

FACULDADE PIO DÉCIMO
COORDENAÇÃO DO CURSO DE GRADUAÇÃO EM
ENGENHARIA ELÉTRICA

MONOGRAFIA

MODELAGENS DE LINHAS DE TRANSMISSÃO
UTILIZANDO APLICAÇÕES COMPUTACIONAIS DO
VISUAL BASIC

RAPHAEL MENDONÇA BRÜGGER

Aracaju – Brasil

Dezembro de 2008

FACULDADE PIO DÉCIMO
COORDENAÇÃO DO CURSO DE GRADUAÇÃO EM
ENGENHARIA ELÉTRICA

**MODELAGENS DE LINHAS DE TRANSMISSÃO
UTILIZANDO APLICAÇÕES COMPUTACIONAIS DO
VISUAL BASIC**

Monografia apresentada à Coordenação do Curso de Engenharia Elétrica, em cumprimento ao pré-requisito para a obtenção do grau de graduado em Engenharia Elétrica.

RAPHAEL MENDONÇA BRÜGGER

Prof. José Valter Alves Santos

Aracaju – Brasil

Dezembro de 2008

FACULDADE PIO DÉCIMO
COORDENAÇÃO DO CURSO DE GRADUAÇÃO EM
ENGENHARIA ELÉTRICA

RAPHAEL MENDONÇA BRÜGGER

MODELAGENS DE LINHAS DE TRANSMISSÃO UTILIZANDO
APLICAÇÕES COMPUTACIONAIS DO VISUAL BASIC

Monografia aprovada 12/12/2008

Prof. José Valter Alves Santos, M.Sc
Presidente da Banca Examinadora

Prof. Dárcio Hersch Gomes de Souza Sá, M.Sc
Componente da Banca Examinadora

Prof^a. Teresinha Maria dos Santos
Componente da Banca Examinadora

Aracaju – Brasil

Dezembro de 2008

AGRADECIMENTOS

Agradeço a Deus, por ter dado muitas forças e coragem para me dedicar aos estudos após uma jornada de trabalho, pela vida, saúde e caminhada.

A Nossa Senhora pela proteção maternal nessa difícil jornada.

Aos meus pais que, apesar das dificuldades, estiveram dispostos a investir na educação.

A toda minha família que contribuiu de forma direta ou indireta para mais uma etapa da minha vida.

Aos meus colegas que de certa forma contribuíram para meu sucesso e que torceram por mim.

A Faculdade Pio Décimo, que propiciou caminho para conclusão da minha formação acadêmica.

A empresa Aperipê, que subsidiou parte dos estudos.

A todos os demais professores que nos orientaram em todos os períodos, dedico esta conquista com a mais profunda admiração e respeito.

SUMÁRIO

LISTA DE FIGURAS.....	viii
LISTA DE TABELAS	ix
RESUMO	x
1 – CAPÍTULO – INTRODUÇÃO	01
1.0 – Sistemas elétricos de potência	01
1.1 – Introdução.....	01
1.2 – O valor econômico da energia elétrica.....	02
1.3 – O custo da falta de suprimento de energia elétrica.....	03
1.4 – A qualidade no fornecimento de energia elétrica	04
1.5 – Representações dos Sistemas de Potência.....	06
1.5.1 – Diagrama unifilar.....	06
1.6 – O sistema elétrico de potência e sua proteção	08
1.7 – Divisão do sistema elétrico de potência	10
1.8 – Causa dos defeitos em um SEP	11
1.9 – Efeitos da falta em um sistema elétrico de potência.....	11
2 – CAPÍTULO – LINHAS DE TRANSMISSÃO.....	13
2.0 – Sistemas elétricos de potência	13
2.1 – Introdução	13
2.2 – Função das linhas de transmissão	15
2.3 – A necessidade da proteção de linhas de transmissão	17
2.4 – Linha de transmissão curta.....	18
2.5 -- Linha de transmissão média	21
2.6 – Linha de transmissão longa.....	23
2.6.1 -- Linha de transmissão longa: interpretação das equações	26
2.7 – Linha de transmissão longa:Forma hiperbólica das equações	26
2.8 – Indutância nas linhas de transmissão	28
2.8.1 -- Transposição de condutores	29
2.8.2 – Linhas com vários condutores por fase.....	30

2.8.3 -- Sistema de transmissão de Itaipu	32
2.9 -- Capacitância nas linhas de transmissão.....	35
2.9.1 – Capacitância de uma linha a dois fios.....	35
2.9.2 – Capacitância de uma linha trifásica com espaçamento equilátero.....	39
2.9.3 – Cabos múltiplos	40
2.10 – Cabos condutores em linhas aéreas de transmissão.....	41
2.11 -- Isoladores e ferragens em linhas aéreas de transmissão.....	43
2.11.1 – Isoladores	43
2.11.2 – Ferragens.....	47
2.12 – Vibrações por carga de gelo nos condutores.....	50
2.13 – Estruturas de linhas aéreas de transmissão	51
2.14 – Tensão mecânica e flecha de um condutor	54
3 – CAPÍTULO – VISUAL BASIC.....	56
3.0 – Visual Basic.....	56
3.1 – Introdução	56
3.2 – Requisitos do Sistema.....	57
3.3 – As ferramentas do Visual Studio	58
3.4 – Desenvolvimento	60
3.4.1 – A janela Properties.....	60
3.5 – Movendo e redimensionando as ferramentas de programação	60
4 – CAPÍTULO – DESENVOLVIMENTO DO PROGRAMA MODELAGEM LINHAS DE TRANSMISSÃO	63
4.0 – Desenvolvimento do programa modelagem de linhas de transmissão	63
4.1 -- Desenvolvendo o programa - Isoladores.....	63
4.2 -- Desenvolvendo o programa – Tensão mecânica e flecha de um condutor	66
4.3 – Desenvolvendo o programa – Indutância em linhas de transmissão - Linha trifásica para um circuito simples	69
4.4 – Desenvolvendo o programa – Indutância em linhas de transmissão - Linha múltipla	72
4.5 – Desenvolvendo o programa – Indutância em linhas de transmissão – Linha trifásica de circuito duplo	77

4.6 -- Desenvolvendo programa – Capacitância em linhas de transmissão – Susceptância capacitiva em uma linha monofásica	81
4.7 – Desenvolvendo programa – Capacitância em linhas de transmissão – Linha trifásica de um circuito simples	85
4.8 – Desenvolvendo programa – Capacitância em linhas de transmissão – Linha múltipla	93
4.9 – Desenvolvendo programa – Capacitância em linhas de transmissão – Linha trifásica de circuito duplo	95
4.10 – Desenvolvendo programa – Linhas de transmissão curta.....	102
4.11 – Desenvolvendo programa – Linhas de transmissão média	105
4.12 – Desenvolvendo programa – Linhas de transmissão longa.....	113
5.0 – REFERÊNCIAS BIBLIOGRÁFICAS.....	122

LISTA DE FIGURAS

Figura 1.0: Tipos de ligações do SEP	07
Figura 1.1: Símbolos utilizados na representação do sistema.....	08
Figura 2.0: Circuito equivalente de uma linha de transmissão curta	18
Figura 2.1: Diagramas fasoriais de uma linha de transmissão curta.....	20
Figura 2.2: Modelo T da linha de transmissão.....	21
Figura 2.3: Modelo π da linha de transmissão	21
Figura 2.4: Diagrama simplificado de uma linha de transmissão longa	23
Figura 2.5: Transposição de linha trifásica	29
Figura 2.6: Linha trifásica com arranjo linear e com dois condutores por fase.....	32
Figura 2.7: Seção reta de um circuito trifásico do sistema de transmissão de Itaipu	33
Figura 2.8: Cabos encordoados	34
Figura 2.9: Seção transversal de uma linha de fios paralelos	36
Figura 2.10: Seção transversal de uma linha com espaçamento equilátero	39
Figura 2.11: Seção transversal de uma linha trifásica de cabos múltiplos.....	40
Figura 2.12: Isoladores de pino.....	44
Figura 2.13: Tensão de descarga de impulso	45
Figura 2.14: Cadeia de isoladores tipo suspensão	46
Figura 2.15: Cadeia de suspensão simples 230kV	48
Figura 2.16: Cadeia de ancoragem.....	48
Figura 2.17: Espaçadores	49
Figura 2.18: Disposição triangular.....	52
Figura 2.19: Partes de uma torre	53
Figura 2.20: Catenária.....	54
Figura 3.0: Painel Start do Visual Basic	57
Figura 3.1: Ferramentas e janelas do Visual Studio	59

LISTA DE TABELAS

Tabela 1.0: Valores dos custos da falhas obtidos em pesquisas	04
Tabela 2.0: Classe de tensão para uso no Brasil	14
Tabela 2.1: Dados do Sistema de Transmissão Brasileiro.....	16
Tabela 2.2: Ocorrência de faltas no SEP em um sistema de 500 kV, num período de 10 anos	18
Tabela 2.3: Característica dos cabos ACRS	34
Tabela 2.4: Características dos principais condutores	38
Tabela 2.5: Propriedades de materiais condutores.....	41
Tabela 2.6: (Ontário Hydro)	50

Resumo

Sistema elétrico de potência é um conjunto de equipamentos destinados a gerar, transmitir e distribuir a energia elétrica. A disposição de suas redes pode ser classificada como em anel ou radial. Como a definição nos diz, nesse tipo de sistema todas as usinas geradoras e as linhas de transmissão estão interligadas, no Brasil, mais de 95% do sistema está ligado em anel, o chamado Sistema Interligado Nacional. Em análise, iremos aplicar um software chamado Visual Basic e é nele que iremos fazer modelagem matemática de sistema de geração e transmissão de energia elétrica com o objetivo de simular as equações de um sistema de potência através da programação.

Abstract

Electrical system of power is a set of destined equipment to generate, to transmit and to distribute the electric energy. The disposal of its nets can be classified as in radial ring or. As the definition in says them, in this type of system all the generating plants and the lines of transmission are linked, in Brazil, more than 95% of the system are on in ring, the call *National Linked System*. In analysis, we will go to apply a called software Visual beginner's all-purpose symbolic instruction code and is in it that we will go to make mathematical modeling of generation system and transmission of electric energy with the objective to simulate the equations of a system of power through the programming.

Capítulo 1

1.0 Sistemas elétricos de potência

1.1 Introdução

Sistema elétricos de potência (SEP) conjunto de equipamentos que operam em conjunto e de maneira coordenada de forma a gerar, transmitir e fornecer energia elétrica aos consumidores, mantendo o melhor padrão de qualidade possível. E nela consiste Equipamentos {geradores, transformadores, linhas de transmissão, disjuntores, pára-raios, relés, medidores etc. E no padrão de qualidade existem alguns requisitos básicos a serem satisfeitos pelas empresas concessionárias de energia elétrica com relação ao fornecimento aos consumidores como os níveis de tensão devem estar dentro de uma faixa especificada, assim como a frequência também, o serviço não deve sofrer interrupções (na prática: o serviço deve sofrer o mínimo número de interrupções, e estas devem durar o menor tempo possível); a energia deve ser entregue ao consumidor com o mínimo custo (geração econômica, transmissão com mínima perda, etc.).

Dentre os diversos tipos de distúrbios que podem vir a se manifestar em um sistema elétrico, destacam-se as distorções harmônicas. As componentes harmônicas, por definição, ocorrem em um estado estacionário, e são múltiplos inteiros da frequência fundamental do sistema, estando presentes continuamente ou, no mínimo, por alguns segundos. Tais distúrbios estão associados à operação contínua de cargas com características

não-lineares inclusas no sistema, como por exemplo, do uso de inversores, conversores CA e CC, fornos elétricos, lâmpadas fluorescentes e computadores, entre outros.

1.2 O valor econômico da energia elétrica

A dependência da sociedade moderna em relação ao fornecimento de energia, em suas diversas formas, seja para garantir a competitividade da nação em relação a mercados existentes e globalizados, ou seja, para manter ou mesmo elevar o padrão de vida das populações, salienta a necessidade do uso mais racional e efetivo dos recursos energéticos, principalmente os não renováveis.

Esta dependência energética é satisfeita pelas fontes energéticas tradicionais como os combustíveis fósseis tipo petróleo, carvão e gás, assim como pelas fontes energéticas renováveis tais como a energia hidrelétrica, a biomassa, a energia solar, eólica, etc. A energia elétrica, objetivo deste estudo, torna-se disponível aos consumidores após sua extração, transformação e transporte.

O setor energético é parte integrante da macroeconomia (relacionando o setor de energia com o resto da economia), estruturando-se e se integrando aos diversos setores da economia, enfatizando a necessidade de um consumo racional da energia de modo a maximizar o uso da energia elétrica.

A questão econômica da energia elétrica é usualmente estudada com o auxílio de modelos. A modelagem dos sistemas de energia pode ser entendida como a tarefa de formular modelos, direta ou indiretamente associados com o processo de tomada de decisão.

O sistema elétrico compreende centenas de equipamentos interligados entre si e se desenvolvem por extensas áreas territoriais. Estes sistemas são planejados, construídos e operados de modo a atender os tipos de cargas mais variados.

1.3 O custo da falta de suprimento de energia elétrica

Como todo mercado econômico, a produção e posterior venda da energia (produto) também apresenta um balanço entre o custo de produção e manutenção da qualidade pelo produtor.

Os custos associados à interrupção no fornecimento são associados a uma falha no abastecimento de energia demandada pelo usuário. Há também o custo econômico associado a baixos níveis de qualidade de energia ou seja, suprimento de uma energia sujas, com flutuações e harmônicas.

A não restauração do fornecimento energético ou a restauração deste após a ocorrência de uma falta em um grande espaço de tempo, aumentando o tempo não-operativos do sistema também representa um custo econômico da energia.

A qualidade no fornecimento da energia elétrica torna-se, pois, fator de importância capital na luta por maior espaços numa economia globalizada, esta qualidade é função dos seguintes atributos:

- Disponibilidade no fornecimento de energia na quantidade exigida pelo usuário,
- Suprimento de uma energia limpa, com um mínimo de flutuações no nível de tensão e a presença de ruídos,
- Rápida restauração após algum falha no sistema visando a minimização do tempo não operativo,
- A acomodação do sistema a mudanças, planejadas ou acidentais, em sua estrutura topológica, como a retirada de elementos através de manobras na rede

elétrica.

Geralmente os custos de falhas no suprimento de energia elétricos são pagos pelo consumidor. No setor industrial, por exemplo, destacam-se:

- Danos aos equipamentos/instalações;
- Danos à matéria prima/produto final;
- Custo de reinício da produção (após interrupção);
- Perda da produção (durante a falha e/ou reinício);
- Horas extras pagas para recuperar a produção;
- Outros custos e efeitos (ex. Aquisição de equipamentos de emergência).

Admite-se que o corte no suprimento de energia elétrico diminui o PIB

(Produto Interno Bruto) do país.

País	Consumo	Custos de falhas (US\$/KWH)
Reino Unido	Residencial	1,02
Suécia	Residencial	1,46
Brasil	Residencial	1,95 – 3,00
Estados Unidos	Industrial	2,54
Reino Unido	Industrial	3,04
Brasil	Industrial	1,50 – 7,12
Suécia	Industrial	1,44 – 2,96
Estados Unidos	Comercial	4,99
Reino Unido	Comercial	5,65

Tabela 1.0 – Valores dos custos da falhas obtidos em pesquisas.

Assim, a necessidade crescente de energia elétrica pela sociedade moderna e os altos custos provenientes do não suprimento desta energia, como mostra a tabela acima, colocam as concessionárias de energia diante da necessidade de aumentar a oferta bem como zelar pelo bom fornecimento desta.

1.4 A qualidade no fornecimento de energia elétrica

Entende-se por qualidade no fornecimento de energia elétrica a disponibilidade, ou seja, o fornecimento ininterrupto de energia na qualidade desejada pelo usuário e a conformidade, ou seja, as concessionárias devem buscar o fornecimento de uma energia limpa, praticamente isenta de flutuações e de harmônicas, com forma de onda senoidal.

Esta tarefa é dificultada pela variedade de cargas ligadas à rede elétrica, ocasionando flutuações nas tensões e distorções na forma senoidal pura. Neste sentido, a concessionária deve monitorar continuamente os distúrbios, ajudando os consumidores a ligar corretamente seus equipamentos e, sobretudo, agir para que os problemas de um usuário não se propagem via rede elétrica.

Os principais distúrbios que podem surgir no fornecimento de energia elétrica são:

- Surtos de tensão,
- Baixa de tensão,
- Distorções harmônicas,
- Modulação de baixa frequência (flicker).

A qualidade do fornecimento de energia elétrica está intimamente ligado a restaurabilidade deste, que é a capacidade dos sistemas de energia elétrica de rapidamente restaurar o fornecimento de energia, minimizando os tempos de saída de operação.

Um sistema elétrico que fornece energia de qualidade deve possuir as seguintes características:

- Curto intervalo de tempo de indisponibilidade, com alto índice de

disponibilidade.

- A capacidade de retornar ao estado normal de operação após uma falha no sistema.
- A capacidade de manter-se funcionando por mais tempo, inibindo as falhas devido ao desgaste dos equipamentos.

Porém, a qualidade de energia não se resume a entregar ao cliente uma energia com os atributos mencionados anteriormente.

1.5 Representação dos Sistemas de Potência

O comportamento de um Sistema Elétrico de Potência (SEP) deve ser acompanhado sistematicamente e analisado frente às suas contingências e alterações a fim de que um diagnóstico correto dos efeitos seja feito e medidas corretas sejam adotadas. Para isso, o sistema elétrico deve ser criteriosamente representado através de uma modelagem adequada ao tipo de estudo a ser realizado.

1.5.1 Diagrama unifilar

Um SEP típico é formado por várias estações geradoras conectadas através de linhas de transmissão a grandes centros de carga, onde a potência é distribuída aos consumidores pelo sistema de distribuição, formado por linhas de distribuição e transformadores. Assim, é adotada uma simplificação onde os componentes do sistema são representados por símbolos simples denominada **diagrama unifilar**. Em um diagrama unifilar, o sistema trifásico é representado por um sistema monofásico (uma das três fases e o

neutro). Frequentemente este diagrama é ainda mais simplificado, suprimindo-se o neutro e indicando as partes componentes por símbolos padronizados.

A importância do diagrama unifilar é fornecer de maneira concisa os dados mais significativos de um sistema de potência bem como sua topologia.

As informações contidas num diagrama unifilar variam de acordo com o problema a ser estudado. Por exemplo, no estudo da proteção de um sistema a informação da localização dos relés e disjuntores no circuito é muito importante bem como os valores das correntes de curto-circuito que deverão ser calculadas.

Os componentes de um sistema de potência trifásico que são representadas em um diagrama unifilar são:

- Máquinas síncronas;
- Transformadores;
- Linhas de transmissão;
- Cargas estáticas ou dinâmicas.

O diagrama unifilar também deve incluir:

- Informações sobre cargas;
- Valores nominais dos geradores, trafos e reatâncias dos diversos

componentes.

É importante conhecer também os pontos onde o sistema é ligado a terra. A

figura 1.0 mostra alguns tipos de ligações.

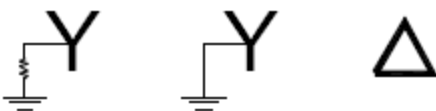


Figura 1.0 – Tipos de ligações do SEP.

Alguns dos símbolos utilizados nesta representação foram normalizados pela *American National Standards Institute* (ANSI) e pelo *Institute of Electrical and Electronic Engineers* (IEEE) e são mostrados na **figura 1.1**.

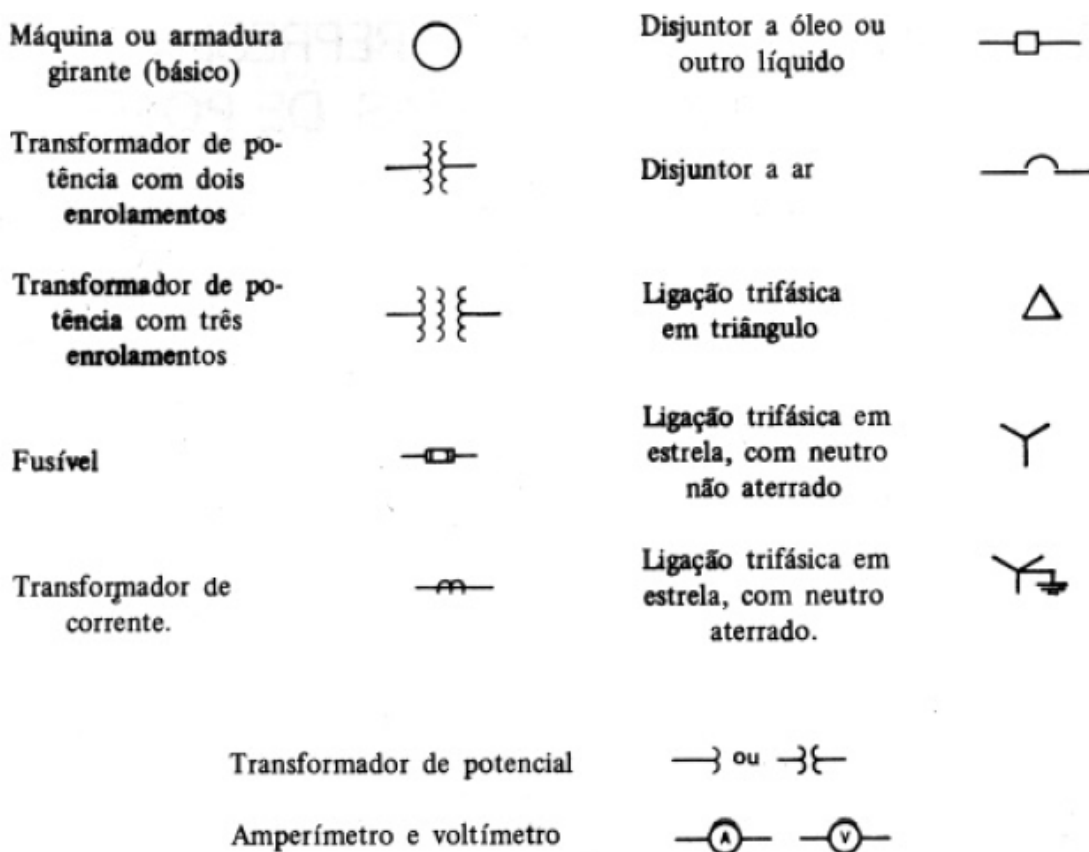


Figura 1.1 - Símbolos utilizados na representação do sistema

1.5.2 O Gerador

Os geradores são responsáveis pela geração de energia em corrente alternada no sistema elétrico e suprem a energia solicitada pelas cargas em um SEP, mantendo os níveis de tensão dentro de uma faixa estreita, e garantindo a continuidade e a estabilidade do sistema.

Normalmente, as tensões nos terminais de um gerador são senoidais, conforme mostra a equação (1):

$$E = E_M \cdot \cos.(wt + \Theta)$$

onde Θ representa o ângulo inicial da bobina do rotor.

A figura 1.2 representa as principais partes de um gerador; o estator e o rotor.

Quando o rotor gira de maneira síncrona, as voltagens geradas nos enrolamentos do estator (defasadas 120° entre si) são representadas no gráfico 1.3.

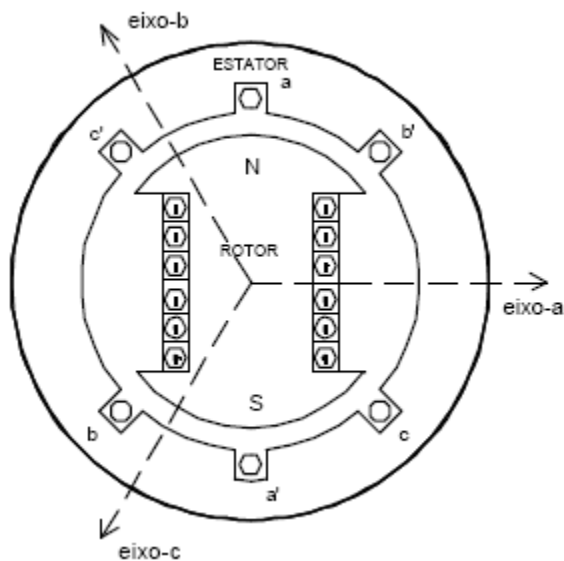


Figura 1.2 – Representação do gerador

Defini-se como sistema de tensões trifásicas e simétricas (3 fases) um sistema de tensões representado matematicamente pelo conjunto de equações (2):

$$\begin{aligned} e_{cc'} &= \sqrt{2} \cdot E \cdot \cos(wt + 120) \\ e_{bb'} &= \sqrt{2} \cdot E \cdot (\cos wt - 120) \\ e_{aa'} &= \sqrt{2} \cdot E \cdot \cos wt \end{aligned}$$

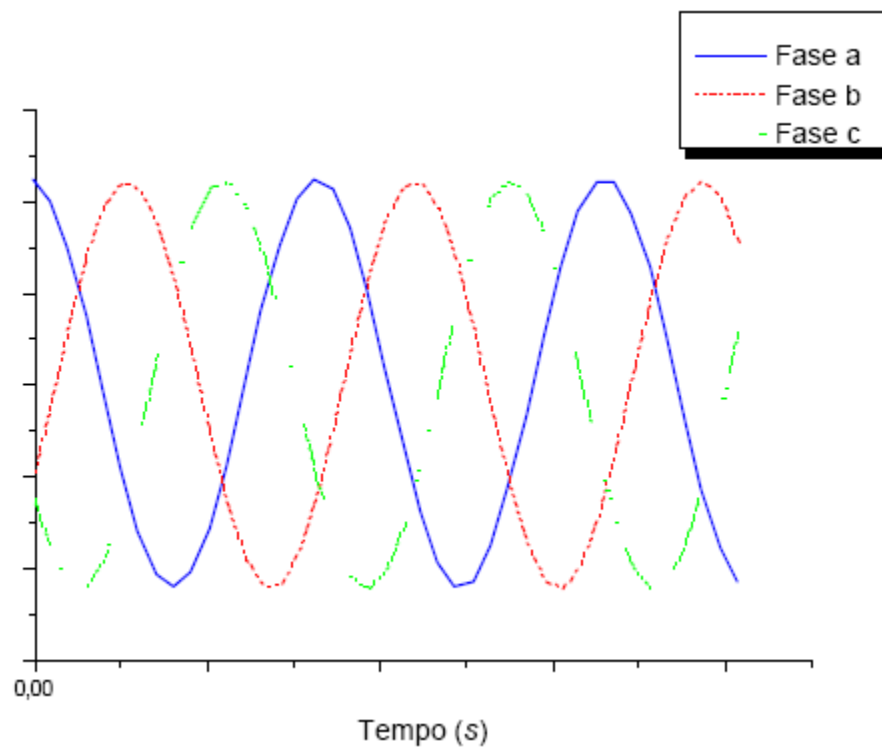


Gráfico 1.3 – Representação de ondas geradas num sistema trifásico.

As tensões e correntes nos sistemas trifásicos são representados por vetores girantes que giram no sentido anti-horário com velocidade angular ωt . Podemos então representar o sistema trifásico pelos vetores girantes na figura 1.4.

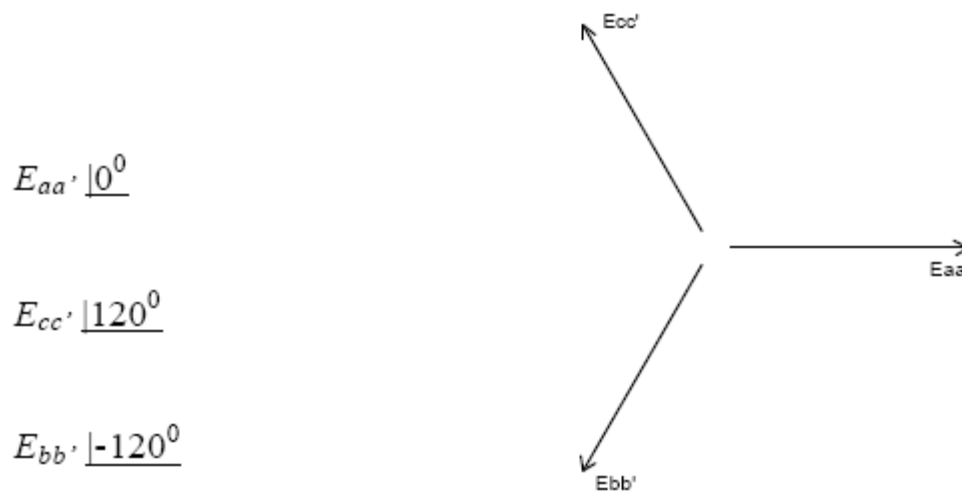


Figura 1.4 - Fasores tensão geradas trifásicas

Definimos, para um sistema polifásico simétrico, a seqüência de fases como sendo a ordem pela qual as tensões das fases passam pelo seu valor máximo. O exemplo em questão constitui um conjunto trifásico balanceado de seqüência positiva *abc*.

1.5.3 O transformador

O transformador possibilita a conexão de vários equipamentos elétricos com tensões elétricas distintas, podendo ser abaixador ou elevador de tensão. Em um SEP ele é responsável pela elevação do nível de tensão para transmissão de energia elétrica.

Considera-se um transformador com um enrolamento primário de N_p espiras e um enrolamento secundário de N_s espiras, como mostrado esquematicamente na figura 1.5.

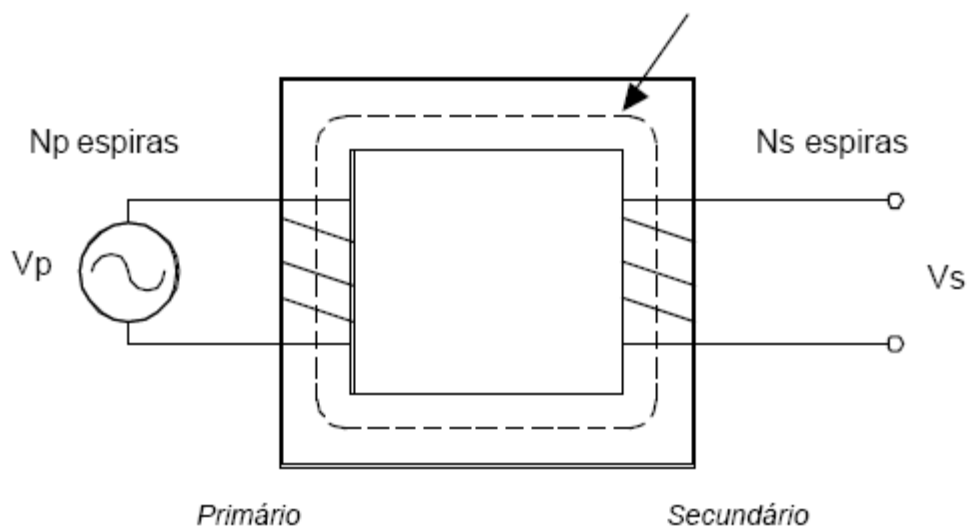


Figura 1.5 - Esquema de um transformador monofásico

Onde: N_P e N_S = número de espiras do primário e secundário, respectivamente;

v_p e v_s = tensão no primário e secundário, respectivamente;

e_p e e_s = tensão induzida no primário e secundário, respectivamente;

r_p e r_s resistência da bobina no primário e secundário, respectivamente;

Θ_{L1} e Θ_{L2} Fluxo induzido;

Θ indutância própria.

Seu princípio de operação é o da indução mútua entre duas bobinas enroladas no mesmo núcleo, que produz uma força eletromotriz (*fem*) induzida pela variação de fluxo por espira que é a mesma tanto no primário como no secundário. Assim:

$$\frac{V_p}{V_s} = \frac{N_p}{N_s}$$

(Transformador ideal)

Numa situação ideal, pode-se pressupor que não haverá perda de potência, ou seja, as propriedades deste transformador são ideais no sentido de que as resistências de enrolamento são desprezíveis, todo o fluxo está confinado no núcleo e se concatenam com ambas os enrolamentos e as perdas no núcleo são desprezíveis. Tais propriedades são aproximadas, mas nunca realmente atingidas em transformadores reais. Um transformador hipotético tendo estas propriedades é freqüentemente chamado de transformador ideal.

(Potência aparente do primário) $I_p \cdot V_p = I_s \cdot V_s$ (Potência aparente do secundário)

Note que a potência de entrada é igual à potência de saída (uma condição decorrente de se ter desprezado todas as causas de perdas de potência ativa e reativa no transformador).

Assim:

$$\frac{V_p}{V_s} = \frac{I_s}{I_p} = \frac{N_p}{N_s}$$

Então, um transformador ideal transforma as tensões na relação direta do número de espiras dos respectivos enrolamentos. Assim:

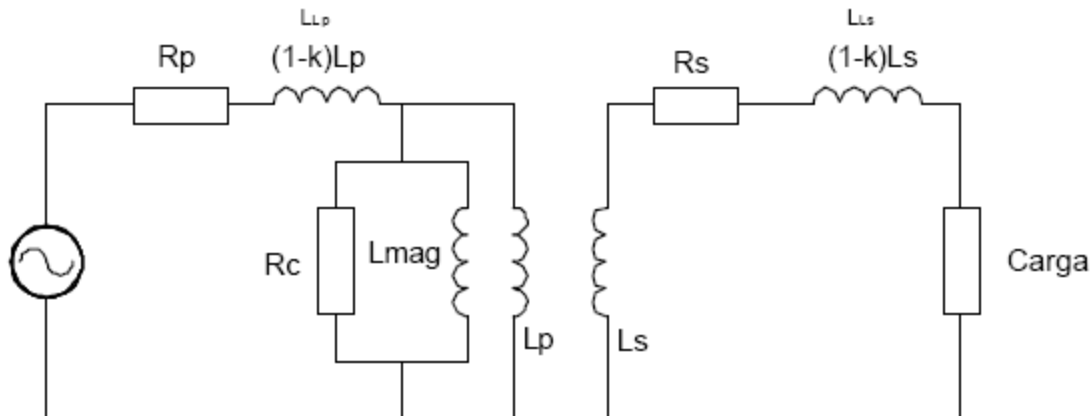
$$I_p \cdot N_p = N_s \cdot V_s$$

E então:

$$\frac{I_p}{I_s} = \frac{N_s}{N_p} \text{ (O sinal de menos elementos indica corrente primária e secundária em sentido oposto)}$$

Este é o modo pelo qual o primário toma conhecimento da presença de corrente no secundário. Assim, um transformador ideal transforma as correntes na razão inversa do número de espiras nos respectivos enrolamentos.

Num transformador real, devem ser considerados os efeitos da resistência dos enrolamentos, da dispersão magnética e da corrente de excitação. Às vezes, a indutância dos enrolamentos também tem efeitos importantes, caracterizando perdas e fluxos de dispersão (perdas no cobre e no ferro). A figura 1.6 mostra o circuito equivalente de um transformador real.



Trafo

Figura 1.6 - Circuito equivalente com perdas no cobre e no ferro

OBS:

- 1 - Transformadores reais \Rightarrow possuem perdas e fluxo de dispersão (perdas no cobre e no ferro).
- 2 - Transformador real \Rightarrow transformador ideal + representação das perdas.

1.6 O sistema elétrico de potência e sua proteção

Com tamanho e características que permitem considerá-lo único em âmbito mundial, o sistema elétrico brasileiro se caracteriza por ser um sistema hidrotérmico de grande porte com forte predominância de usinas hidroelétricas, ou seja, a energia elétrica pode ser gerada através de fontes renováveis de energia, como por exemplo, a força das águas, ou através de fontes não renováveis, como por exemplo, combustíveis fósseis, carvão, etc. No Brasil, a opção hidráulica é a mais utilizada e apenas uma pequena parte da energia consumida é gerada a partir de fontes não renováveis, em usinas térmicas.

Em períodos de condições hidrológicas desfavoráveis, as usinas térmicas contribuem para o atendimento ao mercado como um todo. A exploração coordenada dos recursos hidro e termoeletrônicos permite que se maximize a disponibilidade e a confiabilidade do suprimento de energia com redução dos custos para os consumidores. Assim, a participação complementar das usinas térmicas no mercado consumidor exige interconexão e interligação.

O sistema hidrotérmico brasileiro é formado por dois grandes sistemas, interligados através de linhas de transmissão, um reunindo as empresas das regiões Sul, Sudeste e Centro-Oeste e outro as concessionárias da região Nordeste e parte da região Norte.

Estes sistemas foram unidos por uma linha de transmissão, a Interligação Norte-Sul, passando a formar um único sistema interligado de âmbito nacional. Segundo a ONS – Operadora Nacional de Sistemas Elétricos, atualmente, apenas 3,4% da capacidade de produção de eletricidade do país encontra-se fora desse sistema, em pequenos sistemas isolados, localizados principalmente na região amazônica.

A opção pela intensa utilização do potencial hidroelétrico do país determinou as características singulares do sistema. As usinas desse tipo são construídas onde melhor se podem aproveitar as aflúências e os desníveis dos rios, muitas vezes em locais distantes dos centros consumidores. Conseqüentemente, para atender ao mercado, foi necessário desenvolver um extenso sistema de transmissão, em que as linhas de transmissão criam uma complexa rede de caminhos alternativos para escoar, com segurança, toda a energia produzida nas usinas até os centros de consumo.

O sistema de transmissão forma caminhos alternativos que permite transportar com segurança a energia produzida até os centros consumidores. Mais ainda, as grandes interligações possibilitam a troca de energia entre regiões, permitindo obter benefícios a partir da diversidade de comportamento das vazões entre rios de diferentes bacias hidrográficas.

O sistema é operado de forma coordenada, visando obter ganhos a partir da interação entre as bacias. A operação coordenada visa minimizar os custos globais de produção de energia elétrica e aumentando a confiabilidade do fornecimento de energia ao mercado consumidor através da interdependência operativa entre as usinas.

Entende-se por interdependência operativa o aproveitamento conjunto dos recursos hidroelétricos, através da construção e da operação de usinas e reservatórios localizados em seqüência em várias bacias hidrográficas. Desta forma a operação de uma determinada usina depende das vazões liberadas a montante por outras usinas que podem ser de outras empresas.

1.7 Divisão do sistema elétrico de potência

O sistema elétrico de potência é tradicionalmente dividido, para efeito dos estudos de planejamento, em três componentes principais: as usinas geradoras, o sistema de

transmissão, o sistema de subtransmissão e o sistema de distribuição. Essa divisão é necessária devido ao grau de desagregação do mercado consumidor que cada uma dessas redes enxerga: a subtransmissão vê um mercado mais desagregado que a transmissão. Como o sistema de transmissão está em constante crescimento, torna-se difícil a caracterização das fronteiras existentes entre uma rede de transmissão e a de subtransmissão. Assim, pode-se caracterizar a primeira como sendo aquela que possui tensão igual ou superior a 230 kV e a segunda como sendo a que engloba as tensões de 69 a 138 kV. Tal classificação não é rígida, porque a tensão de 138 kV também pode ser enquadrada como sendo de transmissão. Isso ocorre porque há linhas de transmissão em 138 kV que são importantes para dar continuidade de fluxo na eventualidade de contingências em linhas de tensão superior paralelas a elas.

O sistema de subtransmissão é a continuidade do sistema de transmissão e tem a finalidade de transmitir energia às pequenas cidades ou grupamentos de cidades, ao interior de grandes centros urbanos e a consumidores industriais de grande porte. Esse sistema reparte espacialmente, entre as subestações de distribuição, a energia recebida em grosso de subestações de transmissão.

1.8 Causa dos defeitos em um SEP

Oscilações causadas por distúrbios no sistema envolvem variações súbitas de tensão ou corrente nos sistemas elétricos, na grande maioria dos casos, inicialmente em regime permanente. Essas variações são provocadas por descargas atmosféricas, faltas no sistema ou operação de disjuntores.

- Ar - curto-circuito por aves, roedores, galhos de árvores, TC, rigidez.

- Dielétrica afetada por frio o calor
- Isoladores de porcelana - curto-circuitos ou rachaduras
- Isolação de trafos e geradores afetados pela umidade
- Descargas atmosféricas
- Surtos de chaveamento

Como consequência, a propagação destes distúrbios ao longo das linhas de transmissão afeta a análise do sistema.

1.9 Efeitos da falta em um sistema elétrico de potência

Entende-se por falta em linhas de transmissão como sendo uma falha total ou parcial na continuidade do fornecimento de energia elétrica. A ocorrência de uma falta pode ser um fenômeno interno ou externo ao sistema, isto é, sobretensões no sistema oriundas de quebra de isolador, raios, sobrecargas nos equipamentos, aumento repentino de carga, perda de grandes blocos de carga ou perda de geração.

Na ocorrência destes problemas, podem surgir:

- Danos ao sistema devido aos efeitos dinâmicos e térmicos da corrente de falta;
- Descontinuidade do sistema;
- Perda de sincronismo;
- Redução das margens de estabilidade do sistema;
- Danos aos equipamentos;

- Desligamento de áreas que não estão sob falta, produzindo um efeito conhecido como efeito cascata.
- Explosões;

A principal causa de faltas em linhas de transmissão são as descargas atmosféricas que podem ocorrer em pontos aleatórios do sistema. Se a falta não for eliminada rapidamente, os danos aos equipamentos que integram o sistema poderão ser elevados.

No gráfico 1.7 e no gráfico 1.8 são apresentadas formas de onda de tensão e corrente no terminal P, resultados de uma simulação de uma falta, onde as distorções provocadas pela inserção da falta podem ser notadas.

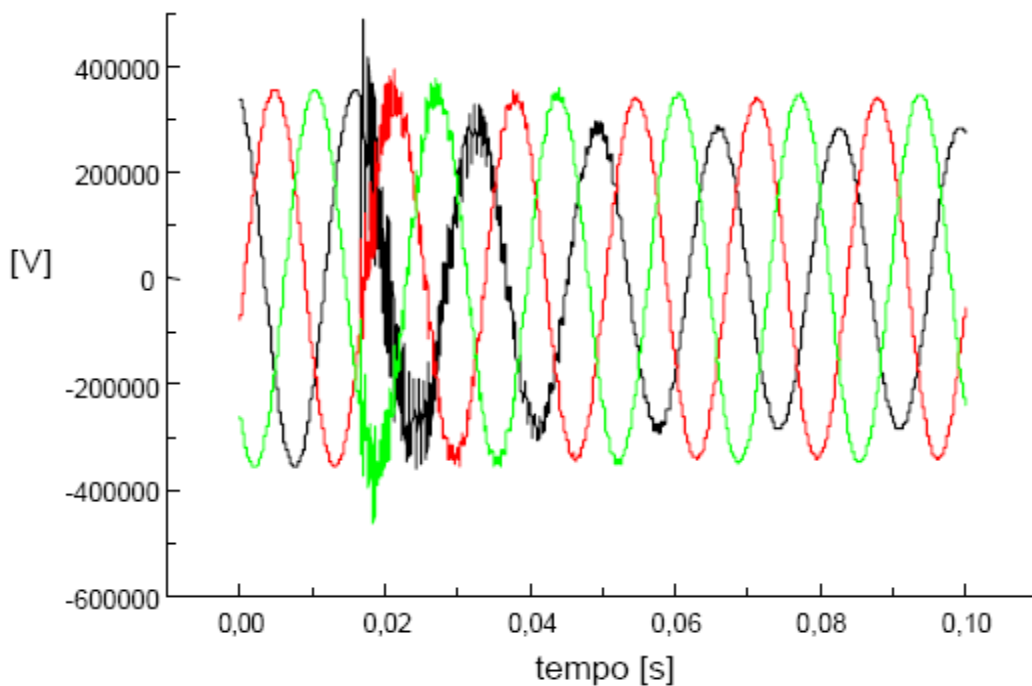


Gráfico 1.7 - Forma de onda de tensão.

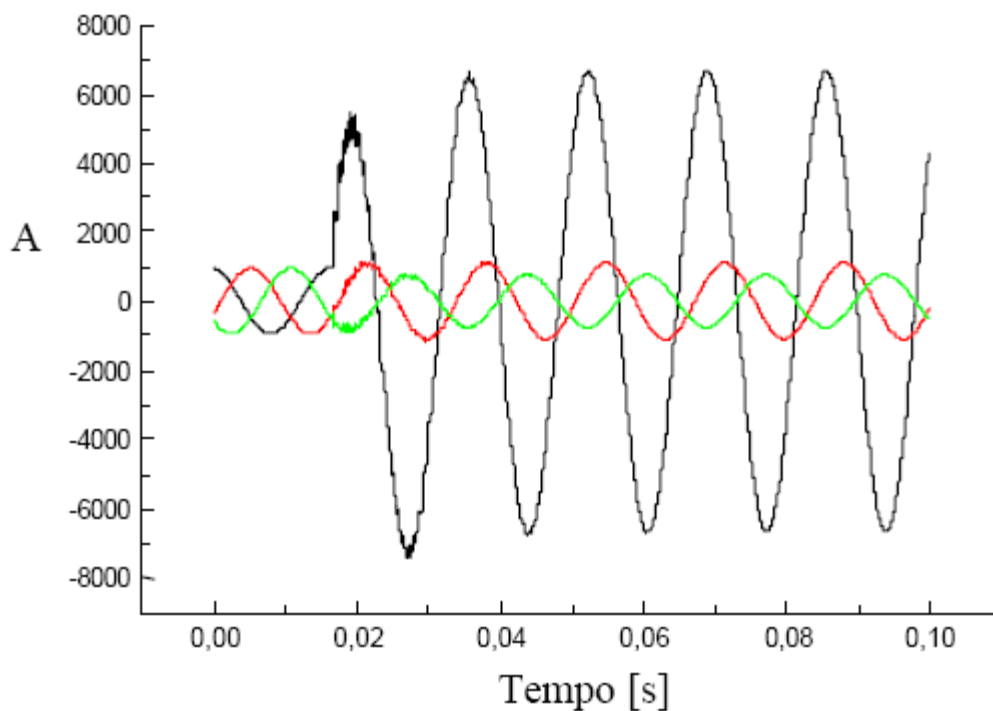


Gráfico 1.8 - Forma de onda de corrente.

As faltas podem ser do tipo permanente ou transitória. As faltas permanentes, como o próprio nome já indica, são do tipo irreversíveis, ou seja, após a abertura do disjuntor, a continuidade do fornecimento de energia não poderá ser restabelecido. As faltas temporárias são aquelas que ocorrem sem haver danos físicos ao sistema, ou seja, após a atuação da proteção, o sistema poderá ser restabelecido sem problemas.

Capítulo 2

2.0 Linhas de transmissão

2.1 Introdução

As primeiras aplicações de caráter econômico de energia elétrica datam de

1870, época em que as máquinas (dínamos e motores de correntes contínuas), atingiram o estágio que permitiu seu uso na geração e na utilização da energia elétrica como força motriz em indústrias e nos transportes. Foi por volta de 1883, que foi construída a primeira linha de transmissão do Brasil, na cidade de Diamantina em Minas Gerais, tinha por fim transportar a energia produzida em uma usina hidrelétrica, constituída por duas rodas d'água e dois dínamos Grame, a uma distancia de 2km aproximadamente. A energia transportada acionava bombas hidráulicas em uma mina de diamantes, consta que era a mais longa do mundo na época.

Em 1901, com a entrada em serviço da central hidrelétrica de Santana do Parnaíba, construiu as primeiras linhas de seus sistemas de 40kV. Em 1914, com a entrada em serviço da usina hidrelétrica de Itupararanga a mesma empresa introduziu o padrão 88kV, que até hoje mantém e que adotou também para Subtransmissão. Entre 1945 e 1947, foi construída a primeira linha de 230 kV no Brasil, com um comprimento aproximado de 330km, destinada a interligar os sistemas Rio Light e São Paulo Light, operando inicialmente em 170kV, passando em 1950, a operar com 230kV, foi a primeira interligação importante realizada no Brasil.

Seguiram-se, a partir daí, em rápida sucessão, as linhas de 345kV da CEMIG e Furnas e 460kV da CESP, as linhas de 500kV do sistema de Furnas e 800kV do sistema de Itaipu.

No Brasil são as seguintes as classes de altas tensões e extra-altas tensões recomendadas pelo COBEI da ABNT, para sistemas trifásicos, tensões fase-a-fase.

Tensões nominais	Tensões máximas	Categoria
33 ou 34,5 kV	38 kV	Altas tensões
62 ou 69 kV	72,5 kV	

132 ou 138 kV	145 kV	Tensões extras elevadas
220 ou 230 kV	242 kV	
330 ou 345 kV	362 kV	
500 kV	550 kV	
750 kV	800 kV	

Tabela 2.0 – Classe de tensão para uso no Brasil.

Uma linha de transmissão compõe-se das seguintes partes:

- Cabos condutores de energia e acessórios;
- Estruturas isolantes;
- Fundações;
- Cabos de guarda ou pára-raios;
- Aterramentos;
- Acessórios diversos;

No Brasil opera uma linha de $\pm 600\text{kV}$, 3150MW e aproximadamente 800km de extensão, interligando o setor de 50Hz de Itaipu com o sistema interligado do sudeste, que opera em 60Hz.

As linhas de transmissão podem ser:

- Linhas Curtas – comprimento de até 80Km;
- Linhas Médias – comprimento entre 80 e 240Km;
- Linhas Longas – comprimento acima de 240Km.

2.2 Função das linhas de transmissão

A energia elétrica é de suma importância na sociedade industrial moderna, e a única maneira de transportar esta energia, sob forma de eletricidade, é utilizando linhas de transmissão. Como as linhas de transmissão não podem armazenar energia elétrica gerada nas usinas, toda ela é convertida simultaneamente em carga, excetuando as perdas do sistema. Portanto, entende-se por transmissão, a transferência de energia através de linhas, entre um centro gerador e um centro consumidor.

Depois de gerada, a energia elétrica é conduzida por cabos até uma subestação elevadora, onde transformadores elevam o valor da tensão elétrica. Assim, nesse nível de tensão, a eletricidade pode percorrer longas distâncias pelas linhas de transmissão, sustentadas por torres, até chegarem nas proximidades de onde será consumida, onde é reduzida novamente em subestações abaixadoras. Finalmente a energia elétrica é transformada para os padrões de consumo local aonde chega às residências, indústrias, etc.

A necessidade de um grande número de linhas de transmissão é justificada pelo fato de que:

- Os centros produtores de energia, que são no Brasil, em sua maioria, usinas hidroelétricas, encontram-se afastados dos grandes centros consumidores, caracterizando assim, a necessidade do sistema de transmissão;
- Um sistema de potência bem projetado compreende um grande número de estações geradoras interligadas através de linhas de transmissão que, em função da otimização dos recursos energéticos, aproveita a diversidade hidrológica das regiões, aportando algo em torno de 20% a mais de geração.

Extensão das linhas de transmissão - 2000	
230 kV	32.582,2 km
345 kV	9.023,5 km
440 kV	6.162,5 km
500 kV	17.657,5 km
600 kV	1.612,0 km
750 kV	2.379,0 km
TOTAL	69.416,6 km

Tabela 2.1 – Dados do Sistema de Transmissão Brasileiro

Ao se constituir um sistema de transmissão em malha, aumenta-se a confiabilidade do sistema elétrico, minimizando-se o seu custo total, além disso, propicia-se a otimização dos recursos energéticos primários ao se utilizar, para atendimento do mercado consumidor, a fonte primária que possui menor custo ou energia armazenada nos reservatórios, característica dos sistemas hidráulicos com grande capacidade de armazenamento.

O sistema de transmissão, levando em consideração a aleatoriedade da disponibilidade dos seus componentes e consumo, a capacidade de produção e os recursos energéticos primários, tem como função:

- a distribuição espacial em grosso da energia gerada pelas usinas aos grandes centros consumidores e a alimentação de eventuais consumidores de grande porte, em termos de investimento global do setor elétrico, o ideal seria alocar as usinas geradoras o mais próximo possível dos centros de carga, no entanto, devido ao fato do potencial hidráulico só poder ser explorado onde ele está disponível e às restrições ambientais para a localização de

usinas térmicas, esta condição é de difícil cumprimento, originando esta função primordial das linhas de transmissão;

- a interligação das usinas geradoras, bacias hidráulicas e regiões de características hidrológicas heterogêneas visando atender os desequilíbrios regionais entre produção e consumo;

- a integração energética com os países vizinhos, assumindo assim uma importância econômica crescente. A interconexão elétrica internacional assume importância crescente, seja por benefício interno aos setores elétricos dos países, ou seja, pela importância como instrumento de integração econômica.

Essas interligações elétricas propiciam a compra e a venda de energia, a otimização do uso dos recursos de geração e o aumento da confiabilidade dos sistemas elétricos.

2.3 A necessidade da proteção de linhas de transmissão

A interligação do sistema elétrico de potência trouxe, além de vantagens econômicas, novos problemas para o sistema como um todo. Em sistemas interligados, as perturbações causadas por uma falta podem se estender a todo o sistema, pois a corrente que circula durante um curto-circuito é aumentada, obrigando a instalação de um sistema de proteção de maior capacidade.

Pela própria natureza do sistema elétrico de potência, o elemento mais vulnerável a falhas é a linha de transmissão, especialmente se for considerada sua dimensão física, visto que ela fica exposta a toda sorte e risco como intempéries, descargas atmosféricas entre outros.

A Tabela 2.2 apresenta uma idéia da ordem de ocorrência das falhas em um sistema elétrico, levando-se em consideração às suas próprias características³. Pode ser observado que cerca de 80% das interrupções acidentais no fornecimento de energia são originados nas linhas de transmissão, ou provocados por elas.

Setor do sistema elétrico	Número de faltas
Linhas de transmissão	82
Circuitos disjuntores	4
Autotransformadores	6
Barramentos	1
Geradores	1
Falha humana	5

Tabela 2.2 – Ocorrência de faltas no SEP em um sistema de 500 kV, num período de 10 anos.

2.4 Linha de transmissão curta

O circuito equivalente de uma linha de transmissão curta é mostrado na Figura 2.0, onde I_S e I_R são, as correntes na barra transmissora e receptora respectivamente e V_S e V_R são as tensão ao neutro nas mesmas barras.

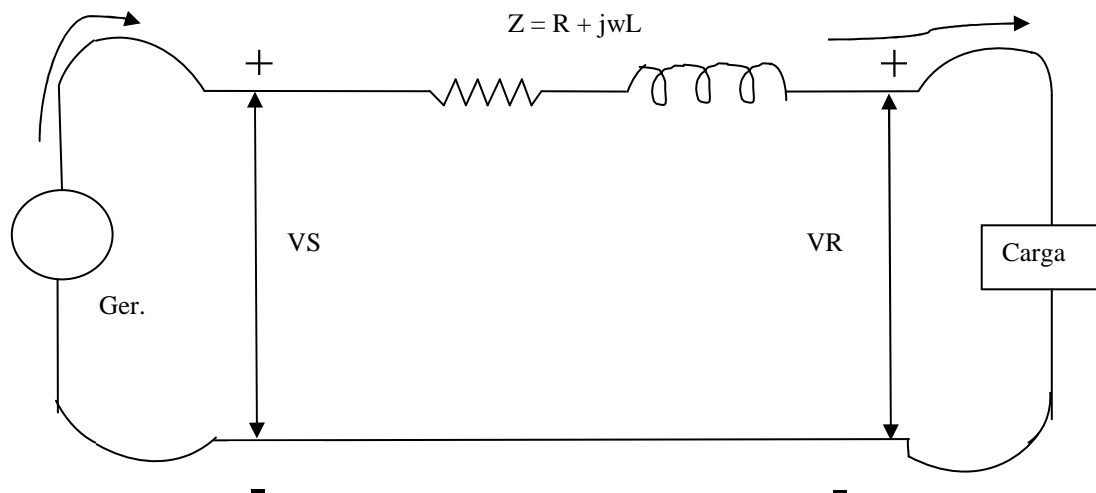


Figura 2.0 - Circuito equivalente de uma linha de transmissão curta

O circuito é resolvido como um simples circuito C.A. série. Não havendo ramos em paralelo, a corrente será a mesma nas extremidades da linha, e $I_S = I_R$.

A tensão na barra transmissora é:

$$V_S = V_R + I_R Z$$

Onde Z é $z l$, impedância total em série da linha.

O efeito da variação do fator de potência na carga sobre a regulação de tensão da linha é mais bem compreendido para o caso da linha curta, razão pela qual iremos considerá-la agora. A regulação de tensão de uma linha de transmissão é o aumento de tensão na barra, dado em percentagem da tensão a plena carga, quando toda a carga, a um determinado fator de potência, é retirada da linha, mantendo constante a tensão na barra receptora, isto é

$$\text{Regulação \%} = \frac{|V_{R,NL}| - |V_{R,FL}|}{|V_{R,FL}|} \times 100$$

Onde $|V_{R,NL}|$ é a amplitude da tensão em vazio na barra receptora e $|V_{R,FL}|$ é a tensão de plena carga na mesma barra, com V_S constante. Quando a carga de uma linha de transmissão curta, como representada no circuito da figura 2.0, é removida, a tensão na barra receptora é igual à tensão na barra na figura 2.0, com a carga conectada, a tensão na barra receptora é designada por V_R e $|V_R| = |V_{R,FL}|$. A tensão na barra transmissora é V_S e $|V_S| = V_{R,NL}$. Os diagramas fasoriais da figura 4 foram traçados para as mesmas amplitudes de tensão e de corrente na barra receptora e mostram que se requer uma tensão mais elevada na barra transmissora para manter uma tensão desejada na barra receptora, quando a corrente nesta barra estiver atrasada em relação a tensão, do que quando esta corrente e tensão estiverem em fase. Uma tensão ainda menor na barra transmissora se faz necessária para

manter a tensão dada na barra receptora quando a corrente estiver adiantada em relação a tensão. A queda de tensão na impedância em série é igual em ambos os casos, mas, devido a diferentes valores de fator de potência, esta queda de tensão é acrescentada à tensão da barra receptora em ângulos diferentes, em cada caso. A regulação é maior para fator de potência atrasado e menor, ou mesmo negativa para fator de potência adiantado. A reatância indutiva é maior do que resistência, em linhas de transmissão, e o princípio da regulação ilustrado na figura 2.1 é verdadeiro para qualquer carga alimentada por um circuito predominantemente indutivo. As amplitudes das quedas de tensão $I_R R$ e $I_R X_L$ para uma linha curta foram exageradas em relação à de V_R no traçado dos diagramas, com objetivo de tornar mais clara a ilustração do fato. A relação entre fator de potência e regulação para linhas mais longas é semelhante à das linhas curtas mas não pode ser tão facilmente visualizada.

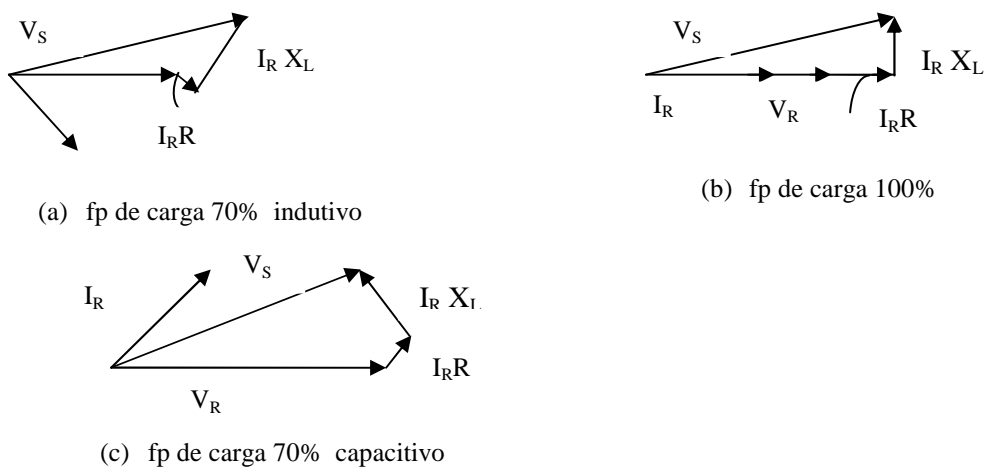


Figura 2.1 - Diagramas fasoriais de uma linha de transmissão curta.

2.5 Linha de transmissão média

A admitância em paralelo, geralmente uma capacitância pura, é incluída nos cálculos de uma linha de transmissão de comprimento médio. Se toda a admitância for

suposta concentrada no meio do circuito representativo da linha, este é denominado circuito nominal T. A figura 2.2 mostra esse circuito, onde Z é z_l , impedância total em série por fase e Y é y_l admitância total em paralelo, por fase.

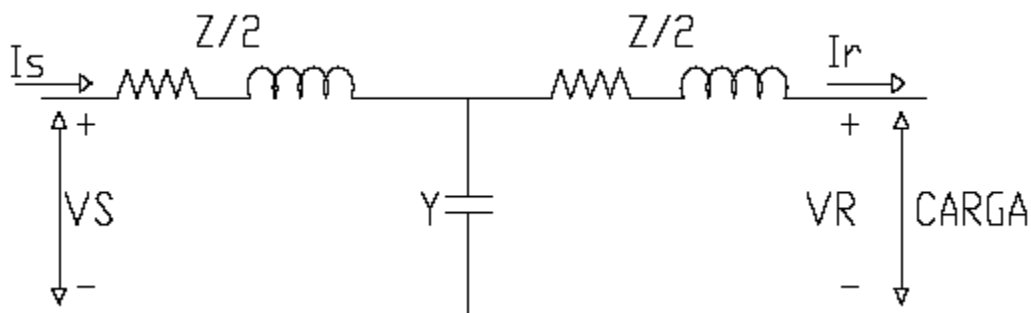


Figura 2.2 - Modelo T da linha de transmissão

O circuito π , mostrado na figura 2.3 é de uso mais freqüente para a representação de linhas médias do que modelo T. A admitância total em paralelo é dividida em duas partes iguais, colocadas junto as barras transmissora e receptora da linha. Para esse circuito, a equação de V_S pode ser deduzida observando-se que a corrente na capacitância da barra receptora é $V_R Y/2$ e a corrente no ramo série é $I_R + V_R \cdot Y/2$.

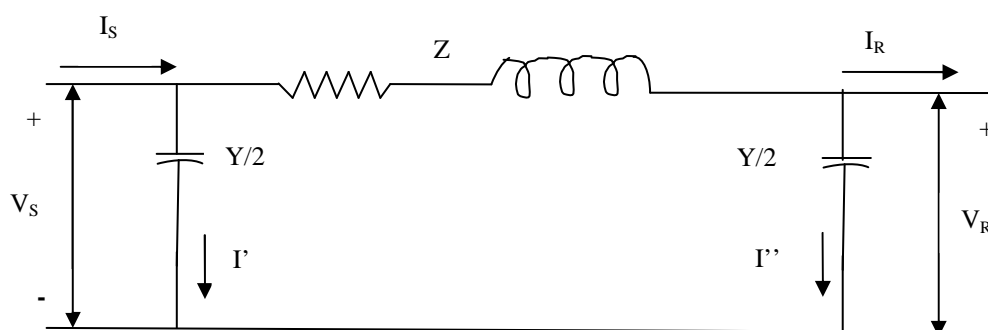


Figura 2.3 - Modelo π da linha de transmissão.

$$V_S = \left(\frac{V_S Y}{2} + I_R \right) Z + V_R$$

$$V_S = \left(\frac{ZY}{2} + 1\right)V_R + Z \cdot I_R$$

$$I_S = V_S \frac{Y}{2} + V_R \frac{Y}{2} + I_R$$

Substituindo V_S na equação acima:

$$I_S = \left[\left(\frac{ZY}{2} + 1\right)V_R + ZI_R\right] \frac{Y}{2} + V_R \frac{Y}{2} + I_R$$

$$I_S = V_R Y \left(\frac{ZY}{4} + 1\right) + \left(\frac{ZY}{2} + 1\right) I_R$$

De uma forma geral, temos:

$$V_S = AV_R + BI_R$$

$$I_S = CV_R + DI_R$$

Estas constantes ABCD denominadas constantes generalizadas do circuito da linha de transmissão. Em geral, elas são números complexos. A e D são adimensionais e iguais se a linha for a mesma quando vista dos dois terminais. As dimensões de B e C são ohms e siemens, respectivamente.

Estas constantes são aplicáveis a redes lineares, passivas e bilaterais com dois pares de terminais (quadripolos).

É fácil um significado físico a elas. Fazendo $I_R = 0$ na equação 8, concluímos que A é a razão V_S/V_R em vazio. Da mesma forma, B é a razão V_S/I_R quando a barra receptora é curto-circuitada. A constante A é útil no cálculo da regulação. Se $V_{R,FL}$ é a tensão na barra receptora em plena carga para uma tensão igual a V_S na barra transmissora:

$$\text{Regulação}\% = \frac{\frac{|V_S|}{|A|} - |V_{R,FL}|}{|V_{R,FL}|} \times 100\%$$

Para uma Linha curta teremos em termos das constantes generalizadas:

$$A=1, B=Z, C=0, D=1$$

Para uma Linha média teremos em termos das constantes generalizadas:

$$A = D = \frac{ZY}{2} + 1$$

$$B = Z$$

$$C = Y \left(\frac{ZY}{4} + 1 \right)$$

2.6 Linha de transmissão longa

Na solução exata de qualquer linha de transmissão e na solução com alto grau de precisão de linhas de 60Hz com mais de 150 milhas de comprimento, devemos considerar que os parâmetros da linha são atribuídos uniformemente ao longo dela e não concentrados.

A figura 2.4 mostra a conexão de uma fase e do neutro de uma linha trifásica. Não são mostrados parâmetros concentrados porque vamos considerar a solução da linha com admitância e impedância distribuída uniformemente. O mesmo diagrama pode também representar uma linha monofásica se utilizarmos a impedância em série do laço da linha monofásica do lugar da impedância por fase da linha trifásica e se utilizarmos a admitância em derivação entre linha e neutro da linha trifásica.

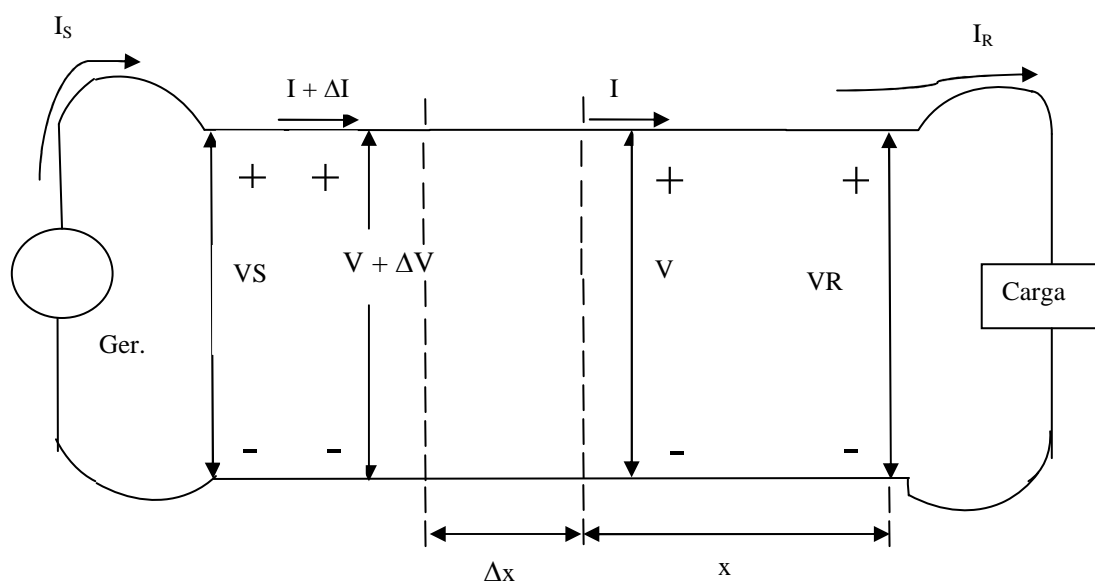


Figura 2.4 - Diagrama simplificado de uma linha de transmissão longa

Consideremos um elemento muito pequeno da linha e calculemos a diferença de tensão e a diferença de correntes entre as duas extremidades do elemento. Chamaremos x a distância medida a partir da barra receptora até o pequeno elemento da linha, e chamaremos o comprimento do elemento Δx .

Então, $z\Delta x$ será a impedância em série, e $y\Delta x$ a admitância em derivação do elemento da linha. A tensão ao neutro na extremidade do elemento do lado da carga é V , que é a expressão complexa do valor eficaz da tensão, cuja amplitude e fase variam com a distância ao longo da linha. A tensão na extremidade do lado do gerador é $V + \Delta V$. A elevação de tensão neste elemento da linha no sentido do crescimento de x é ΔV , que é a tensão na extremidade do lado do gerador menos a tensão na do lado da carga. Esta elevação de tensão é também o produto da corrente que circula em direção oposta ao crescimento de x pela impedância do elemento, ou $Iz \Delta x$. Então:

$$\Delta V = Iz\Delta x$$

$$\frac{\Delta V}{\Delta x} = Iz$$

e, quando $\Delta x \rightarrow 0$, o limite do quociente acima se torna:

$$\frac{dV}{dx} = Iz$$

Da mesma forma, a corrente que flui para fora do elemento no lado da carga é I . A amplitude e a fase da corrente I variam com a distância ao longo da linha, devido a admitância em derivação distribuída pela mesma. A corrente que flui para dentro do elemento, do lado do gerador, é $I + \Delta I$. A corrente que entra pelo lado do gerador é maior do que a que sai pelo lado da carga de uma quantidade ΔI . Esta diferença de corrente é a corrente Vy , Δx que flui pela admitância em derivação do elemento. Portanto

$$\Delta I = Iz\Delta x$$

e, através de passos semelhantes, obtemos:

$$\frac{dI}{dx} = Vy$$

Se substituirmos os valores de dI/dx e dV/dx das equações 15 e 14 nas equações 16 e 17 respectivamente, obtemos:

$$\frac{d^2V}{dx^2} = yzV \quad (18)$$

$$\frac{d^2I}{dx^2} = yzI \quad (19)$$

Agora, temos uma equação, 18, onde as únicas variáveis são V e x e outra equação, 19, na qual as únicas variáveis são I e x . As soluções das equações 18 e 19 para V e I , respectivamente, podem ser expressões que quando derivadas duas vezes em relação à x levem à expressão original multiplicada pela constante yz . Por exemplo, à solução de V

quando derivada duas vezes em relação à x deve dar yzV . Isto sugere uma equação exponencial.

2.6.1 Linha de transmissão longa: interpretação das equações

Se $Z_c = \sqrt{(z/y)}$ é chamada a impedância característica da linha e $\gamma = \sqrt{yz}$, é chamada a constante de propagação.

Tanto γ quanto Z_c , são grandezas complexas. A parte real da constante de propagação γ é denominada constante de atenuação α e é medida em nepers por unidade de comprimento. A parte imaginária de γ é chamada constante de fase β e é medida em radianos por unidade de comprimento. Então

$$\gamma = \alpha + j\beta \quad (25)$$

E as equações 23 e 24 se tornam

$$V = \left(\frac{V_R + I_R Z_c}{2} \right) e^{\alpha x} e^{j\beta x} + \left(\frac{V_R - I_R Z_c}{2} \right) e^{-\alpha x} e^{-j\beta x} \quad (26)$$

$$I = \left(\frac{V_R/Z_c + I_R}{2} \right) e^{\alpha x} e^{j\beta x} - \left(\frac{V_R/Z_c - I_R}{2} \right) e^{-\alpha x} e^{-j\beta x} \quad (27)$$

As propriedades de $e^{\alpha x}$ e de $e^{j\beta x}$ ajudam a explicar a variação dos valores fasoriais de tensão e de corrente em função da distância ao longo da linha. Com variações de x , varia em amplitude, enquanto $e^{j\beta x}$, que é idêntico a $\cos \beta x + j \operatorname{sen} \beta x$, tem amplitude constante igual a 1,0 e provoca um deslocamento de fase de β radianos por unidade de comprimento da linha.

O primeiro termo da equação 26, $[(V_R + I_R Z_c)/2] \cdot e^{\alpha x} \cdot e^{j\beta x}$, cresce em amplitude e avança em fase, à medida que cresce a distância a partir da barra receptora. Inversamente, se considerarmos o deslocamento pela linha a partir da barra transmissora, este

termo diminui em amplitude e se atrasa em fase. Esta é a característica de uma onda viajante e é semelhante ao comportamento de uma onda na água, cuja amplitude varia com o tempo, em qualquer ponto, enquanto sua fase é retardada e seu valor máximo diminui com a distância à origem. A variação no valor instantâneo não é representada no termo mas é evidente, por serem V_R e I_R fasores. O primeiro termo da equação 26 é chamado tensão incidente.

O segundo termo da equação 26, $[(V_R + IRZ_c)/2] \cdot e^{\alpha x} \cdot e^{j\beta x}$, diminui em amplitude e se atrasa em fase da barra receptora para a barra transmissora. Ele é chamado tensão refletida. Em qualquer ponto ao longo da linha, a tensão é a soma das componentes incidente refletida naquele ponto.

Como a equação da corrente é semelhante à equação de tensão, ela também pode ser considerada composta de correntes incidente e refletida.

V_R será igual a $I_R Z_C$ e não haverá ondas refletidas de corrente nem de tensão na barra receptora V_R por $I_R Z_C$ nas equações 26 e 27. Uma linha que alimenta a sua impedância característica é chamada linha plana ou linha infinita. A segunda denominação se baseia no fato de que uma linha infinita não pode produzir onda refletida. Normalmente, as linhas de potência não alimentam sua impedância característica, mas as linhas de comunicação freqüentemente são assim projetadas, de modo a eliminar a onda refletida. Um valor típico de Z_c é de 400Ω para um linha aérea e de circuito simples e de 200Ω para dois circuitos em paralelo. O ângulo de fase de Z_c situa-se quase sempre entre 0° e -15° . As linhas de cabos múltiplos possuem valores de Z_c mais baixos porque possuem valores menores de L e valores maiores de C do que as linhas com um único condutor por fase.

2.7 Linha de transmissão longa: Forma hiperbólica das equações

No cálculo da tensão de uma linha de potencia, raramente se determina à onda incidente e a onda refletida. O motivo de havermos discutido a tensão e a corrente de uma linha de transmissão em termos das componentes incidente e refletida se deve ao fato de que tal análise é útil na obtenção de um entendimento mais completo de alguns dos fenômenos das linhas de transmissão. Uma forma de equações mais conveniente para o cálculo da corrente e da tensão de uma linha de potencia é obtida pela introdução de funções hiperbólicas, as quais são definidas em forma exponencial por:

$$\sinh\theta = \frac{e^{\theta} - e^{-\theta}}{2} \quad (31)$$

$$\cosh\theta = \frac{e^{\theta} + e^{-\theta}}{2} \quad (32)$$

Reagrupando as equações 23 e 24 e substituindo os termos exponenciais por funções hiperbólicas, obtemos um novo conjunto de equações. As novas equações, que dão a tensão e a corrente ao longo da linha, são

$$V = V_R \cosh \gamma x + I_R Z_c \sinh \gamma x \quad (33)$$

$$I = \frac{V_R}{Z_c} \sinh \gamma x + I_R \cosh \gamma x \quad (34)$$

Fazendo $x = l$, obtemos a tensão e a corrente na barra transmissora:

$$V_S = V_R \cosh \gamma l + I_R Z_c \sinh \gamma l \quad (35)$$

$$I_S = \frac{V_R}{Z_c} \sinh \gamma l + I_R \cosh \gamma l \quad (36)$$

Examinando as equações acima, vemos que as constantes generalizadas de uma linha longa são $A = \cosh \gamma l$

$$B = Z_c \sinh \gamma l \quad (37)$$

$$C = \frac{\sinh \gamma l}{Z_c}$$

$$D = \cosh \gamma l$$

As equações 35 e 36 podem ser resolvidas para se obter V_R e I_R em termos de V_S e I_S obtendo-se

$$V_R = V_S \cosh \gamma l - I_S Z_c \sinh \gamma l \quad (38)$$

$$I_R = I_S \cosh \gamma l - \frac{V_S}{Z_c} \sinh \gamma l \quad (39)$$

Para linhas trifásicas equilibradas, usa-se a corrente de linha e a tensão por fase, Isto é, a tensão de linha dividida por $\sqrt{3}$ nas equações acima. Para resolver as equações, devem ser calculadas as funções hiperbólicas. Como usualmente γl é um numero complexo, as funções hiperbólicas também são complexas e não podem ser obtidas diretamente de tabelas comuns ou por meio de calculadoras eletrônicas. Antes da difusão do computador digital, muitos métodos gráficos, alguns dos quais especialmente adaptados aos valores normalmente encontrados em cálculos de linhas de transmissão, foram muito usados para o calculo de funções hiperbólicas de argumentos complexos, hoje, o computador digital é o meio usual de incorporação destas funções em nossos cálculos.

Existem diversas opções para a solução de um problema ocasional sem a necessidade de se recorrer a um computador ou a gráficos. Um primeiro método, cujas equações damos abaixo, consiste em desenvolver os senos e co-senos hiperbólicos em termos de funções hiperbólicas e trigonométricas de argumentos reais.

$$\cosh(\alpha l + j\beta l) = \cosh \alpha l \cdot \cos \beta l + j \sinh \alpha l \cdot \sin \beta l \quad (40)$$

As equações 40 e 41 tornam possível o calculo de funções hiperbólicas de argumentos complexos. A unidade correta para βl é o radiano, e o radiano é a unidade obtida para βl quando se calcula a parte imaginaria de γl . As equações 40 e 41 podem ser comprovadas por substituição das formas exponenciais das funções hiperbólicas e circulares.

2.8 Indutância nas linhas de transmissão

Fisicamente, as linhas de transmissão nada mais são que conjuntos de condutores (de cobre ou alumínio) que transportam energia elétrica dos geradores as cargas. Um dos parâmetros mais importantes na definição da capacidade de transmissão de uma linha de transmissão é a impedância da linha que, por sua vez, depende basicamente da indutância (além é claro da resistência). Sabemos que uma corrente elétrica produz um campo magnético e um fluxo magnético a ele associado. A intensidade do fluxo varia diretamente com a magnitude da corrente; depende também da distribuição espacial (geometria do condutor) e do meio no qual o condutor está inserido. A relação geral entre fluxo e corrente é dada pela Lei de Ampère, que é uma das equações de Maxwell.

Em particular, veremos que a indutância das linhas de transmissão em corrente alternada depende do comprimento da linha: quanto mais longas, maiores indutâncias e, portanto, maiores impedâncias e a oposição oferecida pela linha à transmissão de potência elétrica. Esta é uma das razões pelas quais, para distâncias mais longas (por exemplo, acima de mil quilômetros), linhas de transmissão de corrente contínua se tornam economicamente mais competitivas.

2.8.1 Transposição de condutores

Quando o espaçamento entre condutores é tal que a seção reta da linha não é equilátera, a matriz indutância resultante será assimétrica, conforme visto anteriormente. A solução para esse problema é a transposição dos condutores conforme ilustrado na figura 2.5.

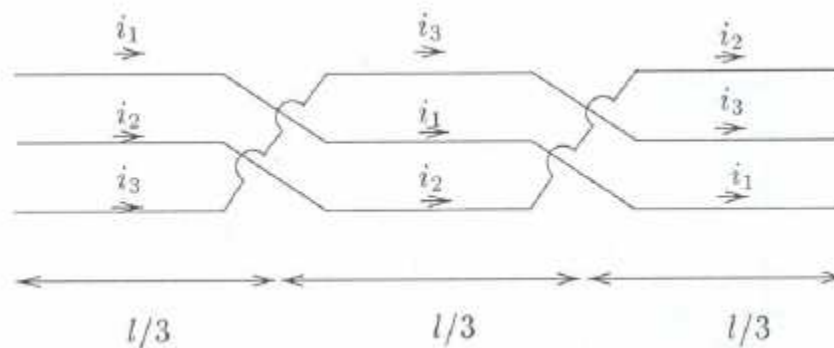


Figura 2.5 - Transposição de linha trifásica

A transposição é realizada em intervalos regulares, podendo ser feita em estações de chaveamento. Os condutores ficam a 1/3 do percurso em cada posição possível (esquerda, centro e direita). Assim, o fluxo concatenado com o condutor 1 terá três componentes: uma componente para cada uma das três posições possíveis. Supondo-se que o condutor fica em cada uma das posições por 1/3 de uma unidade de comprimento, conforme indicado na figura 14, teremos que um fluxo concatenado dado por

$$\phi_{c1} = \frac{\mu_0}{6\pi} \left[i_1 \ln \frac{2D}{R'} + i_2 \ln 2 + i_1 \ln \frac{D}{R'} + i_3 \ln 2 + i_1 \ln \frac{2D}{R'} \right]$$

Considerando-se que, no primeiro segmento de linha (primeiro terço), a corrente i_1 passa pelo condutor 1, a corrente i_2 passa pelo condutor 2 e a corrente i_3 passa pelo condutor 3; no segundo segmento, a corrente i_1 passa pelo condutor 2, a corrente i_2 passa pelo condutor 3 e a corrente i_3 passa pelo condutor 1; e analogamente, para o terceiro segmento. Notar que o fluxo concatenado com a corrente i tem três componentes que correspondem às três linhas da matriz que aparecem na expressão 3. Observar ainda que, no cálculo das contribuições ao fluxo correspondentes às linhas 2 e 3 da matriz, as correntes (componentes do vetor de correntes) devem respectivamente, as seguintes seqüências de correntes para as

três linhas da matriz: (i_1, i_2, i_3) para a primeira linha, (i_3, i_1, i_2) para a segunda linha e (i_2, i_3, i_1) para a terceira linha.

Considerando agora que $i_1 + i_2 + i_3 = 0$, obtemos

$$\emptyset_{c1} = \frac{\mu_0}{6\pi} i_1 \ln \frac{2D^3}{R'^3} = \frac{\mu_0}{2\pi} i_1 \ln \frac{\sqrt[3]{2D}}{R'}$$

E assim a indutância associada à fase 1 (condutor 1) será dada por

$$L_1 = \frac{\mu_0}{2\pi} \ln \frac{D'}{R'} \quad H/m$$

Observação:

O lado do triângulo equilátero da linha equivalente mostrada na figura 2.6 é dado por:

$$D' = \sqrt[3]{D \cdot D \cdot 2D} = \sqrt[3]{2D^3}$$

Ou seja, pela média geométrica dos espaçamentos entre os condutores na linha original. Isto não é uma mera coincidência como veremos mais adiante; trata-se de uma propriedade geral com conseqüências práticas importantes.

2.8.2 Linhas com vários condutores por fase

Inicialmente, talvez seja importante lembrar que, além das cargas que se movem e produzem corrente elétrica que circula nos condutores de uma linha, podem existir cargas excedentes que produzem uma diferença de potencial entre os condutores (tensão aplicada). As cargas associadas às correntes são eletricamente equilibradas pelas cargas dos núcleos, pois há apenas um deslocamento relativo sem perda de equilíbrio entre cargas positivas e negativas; e essas cargas se movem no interior do condutor, de maneira uniforme, se desprezarmos o chamado efeito pelicular (o que é válido para baixas freqüências). Já as

cargas excedentes se repelem e tendem a se localizar na superfície dos condutores. A intensidade do campo elétrico produzido por essas cargas pode-se tornar muito elevada nas proximidades dos condutores (pois, em geral, variam na razão inversa das distâncias dos eixos dos condutores), podendo, em alguns casos, provocar o efeito corona (ruptura dielétrica ao redor do condutor). Uma maneira de combater esse efeito consiste em aumentar o diâmetro dos condutores. Isso, entretanto, acarretaria custos maiores e problemas estruturais desnecessários. O que se faz na prática é utilizar a idéia básica da Gaiola de Faraday que, no caso das linhas de transmissão, equivale utilizar vários condutores por fase, conforme ilustrado na figura 2.6 (dois condutores por fase) e 13 (quatro condutores por fase). Na prática também são encontradas linhas com outros arranjos, como por exemplo, três condutores por fase. Os campos elétricos tendem a ser nulos nas regiões internas aos condutores que formam uma fase, ou seja, o conjunto de condutores se comporta como se fosse um condutor único de raio maior; como os campos só existem nas partes externas, sua intensidade diminui e assim fica também reduzida a possibilidade de ocorrência de efeito corona.

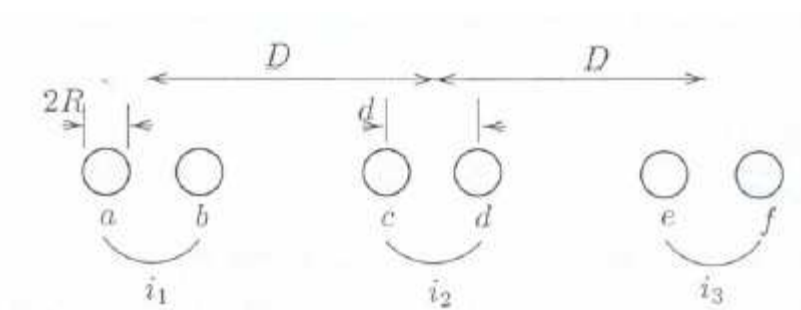


Figura 2.6 - Linha trifásica com arranjo linear e com dois condutores por fase

Seguindo-se os mesmos passos utilizados anteriormente, poderemos verificar que o fluxo concatenado com o condutor a da fase 1 é dado por

$$\phi_{c0} = \frac{\mu_0}{2\pi} \left[\frac{i_1}{2} \ln \frac{d_{aP}}{R'} + \frac{i_1}{2} \ln \frac{d_{bP}}{d} + \frac{i_2}{2} \ln \frac{d_{cP}}{D} + \frac{i_2}{2} \ln \frac{d_{dP}}{D+d} + \frac{i_3}{2} \ln \frac{d_{eP}}{2D} + \frac{i_3}{2} \ln \frac{d_{fP}}{2D+d} \right]$$

Fazendo $P \rightarrow \infty$ e lembrando que $i_1 + i_2 + i_3 = 0$, teremos

$$\phi_{c0} = \frac{\mu_0}{2\pi} \left[\frac{i_1}{2} \ln \frac{1}{R'} + \frac{i_1}{2} \ln \frac{1}{d} + \frac{i_2}{2} \ln \frac{1}{D} + \frac{i_2}{2} \ln \frac{1}{D+d} + \frac{i_3}{2} \ln \frac{1}{2D} + \frac{i_3}{2} \ln \frac{1}{2D+d} \right]$$

A partir dessas expressões de fluxo concatenado, podemos calcular as indutâncias por fase; o que resultará em uma matriz indutância assimétrica, da mesma forma que ocorre ara linhas com um único condutor por fase. Se considerarmos transposição, chegaremos finalmente a indutância por fase dada por

$$L = \frac{\mu_0}{2\pi} \ln \frac{\sqrt[3]{2D}}{\sqrt{R'd}}$$

2.8.3 Sistema de transmissão de Itaipu

Faremos aqui o cálculo da indutância de um dos principais troncos de transmissão de energia elétrica em operação (linha de CA em 750kV de Itaipu, operada por furnas).

A figura 2.7 mostra a seção reta de um dos circuitos trifásicos que constituem o sistema de transmissão de Itaipu (cada segmento de reta na parte referente à corrente alternada da figura 2.7 é de fato uma linha trifásica). Cada fase é formada por um conjunto de quatro condutores.

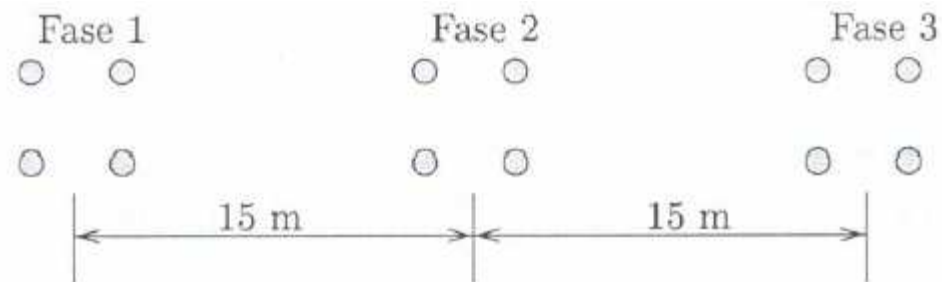


Figura 2.7 - Seção reta de um circuito trifásico do sistema de transmissão de Itaipu.

Se olharmos para cada um dos quatro condutores que formam uma fase, veremos que esses condutores na verdade são cabos encordoados. A figura 2.8 mostra alguns tipos mais simples de encordoamentos. No caso do sistema CA de Itaipu, os cabos são do tipo bluejay (é prática da indústria utilizar nomes de aves para denominar diferentes tipos de cabos; a tabela 2.3 dá alguns desses tipos para efeito de ilustração). No jargão das linhas de transmissão, diz-se que cada fase é do tipo 4 x bluejay ACSR.

Significando: 4 condutores (cabos) bluejay Aluminum Cable Steel Reinforced ou CAA, em português, cabos com alma de aço.

A tabela 2.3 indica que a área da seção do cabo bluejay é de 1.113.000 CM; sendo 1 Circular Mil (CM)=área de um círculo de diâmetro 1 milésimo de polegada. A Tabela indica ainda que, para o cabo bluejay, 45 é o número de fios de alumínio (Al), 7 é o número de fios de aço (St) e que os fios de alumínio estão distribuídos em duas camadas. Existem também outros tipos de cabos. Por exemplo, cabos AAC (All Aluminum Cable). Os cabos são fios colocados em coroas superpostas encordoadas em sentidos opostos.

Esse tipo de encordoamento evita que o cabo se desenrole e faz com que o raio externo de uma coroa coincida com o raio externo da seguinte. A disposição em coroas dá flexibilidade aos cabos de até grande seção transversal. Para os cabos constituídos de apenas um tipo de material (sem alma, conforme a figura 2.8, a seguinte expressão válida: $n = 3x^2 - 3x + 1$, onde n é o número de fios e x é o número de coroas (camadas).

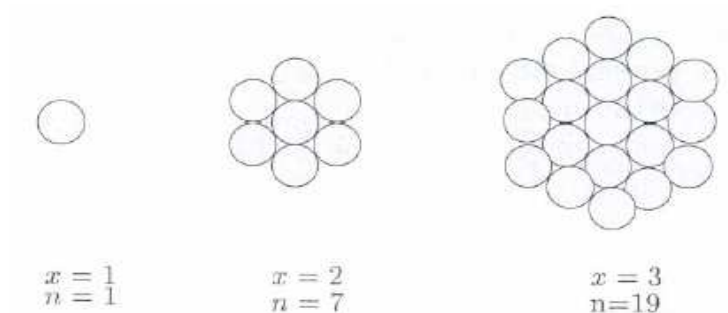


Figura 2.8 – Cabos encordados.

Cabo	CM	Al/St	Camadas Al	Ds ou GMR(pés)
Partridge	266.800	26/7	2	0,0198
Ostrich	300.000	26/7	2	0,0229
Oriole	336.400	30/7	2	0,0255
Pelican	477.000	18/1	2	0,0264
Hawk	477.000	26/7	2	0,0289
Osprey	556.500	18/1	2	0,0284
Drake	795.000	26/7	2	0,0373
Cardinal	954.000	54/7	3	0,0402
Rail	954.000	45/7	3	0,0386
Bluejay	1113.000	45/7	3	0,0415
Pheasant	1.272.000	54/9	3	0,0466
Falcon	1.590.000	54/19	3	0,0523

Tabela 2.3 - Característica dos cabos ACSR

2.9 Capacitância nas linhas de transmissão

A admitância em derivação de uma linha de transmissão consiste em uma

condutância e uma reatância capacitiva. A condutância é usualmente desprezada devido a sua pequena contribuição com a admitância em derivação.

Outra razão para que se despreze a condutância reside no fato de não nenhum meio apropriado de considerá-la, por ser ela muito variável. A fuga pelos isoladores, que é a principal fonte de condutância, varia apreciavelmente com as condições atmosféricas e com as propriedades de condução da poeira que se deposita sobre os isoladores. O efeito corona, que resulta em fuga através dos condutores das linhas, é também bastante variável com as condições atmosféricas. Felizmente, o efeito da condutância é um componente tão desprezível da admitância em derivação que pode ser ignorado.

A capacitância de uma linha de transmissão resulta da diferença de potencial entre os condutores; ela faz com que estes se tornem carregados de modo semelhante as placas de um capacitor entre as quais exista uma diferença de potencial. A capacitância entre os condutores é a carga por unidade de diferença de potencial. A capacitância entre condutores em paralelo é uma constante que depende das dimensões e do afastamento entre os condutores. Para linhas menores que 80km (50 milhas) de comprimento, o efeito da capacitância é mínimo e usualmente é desprezado. Para linhas mais longas de tensões mais elevadas, torna-se mais importante a capacitância.

Uma tensão alternada aplicada sobre uma linha de transmissão faz com que, em qualquer ponto, as cargas dos condutores cresçam e decresçam com o aumento e a diminuição do valor instantâneo da tensão entre os condutores naquele ponto. O deslocamento de cargas é uma corrente, e a corrente causada pelo carregamento e descarregamento alternados de uma linha devidos a uma tensão alternada é chamada de corrente de carregamento da linha. A corrente de carregamento existe na linha de transmissão mesmo quando ela está a vazio. Ela afeta tanto a queda de tensão ao longo da linha quanto o seu rendimento, o fator de potência e

a estabilidade do sistema ao qual pertence a linha.

2.9.1 Capacitância de uma linha a dois fios

A capacitância entre dois condutores de uma linha a dois fios foi definida como a quantidade de carga nos condutores por unidade de diferença de potencial entre eles. Na forma de uma Equação, a capacitância por unidade de comprimento da linha é

$$C = \frac{q}{v} \text{ F/m}$$

Onde q é a carga sobre a linha, em coulombs por metro e v é a diferença de potencial entre os condutores em volts. Daqui pra frente, por comodidade, diremos apenas capacitância quando nos referimos à capacitância por unidade de comprimento. Podemos obter a capacitância entre dois condutores substituindo na Equação 4 o valor de v_{ab} em função de q . A tensão v_{ab} entre os condutores da linha a dois fios mostrada na figura 2.9 pode ser obtida determinando a diferença de potencial entre os dois condutores da linha, calculando primeiro a queda de tensão devida à carga q_a do condutor a e depois a queda de tensão devida à carga q_b do condutor b . Pelo princípio da superposição, a queda de tensão entre o condutor a e o condutor b , devida a carga dos dois condutores, é a soma da queda de tensão devida a cada um deles.



Figura 2.9 - Seção transversal de uma linha de fios paralelos

a capacitância entre os condutores é:

$$C_{ab} = \frac{q_a}{V_{ab}} = \frac{2\pi k}{\ln(D^2/r_a r_b)} F/m$$

se $r_a = r_b = r$,

$$C_{ab} = \frac{\pi k}{\ln(D/r)} F/m$$

A última equação dá a capacitância entre condutores de uma linha a dois fios.

As vezes, é desejável conhecer a capacitância entre um dos condutores e um ponto neutro entre eles. Por exemplo, se a linha for alimentada por um transformador com uma derivação central aterrada, a diferença de potencial entre cada condutor e a terra será igual a metade da diferença de potencial entre os dois condutores e a capacitância à terra, ou capacitância ao neutro será a carga em um condutor por unidade de diferença de potencial entre o condutor e a terra. A capacitância ao neutro para a linha a dois fios é o dobro da capacitância linha-linha (capacitância entre condutores). Se a capacitância entre linhas for considerada composta por duas capacitâncias iguais em serie, a tensão de linha se dividirá igualmente entre elas e sua junção estará ao potencial de terra. A capacitância ao neutro será, portanto, igual a qualquer das duas capacitâncias iguais em serie, ou seja, duas vezes a capacitância linha-linha. Portanto,

$$C_n = C_{an} = C_{bn} = \frac{2\pi k}{\ln(D/r)} F/m \quad (\text{ao neutro}).$$

E esta última equação corresponde à equação:

$$L = 2 \times 10^{-7} \ln \frac{D}{r'} \quad (\text{para indutância}).$$

Tendo sido obtida a capacitância ao neutro, a reatância capacitiva entre um

condutor e o neutro, para uma permissividade relativa $k_r = 1$, é calculada usando a expressão de $C_n = C_{an} = C_{bn}$.

$$X_c = \frac{1}{2\pi f C} = \frac{2,862}{f} \times 10^9 \ln \frac{D}{r} \Omega.m \text{ (para neutro)}$$

Dividindo a Equação anterior por 1.609, obtemos a reatância capacitiva em ohms-milhas.

$$X_c = \frac{1,779}{f} \times 10^6 \ln \frac{D}{r} \Omega.mi$$

A tabela 2.4 apresenta os diâmetros externos mais usados em condutores CAA. 1 incluindo os valores de X'_a para as bitolas mais comuns de cabos CAA, e existem a disposição tabelas para outros tipos de condutores, em suas diferentes bitolas.

Cabo	CM	Al/St	Camadas	Ds ou GMR(pés)	Diâmetro	X'a, MΩ/mi
Partridg e	266.800	26/7	2	0,0198	0,642	0,1074
Ostrich	300.000	26/7	2	0,0229	0,680	0,1057
Oriole	336.400	30/7	2	0,0255	0,741	0,1032
Pelican	477.000	18/1	2	0,0264	0,814	0,1004
Hawk	477.000	26/7	2	0,0289	0,858	0,0988
Osprey	556.500	18/1	2	0,0284	0,879	0,0981
Drake	795.000	26/7	2	0,0373	1,108	0,0912
Cardinal	954.000	54/7	3	0,0402	1,196	0,0890
Rail	954.000	45/7	3	0,0386	1,165	0,0897
Bluejay	1113.00	45/7	3	0,0415	1,259	0,0874

	0					
Pheasant	1.272.00	54/9	3	0,0466	1,382	0,0874
	0					
Falcon	1.590.00	54/19	3	0,0523	1,545	0,814
	0					

Tabela 2.4 - Características dos principais condutores

2.9.2 Capacitância de uma linha trifásica com espaçamento equilateral

Três condutores idênticos de raio r são mostrados na figura 2.10, formando uma linha trifásica de espaçamento equilateral. Para uma distribuição de cargas sobre os condutores considerada uniforme, a equação abaixo fornece a tensão entre dois condutores devida a carga em cada um deles. Portanto, a tensão v_{ab} de uma linha trifásica devida somente às cargas dos condutores a e b é

$$V_{ab} = \frac{1}{2\pi k} \left[q_a \ln \frac{D}{r} + q_b \ln \frac{r}{D} \right] V$$

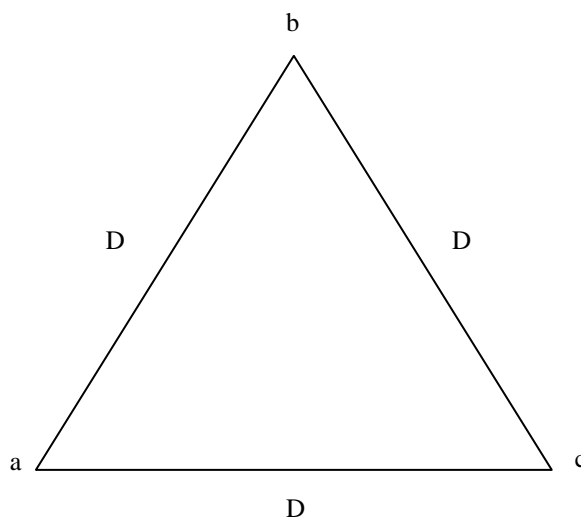


Figura 2.10 - Seção transversal de uma linha com espaçamento equilateral

A distribuição uniforme de carga sobre a superfície de um condutor é

equivalente a uma carga concentrada no centro do condutor. Portanto, devido somente à carga

q_c que é igual a zero, pois q_c é equidistante de a e de b . Entretanto, para mostrar que estamos considerando todas as três cargas, podemos escrever

$$V_{ab} + V_{ac} = \frac{1}{2\pi k} \left[2q_a \ln \frac{D}{r} + (q_b + q_c) \ln \frac{r}{D} \right] V$$

Como a capacitância ao neutro é o quociente da carga em um condutor pela queda de tensão entre aquele condutor e o neutro, temos

$$C_n = \frac{q_a}{V_{an}} = \frac{2\pi k}{\ln(D/r)} F/m$$

2.9.3 Cabos múltiplos

A figura 2.11 mostra uma linha de cabos múltiplos para o qual podemos escrever uma Equação para a tensão entre os cabos a e b , exceto que neste caso devemos considerar as cargas em todos os seis condutores individuais.

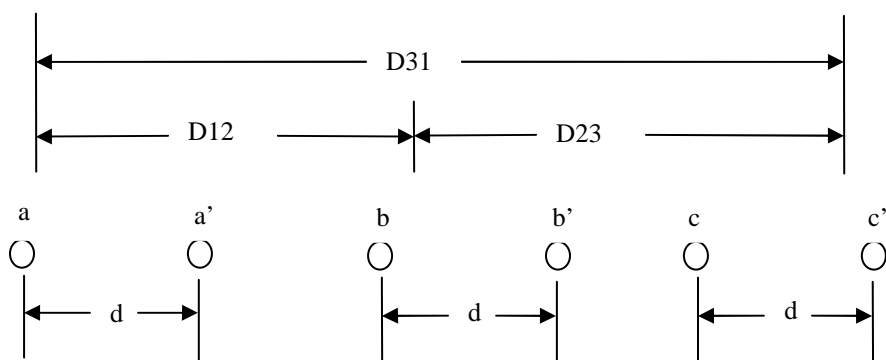


Figura 2.11 - Seção transversal de uma linha trifásica de cabos múltiplos

Os condutores de cada cabo estão em paralelo, e podemos admitir que a carga por cabo se divide igualmente entre os condutores desse cabo, pois o afastamento entre os

condutores de mesma fase é usualmente mais do que 15 vezes maior do que o espaçamento entre cabos. Também, como D_{12} é substituir as expressões mais exatas obtidas no cálculo de V_{ab} pelas distancias entre os centros dos cabos múltiplos. Mesmo quando os cálculos são realizados com 5 ou 6 algarismos significativos, não se pode detectar as diferenças que surgem nos resultados finais, devidas a essas simplificações.

$$V_{ab} = \frac{1}{2\pi k} \left[\underbrace{\frac{q_a}{2} \left(\ln \frac{D_{12}}{r} + \ln \frac{D_{12}}{d} \right)}_{\text{As let a ob a' a termo lo, b ic b' licam o c i c or c' carga é}} + \underbrace{\frac{q_b}{2} \left(\ln \frac{r}{D_{12}} + \ln \frac{d}{D_{12}} \right)}_{\text{As let a ob a' a termo lo, b ic b' licam o c i c or c' carga é}} + \underbrace{\frac{q_c}{2} \left(\ln \frac{D_{23}}{D_{31}} + \ln \frac{D_{23}}{D_{31}} \right)}_{\text{As let a ob a' a termo lo, b ic b' licam o c i c or c' carga é}} \right]$$

considerada naquele termo. Combinando os termos, obtemos

$$V_{ab} = \frac{1}{2\pi k} \left[q_a \ln \frac{D_{12}}{\sqrt{rd}} + q_b \ln \frac{\sqrt{rd}}{D_{12}} + q_c \ln \frac{D_{23}}{D_{31}} \right]$$

A equação anterior é semelhante à Equação 26 com exceção da substituição de r por $\sqrt{r.d}$. Segue-se, portanto, que se a linha fosse considerada transposta, teríamos obtido

$$C_n = \frac{2\pi k}{\ln \left(\frac{D_{eq}}{\sqrt{rd}} \right)} F/m \text{ (para o neutro)}$$

2.9.4 Linhas trifásicas de circuitos em paralelo

Em nossa discussão da capacitância, temos notado a semelhança das equações de indutância e de capacitância. Foi obtido um método de DMG modificado para ser aplicado na obtenção da capacitância de linhas de cabos múltiplos. Poderíamos mostrar que esse método é igualmente válido para linhas trifásicas duplas transpostas com espaçamento equilátero (com os condutores no vértice de um hexágono) para espaçamento vertical (com os condutores das três fases de cada circuito colocados no mesmo plano vertical). É razoável supor que o método da DMG modificado possa ser usado para as disposições intermediárias

entre as equiláteras e as verticais. Embora não tenha sido demonstrado, este método é geralmente utilizado.

2.10 Características mecânicas e elétricas das linhas de transmissão

O correto desempenho de uma linha de transmissão deverá ser fruto de uma escolha adequada de seus elementos constituintes. Com efeito, o emprego de determinado cabo para uma linha aérea, por exemplo, poderá levar a um maior ou menor grau de perdas na transmissão ou mesmo à necessidade de equipamentos para suporte de reativos, encarecendo os custos desta linha e tornado mais complexa sua operação.

De modo análogo, a estrutura de uma linha de transmissão é peça fundamental no comportamento desta o surto de qualquer natureza; da opção por determinado tipo de estrutura, poderão advir benefícios econômicos, facilidades de manutenção, maior grau de confiabilidade na transmissão e, não menos importante, menor grau de agressão ao meio ambiente.

Torna-se pois imperioso uma análise mais apurada das características mecânicas e elétricas das linhas de transmissão, de modo a propiciar um projeto bem dimensionado e uma operação mais efetiva destas.

2.11 Cabos condutores em linhas aéreas de transmissão

Por meio dos cabos condutores, realiza-se o processo de transmissão de energia elétrica em uma linha de transmissão. Um cabo condutor ideal deveria ter alta condutividade elétrica, possuir uma boa resistência mecânica, resistir bem as intempéries, ter baixo peso específico e custo não elevados. Dentre os materiais que melhor preenchem estas

características, destacam-se o alumínio e o cobre, cujas propriedades principais são mostradas na tabela 2.5.

Características	Alumínio	Cobre
Condutividade	61% IACS*	97% IACS*
Resistividade em microhm/cm a 20°C	2,828	1,7774
Coefficiente térmico de resistividade, em microhm/cm por °C	0,0115	0,00681
Coefficiente térmico de expansão linear por °C	0,000023	0,000017
Densidade a 20°C em gr/cm ³	2,703	8,89
Carga de ruptura em kg/mm ²	16 – 21	35 - 47
Módulo de elasticidade final, kg/mm ²	7000	12000

Tabela 2.5 – Propriedades de materiais condutores

* IACS – Referente a cobre quimicamente puro, padrão internacional, condutividade de 100% medido a 20°C. Abreviatura do termo inglês “International Annealed Copper Standard”

Nas linhas de transmissão aéreas são utilizados cabos condutores obtidos pelo enrolamento de fios, visto que um condutor sólido, devido às vibrações, produziria fadiga mecânica e conseqüente fratura no ponto de conexão à cadeia de isoladores. Estes cabos encordoados, composto de fio de mesmo diâmetro, obedecem à seguinte lei de formação:

$$N = 3n(n + 1) + 1, \text{ onde}$$

N = número total de fios componentes e

n = número de camadas ou coroas.

Deste modo, um cabo condutor com duas camadas, por exemplo, terá $6 \times 3 + 1 = 19$ fios. Caso o diâmetro de cada fio for d, o diâmetro do cabo condutor será: $D = (2n + 1)d$.

As camadas adjacentes são espiraladas em direções opostas, resultando numa aderência completa das camadas uma sobre as outras. Normalmente os cabos condutores empregados em linhas de transmissão possuem fios de alumínio dispostos em camadas, com o sem alma de aço.

No primeiro caso, tem-se o cabo de alumínio reforçado por um núcleo, ou alma, de aço, ACSR, (Aluminium Conductor Steel Reinforced, ou CAA, cabo de alumínio com alma de aço); no segundo caso, o cabo possui somente fios de alumínio encordoados, constituindo-se no cabo tipo CA, cabo de alumínio.

A consideração de um material ferromagnético no núcleo do cabo condutor acarreta implicações elétricas no desempenho deste. Este núcleo aumentará as perdas devido à histerese e correntes parasitas, aumentando a indutância da linha.

Uma camada de fios condutores encordoados de forma helicoidal constitui, em essência, um longo de solenóide. A corrente a fluir nestes fios irá produzir uma componente de força magnetomotriz, atuando ao longo do eixo do solenóide. Nestes termos, a colocação de um núcleo de material ferromagnético concorre para aumentar o fluxo oriundo desta força magnetomotriz.

Este aumento de fluxo fará crescer as perdas por correntes parasitas nos fios de alumínio, que, somadas às perdas no próprio núcleo, aumentarão as perdas de energia no cabo condutor. O número de camadas irá influenciar este aumento de impedâncias do cabo; um número par de camadas pode, por compensação de fluxos, anular as f.m.m. agindo no núcleo, impedindo a majoração da impedância do cabo.

Testes efetuados pela ALCOA, e, 1960, mostraram ainda que este efeito será desprezível para cabos com número ímpar de camadas acima de 5. Assim os efeitos

apreciáveis, da ordem de uns poucos por cento, restringem-se a cabos de alumínio com alma de aço com uma ou três camadas.

No Brasil, a nomenclatura empregada para identificar os condutores de alumínio ou alumínio-aço segue padrões canadenses, com os cabos CA sendo codificados por nomes de flores e os cabos CAA por nomes de pássaros, em inglês.

Como exemplo, o cabo CAA HAWK (falcão) compreende o cabo com área de alumínio 477MCM, 26 fios de alumínio em 2 camadas e 7 fios de aço. Já o cabo CA VIOLET, compreende o cabo com área de 715,5MCM, 37fios e 3 camadas de alumínio sobre o condutor central.

Em linhas de transmissão a partir de 230kV, empregam-se feixe de condutores por fase, procurando melhorar as características técnicas desta.

As vantagens deste procedimento prendem-se à necessidade de redução do gradiente de tensão e conseqüentemente melhoria nos padrões de rádio-interferência, diminuindo os seus efeitos. Adicionalmente, reduz-se a reatância da linha, com a conseqüente majoração de potencia transmitida por esta.

A experiência tem mostrado que o tamanho do condutor é o fator mais importante no controle gradiente de tensa para linhas com feixes de 2 a 4 condutores, faixa mais usada.

O espaçamento entre os condutores do feixe é outro item de importância, afetando também o gradiente de potencial. O espaçamento normal varia de 10 a 30 vezes o diâmetro dos condutores do feixe, com valores usuais entre 400 a 460 mm. As linhas de 500kV da Eletrosul, por exemplo, são constituídas por quatro cabos ACSR de 636MCM, 26/7, GROSSEAK, por fase. Os condutores possuem as seguintes características:

- Diâmetro.....25,15 mm;

- Peso.....1,30kg/m;
- Carga de ruptura.....11300kg;
- Espaçamento entre os condutores.....457mm.

2.12 Isoladores e ferragens em linhas aéreas de transmissão

2.12.1 Isoladores

Os cabos condutores são ligados às estruturas de transmissão pelo emprego de isoladores, cuja função consiste em isolar eletricamente os condutores energizados das estruturas de transmissão e resistir às solicitações mecânicas e elétricas oriundas dos cabos condutores.

As solicitações impostas aos isoladores são de três tipos:

- Forças verticais devido ao peso próprio dos condutores;
- Forças horizontais axiais, oriundas da suspensão dos condutores sobre o solo, no sentido horizontal da linha;
- Forças horizontais transversais, no sentido perpendicular aos eixos longitudinais das linhas, devido à ação do vento sobre os cabos.

Por outro lado, as solicitações de natureza elétrica a que um isolador deve resistir são aquelas oriundas das tensões mais elevadas que podem ocorrer nas linhas:

- Tensão nominal e sobretensões à frequência industrial;
- Surtos de sobretensão de manobra, de curta duração, atingindo níveis de duas a cinco vezes a tensão nominal entre fase-terra;
- Sobretensões de origem atmosférica.

Um isolado deve ainda ser resistente às variações e poluições ambientais, suportar bem os choques térmicos, minimizar número de reposições ao longo do tempo e assegurar, pelo seu projeto e fabricação, uma distribuição balanceada de potenciais e, em consequência, tensões de descargas adequadas. A porcelana vitrificada, o vidro temperado e resinas sintéticas são matérias normalmente empregados na fabricação de isoladores.

É interessante salientar que as falhas de isolamento elétrico em um isolador podem ocorrer por perfurações do dielétrico (interior) ou em descarga externa, pelo ar que o rodeia. No primeiro caso, o isolador deve ser substituído. Basicamente são usados dois tipos de isoladores em linhas de transmissão.

- Isoladores de pino, monocorpo ou multicorpo, fixados à estrutura através de um pino de aço, cuja parte superior possui uma cabeça de chumbo filetada, sobre a qual se atarracha o isolador. São normalmente usados até tensões de 69kV.

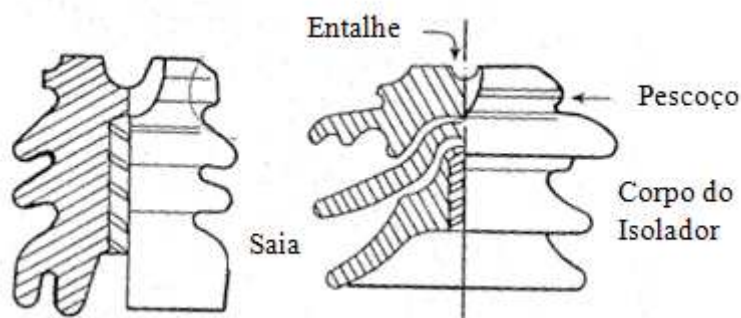


Figura 2.12 – Isoladores de pino

- Para tensões acima de 69kV, são usados isoladores do tipo suspensão, conforme a figura 2.12.

As características fundamentais que devem ter os isoladores de suspensão são as seguintes:

1. Resistência eletromecânica;
2. Carga máxima de trabalho ;
3. Resistência ao impacto;
4. Resistência aos choques térmicos;
5. Tensões disruptivas a seco e sob chuva à frequência industrial;
6. Tensões disruptivas sob impulso;
7. Tensão de perfuração;
8. Tensão de radiointerferência e corona;

Geralmente usado em linhas de alta-tensão e extra-alta-tensão, os isoladores tipo suspensão possuem as seguintes vantagens sobre os isoladores de pino:

- O custo inicial da cadeia de suspensão é mais baixo do que o isolador de pino equivalente. Também os custos de substituição são menores, visto que as unidades com defeito podem ser trocadas sem substituir toda cadeia;
- É mais fácil fabricar dielétricos homogêneos para os isoladores do tipo suspensão do que tipo pino;
- A flexibilidade das cadeias de suspensão reduz as tensões mecânicas e equaliza as tensões nos condutores adjacentes aos vãos.

A mesma cadeia de suspensão pode ser usada para suportar linhas de qualquer tensão, variando o número de isoladores na cadeia.

a) Teste de descarga a seco

A tensão de descarga é aquela em que o isolamento externo do isolador não consegue impedir a corrente elétrica de fluir pela parte externa deste, entre o condutor e o suporte do isolador (estrutura de transmissão).

O isolador limpo é montado na posição em que irá trabalhar e uma tensão é aplicada a este. Esta tensão é gradualmente aumentada até um mínimo especificado, devendo ser mantido por pelo menos 30 segundos sem ocorrência de descarga.

b) Teste de descarga sob chuva

É similar ao caso anterior, mas sob chuva de determinada intensidade e sob certa temperatura, especificando também o ângulo de inclinação da chuva.

c) Teste de deacarga sob poluição

É similar ao caso b, com condições de fumaça, ambiente salino, poeira e agentes químicos. A tensão mínima aqui especificada é a de ordem 50% daquela do caso b.

d) Teste de perfuração

Neste caso, procura-se projetar o isolador de modo que a descarga externa ocorra uma tensão mais baixa do que a tensão que ocasiona a perfuração. Esta perfuração permite a passagem de corrente dentro do isolador, acarretando sua substituição.

e) Teste de impulso

A razão do crescimento da onda de tensão imposta aos isoladores influencia o comportamento destes. A tensão de descarga a impulso é obtida pela aplicação de impulsos tendo a forma da gráfico 2.13.

Denomina-se razão de impulso a relação entre a tensão de descarga a impulso e a tensão de descarga a frequência industrial. Esta razão não deverá ser menor de que 1,3 para isoladores tipo suspensão e 1,4 para tipo pino.

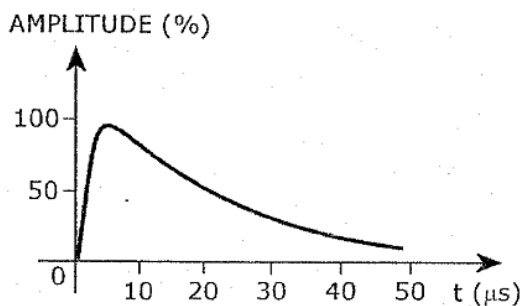


Gráfico 2.13 – Tensão de descarga de impulso

f) Teste mecânico

Uma unidade em suspensão é submetida a uma tensão mecânica de ordem de 1,2 vezes a máxima carga de trabalho e um isolador de pino a um momento fletor cerca de 2,5 a carga máxima de trabalho.

g) Teste de temperatura

O isolador é imerso alternativamente em banhos de água a 7°C e 70°C durante 1 hora.

g) Teste de porosidade

O isolador é pesado, imerso em água sob pressão por 24 horas e então repesado. Qualquer alteração do peso indica ingresso de água do isolador.

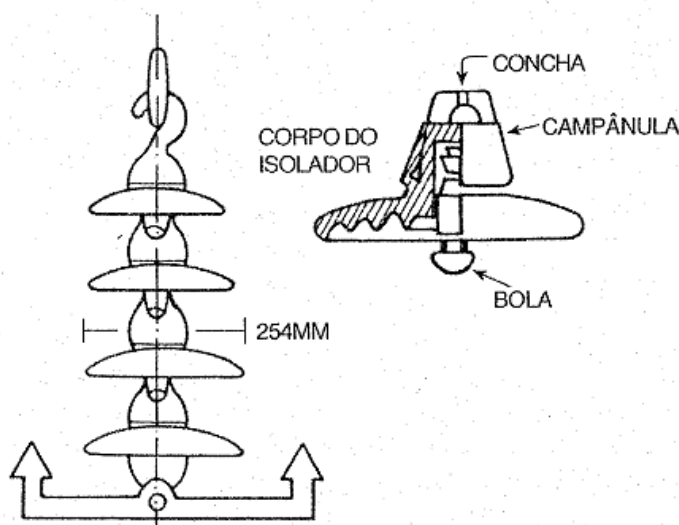


Figura 2.14 – Tensão de descarga de impulso

A figura 2.14 mostra um isolador tipo disco, peça correspondente de uma cadeia de suspensão. Nela estão mostradas as partes constituintes do isolador: concha, campânula, corpo do isolador e haste (bola). Introduzindo a haste (bola) de um isolador no concha seguinte, uma cadeia de isoladores com o comprimento necessário para uma determinada tensão pode ser construída. Diâmetro de 10 e 11 polegadas são usuais em cadeias cheias de isoladores de suspensão.

A distribuição de tensões ao longo de uma cadeia de isoladores não é linear, o isolador mais perto do condutor tem uma tensão aplicada sobre ele, maior que aquela aplicada sobre o topo.

Isto leva à necessidade de adicionar isoladores a mais na cadeia, de modo a evitar solicitações muito severas ao isolador mais próximo do condutor; colocar anéis de guarda e gaps condutores de modo a conduzir o arco de potencia que se segue a uma descarga de longe daq superfície do isolador, reduzindo a probabilidade de danifica-lo e melhorando a distribuição de tensão.

No circuito equivalente mostrada no figura 2.15, com uma cadeia de n isoladores idênticos, C é a capacitância por meio de cada isolador e C_0 é a capacitância de cada campânula à terra.

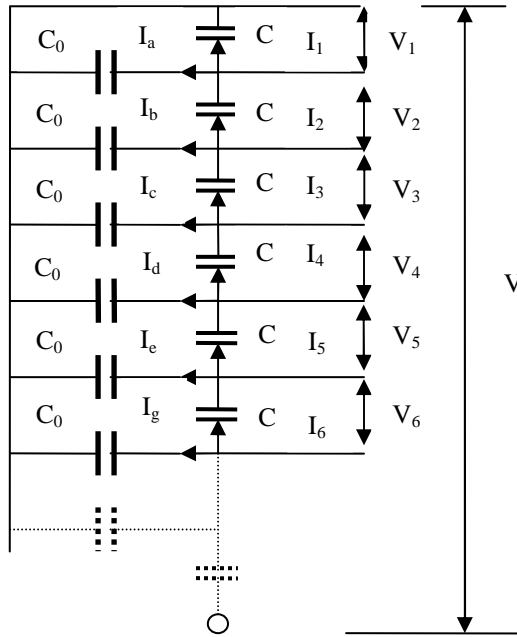


Figura 2.15 – Circuito equivalente a cadeia de suspensão.

Assumindo que a capacitância C e a capacitância em derivação C_0 são as mesmas ao longo da cadeia, $C = \Phi C_0 \Phi > 1$, temos que o processo pode ser continuado para qualquer número de isoladores, como por exemplo, para quatro e cinco isoladores respectivamente.

$$V_2 = \frac{I_2}{\omega \alpha C_0} = V_1 \left(1 + \frac{1}{\alpha} \right)$$

$$V_3 = \frac{I_3}{\omega \alpha C_0} = \frac{V_1}{\alpha} \left(3 + \alpha + \frac{1}{\alpha} \right) = V_3 = V_1 \left(1 + \frac{3}{\alpha} + \frac{1}{\alpha^2} \right)$$

$$V_4 = V_1 \left(1 + \frac{6}{\alpha} + \frac{5}{\alpha^2} + \frac{1}{\alpha^3} \right)$$

$$V_5 = V_1 \left(1 + \frac{10}{\alpha} + \frac{15}{\alpha^2} + \frac{7}{\alpha^3} + \frac{1}{\alpha^4} \right)$$

2.12.2 Ferragens

As ferragens constituem as peças que devem suportar os cabos e conectá-los às cadeias de isoladores; seu projeto requer cuidados especiais pois podem gerar manifestações devido ao efeito corona ou a radiointerferência.

O grampo de suspensão é um dispositivo que recebe o cabo condutor e o conecta à cadeia de suspensão, protegendo os filamentos condutores de esmagamento pela curvatura natural da parte inferior a sua calha. Normalmente é uma peça multiarticulada, permitindo movimento em qualquer direção a que esteja sujeito o condutor.

Um grampo de suspensão mostrado na figura 25, onde se percebe o engate concha-garfo entre a cadeia e o grampo.

As cadeias de ancoragem, além dos esforços normais que devem suportar as cadeias de suspensão, são dimensionadas para receber os esforços devidos aos tracionamentos dos cabos condutores. Normalmente são constituídas de uma simples coluna de isoladores ou diversas colunas em paralelo. A figura 2.16 mostra uma cadeia de ancoragem e a passagem do condutor, para nível de tensão 230kV, com as diversas partes constituintes do equipamento.

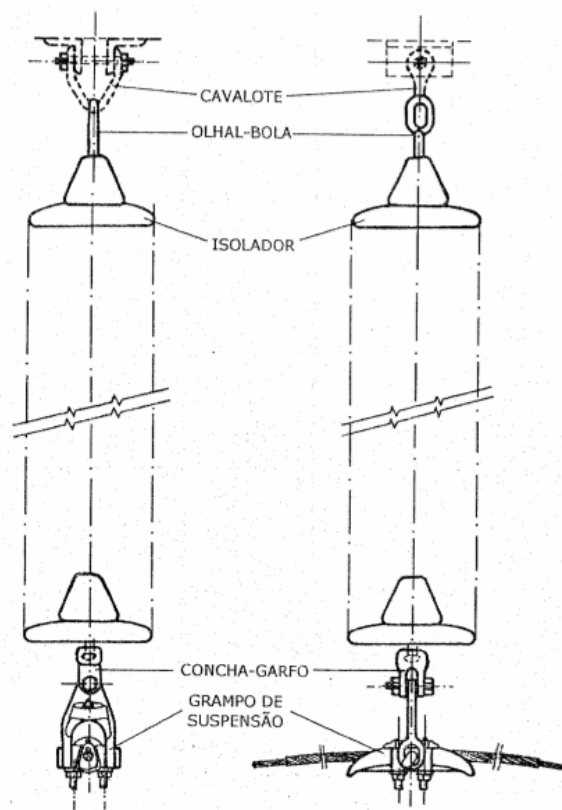


Fig.2.15 – Cadeia de suspensão simples 230kV.

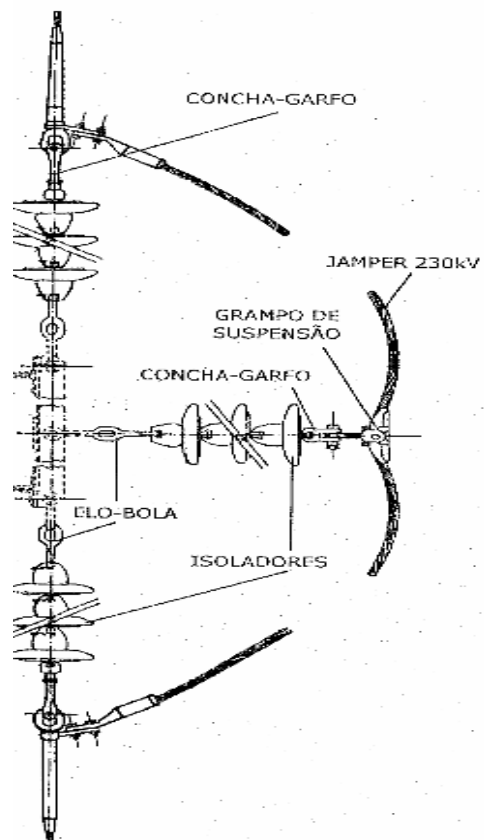


Figura 2.16 – Cadeia de ancoragem.

Convém ainda mencionar os espaçadores, dispositivos usados em linhas de transmissão que possuem mais de um condutor por fase, de forma a impedir que estes se toquem, procurando evitar danos nos condutores, como o tipo mostrado na figura 2.17.

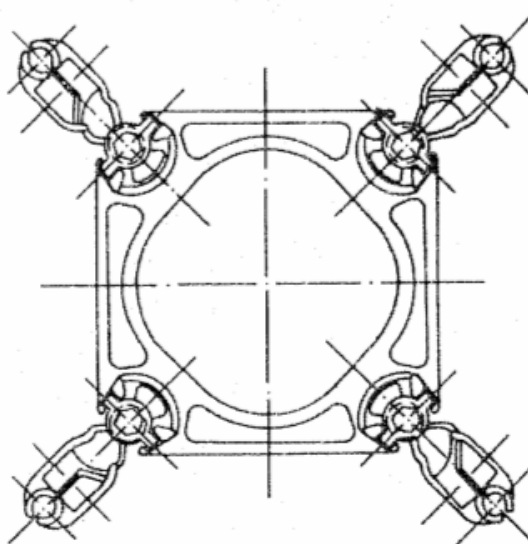


Figura 2.17 – Espaçadores.

2.13 Vibrações por carga de gelo nos condutores

A ação do vento sobre os condutores com camadas de gelo em sua superfície ocasiona o fenômeno conhecido como “galope”, onde os condutores movem-se com grande amplitude, com baixa frequência de oscilação, basicamente no plano vertical.

Este movimento pode resultar em danos aos condutores, principalmente os alinhados verticalmente, ocasionando descargas elétricas e a conseqüente interrupção do fornecimento de energia.

Adicionalmente, danos mecânicos e fadiga de componentes da linha de transmissão acarretam cuidados especiais e maior atenção do pessoal de manutenção.

Normalmente os projetistas de linhas colocam fatores de segurança em cima das distancias necessárias ao isolamento elétrico e segurança para manutenção, diminuindo o tamanho do vão, de forma a contrabalançar o impacto do “galope” sobre a operação do sistema.

Trabalhos realizados no campo, envolvendo amplo leque de condutores, mostraram que o uso de pêndulos amortecedores constitui-se em medida eficaz no controle do fenômeno para linhas com um condutor por fase. Também foram observados bons desempenhos dos pêndulos em linhas com feixe de condutores por fase.

A tabela 2.6 faz um apanhado de alguns resultados obtidos em linhas da Ontário Hydro, no Canadá, com o uso dos pêndulos amortecedores.

LT com um condutor for fase

Amplitude do galope – pico a pico

Sem pêndulo (m)	Com pêndulo (m)	Flecha com gelo e vento (m)
6,0	1,2	4,7
0,6	0,0	4,0
3,1	0,0	6,8
2,4	0,0	7,4
6,1	0,6	6,2
2,4	0,9	4,7
1,5	0,0	7,4

Tabela 2.6 – (Ontário Hydro)

Com o uso de amortecedores, a probabilidade de descargas fica reduzida em cerca de 1000 vezes, o que é bastante significativo em termos de “galope” se fixe nos condutores das fases, o cabo pára-raios também deve ser objeto de cuidados, conforme mostraram teses de campo realizados pela ALCOA, num condutor de aço 3/8”. O vão de teste tinha 240m de comprimento e as medições realizam a 40cm dos suportes.

Outros tipos de vibrações em linhas aéreas também são objetos de atenção e cuidado, destacando-se as eólicas e as de subvão. Nas eólicas ventos suaves na faixa de 1 a 7m/s ocasionam vibrações com frequências de 5 a 150Hz e amplitudes pequenas, da ordem do diâmetro do condutor.

Já as vibrações de subvão ocorrem sobre os ventos de 4 a 18m/s, com os condutores sendo induzidos pela esteira criada por outro condutor adjacente.

2.14 Estruturas de linhas aéreas de transmissão

As estruturas de transmissão têm por objetivo sustentar os cabos condutores e elementos associados, como isoladores, ferragens e cabos pára-raios. Podendo ser encaradas como uma viga vertical engastada no solo, a torre de transmissão fica sujeita aos seguintes esforços:

- Cargas verticais – são oriundas das componentes verticais do tracionamento dos cabos, peso próprio e peso dos acessórios de fixação dos cabos;
- Cargas horizontais e transversais – oriundas a ação do vento sobre os cabos; esforços devido a estaiamento. Componentes transversais de esforços de tracionamento dos cabos;
- Cargas horizontais longitudinais – aparecem devido a ação do vento em direção da linha e pelas componentes horizontais longitudinais dos tracionamento dos cabos.

Decorrem destes tipos de esforços que as estruturas de transmissão podem ser classificadas segundo dois tipos: estruturas autoportantes e estruturas estaiadas.

No primeiro caso, a torre transmite todos os esforços diretamente para suas fundações, o que não acontece com as torres estaiadas. Nesse tipo de estrutura, parte das solicitações é absorvidas pelos estais (tirantes) e a outra pelas fundações. Os cabos condutores podem ser colocados nas estruturas segundo disposição triangular, figura 2.18, horizontal e vertical, sendo esta última usadas em linhas a circuito duplo. Já na figura 2.19, são mostradas as partes principais de uma torre autoportante.

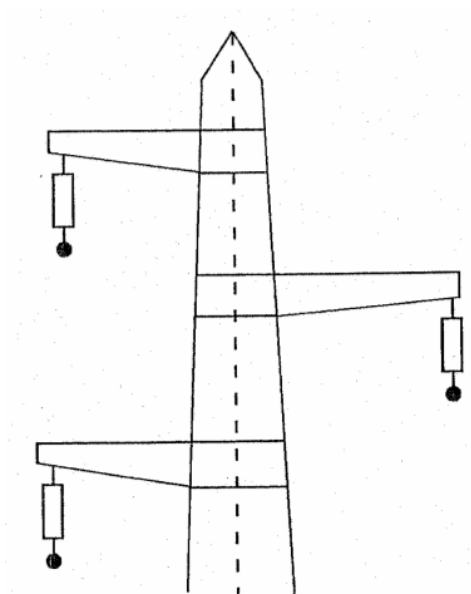


Figura 2.18 – Disposição triangular

Normalmente as torres de transmissão são fabricadas em aço-carbono, principalmente acima de 230kV. Para tensões mais baixas, a madeira e o concreto tem sido utilizados.

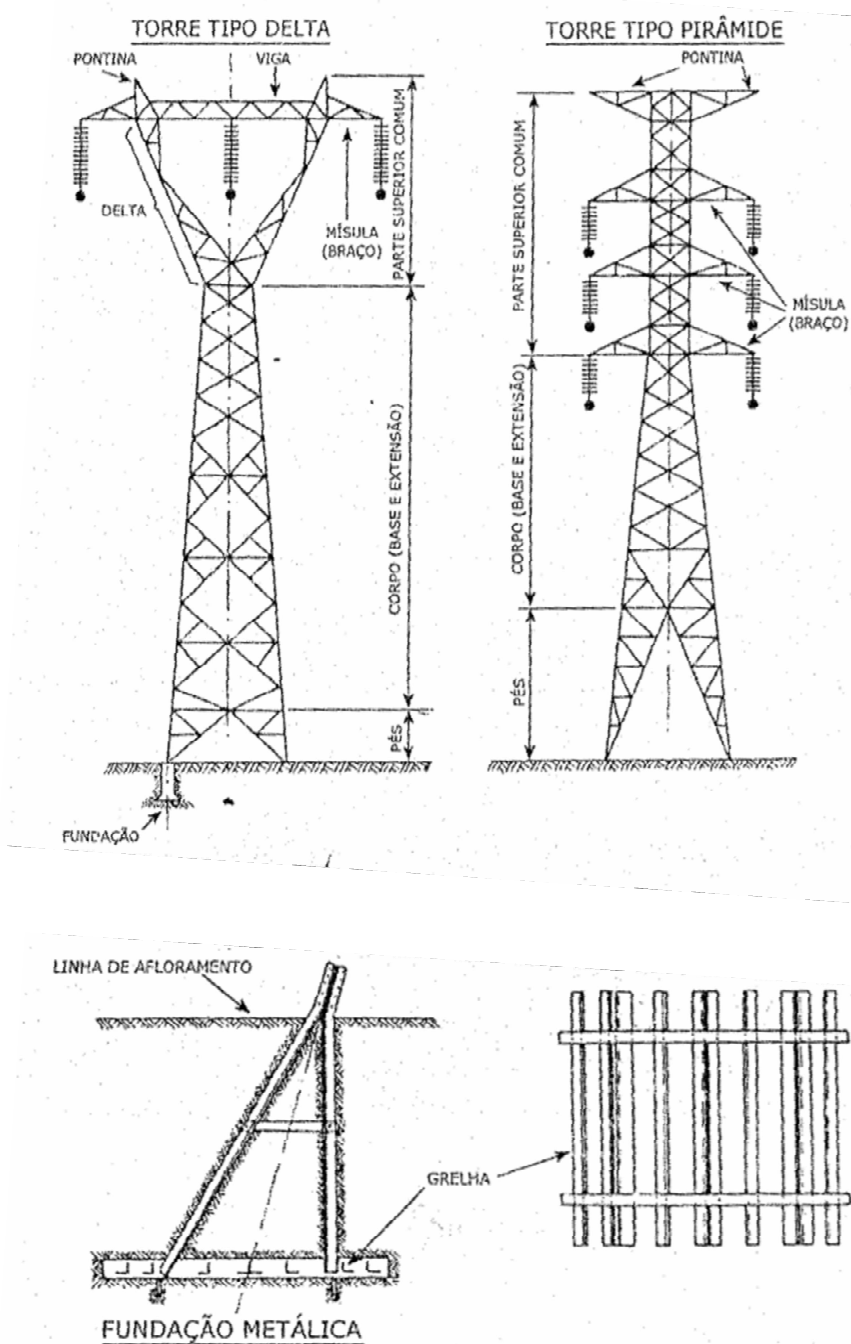


Figura 2.19 – Partes de uma torre

2.15 Tensão mecânica e flecha de um condutor

Na colocação dos condutores, durante a ereção de linhas aéreas, a flecha e a tensão mecânica de cada vão devem ser ajustadas a um valor calculado, de modo a assegurar que a carga imposta ao condutor nos pontos de suspensão não exceda valores permissíveis de carregamento, incluindo vento, gelo e fatores de segurança.

Um condutor apoiado entre dois pontos sem vento irá tomar a forma de uma catenária, figura 2.20.

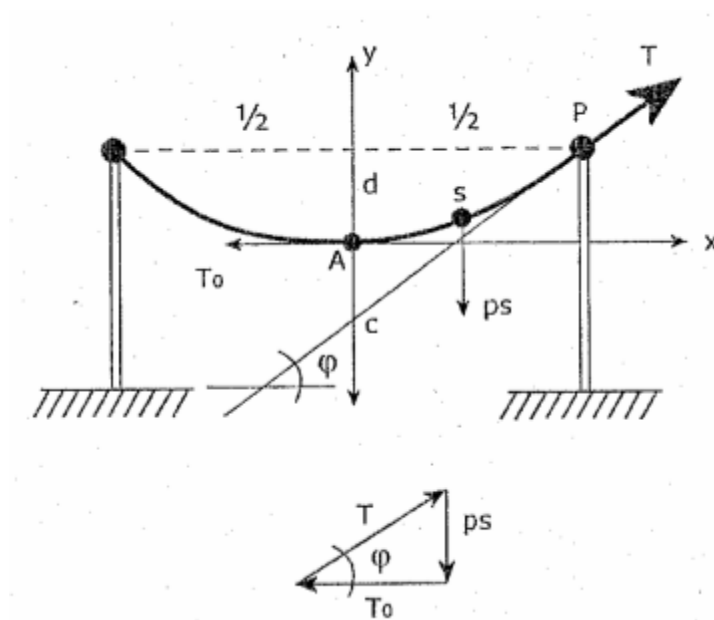


Figura 2.20 – Catenária.

O equilíbrio do trecho AP pressupõe o triângulo de forças T , T_0 e ps , sendo T a tensão no ponto P e T_0 a tensão no ponto A . Admitindo um peso por metro linear do condutor igual a p , o peso do condutor entre A e P será igual a $p.s$, com o comprimento, em metros, do trecho curvo AP .

Desta abordagem, conhecida como aproximação parabólica da catenária, pode-se, pois, finalmente, calcular a expressão para a tensão imposta ao condutor no ponto de máxima flecha:

$$T_0 = \frac{pl^2}{8d}$$

Embora aproximações, são suficientes precisas para a maioria das aplicações práticas, onde a flecha é muito menor que o comprimento do vão.

$$S \approx l + \frac{8}{3l} d^2,$$

Expressão que fornece o comprimento do condutor em função do comprimento do vão e do valor da flecha máxima.

Quando a carga sobre o condutor incluir a pressão do vento agindo transversalmente e ainda o peso do gelo, tem que se compor o valor p , igual a soma do peso do condutor com o peso do gelo por unidade de comprimento, com o valor v , pressão do vento, originando a resultante $R = \sqrt{P_{12}^2 + V^2}$ passando esta expressão a substituir o valor p das formulas anteriores.

$$P_1^2 + v^2$$

Capítulo 3

3.0 Visual Basic

3.1 Introdução

O Visual Basic 2005 é um ótimo programa para construir aplicativos para sistema operacional Windows, embora a linguagem de programação que iremos abordar seja a Visual Basic, o ambiente de desenvolvimento que você vai utilizar para escrever seus programas é chamado Microsoft Visual Studio Integrated Development Environment, ou IDE para abreviar. O Visual Studio é um work-shop de programação poderoso e personalizável que

contém todas as ferramentas necessárias para construir programas robustos para Windows e para Web de maneira rápida e eficiente.

O Visual Basic é um membro do Visual Studio, onde ele além de compartilhar o ambiente de desenvolvimento do Visual Studio também compartilha com o Visual C++ 2005, o Visual C# 2005, o Visual J# 2005 e várias outras ferramentas de programação. Embora o Visual Basic, o Visual C++, o Visual C# e o Visual J#, ainda sejam linguagens de programação diferentes, eles compartilham o mesmo com o IDE.

Como parte de seu novo ambiente de desenvolvimento, o Visual Studio oferece um novo painel Start Page, como mostrado na figura 3.0, que permite abrir arquivos novos ou existentes. Links adicionais no painel Start Page fornecem acesso a sites relacionados ao Visual Studio, notícias e artigos sobre programação em Visual Studio e contatos dentro da comunidade de desenvolvimento do Visual Studio.

O ambiente de desenvolvimento do Visual Studio contém várias ferramentas de programação novas e modificadas. A janela project agora é chamada de Solution Explorer e as informações de ajudas são obtidas por meio de uma ferramenta chamada Microsoft Document Explorer. A maioria das janelas de ferramentas de programação (incluindo a ToolBox) tem um recurso Auto Hide que oculta as ferramentas como uma guia quando ela não é necessária. Auto-ocultar ferramentas economiza espaço para o projeto e codificação no ambiente de desenvolvimento.

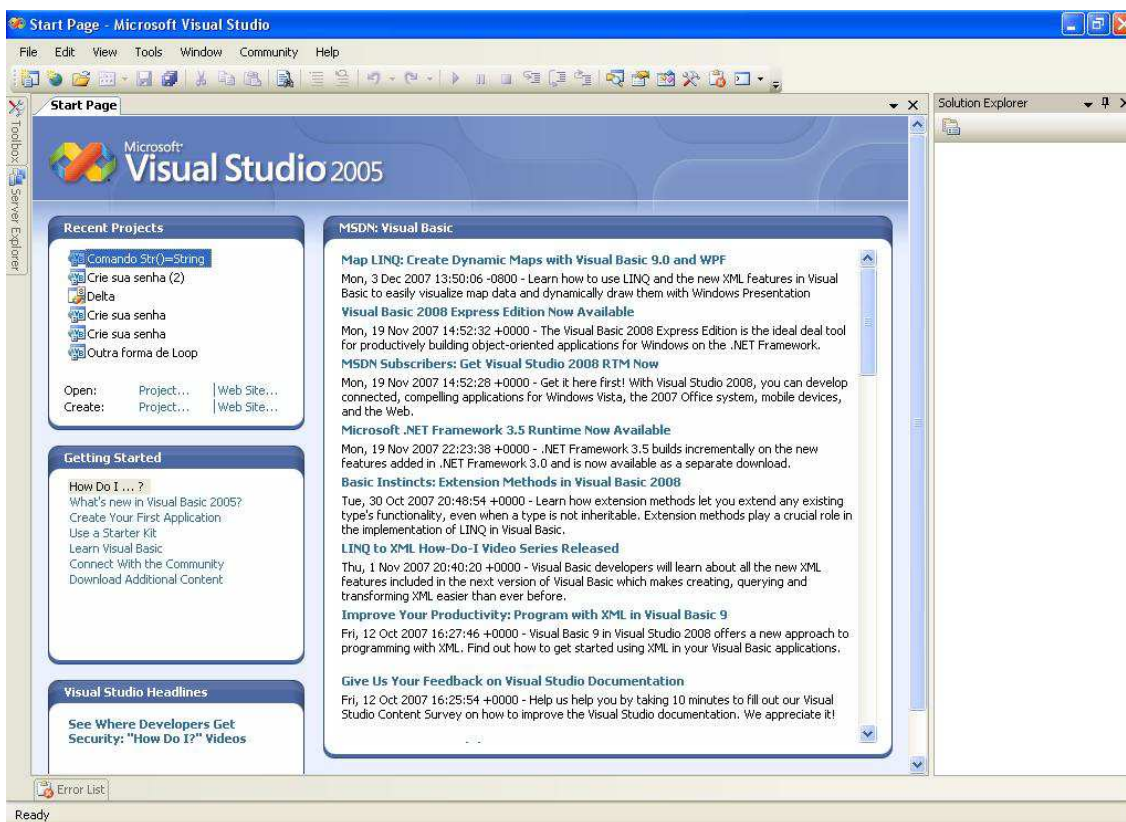


Figura 3.0 – Painel Start do Visual Basic.

3.2 Requisitos do Sistema

Você precisa dos seguintes hardwares para a instalação do software Visual Studio Basic 2005:

- Microsoft Windows XP com Service Pack 2, Microsoft Windows Server 2003 com o Service pack 1 ou o Microsoft Windows 2000 com o Service Pack 4.
- Processador Pentium de 1GHz ou compatível.
- 384 MB RAM (recomenda-se 512 MB ou mais)
- Placa de vídeo (com resolução de 800 x 600 ou superior) e monitor com pelo menos 256 cores (recomenda-se 1,024 x 768 e High Color de 16 bits)
- Microsoft Mouse ou dispositivo indicador compatível.

3.3 As ferramentas do Visual Studio

Nesse ponto, iremos abordar o Visual Studio IDE e identificar algumas ferramentas de programação e janelas que você irá utilizar para o desenvolvimento dos seus programas. Coletivamente esses recursos são os componentes que você irá utilizar para construir, organizar e testar programas do Visual Basic. Algumas das ferramentas de programação também ajudam a aprender mais sobre os recursos em seu sistema, incluindo o mundo maior de conexões de banco de dados e sites web disponíveis para você. Há várias ferramentas de ajuda poderosa.

A barra de menus fornece acesso a maioria dos comandos que controlam o ambiente de desenvolvimento. Os menus e comandos como em todos os programas baseados no Windows e barra de ferramentas Standard, uma coleção de botões que serve como atalhos para executar comandos e controlar o Visual Studio IDE. Se você já utilizou Excel ou Word, a barra de ferramenta deve ser um conceito familiar. Para ativar um botão na barra de ferramentas, utilize o mouse para clicar no botão. Você pode ver a lista completa de barra de ferramenta disponíveis clicando com o botão direito do mouse na barra de ferramentas do IDE.

Ao longo da parte inferior da tela está localizada a barra de tarefas Windows. Você pode usar a barra de tarefas para alternar entre vários componentes do Visual Studio e ativar outros programas baseados no Windows. Você também pode ver ícones da barra de tarefas do Microsoft Internet Explorer, utilitários e outros programas instalados no sistema.

A figura 3.1 mostra algumas ferramentas e janelas no Visual Studio IDE. Não se preocupe que no decorrer você irá aprenderá mais sobre esses elementos (e como ajustar suas visualizações)

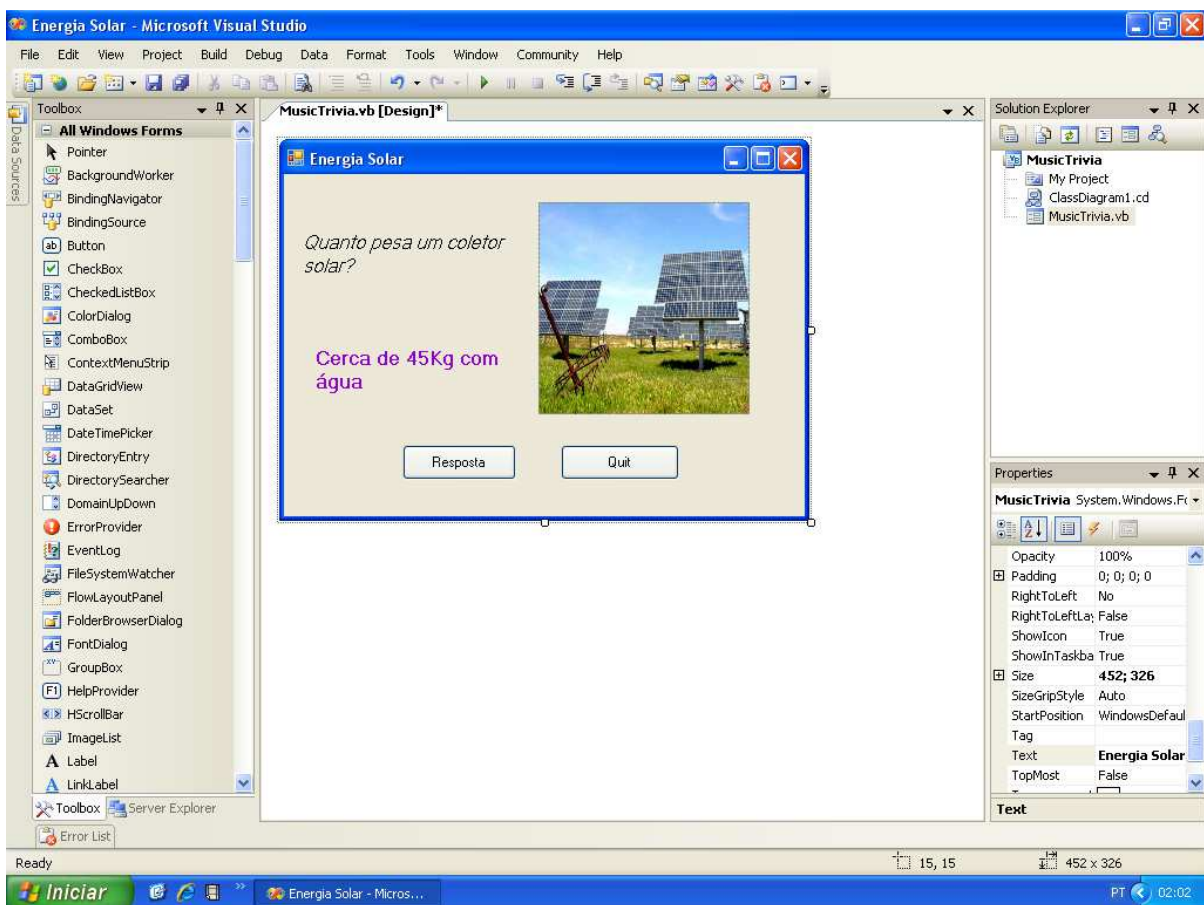


Figura 3.1 - Ferramentas e janelas do Visual Studio.

As principais ferramentas visíveis neste Visual Studio IDE são os Designers, Solution Explorer, a janela Properties, a Toolbox e a janela Output. Também temos as ferramentas mais especializadas como o Server Explorer e o Object Browser, ou elas podem aparecer como guias dentro do IDE.

3.4 Desenvolvimento

3.4.1A janela Properties

Você utiliza a janela properties para alterar as características, ou configurações de propriedades, dos elementos da interface do usuário em um formulário. Uma configuração de propriedade é uma qualidade de um dos objetos do programa. Você pode alterar configurações de propriedades usando a janela Properties enquanto estiver criando sua

interface de usuário ou pode adicionar código de programa via o Code Editor para alterar uma ou mais configurações de propriedades enquanto seu programa estiver executando. Por exemplo, a pergunta que o programa Energia Solar exibe pode ser modificada para aparecer em uma fonte ou tamanho de fonte diferente ou com outro alinhamento. (Com o Visual Studio, você pode exibir texto em qualquer fonte instalada no seu sistema, assim como no Excel ou Word).

A janela Properties contém uma lista Object que relaciona todos os objetos da interface de usuários no formulário. Você pode clicar em um de dois botões convenientes para ver as propriedades alfabeticamente. Agora iremos desenvolver o programa Energia Solar

3.4.2 Altere uma propriedade

1. Clique no objeto Label1 do formulário. (Label1 contém no texto “Quanto pesa um coletor solar?”).

Para trabalhar com um objeto em um formulário, você deve primeiro selecionar o objeto. Ao selecionar um objeto, redimensione as alças para circundá-lo, e as configurações de propriedade para o objeto são exibidas na janela properties.

2. Clique no botão Properties Window na barra de ferramenta Standard.

A janela Properties pode estar visível ou não no Visual Studio, dependendo de como foi configurada e utilizada no sistema. Normalmente ela aparece abaixo do Solution Explorer à direita do ambiente de desenvolvimento. (Se ela estiver visível, você não precisa clicar no botão, mas deve clicar na janela para ativá-la.).

Você verá uma janela parecida como a figura 3.2:



Figura 3.2 - Janela Properties.

A janela properties lista todas as configurações de propriedade para o primeiro objeto de rótulo (Label1) no formulário. Os nomes da propriedade são listados na coluna esquerda da janela e a configuração atual para cada propriedade é listada na coluna direita. Como há tantas propriedades, o Visual Studio as organiza em categorias e as exibe na visualização de estruturas de tópicos. Se uma categoria tiver um sinal de adição (+) ao lado dela, você pode clicar no rótulo de coleção para exibir todas as propriedades nessa categoria. Se uma categoria tiver um sinal de subtração (-) ao lado dela, todas as propriedades estão visíveis, mas é possível ocultar a lista sob o nome da categoria clicando no sinal de subtração.

3. Role pela caixa de listagem da janela Properties até a propriedade Font ficar visível.

A janela Properties rola como uma caixa de listagem comum. Se você estiver na visão categorizada, Font está na categoria Appearance.

4. Clique no nome de propriedade Font (na coluna esquerda).

A fonte atual (Microsoft Sans Serif) é parcialmente exibida na coluna direita e um botão com três pontos aparece depois do nome parcial fonte. Esse botão é chamado de botão de reticências e indica que uma caixa de diálogo esta disponível para personalizar a configuração de propriedade.

5. Clique no botão de reticências Font na janela Properties.

O Visual Studio exibe a caixa de diálogo Font, que pode ser utilizada para especificar novas características de formatação para o texto no rótulo selecionado em seu formulário. A caixa de diálogo Font contém mais de uma opção de formatação.

6. Mude o tamanho da fonte de 11 para 12 pontos e, então, mude o estilo de fonte de Regular para Italic. Clique em Ok para confirmar suas alterações.

Agora altere uma configuração de propriedade para o objeto Label2 (o rótulo que contém o texto “Cerca de 45Kg com água”)

7. No Designer, clique no segundo objeto de rótulo (Label2).

Quando você selecionar o objeto, as laças de redimensionamento aparecem em volta dele.

Clique na propriedade Font da janela Properties.



Figura 3.3 – Ambiente de trabalho do Visual Studio

Como mostra a figura 3.3, o objeto Label2 tem seu próprio conjunto único de configurações de propriedade. Embora os nomes de propriedade sejam os mesmos do objeto Label1, os valores nas configurações de propriedade são distintos e permitem ao objeto Label2 atuar independente no formulário.

9. Clique no botão de reticências Font, configure o estilo de fonte como Bold e então clique em Ok.

10. Role para a propriedade ForeColor na janela Properties e, então, clique nela na coluna esquerda.

11. Clique na seta Fore Color na coluna direita, clique na guia Custom e, então, clique em uma cor roxa escura.

O texto no objeto Label2 está agora formatado como negrito e roxo no formulário.

3.5 Movendo e redimensionando as ferramentas de programação

Para mover uma das janelas de ferramentas do Visual Studio, como mostra a figura 3.4, simplesmente clique na barra de título e arraste o objeto para uma nova localização. Se você alinhar uma janela ao longo da borda de outra janela, ela se anexa a esta janela, ou se encaixa. Janelas encaixáveis são vantajosas porque permanecem sempre visíveis. (Elas não ficam ocultas atrás de outras janelas). Se você quiser ver mais de uma janela encaixada, simplesmente arraste uma de suas bordas para visualizar mais conteúdo.

Se quiser fechar completamente uma janela, clique no botão Close no canto superior direito da janela. Você pode sempre abrir a janela novamente clicando no comando apropriado do menu View.

Se quiser uma opção em algum lugar entre encaixar e fechar janela, você poderia tentar auto-ocultar uma janela de ferramenta no lado do Visual Studio clicando no pequeno botão com um ícone de alfinete Auto Hide à direita da barra de título de ferramenta. Essa ação remove a janela da posição encaixada e coloca o título de ferramenta na borda do ambiente de desenvolvimento. Ao auto-ocultar uma janela, você notará que a janela de ferramentas permanece visível contanto que mantenha o ponteiro do mouse na área da janela. Quando você move o mouse para outra parte qualquer, a janela se desliza para fora do campo de visão da tela.

Para restaurar uma janela que você auto-ocultou, clique na guia de ferramentas da borda do ambiente de desenvolvimento ou mantenha o mouse sobre a guia. (Você pode reconhecer uma janela que está auto-ocultada porque o alfinete em sua barra de título está apontada para o lado). Mantendo o ponteiro do mouse sobre o título, em outras palavras, para exibir rapidamente uma janela auto-oculta, clique em seu guia, marque ou configure as informações do que você precisa e, então, mova o mouse para fazer a janela desaparecer. Se alguma vez precisar que a ferramenta seja exibida permanentemente, clique de novo no botão

alfinete Auto Hide para que a ponta do alfinete aponte para baixo e a janela permanece então visível.

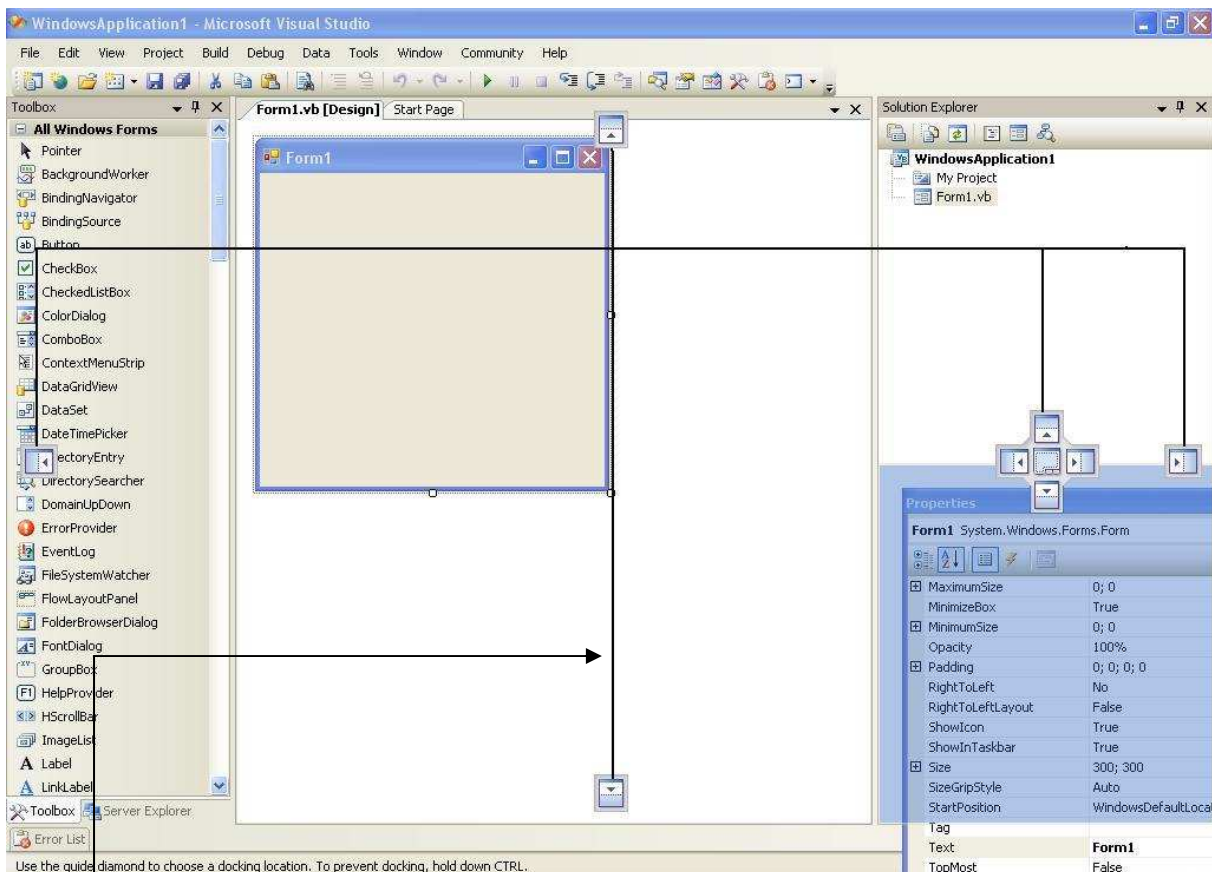


Figura 3.4 - Movendo e redimensionando as ferramentas de programação

Guias de encaixe

3.6 Trabalhando com controles da Toolbox

Os controles de formulários do Windows são projetados especificamente para construir aplicações Windows e estão organizados na guia All Windows Form da Toolbox, embora muito dos controles também sejam acessíveis em guias como Common Controls, Containers e Printing. Utilizamos alguns desses controles no programa anterior. Agora vamos aprender outros controles, incluindo as ferramentas que utiliza para contruir aplicações.

3.6.1 A utilização dos controles: O programa Hello World

Nos dias da programação baseada em caractere, Hello World era normalmente um programa de duas ou três linhas digitadas em editor de programa e era construído com um compilador separado e independente. Mas, com o advento de complexos sistemas operacionais e ferramentas de programação, o típico Hello World vem a ser um programa mais sofisticado que contém dezenas de linhas e requer várias ferramentas de programação para sua construção. Felizmente, criar um programa Hello World é ainda bem simples com o Visual Studio e Visual Basic 2005. Você pode construir uma completa interface de usuário criando dois objetos, configurando duas propriedades e inserindo uma linha de código.

3.6.2 Crie um programa Hello World

1. Início o Visual Studio 2005 se ele não estiver aberto ainda.

2. No menu File, clique em New Project.

3. Certifique-se de que o tipo de projeto do Visual Basic e a categoria Windows estão selecionadas e, então, clique no modelo de aplicação Windows

4. Clique em OK para criar seu novo projeto.

Os controles que serão utilizados nesse programa, Button e TextBox, são visíveis na Toolbox, se suas ferramentas de programação estão configuradas diferentemente, reserve um tempo para organiza-las.

5. Clique no controle TextBox na guia Common Controls da Toolbox.

6. Desenhe uma caixa de texto.

As caixas de textos são usadas para exibir texto em um formulário ou obter entrada de usuário enquanto seu programa está em execução. A maneira como uma caixa de texto funciona depende de como você configura suas propriedades e como referencia a caixa

de texto no código de programa. Nesse programa, um objeto caixa de texto será utilizado para exibir a mensagem “Hello, world!” quando vcpe clica em um objeto botão no formulário.

Agora iremos adicionar um botão no formulário.

7. Clique no controle Button da ToolBox.

8. Desenhe um botão embaixo da caixa de texto no formulário.

Como você viu no: “Escrevendo seu primeiro programa”, os botões são utilizados para obter a entrada mais básica de um usuário. Quando um usuário clica em um botão, ele está solicitando que o programa realize uma ação específica imediatamente. Em termos do Visual Basic, o usuário está utilizando o botão para criar um evento que precisa ser processado no programa. Botões típicos em um programa são o botão OK, no qual um usuário clica para aceitar uma lista de opções e indicar que ele está pronto para prosseguir; o botão Cancel, no qual um usuário clica para descartar uma lista de opções; e o botão Quit, em que um usuário clica para sair do programa. Em cada caso, você deve utilizar esses botões de maneira padrão para que funcione de modo esperado quando o usuário clica neles. As características de um botão (como as de todos os objetos) podem ser modificadas como configurações de propriedade e referência ao objeto código de programa.

9. Configure a seguinte propriedade para o objeto botão utilizando a janela

Properties:

Objeto	Propriedade	Configuração
Button1	Text	“OK”

Tabela 3.0

10. Dê um clique duplo no botão OK e digite a seguinte instrução de programa entre as instruções Private Sub Button1_Click e End Sub no Code Editor:

```
TextBox1.Text = "Hello, world!"
```

A instrução que inserimos muda a propriedade Text da caixa do texto para “Hello, World!” quando o usuário clica no botão em tempo de execução. (O sinal igual (=) atribui tudo entre aspas à propriedade Text do objeto TextBox1).

3.6.3 Execute o programa Hello World

1. Clique no botão Start Debugging na barra de ferramentas Standard.
2. Clique no botão OK do formulário. O programa exibe a saudação “Hello, world” na caixa de texto, como mostra a figura 3.5:



Figura 3.5 - Formulário do programa Hello World

Quando você clicou no botão Ok, o código de programa mudou a propriedade Text da caixa de texto TextBox1 vazia para “Hello, world!” e exibiu esse texto na caixa. Se

não obteve esse resultado, os procedimentos na seção anterior e construa o programa novamente.

3. Clique no botão Close no canto superior direito da janela de programa Hello World para parar o programa.

4. Clique no botão Save All na barra de ferramentas Standard para salvar seu projeto no disco

3.6.4 Utilizando o controle DateTimePicker

Alguns controles do Visual Basic exibem informações e outro as coletam do usuário ou processam dados no bastidores. Agora, iremos trabalhar com o controle DateTimePicker, que solicita ao usuário uma data ou hora utilizando um calendário gráfico com setas de rolagem. Embora a sua utilização de controle do Visual Basic pode fazer por você automaticamente e de como você processa as informações que vêm deles.

3.6.5 O programa Birthday

O programa Birthday utiliza um controle DateTimePicker e um controle Button para solicitar ao usuário a data de seu aniversário. Ele então exibe essas informações em uma caixa de mensagem.

3.6.6 Construa o programa Birthday

1. No menu File, clique em New Project.
2. Crie um novo projeto Visual Basic Windows Application.
3. Clique no controle DateTimePicker na Toolbox.
4. Desenhe um objeto seletor data/hora no meio do formulário.

O objeto seletor de data/hora por padrão exibe a data atual, mas você pode ajustar a data exibida alterando a propriedade Value do objeto. Exibir a data é um guia útil de design, isso permite dimensionar o objeto seletor de data/hora apropriadamente ao criá-lo.

5. Clique no controle Button na Toolbox e, então, adicione um novo objeto botão embaixo do seletor data/hora.

Você utilizará esse botão para exibir sua data de nascimento e verificar se o seletor de data/hora funciona corretamente.

6. Na janela Properties, mude a propriedade Text do objeto botão para que mostre meu aniversário.

7. Agora você adicionará algumas linhas de código de programa a um procedimento associado com o objeto botão. Esse é um procedimento de evento porque ele executa quando um evento, como clique de mouse, ocorre, ou dispara, no objeto.

8. Dê um clique duplo no objeto botão do formulário para exibir seu procedimento de evento padrão e, então, digite as seguintes instruções de programa entre as instruções Private Sub e End Sub no procedimento de evento Button1_Click:

```
MsgBox("Your birth date was " & DateTimePicker1.Text)
    MsgBox("Day of the year: " & _
        DateTimePicker1.Value.DayOfYear.ToString())
```

Essas instruções de programa exibem duas caixas de mensagens (caixas de diálogo pequenas) com informações do objeto seletor data/hora. A primeira linha utilizada a propriedade Text do seletor de data/hora para exibir informações sobre a data de nascimento que você seleciona ao utilizar o objeto em tempo de execução. A função MsgBox exibe o valor da String “Your birth data was” além do valor textual mantido na propriedade Text do

seletor de data/hora. Esses dois fragmentos de informações são interconectados pelo operador de concatenação de Strings (&). A segunda e terceira linha formam coletivamente uma instrução de programa e foram quebradas pelo caractere de continuação de linha (_) porque a instrução era longa demais.

9. Clique no botão Save All para salvar suas alterações no disco.

3.6.7 Execute o programa Birthday

1. Clique no botão Start Debugging na barra de ferramentas Standard.
2. Clique na seta do seletor de data/hora para exibir o objeto na visualização de calendário.

O formulário é parecido como mostra a figura 3.6:



Figura 3.6 – Formulário do programa Birthday

3. Clique na seta de rolagem esquerda para ver meses anteriores no calendário.
4. Clique no botão Mostre meu aniversário.

O Visual Basic executa o código de programa e exibe uma caixa de mensagem contendo o dia e ano do seu nascimento.

3.7 Uma palavra sobre a terminologia

Utilizei até aqui vários elementos para descrever itens em um programa Visual Basic. Iremos listar vários deles agora para eliminar qualquer confusão.

- **Instrução de programa** → Uma instrução de programa é uma linha de código em um programa Visual Basic, uma instrução auto contida executada pelo compilador do Visual Basic que realiza trabalho útil dentro da aplicação. As instruções de programa podem variar em comprimento (algumas contêm apenas uma palavra-chave do Visual Basic!), mas todas as instruções de programa devem seguir as regras de sintaxe definidas e impostas pelo compilador do Visual Basic. No Visual Studio 2005, as instruções de programa podem ser compostas de palavras-chave, propriedade, nomes de objetos, variáveis, números, símbolos especiais e outros valores.

- **Palavra-chave** → Uma palavra chave é uma palavra reservada dentro da linguagem Visual Basic que é reconhecida pelo compilador Visual Basic e faz um trabalho útil. (Por exemplo, a palavra-chave End pára a execução do programa). As palavras-chave são um dos blocos de construção básicos de instruções de programa; elas trabalham com objetos, propriedades, variáveis e outros valores para formar linha de código completo e, portanto instruções para o compilador e o sistema operacional.

- **Controle**→ Um controle é uma ferramenta utilizada para criar objetos em um formulário do Visual Basic. Você seleciona controles da Toolbox e os utiliza para desenhar objetos com o mouse em um formulário. Você utiliza a maioria dos controles para criar elementos da interface de usuário, como botões, caixas de figura e caixa de listagem.

- **Objeto**→ Um objeto é um elemento de interface de usuário que você cria em um formulário do Visual Basic, o formulário em si também é um objeto. Você pode mover, redimensionar e personalizar utilizando configurações de propriedade de objetos.

- **Espaço de nomes**→ Um espaço de nomes é uma biblioteca hierárquica de classes organizado sob um nome único, como System.Windows ou System.Diagnostics. Para acessar as classes e objetos subjacentes dentro de um espaço de nomes, você coloca uma instrução Imports na parte superior do seu código de programa. Todo projeto no Visual Studio também tem um espaço de nomes raiz, que é configurado utilizando página Properties do projeto. Os espaços de nomes são referidos como biblioteca de objetos ou biblioteca de classes em livros e documentação do Visual Studio.

- **Propriedade**→ Uma propriedade é um valor, ou característica, possuído por um objeto. Por exemplo um objeto botão tem uma propriedade Text para especificar o texto que aparece no botão e uma propriedade Image para especificar o caminho para um arquivo de imagem que deve aparecer na face do botão. No Visual Basic, as propriedades podem ser configuradas no momento do projeto utilizando a janela Properties ou

no momento da execução utilizando instrução no código de programa. No código, o formato para configurar uma propriedade é:

```
Objeto.Propriedade = valor
```

Onde objeto é o nome do objeto que você está personalizando, propriedade é a característica que você quer alterar o valor é a nova configuração da propriedade.

- **Procedimento de evento** → O procedimento de evento é um bloco de código que é executado quando um objeto é manipulado em um programa. Por exemplo, quando o objeto Button1 é clicado, o procedimento de evento Button_Click é executado. Os procedimentos de eventos, em geral, avaliam e configuram propriedades e utilizam outras instruções de programa para realizar o trabalho do programa

- **Método** → Um método é uma instrução especial que realiza uma instrução ou serviço para um objeto particular em um programa. No código de programa, a notação para utilizar método é:

```
Objeto.método(valor)
```

Onde objeto é o nome do objeto com o qual você quer trabalhar. Método é a ação que você quer realizar e valor é um argumento opcional a ser utilizado pelo método.

Capítulo 4

4.0 Desenvolvimento do programa modelagem de linhas de transmissão

Inicie o Visual Studio, no menu File, clique em New Project, após isso, um novo projeto de aplicação Windows chamado Linhas de Transmissão. Clique no controle MenuStrip na guia Menus & Toolbars da Textbox e, então, desenhe um controle de menu no formulário, não se preocupe com a localização, o Visual Basic moverá o controle e o redimensionará automaticamente.

O objeto menu de faixa não aparece no formulário, mas abaixo dele. Além do objeto menu de faixa na bandeja de componentes, o Visual Basic exibe uma representação visual do menu que você criou na parte superior do formulário. O Tag Type Here o encoraja a clicar no Tag e inserir um título no menu. Depois de inserir o primeiro título de menu, você pode inserir títulos de submenu e outros nomes de menu pressionando as teclas e setas e digitando nomes adicionais. Melhor ainda, você pode voltar mais tarde para esse menu Designer em linha e editar o que você fez ou acrescentar itens de menus adicionais, o objeto menu de faixa é completamente personalizável e com ele você pode criar uma interessante interface de usuário baseada em menus. Sendo que será adicionado três menus.

1. Clique no Tag Type Here digite Linhas aéreas de transmissão e, então, pressione Enter.
2. Digite Isoladores para criar um comando para o menu Linhas aéreas de transmissão e, então, pressione Enter.
3. Digite Tensão mecânica e flecha de um condutor para criar um comando para o menu e, então, pressione Enter.

Agora adicionaremos o segundo menu chamado Parâmetros da LT de potência.

1. Digite Indutância em Linhas de transmissão para criar um comando para o menu chamado Parâmetros da LT de potência e ainda criaremos os sub-menus, de um clique e digite no Tag Type Here Linha trifásica de um

circuito simples e logo abaixo desse sub-menu, digite, Linha múltipla e Linha trifásica de circuito duplo.

Digite Capacitância em linhas de transmissão para criar um comando para o menu chamado Parâmetros da LT de potência e ainda criaremos os sub-menus, de um clique e digite no Tag Type Here Susceptância capacitiva em linhas de transmissão e logo abaixo desse sub-menu, digite, Linha trifásica de um circuito simples, Linha múltipla, Linha trifásica de um circuito duplo.

Agora adicionaremos o terceiro menu chamado Linhas de transmissão aéreas.

1. Digite Linha curta para criar um comando para o menu chamado Linhas de transmissão aéreas e, então, pressione Enter.
2. Digite Linha média para criar um comando para o menu e, então, pressione Enter.
3. Digite Linha longa para criar um comando para o menu e, então, pressione Enter.

Após isso clique em Form1.vb vai em Properties e em seguida Text e digite:

Linhas de transmissão.

4.1 Adicione um objeto rótulo ao formulário

1. Clique no controle Label na Toolbox.
2. Crie um rótulo no meio do formulário.

O objeto rótulo aparece no formulário e traz o nome Label1 no código de programa. Se, mais tarde, vir que esse rótulo é muito pequeno para o texto caber, você pode redimensioná-lo como redimensionaria qualquer objeto em um formulário.

3. Configure as seguintes propriedades para o rótulo:

Objeto	Propriedade	Configuração
Label1	AutoSize	True
	BorderStyle	None
	Font	Microsoft Sans Serif, 14-point

	Text	Modelagem de linhas de transmissão aplicadas Visual Basic
	TextAlign	TopLeft

Tabela 4.0 - Propriedades para o rótulo

4.2 Edite os procedimentos de evento de menu Linhas aéreas de transmissão

1. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como Isoladores.vb e clique em Ok.
2. Clique no menu Linhas aéreas de transmissão do formulário para exibir seus comandos.
3. Dê um clique duplo no comando Isoladores do menu para exibir um procedimento de evento para o comando no Code Editor.

O procedimento de evento IsoladoresToolStripMenuItem_Click aparece no Code Editor.

4. Digite a seguinte instrução de programa:

```
Isoladores.Show( )
```

Esta instrução exibirá uma nova janela, para que o usuário possa entrar com os valores e calcular a tensão de cada isolador.

Faça o mesmo procedimento para Tensão mecânica e flecha de um condutor, só que nomeie como Tensao.vb, e digite a seguinte instrução de programa:

```
Tensao.Show( )
```

4.2.1 Insira objetos ao menu Isoladores

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e groupboxes.

Objeto	Propriedade	Configuração
Label1	Text	Número de isoladores (limite de 10):
Label2	Text	Tensão (kV):
TextBox1	---	---
TextBox2	---	---
TextBox3	---	---
GroupBox1	Text	Entre com os valores
GroupBox2	Text	Observação
Button1	Text	“Calcular”
Button2	Text	“Mostrar resultados”
Button3	Text	“Fechar”

Tabela 4.1 – Objetos do menu Isoladores.

E para que se possa expandir o TextBox3, clique nele e marque a opção Multiline. Para o GroupBox2 (Observação) insira o objeto Label3 e digite:

Para cada cálculo que for feito, feche o programa e abra de novo para que possa ser calculada daquele isolador específico.

Por exemplo se o número que for digitado for 7 e em seguida for 5, feche o programa antes de digitar o $n^{\circ}5$ para que ele calcule aquela tensão do número de isoladores que nesse caso seria 5, ou seja, só irá valer os 5 primeiros valores.

1. Dê um clique duplo no botão Calcular e digite a seguinte instrução de programa entre as instruções Private Sub Button1_Click e End Sub no Code

Editor:

```

Dim V2, V3, V4, V5, V6, V7, V8, V9, V10, tensao,
n_isoladores, alpha, V, V_1, VT2, VT3, VT4, VT5, VT6, VT7, VT8,
VT9, _ VT10 As Double
n_isoladores = TextBox1.Text
tensao = TextBox2.Text
alpha = TextBox1.Text

```

```

Dim i As Short
'Tensao de cada isolador (limite de 10):
V2 = (1 + 1 / (alpha))
V3 = (1 + 3 / (alpha) + 1 / (alpha) ^ 2)
V4 = (1 + 6 / (alpha) + 5 / (alpha) ^ 2 + 1 / (alpha) ^
3)
V5 = (1 + 10 / (alpha) + 15 / (alpha) ^ 2 + 7 / (alpha)
^ 3 + 1 / (alpha) ^ 4)
V6 = (1 + 16 / (alpha) + 35 / (alpha) ^ 2 + 28 /
(alpha) ^ 3 + 9 / (alpha) ^ 4 + 1 / (alpha) ^ 5)
V7 = (1 + 24 / (alpha) + 71 / (alpha) ^ 2 + 84 /
(alpha) ^ 3 + 45 / (alpha) ^ 4 + 11 / (alpha) ^ 5 + 1 / (alpha)
^ 6)
V8 = (1 + 34 / (alpha) + 131 / (alpha) ^ 2 + 211 /
(alpha) ^ 3 + 165 / (alpha) ^ 4 + 66 / (alpha) ^ 5 + 13 /
(alpha) ^ 6 + 1 / (alpha) ^ 7)
V9 = (1 + 46 / (alpha) + 225 / (alpha) ^ 2 + 469 /
(alpha) ^ 3 + 496 / (alpha) ^ 4 + 286 / (alpha) ^ 5 + 91 /
(alpha) ^ 6 + 15 / (alpha) ^ 7 + 1 / (alpha) ^ 8)
V10 = (1 + 60 / (alpha) + 365 / (alpha) ^ 2 + 952 /
(alpha) ^ 3 + 1296 / (alpha) ^ 4 + 1002 / (alpha) ^ 5 + 455 /
(alpha) ^ 6 + 120 / (alpha) ^ 7 + 17 / (alpha) ^ 8 + 1 /
(alpha) ^ 9)

V_1 = (tensao / Math.Sqrt(3)) * 10 ^ -1

If n_isoladores > 10 Or n_isoladores < 1 Then
    MsgBox("Erro!")
End If
'Valor para tensao do isolador 1:
If n_isoladores = 1 Then
    isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (1), "3")
End If
'Valor para tensao do isolador 2:
If n_isoladores = 2 Then
    VT2 = (1 + V2)
    isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT2), "3")
    isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
End If
'Valor para tensao do isolador 3:
If n_isoladores = 3 Then
    VT3 = (1 + V2 + V3)
    isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT3), "3")
    isoladores(i + 1) = FormatNumber((V_1 * V2), "3")

```

```

        isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
    End If
    'Valor para tensao do isolador 4:
    If n_isoladores = 4 Then
        VT4 = (1 + V2 + V3 + V4)
        isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT4), "3")
        isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
        isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
        isoladores(i + 3) = FormatNumber((V_1 * V4), "3")
    End If
    'Valor para tensao do isolador 5:
    If n_isoladores = 5 Then
        VT5 = (1 + V2 + V3 + V4 + V5)
        isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT5), "3")
        isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
        isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
        isoladores(i + 3) = FormatNumber((V_1 * V4), "3")
        isoladores(i + 4) = FormatNumber((V_1 * V5), "3")
    End If
    'Valor para tensao do isolador 6:
    If n_isoladores = 6 Then
        VT6 = 1 + V2 + V3 + V4 + V5 + V6
        isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT6), "3")
        isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
        isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
        isoladores(i + 3) = FormatNumber((V_1 * V4), "3")
        isoladores(i + 4) = FormatNumber((V_1 * V5), "3")
        isoladores(i + 5) = FormatNumber((V_1 * V6), "3")
    End If

    '    V_6 = V_1 * V6
    'Valor para tensao do isolador 7:
    If n_isoladores = 7 Then
        VT7 = 1 + V2 + V3 + V4 + V5 + V6 + V7
        isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT7), "3")
        isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
        isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
        isoladores(i + 3) = FormatNumber((V_1 * V4), "3")
        isoladores(i + 4) = FormatNumber((V_1 * V5), "3")
        isoladores(i + 5) = FormatNumber((V_1 * V6), "3")
        isoladores(i + 6) = FormatNumber((V_1 * V7), "3")
    End If
    'Valor para tensao do isolador 8:

```

```

If n_isoladores = 8 Then
    VT8 = 1 + V2 + V3 + V4 + V5 + V6 + V7 + V8
    isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT8), "3")
    isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
    isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
    isoladores(i + 3) = FormatNumber((V_1 * V4), "3")
    isoladores(i + 4) = FormatNumber((V_1 * V5), "3")
    isoladores(i + 5) = FormatNumber((V_1 * V6), "3")
    isoladores(i + 6) = FormatNumber((V_1 * V7), "3")
    isoladores(i + 7) = FormatNumber((V_1 * V8), "3")
End If
'Valor para tensao do isolador 9:
If n_isoladores = 9 Then
    VT9 = 1 + V2 + V3 + V4 + V5 + V6 + V7 + V8 + V9
    isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT9), "3")
    isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
    isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
    isoladores(i + 3) = FormatNumber((V_1 * V4), "3")
    isoladores(i + 4) = FormatNumber((V_1 * V5), "3")
    isoladores(i + 5) = FormatNumber((V_1 * V6), "3")
    isoladores(i + 6) = FormatNumber((V_1 * V7), "3")
    isoladores(i + 7) = FormatNumber((V_1 * V8), "3")
    isoladores(i + 8) = FormatNumber((V_1 * V9), "3")
End If
'Valor para tensao do isolador 10:
If n_isoladores = 10 Then
    VT10 = 1 + V2 + V3 + V4 + V5 + V6 + V7 + V8 + V9 +
V10
    isoladores(i) = FormatNumber((tensao /
Math.Sqrt(3)) / (VT10), "3")
    isoladores(i + 1) = FormatNumber((V_1 * V2), "3")
    isoladores(i + 2) = FormatNumber((V_1 * V3), "3")
    isoladores(i + 3) = FormatNumber((V_1 * V4), "3")
    isoladores(i + 4) = FormatNumber((V_1 * V5), "3")
    isoladores(i + 5) = FormatNumber((V_1 * V6), "3")
    isoladores(i + 6) = FormatNumber((V_1 * V7), "3")
    isoladores(i + 7) = FormatNumber((V_1 * V8), "3")
    isoladores(i + 8) = FormatNumber((V_1 * V9), "3")
    isoladores(i + 9) = FormatNumber((V_1 * V10), "3")
End If
Dim titulo As String
titulo = "Tensão" & (1) 'Define exatamente quantos
valores o arrays armazena.
For i = 0 To 9 'Onde ficara coluna das tensoes
Next

```

```
For i = 0 To UBound(isoladores)
    Next
```

Como foi visto no capítulo 2, essas fórmulas foram usadas para que o usuário possa saber a tensão de cada isolador, após clicar no botão calcular com os seus respectivos valores.

2. No botão Mostrar resultados e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code Editor:

```
Dim resultado As String
Dim i As Short
resultado = "Tensão de cada Isolador:" &
vbCrLf & vbCrLf
For i = 0 To UBound(isoladores)
    resultado = resultado & "Tensão " & (i +
1) & " =" & vbCrLf & _
isoladores(i) & vbCrLf
Next
TextBox3.Text = resultado
```

3. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Isoladores.

4.2.2 Insira objetos ao menu Tensão mecânica e flecha de um condutor

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e ComboBox.

Objeto	Propriedade	Configuração
Label1	Text	“Tipo do Condutor:”
Label2	Text	“Tração:”

Label3	Text	“Peso específico (kg/m³):”
Label4	Text	“Camada uniforme de gelo:”
Label5	Text	“Vão Médio:”
Label6	Text	“Carga de vento:”
Label7	Text	“Valor da flecha:”
ComboBox1	---	---
Button1	Text	“Calcular”
Button2	Text	“Fechar”
GroupBox1	Text	“Entre com os valores”

Tabela 4.2 – Objetos do menu Tensão mecânica e flecha de um condutor.

E para cada Label insira um TextBox.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.
2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```

0:      ComboBox1.Items.Add( "Linnet" )
1:      ComboBox1.Items.Add( "Hawk" )
2:      ComboBox1.Items.Add( "Grosbeak" )
3:      ComboBox1.Items.Add( "Dove" )
4:      ComboBox1.Items.Add( "Raven" )
5:      ComboBox1.Items.Add( "Osprey" )
6:      ComboBox1.Items.Add( "Teal" )
7:      ComboBox1.Items.Add( "Kingbird" )
8:      ComboBox1.Items.Add( "Rook" )
9:      ComboBox1.Items.Add( "Oriote" )
10:     ComboBox1.Items.Add( "Flicker" )

```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1) para editar seu procedimento de evento:

```

Dim p, l, T0, Area_gelo, camada, Peso_metro, _
    Carga_vento, Carga_de_vento, R As Double
    ' d = TextBox1.Text           'Diâmetro do condutor

```

```

p = TextBox2.Text           'peso específico
l = TextBox3.Text           'Vão médio
T0 = TextBox4.Text          'Tração
camada = TextBox5.Text      'camada uniforme de gelo
Carga_de_vento = TextBox6.Text 'carga de vento
Select Case ComboBox1.SelectedIndex
Case 0
    Area_gelo = (3.14159265 / 4) * ((18.29 +
camada) ^ 2 - (18.29) ^ 2)
    Peso_metro = Area_gelo * p * 0.000001

    Carga_vento = (18.29 + camada) * 0.001 *
Carga_de_vento
    R = Math.Sqrt((0.6883 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
    TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
Case 1
    Area_gelo = (3.14159265 / 4) * ((21.79 +
camada) ^ 2 - (21.79) ^ 2)
    Peso_metro = Area_gelo * p * 0.000001
    Carga_vento = (21.79 + camada) * 0.001 *
Carga_de_vento
    R = Math.Sqrt((0.9764 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
    TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
Case 2
    Area_gelo = (3.14159265 / 4) * ((25.15 +
camada) ^ 2 - (25.15) ^ 2)
    Peso_metro = Area_gelo * p * 0.000001
    Carga_vento = (25.15 + camada) * 0.001 *
Carga_de_vento
    R = Math.Sqrt((1.3 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
    TextBox7.Text = (R * (l) ^ 2) / (8 * T0)

Case 3
    Area_gelo = (3.14159265 / 4) * ((23.53 +
camada) ^ 2 - (23.53) ^ 2)
    Peso_metro = Area_gelo * p * 0.000001
    Carga_vento = (23.53 + camada) * 0.001 *
Carga_de_vento
    R = Math.Sqrt((1.14 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
    TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
Case 4
    Area_gelo = (3.14159265 / 4) * ((10.11 +
camada) ^ 2 - (10.11) ^ 2)

```

```

        Peso_metro = Area_gelo * p * 0.000001
        Carga_vento = (10.11 + camada) * 0.001 *
Carga_de_vento
        R = Math.Sqrt((0.2163 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
        TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
    Case 5
        Area_gelo = (3.14159265 / 4) * ((22.33 +
camada) ^ 2 - (22.33) ^ 2)
        Peso_metro = Area_gelo * p * 0.000001
        Carga_vento = (22.33 + camada) * 0.001 *
Carga_de_vento
        R = Math.Sqrt((0.8988 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
        TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
    Case 6
        Area_gelo = (3.14159265 / 4) * ((25.25 +
camada) ^ 2 - (25.25) ^ 2)
        Peso_metro = Area_gelo * p * 0.000001
        Carga_vento = (25.25 + camada) * 0.001 *
Carga_de_vento
        R = Math.Sqrt((1.398 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
        TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
    Case 7
        Area_gelo = (3.14159265 / 4) * ((23.88 +
camada) ^ 2 - (23.88) ^ 2)
        Peso_metro = Area_gelo * p * 0.000001
        Carga_vento = (23.88 + camada) * 0.001 *
Carga_de_vento
        R = Math.Sqrt((1.028 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
        TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
    Case 8
        Area_gelo = (3.14159265 / 4) * ((24.81 +
camada) ^ 2 - (24.81) ^ 2)
        Peso_metro = Area_gelo * p * 0.000001
        Carga_vento = (24.81 + camada) * 0.001 *
Carga_de_vento
        R = Math.Sqrt((1.219 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
        TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
    Case 9
        Area_gelo = (3.14159265 / 4) * ((18.83 +
camada) ^ 2 - (18.83) ^ 2)

```



```

        Peso_metro = Area_gelo * p * 0.000001
        Carga_vento = (18.83 + camada) * 0.001 *
Carga_de_vento
        R = Math.Sqrt((0.7845 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
        TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
        Case 10
            Area_gelo = (3.14159265 / 4) * ((21.48 +
camada) ^ 2 - (21.48) ^ 2)
            Peso_metro = Area_gelo * p * 0.000001
            Carga_vento = (21.48 + camada) * 0.001 *
Carga_de_vento
            R = Math.Sqrt((0.9146 + Peso_metro) ^ 2 +
(Carga_vento) ^ 2)
            TextBox7.Text = (R * (l) ^ 2) / (8 * T0)
        End Select

```

Como existem 11 tipos de condutores, então teremos 11 casos que o usuário irá selecionar o tipo do condutor e atribuir com os seus respectivos valores. A propriedade SelectedIndex sempre contém o número do item selecionado na caixa Combox, utilizando SelectedIndex, a estrutura Select Case pode identificar rapidamente a escolha do usuário e exibir o tipo do condutor desejado, onde iremos calcular o valor da flecha, visto no capítulo 2.

4. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu tensão mecânica e flecha de um condutor.

4.3 Edite os procedimentos de evento de menu Indutância em linhas de transmissão

1. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como trifasica_simples.vb e clique em Ok.
2. Clique no menu Parâmetros da LT de potência do formulário para exibir seus comandos e em seguida vai para o menu Indutância em linhas de transmissão.
3. Dê um clique duplo no comando Linha trifásica de circuito simples do menu para exibir um procedimento de evento para o comando no Code Editor.

O procedimento de evento

LinhaTrifásicaDeCircuitoSimplesToolStripMenuItem_Click aparece no Code Editor. Digite a seguinte instrução de programa:

```
trifasica_simples.Show()
```

Esta instrução exibirá uma nova janela, para que o usuário possa entrar com os valores e calcular a reatância indutiva.

Faça o mesmo procedimento para Linha múltipla de um condutor, só que nomeie como trifasica_multipla.vb, e digite a seguinte instrução de programa:

```
trifasica_multipla.Show()
```

Faça o mesmo procedimento também para Linha trifásica de circuito duplo, só que nomeie como trifasico_duplo.vb, e digite a seguinte instrução de programa:

```
trifasico_duplo.Show()
```

4.3.1 Insira objetos ao sub-menu Linha trifásica de circuito simples

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e ComboBox.

Objeto	Propriedade	Configuração
Label1	Text	“Tipo do Condutor:”
Label2	Text	“Frequência da operação:”
Label3	Text	“Distância do Condutor 1:”
Label4	Text	“Distância do Condutor 2:”
Label5	Text	“Distância do Condutor 3:”
Label6	Text	“Reatância Indutiva (r/mi):”
ComboBox1	--	--
Button1	Text	“Calcular”
Buton2	Text	“Fechar”

Tabela 4.3 – Objetos do menu Linha trifásica de circuito simples.

E para cada Label insira um TextBox.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.

2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```

0:      ComboBox1.Items.Add( "Partridge" )
1:      ComboBox1.Items.Add( "Ostrich" )
2:      ComboBox1.Items.Add( "Oriole" )
3:      ComboBox1.Items.Add( "Pelican" )
4:      ComboBox1.Items.Add( "Hawk" )
5:      ComboBox1.Items.Add( "Osprey" )
6:      ComboBox1.Items.Add( "Drake" )
7:      ComboBox1.Items.Add( "Cardinal" )
8:      ComboBox1.Items.Add( "Rail" )
9:      ComboBox1.Items.Add( "Bluejay" )
10:     ComboBox1.Items.Add( "Pheasant" )
11:     ComboBox1.Items.Add( "Falcon" )

```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1) para editar seu procedimento de evento:

```

Dim D1, D2, D3, f, Deq, L, Xl As Double

D1 = TextBox1.Text
D2 = TextBox2.Text
D3 = TextBox3.Text
f = TextBox5.Text

Select Case ComboBox1.SelectedIndex
    Case 0
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0198)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

    Case 1
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0229)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

    Case 2
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0255)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

    Case 3
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0264)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

    Case 4
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0289)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

    Case 5
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0284)

```

```

        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

```

Case 6

```

        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0373)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

```

Case 7

```

        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0402)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

```

Case 8

```

        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0386)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

```

Case 9

```

        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0415)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

```

Case 10

```

        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0466)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

```

Case 11

```

        Deq = (D1 * D2 * D3) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * Math.Log(Deq / 0.0523)
        TextBox4.Text = (2 * Math.PI * f * L * 1609)
'Resultado em r/mi lmi = 1609m

```

End Select

Como existem 12 tipos de condutores, então teremos 12 casos que o usuário irá selecionar o tipo do condutor e atribuir com os seus respectivos valores. A propriedade `SelectedIndex` sempre contém o número do item selecionado na caixa `ComboBox`, utilizando `SelectedIndex`, a estrutura `Select Case` pode identificar rapidamente a escolha do usuário e exibir o tipo do condutor desejado, onde iremos calcular a reatância indutiva em Ω/mi , visto no capítulo 3.

4. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções `Private Sub Button2_Click` e `End Sub` no Code Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha trifásica de circuito simples.

4.3.2 Insira objetos ao sub-menu Linha múltipla

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e `ComboBox`.

Objeto	Propriedade	Configuração
Label1	Text	“Tipo do Condutor:”
Label2	Text	“Distância do Condutor:”
Label3	Text	“Comprimento da linha (km):”
Label4	Text	“Distância da fase (cm):”
Label5	Text	“Tensão da base (MVA):”
Label6	Text	“Frequência da Operação:”
Label7	Text	“Tensão da linha (kV):”
Label8	Text	“Reatância Indutiva:”
Label9	Text	“Reatância indutiva em p.u.:”
Button1	Text	“Calcular”
Button2	Text	“Calcular”
Buton3	Text	“Fechar”
ComboBox1	--	--

GroupBox1	Text	“Entre com os valores”
-----------	------	------------------------

Tabela 4.3 – Objetos do menu Linha múltipla.

E para cada Label insira um TextBox.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.
2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```
0:      ComboBox1.Items.Add( "Partridge" )
1:      ComboBox1.Items.Add( "Ostrich" )
2:      ComboBox1.Items.Add( "Oriole" )
3:      ComboBox1.Items.Add( "Pelican" )
4:      ComboBox1.Items.Add( "Hawk" )
5:      ComboBox1.Items.Add( "Osprey" )
6:      ComboBox1.Items.Add( "Drake" )
7:      ComboBox1.Items.Add( "Cardinal" )
8:      ComboBox1.Items.Add( "Rail" )
9:      ComboBox1.Items.Add( "Bluejay" )
10:     ComboBox1.Items.Add( "Pheasant" )
11:     ComboBox1.Items.Add( "Falcon" )
```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1- Reatância Indutiva) para editar seu procedimento de evento:

```
Dim Ds, Dsb, Deq, d1, D, x1, X1, f As Double
```

```
d1 = TextBox2.Text
f = TextBox7.Text
D = TextBox1.Text
```

```
Select Case ComboBox1.SelectedIndex
    Case 0
        x1 = Math.Sqrt(0.0198 + 0.0198)
        Dsb = Math.Sqrt(0.0198 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
```

```

        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 1
        x1 = Math.Sqrt(0.0229 + 0.0229)
        Dsb = Math.Sqrt(0.0229 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 2
        x1 = Math.Sqrt(0.0255 + 0.0255)
        Dsb = Math.Sqrt(0.0255 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 3
        x1 = Math.Sqrt(0.0264 + 0.0264)
        Dsb = Math.Sqrt(0.0264 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 4
        x1 = Math.Sqrt(0.0289 + 0.0289)
        Dsb = Math.Sqrt(0.0289 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 5
        x1 = Math.Sqrt(0.0284 + 0.0284)
        Dsb = Math.Sqrt(0.0284 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 6
        x1 = Math.Sqrt(0.0373 + 0.0373)
        Dsb = Math.Sqrt(0.0373 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 7
        x1 = Math.Sqrt(0.0402 + 0.0402)

```



```

        Dsb = Math.Sqrt(0.0402 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 8
        x1 = Math.Sqrt(0.0386 + 0.0386)
        Dsb = Math.Sqrt(0.0386 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 9
        x1 = Math.Sqrt(0.0415 + 0.0415)
        Dsb = Math.Sqrt(0.0415 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 10
        x1 = Math.Sqrt(0.0466 + 0.0466)
        Dsb = Math.Sqrt(0.0466 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
    Case 11
        x1 = Math.Sqrt(0.0523 + 0.0523)
        Dsb = Math.Sqrt(0.0523 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        Xl = (2 * Math.PI * f * 2 * 10 ^ (-7)) *
Math.Log(Deq / Dsb)
        TextBox6.Text = Xl * 1609
End Select

```

4. Agora dê um clique duplo no objeto (Button2- Reatância indutiva em p.u.)

para editar seu procedimento de evento:

```

Dim Ds, Dsb, Deq, d1, D, BaseZ, DL, X, x1, Xl, Xl2, f,
tensao_base, tensao As Double

```

```

d1 = TextBox2.Text
DL = TextBox3.Text
f = TextBox7.Text
D = TextBox1.Text
tensao_base = TextBox4.Text

```

```

tensao = TextBox8.Text

Select Case ComboBox1.SelectedIndex
Case 0
    x1 = Math.Sqrt(0.0198 + 0.0198)
    Dsb = Math.Sqrt(0.0198 * x1 * (d1 / 100))
    Deq = (D * D * (D + D)) ^ (1 / 3)
    X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
    TextBox6.Text = X1 * 1609
    BaseZ = (tensao / tensao_base)
    X12 = (X1 * DL) / BaseZ
    BaseZ = (((tensao) ^ 2) / (tensao_base))
    TextBox5.Text = (X1 * DL) / (BaseZ)
Case 1
    x1 = Math.Sqrt(0.0229 + 0.0229)
    Dsb = Math.Sqrt(0.0229 * x1 * (d1 / 100))
    Deq = (D * D * (D + D)) ^ (1 / 3)
    X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
    X12 = X1 * 1609
    BaseZ = (((tensao) ^ 2) / (tensao_base))
    TextBox5.Text = (X1 * DL) / (BaseZ)
Case 2
    x1 = Math.Sqrt(0.0255 + 0.0255)
    Dsb = Math.Sqrt(0.0255 * x1 * (d1 / 100))
    Deq = (D * D * (D + D)) ^ (1 / 3)
    X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
    X12 = X1 * 1609
    BaseZ = (((tensao) ^ 2) / (tensao_base))
    TextBox5.Text = (X1 * DL) / (BaseZ)
Case 3
    x1 = Math.Sqrt(0.0264 + 0.0264)
    Dsb = Math.Sqrt(0.0264 * x1 * (d1 / 100))
    Deq = (D * D * (D + D)) ^ (1 / 3)
    X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
    X12 = X1 * 1609
    BaseZ = (((tensao) ^ 2) / (tensao_base))
    TextBox5.Text = (X1 * DL) / (BaseZ)
Case 4
    x1 = Math.Sqrt(0.0289 + 0.0289)
    Dsb = Math.Sqrt(0.0289 * x1 * (d1 / 100))
    Deq = (D * D * (D + D)) ^ (1 / 3)
    X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)

```

```

X12 = X1 * 1609
BaseZ = (((tensao) ^ 2) / (tensao_base))
TextBox5.Text = (X1 * DL) / (BaseZ)
Case 5
x1 = Math.Sqrt(0.0284 + 0.0284)
Dsb = Math.Sqrt(0.0284 * x1 * (d1 / 100))
Deq = (D * D * (D + D)) ^ (1 / 3)
X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
X12 = X1 * 1609
BaseZ = (((tensao) ^ 2) / (tensao_base))
TextBox5.Text = (X1 * DL) / (BaseZ)
Case 6
x1 = Math.Sqrt(0.0373 + 0.0373)
Dsb = Math.Sqrt(0.0373 * x1 * (d1 / 100))
Deq = (D * D * (D + D)) ^ (1 / 3)
X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
X12 = X1 * 1609
BaseZ = (((tensao) ^ 2) / (tensao_base))
TextBox5.Text = (X1 * DL) / (BaseZ)
Case 7
x1 = Math.Sqrt(0.0402 + 0.0402)
Dsb = Math.Sqrt(0.0402 * x1 * (d1 / 100))
Deq = (D * D * (D + D)) ^ (1 / 3)
X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
X12 = X1 * 1609
BaseZ = (((tensao) ^ 2) / (tensao_base))
TextBox5.Text = (X1 * DL) / (BaseZ)
Case 8
x1 = Math.Sqrt(0.0386 + 0.0386)
Dsb = Math.Sqrt(0.0386 * x1 * (d1 / 100))
Deq = (D * D * (D + D)) ^ (1 / 3)
X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
X12 = X1 * 1609
BaseZ = (((tensao) ^ 2) / (tensao_base))
TextBox5.Text = (X1 * DL) / (BaseZ)
Case 9
x1 = Math.Sqrt(0.0415 + 0.0415)
Dsb = Math.Sqrt(0.0415 * x1 * (d1 / 100))
Deq = (D * D * (D + D)) ^ (1 / 3)
X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
X12 = X1 * 1609
BaseZ = (((tensao) ^ 2) / (tensao_base))

```

```

        TextBox5.Text = (X1 * DL) / (BaseZ)
    Case 10
        x1 = Math.Sqrt(0.0466 + 0.0466)
        Dsb = Math.Sqrt(0.0466 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
        X12 = X1 * 1609
        BaseZ = (((tensao) ^ 2) / (tensao_base))
        TextBox5.Text = (X1 * DL) / (BaseZ)
    Case 11
        x1 = Math.Sqrt(0.0523 + 0.0523)
        Dsb = Math.Sqrt(0.0523 * x1 * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        X1 = (2 * Math.PI * f * 2 * 10 ^ (-7) * 10 ^ 3)
* Math.Log(Deq / Dsb)
        X12 = X1 * 1609
        BaseZ = (((tensao) ^ 2) / (tensao_base))
        TextBox5.Text = (X1 * DL) / (BaseZ)

```

5. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha múltipla.

4.3.3 Insira objetos ao sub-menu Linha trifásica de circuito duplo

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e ComboBox.

Objeto	Propriedade	Configuração
Label1	Text	“Tipo do Condutor:”
Label2	Text	“Frequencia da Operação:”
Label3	Text	“Distância do Condutor a - c'=c - a'.”
Label4	Text	“Distância do Condutor b - b'.”
Label5	Text	“Distância do Condutor a - b=b - c (altura):”

Label6	Text	“Reatância Indutiva:”
ComboBox1	--	--
Button1	Text	“Calcular”
Button2	Text	“Fechar”
GroupBox1	Text	“Entre com os valores”

Tabela 4.4 – Objetos do menu Linha trifásica de circuito duplo.

E para cada Label insira um TextBox.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.
2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```

0:      ComboBox1.Items.Add( "Partridge" )
1:      ComboBox1.Items.Add( "Ostrich" )
2:      ComboBox1.Items.Add( "Oriole" )
3:      ComboBox1.Items.Add( "Pelican" )
4:      ComboBox1.Items.Add( "Hawk" )
5:      ComboBox1.Items.Add( "Osprey" )
6:      ComboBox1.Items.Add( "Drake" )
7:      ComboBox1.Items.Add( "Cardinal" )
8:      ComboBox1.Items.Add( "Rail" )
9:      ComboBox1.Items.Add( "Bluejay" )
10:     ComboBox1.Items.Add( "Pheasant" )
11:     ComboBox1.Items.Add( "Falcon" )

```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1) para editar seu procedimento de evento:

```

Dim f, DH, DL1, DL2, Dab, Dab_, d1, d2, Dabp, Dbcp,
Dcap, _
Deq, Daa_, aa_, bb_, cc_, Ds, L As Double

DH = TextBox4.Text
DL1 = TextBox1.Text
DL2 = TextBox2.Text
f = TextBox5.Text

```

```

d1 = (DL2 - DL1) / 2
d2 = (DL2 + DL1) / 2
Dab = Math.Sqrt((DH) ^ 2 + (d1) ^ 2)
Dab_ = Math.Sqrt((DH) ^ 2 + (d2) ^ 2)

Dabp = ((Dab * Dab_) ^ 2) ^ (1 / 4)
Dbcp = ((Dab * Dab_) ^ 2) ^ (1 / 4)
Dcap = (((DH + DH) * DL1) ^ 2) ^ (1 / 4)
Deq = (Dabp * Dbcp * Dcap) ^ (1 / 3)

Daa_ = Math.Sqrt((DH + DH) ^ 2 + (DL1) ^ 2)
'=====

Select Case ComboBox1.SelectedIndex
    Case 0
        aa_ = Math.Sqrt(Daa_ * 0.0198)
        bb_ = Math.Sqrt(DL2 * 0.0198)
        cc_ = Math.Sqrt(Daa_ * 0.0198)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
        TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

    Case 1

        aa_ = Math.Sqrt(Daa_ * 0.0229)
        bb_ = Math.Sqrt(DL2 * 0.0229)
        cc_ = Math.Sqrt(Daa_ * 0.0229)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
        TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

    Case 2

        aa_ = Math.Sqrt(Daa_ * 0.0255)
        bb_ = Math.Sqrt(DL2 * 0.0255)
        cc_ = Math.Sqrt(Daa_ * 0.0255)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
        TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

    Case 3
        aa_ = Math.Sqrt(Daa_ * 0.0264)

```

```

        bb_ = Math.Sqrt(DL2 * 0.0264)
        cc_ = Math.Sqrt(Daa_ * 0.0264)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
        L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
        TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

        Case 4
            aa_ = Math.Sqrt(Daa_ * 0.0289)
            bb_ = Math.Sqrt(DL2 * 0.0289)
            cc_ = Math.Sqrt(Daa_ * 0.0289)
            Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
            L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
            TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

            Case 5
                aa_ = Math.Sqrt(Daa_ * 0.0284)
                bb_ = Math.Sqrt(DL2 * 0.0284)
                cc_ = Math.Sqrt(Daa_ * 0.0284)
                Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
                L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
                TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

                Case 6
                    aa_ = Math.Sqrt(Daa_ * 0.0373)
                    bb_ = Math.Sqrt(DL2 * 0.0373)
                    cc_ = Math.Sqrt(Daa_ * 0.0373)
                    Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
                    L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
                    TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

                    Case 7
                        aa_ = Math.Sqrt(Daa_ * 0.0402)
                        bb_ = Math.Sqrt(DL2 * 0.0402)
                        cc_ = Math.Sqrt(Daa_ * 0.0402)
                        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
                        L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
                        TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

                        Case 8

```

```

aa_ = Math.Sqrt(Daa_ * 0.0386)
bb_ = Math.Sqrt(DL2 * 0.0386)
cc_ = Math.Sqrt(Daa_ * 0.0386)
Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
  TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

Case 9
aa_ = Math.Sqrt(Daa_ * 0.0415)
bb_ = Math.Sqrt(DL2 * 0.0415)
cc_ = Math.Sqrt(Daa_ * 0.0415)
Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
  TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

Case 10
aa_ = Math.Sqrt(Daa_ * 0.0466)
bb_ = Math.Sqrt(DL2 * 0.0466)
cc_ = Math.Sqrt(Daa_ * 0.0466)
Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
  TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

Case 11
aa_ = Math.Sqrt(Daa_ * 0.0523)
bb_ = Math.Sqrt(DL2 * 0.0523)
cc_ = Math.Sqrt(Daa_ * 0.0523)
Ds = (aa_ * bb_ * cc_) ^ (1 / 3)
L = (2 * 10 ^ (-7)) * (Math.Log(Deq / Ds))
'Indutancia por fase H/m
  TextBox6.Text = 2 * Math.PI * f * L * 1609
'Reatancia indutiva em r/mi

```

End Select

4. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:


```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha trifásica de circuito duplo.

4.4 Edite os procedimentos de evento de menu Capacitância em linhas de transmissão

1. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como susceptancia_capacitiva.vb e clique em Ok.
2. Clique no menu Parâmetros da LT de potência e em seguida Capacitância em linhas de transmissão do formulário para exibir seus comandos.
3. Dê um clique duplo no comando Susceptância capacitiva de uma linha monofásica do menu para exibir um procedimento de evento para o comando no Code Editor.

O procedimento de evento SusceptânciaCapacitiToolStripMenuItem_Click aparece no Code Editor.

4. Digite a seguinte instrução de programa:

```
susceptancia_capacitiva.Show()
```

Faça o mesmo procedimento para Linha trifásica de um circuito simples, só que nomeie como trifasica_simples(C).vb, e digite a seguinte instrução de programa:

```
trifasica_simples_C_.Show()
```

Faça o mesmo procedimento também para Linha múltipla, só que nomeie como linha_multipla(C).vb, e digite a seguinte instrução de programa:

```
linha_multipla_C_.Show()
```

Faça o mesmo procedimento também para Linha trifásica de circuito duplo, só que nomeie como trifasico_duplo(C).vb, e digite a seguinte instrução de programa:

```
trifasico_duplo_C_.Show()
```

4.4.1 Insira objetos ao sub-menu Susceptância capacitiva de uma linha monofásica

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e ComboBox.

Objeto	Propriedade	Configuração
Label1	Text	“Tipo do Condutor:”
Label2	Text	“Frequência de Operação:”
Label3	Text	“Espaçamento entre os centros (ft):”
Label4	Text	“Reatância capacitiva da linha:”
Label5	Text	“Susceptância capacitiva da linha:”
ComboBox1	--	--
Button1	Text	“Calcular”
Buton2	Text	“Calcular”
Button3	Text	“Fechar”
GroupBox1	Text	“Entre com os valores”

Tabela 4.5 – Objetos do menu Susceptância capacitiva de uma linha monofásica.

E para cada Label insira um TextBox.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.
2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```
0:      ComboBox1.Items.Add( "Partridge" )
1:      ComboBox1.Items.Add( "Ostrich" )
2:      ComboBox1.Items.Add( "Oriole" )
3:      ComboBox1.Items.Add( "Pelican" )
4:      ComboBox1.Items.Add( "Hawk" )
5:      ComboBox1.Items.Add( "Osprey" )
6:      ComboBox1.Items.Add( "Drake" )
7:      ComboBox1.Items.Add( "Cardinal" )
8:      ComboBox1.Items.Add( "Rail" )
```

```

9:      ComboBox1.Items.Add( "Bluejay" )
10:     ComboBox1.Items.Add( "Pheasant" )
11:     ComboBox1.Items.Add( "Falcon" )

```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1-reatância capacitiva da linha) para editar seu procedimento de evento:

```

Dim r, r1, Xc, D, f, Bc As Double

f = TextBox4.Text
D = TextBox1.Text

Select Case ComboBox1.SelectedIndex
Case 0
    r = 0.642 / (2 * 12)
    Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

    TextBox2.Text = 2 * Xc 'unidade: r/mi

Case 1
    r = 0.68 / (2 * 12)
    Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

    TextBox2.Text = 2 * Xc 'unidade: r/mi

Case 2
    r = 0.741 / (2 * 12)
    Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

    TextBox2.Text = 2 * Xc 'unidade: r/mi

Case 3
    r = 0.814 / (2 * 12)
    Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

    TextBox2.Text = 2 * Xc 'unidade: r/mi

Case 4
    r = 0.858 / (2 * 12)
    Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

    TextBox2.Text = 2 * Xc 'unidade: r/mi

Case 5

```

```

r = 0.879 / (2 * 12)
Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

TextBox2.Text = 2 * Xc 'unidade: r/mi
Case 6
r = 1.108 / (2 * 12)
Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

TextBox2.Text = 2 * Xc 'unidade: r/mi
Case 7
r = 1.196 / (2 * 12)
Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

TextBox2.Text = 2 * Xc 'unidade: r/mi
Case 8
r = 1.165 / (2 * 12)
Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

TextBox2.Text = 2 * Xc 'unidade: r/mi
Case 9
r = 1.259 / (2 * 12)
Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

TextBox2.Text = 2 * Xc 'unidade: r/mi
Case 10
r = 1.382 / (2 * 12)
Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

TextBox2.Text = 2 * Xc 'unidade: r/mi
Case 11
r = 1.545 / (2 * 12)
Xc = ((1.779 / f) * 10 ^ (6)) * Math.Log(D / r)

TextBox2.Text = 2 * Xc 'unidade: r/mi
End Select

```

4. Agora dê um clique duplo no objeto (Button2- Susceptância capacitiva da linha) para editar seu procedimento de evento:

```
Dim r, r1, Xc, Xc1, D, f, Bc As Double
```

```
f = TextBox4.Text
D = TextBox1.Text
```

```
Select Case ComboBox1.SelectedIndex
Case 0
```

```

r)
    r = 0.642 / (2 * 12)
    Xc1 = ((1.779 / f) * 10 ^ (6)) * Math.Log(D /

    Bc = 1 / Xc1

    Xc = 2 * Xc1
    TextBox3.Text = 1 / Xc 'unidade: S/mi

```

Case 1

```

r)
    r = 0.68 / (2 * 12)
    Xc1 = ((1.779 / f) * 10 ^ (6)) * Math.Log(D /

    Bc = 1 / Xc1

    Xc = 2 * Xc1
    TextBox3.Text = 1 / Xc 'unidade: S/mi

```

Case 2

```

r)
    r = 0.741 / (2 * 12)
    Xc1 = ((1.779 / f) * 10 ^ (6)) * Math.Log(D /

    Bc = 1 / Xc1

    Xc = 2 * Xc1
    TextBox3.Text = 1 / Xc 'unidade: S/mi

```

Case 3

```

r)
    r = 0.814 / (2 * 12)
    Xc1 = ((1.779 / f) * 10 ^ (6)) * Math.Log(D /

    Bc = 1 / Xc1

    Xc = 2 * Xc1
    TextBox3.Text = 1 / Xc 'unidade: S/mi

```

Case 4

```

r)
    r = 0.858 / (2 * 12)
    Xc1 = ((1.779 / f) * 10 ^ (6)) * Math.Log(D /

    Bc = 1 / Xc1

    Xc = 2 * Xc1
    TextBox3.Text = 1 / Xc 'unidade: S/mi

```

Case 5

```

    r = 0.879 / (2 * 12)

```

r) $Xc1 = ((1.779 / f) * 10 ^ (6)) * \text{Math.Log}(D /$
 $Bc = 1 / Xc1$
 $Xc = 2 * Xc1$
 TextBox3.Text = 1 / Xc 'unidade: S/mi

Case 6

r) $r = 1.108 / (2 * 12)$
 $Xc1 = ((1.779 / f) * 10 ^ (6)) * \text{Math.Log}(D /$
 $Bc = 1 / Xc1$
 $Xc = 2 * Xc1$
 TextBox3.Text = 1 / Xc 'unidade: S/mi

Case 7

r) $r = 1.196 / (2 * 12)$
 $Xc1 = ((1.779 / f) * 10 ^ (6)) * \text{Math.Log}(D /$
 $Bc = 1 / Xc1$
 $Xc = 2 * Xc1$
 TextBox3.Text = 1 / Xc 'unidade: S/mi

Case 8

r) $r = 1.165 / (2 * 12)$
 $Xc1 = ((1.779 / f) * 10 ^ (6)) * \text{Math.Log}(D /$
 $Bc = 1 / Xc1$
 $Xc = 2 * Xc1$
 TextBox3.Text = 1 / Xc 'unidade: S/mi

Case 9

r) $r = 1.259 / (2 * 12)$
 $Xc1 = ((1.779 / f) * 10 ^ (6)) * \text{Math.Log}(D /$
 $Bc = 1 / Xc1$
 $Xc = 2 * Xc1$
 TextBox3.Text = 1 / Xc 'unidade: S/mi

Case 10

r) $r = 1.382 / (2 * 12)$
 $Xc1 = ((1.779 / f) * 10 ^ (6)) * \text{Math.Log}(D /$

```

Bc = 1 / Xc1
Xc = 2 * Xc1
TextBox3.Text = 1 / Xc 'unidade: S/mi

```

```
Case 11
```

```

r = 1.545 / (2 * 12)
Xc1 = ((1.779 / f) * 10 ^ (6)) * Math.Log(D /

```

r)

```
Bc = 1 / Xc1
```

```

Xc = 2 * Xc1
TextBox3.Text = 1 / Xc 'unidade: S/mi

```

```
End Select
```

5. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu

Susceptância capacitiva de uma linha monofásica.

4.4.2 Insira objetos ao sub-menu Linha trifásica de um circuito simples

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e ComboBox.

Objeto	Propriedade	Configuração
Label1	Text	Tipo do Condutor:
Label2	Text	Frequência da operação:
Label3	Text	Comprimento da linha (mi):
Label4	Text	Tensão da Linha (kV):
Label5	Text	Distância do Condutor 1:
Label6	Text	Distância do Condutor 2:
Label7	Text	Distância do Condutor 3:
Label8	Text	Capacitância ao neutro:
Label9	Text	Reatância Capacitiva:
Label10	Text	Reatância Capacitiva do

		comprimento da linha:
Label11	Text	Potência reativa (MVA _r):
ComboBox1	--	--
Button1	Text	“Calcular”
Button2	Text	“Calcular”
Button3	Text	“Calcular”
Button4	Text	“Calcular”
Button5	Text	“Fechar”
GroupBox1	Text	“Entre com os valores”

Tabela 4.5 – Objetos do menu Linha trifásica de um circuito simples.

E para cada Label insira um TextBox, exceto o ComboBox1.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.
2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```

0:      ComboBox1.Items.Add( "Partridge" )
1:      ComboBox1.Items.Add( "Ostrich" )
2:      ComboBox1.Items.Add( "Oriole" )
3:      ComboBox1.Items.Add( "Pelican" )
4:      ComboBox1.Items.Add( "Hawk" )
5:      ComboBox1.Items.Add( "Osprey" )
6:      ComboBox1.Items.Add( "Drake" )
7:      ComboBox1.Items.Add( "Cardinal" )
8:      ComboBox1.Items.Add( "Rail" )
9:      ComboBox1.Items.Add( "Bluejay" )
10:     ComboBox1.Items.Add( "Pheasant" )
11:     ComboBox1.Items.Add( "Falcon" )

```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1-Capacitância ao neutro) para editar seu procedimento de evento:

```

Dim r, D1, D2, D3, Deq As Double
    D1 = TextBox7.Text
    D2 = TextBox8.Text
    D3 = TextBox9.Text
    'Cn= textbox3.text

```



```

Select Case ComboBox1.SelectedIndex
Case 0
    r = (0.642) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 1
    r = (0.68) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 2
    r = (0.741) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 3
    r = (0.814) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m

Case 4
    r = (0.858) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 5
    r = (0.879) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 6
    r = (1.108) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 7
    r = (1.196) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 8
    r = (1.165) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m

```

```

Case 9
    r = (1.259) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 10
    r = (1.382) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
Case 11
    r = (1.545) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / r)) 'Unidade: F/m
End Select

```

4. Agora dê um clique duplo no objeto (Button2 - Reatância Capacitiva) para editar seu procedimento de evento:

```

Dim r, D1, D2, D3, Cn, f, Deq As Double
D1 = TextBox7.Text
D2 = TextBox8.Text
D3 = TextBox9.Text
f = TextBox1.Text
Select Case ComboBox1.SelectedIndex
Case 0
    r = (0.642) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
Case 1
    r = (0.68) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
Case 2
    r = (0.741) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m

```

```

        TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 3
            r = (0.814) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 4
            r = (0.858) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 5
            r = (0.879) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 6
            r = (1.108) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 7
            r = (1.196) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 8
            r = (1.165) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 9
            r = (1.259) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)

```

```

        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 10
            r = (1.382) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        Case 11
            r = (1.545) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            TextBox4.Text = (1) / (2 * Math.PI * f * Cn *
1609) 'Unidade: r.mi
        End Select

```

5. Agora dê um clique duplo no objeto (Button3 - Reatância Capacitiva do comprimento da linha) para editar seu procedimento de evento:

```

Dim r, D1, D2, D3, Cn, f, Deq, Xc, linha As Double
    D1 = TextBox7.Text
    D2 = TextBox8.Text
    D3 = TextBox9.Text
    f = TextBox1.Text
    linha = TextBox2.Text
    Select Case ComboBox1.SelectedIndex

        Case 0
            r = (0.642) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
            TextBox5.Text = Xc / linha
        Case 1
            r = (0.68) / (2 * 12)
            Deq = (D1 * D2 * D3) ^ (1 / 3)
            Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
            Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
            TextBox5.Text = Xc / linha

```

```

Case 2
    r = (0.741) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    TextBox5.Text = Xc / linha
Case 3
    r = (0.814) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    TextBox5.Text = Xc / linha
Case 4
    r = (0.858) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    TextBox5.Text = Xc / linha
Case 5
    r = (0.879) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    TextBox5.Text = Xc / linha
Case 6
    r = (1.108) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    TextBox5.Text = Xc / linha
Case 7
    r = (1.196) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi

```

```

        TextBox5.Text = Xc / linha
    Case 8
        r = (1.165) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        TextBox5.Text = Xc / linha
    Case 9
        r = (1.259) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        TextBox5.Text = Xc / linha
    Case 10
        r = (1.382) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        TextBox5.Text = Xc / linha
    Case 11
        r = (1.545) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        TextBox5.Text = Xc / linha
    End Select

```

6. Agora dê um clique duplo no objeto (Button4 - Potência reativa (MVA_r))

para editar seu procedimento de evento:

```

    Dim r, D1, D2, D3, Cn, f, Deq, Xc, linha, V, Rea_C, Ichg,
Ichg2 As Double
    D1 = TextBox7.Text
    D2 = TextBox8.Text
    D3 = TextBox9.Text
    f = TextBox1.Text
    linha = TextBox2.Text
    V = TextBox6.Text
    Select Case ComboBox1.SelectedIndex

```

```

Case 0
    r = (0.642) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    Rea_C = Xc / linha
    TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
Case 1
    r = (0.68) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    Rea_C = Xc / linha
    TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
Case 2
    r = (0.741) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    Rea_C = Xc / linha
    TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
Case 3
    r = (0.814) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    Rea_C = Xc / linha
Case 4
    r = (0.858) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)
    Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
    Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
    Rea_C = Xc / linha
    TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
Case 5
    r = (0.879) / (2 * 12)
    Deq = (D1 * D2 * D3) ^ (1 / 3)

```

```

        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        Rea_C = Xc / linha 'Unidade: ohm
        TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
    Case 6
        r = (1.108) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        Rea_C = Xc / linha
        Ichg = 2 * Math.PI * f * (V / Math.Sqrt(3)) *
Cn * 1609 'corrente de carregamento: A/mi
        Ichg2 = Ichg * linha 'Unidade: A
        TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
    Case 7
        r = (1.196) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        Rea_C = Xc / linha
        TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
    Case 8
        r = (1.165) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        Rea_C = Xc / linha
        TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
    Case 9
        r = (1.259) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        Rea_C = Xc / linha
        TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
    Case 10
        r = (1.382) / (2 * 12)

```



```

        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        Rea_C = Xc / linha
        TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
Case 11
        r = (1.545) / (2 * 12)
        Deq = (D1 * D2 * D3) ^ (1 / 3)
        Cn = (2 * Math.PI * 8.85 * 10 ^ (-12)) /
(Math.Log(Deq / r)) 'Unidade: F/m
        Xc = (1) / (2 * Math.PI * f * Cn * 1609)
'Unidade: r.mi
        Rea_C = Xc / linha
        TextBox10.Text = Math.Sqrt(3) * V * Ichg2 * 1
End Select

```

7. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha trifásica de um circuito simples.

4.4.3 Insira objetos ao sub-menu Linha múltipla

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e ComboBox.

Objeto	Propriedade	Configuração
Label1	Text	“Tipo do Condutor:”
Label2	Text	“Distância da fase (cm):”
Label3	Text	“Distância do Condutor:”
Label4	Text	“Frequência da Operação:”
Label5	Text	“Capacitância:”
ComboBox1	Text	--
Button1	Text	“Calcular”
Button2	Text	“Fechar”
GroupBox1	Text	“Entre com os valores”

Tabela 4.6 – Objetos do menu Linha múltipla.

E para cada Label insira um TextBox, exceto o ComboBox1.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.
2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```
0:      ComboBox1.Items.Add("Partridge")
1:      ComboBox1.Items.Add("Ostrich")
2:      ComboBox1.Items.Add("Oriole")
3:      ComboBox1.Items.Add("Pelican")
4:      ComboBox1.Items.Add("Hawk")
5:      ComboBox1.Items.Add("Osprey")
6:      ComboBox1.Items.Add("Drake")
7:      ComboBox1.Items.Add("Cardinal")
8:      ComboBox1.Items.Add("Rail")
9:      ComboBox1.Items.Add("Bluejay")
10:     ComboBox1.Items.Add("Pheasant")
11:     ComboBox1.Items.Add("Falcon")
```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1) para editar seu procedimento de evento:

```
Dim D, d1, r, Ds, f, Deq As Double
D = TextBox1.Text
d1 = TextBox2.Text
f = TextBox3.Text

Select Case ComboBox1.SelectedIndex
    Case 0
        r = (0.642 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 1
        r = (0.68 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
```

```

        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 2
        r = (0.741 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 3
        r = (0.814 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 4
        r = (0.858 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 5
        r = (0.879 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 6
        r = (1.108 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 7
        r = (1.196 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 8
        r = (1.165 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 9
        r = (1.259 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))

```

```

        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 10
        r = (1.382 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
    Case 11
        r = (1.545 * 0.3048) / (2 * 12)
        Ds = Math.Sqrt(r * (d1 / 100))
        Deq = (D * D * (D + D)) ^ (1 / 3)
        TextBox4.Text = (2 * Math.PI * 8.85 * 1 * 10 ^
(-12)) / (Math.Log(Deq / Math.Sqrt(r * d1 / 100)))
End Select

```

4. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha múltipla.

4.4.4 Insira objetos ao sub-menu Linha trifásica de circuito duplo

Configure as seguintes propriedades para os objetos botões, caixas de textos, labels e ComboBox.

Objeto	Propriedade	Configuração
Label1	Text	“Tipo do Condutor:”
Label2	Text	“Frequencia da Operação:”
Label3	Text	“Distância do Condutor a - c'=c - a'.”
Label4	Text	“Distância do Condutor b - b'.”
Label5	Text	“Distância do Condutor a - b=b - c (altura):”
Label6	Text	“Reatância Indutiva:”
ComboBox1	--	--
Button1	Text	“Calcular”
Button2	Text	“Calcular”

Button3	Text	“Fechar”
GroupBox1	Text	“Entre com os valores”

Tabela 4.7 – Objetos do menu Linha trifásica de circuito duplo.

E para cada Label insira um TextBox, exceto ComboBox1.

1. Dê um clique duplo no formulário, o procedimento de evento Form1_Load aparece no Code Editor.
2. Digite o seguinte código de programa para inicializar a caixa de listagem:

```
0:      ComboBox1.Items.Add( "Partridge" )
1:      ComboBox1.Items.Add( "Ostrich" )
2:      ComboBox1.Items.Add( "Oriole" )
3:      ComboBox1.Items.Add( "Pelican" )
4:      ComboBox1.Items.Add( "Hawk" )
5:      ComboBox1.Items.Add( "Osprey" )
6:      ComboBox1.Items.Add( "Drake" )
7:      ComboBox1.Items.Add( "Cardinal" )
8:      ComboBox1.Items.Add( "Rail" )
9:      ComboBox1.Items.Add( "Bluejay" )
10:     ComboBox1.Items.Add( "Pheasant" )
11:     ComboBox1.Items.Add( "Falcon" )
```

Essas linhas utilizam o método Add do objeto caixa de listagem para adicionar entradas à caixa de listagem do formulário.

3. Clique na guia Form1.vb [Design] na parte superior do Code Editor para voltar para o Designer e, então, dê um clique duplo no objeto (Button1 - Capacitância ao neutro) para editar seu procedimento de evento:

```
Dim f, DH, DL1, DL2, Dab, Dab_, d1, d2, Dabp, Dbcp,
Dcap, Deq, Daa_, aa_, bb_, cc_, Ds, r As Double
```

```
DH = TextBox4.Text
DL1 = TextBox1.Text
DL2 = TextBox2.Text
f = TextBox5.Text
```

```
d1 = (DL2 - DL1) / 2
d2 = (DL2 + DL1) / 2
Dab = Math.Sqrt((DH) ^ 2 + (d1) ^ 2)
Dab_ = Math.Sqrt((DH) ^ 2 + (d2) ^ 2)
```

```

Dabp = ((Dab * Dab_) ^ 2) ^ (1 / 4)
Dbcp = ((Dab * Dab_) ^ 2) ^ (1 / 4)
Dcap = (((DH + DH) * DL1) ^ 2) ^ (1 / 4)
Deq = (Dabp * Dbcp * Dcap) ^ (1 / 3)

Daa_ = Math.Sqrt((DH + DH) ^ 2 + (DL1) ^ 2)
Select Case ComboBox1.SelectedIndex
    Case 0
        r = (0.642) / (2 * 12)
        aa_ = Math.Sqrt(Daa_ * r)
        bb_ = Math.Sqrt(DL2 * r)
        cc_ = Math.Sqrt(Daa_ * r)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

        TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 1
        r = (0.68) / (2 * 12)
        aa_ = Math.Sqrt(Daa_ * r)
        bb_ = Math.Sqrt(DL2 * r)
        cc_ = Math.Sqrt(Daa_ * r)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

        TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 2
        r = (0.741) / (2 * 12)
        aa_ = Math.Sqrt(Daa_ * r)
        bb_ = Math.Sqrt(DL2 * r)
        cc_ = Math.Sqrt(Daa_ * r)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

        TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 3
        r = (0.814) / (2 * 12)
        aa_ = Math.Sqrt(Daa_ * r)
        bb_ = Math.Sqrt(DL2 * r)
        cc_ = Math.Sqrt(Daa_ * r)
        Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

        TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 4
        r = (0.858) / (2 * 12)
        aa_ = Math.Sqrt(Daa_ * r)

```

```

bb_ = Math.Sqrt(DL2 * r)
cc_ = Math.Sqrt(Daa_ * r)
Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 5
    r = (0.879) / (2 * 12)
    aa_ = Math.Sqrt(Daa_ * r)
    bb_ = Math.Sqrt(DL2 * r)
    cc_ = Math.Sqrt(Daa_ * r)
    Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 6
    r = (1.108) / (2 * 12)
    aa_ = Math.Sqrt(Daa_ * r)
    bb_ = Math.Sqrt(DL2 * r)
    cc_ = Math.Sqrt(Daa_ * r)
    Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 7
    r = (1.196) / (2 * 12)
    aa_ = Math.Sqrt(Daa_ * r)
    bb_ = Math.Sqrt(DL2 * r)
    cc_ = Math.Sqrt(Daa_ * r)
    Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 8
    r = (1.165) / (2 * 12)
    aa_ = Math.Sqrt(Daa_ * r)
    bb_ = Math.Sqrt(DL2 * r)
    cc_ = Math.Sqrt(Daa_ * r)
    Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

    TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
    Case 9
    r = (1.259) / (2 * 12)
    aa_ = Math.Sqrt(Daa_ * r)
    bb_ = Math.Sqrt(DL2 * r)
    cc_ = Math.Sqrt(Daa_ * r)

```

```

Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
Case 10
r = (1.382) / (2 * 12)
aa_ = Math.Sqrt(Daa_ * r)
bb_ = Math.Sqrt(DL2 * r)
cc_ = Math.Sqrt(Daa_ * r)
Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
Case 11
r = (1.545) / (2 * 12)
aa_ = Math.Sqrt(Daa_ * r)
bb_ = Math.Sqrt(DL2 * r)
cc_ = Math.Sqrt(Daa_ * r)
Ds = (aa_ * bb_ * cc_) ^ (1 / 3)

TextBox3.Text = (2 * Math.PI * 8.85 * 10 ^ (-
12)) / (Math.Log(Deq / Ds))
End Select

```

4. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha múltipla.

4.5 Edite os procedimentos de evento de menu Linhas de Transmissão Aéreas

1. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como linha_curta.vb e clique em Ok.
2. Clique no menu Linhas de Transmissão Aéreas do formulário para exibir seus comandos.

3. Dê um clique duplo no comando Linha curta do menu para exibir um procedimento de evento para o comando no Code Editor.

O procedimento de evento `LinhaCurtaToolStripMenuItem_Click` aparece no Code Editor.

4. Digite a seguinte instrução de programa:

```
linha_curta.Show( )
```

Faça o mesmo procedimento para Linha Linha média, só que nomeie como `linha_media.vb`, e digite a seguinte instrução de programa:

```
linha_media.Show( )
```

Faça o mesmo procedimento também para Linha longa, só que nomeie como `linha_longa.vb`, e digite a seguinte instrução de programa:

```
linha_longa.Show( )
```

4.5.1 Insira objetos ao menu Linha curta

Configure as seguintes propriedades para os objetos: 15 Labels, 6 Buttons, 2 RadioButton, 11 TextBox, 2 GroupBox. O menu irá ficar como mostra a figura 3.7.

Linhas de transmissão curta

Entre com os valores

Carga na linha:

Potência ativa (MW):
 Potência aparente (MVA):

Fator de potência:

Impedância série:
 Real: Imaginária:

Não achou a impedância série?

Tensão na linha (kV):

Corrente no gerador:
 Real: Defasagem:

Tensão no gerador:
 Real: Defasagem:

Potência fornecida:

Regulação de tensão:

Figura 3.7 – Menu linhas de transmissão curta.

1. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como impedancia_curta.vb e clique em Ok.
2. Dê um clique duplo no objeto (Clique aqui) para editar seu procedimento de evento:


```
impedancia_curta.Show( )
```
3. Clique duas vezes impedancia_curta.vb no Solution Explorer e configure as seguintes propriedades.

Objeto	Propriedade	Configuração
Label1	Text	“Digite o valor da resistência:”
Label2	Text	“Digite o valor da indutância:”
Label3	Text	“Valor da impedância:”
Label4	Text	“Real:”
Label5	Text	“Imaginária:”
Buton1	Text	“Calcular”
impedancia_curta	Text	“Cálculo da impedância”

Tabela 4.8 – Objetos do menu Cálculo da impedância.

Atribua TextBox somente para os Labels: “Digite o valor da resistência”, “Digite o valor da indutância”, “Real” e “Imaginário”

4. Clique duas vezes no objeto calcular editar seu procedimento de evento:

```
Dim R, L, XL As Double
R = TextBox1.Text
L = TextBox2.Text

XL = (2 * Math.PI * 60 * L)

TextBox3.Text = R
    TextBox4.Text = XL
```

5. Agora volte para linha_curta.vb no Solution Explorer e clique duas vezes no objeto calcular (que esta relacionado a Corrente no gerador) editar seu procedimento de evento:

```
Dim MVA, MW, fp, VR, IR, r, j, defasagem, graus, defasagem2 As Double

MW = TextBox3.Text
MVA = TextBox3.Text
fp = TextBox5.Text
VR = TextBox4.Text
' IR=
graus = -(180 / Math.PI)
defasagem = (Math.Acos(fp))
defasagem2 = defasagem * graus
If RadioButton1.Checked = True Then
    TextBox6.Text = (MW * 10 ^ 6) / (Math.Sqrt(3) * VR
* 10 ^ 3 * fp)

ElseIf RadioButton2.Checked = True Then
```

```

* 10 ^ 3)
    TextBox6.Text = (MVA * 10 ^ 6) / (Math.Sqrt(3) * VR
    TextBox6.Text = FormatNumber(TextBox6.Text, "5")

```

```
End If
```

```
TextBox9.Text = defasagem2
```

```
TextBox9.Text = FormatNumber(TextBox9.Text, "5")
```

6. Clique duas vezes no objeto calcular (que esta relacionado a Tensão no gerador) para editar seu procedimento de evento:

```

Dim MVA, MW, fp, VR, IR, Zr, Zimg, Z1, a, An, IR_real, Vs_real,
-
-
IR_img, Vs_def, defasagem2, defasagem, graus, Vs, x, y,
-
Vs2, Vs3, Ireal_MW, Iimg_MW As Double
MW = TextBox3.Text
MVA = TextBox3.Text
Zr = TextBox1.Text
Zimg = TextBox2.Text
fp = TextBox5.Text
VR = TextBox4.Text
IR_real = TextBox6.Text
IR_img = TextBox9.Text

Ireal_MW = TextBox6.Text
Iimg_MW = TextBox9.Text

'Calculo da corrente no gerador (IR):
graus = -(180 / Math.PI)
defasagem = (Math.Acos(fp))
defasagem2 = defasagem * graus

If RadioButton2.Checked = True Then
^ 3)
    IR_real = (MVA * 10 ^ 6) / (Math.Sqrt(3) * VR * 10

    IR_img = defasagem2
    'Impedancia serie:
    'Transformando de retangular para POLAR
    'Onde Z1(parte real) An(defasagem)
    Z1 = Math.Sqrt(Zr ^ 2 + Zimg ^ 2)
    a = Math.Atan(Zimg / Zr)
    An = (a * 180) / 3.14159265359
    'Para a forma polar:
    Vs = (Z1 * IR_real) 'Real
    Vs_def = An + IR_img ' Defasagem

```

```

Vs_def)
    'Transformando de polar para retangular (Vs e
x = Vs * Math.Sin((Vs_def * 3.1415926535) / 180)
y = Vs * Math.Cos((Vs_def * 3.1415926535) / 180)

    'Ainda na forma retangular
Vs2 = Vs + (VR * 10 ^ 3)

    'Transformando de retangular para polar (Vs2 e x)
Vs3 = Math.Sqrt(Vs2 ^ 2 + x ^ 2)
a = Math.Atan(x / Vs2)
An = (a * 180) / (Math.PI)

    TextBox7.Text = Vs3      'Real
    TextBox10.Text = An * 100    'Defasagem

ElseIf RadioButton1.Checked = True Then
    Ireal_MW = (MW * 10 ^ 6) / (Math.Sqrt(3) * VR * 10
^ 3 * fp)
    'Impedancia serie:
    'Transformando de retangular para POLAR
    'Onde Z1(parte real) An(defasagem)
    Z1 = Math.Sqrt(Zr ^ 2 + Zimg ^ 2)
    a = Math.Atan(Zimg / Zr)
    An = (a * 180) / 3.14159265359
    'Para a forma polar:
    Vs = (Z1 * Ireal_MW) 'Real
    Vs_def = An + IR_img ' Defasagem

Vs_def)
    'Transformando de polar para retangular (Vs e
x = Vs * Math.Sin((Vs_def * 3.1415926535) / 180)
y = Vs * Math.Cos((Vs_def * 3.1415926535) / 180)

    'Ainda na forma retangular
Vs2 = Vs + (VR * 10 ^ 3)

    'Transformando de retangular para polar (Vs2 e x)
Vs3 = Math.Sqrt(Vs2 ^ 2 + x ^ 2)
a = Math.Atan(x / Vs2)
An = (a * 180) / (Math.PI)

    TextBox7.Text = Vs3      'Real
    TextBox10.Text = An * 100    'Defasagem

End If

```

7. Clique duas vezes no objeto calcular (que esta relacionada Potência fornecida) editar seu procedimento de evento:

```
Dim VL, I, fp As Double
fp = TextBox5.Text
VL = TextBox7.Text
I = TextBox6.Text

TextBox8.Text = Math.Sqrt(3) * fp * VL * I & " W"
```

8. Clique duas vezes no objeto calcular (que esta relacionada Regulação de tensão) editar seu procedimento de evento:

```
Dim R, Vs, VR, VR1 As Double
'R%= ((Vs-VR)/VR)*100
Vs = TextBox7.Text
VR1 = TextBox4.Text

'Fase:
VR = VR1 / Math.Sqrt(3)

R = ((Vs - VR * 10 ^ 3) / (VR * 10 ^ 3)) * 100

TextBox11.Text = R & " %"
```

9. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code

Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha

curta.

4.5.2 Insira objetos ao menu Linha média

Configure as seguintes propriedades para os objetos: 19 Labels, 12 TextBox, 7 Buttons, 2 RadioButton, 2 GroupBox. O menu irá ficar como mostra a figura 3.8.

The screenshot shows a Windows application window titled "Linhas de transmissão média". The window contains the following elements:

- Impedância série:** Two input fields for "Real" and "Imaginária". Below them is a button labeled "Clique aqui" with the text "Não achou a impedância série?".
- Admitância:** Two input fields for "Real" and "Defasagem". Below them is a button labeled "Clique aqui" with the text "Não achou a admitância?".
- Carga na linha:** An input field for "Carga na linha". Below it are two radio buttons: "Potência ativa (MW)" and "Potência aparente (MVA)".
- Tensão na linha (kV):** An input field for "Tensão na linha (kV)".
- Fator de potência:** An input field for "Fator de potência".
- Corrente no gerador:** Two input fields for "Real" and "Defasagem". Below them is a "Calcular" button.
- Tensão no gerador:** Two input fields for "Real" and "Defasagem". Below them is a "Calcular" button.
- Potência fornecida:** An input field for "Potência fornecida". Below it is a "Calcular" button.
- Regulação de tensão:** An input field for "Regulação de tensão". Below it is a "Calcular" button.
- A "Fechar" button is located at the bottom center of the window.

Figura 3.8 – Menu Linhas de transmissão média.

1. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como impedancia_media.vb e clique em Ok.
2. Dê um clique duplo no objeto (Clique aqui - impedância) para editar seu procedimento de evento:


```
impedancia_media.Show( )
```
3. Clique duas vezes impedancia_media.vb no Solution Explorer e configure as seguintes propriedades.

Objeto	Propriedade	Configuração
Label1	Text	“Digite o valor da resistência:”
Label2	Text	“Digite o valor da indutância:”
Label3	Text	“Valor da impedância:”
Label4	Text	“Real:”
Label5	Text	“Imaginária:”
Buton1	Text	“Calcular”
impedancia_media	Text	“Cálculo da impedância”

Tabela 4.9 – Objetos do menu Cálculo da impedância.

Atribua TextBox somente para os Labels: “Digite o valor da resistência”, “Digite o valor da indutância”, “Real” e “Imaginário”

4. Clique duas vezes no objeto calcular editar seu procedimento de evento:

```
Dim R, L, XL As Double
R = TextBox1.Text
L = TextBox2.Text

XL = (2 * Math.PI * 60 * L)

TextBox3.Text = R
TextBox4.Text = XL
```

5. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como admitancia_media.vb e clique em Ok.
6. Dê um clique duplo no objeto (Clique aqui - admitância) para editar seu procedimento de evento:

```
admitancia_media.Show()
```

7. Clique duas vezes admitância_media.vb no Solution Explorer e configure as seguintes propriedades.

Objeto	Propriedade	Configuração
Label1	Text	“Digite o valor da condutância:”
Label2	Text	“Digite o valor da capacitância:”
Label3	Text	“Valor da admitância:”
Label4	Text	“Real:”
Label5	Text	“Defasagem:”

Buton1	Text	“Calcular”
impedancia_media	Text	“Cálculo da admitância”

Tabela 4.10 – Objetos do menu Cálculo da admitância.

Atribua TextBox somente para os Labels: “Digite o valor da condutância:”, “Digite o valor da capacitância:”, “Real” e “Defasagem”.

8. Clique duas vezes no objeto calcular editar seu procedimento de evento:

```
Dim G, C, Xc, G_real, Xc_def, a As Double
G = TextBox1.Text
C = TextBox2.Text

Xc = 1 / (2 * Math.PI * 60 * C)

G_real = Math.Sqrt(G ^ 2 + Xc ^ 2)
a = Math.Atan(Xc / G)
Xc_def = (a * 180) / (Math.PI)

TextBox3.Text = G_real
TextBox4.Text = Xc_def
```

9. Agora volte para linha_media.vb no Solution Explorer e clique duas vezes no objeto calcular (que esta relacionado a Corrente no gerador) editar seu procedimento de evento:

```
Dim VR, VR1, fp, MVA, MW, Zreal, Yreal, Zimg, Ydef, graus,
defasagem, defasagem2, IR, IR2, IR_def, Z, a, An, I, I_def, x,
y, I2, I_def2, I3, I_def3, a1, Z1, An1, I4, I_def4, I5, I_def5,
x1, y1, I6, I_def6, Z2, a2, An2, I7, I_def7, I8, I_def8, x2,
y2, x3, y3, I9, I_def9, I10, I_def10, a3 As Double

VR1 = TextBox6.Text
fp = TextBox5.Text
MW = TextBox7.Text
Zreal = TextBox1.Text
Zimg = TextBox2.Text
Yreal = TextBox3.Text
Ydef = TextBox4.Text

graus = -(180 / Math.PI)
defasagem = (Math.Acos(fp))
defasagem2 = defasagem * graus
```

```

VR = (VR1 * 10 ^ 3) / Math.Sqrt(3)

If RadioButton1.Checked = True Then
    IR = ((MW * 10 ^ 6) / ((Math.Sqrt(3) * VR * 10 ^ 3
* fp))) * 10 ^ 3
    IR_def = defasagem2

    'Fórmula geral:
    'Is= Y*((Z*Y/4)+1)+((Z*Y/2)+1)*IR

    'Transformar Z para forma polar:
    Z = Math.Sqrt(Zreal ^ 2 + Zimg ^ 2) 'impedancia Z
    a = Math.Atan(Zimg / Zreal)
    An = (a * 180) / (Math.PI)           'Defasagem Z

    'Multiplicar Z e Y na forma polar e dividir por 4:
    I = (Z * Yreal) / 4
    I_def = An + Ydef

    'Somar +1 e multiplicar por Y
    'Transformar de polar para RETANGULAR (I e I_def):
    x = I * Math.Sin((I_def * 3.1415926535) / 180)
'Img
    y = I * Math.Cos((I_def * 3.1415926535) / 180)
'Real

    'Soma +1:
    I2 = y + 1
    I_def2 = x

    'Transformar para polar (I2 e I_def2):
    Z1 = Math.Sqrt(I2 ^ 2 + I_def2 ^ 2)
    a1 = Math.Atan(I_def2 / I2)
    An1 = (a1 * 180) / (Math.PI)

    I3 = Z1
    I_def3 = An1

    'Multiplicar ((Z*Y/4)+1) por Y:

    I4 = Z1 * Yreal
    I_def4 = An1 + Ydef
    'fica faltando.... +((Z*Y/2)+1)*IR
    'Multiplicar Z por Y e dividir por 2 na forma
polar:

    I5 = (Z * Yreal) / 2
    I_def5 = An + Ydef

```

```

'Somar +1 e muktiplicar por IR:
'Transformar de polar para RETANGULAR (I e I_def):
x1 = I5 * Math.Sin((I_def5 * 3.1415926535) / 180)
'Img
y1 = I5 * Math.Cos((I_def5 * 3.1415926535) / 180)
'Real

'Soma +1:
I6 = y + 1
I_def6 = x

'Transformar para polar (I6 e I_def6):
Z2 = Math.Sqrt(I6 ^ 2 + I_def6 ^ 2)
a2 = Math.Atan(I_def6 / I6)
An2 = (a1 * 180) / (Math.PI)

I7 = Z2
I_def7 = An2

'Multiplicando por IR:
I8 = Z2 * IR
I_def8 = An2 + IR_def
'Transformar para retangular: (I4, I_def4) e (I8,
Idef8):
'Para I4:
x2 = I4 * Math.Sin((I_def4 * 3.1415926535) / 180)
'Img
y2 = I4 * Math.Cos((I_def4 * 3.1415926535) / 180)
'Real

'Para I8:
x3 = I8 * Math.Sin((I_def8 * 3.1415926535) / 180)
'Img
y3 = I8 * Math.Cos((I_def8 * 3.1415926535) / 180)
'Real

I9 = y2 + y3
I_def9 = x2 + x3

'Transformar para polar:
I10 = Math.Sqrt(I9 ^ 2 + I_def9 ^ 2)
a3 = Math.Atan(I_def9 / I9)
I_def10 = (a3 * 180) / (Math.PI)

TextBox9.Text = I10
TextBox10.Text = I_def10

```

```

ElseIf RadioButton2.Checked = True Then
    graus = -(180 / Math.PI)
    defasagem = (Math.Acos(fp))
    defasagem2 = defasagem * graus
    VR = (VR1 * 10 ^ 3) / Math.Sqrt(3)
    IR2 = ((MVA * 10 ^ 6) / ((Math.Sqrt(3) * VR * 10 ^
3))) * 10 ^ 3
    IR_def = defasagem2

    'Fórmula geral:
    'Is= Y*((Z*Y/4)+1)+((Z*Y/2)+1)*IR

    'Transformar Z para forma polar:
    Z = Math.Sqrt(Zreal ^ 2 + Zimg ^ 2) 'impedancia Z
    a = Math.Atan(Zimg / Zreal)
    An = (a * 180) / (Math.PI)           'Defasagem Z

    'Multiplicar Z e Y na forma polar e dividir por 4:
    I = (Z * Yreal) / 4
    I_def = An + Ydef

    'Somar +1 e multiplicar por Y
    'Transformar de polar para RETANGULAR (I e I_def):
    x = I * Math.Sin((I_def * 3.1415926535) / 180)
'Img
    y = I * Math.Cos((I_def * 3.1415926535) / 180)
'Real

    'Soma +1:
    I2 = y + 1
    I_def2 = x

    'Transformar para polar (I2 e I_def2):
    Z1 = Math.Sqrt(I2 ^ 2 + I_def2 ^ 2)
    a1 = Math.Atan(I_def2 / I2)
    An1 = (a1 * 180) / (Math.PI)

    I3 = Z1
    I_def3 = An1

    'Multiplicar ((Z*Y/4)+1) por Y:

    I4 = Z1 * Yreal
    I_def4 = An1 + Ydef
    'fica faltando.... +((Z*Y/2)+1)*IR2
    'Multiplicar Z por Y e dividir por 2 na forma
polar:

```

```

I5 = (Z * Yreal) / 2
I_def5 = An + Ydef

'Somar +1 e moltiplicar por IR2:
'Transformar de polar para RETANGULAR (I e I_def):
x1 = I5 * Math.Sin((I_def5 * 3.1415926535) / 180)
'Img
y1 = I5 * Math.Cos((I_def5 * 3.1415926535) / 180)
'Real

'Soma +1:
I6 = y + 1
I_def6 = x
'Transformar para polar (I6 e I_def6):
Z2 = Math.Sqrt(I6 ^ 2 + I_def6 ^ 2)
a2 = Math.Atan(I_def6 / I6)
An2 = (a1 * 180) / (Math.PI)

I7 = Z2
I_def7 = An2

'Multiplicando por IR2:
I8 = Z2 * IR2
I_def8 = An2 + IR_def
'Transformar para retangular: (I4, I_def4) e (I8,
Idef8):
'Para I4:
x2 = I4 * Math.Sin((I_def4 * 3.1415926535) / 180)
'Img
y2 = I4 * Math.Cos((I_def4 * 3.1415926535) / 180)
'Real

'Para I8:
x3 = I8 * Math.Sin((I_def8 * 3.1415926535) / 180)
'Img
y3 = I8 * Math.Cos((I_def8 * 3.1415926535) / 180)
'Real

I9 = y2 + y3
I_def9 = x2 + x3

'Transformar para polar:
I10 = Math.Sqrt(I9 ^ 2 + I_def9 ^ 2)
a3 = Math.Atan(I_def9 / I9)
I_def10 = (a3 * 180) / (Math.PI)

TextBox9.Text = I10
TextBox10.Text = I_def10

```

End If

10. Clique duas vezes no objeto calcular (que esta relacionado a Tensão no

gerador) editar seu procedimento de evento:

```
Dim VR, fp, MVA, MW, Zreal, Yreal, Zimg, Ydef, graus,
defasagem, defasagem2, IR, IR2, IR_def, VR1, Z, a, An, Vs,
Vs_def, x, y, Vs2, Vs_def2, Z1, An2, Vs3, Vs_def3, Vs4,
Vs_def4, x1, y1, x2, y2, Vs5, Vs_def5, Vs6, Vs_def6 As Double
```

```
VR1 = TextBox6.Text
fp = TextBox5.Text
MW = TextBox7.Text
MVA = TextBox7.Text
Zreal = TextBox1.Text
Zimg = TextBox2.Text
Yreal = TextBox3.Text
Ydef = TextBox4.Text
If RadioButton1.Checked = True Then
    graus = -(180 / Math.PI)
    defasagem = (Math.Acos(fp))
    defasagem2 = defasagem * graus

    VR = (VR1 * 10 ^ 3) / Math.Sqrt(3)
    IR = (MVA * 10 ^ 6) / ((Math.Sqrt(3) * VR1 * 10 ^
3))

    IR_def = defasagem2
    'Transformar Z para forma polar:
    Z = Math.Sqrt(Zreal ^ 2 + Zimg ^ 2)
    a = Math.Atan(Zimg / Zreal)
    An = (a * 180) / (Math.PI)

    'Multiplicar Z por Y e dividir por 2 na forma
polar:
    Vs = (Z * Yreal) / 2
    Vs_def = An + Ydef

    'Transformar Vs e Vs_def para a forma retangular
para somar +1:
    x = Vs * Math.Sin((Vs_def * 3.1415926535) / 180)
'Img
    y = Vs * Math.Cos((Vs_def * 3.1415926535) / 180)
'Real

    Vs2 = Vs + y + 1
    Vs_def2 = x
```

```

Vs_def2)
    'Transformar de retangular para polar (Vs2 e
Vs_def2)
    Z1 = Math.Sqrt(Vs2 ^ 2 + Vs_def2 ^ 2)
    a = Math.Atan(Vs_def2 / Vs2)
    An2 = (a * 180) / (Math.PI)

    Vs3 = Z1 * VR
    Vs_def3 = An2 + 0

    'Multiplicar Z com IR para depois somar
((ZY/2)+1)*VR = (Vs3 e Vs_def)
    'Transformar Z para forma polar:
    Vs4 = Z * IR
    Vs_def4 = An + IR_def

    'Transformar Vs4, Vs_def4, Vs3 e Vs_def3 para
retangular para poder somar:
    'Para Vs3 e Vs_def3:
    x1 = Vs3 * Math.Sin((Vs_def3 * 3.1415926535) / 180)
    y1 = Vs3 * Math.Cos((Vs_def3 * 3.1415926535) / 180)
    'Para Vs4 e Vs_def4:
    x2 = Vs4 * Math.Sin((Vs_def4 * 3.1415926535) / 180)
    y2 = Vs4 * Math.Cos((Vs_def4 * 3.1415926535) / 180)

    Vs5 = y1 + y2
    Vs_def5 = x1 + x2

    'Transformar para polar:
    Vs6 = Math.Sqrt(Vs5 ^ 2 + Vs_def5 ^ 2)
    a = Math.Atan(Vs_def5 / Vs5)
    Vs_def6 = (a * 180) / (Math.PI)

    TextBox11.Text = Vs6
    TextBox12.Text = Vs_def6 * 100

ElseIf RadioButton2.Checked = True Then
    graus = -(180 / Math.PI)
    defasagem = (Math.Acos(fp))
    defasagem2 = defasagem * graus

    VR = (VR1 * 10 ^ 3) / Math.Sqrt(3)
    IR2 = (MW * 10 ^ 6) / ((Math.Sqrt(3)) * VR1 * 10 ^ 3
* fp))

    IR_def = defasagem2
    'Transformar Z para forma polar:
    Z = Math.Sqrt(Zreal ^ 2 + Zimg ^ 2)
    a = Math.Atan(Zimg / Zreal)

```

```

An = (a * 180) / (Math.PI)

'Multiplicar Z por Y e dividir por 2 na forma
polar:
Vs = (Z * Yreal) / 2
Vs_def = An + Ydef

'Transformar Vs e Vs_def para a forma retangular
para somar +1:
x = Vs * Math.Sin((Vs_def * 3.1415926535) / 180)
'Img
y = Vs * Math.Cos((Vs_def * 3.1415926535) / 180)
'Real

Vs2 = Vs + y + 1
Vs_def2 = x

'Transformar de retangular para polar (Vs2 e
Vs_def2)
Z1 = Math.Sqrt(Vs2 ^ 2 + Vs_def2 ^ 2)
a = Math.Atan(Vs_def2 / Vs2)
An2 = (a * 180) / (Math.PI)

Vs3 = Z1 * VR
Vs_def3 = An2 + 0

'Multiplicar Z com IR2 para depois somar
((ZY/2)+1)*VR = (Vs3 e Vs_def)
'Transformar Z para forma polar:
Vs4 = Z * IR2
Vs_def4 = An + IR_def

'Transformar Vs4, Vs_def4, Vs3 e Vs_def3 para
retangular para poder somar:
'Para Vs3 e Vs_def3:
x1 = Vs3 * Math.Sin((Vs_def3 * 3.1415926535) / 180)
y1 = Vs3 * Math.Cos((Vs_def3 * 3.1415926535) / 180)
'Para Vs4 e Vs_def4:
x2 = Vs4 * Math.Sin((Vs_def4 * 3.1415926535) / 180)
y2 = Vs4 * Math.Cos((Vs_def4 * 3.1415926535) / 180)

Vs5 = y1 + y2
Vs_def5 = x1 + x2

'Transformar para polar:
Vs6 = Math.Sqrt(Vs5 ^ 2 + Vs_def5 ^ 2)
a = Math.Atan(Vs_def5 / Vs5)

```



```

Vs_def6 = (a * 180) / (Math.PI)

TextBox11.Text = Vs6
TextBox12.Text = Vs_def6 * 100
End If

```

11. Clique duas vezes no objeto calcular (que esta relacionado a Potência fornecida) para editar seu procedimento de evento:

```

Dim VL, I, fp As Double
fp = TextBox5.Text
VL = TextBox11.Text
I = TextBox9.Text

TextBox8.Text = Math.Sqrt(3) * fp * VL * I & " W"

```

12. Clique duas vezes no objeto calcular (que esta relacionado a Regulação de tensão) para editar seu procedimento de evento:

```

Dim R, Vs, VR, VR1, A, Zreal, Zdef, Yreal, Ydef, A1, Zreal1,
Zimg, A_2, A_2def, A_3, A_3img, A_4, A_4def, A_5, A_5def, Vss
As Double
'R%= (((Vs/A)-VR*10^3))/(VR*10^3))*100
'A= ((Z*Y)/2)+1
Vs = TextBox11.Text
VR1 = TextBox6.Text
Zreal1 = TextBox1.Text
Zimg = TextBox2.Text
Yreal = TextBox3.Text
Ydef = TextBox4.Text
'Transformar Z para polar:
Zreal = Math.Sqrt(Zreal1 ^ 2 + Zimg ^ 2)
A1 = Math.Atan(Zimg / Zreal1)
Zdef = (A1 * 180) / (Math.PI)

'Multiplicando Z * Y:
A_2 = (Zreal * Ydef) / 2
A_2def = Zdef + Ydef

'Transformando A_2 e A_2def para retangular(para somar
+1):
A_3 = A_2 * Math.Sin((A_2def * 3.1415926535) / 180)
A_3img = A_2 * Math.Cos((A_2def * 3.1415926535) / 180)

'Somando +1

```

```

A_4 = A_3 + 1
A_4def = A_3img

'Transformando de novo para polar:
A_5 = Math.Sqrt(A_4 ^ 2 + A_4def ^ 2)
A = Math.Atan(A_4def / A_4)
A_5def = (A * 180) / (Math.PI)

'Calculo da Regulação de tensao:
'Fase:
Vss = Vs / A_5
VR = VR1 / Math.Sqrt(3)
R = (((Vss) - VR * 10 ^ 3)) / (VR * 10 ^ 3) * 100
TextBox13.Text = R

```

13. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button2_Click e End Sub no Code Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha média.

4.5.3 Insira objetos ao menu Linha longa

Configure as seguintes propriedades para os objetos: 20 Labels, 14 TextBox, 7 Buttons, 2 RadioButton, 2 GroupBox. O menu irá ficar como mostra a figura 3.9.

Linhas de transmissão longa

Entre com os valores

Impedância série: Real: Defasagem: Admitância: Real: Defasagem:

Não achou a impedância série? Clique aqui Não achou a admitância? Clique aqui

Comprimento da linha (mi): Tensão na linha (kV):

Carga na linha: Potência ativa (MW) Potência aparente (MVA)

Fator de potência:

Corrente no gerador: Real: Defasagem: Tensão no gerador: Real: Defasagem:

Calcular Calcular

Potência fornecida: Regulação de tensão:

Calcular Calcular

Fechar

Figura 3.9 - Menu Linhas de transmissão média.

1. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como impedancia_longa.vb e clique em Ok.
2. Dê um clique duplo no objeto (Clique aqui - impedância) para editar seu procedimento de evento:


```
impedancia_longa.Show( )
```
3. Clique duas vezes impedancia_longa.vb no Solution Explorer e configure as seguintes propriedades.

Objeto	Propriedade	Configuração
Label1	Text	“Digite o valor da resistência:”
Label2	Text	“Digite o valor da indutância:”
Label3	Text	“Valor da impedância:”
Label4	Text	“Real:”
Label5	Text	“Defasagem:”
Buton1	Text	“Calcular”
impedancia_media	Text	“Cálculo da impedância”

Tabela 4.11 – Objetos do menu Cálculo da impedância.

Atribua TextBox somente para os Labels: “Digite o valor da resistência”, “Digite o valor da indutância”, “Real” e “Defasagem”

4. Clique duas vezes no objeto calcular editar seu procedimento de evento:

```
Dim R, L, XL, R_real, XL_def, a As Double
```

```
R = TextBox1.Text
```

```
L = TextBox2.Text
```

```
XL = (2 * Math.PI * 60 * L)
```

```
R_real = Math.Sqrt(R ^ 2 + XL ^ 2)
```

```
a = Math.Atan(XL / R)
```

```
XL_def = (a * 180) / (Math.PI)
```

```
TextBox3.Text = R_real
```

```
TextBox4.Text = XL_def
```

5. No Visual Studio, vai ao menu Project e em seguida clique em Add Windows Form e escolha Windows Form, nomeie como admitancia_longa.vb e clique em Ok.

6. Dê um clique duplo no objeto (Clique aqui - admitância) para editar seu procedimento de evento:

```
admitancia_longa.Show()
```

7. Clique duas vezes admitância_longa.vb no Solution Explorer e configure as seguintes propriedades.

Objeto	Propriedade	Configuração
Label1	Text	“Digite o valor da condutância:”
Label2	Text	“Digite o valor da capacitância:”
Label3	Text	“Valor da admitância:”
Label4	Text	“Real:”
Label5	Text	“Defasagem:”
Button1	Text	“Calcular”
impedancia_media	Text	“Cálculo da admitância”

Tabela 4.12 – Objetos do menu Cálculo da admitância.

Atribua TextBox somente para os Labels: “Digite o valor da condutância:”, “Digite o valor da capacitância:”, “Real” e “Defasagem”.

8. Clique duas vezes no objeto calcular editar seu procedimento de evento:

```
Dim G, C, Xc, a, G_real, Xc_def As Double
G = TextBox1.Text
C = TextBox2.Text
Xc = 1 / (2 * Math.PI * 60 * C)
G_real = Math.Sqrt(G ^ 2 + Xc ^ 2)
a = Math.Atan(Xc / G)
Xc_def = (a * 180) / (Math.PI)
TextBox3.Text = G_real
TextBox4.Text = Xc_def
```

9. Agora volte para linha_longa.vb no Solution Explorer e clique duas vezes no objeto calcular (que esta relacionado a Corrente no gerador) editar seu procedimento de evento:

```
Dim l, fp, MW, MVA, VR1, VR, Zreal, Zdef, Yreal, Ydef, Zc_real,
Zc_def, IR, IR2, yl_alpha, yl_beta, x1, y1, alpha, beta,
cosh_real, cosh_img, cosh_reall1, cosh_def1, a, sinh_real,
sinh_img, sinh_reall1, sinh_def1, I_real, I_def, I_real2, I_def2,
x2, y2, I_real3, I_def3, x3, y3, I_real4, I_def4, I_real5,
I_def5, I_realF, I_defF As Double

l = TextBox5.Text
VR1 = TextBox7.Text
MW = TextBox8.Text
MVA = TextBox8.Text
fp = TextBox6.Text
Zreal = TextBox1.Text
Zdef = TextBox2.Text
Yreal = TextBox3.Text
```

```

Ydef = TextBox4.Text

'Calculo da impedancia caracteristica:
Zc_real = Math.Sqrt((Zreal / Yreal))
Zc_def = (Zdef - Ydef) / 2

'Calculo da tensao:
VR = (VR1 * 10 ^ 3) / (Math.Sqrt(3))

'Calculo da corrente:
If RadioButton1.Checked = True Then
    IR = (MW * 10 ^ 6) / (Math.Sqrt(3) * VR1 * 10 ^ 3 *
fp)

    yl_alpha = 1 * (Math.Sqrt(Zreal * Yreal))
    yl_beta = (Zdef + Ydef) / 2

'Converter yl_alpha e yl_beta para a forma
retangular:
x1 = yl_alpha * Math.Sin((yl_beta * 3.1415926535) /
180) 'imgg
y1 = yl_alpha * Math.Cos((yl_beta * 3.1415926535) /
180) 'REAL
alpha = y1
beta = x1
cosh_real = (Math.Cosh(alpha) * Math.Cos(beta))
cosh_img = (Math.Sinh(alpha) * Math.Sin(beta))

'Converter cosh_real e cosh_img para a forma polar:
cosh_reall = Math.Sqrt(cosh_real ^ 2 + cosh_img ^
2)

a = Math.Atan(cosh_img / cosh_real)
cosh_def1 = (a * 180) / (Math.PI)
senh_real = (Math.Sinh(alpha) * Math.Cos(beta))
senh_img = (Math.Cosh(alpha) * Math.Sin(beta))

'Converter para senh_real e senh_img a forma polar:
senh_reall = Math.Sqrt(senh_real ^ 2 + senh_img ^
2)

a = Math.Atan(senh_img / senh_real)
senh_def1 = (a * 180) / (Math.PI)
CALCULO DA CORRENTE NO GERADOR
'FORMULA: Is= (IR*cosh) + (VR/Zc)*Senh
I_real = (IR * cosh_reall)
I_def = cosh_def1

'Converter Vs_real e Vs_def para retangular:

```

```

x2 = I_real * Math.Sin((I_def * 3.1415926535) /
180) 'img
y2 = I_real * Math.Cos((I_def * 3.1415926535) /
180)'real
I_real2 = y2
I_def2 = x2
'Dividir VR por Zc e multiplicar por Senh:
I_real3 = (VR / Zc_real) * senh_reall
I_def3 = (0 - Zc_def) + (senh_def1)
'Converter I_real3 e I_def3 para retangular:
x3 = I_real3 * Math.Sin((I_def3 * 3.1415926535) /
180)
y3 = I_real3 * Math.Cos((I_def3 * 3.1415926535) /
180)

I_real4 = y3
I_def4 = x3

'Somar (I_real2; I_def2) e (I_real4; Idef4):
I_real5 = y2 + y3
I_def5 = x2 + x3

'Transformar de retangular para polar:

I_realF = Math.Sqrt(I_real5 ^ 2 + I_def5 ^ 2)
a = Math.Atan(I_def5 / I_real5)
I_defF = (a * 180) / (Math.PI)
TextBox9.Text = I_realF
TextBox10.Text = I_defF
ElseIf RadioButton2.Checked = True Then
IR2 = (MVA * 10 ^ 6) / (Math.Sqrt(3) * VR * 10 ^ 3)
yl_alpha = 1 * (Math.Sqrt(Zreal * Yreal))
yl_beta = (Zdef + Ydef) / 2
'Converter yl_alpha e yl_beta para a forma
retangular:
x1 = yl_alpha * Math.Sin((yl_beta * 3.1415926535) /
180)
y1 = yl_alpha * Math.Cos((yl_beta * 3.1415926535) /
180)

alpha = y1 'REAL
beta = x1 'imgg
cosh_real = (Math.Cosh(alpha) * Math.Cos(beta))
cosh_img = (Math.Sinh(alpha) * Math.Sin(beta))
'Converter cosh_real e cosh_img para a forma polar:
cosh_reall = Math.Sqrt(cosh_real ^ 2 + cosh_img ^
2)

```

```

a = Math.Atan(cosh_img / cosh_real)
cosh_def1 = (a * 180) / (Math.PI)
senh_real = (Math.Sinh(alpha) * Math.Cos(beta))
senh_img = (Math.Cosh(alpha) * Math.Sin(beta))

'Converter para senh_real e senh_img a forma polar:
senh_reall = Math.Sqrt(senh_real ^ 2 + senh_img ^
2)

a = Math.Atan(senh_img / senh_real)
senh_def1 = (a * 180) / (Math.PI)
CALCULO DA CORRENTE NO GERADOR
'FORMULA: Is= (IR*cosh) + (VR/Zc)*Senh
I_real = (IR2 * cosh_reall)
I_def = cosh_def1

'Converter Vs_real e Vs_def para retangular:
x2 = I_real * Math.Sin((I_def * 3.1415926535) /
180) 'img
y2 = I_real * Math.Cos((I_def * 3.1415926535) /
180) 'real

I_real2 = y2
I_def2 = x2

'Dividir VR por Zc e multiplicar por Senh:
I_real3 = (VR / Zc_real) * senh_reall
I_def3 = (0 - Zc_def) + (senh_def1)

'Converter I_real3 e I_def3 para retangular:
x3 = I_real3 * Math.Sin((I_def3 * 3.1415926535) /
180)
y3 = I_real3 * Math.Cos((I_def3 * 3.1415926535) /
180)

I_real4 = y3
I_def4 = x3

'Somar (I_real2; I_def2) e (I_real4; Idef4):
I_real5 = y2 + y3
I_def5 = x2 + x3

'Transformar de retangular para polar:

I_realF = Math.Sqrt(I_real5 ^ 2 + I_def5 ^ 2)
a = Math.Atan(I_def5 / I_real5)
I_defF = (a * 180) / (Math.PI)

```



```

        TextBox9.Text = I_realF
        TextBox10.Text = I_defF
    End If

```

10. Clique duas vezes no objeto calcular (que esta relacionado a Tensão no gerador) editar seu procedimento de evento:

```

Dim l, fp, MW, MVA, VR1, VR, Zreal, Zdef, Yreal, Ydef, Zc_real,
Zc_def, IR, IR2, yl_alpha, yl_beta, x1, y1, alpha, beta,
cosh_real, cosh_img, cosh_reall, cosh_def1, a, senh_real,
senh_img, senh_reall, senh_def1, Vs_real, Vs_def, Vs_real2,
Vs_def2, x2, y2, Vs_real3, Vs_def3, x3, y3, Vs_real4,
Vs_def4, Vs_realF, Vs_defF As Double

```

```

    l = TextBox5.Text
    VR1 = TextBox7.Text
    MVA = TextBox8.Text
    MW = TextBox8.Text
    fp = TextBox6.Text
    Zreal = TextBox1.Text
    Zdef = TextBox2.Text
    Yreal = TextBox3.Text
    Ydef = TextBox4.Text

```

```

    'Calculo da impedancia caracteristica:

```

```

    Zc_real = Math.Sqrt((Zreal / Yreal))
    Zc_def = (Zdef - Ydef) / 2

```

```

    'Calculo da tensao:

```

```

    VR = (VR1 * 10 ^ 3) / (Math.Sqrt(3))

```

```

    'Calculo da corrente:

```

```

    If RadioButton1.Checked = True Then

```

```

        IR = (MW * 10 ^ 6) / (Math.Sqrt(3) * VR1 * 10 ^ 3 *

```

fp)

```

        yl_alpha = l * (Math.Sqrt(Zreal * Yreal))
        yl_beta = (Zdef + Ydef) / 2

```

```

        'Converter yl_alpha e yl_beta para a forma

```

retangular:

```

        x1 = yl_alpha * Math.Sin((yl_beta * 3.1415926535) /

```

180) 'imgg

```

180) 'REAL
y1 = yl_alpha * Math.Cos((yl_beta * 3.1415926535) /
alpha = y1
beta = x1
cosh_real = (Math.Cosh(alpha) * Math.Cos(beta))
cosh_img = (Math.Sinh(alpha) * Math.Sin(beta))

'Converter cosh_real e cosh_img para a forma polar:
cosh_reall = Math.Sqrt(cosh_real ^ 2 + cosh_img ^
2)
a = Math.Atan(cosh_img / cosh_real)
cosh_def1 = (a * 180) / (Math.PI)
senh_real = (Math.Sinh(alpha) * Math.Cos(beta))
senh_img = (Math.Cosh(alpha) * Math.Sin(beta))
'Converter para senh_real e senh_img a forma polar:
senh_reall = Math.Sqrt(senh_real ^ 2 + senh_img ^
2)
a = Math.Atan(senh_img / senh_real)
senh_def1 = (a * 180) / (Math.PI)
CALCULO DA TENSAO NO GERADOR
Vs_real = (VR * cosh_reall)
Vs_def = cosh_def1
'Converter Vs_real e Vs_def para retangular:
x2 = Vs_real * Math.Sin((Vs_def * 3.1415926535) /
180) 'img
y2 = Vs_real * Math.Cos((Vs_def * 3.1415926535) /
180) 'real

Vs_real2 = y2
Vs_def2 = x2

'Multiplicando IR*Zc*senh:
Vs_real3 = (IR * Zc_real * senh_reall)
Vs_def3 = (Zc_def + senh_def1)

'Converter Vs_real e Vs_def para retangular:
x3 = Vs_real3 * Math.Sin((Vs_def3 * 3.1415926535) /
180) 'img
y3 = Vs_real3 * Math.Cos((Vs_def3 * 3.1415926535) /
180) 'real

'Somar (Vs_real2;Vs_def2) e (Vs_real3;Vs_def3):
Vs_real4 = Vs_real2 + y3
Vs_def4 = Vs_def2 + x3

'Transformar para forma polar:
Vs_realF = Math.Sqrt(Vs_real4 ^ 2 + Vs_def4 ^ 2)

```

```

a = Math.Atan(Vs_def4 / Vs_real4)
Vs_defF = (a * 180) / (Math.PI)

TextBox11.Text = Vs_realF
TextBox12.Text = Vs_defF
ElseIf RadioButton2.Checked = True Then

    IR2 = (MVA * 10 ^ 6) / (Math.Sqrt(3) * VR * 10 ^ 3)
    yl_alpha = 1 * (Math.Sqrt(Zreal * Yreal))
    yl_beta = (Zdef + Ydef) / 2

    'Converter yl_alpha e yl_beta para a forma
retangular:
180) 'imgg
180) 'REAL
    x1 = yl_alpha * Math.Sin((yl_beta * 3.1415926535) /
    y1 = yl_alpha * Math.Cos((yl_beta * 3.1415926535) /
    alpha = y1
    beta = x1
    cosh_real = (Math.Cosh(alpha) * Math.Cos(beta))
    cosh_img = (Math.Sinh(alpha) * Math.Sin(beta))

    'Converter cosh_real e cosh_img para a forma polar:
2)
    cosh_reall = Math.Sqrt(cosh_real ^ 2 + cosh_img ^
    a = Math.Atan(cosh_img / cosh_real)
    cosh_def1 = (a * 180) / (Math.PI)
    senh_real = (Math.Sinh(alpha) * Math.Cos(beta))
    senh_img = (Math.Cosh(alpha) * Math.Sin(beta))
    'Converter para senh_real e senh_img a forma polar:
2)
    senh_reall = Math.Sqrt(senh_real ^ 2 + senh_img ^
    a = Math.Atan(senh_img / senh_real)
    senh_def1 = (a * 180) / (Math.PI)
    CALCULO DA TENSAO NO GERADOR
    Vs_real = (VR * cosh_reall)
    Vs_def = cosh_def1

    'Converter Vs_real e Vs_def para retangular:
180) 'img
180) 'real
    x2 = Vs_real * Math.Sin((Vs_def * 3.1415926535) /
    y2 = Vs_real * Math.Cos((Vs_def * 3.1415926535) /
    Vs_real2 = y2
    Vs_def2 = x2

```

```

'Multiplicando IR*Zc*senh:
Vs_real3 = (IR2 * Zc_real * senh_real1)
Vs_def3 = (Zc_def + senh_def1)

'Converter Vs_real e Vs_def para retangular:
x3 = Vs_real3 * Math.Sin((Vs_def3 * 3.1415926535) /
180) 'img

y3 = Vs_real3 * Math.Cos((Vs_def3 * 3.1415926535) /
180) 'real

'Somar (Vs_real2;Vs_def2) e (Vs_real3;Vs_def3):
Vs_real4 = Vs_real2 + y3
Vs_def4 = Vs_def2 + x3

'Transformar para forma polar:
Vs_realF = Math.Sqrt(Vs_real4 ^ 2 + Vs_def4 ^ 2)
a = Math.Atan(Vs_def4 / Vs_real4)
Vs_defF = (a * 180) / (Math.PI)

TextBox11.Text = Vs_realF
TextBox12.Text = Vs_defF

End If

```

11. Clique duas vezes no objeto calcular (que esta relacionado a Potência fornecida) para editar seu procedimento de evento:

```
Dim l, fp, MW, VR1, VR, VSL, Vs, I, Pt, Zreal, Zdef, Yreal,
Ydef As Double
```

```

l = TextBox5.Text
VR1 = TextBox7.Text
MW = TextBox8.Text
fp = TextBox6.Text
Zreal = TextBox1.Text
Zdef = TextBox2.Text
Yreal = TextBox3.Text
Ydef = TextBox4.Text
Vs = TextBox11.Text
I = TextBox9.Text
'FORMULA; PT= RAIZ(3)*VSL*Is*fp
VSL = Math.Sqrt(3) * Vs
Pt = (Math.Sqrt(3) * fp * I * VSL) / 1000
TextBox13.Text = Pt & " kW"

```

12. Clique duas vezes no objeto calcular (que esta relacionado a Regulação de tensão) para editar seu procedimento de evento:

```
Dim l, fp, MW, VR1, VR, Zreal, Zdef, Yreal, Ydef,
Zc_real, Zc_def, IR, yl_alpha, yl_beta, x1, y1, alpha, beta,
cosh_real, cosh_img, cosh_reall, cosh_def1, a, senh_real,
senh_img, senh_reall, senh_def1, I_real, I_def, I_real2, I_def2,
x2, y2, I_real3, I_def3, x3, y3, I_real4, I_def4, I_real5,
I_def5, I_realF, I_defF As Double
```

```
l = TextBox5.Text
VR1 = TextBox7.Text
MW = TextBox8.Text
fp = TextBox6.Text
Zreal = TextBox1.Text
Zdef = TextBox2.Text
Yreal = TextBox3.Text
Ydef = TextBox4.Text
'Calculo da impedancia caracteristica:
Zc_real = Math.Sqrt((Zreal / Yreal))
Zc_def = (Zdef - Ydef) / 2
'Calculo da tensao:
VR = (VR1 * 10 ^ 3) / (Math.Sqrt(3))
'Calculo da corrente:
IR = (MW * 10 ^ 6) / (Math.Sqrt(3) * VR1 * 10 ^ 3)
yl_alpha = l * (Math.Sqrt(Zreal * Yreal))
yl_beta = (Zdef + Ydef) / 2
'Converter yl_alpha e yl_beta para a forma retangular:
180)'imgg
y1 = yl_alpha * Math.Sin((yl_beta * 3.1415926535) /
180)'REAL
alpha = y1
beta = x1
CALCULO REGULAÇÃO DE TENSAO
cosh_real = (Math.Cosh(alpha) * Math.Cos(beta))

Dim R, Vs As Double
'R%= (((Vs/A)-VR))/VR)*100
Vs = TextBox11.Text
VR1 = TextBox7.Text
'Fase:
VR = VR1 / Math.Sqrt(3)
R = (((Vs / cosh_real) - VR * 10 ^ 3) / (VR * 10 ^ 3))
* 100
TextBox14.Text = R
```

13. Agora de um duplo clique no botão Fechar e digite a seguinte instrução de programa entre as instruções Private Sub Button7_Click e End Sub no Code Editor:

```
Me.Close()
```

Esta instrução serve para que o usuário possa fechar a janela do menu Linha longa.

Para fazer o download do programa, segue o link abaixo:

<http://www.megaupload.com/pt/?d=H0OZS545>

http://www.4shared.com/rar/4oRBWVYR/Modelagem_de_Linhas_de_Transmi.html?

4.6 Conclusão

Concluimos que os software tem uma grande importância na operação do sistema elétrico de transmissão desenvolvendo uma ferramenta computacional capaz de determinar a operação da modelagem de linhas de transmissão que atenda a demanda de forma confiável e a baixos custos em um determinado horizonte de planejamento, dadas as condições de operações matemáticas modelagem de linhas de transmissão às quais estarão sujeitas. Além da aplicação em software e operação, essa ferramenta tem grande importância nos estudos dos sistemas, incluindo a caracterização do comportamento das linhas de transmissão, individualmente, e do comportamento das operações.

4.7 Bibliografia

MONTICELLI, Alcir & GARCIA, Ariovaldo. Introdução a Sistemas de Energia Elétrica. Ed. UNICAMP, 1999, 251p.

GRAINGER, John J. & STEVENSON, William D.. Power System Analysis. 1ª. Ed. MCGRAW HILL BOOK CO, 1993. 787p.

OLIVEIRA, Carlos César Barioni. Introdução a Sistemas Elétricos de Potência. 2ª. Ed. Edgard Blucher, 1996 467p.

BICHELS, Arlei. Carregamento de linhas de transmissão. Determinação de limites “normal” a de “sobrecarga”. Nota técnica SPE/DPET – 009/83 – COPEL.

CATÁLOGO ALCOA – Alumínio Company of America. Resistance and Reactance of Aluminium Conductors. 1960 – EUA.

CHARD, F. de la. Electricity Supply: Transmission and Distribution. Longman Group Limited, London, 1976.

CHECA, L. M. Líneas de transporte de Energia. Editora Marcombo, 1979. 457p. Barcelona, Espanha.

DANIEL, Eduardo. Novas ligas de alumínio para linhas de transmissão e distribuição. Revista mundo Elétrico, março 1985, p. 43-47.

ELETRICIDADE MODERNA – Isoladores Sintéticos. Março de 1988, p.9-15. São Paulo, SP.

FREEMAN, P. J. Electric power transmission and distribution. George G. Harrap & Co. Ltd, London, 1968.

FUCHS, Rubens Dario. Transmissão de energia elétrica, v.1. LTC/EFEI, 1977.

GOPUR, R. S.; SHAFFNER, D.; CLARK, W.; VINSON, R.; RUFF, D. Utilities share their insulator field experience. Transmission and distribution world, April 2005, p. 17-27, USA.

HAVARD, D. G.; PON, C. J.; WATT, G. S. Clearances for galloping on overhead lines. Ontario Hydro, Canadá, 1989.

INDULKAR, C. S. Electrical energy systems engineering. Khanna Publishers, Delhi, Índia, 1976.

LIMA, Osveraldo; BASTO, Oscar Teixeira. Isoladores sintéticos: A experiência da CHESF. Revista Eletricidade Moderna, mar. 1988, p. 16-25.

NOGUEIRA, Murilo M; AZZAM, Georges; SOUZA, Flávio L. A. Carregamento de linhas de transmissão aéreas: dos critérios determinísticos à monitoração em tempo real”. GLT/005, XVI SNPTEE, 2001, Campinas, SP.

OLIVEIRA, Agamenon. Revisão das metodologias usadas na previsão de vibrações com linhas de transmissão. X SNPTEE, Curitiba, 1989.

ROWLINS, Charles B. Vibration of ice-coated overhead ground wires. Alcoa Conductor Products, 1989.

STEVENSON JR., W. D. Elements of the power systems analysis. McGraw Hill Book Company, 1962, USA.

STARR, A. T. Generation, transmission and utilization of electrical power. Pitman and Sons Ltda., 1969, London