chr(34))

            '*** Ambil R sebelah kiri
            If Mid(pcLangkah, nP1 - 1, 1) = "]" Then
                nP2 = nP1 - 1
                While Mid(pcLangkah, nP2, 1) <> "T"
                    nP2 = nP2 - 1
                Wend
            Else
                nP2 = nP1 - 1
            End If

            '*** Ambil R sebelah kanan
            If Mid(pcLangkah, nP1 + 1, 1) = "T"
Then
                nP3 = nP1 + 1
                While Mid(pcLangkah, nP3, 1) <> "]"
                    nP3 = nP3 + 1
                Wend
            Else
                nP3 = nP1 + 1
            End If

            'Simpan

```

```

With pcRangkaian1(nRangkaian)
    .Nama = "T[" & nRangkaian & "]"
    .Hubungan = "S"

    .Resistor1 = Mid(pcLangkah, nP2, nP1
- nP2)
    If IsTVar(.Resistor1) Then
        .Keterangan = .Keterangan & _
            Rangkaian(Mid(.Resistor1, 3,
Len(.Resistor1) - 3)).Keterangan
    Else
        .Keterangan = .Keterangan &
.Resistor1
    End If

    .Resistor2 = Mid(pcLangkah, nP1 + 1,
nP3 - nP1)
    If IsTVar(.Resistor2) Then
        .Keterangan = .Keterangan & _
            Rangkaian(Mid(.Resistor2, 3,
Len(.Resistor2) - 3)).Keterangan
    Else
        .Keterangan = .Keterangan &
.Resistor2
    End If
End With

    pcLangkah = Left(pcLangkah, nP2 - 1)
    & "T[" & nRangkaian & "]" &
    Right(pcLangkah, Len(pcLangkah) - nP3)
    ElseIf InStr(1, pcLangkah, "I") > 0 Then
        'Rangkaian paralel

        nP1 = InStr(1, pcLangkah, "I")

        '*** Ambil R sebelah kiri
        If Mid(pcLangkah, nP1 - 1, 1) = "]" Then
            nP2 = nP1 - 1
            While Mid(pcLangkah, nP2, 1) <> "T"
                nP2 = nP2 - 1
            Wend
        Else
            nP2 = nP1 - 1
        End If

        '*** Ambil R sebelah kanan
        If Mid(pcLangkah, nP1 + 1, 1) = "T"
Then
            nP3 = nP1 + 1
            While Mid(pcLangkah, nP3, 1) <> "]"

```

```

        nP3 = nP3 + 1
    Wend
Else
    nP3 = nP1 + 1
End If

'Simpan
With pcRangkaian1(nRangkaian)
    .Nama = "T[" & nRangkaian & "]"
    .Hubungan = "P"

    .Resistor1 = Mid(pcLangkah, nP2, nP1
- nP2)
    If IsTVar(.Resistor1) Then
        .Keterangan = .Keterangan & _
Rangkaian(Mid(.Resistor1, 3,
Len(.Resistor1) - 3)).Keterangan
    Else
        .Keterangan = .Keterangan &
.Resistor1
    End If

    .Resistor2 = Mid(pcLangkah, nP1 + 1,
nP3 - nP1)
    If IsTVar(.Resistor2) Then
        .Keterangan = .Keterangan & _
Rangkaian(Mid(.Resistor2, 3,
Len(.Resistor2) - 3)).Keterangan
    Else
        .Keterangan = .Keterangan &
.Resistor2
    End If
End With

pcLangkah = Left(pcLangkah, nP2 - 1)
& "T[" & nRangkaian & "]" &
Right(pcLangkah, Len(pcLangkah) - nP3)
Else

'Simpan
With pcRangkaian1(nRangkaian)
    .Nama = "T[" & nRangkaian & "]"
    .Hubungan = "="

    .Resistor1 = pcLangkah
    .Keterangan = .Keterangan &
.Resistor1

    .Resistor2 = ""
End With

```

```

        pcLangkah = "T[" & nRangkaian & "]"
    End If
Wend

End Sub

'Format Text
Private Sub RefreshText(poRTB As
RichTextBox, pbPosisi As Boolean)
    Dim i As Integer
    Dim nP As Integer

    On Error Resume Next

    With poRTB
        nP = .SelStart
        .Visible = False
        For i = 1 To Len(.Text)
            If Asc(Mid(.Text, i, 1)) = 34 Or
Mid(.Text, i, 1) = "I" Then
                .SelStart = i - 1
                .SelLength = 1
                .SelFontName = "Wingdings 3"
                .SelFontSize = 11
                .SelColor = &HFF&
            Else
                .SelStart = i - 1
                .SelLength = 1
                .SelFontName = "Times New Roman"
                .SelFontSize = 11
                .SelColor = 0
            End If
        Next i
        DoEvents: DoEvents
        .Visible = True
        .SetFocus
        If pbPosisi Then
            .SelStart = nPosisi + 1
        Else
            .SelStart = nP
        End If
    End With
End Sub

```

• **Form Penghitungan**

Option Explicit

Const nSkalaX = 765

Const nSkalaY = 765

Private nResistor As Integer

Private nRTotal As Double

Private nVTotal As Double

Private nITotal As Double

Private Sub cmdCalc1_Click()

Dim A As Integer

If Val(txtVolt) < 0 Or Val(txtVolt) > 255

Then

MsgBox "Nilai tegangan (v) berupa byte : 0

- 255 !", vbCritical

Exit Sub

End If

txtEks1 = "BESAR R-total dan I-total" &
vbCrLf & _
"_____"

For A = 1 To UBound(Rangkaian)

If A > 1 Then txtEks1 = txtEks1 & vbCrLf

With Rangkaian(A)

If .Hubungan = "S" Then

'Rangkaian seri

txtEks1 = txtEks1 & vbCrLf & _
A & ") R(" &

AmbilKet(.Resistor1) & ") dan " &

"R(" & AmbilKet(.Resistor2) &
") merupakan " &

"rangkaiian seri," & vbCrLf &

"sehingga perhitungan R(" & _
.Keterangan & ") adalah : "

'Perhitungan seri

.NilaiR = AmbilNilai(.Resistor1, "R")

+ AmbilNilai(.Resistor2, "R")

txtEks1 = txtEks1 & vbCrLf & _
"R(" & .Keterangan & ") = R(" &

& AmbilKet(.Resistor1) & ") " &

" + R(" & AmbilKet(.Resistor2)

& ") " & vbCrLf &

String(Len("R(" & .Keterangan

& ")"), " ") & " = " &

AmbilNilai(.Resistor1, "R") & "
+ " & AmbilNilai(.Resistor2, "R") &

vbCrLf & "R(" & .Keterangan
& ") = " & .NilaiR & " ohm"

ElseIf .Hubungan = "P" Then

'Rangkaian paralel

txtEks1 = txtEks1 & vbCrLf & _

A & ") R(" &

AmbilKet(.Resistor1) & ") dan " &

"R(" & AmbilKet(.Resistor2) &

") merupakan " &

"rangkaiian paralel," & vbCrLf

& "sehingga perhitungan R(" &

.Keterangan & ") adalah : "

'Perhitungan paralel

.NilaiR =

Bagi((AmbilNilai(.Resistor1, "R") *

AmbilNilai(.Resistor2, "R")), _
(AmbilNilai(.Resistor1, "R") +

AmbilNilai(.Resistor2, "R"))

txtEks1 = txtEks1 & vbCrLf & _
"R(" & .Keterangan & ") = (R(" &

& AmbilKet(.Resistor1) & ") " &

" * R(" & AmbilKet(.Resistor2)

& ")) / (R(" & AmbilKet(.Resistor1) &

) + R(" & AmbilKet(.Resistor2)

& ")) " & vbCrLf &

String(Len("R(" & .Keterangan

& ")"), " ") & " = (" &

AmbilNilai(.Resistor1, "R") & "

* " & AmbilNilai(.Resistor2, "R") & ") / " &

(" & AmbilNilai(.Resistor1,

"R") & " + " & AmbilNilai(.Resistor2, "R") &

") " &

vbCrLf & "R(" & .Keterangan

& ") = " & .NilaiR & " ohm"

Else

'=

.NilaiR = AmbilNilai(.Resistor1, "R")

txtEks1 = txtEks1 & vbCrLf &

A & ") R(" & .Keterangan & ")

= " & .NilaiR & " ohm"

End If

End With

Next A

nRTotal =

Rangkaian(UBound(Rangkaian)).NilaiR

nVTotal = Val(txtVolt)

```

nITotal = Bagi(nVTTotal, nRTTotal)

With Rangkaian(UBound(Rangkaian))
    .NilaiV = nVTTotal
    .Nilail = nITotal
End With

txtEks1 = txtEks1 & vbCrLf & vbCrLf & _
    "R(total) = R(" &
Rangkaian(UBound(Rangkaian)).Keterangan &
") = " & _
    nRTotal & " Ohm" & vbCrLf & _
    "V(total) = " & nVTTotal & " Volt" &
vbCrLf & _
    "sehingga I(total) = V(total) / R(total) "
& vbCrLf & _
    "          = " & nVTTotal & " / " &
nRTotal & vbCrLf & _
    "          I(total) = " & nITotal & "
Ampere"

End Sub

Private Sub cmdCalc2_Click()
    Dim A As Integer
    Dim R1 As Double
    Dim R2 As Double
    Dim V1 As Double
    Dim V2 As Double
    Dim I1 As Double
    Dim I2 As Double
    Dim Indeks As Integer
    Dim Temp1 As String
    Dim Temp2 As String

    If Val(txtVolt) < 0 Or Val(txtVolt) > 255
    Then
        MsgBox "Nilai tegangan (v) berupa byte : 0
- 255 !", vbCritical
        Exit Sub
    End If

    'Kalkulasi total V-I-R
    Call cmdCalc1_Click

    txtEks2 = ""

    For A = UBound(Rangkaian) To 1 Step -1

```

```

If Rangkaian(A).Hubungan <> "=" Then
    If A < UBound(Rangkaian) Then txtEks2
= txtEks2 & vbCrLf & vbCrLf
    End If

```

```

With Rangkaian(A)

```

```

    If .Hubungan = "S" Then

```

```

        'Rangkaian seri

```

```

        txtEks2 = txtEks2 & _

```

```

            "-----"

```

```

    & vbCrLf & _

```

```

        "R(" & .Keterangan & ") = R("

```

```

    & AmbilKet(.Resistor1) & ") " & _

```

```

        "-SERI-R(" &

```

```

    AmbilKet(.Resistor2) & ") " & vbCrLf & _

```

```

            "-----"

```

```

        R1 = AmbilNilai(.Resistor1, "R")

```

```

        R2 = AmbilNilai(.Resistor2, "R")

```

```

        V1 = Bagi(R1 * .NilaiV, .NilaiR)

```

```

        V2 = Bagi(R2 * .NilaiV, .NilaiR)

```

```

        I1 = .Nilail

```

```

        I2 = .Nilail

```

```

        txtEks2 = txtEks2 & vbCrLf &

```

```

        "Karena merupakan rangkaian seri, maka :"
```

```

        'V untuk resistor / rangkaian - I

```

```

        Temp1 = AmbilKet(.Resistor1)

```

```

        txtEks2 = txtEks2 & vbCrLf & _

```

```

            "V(" & Temp1 & ") = " & _

```

```

            "(R(" & Temp1 & ") / " & _

```

```

            "R(" & .Keterangan & ") * V("

```

```

    & .Keterangan & ") " & vbCrLf & _
    String(Len(Temp1) + 3, " ") & "

```

```

= (" & _
    R1 & " / " & .NilaiR & ") * " &

```

```

.NilaiV & vbCrLf & _
    String(Len(Temp1) + 3, " ") & "

```

```

= " & V1 & " Volt"

```

```

    If IsTVar(.Resistor1) Then

```

```

        'Rangkaian

```

```

        Rangkaian(Mid(.Resistor1, 3,

```

```

        Len(.Resistor1) - 3)).NilaiV = V1

```

```

        Rangkaian(Mid(.Resistor1, 3,

```

```

        Len(.Resistor1) - 3)).Nilail = I1

```

```

    Else

```

```

'Resistor
Indeks = AmbilIndeksR(.Resistor1)

Resistor(Indeks).NilaiV = V1
Resistor(Indeks).NilaiI = I1
End If

'V untuk resistor / rangkaian - II
Temp2 = AmbilKet(.Resistor2)
txtEks2 = txtEks2 & vbCrLf & _
    "V(" & Temp2 & ") = " & _
    "(R(" & Temp2 & ") / " & _
    "R(" & .Keterangan & ")) * V("
& .Keterangan & ")" & vbCrLf & _
    String(Len(Temp2) + 3, " ") & "
= (" & _
    R2 & "/" & .NilaiR & ") * " &
.NilaiV & vbCrLf & _
    String(Len(Temp2) + 3, " ") & "
= " & V2 & " Volt"
If IsTVar(.Resistor2) Then
    'Rangkaian
    Rangkaian(Mid(.Resistor2, 3,
Len(.Resistor2) - 3)),NilaiV = V2
    Rangkaian(Mid(.Resistor2, 3,
Len(.Resistor2) - 3)),NilaiI = I2
Else
    'Resistor
    Indeks = AmbilIndeksR(.Resistor2)

    Resistor(Indeks).NilaiV = V2
    Resistor(Indeks).NilaiI = I2
End If

'I1=I2
txtEks2 = txtEks2 & vbCrLf & _
    "I(" & Temp1 & ") = " & _
    "I(" & Temp2 & ") = " & _
    "I(" & .Keterangan & ") = " &
.NilaiI & " Ampere"

ElseIf .Hubungan = "P" Then
    'Rangkaian paralel
    txtEks2 = txtEks2 & _
        "-----"
& vbCrLf & _
    "R(" & .Keterangan & ") = R("
& AmbilKet(.Resistor1) & ")" & _
    "-PARALEL-R(" &
AmbilKet(.Resistor2) & ")" & vbCrLf & _

```

```

"-----"
R1 = AmbilNilai(.Resistor1, "R")
R2 = AmbilNilai(.Resistor2, "R")

V1 = .NilaiV
V2 = .NilaiV

I1 = Bagi(V1, R1)
I2 = Bagi(V2, R2)

txtEks2 = txtEks2 & vbCrLf &
"Karena merupakan rangkaian paralel, maka :

'V1=V2
Temp1 = AmbilKet(.Resistor1)
Temp2 = AmbilKet(.Resistor2)
txtEks2 = txtEks2 & vbCrLf & _
    "V(" & Temp1 & ") = " & _
    "V(" & Temp2 & ") = " & _
    "V(" & .Keterangan & ") = " &
.NilaiV & " Volt"

'I untuk resistor / rangkaian - I
txtEks2 = txtEks2 & vbCrLf & _
    "I(" & Temp1 & ") = " & _
    "V(" & Temp1 & ") / R(" &
Temp1 & ")" & vbCrLf & _
    String(Len(Temp1) + 3, " ") & "
= " & V1 & "/" & R1 & vbCrLf & _
    String(Len(Temp1) + 3, " ") & "
= " & I1 & " Ampere"
If IsTVar(.Resistor1) Then
    'Rangkaian
    Rangkaian(Mid(.Resistor1, 3,
Len(.Resistor1) - 3)),NilaiV = V1
    Rangkaian(Mid(.Resistor1, 3,
Len(.Resistor1) - 3)),NilaiI = I1
Else
    'Resistor
    Indeks = AmbilIndeksR(.Resistor1)

    Resistor(Indeks).NilaiV = V1
    Resistor(Indeks).NilaiI = I1
End If

'I untuk resistor / rangkaian - II
txtEks2 = txtEks2 & vbCrLf & _
    "I(" & Temp2 & ") = " & _

```



```

    "V(" & Temp2 & ") / R(" &
Temp2 & ")" & vbCrLf & _
    String(Len(Temp2) + 3, " ") & "
= " & V2 & " / " & R2 & vbCrLf & _
    String(Len(Temp2) + 3, " ") & "
= " & I2 & " Ampere"
    If IsTVar(.Resistor2) Then
        'Rangkaian
        Rangkaian(Mid(.Resistor2, 3,
Len(.Resistor2) - 3)).NilaiV = V2
        Rangkaian(Mid(.Resistor2, 3,
Len(.Resistor2) - 3)).NilaiI = I2
    Else
        'Resistor
        Indeks = AmbilIndeksR(.Resistor2)

        Resistor(Indeks).NilaiV = V2
        Resistor(Indeks).NilaiI = I2
    End If

    Else
        '=
        Indeks = AmbilIndeksR(.Resistor1)

        Resistor(Indeks).NilaiV = .NilaiV
        Resistor(Indeks).NilaiI = .NilaiI
    End If
End With
Next A
End Sub

Private Sub cmdCetak1_Click()
    On Error GoTo errPrint

    If Val(txtVolt) < 0 Or Val(txtVolt) > 255
Then
        MsgBox "Nilai tegangan (v) berupa byte : 0
- 255 !", vbCritical
        Exit Sub
    End If

    'Kalkulasi
    Call cmdCalc1_Click

    Printer.FontName = "Courier New"

    Printer.FontSize = "10"

    Printer.PaperSize = vbPRPSLetter

    Printer.Print txtEks1.Text

    Printer.EndDoc

    Exit Sub
errPrint:
    MsgBox "Printer error !", vbCritical
    Printer.EndDoc
End Sub

Private Sub Form_Load()
    Dim A As Integer

    RTBExmp.Text = strRangkaian
    Call RefreshText(RTBExmp)

    GambarRangkaian
    DoEvents

    With TabelR

```

```

Printer.Print txtEks1.Text

Printer.EndDoc

Exit Sub
errPrint:
    MsgBox "Printer error !", vbCritical
    Printer.EndDoc
End Sub

Private Sub cmdCetak2_Click()
    On Error GoTo errPrint

    If Val(txtVolt) < 0 Or Val(txtVolt) > 255
Then
        MsgBox "Nilai tegangan (v) berupa byte : 0
- 255 !", vbCritical
        Exit Sub
    End If

    'Kalkulasi
    Call cmdCalc2_Click

    Printer.FontName = "Courier New"

    Printer.FontSize = "10"

    Printer.PaperSize = vbPRPSLetter

    Printer.Print txtEks2.Text

    Printer.EndDoc

    Exit Sub
errPrint:
    MsgBox "Printer error !", vbCritical
    Printer.EndDoc
End Sub

Private Sub Form_Load()
    Dim A As Integer

    RTBExmp.Text = strRangkaian
    Call RefreshText(RTBExmp)

    GambarRangkaian
    DoEvents

    With TabelR

```



```

.Cols = 3
.ScrollBars = flexScrollBarVertical

.FixedAlignment(0) = 7
.ColAlignment(0) = 7
.ColWidth(0) = 500
.TextMatrix(0, 0) = "No."

.FixedAlignment(1) = 4
.ColAlignment(1) = 4
.ColWidth(1) = 1600
.TextMatrix(0, 1) = "Symbol Resistor"

.FixedAlignment(2) = 4
.ColAlignment(2) = 7
.ColWidth(2) = 1200
.TextMatrix(0, 2) = "Besar (ohm)"

'Looping
.Rows = UBound(Resistor) + 1
For A = 1 To UBound(Resistor)
    .TextMatrix(A, 0) = A & "."
    .TextMatrix(A, 1) = Resistor(A).Symbol
    .TextMatrix(A, 2) = "0"
Next A

End With
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Unload Me
    frmPerhitungan.Show
End Sub

'----- SCROLLING

Private Sub HScroll1_Change()
    P2.Left = CDbl(-HScroll1.Value) * 2
End Sub

Private Sub HScroll1_Scroll()
    Call HScroll1_Change
End Sub

Private Sub imgR_MouseMove(Index As Integer, Button As Integer, Shift As Integer, X As Single, Y As Single)

    With Resistor(Index)
        lblJudul = "Resistor " & .Symbol & ""

        lblTI = "- Kuat arus (I) = " & .NilaiI & "
        Ampere"
        lblTR = "- Hambatan (R) = " & .NilaiR & "
        Ohm"
        lblTV = "- Tegangan (V) = " & .NilaiV & "
        Volt"
    End With

    PicToolTips.Left = imgR(Index).Left + 300
    PicToolTips.Top = imgR(Index).Top + 450
    PicToolTips.Visible = True

    shpFocus.Left = imgR(Index).Left
    shpFocus.Top = imgR(Index).Top - 120
    shpFocus.ZOrder 0
    shpFocus.Visible = True
End Sub

Private Sub P2_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    shpFocus.Visible = False
    PicToolTips.Visible = False
End Sub

Private Sub TabelR_DblClick()
    If TabelR.Col = 2 Then
        Dim str As String
        Dim Nilai As Integer

        str = InputBox("Masukkan besar resistor ""
        & _
        TabelR.TextMatrix(TabelR.Row, 1) &
        "" (dlm ohm) :", _
        "Besar Resistor",
        TabelR.TextMatrix(TabelR.Row, 2))
        If StrPtr(str) = 0 Then Exit Sub

        Nilai = Val(str)
        If Nilai < 0 Or Nilai > 255 Then
            MsgBox "Nilai resistor berupa byte : 0 -
            255 !", vbCritical
        Else
            TabelR.TextMatrix(TabelR.Row, 2) =
            Nilai
            Resistor(TabelR.Row).NilaiR = Nilai
        End If
        TabelR.SetFocus
    End If
End Sub

```

```

Private Sub TabelR_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 And TabelR.Col = 2 Then
        Call TabelR_DbClick
    End If
End Sub

Private Sub txtVolt_KeyPress(KeyAscii As Integer)
    If KeyAscii < vbKeyBack And (KeyAscii < Asc("0") Or KeyAscii > Asc("9")) Then
        KeyAscii = 0
    End If
End Sub

Private Sub VScroll1_Change()
    P2.Top = CDBl(-VScroll1.Value) * 2
End Sub

Private Sub VScroll1_Scroll()
    Call VScroll1_Change
End Sub

Private Sub GambarRangkaian()
    Dim Object As Object
    Dim A As Long
    Dim B As Long
    Dim ArrTag() As String
    Dim PosKiriX1 As Long
    Dim PosKiriY1 As Long
    Dim PosKananX1 As Long
    Dim PosKananY1 As Long
    Dim PosKiriX2 As Long
    Dim PosKiriY2 As Long
    Dim PosKananX2 As Long
    Dim PosKananY2 As Long
    Dim PosX As Long
    Dim PosY As Long
    Dim MinY As Long
    Dim MaxY As Long
    Dim TempTag As String
    Dim Temp1 As Long
    Dim Temp2 As Long

    ReDim Resistor(0)
    nResistor = 0

    'Gambar rangkaian
    For A = 1 To UBound(Rangkaian)

```

```

        With Rangkaian(A)
            'Gambar resistor-1
            If .Hubungan < "=" And .Resistor1 < "" And IsTVar(.Resistor1) = False Then
                Call GambarResistor(.Resistor1, .Resistor1)
            End If

            'Gambar resistor-2
            If .Hubungan < "=" And .Resistor2 < "" And IsTVar(.Resistor2) = False Then
                Call GambarResistor(.Resistor2, .Resistor2)
            End If

            If .Hubungan = "S" Then
                'RANGKAIAN SERI

                'Cari Obj Resistor-1 & simpan posisi
                For Each Object In Me
                    If Object.Tag < "" Then
                        If TypeName(Object) = "Image"
                            Then
                                'Bagi ke array tag
                                ArrTag = Split(Object.Tag, ",")
                                If ArrTag(0) = .Resistor1 Then
                                    PosKiriX1 = ConvX(Val(ArrTag(1)), False)
                                    PosKiriY1 = ConvY(Val(ArrTag(2)), False)
                                    PosKananX1 = ConvX(Val(ArrTag(3)), False)
                                    PosKananY1 = ConvY(Val(ArrTag(4)), False)
                                Exit For
                            End If
                        End If
                    End If
                Next

                'Cari Obj Resistor-2 & simpan posisi
                For Each Object In Me
                    If Object.Tag < "" Then
                        If TypeName(Object) = "Image"
                            Then
                                'Bagi ke array tag
                                ArrTag = Split(Object.Tag, ",")
                                If ArrTag(0) = .Resistor2 Then

```

```

        PosKiriX2 =
ConvX(Val(ArrTag(1)), False)
        PosKiriY2 =
ConvY(Val(ArrTag(2)), False)
        PosKananX2 =
ConvX(Val(ArrTag(3)), False)
        PosKananY2 =
ConvY(Val(ArrTag(4)), False)
        Exit For
    End If
End If
End If
Next

'Geser Resistor-2 di samping Resistor-
1
    PosX = PosKananX1 - PosKiriX2 - 1
    PosY = PosKananY1 - PosKiriY2
    TempTag = .Nama & "," &
ConvX(PosKiriX1) & "," & ConvY(PosKiriY1)
& _
        "," & ConvX(PosKananX2 +
PosX) & "," & ConvY(PosKananY2 + PosY)
    PosKananX2 = PosKananX2 + PosX
    PosKananY2 = PosKananY2 + PosY
    For Each Object In Me
        If Object.Tag <> "" Then
            'Bagi ke array tag
            ArrTag = Split(Object.Tag, ",")
            If ArrTag(0) = .Resistor1 Then
                'Resistor-1
                Object.Tag = TempTag
            Elseif ArrTag(0) = .Resistor2
Then
                'Resistor-2
                If TypeName(Object) =
"Image" Then
                    Object.Left = Object.Left +
ConvX(PosX)
                    Object.Top = Object.Top +
ConvY(PosY)
                Elseif TypeName(Object) =
"Line" Then
                    Object.X1 = Object.X1 +
ConvX(PosX)
                    Object.X2 = Object.X2 +
ConvX(PosX)
                    Object.Y1 = Object.Y1 +
ConvY(PosY)

```

```

        Object.Y2 = Object.Y2 +
ConvY(PosY)
    End If
    Object.Tag = TempTag
    End If
    End If
Next

'Cari yang paling atas -> atau y paling
kecil
'jika terdapat dibawah 1, maka geser
ke bawah
    MinY = -9999
    For Each Object In Me
        If Object.Tag <> "" Then
            If TypeName(Object) = "Image"
Then
                'Bagi ke array tag
                ArrTag = Split(Object.Tag, ",")
                If ArrTag(0) = .Nama Then
                    If MinY = -9999 Or
(ConvRY(Object.Top, False) < MinY) Then
                        MinY =
ConvRY(Object.Top, False)
                    End If
                End If
            End If
        End If
    Next

    PosY = 1 - MinY
    If PosY <> 0 Then
        TempTag = .Nama & "," &
ConvX(PosKiriX1) & "," & ConvY(PosKiriY1
+ PosY) & _
            "," & ConvX(PosKananX2)
& "," & ConvY(PosKananY2 + PosY)
        For Each Object In Me
            If Object.Tag <> "" Then
                'Bagi ke array tag
                ArrTag = Split(Object.Tag, ",")
                If ArrTag(0) = .Nama Then
                    If TypeName(Object) =
"Image" Then
                        Object.Top = Object.Top
+ ConvY(PosY)
                    Elseif TypeName(Object) =
"Line" Then
                        Object.Y1 = Object.Y1 +
ConvY(PosY)

```

```

        Object.Y2 = Object.Y2 +
ConvY(PosY)
        End If
        Object.Tag = TempTag
    End If
End If
Next
End If

Elseif .Hubungan = "P" Then

    RANGKAIAN PARALEL

    'Y Maksimum dari resistor - 1
    MaxY = 0
    For Each Object In Me
        If Object.Tag <> "" Then
            If TypeName(Object) = "Image"
Then
                'Bagi ke array tag
                ArrTag = Split(Object.Tag, ",")
                If ArrTag(0) = .Resistor1 Then
                    If (ConvRY(Object.Top,
False) > MaxY) Then
                        MaxY =
ConvRY(Object.Top, False)
                    End If
                End If
            End If
        End If
    Next

    'Y Minimum dari resistor - 2
    MinY = -9999
    For Each Object In Me
        If Object.Tag <> "" Then
            If TypeName(Object) = "Image"
Then
                'Bagi ke array tag
                ArrTag = Split(Object.Tag, ",")
                If ArrTag(0) = .Resistor2 Then
                    If MinY = -9999 Or
(ConvRY(Object.Top, False) < MinY) Then
                        MinY =
ConvRY(Object.Top, False)
                    End If
                End If
            End If
        End If
    Next

```

```

        'Geser Resistor-2 ke sebelah kanan &
        bawah
        ' Resistor-1 ke sebelah kanan
        PosKiriX1 = 0: PosKiriX2 = 0
        PosY = (MaxY - MinY) + 2
        For Each Object In Me
            If Object.Tag <> "" Then
                'Bagi ke array tag
                ArrTag = Split(Object.Tag, ",")
                If ArrTag(0) = .Resistor1 Then
                    TempTag = .Resistor1 & "," &
Val(ArrTag(1)) + ConvX(1) & _
                    "," & ArrTag(2) & _
                    "," & Val(ArrTag(3)) +
ConvX(1) & _
                    "," & ArrTag(4)
                'Isi posisi
                If PosKiriX1 = 0 Then
                    PosKiriX1 = Val(ArrTag(1))
                    PosKiriY1 = Val(ArrTag(2))
                    PosKananX1 =
Val(ArrTag(3)) + ConvX(1)
                    PosKananY1 =
Val(ArrTag(4))
                End If
                If TypeName(Object) =
"Image" Then
                    Object.Left = Object.Left +
ConvX(1)
                ElseIf TypeName(Object) =
"Line" Then
                    Object.X1 = Object.X1 +
ConvX(1)
                    Object.X2 = Object.X2 +
ConvX(1)
                End If
                Object.Tag = TempTag
            ElseIf ArrTag(0) = .Resistor2
Then
                TempTag = .Resistor2 & "," &
Val(ArrTag(1)) + ConvX(1) & _
                "," & Val(ArrTag(2)) +
ConvY(PosY) & _
                "," & Val(ArrTag(3)) +
ConvX(1) & _
                "," & Val(ArrTag(4)) +
ConvY(PosY)
                'Isi posisi

```

```

        If PosKiriX2 = 0 Then
            PosKiriX2 = Val(ArrTag(1))
        + ConvX(1)
            PosKiriY2 = Val(ArrTag(2))
        + ConvY(PosY)
            PosKananX2 =
Val(ArrTag(3)) + ConvX(1)
            PosKananY2 =
Val(ArrTag(4)) + ConvY(PosY)
        End If
        If TypeName(Object) =
"Image" Then
            Object.Top = Object.Top +
ConvY(PosY)
            Object.Left = Object.Left +
ConvX(1)
        "Line" Then
            Elseif TypeName(Object) =
"Line" Then
            Object.X1 = Object.X1 +
ConvX(1)
            Object.X2 = Object.X2 +
ConvX(1)
            Object.Y1 = Object.Y1 +
ConvY(PosY)
            Object.Y2 = Object.Y2 +
ConvY(PosY)
        End If
        Object.Tag = TempTag
    End If
Next

'Ambil selisih panjang
PosX = (PosKananX1 - PosKiriX1) -
(PosKananX2 - PosKiriX2)
    If PosX < 0 Then
        If PosX > 0 Then
            'PosX > 0 - R1 lebih panjang
            'Tambah panjang R2

            TempTag = .Resistor2 & _
                "," & PosKiriX2 & _
                "," & PosKiriY2 & _
                "," & PosKananX2 + PosX
            & _
                "," & PosKananY2
            Temp2 = PosX
            PosX = PosX / 2

            'Geser sebesar posX

```

```

For Each Object In Me
    If Object.Tag <> "" Then
        'Bagi ke array tag
        ArrTag = Split(Object.Tag,
";")
        If ArrTag(0) = .Resistor2
            Then
                If TypeName(Object) =
"Image" Then
                    Object.Left =
Object.Left + PosX
                "Line" Then
                    Object.X1 = Object.X1
                    + PosX
                    Object.X2 = Object.X2
                    + PosX
                End If
                Object.Tag = TempTag
            End If
        End If
    Next

    'Garis horizontal-1
    Temp1 = LH.UBound + 1
    Load LH(Temp1)
    With LH(Temp1)
        .X1 = PosKiriX2: .X2 =
PosKiriX2 + PosX
        .Y1 = PosKiriY2: .Y2 =
PosKiriY2
        .Visible = True
        .Tag = TempTag
    End With

    'Garis horizontal-2
    Temp1 = LH.UBound + 1
    Load LH(Temp1)
    With LH(Temp1)
        .X1 = PosKananX2 + PosX:
.X2 = PosKananX2 + (2 * PosX)
        .Y1 = PosKananY2: .Y2 =
PosKananY2
        .Visible = True
        .Tag = TempTag
    End With
    PosKananX2 = PosKananX2 +
Temp2
Else

```

```

'PosX < 0 - R2 lebih panjang
'Tambah panjang R1

TempTag = .Resistor1 & _
          "," & PosKiriX1 & _
          "," & PosKiriY1 & _
          "," & PosKananX1 +
Abs(PosX) & _
          "," & PosKananY1
Temp2 = Abs(PosX)
PosX = Abs(PosX) / 2

'Geser sebesar posX
For Each Object In Me
  If Object.Tag <> "" Then
    'Bagi ke array tag
    ArrTag = Split(Object.Tag,
";")
    If ArrTag(0) = .Resistor1
      If TypeName(Object) =
"Image" Then
        Object.Left =
Object.Left + PosX
      Elseif TypeName(Object)
= "Line" Then
        Object.X1 = Object.X1
        + PosX
        Object.X2 = Object.X2
        + PosX
      End If
      Object.Tag = TempTag
    End If
  End If
Next

'Garis horizontal-1
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
  .X1 = PosKiriX1: .X2 =
PosKiriX1 + PosX
  .Y1 = PosKiriY1: .Y2 =
PosKiriY1
  .Visible = True
  .Tag = TempTag
End With

'Garis horizontal-2

```

```

Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
  .X1 = PosKananX1 + PosX:
.X2 = PosKananX1 + (2 * PosX)
  .Y1 = PosKananY1: .Y2 =
PosKananY1
  .Visible = True
  .Tag = TempTag
End With
PosKananX1 = PosKananX1 +
Temp2
End If
End If

'Selisih-Y
Temp2 = ConvY((PosKiriY2 -
PosKiriY1) \ 2, False)

'Temporary Tag
TempTag = .Nama & _
          "," & ConvX(1) & _
          "," & PosKiriY1 +
ConvY(Temp2) & _
          "," & PosKananX1 + ConvX(1)
& _
          "," & PosKananY1 +
ConvY(Temp2)

'Garis vertikal-1
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
  .X1 = PosKiriX1: .X2 = PosKiriX2
  .Y1 = PosKiriY1: .Y2 = PosKiriY2
  .Visible = True
  .Tag = TempTag
End With

'Garis vertikal-2
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
  .X1 = PosKananX1: .X2 =
PosKananX2
  .Y1 = PosKananY1: .Y2 =
PosKananY2
  .Visible = True
  .Tag = TempTag
End With

```

```

'Garis horizontal - 1
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
.X1 = ConvX(1): .X2 = ConvX(2)
.Y1 = PosKiriY1 + ConvY(Temp2)
.Y2 = PosKiriY1 + ConvY(Temp2)
.Visible = True
.Tag = TempTag
End With

'Garis horizontal - 2
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
.X1 = PosKananX1: .X2 =
PosKananX1 + ConvX(1)
.Y1 = PosKiriY1 + ConvY(Temp2)
.Y2 = PosKiriY1 + ConvY(Temp2)
.Visible = True
.Tag = TempTag
End With

'Rename
For Each Object In Me
If Object.Tag <> "" Then
'Bagi ke array tag
ArrTag = Split(Object.Tag, ",")
If ArrTag(0) = .Resistor1 Or _
ArrTag(0) = .Resistor2 Then
Object.Tag = TempTag
End If
End If
Next
Else
'RESISTOR TUNGGAL
Call GambarResistor(.Nama,
.Resistor1)
End If

End With
Next A

'Susun posisi label
For A = 1 To lblR.UBound
lblR(A).Top = imgR(A).Top - 120
lblR(A).Left = imgR(A).Left + 315
lblR(A).ZOrder 0
Next A

```

```

'Gambar voltase
TempTag = imgR(1).Tag
ArrTag = Split(TempTag, ",")
PosKiriX1 = Val(ArrTag(1))
PosKiriY1 = Val(ArrTag(2))
PosKananX1 = Val(ArrTag(3))
PosKananY1 = Val(ArrTag(4))

'Y Maksimum
MaxY = 0
For Each Object In Me
If Object.Tag <> "" Then
If TypeName(Object) = "Image" Then
'Bagi ke array tag
ArrTag = Split(Object.Tag, ",")
If (Object.Top > MaxY) Then
MaxY = Object.Top
End If
End If
End If
Next
MaxY = ConvRY(MaxY, False)
MaxY = MaxY + 2
MaxY = ConvRY(MaxY)

'Garis vertikal-1
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
.BorderColor = &H80&
.X1 = PosKiriX1: .X2 = PosKiriX1
.Y1 = PosKiriY1: .Y2 = MaxY
.Visible = True
.Tag = "VOLT"
End With

'Garis vertikal-1
Temp1 = LV.UBound + 1
Load LV(Temp1)
With LV(Temp1)
.BorderColor = &H80&
.X1 = PosKananX1: .X2 = PosKananX1
.Y1 = PosKananY1: .Y2 = MaxY
.Visible = True
.Tag = "VOLT"
End With

'Garis horizontal-1
Temp1 = LH.UBound + 1

```



```

Load LH(Temp1)
With LH(Temp1)
    .BorderColor = &H80&
    .X1 = PosKiriX1: .X2 = PosKananX1
    .Y1 = MaxY: .Y2 = MaxY
    .Visible = True
    .Tag = "VOLT"
End With

```

```

imgV.Top = MaxY - 450
imgV.Left = nSkalaX + (PosKananX1 -
PosKiriX1 - imgV.Width) \ 2

```

```

lblV.Top = imgV.Top - 225
lblV.Left = imgV.Left
End Sub

```

```

Private Sub GambarResistor(pSymbol As String,
pLabel As String)
    Dim i As Integer
    Dim J As Integer
    Dim Temp1 As Integer
    Dim Temp2 As String

```

```

    Temp2 = pSymbol & "," & ConvX(1) & "," &
ConvY(1) &
    ", " & ConvX(4) & "," & ConvY(1)

```

```

'Garis horizontal-1
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
    .X1 = ConvX(1): .X2 = ConvX(2)
    .Y1 = ConvY(1): .Y2 = ConvY(1)
    .Visible = True
    .Tag = Temp2
End With

```

```

'Tambah array resistor
nResistor = nResistor + 1
ReDim Preserve Resistor(nResistor)
With Resistor(nResistor)
    .Indeks = imgR.UBound + 1
    .Symbol = pLabel
End With

```

```

'Gambar resistor
Temp1 = imgR.UBound + 1
Load imgR(Temp1)
With imgR(Temp1)

```

```

    .Left = ConvX(2)
    .Top = ConvRY(1)
    .Visible = True
    .Tag = Temp2
End With

```

```

'Label resistor
Temp1 = lblR.UBound + 1
Load lblR(Temp1)
With lblR(Temp1)
    .Caption = pLabel
    .Visible = True
End With

```

```

'Garis horizontal-2
Temp1 = LH.UBound + 1
Load LH(Temp1)
With LH(Temp1)
    .X1 = ConvX(3): .X2 = ConvX(4)
    .Y1 = ConvY(1): .Y2 = ConvY(1)
    .Visible = True
    .Tag = Temp2
End With
End Sub

```

```

Private Function ConvX(ByVal pX As Long,
Optional pbConv As Boolean = True) As Long
    If pbConv Then
        ConvX = pX * nSkalaX
    Else
        ConvX = pX / nSkalaX
    End If
End Function

```

```

Private Function ConvY(ByVal pY As Long,
Optional pbConv As Boolean = True) As Long
    If pbConv Then
        ConvY = pY * nSkalaY
    Else
        ConvY = pY / nSkalaY
    End If
End Function

```

```

Private Function ConvRY(ByVal pY As Long,
Optional pbConv As Boolean = True) As Long
    If pbConv Then
        ConvRY = (pY * nSkalaY) - 360
    Else
        ConvRY = (pY + 360) / nSkalaY
    End If

```



```

End Function

'Format Text
Private Sub RefreshText(poRTB As
RichTextBox)
    Dim i As Integer
    Dim nP As Integer

    On Error Resume Next

    With poRTB
        nP = .SelStart
        .Visible = False
        For i = 1 To Len(.Text)
            If Asc(Mid(.Text, i, 1)) = 34 Or
Mid(.Text, i, 1) = "I" Then
                .SelStart = i - 1
                .SelLength = 1
                .SetFontName = "Wingdings 3"
                .FontSize = 11
                .SetColor = &HFF&
                .SelBold = True
            Else
                .SelStart = i - 1
                .SelLength = 1
                .SetFontName = "Times New Roman"
                .FontSize = 11
                .SetColor = 0
                .SelBold = True
            End If
        Next i
        DoEvents: DoEvents
        .Visible = True
    End With
End Sub

Private Function AmbilKet(pSymbol As String)
As String

    If IsTVar(pSymbol) Then
        AmbilKet = Rangkaian(Mid(pSymbol, 3,
Len(pSymbol) - 3)).Keterangan
    Else
        AmbilKet = pSymbol
    End If

End Function

Private Function AmbilNilai(pSymbol As
String, pTipe As String) As Double

```

```

    If IsTVar(pSymbol) Then
        If pTipe = "R" Then
            AmbilNilai = Rangkaian(Mid(pSymbol,
3, Len(pSymbol) - 3)).NilaiR
        ElseIf pTipe = "I" Then
            AmbilNilai = Rangkaian(Mid(pSymbol,
3, Len(pSymbol) - 3)).NilaiI
        ElseIf pTipe = "V" Then
            AmbilNilai = Rangkaian(Mid(pSymbol,
3, Len(pSymbol) - 3)).NilaiV
        End If
    Else
        Dim Indeks As Integer

        Indeks = AmbilIndeksR(pSymbol)

        If pTipe = "R" Then
            AmbilNilai = Resistor(Indeks).NilaiR
        ElseIf pTipe = "I" Then
            AmbilNilai = Resistor(Indeks).NilaiI
        ElseIf pTipe = "V" Then
            AmbilNilai = Resistor(Indeks).NilaiV
        End If
    End If

```

End Function

```

Private Function AmbilIndeksR(pSymbol As
String) As Integer
    Dim i As Integer

    For i = 1 To UBound(Resistor)
        If Resistor(i).Symbol = pSymbol Then
            AmbilIndeksR = i
            Exit For
        End If
    Next i
End Function

```

```

Private Function Bagi(ByVal A1 As Double,
ByVal A2 As Double) As Double
    If A2 = 0 Then
        Bagi = 0
    Else
        Bagi = Round(A1 / A2, 2)
    End If
End Function

```

- **Form Teori Arus**

```
Option Explicit
Dim Hal As Byte
Private Sub CmdBack_Click()
    If Hal = 2 Then
        Hal = 1
        TxtTeori1.LoadFile (App.Path &
"\Arus1.rtf")
        TxtTeori1.Height = 5000
        SF1.Visible = True
        SF1.Movie = App.Path & "\Arus.swf"
        SF1.Menu = False
        SF1.Play
        CmdNext.Enabled = True
        CmdBack.Enabled = False
    End If
End Sub

Private Sub CmdNext_Click()
    If Hal = 1 Then
        TxtTeori1.LoadFile (App.Path &
"\Arus2.rtf")
        TxtTeori1.Height = 8500
        SF1.Visible = False
        Hal = Hal + 1
        CmdNext.Enabled = False
        CmdBack.Enabled = True
    End If
End Sub

Private Sub Form_Load()
    Hal = 1
    TxtTeori1.LoadFile (App.Path & "\Arus1.rtf")
    TxtTeori1.Height = 5000
    SF1.Movie = App.Path & "\Arus.swf"
    SF1.Menu = False
    SF1.Play
    CmdNext.Enabled = True
    CmdBack.Enabled = False
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
    FrmMenuUtama.Show
End Sub
```

- **Form Teori Hambatan**

```
Option Explicit
Dim Hal As Byte
Private Sub CmdBack_Click()
    If Hal = 2 Then
```

```
        Hal = 1
        TxtTeori1.LoadFile (App.Path &
"\Hambatan1.rtf")
        TxtTeori1.Height = 4500
        SF1.Visible = True
        SF1.Movie = App.Path & "\Hambatan.swf"
        SF1.Menu = False
        SF1.Play
        CmdNext.Enabled = True
        CmdBack.Enabled = False
    End If
End Sub
```

```
Private Sub CmdNext_Click()
    If Hal = 1 Then
        TxtTeori1.LoadFile (App.Path &
"\Hambatan2.rtf")
        TxtTeori1.Height = 8500
        SF1.Visible = False
        Hal = Hal + 1
        CmdNext.Enabled = False
        CmdBack.Enabled = True
    End If
End Sub
```

```
Private Sub Form_Load()
```

```
    Hal = 1
    TxtTeori1.LoadFile (App.Path &
"\Hambatan1.rtf")
    TxtTeori1.Height = 4500
    SF1.Movie = App.Path & "\Hambatan.swf"
    SF1.Menu = False
    SF1.Play
    CmdNext.Enabled = True
    CmdBack.Enabled = False
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
    FrmMenuUtama.Show
End Sub
```

- **Form Teori Tegangan**

```
Option Explicit
Dim Hal As Byte
Private Sub CmdBack_Click()
    If Hal = 2 Then
        Hal = 1
        TxtTeori1.LoadFile (App.Path &
"\Tegangan1.rtf")
        TxtTeori1.Height = 9500
        SF1.Visible = True
        SF1.Movie = App.Path & "\Tegangan.swf"
```

```

SF1.Menu = False
SF1.Play
CmdNext.Enabled = True
CmdBack.Enabled = False
End If
End Sub

```

```

Private Sub CmdNext_Click()
If Hal = 1 Then
    TxtTeori1.LoadFile (App.Path &
"\Tegangan2.rtf")
    TxtTeori1.Height = 8500
    SF1.Visible = False
    Hal = Hal + 1
    CmdNext.Enabled = False
    CmdBack.Enabled = True
End If
End Sub

```

```

Private Sub Form_Load()
    Hal = 1
    TxtTeori1.LoadFile (App.Path &
"\Tegangan1.rtf")
    TxtTeori1.Height = 9500
    SF1.Visible = True
    SF1.Movie = App.Path & "\Tegangan.swf"
    SF1.Menu = False
    SF1.Play
    CmdNext.Enabled = True
    CmdBack.Enabled = False
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
    FrmMenuUtama.Show
End Sub

```

• Form Petunjuk

```

Private Sub Form_Load()
    Text.LoadFile (App.Path &
"\PETUNJUK.rtf")
End Sub

```

```

Private Sub RichTextBox1_Change()

```

```

End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
    FrmMenuUtama.Show
End Sub

```

• Form Profil

```

Private Sub Form_Unload(Cancel As Integer)
    FrmMenuUtama.Show
End Sub

```