



UNIVERSIDADE ESTADUAL DA PARAÍBA - UEPB  
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS – CCEA  
CAMPUS VII – GOVERNADOR ANTÔNIO MARIZ

TALES AUGUSTO CARVALHO DE BARROS

**IMPLEMENTAÇÃO COMPUTACIONAL DE  
DISPOSITIVOS ELÉTRICOS SELETIVOS EM BAIXA  
FREQUÊNCIA: UTILIZANDO A LINGUAGEM C/C++, NA  
SIMULAÇÃO DE FILTROS CL/LC EM BAIXA FREQUÊNCIA**

**PATOS-PB  
2017**

TALES AUGUSTO CARVALHO DE BARROS

IMPLEMENTAÇÃO COMPUTACIONAL DE DISPOSITIVOS  
ELÉTRICOS SELETIVOS EM BAIXA FREQUÊNCIA

Trabalho de conclusão de curso de  
Graduação submetida à Coordenação  
de Licenciatura Plena em Física da  
Universidade Estadual da Paraíba, com  
o requisito necessário à obtenção do  
título de Licenciado em Física.

Orientador: Prof. Dr. PEDRO CARLOS DE ASSIS JUNIOR

**PATOS-PB**  
**2017**

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

B277i Barros, Tales Augusto Carvalho de.  
Implementação computacional de dispositivos elétricos seletivos em baixa frequência [manuscrito] : utilizando a Linguagem C/C++, na simulação de filtros CL/LC em baixa frequência / Tales Augusto Carvalho de Barros. - 2017.  
87 p. : il. colorido.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Física) - Universidade Estadual da Paraíba, Centro de Ciências Exatas e Sociais Aplicadas, 2017.

"Orientação : Prof. Dr. Pedro Carlos de Assis Junior, Coordenação do Curso de Ciências Exatas - CCEA."

1. Filtro elétrico. 2. Linguagem computacional. 3. Linguagem computacional C++. 4. Circuito CL/LC.

21. ed. CDD 005

TALES AUGUSTO CARVALHO DE BARROS

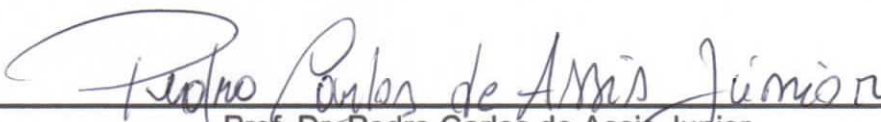
IMPLEMENTAÇÃO COMPUTACIONAL DE DISPOSITIVOS ELÉTRICOS  
SELETIVOS EM BAIXA FREQUÊNCIAS

Trabalho de conclusão de curso de  
Graduação submetida à Coordenação  
de Licenciatura Plena em Física da  
Universidade Estadual da Paraíba, com  
o requisito necessário à obtenção do  
título de Licenciado em Física.

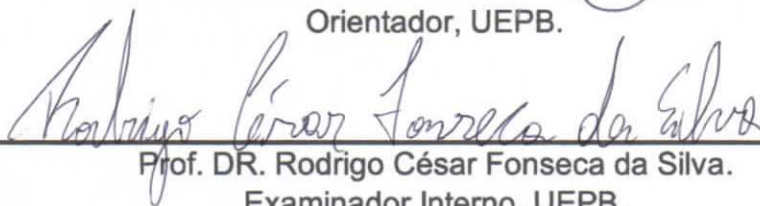
Área de concentração: Educação.

Aprovada em: 06/12/2017

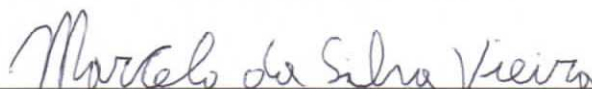
BANCA EXAMINADORA



Prof. Dr. Pedro Carlos de Assis Junior.  
Orientador, UEPB.



Prof. DR. Rodrigo César Fonseca da Silva.  
Examinador Interno, UEPB.



Prof. Dr. Marcelo Vieira Silva.  
Examinador Interno, UEPB.

PATOS  
2017



## **AGRADECIMENTOS**

Agradeço primeiramente a Deus por ter concluído este importantíssimo curso de Licenciatura Plena em Física, “O esforço sem um motivo torna inútil, a sabedoria sem humildade torna falha, a vida sem um princípio não faz sentido.”

Dedico ao professor e orientador Dr. Pedro Carlos de Assis Jr, meus agradecimentos, a sua dedicação e idoneidade no ato de generosidade de ensinar e orientar este trabalho.

Minhas Homenagens ao professor e co-orientador Dr. Marcelo Vieira Silva, com uma didática simples e funcional tornou possível a conclusão deste trabalho.

Agradeço ao professor e Dr. Rodrigo C. F. da Silva, pela sua dedicação e perseverança na pesquisa e também como co-orientador no aprendizado da linguagem computacional.

Agradeço sinceramente ao professor e Dr. Elder Eldervitch, o qual me apresentou o grupo de eletromagnetismo matemática e computação aplicada (GEMCA).

Estes professores foram os quais me ajudaram nesta jornada tanto na formação como profissional quanto na formação do meu caráter como profissional. Obrigado a minha família e aos meus amigos que me ajudaram a desenvolver dentre outras atividades extraclases, as quais foram necessárias para conclusão deste curso.

## RESUMO

Desde o surgimento do “ábaco” há 2000 anos a.c. considerado por alguns historiadores como sendo o primeiro computador, o homem utiliza-o para obter respostas e armazena-las, e em 1946 os norte-americanos John Presper Eckert e John W. constroem o primeiro computador eletrônico, o ENIAC (Electrical Numerical Integrator and Calculator), com uma capacidade de resolver 5000 equações por segundo. O computador foi usado na segunda guerra mundial para resolver cálculos balísticos. Após a segunda guerra, os computadores passaram a ser usados pelos cientistas nos mais diversos problemas, obtendo respostas numéricas e/ou complexas. As respostas numéricas são obtidas para intervalos de tempo muito curto. Diversos problemas tecnológicos foram interpretados, tais como: fenômenos quânticos e relativísticos. O método mais comum de utilizarmos os computadores como instrumento de pesquisa é introduzindo uma linguagem computacional que crie uma interface de interação entre a máquina e o fenômeno de interesse. Nesse trabalho utilizamos a linguagem C/C++, para a implementação de filtros elétricos, em baixa frequência, descrito no capítulo 10 do livro de Física Básica 3, do autor H. Moysés Nunssenzeveig. O livro não traz o comportamento gráfico desses filtros elétricos. Assim, criamos um código computacional capaz de gerar o comportamento do espectro de frequência  $\omega$ , dos filtros elétricos sugeridos pelo Moysés. A partir desse código computacional, geramos um banco de dados que permite construir gráficos do espectro de frequência. Com esses gráficos, a visualização dos fenômenos físicos para esses filtros elétricos ficaram visivelmente explicados. Os filtros elétricos investigados são circuitos do tipo: CL/LC (capacitor e indutor; indutor e capacitor), e estão em perfeito acordo as teorias apresentadas no livro, tornando assim o código implementado cabível e verídico para uso educacional. Consideramos este trabalho uma importante ferramenta para a compreensão de filtros elétricos, bem como, para futuros estudos nessa temática de ensino.

**Palavras chave:** filtros elétricos, linguagem computacional C++, circuitos CL/LC.

## ABSTRACT

Since the emergence of the "abacus" 2000 years ago a.c. considered by some historians as the first computer, man uses it to obtain answers and stores them, and in 1946 the Americans John Presper Eckert and John W. construct the first electronic computer, ENIAC (Electrical Numerical Integrator and Calculator), with a capacity to solve 5000 equations per second. The computer was used in World War II to solve ballistic calculations. After the second war, computers began to be used by scientists in various problems, obtaining numerical and / or complex answers. Numerical responses are obtained for very short time intervals. Several technological problems have been interpreted, such as: quantum and relativistic phenomena. The most common method of using computers as a research tool is to introduce a computer language that creates an interaction interface between the machine and the phenomenon of interest. In this work we use the C / C ++ language, for the implementation of low-frequency electric filters, described in chapter 10 of the book Basic Physics 3, author H. Moyses Nunssenzveig. The book does not bear the graphical behavior of these electric filters. Thus, we constructed a computational code capable of generating the **behavior of the frequency spectrum** of the electric filters suggested by Moysés. From this computational code, we generate a database that allows us to construct graphs of the frequency spectrum. With these graphs the visualization of the physical phenomena for these electric filters are visibly explained. The investigated electric filters are circuits of the type: CL / LC (capacitor and inductor, inductor and capacitor), and are in perfect agreement the theories presented in the book, thus making the implemented code correct and true for educational use. We consider this work an important tool for the understanding of electric filters, as well as for future studies in this subject of teaching.

**Keywords:** electric filters, C ++ computational language, CL / LC circuits.

## LISTA DE ILUSTRAÇÕES

Figura 1 - Exemplos de Filtros: a) filtros passa baixa, b) filtros passa alta, c) filtros passa banda e d) filtros bloqueia banda.....	5
Figura 2 - Exemplos de indutores com: a) Núcleo de ar, b) Núcleo de ferrite, c) Laminado, d) Toroidal, e) Ferromagnético .....	7
Figura 3 - Exemplos de capacitor: a) cerâmico, b) eletrolítico, c) óleo, d) poliéster, e) tântalo e f) variável .....	10
Figura 4 - Circuito simples.....	14
Figura 5 - a) Indutor ilustrado em um circuito e b) Ilustração de um capacitor .	15
Figura 6 - Associação em série de capacitores.....	16
Figura 7 - Associação em série de indutores .....	16
Figura 8 - Associação em paralelo de capacitores.....	17
Figura 9 - Associação em paralelo de indutores .....	18
Figura 10 - Associação mista de capacitores e indutores .....	19
Figura 11 - Interface do compilador Dev-C/C++.....	22
Figura 12 - Equações implementadas.....	23
Figura 13 - Tela gráfica a direita e a esquerda a tela visual c.....	24
Figura 14 - Filtro passa baixas frequências.....	26
Figura 15 - Filtro bloqueia baixas frequências.....	31
Figura 16 - Filtro Periódico infinito.....	35
Figura 17 - Redivisão de filtro infinito .....	51
Figura 18 - Substituição das impedâncias por uma total.....	52
Figura 19 - Filtro transmissor de baixas frequências.....	55
Figura 20 - Filtro transmissor de banda.....	65

## LISTA DE TABELAS

Tabela 1 - Múltiplos de indutância.....	9
Tabela 2 - Múltiplos de farad.....	13
Tabela 3 - Código do FPB, no código substituímos $\omega$ por $x$ , $\omega_0$ por $W_0$ e $\varepsilon$ por $E$ .....	29
Tabela 4 - Código FBB.....	33
Tabela 5 - Código OHP. No código substituímos as variáveis: $\beta$ por $B$ e $I_{0m}$ por $I_0$ , o sentido de OHP depende do sinal de $\beta$ .....	40
Tabela 6 - Código OA. Apenas mudando o sinal de $\alpha$ , o código mudara o sentido da onda.....	45
Tabela 7 - Código OAFI, substituímos os sinais das raízes: $-P^\alpha$ para $-P^{-\alpha}$ , obtemos respostas diferentes.....	49
Tabela 8 - Código FI, substituímos no código as variáveis: $\bar{Z}_T$ por $ZT$ , $\bar{Z}_1$ por $Z1$ e $\bar{Z}_2$ por $Z2$ .....	53
Tabela 9 - Código que gera o gráfico da defasagem $\beta$ em função de $\omega$ . Variáveis substituídas no código foram: $\omega_0$ por $W_0$ e $\beta$ por $B$ .....	59
Tabela 10 - Código que gera a constante de atenuação $\alpha$ em função de $\omega$ , no código substituímos $\alpha$ por $A$ .....	61
Tabela 11 - Código que gera a parte real da impedância total $\bar{Z}_T$ em função de $\omega$ , no código substituímos $\bar{Z}_T$ por $ZT$ .....	63
Tabela 12 – Código da função de defasagem $\beta$ para as condições de $\omega_1$ e $\omega_2$ , substituímos no código $\omega_1$ por $W1$ e $\omega_2$ por $W2$ .....	67
Tabela 13 - Código gera a constante de atenuação $\alpha$ , para a condição $\omega_1 \ll \omega$ .....	69
Tabela 14 - Código $\omega \gg \omega_2$ em função de $\alpha$ .....	71

## LISTA DE REDUÇÕES

$\epsilon_0$  – Permissividade elétrica.

$Z_C$  – Impedância capacitiva.

$Z_L$  – Impedância indutiva

$I$  – Corrente elétrica.

$R$  – Resistência.

$\omega_0$  – Frequência oscilação livre/corte.

$\omega$  – Frequência de oscilação.

$\mu_0$  - Permeabilidade magnética no espaço.

$\sigma$  – Condutividade do material.

$f$  – Frequência.

$f_p$  – Frequência passante.

$f_b$  – Frequência bloqueada.

$\lambda$  – Comprimento de onda.

$c$  – Velocidade da luz.

$\epsilon$  – Força Eletromotriz (fem).

$V$  – Tensão.

$C$  – Capacitância.

$L$  – Indutância.

SI – Sistema Internacional de Unidades.

$\bar{Z}_T$  – Impedância terminal.

OA – Onda Atenuando.

FPB – Filtro Passa Baixa.

FBB – Filtro Bloqueia Baixa.

OHP – Onda Harmônica Progressiva.

OAFI – Onda Atenuando com Fase invertida.

FI – Filtro Iterativo.

## LISTA DE PUBLICAÇÕES

OLIVEIRA, E. E. C. D.; BEZERRA, R. R. M.; **BARROS, T. A. C.**; SILVA, R. C. F.; MEDEIROS, T. C., ANTENA PATCH BIOINSPIRADA EM UM TREVO DE TRÊS FOLHAS PARA A FAIXA DE FREQUÊNCIA DE COMUNICAÇÃO SEM FIO. Revista Científica Semana Acadêmica, v.01 - 2017, p. <http://semanaac> -, 2017.

OLIVEIRA, E. E. C.; SILVA, R. C. F.; **BARROS, T. A. C.** GERENCIAMENTO DE UMA REDE DE COMPUTADORES EM UM AMBIENTE CORPORATIVO UTILIZANDO O SOFTWARE ZABBIX. Revista Científica Semana Acadêmica, v.01 - 2017, p. <http://semanaac> -, 2017.

SILVA, R. C. F.; OLIVEIRA, E. E. C.; **BARROS, T. A. C.**; ASSIS JUNIOR, P. C.; VIEIRA, M. S. ANÁLISE NUMÉRICA DE UMA ONDA ELETROMAGNÉTICA EM MEIOS DIELETRICOS MÓVEIS USANDO O MÉTODO FDTD. Revista Científica Semana Acadêmica, v.01 - 2016, p. <http://semanaac> -, 2016.

OLIVEIRA, E. E. C.; SILVA, R. C. F.; ASSIS JUNIOR, P. C.; **BARROS, T. A. C.**; OLIVEIRA, R. A. P. PROJETO DE UM FILTRO DE MICROFITA COM DEFORMAÇÃO CIRCULAR. Revista Científica Semana Acadêmica, v.01, p.1 - 12 - 1, 2016.

OLIVEIRA, E. E. C.; SILVA, R. C. F.; **BARROS, T. A. C.**; ASSIS JUNIOR, P. C. PROJETO E ANÁLISE DE UM FILTRO DE MICROFITA COM LINHAS ACOPLADAS E COM DEFORMAÇÕES PERIÓDICAS (DGS) NO PLANO DE TERRA. Revista Científica Semana Acadêmica, v.01 - 2016, p. <http://semanaac> -, 2016.



## SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	1
1.1 Organização do documento .....	3
<b>2. MARCO TEÓRICO DOS FILTROS ELÉTRICOS</b> .....	4
2.1 Filtros elétricos .....	4
2.2 Surgimentos dos filtros elétricos e atuais .....	5
2.3 Definições dos elementos passivos C e L .....	6
2.3.1 Classificações dos indutores .....	6
2.3.2 Classificações dos capacitores .....	9
<b>3. CARACTERIZAÇÃO DO OBJETO DE ESTUDO</b> .....	14
3.1 Associações em série .....	15
3.2 Associações em paralelo .....	17
3.3 Associação mista .....	18
<b>4. METODOLOGIA</b> .....	21
4.1 Estudo bibliográfico .....	21
4.2 Procedimento computacional .....	21
4.3 Veracidade computacional .....	24
4.4 Contribuição científica .....	25
<b>5. RESULTADOS COMPUTACIONAIS PARA OS FILTROS LC E CL</b> .....	26
5.1 Filtros LC .....	26
5.2 Filtros CL .....	31
5.3 Circuitos em redes periódicas .....	35
5.3.1 Verificando o caso (i): .....	43
5.3.2 Verificando o caso (ii): .....	47
5.4 Impedância interativa .....	51
5.4.1 Filtro transmissor de baixa frequência .....	55
5.4.2 Filtro transmissor de banda .....	65
5.5 Aplicações dos filtros .....	71
<b>6. CONCLUSÃO</b> .....	73
<b>BIBLIOGRAFIA</b> .....	74
<b>APÊNDICE</b> .....	76
Tutorial C++ .....	76

## 1. INTRODUÇÃO

A palavra denominada dos filtros, com base na literatura, são dispositivos responsáveis por separar, bloquear ou filtrar determinados objetos, partículas sedimentos, ondas eletromagnéticas, dentre outros. Atribuindo-lhe essa característica de filtrar, podemos imaginar que nessa configuração seja possível a passagem ou bloqueio de algo. Conhecemos diversos processos de filtragem utilizados no cotidiano, tais como: a peneira de areia, o protetor solar, óculos escuros (extremamente comum), o famoso filtro de água (feito de barro). Todos esses exemplos de filtros conseguem de alguma forma separar ou bloquear coisas do nosso interesse, seja a benefício da saúde ou apenas para proporcionar melhor performance na construção civil, dispositivos eletrônicos, etc. Existem infinitos produtos com essa característica de filtrar.

Literalmente estes dispositivos elétricos são formados por um conjunto de elementos passivos que interagem com a corrente elétrica. Os denominados elementos passivos na eletrônica são conhecidos como capacitor ou indutor, e suas associações em série e em paralelos também representam um elemento passivo. Esses componentes eletrônicos são os que mais compõem os circuitos elétricos atualmente.

Para compreender o funcionamento de filtros seletores em frequência, é necessário ter conhecimento de circuitos elétricos construídos a partir dos capacitores e indutores. Um circuito elétrico pode conter diversos componentes eletrônicos, tais como: resistores, transistores, amplificadores, diodos, transformadores, alimentados por um gerador de tensão. A forma de associação entre os componentes eletrônicos é determinante para definir a classificação do filtro eletrônico.

Os filtros construídos a partir de dispositivos eletrônicos apresentam comportamento de filtragem em baixas frequências, na ordem de mega-hertz. Como exemplos citamos: a faixa de transmissão do rádio. Outros exemplos cotidianos são as TVs digitais ou Rádios digitais, existem para este tipo de aplicação em materiais feitos de microfita, que dispensa componentes eletrônicas, o qual interagem com sinal de comprimento de onda muito pequeno,

e conseqüentemente os caracterizamos como sendo de altas frequências, da ordem de giga-hertz.

Os filtros podem ser classificados em quatro seguimentos: filtro bloqueia frequência, passa frequência, bloqueia bandas de frequências e passa bandas de frequências. No segmento bloqueia frequência aplica-se duas derivações, que ainda pode ser classificada como sendo: bloqueia baixa frequência ou bloqueia alta frequência.

Neste trabalho acadêmico apresentamos algumas aplicações simples associados a filtros eletrônicos, utilizando a linguagem de computação C++, e implementando pequenas técnicas dessa linguagem nos estudos de filtros eletrônicos, dos quais são capazes de selecionar sinais elétricos em determinadas frequências. Esses filtros são mais conhecidos na literatura por: filtros elétricos seletores de frequências.

Observamos que o autor H. Moysés Nunssenzveig, em seu livro de Física Básica, volume 3, mais especificamente no capítulo 10, apresenta um pequeno estudo teórico em filtros de baixa frequência, onde investiga o comportamento de filtragem de forma teórica, sem ilustrações gráficas do espectro de frequência.

Buscamos métodos computacionais, o qual implementamos em códigos no compilador gratuito DevC/C++ as funções que caracterizam os filtros seletores em baixa frequência, previamente apresentados no capítulo 10 do Moysés. Assim, nos concentramos em gerar gráficos ilustrativos do espectro de frequência e conseqüentemente associamos a compreensão das regiões de passagem ou bloqueio da frequência. Comparamos nossos gráficos com as expressões teóricas apresentadas por Moysés e observamos se estavam de acordo com as teorias apresentadas.

Atualmente, os filtros possuem grande importância nas investigações científicas. Como exemplo de pesquisas recentes cito: o filtro de íons ou seletores de velocidades, formado por um cilindro e duas placas internas com diferença de potencial onde os campos elétricos e magnéticos selecionam um feixe de íons que atravessam o cilindro, através das velocidades das partículas. Como notamos, os filtros também apresentam aplicações sofisticadas, desde ruídos sonoros em aparelhos amplificadores, como utilizados na medicina ao

colete de chumbo, que impede danificar células humanas no processo de radiação conhecido como raio X. Dentre infinitas aplicações, os filtros são uteis para ajudar a sociedade em geral.

## 1.1 Organização do documento

Este trabalho está organizado em 5 capítulos, incorporando os motivos éticos e responsáveis de caráter motivacional envolvida até a conclusão com os resultados investigados e explorados, dispostos da seguinte forma:

O **capítulo 1** apresenta uma introdução geral do trabalho que nos propusemos a fazer, como trabalho de conclusão de curso.

O **capítulo 2** aborda as definições que caracterizam a importância dos filtros elétricos desde o seu surgimento, e ilustram os elementos passivos que contribuíram para a construção dos filtros elétricos.

O **capítulo 3** apresenta as bases teóricas para análise do projeto de um filtro elétrico simulado a partir de um circuito simples, e seus tipos de associações: em série, em paralela e mista. E as equações dos filtros elétricos obtidas a partir das leis de Kirchhoff.

O **capítulo 4** descreve os procedimentos metodológicos utilizados para investigar as expressões teóricas que descrevem os filtros, os quais foram definidos por Moisés Nunssenzveig no seu livro, titulado como curso de física básica 3.

O **capítulo 5** traz os resultados das simulações dos filtros elétricos seletores em baixa frequência descritos por Moisés, com implementação dos devidos códigos em linguagem de programação C/C++ esboçados em tabelas.

## 2. MARCO TEÓRICO DOS FILTROS ELÉTRICOS

### 2.1 Filtros elétricos

Os filtros elétricos são circuitos elétricos construídos a partir de elementos simples, tais como: indutor, capacitor e resistor. A ideia fundamental desses filtros é bloquear ou permitir a passagem de determinadas correntes em baixa frequência. Com o surgimento do rádio, ou seja, transmissão de informação através de ondas eletromagnéticas, surgiram problemas em que as antenas não conseguiam transmitir ondas no espectro da frequência da voz, ou seja, em baixas frequências, da ordem do espectro de 50 KHz. Para resolver este problema foi utilizado um filtro, seguido do efeito de modulação. De acordo com comprimento de onda de De Broglie:

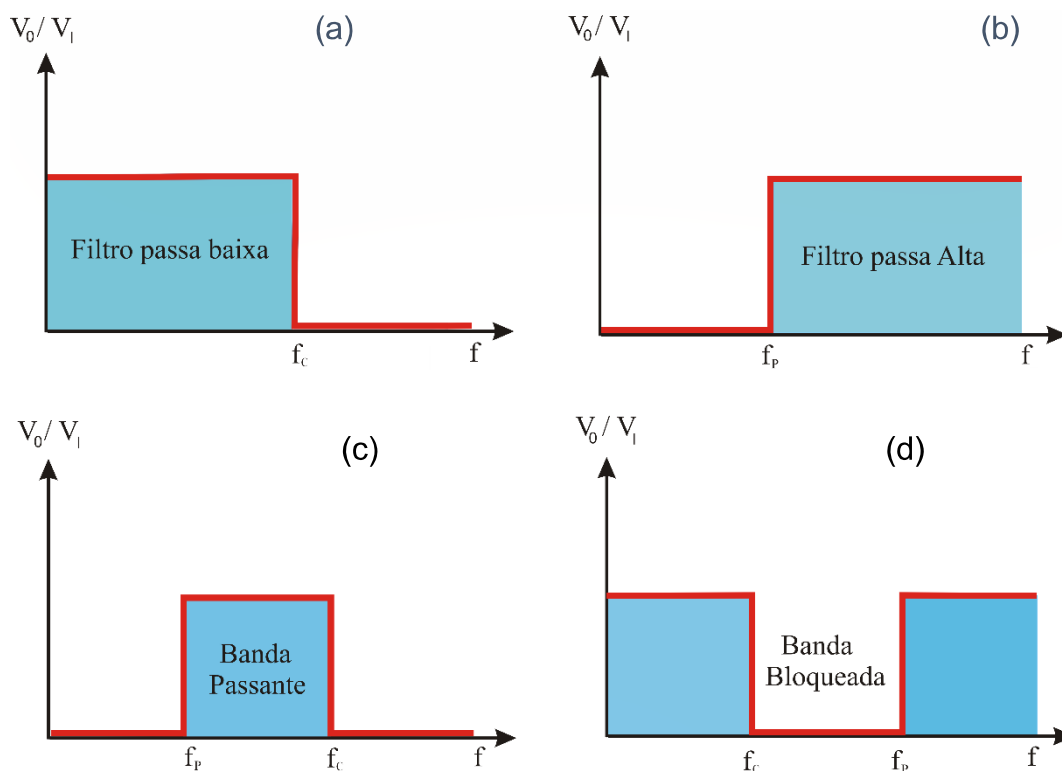
$$\lambda = \frac{c}{f}. \quad (2.1)$$

Observe a equação (2.1), que é impossível projetar antenas com dimensões do comprimento de onda para frequência muito baixa como a do espectro da frequência da voz, assim parecia não ser possível que a voz pudesse ser transmitida. Contudo, foi preciso compreender o fenômeno de modulação.

O fenômeno da modulação tem como característica transmitir um sinal de frequência dentro de outro. Assim, o sinal da voz foi transmitida dentro de uma frequência alta como por exemplo: dentro de uma frequência de 1 GHz. Então é nesse ponto em que os filtros entram na história, quando a antena do rádio recebe o sinal transmitido com as duas frequências moduladas, o filtro receberá o pulso da antena e bloqueará o sinal com a frequência de 1 GHz e permitirá apenas a passagem da frequência baixa, ou seja, o rádio transmitirá apenas a faixa da frequência da voz eliminando o ruído do 1 GHz.

Os filtros possuem algumas classificações conforme Dias (p.100, 1999), ilustradas a seguir, na Figura 1.

Figura 1 - Exemplos de filtros: a) filtros passa baixa, b) filtros passa alta, c) filtros passa banda e d) filtros bloqueia banda



Fonte: Própria(2017)

A razão da diferença de potência  $V_0/V_1$  mostrada nas Figuras 1(a), 1(b), 1(c) e 1(d) são as respostas para cada tipo de filtro. Onde  $f_p$  é a frequência que o filtro permite a passagem da determinada corrente, e  $f_c$  é a frequência de corte que bloqueia o sinal da corrente para qualquer espectro de frequência. Nos exemplos c) e d) da Figura 1, observamos as bandas selecionadas pela diferença das frequências  $f_p$  e  $f_c$ , denominamos esta diferença, como sendo uma faixa, a qual tem como função de permitir a passagem de sinais dos espectros de frequência dentro dela, este caso ocorre no exemplo (c), ou de rejeição no caso do exemplo (d).

## 2.2 Surgimentos dos filtros elétricos e atuais

Os primeiros filtros elétricos segundo Moysés (1997, p. 226), “foram construídos por Campbell 1906”. Eram e ainda são constituídos com elementos

passivos e ativos e tinham as propriedades de impedir a corrente elétrica de fluir no circuito. Essa característica foi muito importante para resolver problemas de transmissão de sinais de rádio, como exemplo o efeito da modulação mencionado na seção 2.1. Atualmente os filtros encontram-se mais modernos a exemplos dos filtros analógicos e digitais, encontrados em qualquer loja de eletrônica.

## **2.3 Definições dos elementos passivos C e L**

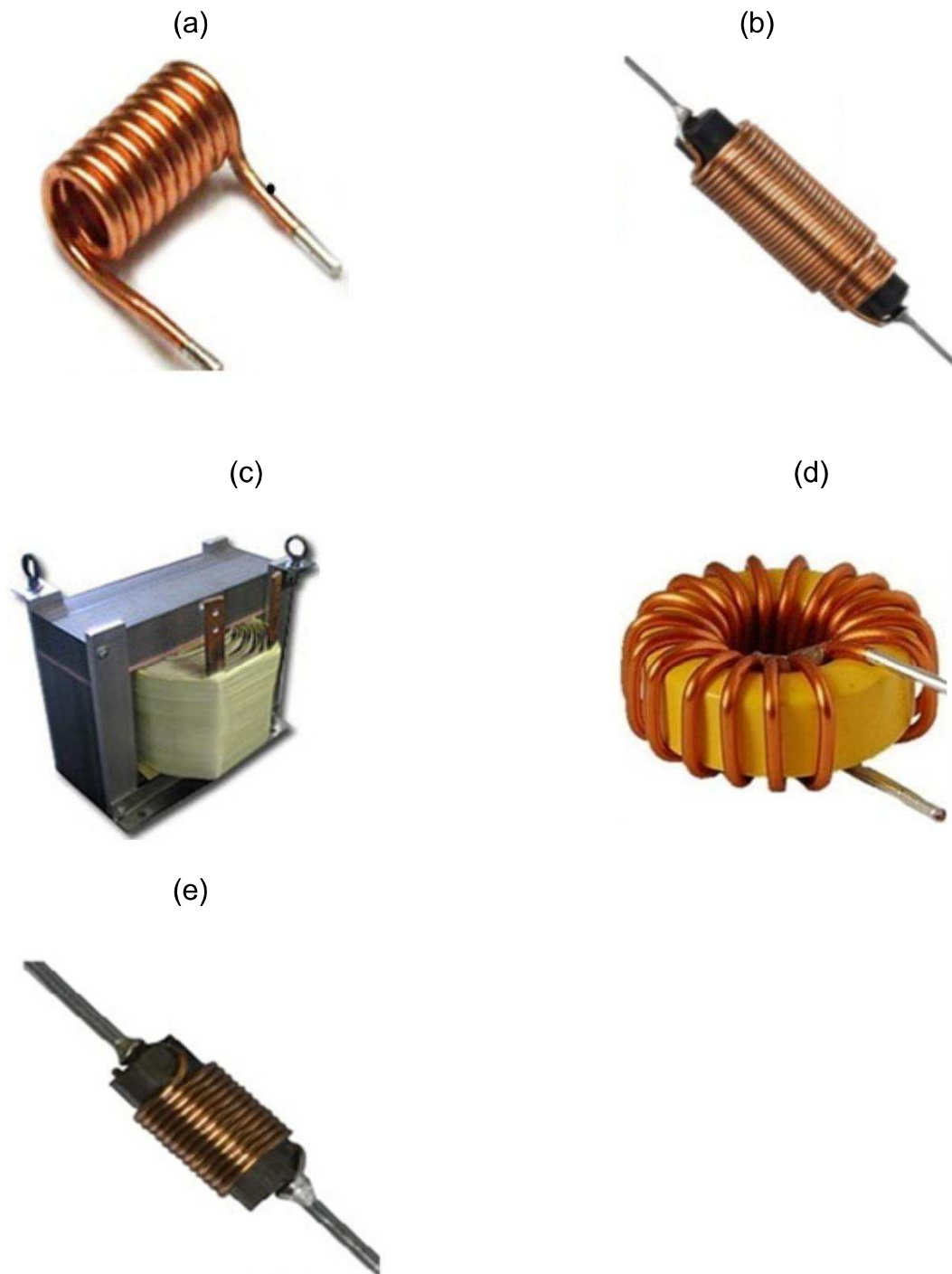
As definições de alguns capacitores e indutores desde as suas classificações as suas características, são importantes para construir o circuito elétrico, sendo estas características que definirão em que aplicação este circuito elétrico terá como filtro. Portanto apresentaremos a seguir as características e tipos dos elementos utilizados neste trabalho os quais definirão o nosso filtro elétrico.

### **2.3.1 Classificações dos indutores**

Indutores são conhecidos também como solenoides ou bobinas, sua constituição é basicamente de fio condutor enrolado em torno de algo, ou apenas o centro vazio, por ser um elemento passivo, este dispositivo tem a capacidade de armazenar energia, sendo esta, armazenada nas linhas de campo magnético no centro do solenoide, mas esse efeito só é possível se a fonte que alimenta o indutor fornecer uma corrente alternada (CA).

O efeito que ocorre no indutor é conhecido como indutância. Sendo este efeito uma grandeza física representada pela letra L e sua unidade de medida no SI é o Henry (H). Grandeza que está relacionada com a tensão induzida no campo magnético e a corrente responsável pelo surgimento deste campo, e a tensão nos terminais do indutor é proporcional a corrente que nela passa. A energia armazenada no indutor é medida no SI em joules (J), e também pode ser expressa pela quantidade de trabalho necessário para estabelecer o fluxo magnético no indutor, ou seja, o campo magnético. Segue na próxima página alguns tipos de indutores:

Figura 2 - Exemplos de indutores: a) núcleo de ar, b) núcleo de ferrite, c) laminado, d) toroidal, e) ferromagnético



Fonte: Mundo da elétrica (2017)

Os materiais usados nos modelos de indutores Figura 2, são responsáveis do desempenho e eficiência de cada um, e caracterizam-se:



**Indutor núcleo de ar:** este tipo de indutor é diferente dos demais, pois não utiliza ferromagnético em seu interior, mas mesmo assim, possui perdas baixas resultando em uma alta frequência, sendo sua baixa indutância utilizada para transmissão em altas frequências.

**Indutor núcleo de ferrite:** caracterizado por ser constituído por cerâmica ferromagnética. Este indutor tem boa eficiência em altas frequências, pois elimina correntes parasitas e a histerese é muito baixa.

**Indutor núcleo laminado:** este indutor é formado por um conjunto de matérias que caracterizam como o mais específico dentre os já citados, pois o seu núcleo é composto por laminas de aço-silício, envolvidos por verniz. O verniz ao envolver as laminas previne perdas por correntes parasitas. E o silício adicionado ao aço diminui a histerese.

**Bobinas toroidais:** indutor caracterizado pelo baixo custo de fabricação e pela orientação circular dos solenoides. Este indutor possui um núcleo formado por ferrite com um formato de rosca. Este formato não foi escolhido por acaso, pois as linhas de campo magnético percorri o caminho circular do núcleo, esta técnica foi desenvolvida para evitar perdas.

**Indutor núcleo ferromagnético:** o núcleo feito de ferromagnético do indutor possibilita uma indutância maior que os aqui já citados, mas as perdas para histerese são significativas, isso é, devido a maior concentração de campos magnéticos em seu interior.

Para obter a capacidade de indução do indutor núcleo de ar, utilizamos a equação (2.2).

$$L = \frac{(A^2 \times N^2)}{22,4 \times (9a + 10\ell)} \quad (2.2)$$

Sendo  $L$  a indutância e é mais utilizada em micro henry ( $\mu\text{H}$ ),  $a$  é o raio da bobina, medida em milímetros (mm),  $\ell$  é o comprimento da bobina, também em milímetros,  $A$  é a área da espira calculada em milímetros,  $N$  é designada ao número de espiras da bobina, número puro. Esta equação é a mais simples para

um indutor cujo núcleo é vazio, mas para um caso mais geral em que usa-se indutores com núcleo preenchido com determinado material, utiliza-se a equação (2.3).

$$L = \frac{(1,26 \times N^2 \times \mu_0 \times A)}{10^7 \times \ell} \quad (2.3)$$

Observe que, tanto o indutor de um núcleo vazio, como de um totalmente preenchido, dependem da área das espiras e do número de voltas, mas o que influencia estes indutores serem melhores está na permeabilidade magnética do material  $\mu_0$ . A Tabela 1, mostra algumas das unidades mais utilizadas para indutores:

Tabela 1 - Múltiplos de indutância

Indutância	Ordem G.	Unidade (SI)
1H	1	Henry
1mH	$\times 10^{-3}$	Mili Henry
1 $\mu$ H	$\times 10^{-6}$	Micro Henry
1nH	$\times 10^{-9}$	Nano Henry
1pH	$\times 10^{-12}$	Pico Henry

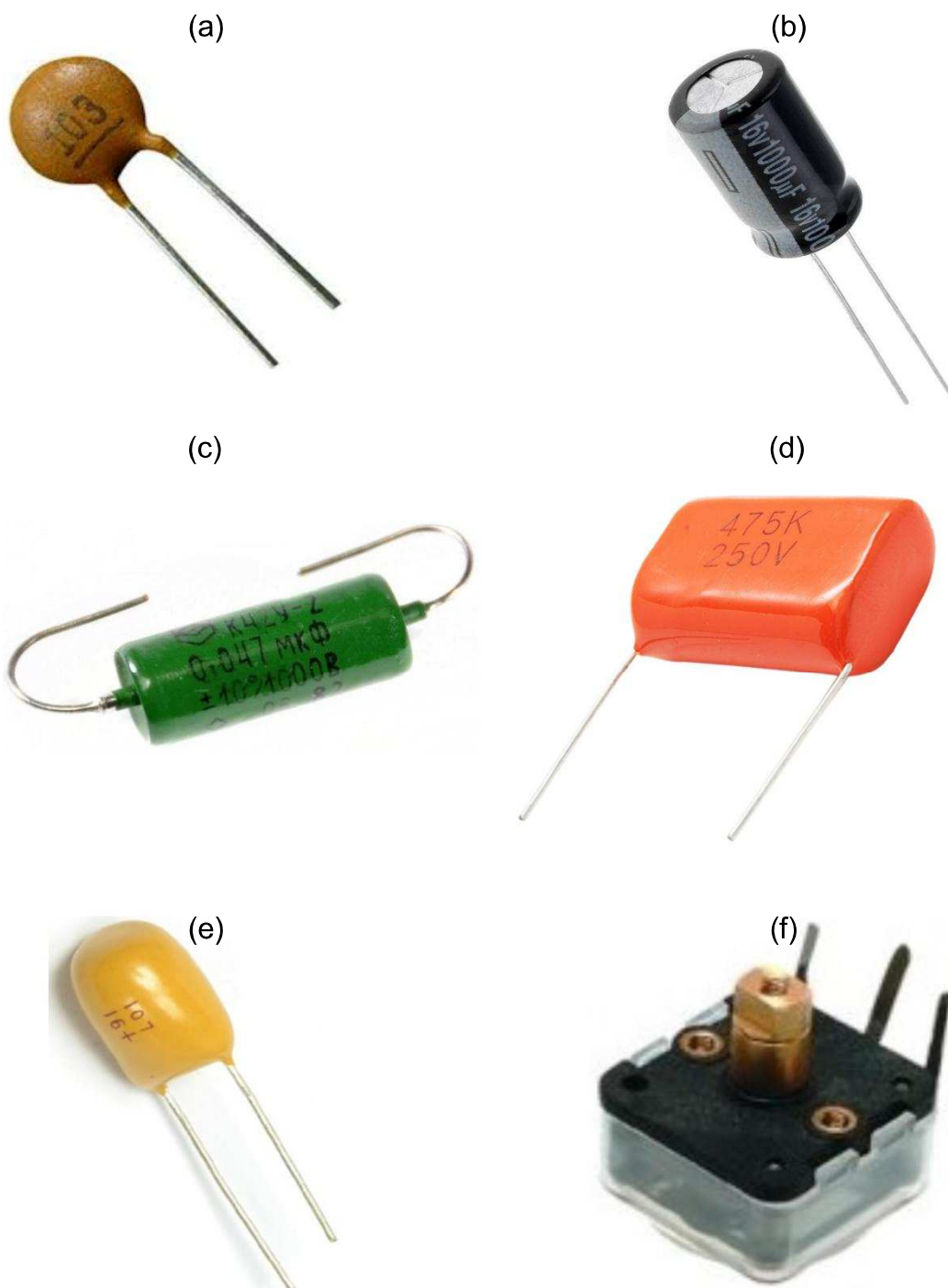
Fonte: Própria(2017)

### 2.3.2 Classificações dos capacitores

Os capacitores são dispositivos elétricos utilizados em vários circuitos eletrônicos, pois sua característica de armazenar energia o faz importante na transmissão de energia. Sua estrutura é basicamente formada por duas placas condutoras em paralelo e separadas por um distância muito próxima uma da outra, mas nunca em contato, pois é justamente esta distância entre as placas, que cria um campo elétrico que é o responsável pelo armazenamento da energia, para que o contato não aconteça, um material resistente ao fluxo de corrente é adicionado entre as placas e o mais conhecido é um substrato dielétrico, e por que o ar não é viável? Por que mesmo o ar pode se tornar um condutor para tensões elevadas, ou seja, a diferença de potencial(DDP) entre as placas forem

significativamente diferentes. Existem vários tipos de capacitores. Ilustraremos alguns na Figura 3.

Figura 3 - Exemplos de capacitor: a) cerâmico, b) eletrolítico, c) óleo, d) p  
e) tântalo e f) variável



Fonte: Mundo da elétrica (2017)

Os capacitores da Figura 3, são os mais encontrados em lojas de eletrônicos. A seguir, faremos um pequeno estudos das especificações características associadas aos materiais que os compõem.

**O capacitor cerâmico:** o capacitor cerâmico é o mais comum de se encontrar em equipamentos eletrônicos. Por exemplo: em rádios, em televisões, dentre outros. Este dispositivo e constituído de forma simples sendo composto por um pó de cerâmica comprimida, formando uma pastilha que será depositado entre duas placas, que são chamadas de armaduras. Um processo chamado de vaporização de prata, é responsável para combinar o substrato (pó de cerâmica comprimida) entre as placas e proteger o capacitor. Este dispositivo fornece uma capacidade de operar circuitos de altas frequências e também podem ser utilizados em circuitos de corrente contínua.

**Capacitor eletrolítico:** este capacitor, assim como os demais, também tem duas placas condutores de alumínio. Contudo, ele difere do cerâmico, pois tem uma camada fina de óxido de alumínio entre as placas, que são enroladas e embebidas em um eletrolítico líquido, de onde surge o seu nome. Esses capacitores conseguem tornar a fonte mais estável e até filtra ruídos de corrente.

**Capacitor óleo:** capacitor óleo, uns dos primeiros a serem feitos e utilizados para armazenamento de energia. Eles não são usados atualmente, mas teve importante função em equipamentos que possuíam válvulas. Esses capacitores eram feitos com duas fitas de alumínio enrolados em papel e embebidos em óleo ou em outro líquido um pouco mais denso.

**Capacitores poliéster:** este dispositivo é o mais compacto aqui citado, sendo formado por várias fitas de alumínio entre camadas de poliéster, a capacitância é determinada pela quantidade de fitas e espessura do poliéster. Possibilitando maior armazenamento de carga que os demais.

**Capacitor tântalo:** o capacitor tântalo tem uma característica, que está relacionada ao terminal positivo ser maior que o negativo, a finalidade é para ser conectado com fontes de correntes contínuas, este dispositivo não pode ser conectado em circuito de correntes alternadas. Este dispositivo fornece alta durabilidade em comparação com outros capacitores, e diminui a perda de corrente de fuga.

**Capacitor variável:** o capacitor variável é mais empregado como sintonizador de frequências em transmissão, seja no rádio, nas televisões e em outros equipamentos. Este capacitor funciona para baixas tensões e baixos potencias. Sua estrutura é formada por placas de latão ou alumínio e o seu interior geralmente é o ar.

Como mencionado, de modo geral, as características fundamentais de um capacitor está diretamente associada ao tipo do dielétrico. Esse é responsável pela qualidade do capacitor em armazenar energia. Desta forma, buscamos conhecer a capacitância de um capacitor. A expressão matemáticas mais simples, é dada por:

$$C = \frac{Q}{V}. \quad (2.4)$$

A equação (2.4) é para um capacitor com placas apenas separadas pelo ar. Para capacitores que utilizam matérias isolantes, entre as placas, utilizamos a expressão matemática, para a capacitância é dada pela equação (2.5)

$$C = \frac{k\epsilon_0 A}{d} \quad (2.5)$$

onde,  $k$  é a constante de coulomb, igual a  $9 \times 10^9 \text{ N} \cdot \frac{\text{m}^2}{\text{C}^2}$ ,  $\epsilon_0$  é a permissividade elétrica do meio material,  $d$  é a distância entre as placas e  $A$  é a área das placas.

Esta equação pode determinar a capacidade de armazenamento de carga ou energia dos capacitores. Na Tabela 2, ilustramos a ordem de grandeza das capacitâncias, medidas no (SI).

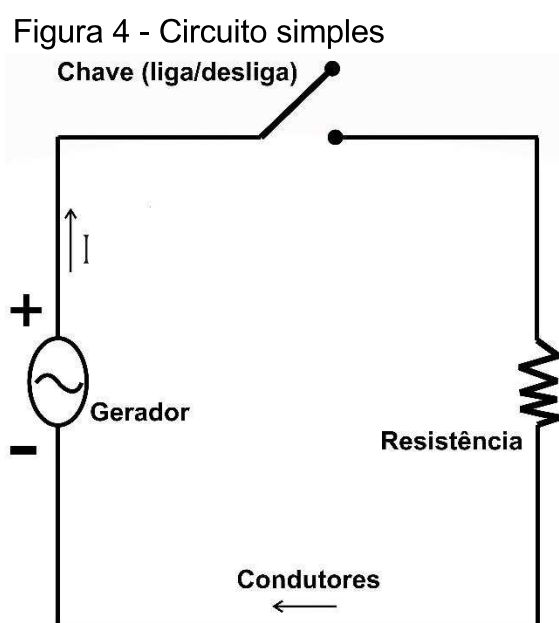
Tabela 2 - Múltiplos de farad

Capacitância	Ordem G.	Unidade (SI)
1F	1	Farad
1mF	$\times 10^{-3}$	Mili Farad
1 $\mu$ F	$\times 10^{-6}$	Micro Farad
1nF	$\times 10^{-9}$	Nano Farad
1pF	$\times 10^{-12}$	Pico Farad

Fonte: Própria(2017)

### 3 CARACTERIZAÇÃO DO OBJETO DE ESTUDO

Como definimos, os filtros elétricos são circuitos construídos a partir de componentes eletrônicas, interligadas no circuito, por condutores. Geralmente, esses condutores são fios de cobre. O circuito possui ainda, uma chave ligar/desligar e um gerador de energia (fonte), geralmente é uma tensão alternada. O gerador também pode ser um dispositivo de corrente contínua, tais como: pilhas ou baterias. Na Figura 4, ilustramos um circuito elétrico simples.

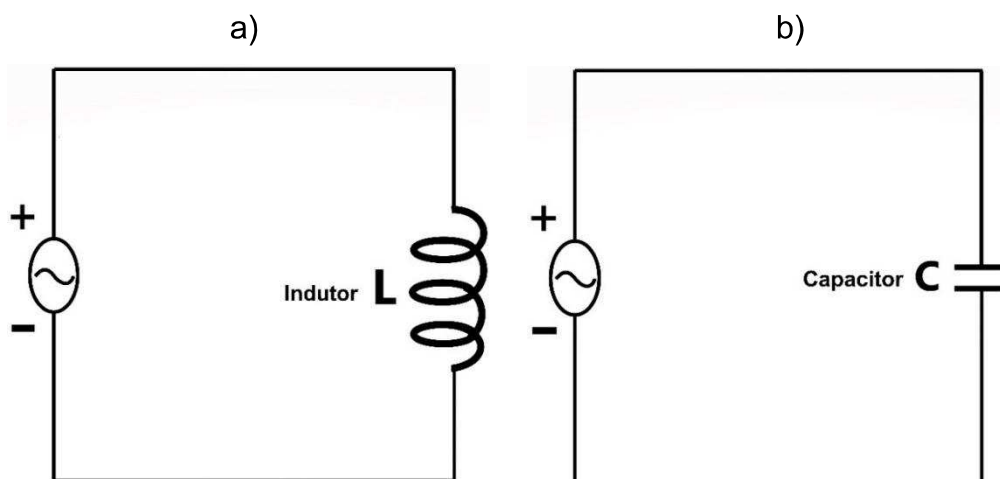


Fonte: Própria(2017)

A resistência mostrada na Figura 4, pode ser qualquer dispositivo que consome energia, por exemplo: uma lâmpada, uma geladeira, uma televisão, etc. Esses aparelhos domésticos constituem circuitos complexos, os quais possuem capacitores e indutores. A unidade de medida da resistência, no SI, é o Ohms, símbolo,  $\Omega$ . Existem também resistores com resistência variável.

Vamos substituir no circuito da Figura 4, a resistência por um capacitor ou por um indutor. Neste novo circuito, Figuras: 5a e 5b, interessamos nas propriedades associadas pela passagem da corrente nos elementos indutivos e capacitivos.

Figura 5 - a) Circuito indutivo e b) Circuito capacitivo



Fonte: Própria(2017)

O indutor e o capacitor quando submetidos a uma passagem de corrente alternada entre seus terminais, provocam efeitos: magnéticos e elétricos, os quais são respectivamente responsáveis pelos fenômenos indutivos e capacitivos do circuito. Como discutimos no capítulo 2.

### 3.1 Associações em série

Para capacitores associados em série, as ligações devem seguir a ordem, polo positivo conectado ao polo negativo do próximo capacitor. Observe a Figura 6.

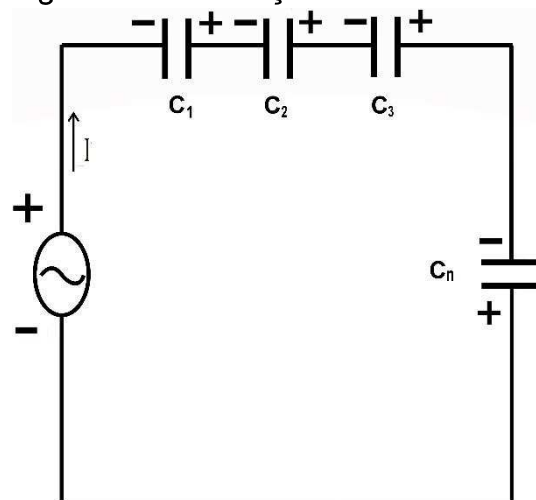
A capacitância equivalente  $C_{eq}$ , desse circuito é dado por:

$$\frac{1}{C_{eq}} = \frac{1}{C_1} + \frac{1}{C_2} + \frac{1}{C_3} + \dots + \frac{1}{C_n}. \quad (2.6)$$

Onde,  $C_{eq}$  será sempre menor do que o valor do menor capacitor do conjunto. Observamos que este tipo de associação é útil para trabalhar com menor tempo possível para carregar e descarregar um circuito.



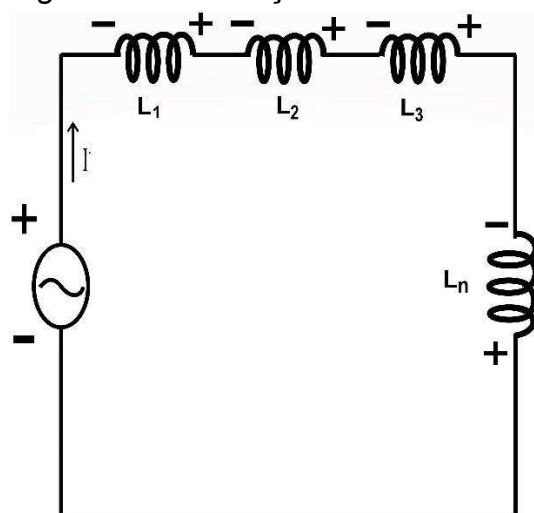
Figura 6 - Associação em série de capacitores



Fonte: Própria(2017)

As ligações em série dos indutores não são diferentes dos capacitores. Seus terminais são ligados da mesma forma que a associação em serie dos capacitores. Na Figura 7, ilustramos as ligações dos indutores.

Figura 7 - Associação em série de indutores



Fonte: Própria(2017)

Contudo, a diferença entre associações de capacitores e indutores em série está na expressão matemática, que para os indutores é dada por:

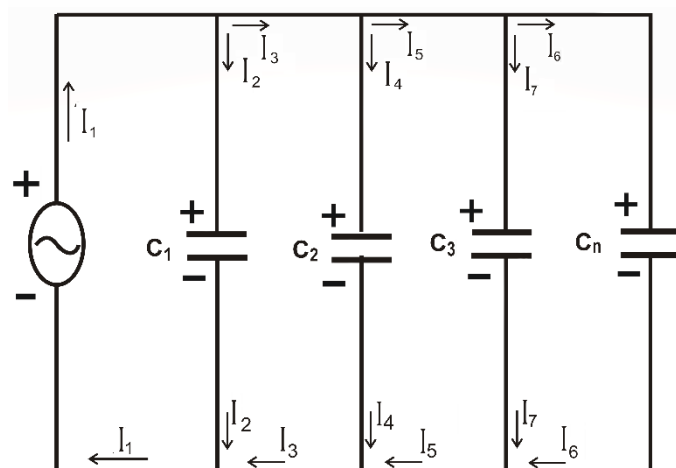
$$L_{eq} = L_1 + L_2 + L_3 + \dots + L_n. \quad (2.7)$$

A indutância equivalente,  $L_{eq}$ , dada pela equação (2.7) é a soma das indutâncias no circuito em série.

### 3.2 Associações em paralelo

A associação em paralelo para capacitores é ilustrada na Figura 8. Observamos a existência de nós no circuito. Um nó é identificado quando a corrente se divide. Na Figura 8, temos as correntes:  $I_2$  e  $I_3$ ,  $I_4$  e  $I_5$ , e  $I_6$  e  $I_7$ , onde correspondem as correntes em cada nó do circuito.

Figura 8 - Associação em paralelo de capacitores



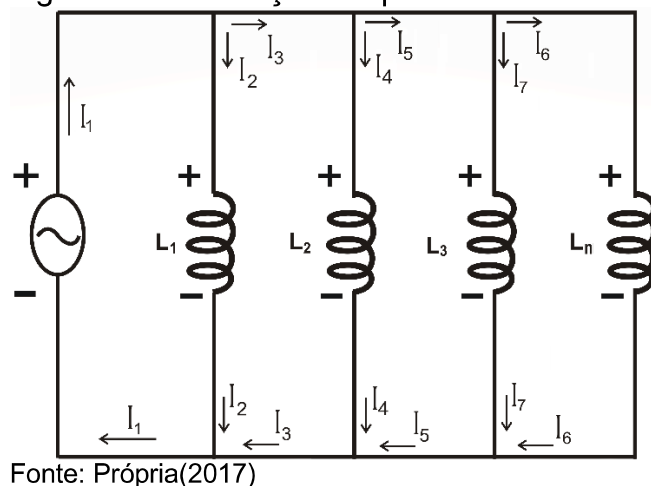
Fonte: Própria(2017)

A expressão matemática dada pela equação (2.8) representa a  $C_{eq}$  de um circuito associado em paralelo.

$$C_{eq} = C_1 + C_2 + C_3 + \dots + C_n. \quad (2.8)$$

A equação (2.8) é análogo a resistência equivalente de resistores associação em série. Na associação em paralela as capacitâncias do circuito somam-se, e conseqüentemente armazenando mais energia no circuito. A associação em paralelo dos indutores, é ilustrada pela Figura 9.

Figura 9 - Associação em paralelo de indutores



Neste caso, a indutância equivalente é dada pela equação (2.9)

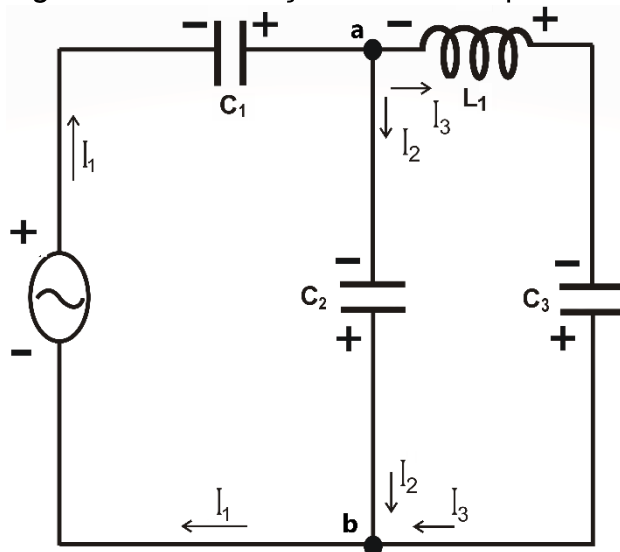
$$L_{eq} = \frac{L_1 L_2}{L_1 + L_2}. \quad (2.9)$$

Nessa associação também aplica-se a lei dos nós, a segunda lei de Kirchhoff. E para a associação em série, a corrente é a mesma em todos os dispositivos do circuito, mas a tensão não. Portanto aplicamos as leis de Kirchhoff para circuitos mistos. Na seção 3.3 discutiremos este tipo de associação.

### 3.3 Associação mista

Quando um circuito é formado por mais de uma associação, ou seja, em paralela e em série, denominamos como sendo associação mista.

Figura 10 - Associação mista de capacitores e indutores



Fonte: Própria(2017)

O circuito da Figura 10, possui dois nós sendo **a** e **b**, os quais são consequência da divisão da corrente. Essas correntes fluem para os capacitores:  $C_2$  e  $C_3$ . Para encontrar as correntes neste circuito, devemos aplicar a segunda lei de Kirchhoff.

Aplicando a lei dos nós para os pontos **a** e **b** da Figura 10, temos:

$$I_2 + I_3 = I_1. \quad (2.10)$$

A lei dos nós ou segunda lei de Kirchhoff, diz que, a soma das correntes que aponta para fora do nó é igual a soma das correntes que chegam até o nó. Isto é uma consequência da conservação das cargas existentes no circuito, ou seja, garantindo que as cargas não se acumulem nos terminais dos circuito.

A regra dos nós aplica-se quando o circuito é formado por mais de uma malha. Para resolver esses tipos de circuitos aplicamos também a primeira lei Kirchhoff ou lei das malhas. Conforme Sears e Zemasnky (2009, p. 173) “a soma algébrica de todas as diferenças de potenciais através de uma malha, incluindo os elementos resistivos e a fem de todas as fontes, que devem ser igual a zero”. Como citado, podemos expressar matematicamente como sendo:

$$\sum V = 0. \quad (2.11)$$

A expressão matemática dada pela equação (2.11) representa a lei das malhas ou primeira lei de Kirchhoff, válida para qualquer circuito com n malhas. A seguir, apresentamos a metodologia utilizada neste trabalho.

## 4 METODOLOGIA

Iniciamos este trabalho acadêmico fazendo um estudo bibliográfico sobre filtros elétricos, a partir de elementos passivos de um circuito elétrico até filtros atuais. Observamos que o capítulo 10, do volume 3, do livro de física básica do autor H. Moysés Nunssenzveig tratava sobre estudos teóricos de filtro seletores em baixas frequência apresentando resultados teóricos para o espectro de frequência. Na obtenção dos resultados desta monografia dividimos nossa metodologia em quatro etapas: estudo bibliográfico, procedimento computacional, veracidade computacional e contribuição científica. Como resultado central desse trabalho apresentamos o comportamento gráfico do espectro de frequência apresentado por Moysés de forma teórica. Os resultados gráficos estão expostos no capítulo quatro deste trabalho. Eles foram obtidos a partir de implementações computacionais das equações que descrevem as características dos filtros CL e LC do livro de Física básica III, do escritor Moysés Nunssenzveig. Os resultados gráficos apresentam excelente concordância com a teórica.

### 4.1 Estudo bibliográfico

- Definições teóricas e expressões matemáticas que dizem respeito aos filtros elétricos;
- Estudo de linguagem de programação C++ para as devidas implementações computacionais das expressões matemáticas dos filtros;
- Estudo de simulações gráficas na implementação computacional;
- Criação dos códigos com as definições das equações para cada problema investigado, obtidos a partir do capítulo 10, do volume 3, de física básica do escritor Moysés Nunssenzveig.

### 4.2 Procedimento computacional

No programa dos códigos ressalvamos antes de iniciar que, não instalamos a biblioteca de texto para as normas da gramática brasileira ou

associação brasileira de normas técnicas (abnt), ou seja, devido a sua versão gratuita o software é limitado e nem todas as bibliotecas estão acessíveis. Como os pacotes das bibliotecas não foram instaladas no compilador Dev-C/C++, parte do código não obedece a gramática. Contudo, a falta deste recurso não interfere nos resultados. As etapas para construção do código:

- Instalação do compilador Dev-C/C++, para realizar a etapa 2, observe a Figura 11, que ilustra a interface do compilador, com algumas variáveis definidas como KE que representa o limite de frequências para  $\omega$ , varrer o circuito. Chamaremos essa frequência de oscilação livre.

Figura 11 - Interface do compilador Dev-C/C++

```

1  #include <graphics.h>
2  #include <limits.h>
3  #include <dos.h>
4  #include <iostream>
5  #include <conio.h>
6  #include <cmath>
7  #include <cstdlib>
8  #include <fstream>
9  #include <iomanip>
10 #include <string>
11
12 #define KF 100
13 using namespace std;
14
15 main()
16
17
18 {
19
20 //criando tela de grafico
21     initwindow(700,600);
22
23     setcolor(YELLOW);
24     moveto(150,250);
25     lineto(150,100);
26     moveto(150,250);
27     lineto(120,250);

```

Fonte: Própria(2017)

Fizemos uma substituição na declaração da variável  $\omega$  para x no código, onde atribuímos para ser uma variável local. O limite de variação desta frequência  $\omega$ , atribuímos uma variável global KE (primeira seta, na linha 12) em

que podemos variar seus valores e isso depende do seu interesse de investigação para o espectro de frequência.

- Para a seta 2 na linha 20 do código da Figura 11 acima, determinamos as dimensões para a tela esboçar os gráficos.
- Figura 12 abaixo, ilustra as equações implementadas em códigos:

Figura 12 - Equações implementadas

```

76      cout<<endl;
77      cin>>E;
78      cout<<endl;
79
80      cout<<"O valor escolhido para a fem E foi " <<E;
81      cout<<endl;
82      cout<<"tudo pronto vamos comecar"<<endl;
83      cout<<endl;
84
85      Wo = 1/(sqrt(L*C));
86
87      cout<<" Observe que o valor da frecuencia de oscilacao livre Wo = " <<Wo<<endl;
88
89
90
91      cout<<"Vamos a resposta " <<endl;
92
93
94      for(x=0; x<=KE; x++)
95      {
96          V[x] = E/(1-pow((x/Wo),2));
97      }
98
99
100
101
102
103      //DESENHA O GRAFICO
104
105      for(x=0;x<=KE; x++)
106      {
107          setcolor(15);
108          lineto(150+x,250+(-V[x]));
109      }
110
111      /* escrever o campo E para um arquivo "Ex" */
112
113
114      ofstream escreve;
115
116      escreve.open("pontos FPB exe 1");
117      escreve<<" "\n";
118      for (x=0; x<=KE; x++)
119          escreve<<x<<" " << setiosflags (ios::fixed)<< setprecision (3)<< V[x]<<endl;
120      escreve.close();

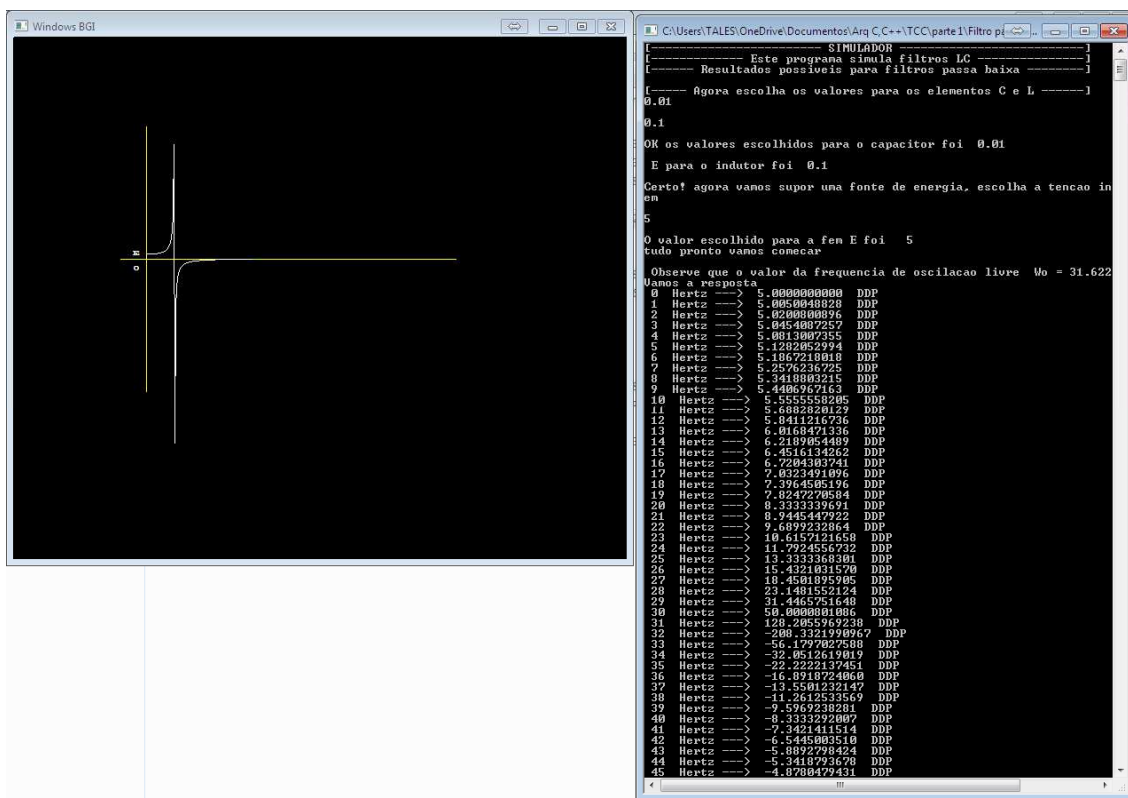
```

Fonte: Própria (2017)

As elipses **a** e **b** selecionam as funções principais do código, que são as equações que geram os pontos, ou seja, os resultados, portanto a elipse **a** representa a equação (4.6), o **for** é o laço do comando que contém as repetições da variação da frequência  $x$  no domínio de 0 a  $KE$ . E a elipse **b** seleciona o laço de repetições do comando **for** para as variáveis  $V[x]$  e  $x$ , sendo estas com comandos de gerar o gráfico, o qual está ilustrado na Figura 13.



Figura 13 - Tela gráfica a direita e a esquerda a tela visual c



Fonte: Própria (2017)

Os pontos da tela à esquerda, o visual c, foram exportados em um formato de texto (txt), em seguida reconstruído o gráfico a direita em um software desenvolvido para isso, por exemplo o Excel da Microsoft. Em seguida investigamos as propriedades dos gráficos e analisamos com as definições deduzidas no livro do autor H. Moysés.

### 4.3 Veracidade computacional

Os resultados gerados pela implementação em C/C++ das expressões matemáticas, concordaram com as definições teóricas definidas pelo autor Moysés N. A seguir, apresentamos algumas aplicações destes filtros CL e LC, onde atualmente são empregados em aparelhos eletrônicos como rádios e televisões.

#### **4.4 Contribuição científica**

Foram desenvolvidas atividades de auxílio no projeto de extensão do grupo de pesquisa GEMCA, com alunos incluídos em projetos de pesquisa de iniciação científica financiados pelo CNPq - conselho nacional de pesquisa e inovação.

Trabalhos publicados em eventos nacionais e regionais como: CONNEPI 2016, MOMAG 2016 e Iniciação científica: 2014/2015, 2015/2016 e 2016/2017. Com projetos desenvolvidos nas áreas de Eletromagnetismo, Computação, Física e Telecomunicações.

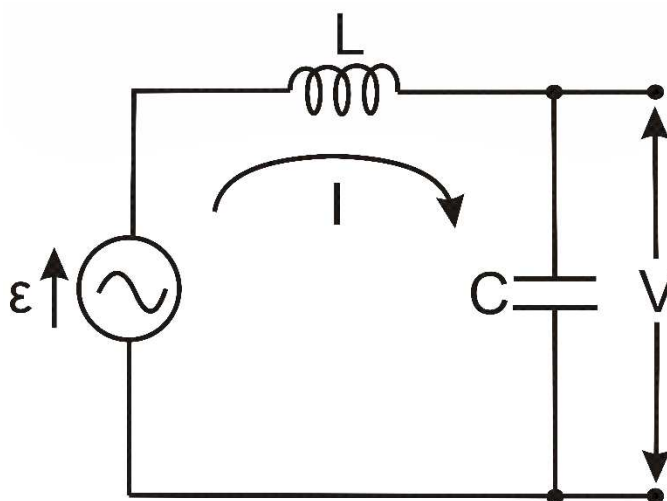
## 5 RESULTADOS COMPUTACIONAIS PARA OS FILTROS LC E CL

Em acordo com o capítulo 10 do livro titulado como “curso de física básica 3” do autor H. Moysés Nunsseinzveig, apresentamos os gráficos dos espectros de frequências para vários filtros. Essa descrição foi realizada através de implementações das equações definidas em cada seção do capítulo 10, utilizando a linguagem computacional C/C++. No final de cada seção, estão as implementações dos códigos em tabelas.

### 5.1 Filtros LC

Os filtros LC, é um dispositivo construído a partir de elementos passivos com a propriedade de interagir com a corrente e são eles capacitores, indutores, resistores e tensores, um exemplo de filtro passa baixa é aquele capaz de bloquear transmissão de sinais acima de sua frequência de corte Figura 14.

Figura 14 - Filtro passa baixas frequências



Fonte: Própria(2017)

Supomos, que a fem  $\varepsilon$  é associada a uma fonte de corrente alternada, portanto, a equação (5.1) demonstra a corrente no circuito.

$$\bar{I} = \frac{\varepsilon}{\bar{Z}} \quad (5.1)$$

Temos  $\bar{Z}$  como uma impedância complexa do circuito da Figura 14, expressa em termos da impedância indutiva e capacitiva:

$$\bar{Z} = i \left( \omega L - \frac{1}{\omega C} \right) \quad (5.2)$$

onde

$$\bar{Z}_L = i\omega L \quad (5.3)$$

e

$$\bar{Z}_C = \frac{1}{i\omega C}. \quad (5.4)$$

Logo, o potencial entre os terminais do capacitor, obtemos a partir da equação (5.5)

$$\bar{V} = \bar{I} \bar{Z}_C. \quad (5.5)$$

Como a impedância complexa  $\bar{Z}$  é expressa em termos do indutor e do capacitor, respectivamente as equações (5.3) e (5.4), então para o potencial  $V$  entre os terminais do capacitor, substituímos a equação (5.1) na equação (5.5), logo:

$$\bar{V} = \frac{\varepsilon}{\bar{Z}} \bar{Z}_C \quad (5.6)$$

substituindo, as equações (5.2) e (5.4) na equação (5.6), obtemos uma nova relação matemática para o potencial.

$$\bar{V} = \frac{\varepsilon}{i^2 \omega C} \left( \omega L - \frac{1}{\omega C} \right). \quad (5.7)$$

Observando a relação para os números complexos, onde  $i^2 = -1$ , e dividimos equação (5.7) por  $\omega^2 LC$ , obtemos a resposta para o potencial, dada por:

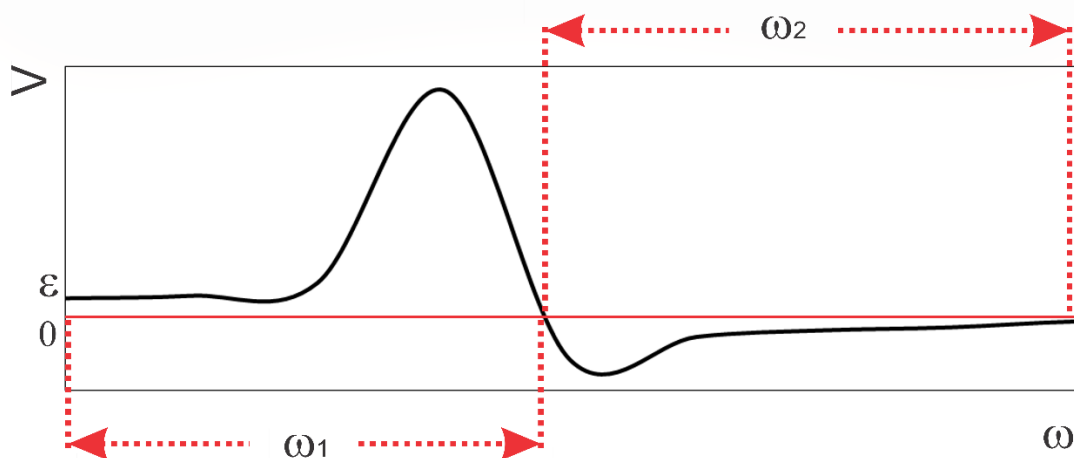
$$\bar{V} = \frac{\varepsilon}{(1 - \omega^2 LC)} \quad (5.8)$$

a frequência de oscilação livre é dada por,  $\omega_0 = \frac{1}{\sqrt{LC}}$ . E o potencial em função da frequência livre é dada pela equação (5.9).

$$\bar{V} = \frac{\varepsilon}{1 - \left(\frac{\omega}{\omega_0}\right)^2} \quad (5.9)$$

Com efeito, vamos observar o potencial médio, a partir das seguintes condições de contorno: o potencial médio tende a  $\varepsilon$ , quando  $\omega$  tende a zero, e o potencial tende a zero, quando  $\omega$  for para o infinito. O comportamento da frequência é mostrado no Gráfico 5.1.

Gráfico 5.1 - Filtro Passa baixa, gerada pelo código FPB da Tabela 3



Fonte: Própria(2017)

Entanto o dispositivo permitirá a “passagem” de sinais de baixa frequência para os quais o indutor L tende à um curto circuito, e “bloquear” sinais de alta frequência e isso ocorre por causa do efeito do indutor e da Lei de Lenz.

Tabela 3 - Código do FPB, no código substituímos  $\omega$  por x,  $\omega_0$  por Wo e  $\varepsilon$  por E

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>

#define KE 100
using namespace std;

main()
{
//criando tela de grafico
    initwindow(700,600);

    setcolor(YELLOW);
    moveto(150,250);
    lineto(150,100);
    moveto(150,250);
    lineto(120,250);
    moveto(150,250);
    lineto(150,400);
    moveto(150,250);
    lineto(500,250);

    moveto(150,250);

//VARIABLES DO CODIGO QUE REPRESENTA O ESPAÇO!

    float W[KE], V[KE];
    int E, x, y;
    float C, L, Wo;

    // DEFININDO MINIMOS, MAXIMOS E OS TEXTOS, NA TELA GRAFICA
    setcolor( WHITE);
    outtextxy(135, 195,"E");
    setcolor( WHITE);
    outtextxy(135, 255,"0");

    // Obs que os C L e E sao constantes atribuidas durante o processo.

    cout<<"[----- SIMULADOR -----]"<<endl;

```

```

cout<<"[----- Este programa simula filtros LC -----]"<<endl;
cout<<"[----- Resultados possiveis para filtros passa baixa -----]"<<endl;
cout<<endl;
cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;

    cin>>C;
cout<<endl;

    cin>>L;
cout<<endl;
cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
cout<<endl;
cout<<" E para o indutor foi "<<L<<endl;
cout<<endl;
    cout<<"Certo! agora vamos supor uma fonte de energia, escolha a tencao
inicial para a fem"<<endl;

        cout<<endl;
        cin>>E;
        cout<<endl;
        cout<<"O valor escolhido para a fem E foi "<<E;
        cout<<endl;
        cout<<"tudo pronto vamos comecar"<<endl;
        cout<<endl;
        Wo = 1/(sqrt(L*C));
        cout<<" Observe que o valor da frequencia de oscilacao livre Wo = "<<Wo<<endl;
        cout<<"Vamos a resposta "<<endl;

for(x=0; x<=KE; x++)
{V[x] = E/(1-pow((x/Wo),2)); }

//DESENHA O GRAFICO
for(x=0;x<=KE; x++) {
    setcolor(15);
    lineto(150+100*x,250+(-V[x])); }

// escrever o campo E para um arquivo "Ex"
ofstream escreve;
escreve.open("pontos FPB exe 1");
escreve<<"\n";
for (x=0; x<=KE; x++)
    escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (3)<< V[x]<<endl;
    escreve.close();
// ESCRIVE NA TELA OS PONTOS!!
for (x=0; x<=KE; x++) {
cout.width();
cout<<" "<<x<<" "<<" Hertz "<<"---> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (10)<<V[x]<<" ";
cout.width();
cout<<" DDP "<<endl; }
cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0;}

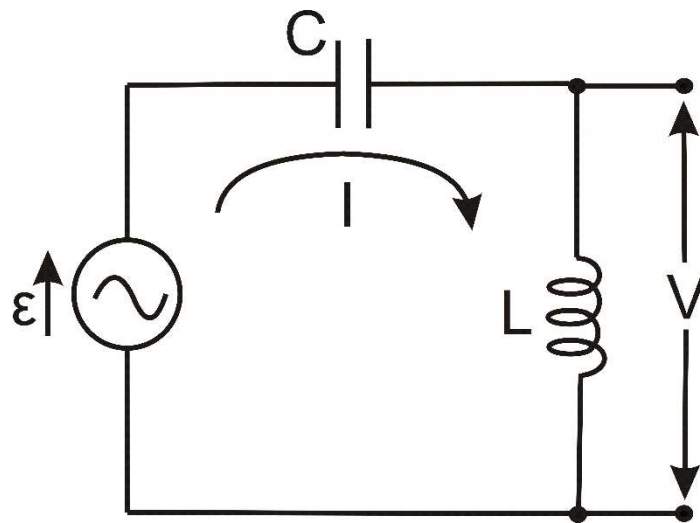
```

Fonte: Própria(2017)

## 5.2 Filtros CL

Após observarmos o filtro LC, faremos uma abordagem a respeito do filtro CL, seguindo o capítulo 10, como escrito no livro. O filtro CL, apresenta um comportamento do tipo que bloqueia sinais de baixa frequência. Repetiremos o mesmo procedimento para mostrar o espectro de frequência do filtro LC.

Figura 15 - Filtro bloqueia baixas frequências



Fonte: Própria(2017)

O circuito apresentado pela Figura 15 é do filtro CL. Observe que suas componentes **L** e **C** foram alocadas entre si, permanecendo  $\bar{I}$  e  $\bar{Z}$  inalterados. Mostraremos que o potencial médio,  $\bar{V}$  entre os terminais do indutor é dado por.

$$\bar{V} = \bar{I} \bar{Z}_L \quad (5.10)$$

substituindo a impedância indutiva da equação (5.3) na equação (5.10), encontramos uma nova expressão para o potencial médio entre os terminais do indutor, dado por:

$$\bar{V} = i\omega L \bar{I}. \quad (5.11)$$



Usando a expressão da corrente, dada pela equação (5.1), e substituindo na equação (5.11), obtemos o potencial médio a partir da força eletromotriz e da impedância, dada pela equação (5.12)

$$\bar{V} = \frac{i\omega L \varepsilon}{\bar{Z}}. \quad (5.12)$$

A impedância complexa é dada pela equação (5.2), onde reescrevemos a equação (5.12) da forma:

$$\bar{V} = \frac{i\omega L \varepsilon}{i\left(\omega L - \frac{1}{\omega C}\right)} \quad (5.13)$$

e dividimos a equação (5.13) por  $i\omega L$ , obtendo a:

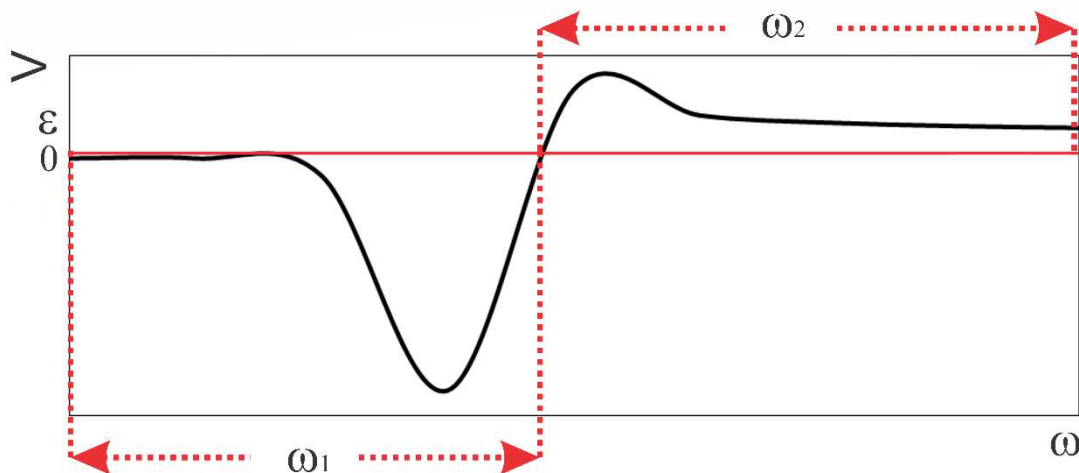
$$\bar{V} = \frac{\varepsilon}{\left(1 - \frac{1}{\omega^2 LC}\right)}. \quad (5.14)$$

Comparando a equação (5.9) com a equação (5.14), notamos que houve apenas uma inversão das frequências. Com apenas esta mudança, que parece simples, obtemos uma resposta para o filtro CL completamente distinta do filtro LC, onde fizemos  $\omega_0 = \frac{1}{\sqrt{CL}}$ . Para obter a equação (5.15).

$$\bar{V} = \frac{\varepsilon}{1 - \left(\frac{\omega_0}{\omega}\right)^2}. \quad (5.15)$$

No Gráfico 5.2, ilustramos graficamente o comportamento do potencial médio, da equação (5.15), com respeito a frequência.

Gráfico 5.2 - Filtro bloqueia baixa. Gerada pelo código FBB da Tabela 4



Fonte: Própria(2017)

Observando o Gráfico 5.2, onde aplicamos as condições de contorno para a frequência, indo a zero pela esquerda e ao infinito pela direita, o potencial médio tende a zero e a força eletromotriz  $\varepsilon$ , respectivamente. Portanto, o circuito apresenta o efeito inverso ao anterior, ou seja, permite a passagem de frequências alta, quando  $C$  tende a um curto-circuito e bloqueia os sinais de baixa frequência.

Tabela 4 - Código FBB

```
#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>
#define KE 100
using namespace std;

main() {
    //criando tela de grafico
    initwindow(700,600);

    setcolor(YELLOW);
    moveto(150,250);
    lineto(150,100);
    moveto(150,250);
    lineto(120,250);
```

```

moveto(150,250);
lineto(150,400);
moveto(150,250);
lineto(500,250);

moveto(150,250);

//VARIAVEIS DO CODIGO Q REPRESENTA O ESPAÇO!

float W[KE], V[KE];
int E, x, y;
float C, L, Wo;

// DEFININDO OS MAXIMOS E OS TEXTOS!!
setcolor( WHITE);
outtextxy(135, 195,"E");
setcolor( WHITE);
outtextxy(135, 255,"0");

// Obs que os C L e E sao constantes atribuidas durante o processo.

cout<<"[----- SIMULADOR -----]"<<endl;
cout<<"[----- Este programa gera resultados -----]"<<endl;
cout<<"[---- Resultados possiveis para filtros bloqueia baixa ----]"<<endl;
cout<<"[---- Agora escolha os valores para os elementos C e L ----]"<<endl;

cin>>C;
cout<<endl;
cin>>L;
cout<<endl;

cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
cout<<endl;
cout<<" E para o indutor foi "<<L<<endl;
cout<<endl;

cout<<"Certo! agora vamos supor uma fonte de energia, escolha a tencao inicial para a
fem"<<endl;

cout<<endl;
cin>>E;
cout<<endl;
cout<<"O valor escolhido para a fem E foi "<<E;
cout<<endl;
cout<<"tudo pronto vamos comecar"<<endl;
cout<<endl;

//INICIO DA FUNCAO
Wo = 1/(sqrt(L*C));

cout<<" Observe que o valor da frequencia de oscilacao livre Wo = "<<Wo<<endl;
cout<<"Vamos a resposta "<<endl;

for(x=0; x<=KE; x++)
{V[x] = E/(1-pow((Wo/x),2)); }

//DESENHA O GRAFICO
for(x=0;x<=KE; x++)
{ setcolor(15);
lineto(150+100*x,250+(-V[x])); }

```

```

// Escreve os dados da tela em txt no arquivo pontos FPB exe 1.

ofstream escreve;
escreve.open("pontos FPB exe 1");
escreve<<"\n";
for (x=0; x<=KE; x++)
    escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (3)<< V[x]<<endl;
    escreve.close();

// ESCREVE NA TELA OS PONTOS!!

for (x=0; x<=KE; x++)
{ cout.width();
cout<<" "<<x<<" "<<" Hertz "<<"---> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (10)<<V[x]<<" ";
cout.width();
cout<<" DDP "<<endl; }
cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0; }

```

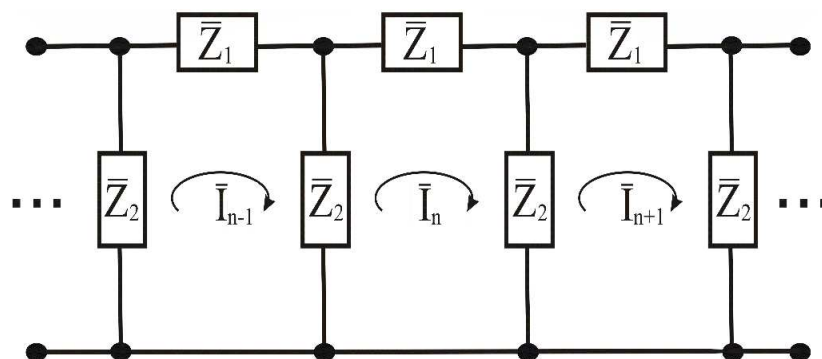
Fonte: Própria(2017)

### 5.3 Circuitos em redes periódicas

Utilizaremos o circuito da Figura 16, como o elemento periódico em uma malha infinita e observaremos como o comportamento do novo circuito afeta na resposta do filtro. Esse filtro é classificado como sendo periódico.

A partir do circuito da Figura 14, substituímos os elementos L e C, por duas impedâncias complexas sendo uma indutiva e outra capacitiva, representada por:  $\bar{Z}_1$  e  $\bar{Z}_2$ .

Figura 16 - Filtro periódico infinito



Fonte: Própria(2017)

Sendo  $\bar{I}_n$  a corrente na n-ésima malha desse filtro. Aplicamos agora a primeira Lei de Kirchhoff. A corrente neste circuito será escrita da forma:

$$(\bar{I}_n - \bar{I}_{n-1})\bar{Z}_2 + \bar{I}_n \bar{Z}_1 + (\bar{I}_n - \bar{I}_{n+1})\bar{Z}_2 = 0. \quad (5.16)$$

Expandimos a equação (5.16), do tipo:

$$\bar{I}_n \bar{Z}_2 - \bar{I}_{n-1} \bar{Z}_2 + \bar{I}_n \bar{Z}_1 - \bar{I}_{n+1} \bar{Z}_2 + \bar{I}_n \bar{Z}_2 = 0 \quad (5.17)$$

colocamos a corrente  $\bar{I}_n$  da equação (2.17) em evidência Logo temos:

$$-\bar{I}_{n+1} \bar{Z}_2 + (\bar{Z}_1 + 2\bar{Z}_2)\bar{I}_n - \bar{I}_{n-1} \bar{Z}_2 = 0. \quad (5.18)$$

Sendo  $\bar{Z}_1$  e  $\bar{Z}_2$  valores dados, encontraremos uma forma em que a  $\bar{I}_n$  dependa de n. A variável discreta n toma os valores inteiros,  $n = 1, 2, 3, 4, \dots, n$ .

A diferença  $\bar{I}_n - \bar{I}_{n-1} = \Delta \bar{I}_n$  chama-se uma diferença finita, análogo e discreta de uma diferencial:  $\Delta df = f(x + \Delta x) - f(x)$ , e a equação (5.18) também é uma equação de diferenças finitas, análogo discreto de uma equação diferencial.

Observe que os coeficientes das diferenças são constantes. Podemos resolve-la da mesma forma de uma equação diferencial linear de coeficientes constantes, por uma exponencial na variável (n):

$$\bar{I}_n = \bar{I}_0 e^{\alpha n} \quad (5.19)$$

onde,  $\alpha$  é uma constante a se determinar. Isso dá

$$\bar{I}_{n+1} = e^{\alpha} \bar{I}_n \equiv p \bar{I}_n \quad (p \equiv e^{\alpha}). \quad (5.20)$$

E

$$\bar{I}_{n-1} = e^{-\alpha} \bar{I}_n \equiv \frac{1}{p} \bar{I}_n. \quad (5.21)$$

Substituímos as equações (5.20) e (5.21) na equação (5.18), temos:

$$-p\bar{Z}_2 + (\bar{Z}_1 + 2\bar{Z}_2) - \frac{1}{p}\bar{Z}_2 = 0. \quad (5.22)$$

Multiplicando a equação (5.22) por  $-p$  e depois dividimos por  $\bar{Z}_2$ , obtemos uma equação característica, dada por:

$$p^2 - \left(\frac{\bar{Z}_1}{\bar{Z}_2} + 2\right)p + 1 = 0 \quad (5.23)$$

sendo suas raízes dadas por:

$$p_{\pm} = 1 + \frac{\bar{Z}_1}{2\bar{Z}_2} \pm \sqrt{\left(1 + \frac{\bar{Z}_1}{2\bar{Z}_2}\right)^2 - 1}. \quad (5.24)$$

É fácil ver que

$$p_+ p_- = 1 \quad \left\{ p_- = \frac{1}{p_+} \right\} \quad \left\{ p_+ = e^{\alpha} \Rightarrow p_- = e^{-\alpha} \right\}. \quad (5.25)$$

Para ambas as raízes, temos:

$$\frac{1}{2} \left( p + \frac{1}{p} \right) = 1 + \frac{\bar{Z}_1}{2\bar{Z}_2} \quad (5.26)$$

Sabendo que,  $\bar{Z}_1$  e  $\bar{Z}_2$  são respectivamente  $i \omega L$  e  $\frac{-i}{\omega C}$ , ou seja, imaginários puros, de forma que,  $\frac{\bar{Z}_1}{2\bar{Z}_2}$ , é real.

Vamos admitir isso e que  $\bar{Z}_1$  e  $\bar{Z}_2$  são reatâncias puras ( $R = 0$ ).

$$\left(1 + \frac{\bar{Z}_1}{2\bar{Z}_2}\right)^2 < 1.$$

O radical em  $p_{\pm}$  é imaginário puro e as raízes são complexas conjugados, caso contrário, elas são reais.

Começemos pelo primeiro caso:

$$\left(1 + \frac{\bar{Z}_1}{2\bar{Z}_2}\right)^2 < 1 \left\{ -1 < 1 + \frac{\bar{Z}_1}{2\bar{Z}_2} < 1 \left\{ -1 < \frac{\bar{Z}_1}{4\bar{Z}_2} < 0. \right. \right. \quad (5.27)$$

Nesse caso,  $p_- = (p_+)^*$  é (complexo conjugado), e  $1 = p_+p_- = |p_+|^2$ . Logo,  $p_{\pm}$  são fatores de fase; as equações (5.20) e (5.21) nos dá:

$$\frac{\bar{I}_{n+1}}{\bar{I}_n} = p_{\pm} = e^{\pm i\beta} \quad (5.28)$$

onde  $\beta$  é a defasagem da corrente entre duas malhas consecutivas.

Obtemos, a partir da equação (5.26), a equação (5.29)

$$\frac{1}{2}\left(p + \frac{1}{p}\right) = \frac{1}{2}(e^{i\beta} + e^{-i\beta}) = \cos \beta = 1 + \frac{\bar{Z}_1}{2\bar{Z}_2}. \quad (5.29)$$

O que dá:

$$\beta = \cos^{-1}\left(1 + \frac{\bar{Z}_1}{2\bar{Z}_2}\right). \quad (5.30)$$

A corrente complexa na malha  $n$  fica, para a raiz  $p_-$ ,

$$\bar{I}_n = \bar{I}_0 e^{-i\beta n}. \quad (5.31)$$

ou seja,

$$\bar{I}_n = \bar{I}_0 e^{-i(\omega t - \beta n)} \quad (5.32)$$

e como  $\varphi = (\omega t - \beta n)$ , substituimos na equação (5.32) logo:

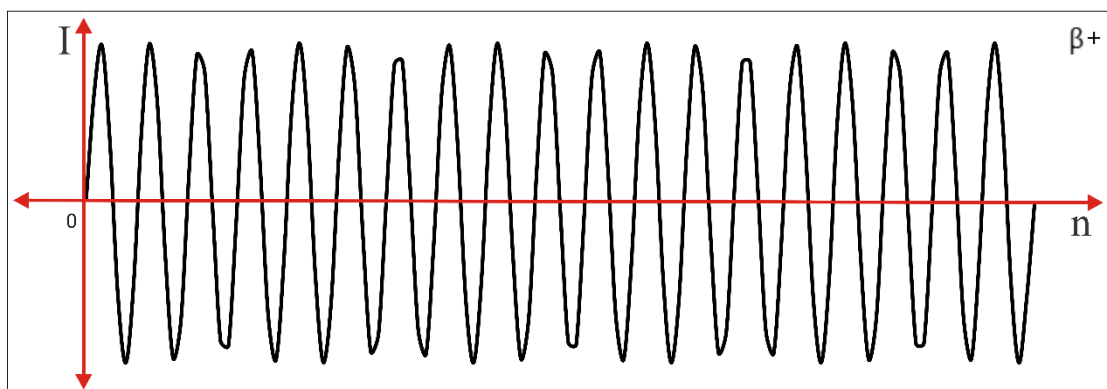
$$\bar{I}_n = \bar{I}_{0m} e^{-i\varphi}. \quad (5.33)$$

Contudo, a corrente real é dada por:

$$\bar{I}_n = \bar{I}_{0m} \cos(\beta n - \omega t - \varphi). \quad (5.34)$$

Portanto, observando a equação (5.34), identificamos que representa uma onda harmônica progressiva de corrente (*Física Básica 2*, Sec. 5.2). Esta onda propaga-se ao longo de uma sequência de malhas, da esquerda para a direita, com “número de onda  $\beta$ ” e “velocidade de propagação”  $\omega/\beta$ .

Gráfico 5.3 - OHP, com defasagem  $\beta +$ , movendo para a direita. Gerada pelo código da Tabela 5

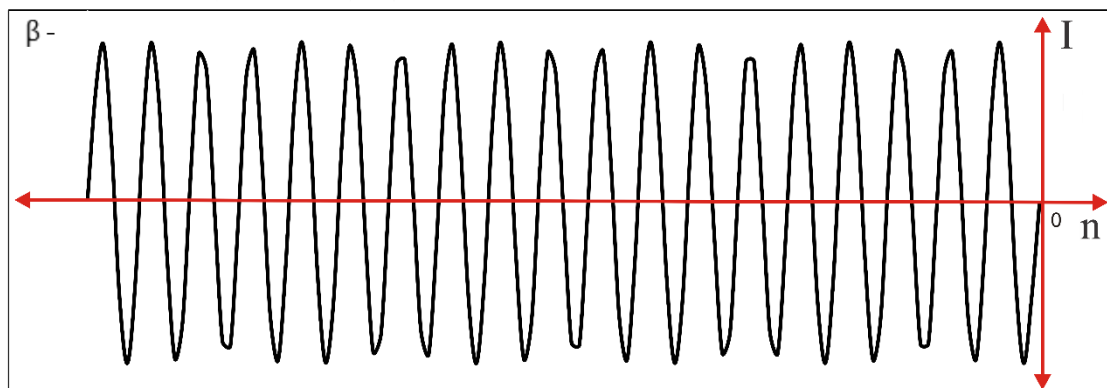


Fonte: Própria(2017)



Para a raiz de  $p_+$  fazemos a inversão,  $\beta \rightarrow -\beta$ . Essa onda é análoga a de beta positivo, mas propaga-se da direita para a esquerda. A superposição dessas ondas representa a solução geral.

Gráfico 5.4 - OHP, com defasagem  $-\beta$ , propagando para a esquerda. E gerada pelo código da Tabela 5



Fonte: Própria(2017)

Se

$$\left(1 + \frac{\bar{z}_1}{2\bar{z}_2}\right)^2 > 1$$

A expressão entre parênteses pode ser  $> 1$  ou  $< -1$ , como  $\omega$  tende ao  $+\infty$ . Então, o cosseno varia de 0 a  $2\pi$ , provando ser de fato, uma onda senoidal.

Tabela 5 - Código OHP. No código substituímos as variáveis:  $\beta$  por B e  $\bar{I}_{0m}$  por I0, o sentido de OHP depende do sinal de  $\beta$

```
#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>
```

```

#define KE 120
using namespace std;

main() { //criando tela de grafico
    initwindow(700,600);

    setcolor(YELLOW);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(50,250);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(500,250);
    moveto(250,250);

//VARIABLES DO CODIGO Q REPRESENTA O ESPAÇO!

    float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE];
    int E, x, y,n,t;
    float C, L,T, Wo,I0, Pi;

    // DEFININDO OS MAXIMOS E OS TEXTOS!!
    setcolor( WHITE);
    outtextxy(245, 255,"0");

    // Obs que os C L e E sao constantes atribuidas durante o processo.

    cout<<"[----- SIMULADOR -----]"<<endl;
    cout<<"[----- Este programa gera resultados -----]"<<endl;
    cout<<"[----- Resultados possiveis para filtros Infinitos 1 -----]"<<endl;
    cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;
    cin>>C;
cout<<endl;
    cin>>L;
cout<<endl;
    cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
    cout<<endl;
    cout<<" E para o indutor foi "<<L<<endl;

cout<<endl;
cout<<"Certo! agora vamos supor uma fonte de energia, escolha a corrente inicial"<<endl;
cout<<endl;
    cin>>I0;

cout<<endl;
cout<<endl;
    cout<<"O valor da corrente inicial foi "<< I0;
    cout<<endl;
    cout<<"tudo pronto vamos comecar"<<endl;
    cout<<endl;

//INICIO DA FUNCAO
// FUNCAO DAS REATANCIAS//
    // x=w
    Wo=1/(sqrt(L*C));
    for(x=1; x<=KE; x++)
        { Z1[x] = Wo*L;
          Z2[x] = 1/(Wo*C); }

//escrever as impedancias Z1 e Z2 na tela

```

```

for(x=1; x<=KE; x++){
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl; }

// funcao de defasagem beta B
for (x=0; x<=KE; x++)
{ B[x] = acos(1-(((pow(Wo,2))*(L*C))/2)); }
// FUNCAO DA CORRENTE NA MALHA N

for (x=0; x<=KE; x++){
I[x] = I0*cos((B[x]-(x+(2*Pi)))); }

// OBS mude apenas o B+ e B-, mas mude o sinal de x para -x apenas na f do grafico
cout<<" Observe que o valor da defasagem e B nas malhas"<<endl;

for(x=0; x<=KE; x++) {
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (8)<<B[x]<<" ";
cout<<endl; }
cout<<"Vamos ao valor da corrente na malha N "<<endl;

//DESENHA O GRAFICO
for(x=0;x<=KE; x++)
{ setcolor(15);
lineto(250+x,250+(I[x])); }
/* salvar os pontos na pasta "pontos F INFINITO Onda +B " */

ofstream escreve;
escreve.open("pontos F INFINITO Onda +B");
escreve<<"\n";
for (x=0; x<=KE; x++)
escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (8)<<I[x]<<endl;
escreve.close();

// ESCRIVE NA TELA OS PONTOS!!
for (x=0; x<=KE; x++)
{ cout.width();
cout<<" "<<x<<" "<<" ----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (8)<<I[x]<<" ";
cout.width();
cout<<" A "<<endl; }

cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0;}

```

Fonte: Própria(2017)

### 5.3.1 Verificando o caso (i):

Para a condição

$$\frac{\bar{Z}_1}{4\bar{Z}_2} > 0 \quad (5.35)$$

neste caso,  $p_+ > 1 = e^\alpha$  com  $\alpha > 0$ , e  $p_- < 1 = e^{-\alpha}$ , onde

$$\frac{1}{2}(e^\alpha + e^{-\alpha}) = \text{ch } \alpha = 1 + \frac{\bar{Z}_1}{2\bar{Z}_2} \quad (5.36)$$

logo  $\alpha$  é a constante de atenuação da equação (5.36):

$$\alpha = \text{ch}^{-1} \left( 1 + \frac{\bar{Z}_1}{2\bar{Z}_2} \right). \quad (5.37)$$

Portanto o ch da equação (5.37) é um cosseno hiperbólico.  
para a raiz  $p_-$ , a corrente na malha n será:

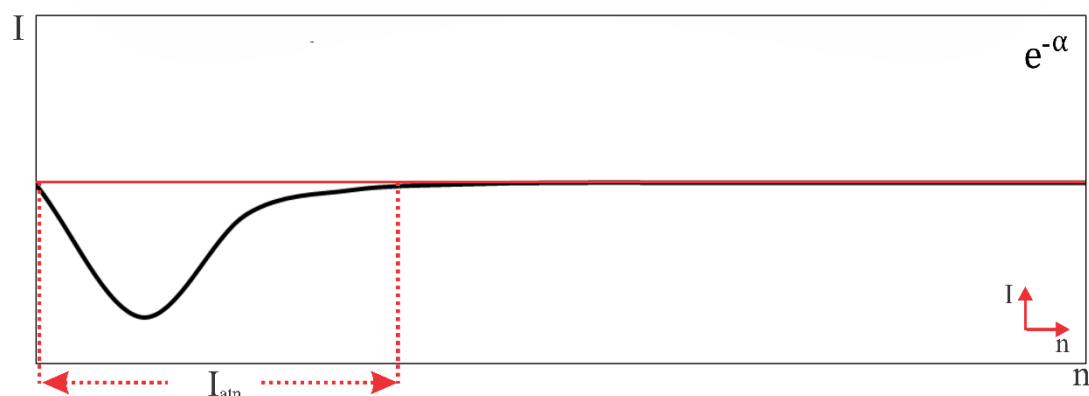
$$\bar{I}_n = \bar{I}_0 e^{-\alpha n} \quad \{\bar{I}_n = \bar{I}_0 e^{-\alpha n + i\omega t}. \quad (5.37)$$

E para a corrente real definimos a equação (5.38) como:

$$\bar{I}_n = \bar{I}_{0m} e^{-\alpha n} \cos(\omega t + \varphi). \quad (5.38)$$

O comportamento da equação (5.38), é uma onda atenuando para a direita Gráfico 5.5, em termos da equação (5.37).

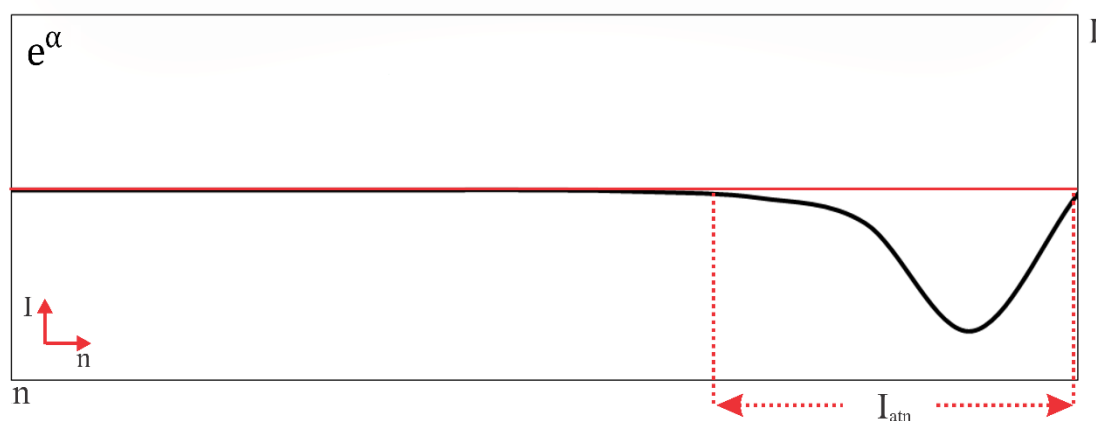
Gráfico 5.5 – Onda atenuando para a direita (OAD). Gerada pelo código OA



Fonte: Própria(2017)

Observe o Gráfico 5.5, que a onda não se propaga, ela apenas atenua-se para a direita, por um fator  $e^{-\alpha}$ , de cada malha para a seguinte. Segundo Moysés (1997, p.221) “Essa solução corresponderia a um gerador de corrente a esquerda alimentado a rede”.

A solução  $p_+ = e^{\alpha}$ , atenua-se para a esquerda, correspondendo à alimentação por um gerador de corrente colocado à direita;  $\alpha$  é a constante de atenuação por seção, observe o Gráfico 5.6.

Gráfico 5.6 – Onda atenuando para a esquerda (OAE). Gerado pelo código OA, com o sinal substituído para  $e^{\alpha}$ 

Fonte: Própria(2017)

Tabela 6 - Código OA. Apenas mudando o sinal de  $\alpha$ , o código mudara o sentido da onda

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>
#define KE 120
using namespace std;
main()
{ //criando tela de grafico
  initwindow(700,600);
  setcolor(YELLOW);
  moveto(250,250);
  lineto(250,250);
  moveto(250,250);
  lineto(50,250);
  moveto(250,250);
  lineto(250,250);
  moveto(250,250);
  lineto(500,250);
  moveto(250,250);
//VARIABLES DO CODIGO Q REPRESENTA O ESPAÇO!
  float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE];
  int E, x, y,n,t, PASSOS;
  float C, L,T, Wo,I0, Pi;

  // DEFININDO OS MAXIMOS E OS TEXTOS!!
  setcolor( WHITE);
  outtextxy(245, 255, "0");
  // Obs que os C L e E sao constantes atribuidas durante o processo.

  cout<<"[----- VAMOS COMECAR-----]"<<endl;
  cout<<"[----- Este programa gera resultados -----]"<<endl;
  cout<<"[--- Resultados possiveis para filtros Infinitos CASO (i) ----]"<<endl;
  cout<<"[---- Agora escolha os valores para os elementos C e L -----]"<<endl;

  cin>>C;
  cout<<endl;
  cin>>L;
  cout<<endl;
  cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
  cout<<endl;
  cout<<" E para o indutor foi "<<L<<endl;
  cout<<endl;
  cout<<"Certo! agora vamos supor uma fonte de energia, escolha a corrente inicial"<<endl;
  cout<<endl;
  cin>>I0;
  cout<<endl;
  cout<<"tudo pronto vamos comecar"<<endl;
  cout<<endl;

  //INICIO DA FUNCAO
  // FUNCAO DAS REATANCIAS//

```

```

        // x=w
        Wo=1/sqrt(L*C);

        for(x=0; x<=KE; x++)
    {
        Z1[x] = Wo*L;
        Z2[x] = 1/(Wo*C); }
        //escrever as impedancias Z1 e Z2 na tela
        for(x=1; x<=KE; x++) {
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl;}
// funcao da constante de atenuacao alfa A
        for (x=0; x<=KE; x++){
            A[x] = acosh(1+(Z1[x]/(2*Z2[x])));}
        // FUNCAO DA CORRENTE NA MALHA N
        for (x=1; x<=KE; x++)
    {
        I[x] = I0*exp(-A[x]*x)*cos(Wo+(2*Pi)); }
        // OBS mude apenas o exp - , para obter a banda para a esquerda. mas mude o sinal de x
        para -x na f do grafico

        cout<<" Observe que o valor da constante de atenuacao Alfa A[x] nas malhas"<<endl;
        for(x=1; x<=KE; x++) {
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (40)<<B[x]<<" ";
cout<<endl; }
        cout<<"Vamos ao valor da corrente na malha N "<<endl;

//DESENHA O GRAFICO
        for(x=1;x<=KE; x++)
    {
        setcolor(15);
        lineto(250+x*5,250-(I[x]*1000)); }
        // a funcao a seguir exporta os pontos em txt na pasta "pontos F INFINITO caso (i)//
ofstream escreve;
escreve.open("pontos F INFINITO caso (i)");
escreve<<"\n";
for (x=1; x<=KE; x++)
escreve<<-x<<" "<< setiosflags (ios::fixed)<< setprecision (3)<<-I[x]<<endl;
escreve.close();

// ESCRIVE NA TELA OS PONTOS!!
        for (x=1; x<=KE; x++)
    { cout.width();
cout<<" "<<x<<" "<<" ----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (40)<<-I[x]<<" ";
cout.width();
cout<<" A "<<endl; }
        cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0; }

```

Fonte: Própria(2017)

### 5.3.2 Verificando o caso (ii):

No caso (i) anterior, verificamos os resultados para condição  $\frac{\bar{Z}_1}{2\bar{Z}_2} > 0$ , a seguir demonstraremos o que acontece para a condição da equação (5.39):

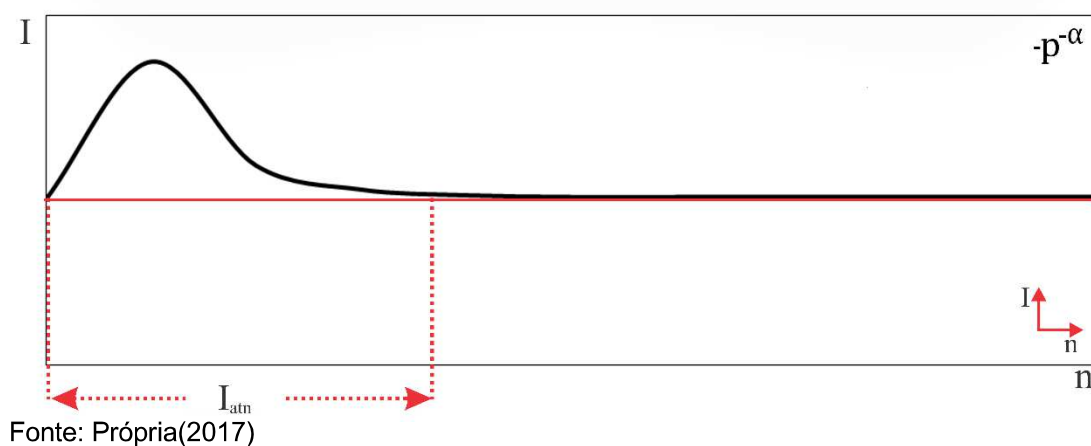
$$1 + \frac{\bar{Z}_1}{2\bar{Z}_2} < -1 \quad \left\{ \frac{\bar{Z}_1}{4\bar{Z}_2} < -1 \right. \quad (5.39)$$

neste caso,  $p_+$  e  $p_-$  são negativos, e podemos tomar a equação (5.39) como:

$$p_{\pm} = -e^{\pm\alpha} (\alpha > 0). \quad (5.40)$$

A única diferença em relação ao caso (i) é que, além de atenuar-se de cada seção para a seguinte, a corrente também troca de sinal (sentido), correspondendo a uma mudança de fase de  $\pi$ . Observe no Gráfico 5.7

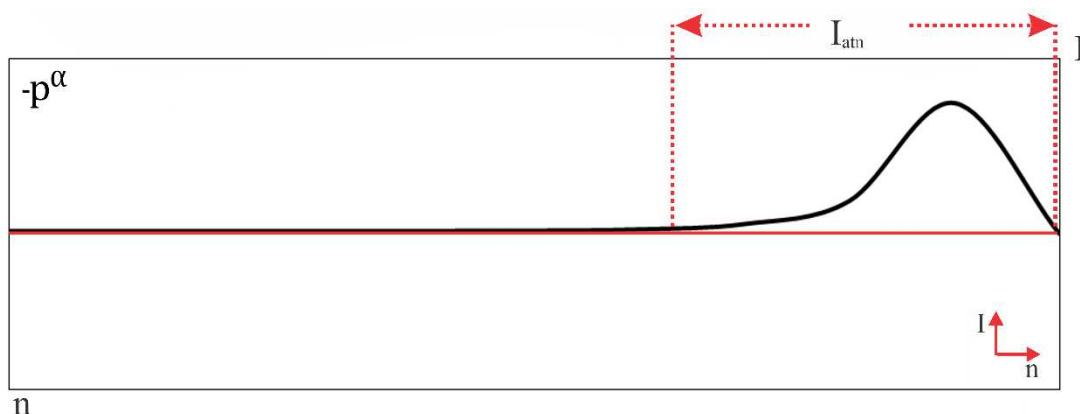
Gráfico 5.7 – Onda com fase invertida para a direita. Gerada pelo código OAFI



E para  $-p^{\alpha}$  temos também uma faixa passante para a direita



Gráfico 5.8 – Onda com fase invertida para a esquerda. Gerada pelo código OAFI



Fonte: Própria(2017)

Em geral a função,

$$\frac{\bar{Z}_1}{\bar{Z}_2}$$

é uma função da frequência  $\omega$ , como as reatâncias, de modo que as condições acima são condições sobre  $\omega$ . Uma faixa de frequências em que a corrente se propaga

$$\left(-1 < \frac{\bar{Z}_1}{4\bar{Z}_2} < 0\right)$$

chama-se banda de passagem, nos outros casos (i) e (ii), em que se atenua, temos uma banda proibida. Conforme Moysés (1997, p.222) “Como pode haver atenuação, se  $\bar{Z}_1$  e  $\bar{Z}_2$  são reatâncias puras, cuja resistência R foi desprezada? A resposta é que a atenuação não corresponde aqui a uma dissipação de energia”. A energia proveniente da fonte de corrente, numa banda proibida, vai ficando acumulada nos capacitores e indutores ao longo da rede, sem que haja dissipação. Em consequência,  $\bar{I}_n \rightarrow 0$  para  $n \rightarrow \infty$ .

Tabela 7 - Código OAFI, substituímos os sinais das raízes:  $-p^\alpha$  para  $-p^{-\alpha}$ , obtemos respostas diferentes

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>
#define KE 120
using namespace std;
main() {
//criando tela de grafico
    initwindow(700,600);
    setcolor(YELLOW);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(50,250);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(500,250);
    moveto(250,250);

//VARIABLES DO CODIGO Q REPRESENTA O ESPAÇO!
    float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE];
    int E, x, y,n,t;
    float C, L,T, Wo,I0, Pi;

    // DEFININDO OS MAXIMOS E OS TEXTOS!!
    setcolor( WHITE);
    outtextxy(245, 255, "0");
    // Obs que os C L e E sao constantes atribuidas durante o processo.
    cout<<"[----- VAMOS COMECAR-----]"<<endl;
    cout<<"[----- Este programa gera resultados -----]"<<endl;
    cout<<"[----- Resultados possiveis para filtros Infinitos CASO (ii) -----]"<<endl;
    cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;
    cin>>C;
    cout<<endl;
    cin>>L;
    cout<<endl;
    cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
    cout<<endl;
    cout<<" E para o indutor foi "<<L<<endl;
    cout<<endl;
    cout<<"Certo! agora vamos supor uma fonte de energia, escolha a corrente inicial"<<endl;
    cout<<endl;
    cin>>I0;
    cout<<endl;
    cout<<"tudo pronto vamos comecar"<<endl;
    cout<<endl;

//INICIO DA FUNCAO

// FUNCAO DAS REATANCIAS//

```

```

        // x=w
        Wo= 2/(sqrt(L*C));
        for(x=1; x<=KE; x++)
            { Z1[x] = Wo*L;
              Z2[x] = 1/(Wo*C); }
//escrever as impedancias Z1 e Z2 na tela

        for(x=1; x<=KE; x++) {
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl; }

// funcao de atenuacao alfa B
for (x=1; x<=KE; x++){
    A[x] = acosh(1+(Z1[x]/(2*Z2[x]))); }
// FUNCAO DA CORRENTE NA MALHA N
for (x=1; x<=KE; x++){
    I[x] = I0*((exp(-A[x]*x)*cos(Wo+(2*Pi)))); }
// OBS mude apenas o exp+ e exp-, mas mude o sinal de x para -x apenas na f do grafico
cout<<" Observe que o valor da cosntante de antenuacao Alfa A nas malhas"<<endl;

        for(x=1; x<=KE; x++){
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (40)<<A[x]<<" ";
cout<<endl; }
cout<<"Vamos ao valor da corrente na malha N "<<endl;

//DESENHA O GRAFICO
for(x=1;x<=KE; x++)
{ setcolor(15);
  lineto(250+x*5,250+(I[x]*1000)); }

// salva os dados em txt do resultado do caso (ii) na pasta "pontos F infinito caso (ii)//
ofstream escreve;
escreve.open("pontos F INFINITO caso (ii)");
escreve<<"\n";
for (x=1; x<=KE; x++)
escreve<<-x<<" "<< setiosflags (ios::fixed)<< setprecision (3)<<-I[x]<<endl;
escreve.close();

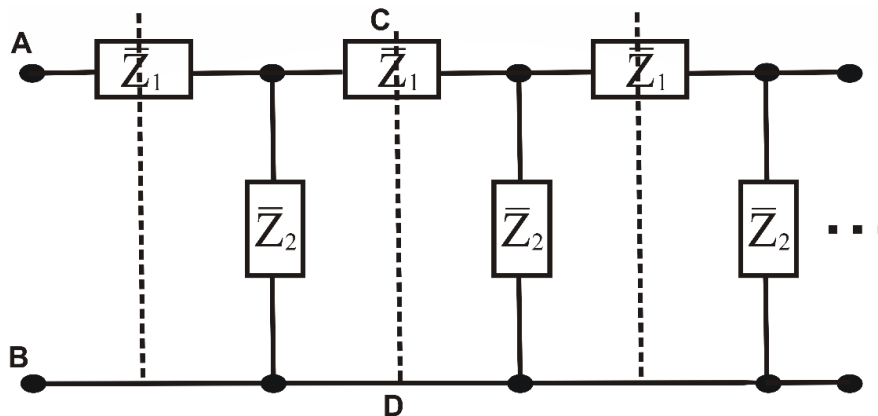
// ESCRIVE NA TELA OS PONTOS!!!
for (x=1; x<=KE; x++)
{cout.width();
cout<<" "<<x<<" "<<" ----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (10)<<I[x]<<" ";
cout.width();
cout<<" A "<<endl; }
cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0;}

```

## 5.4 Impedância iterativa

A partir do princípio que não existem filtros com malhas infinitas, como então aplicar este resultado?

Figura 17 - Redivisão do filtro infinito



Fonte: Própria(2017)

imaginemos a Figura 17, formada pela justaposição de elementos idênticos, obtidos “cortando ao meio” cada seção, ou seja, atribuindo  $1/2 \bar{Z}_1$  a cada lado.

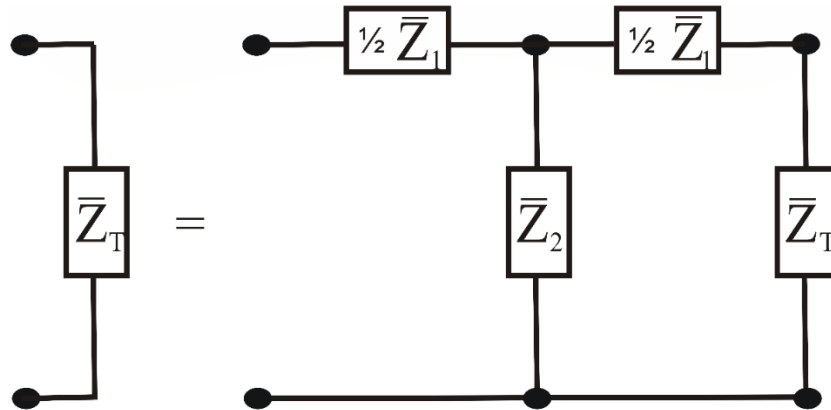
Se propormos um gerador de voltagem  $\bar{V}$  na extremidade esquerda do filtro, entre os pontos A e B a corrente transmitida será (digamos)  $\bar{I}$ . A razão:

$$\frac{\bar{V}}{\bar{I}} = Z_T \quad (5.41)$$

chama-se impedância iterativa ou impedância característica do filtro. Podemos calculá-la através da seguinte lógica.

Iniciamos a partir do ponto de vista de A e B, podemos substituir o filtro pela “impedância equivalente  $Z_T$ ”, mas isso também se aplica (uma vez que ele é semi-infinito), se substituirmos somente a porção a direita dos pontos C e D, por  $Z_T$ . Logo, teremos:

Figura 18 - Substituição das impedâncias por uma total



Fonte: Própria(2017)

Portanto, levando em conta a associação em série e em paralelo, da Figura 18, escrevemos  $Z_T$ , como sendo a equação (5.42):

$$\bar{Z}_T = \frac{1}{2}\bar{Z}_1 + \frac{1}{\frac{1}{\bar{Z}_2} + \frac{1}{\frac{1}{2}\bar{Z}_1 + \bar{Z}_T}} \quad (5.42)$$

o que dá:

$$\bar{Z}_T = \frac{\bar{Z}_1}{2} + \frac{\bar{Z}_2(\frac{\bar{Z}_1}{2} + \bar{Z}_T)}{\frac{\bar{Z}_1}{2} + \bar{Z}_T + \bar{Z}_2} \quad (5.43)$$

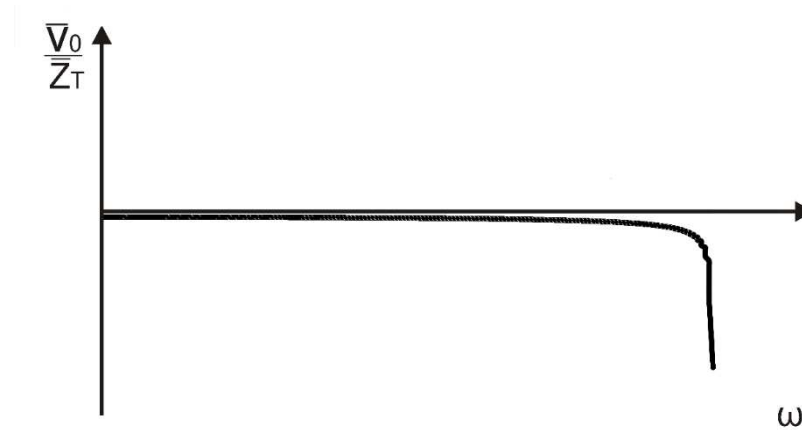
$$\therefore \bar{Z}_T \frac{\bar{Z}_1}{2} + (\bar{Z}_T)^2 + \bar{Z}_T \bar{Z}_2 = \left(\frac{\bar{Z}_1}{2}\right)^2 + \frac{\bar{Z}_1 \bar{Z}_T}{2} + \frac{\bar{Z}_1 \bar{Z}_2}{2} + \frac{\bar{Z}_1 \bar{Z}_2}{2} + \bar{Z}_2 \bar{Z}_T \quad (5.44)$$

$$\Rightarrow \bar{Z}_T = \sqrt{\bar{Z}_1 \bar{Z}_2 + \frac{1}{4}(\bar{Z}_1)^2} + \sqrt{\bar{Z}_1 \bar{Z}_2} \sqrt{1 + \frac{\bar{Z}_1}{4\bar{Z}_2}} \quad (5.45)$$

Se conseguíssemos realizar fisicamente uma impedância desse valor para todas as frequências de  $\omega$ , bastaria empregá-la para terminar o fim de seções do filtro, para que ele se comportasse como se fosse semi-infinito. Na

prática, isto em geral só pode ser feito de forma aproximada, em alguns domínios da frequência, conforme será discutido nas próximas seções.

Gráfico 5.9 – Onda refletindo para origem. Gerada pelo código FI



Quando a impedância terminal se afasta de  $\bar{Z}_T$ , da equação (5.45), é gerada uma onda refletida, que se propaga em sentido oposto. Conforme o Gráfico 5.9.

Tabela 8 - Código FI, substituímos no código as variáveis:  $\bar{Z}_T$  por ZT,  $\bar{Z}_1$  por Z1 e  $\bar{Z}_2$  por Z2

```
#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>

#define KE 1000
using namespace std;

main() {
    //criando tela de grafico
    initwindow(700,600);

    setcolor(YELLOW);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
```

```

        lineto(50,250);
        moveto(250,250);
        lineto(250,250);
        moveto(250,250);
        lineto(500,250);

        moveto(250,250);

//VARIAVEIS DO CODIGO Q REPRESENTA O ESPAÇO!

        float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE],ZT[KE];
        int E, x, y;
        float C, L,T, Wo,I0, Pi, V0, W1, W2;

        // DEFININDO OS MAXIMOS E OS TEXTOS!!
setcolor( WHITE);
outtextxy(245, 255,"0");

        // Obs que os C L e E sao constantes atribuidas durante o processo.

cout<<"[----- VAMOS COMECAR-----]"<<endl;        cout<<"[-----
----- Este programa gera resultados -----]"<<endl;
cout<<"[----- Resultados da impedancia interativa -----]"<<endl;
cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;
        cin>>C;
cout<<endl;
        cin>>L;
cout<<endl;
        cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
        cout<<endl;
        cout<<" E para o indutor foi "<<L<<endl;
cout<<endl;
cout<<"Certo! agora vamos supor uma fonte de energia, escolha a corrente inicial"<<endl;
cout<<endl;
        cin>>V0;
cout<<endl;
        cout<<"tudo pronto vamos comecar"<<endl;
        cout<<endl;
        //INICIO DA FUNCAO
        // C 0.01 e I 0.1 e V 10 volts
        // FUNCAO DAS REATANCIAS//
        Wo=2/(sqrt(L*C));
        for(x=0; x<=KE; x++)
{
        Z1[x] = (Wo*L)/2;
        Z2[x] = (1/(Wo*C)); }
//escrever as impedancias Z1 e Z2 na tela
for(x=0; x<=KE; x++) {
        cout.width();
        cout<<" "<<x<<" "<<" "<<"----> ";
        cout.width();
        cout<<setiosflags (ios::fixed)<< setprecision (10)<<Z1[x]<<" ";
        cout.width();
        cout<<setiosflags (ios::fixed)<< setprecision (10)<<Z2[x]<<" "<<endl;}
// funcao da impedancia ZT na n malha!
        for (x=0; x<=Wo; x++)
{ ZT[x] = ((sqrt(L/C))*(sqrt(1-((pow(x,2))/(pow(Wo,2)))))); }
// FUNCAO DA CORRENTE NA MALHA n
        for (x=0; x<=Wo; x++)
{ I[x] = (V0/ZT[x]); }
cout<<" Observe que o valor da impedancia terminal em funcao de w nas malhas"<<endl;

```

```

// escreve os valores da impedancia para n malhas na TELA
for(x=0; x<=Wo; x++)
    {cout.width();
    cout<<" "<<x<<" "<<" "<<"----> ";
    cout.width();
    cout<<setiosflags (ios::fixed)<< setprecision (4)<<ZT[x]<<" ";
    cout<<endl; }
cout<<"Vamos ao valor da corrente na malha N "<<endl;
// ESCRIBE NA TELA OS PONTOS!!
cout<<" agora observe o vaor da corrente "<<endl;
for (x=0; x<=Wo; x++)
    {cout.width();
    cout<<" "<<x<<" "<<" ----> ";
    cout.width();
    cout<<setiosflags (ios::fixed)<< setprecision (4)<<I[x]<<" ";
    cout.width();
    cout<<" A "<<endl; }
//DESENHA O GRAFICO
for(x=0;x<=Wo; x++)
{ setcolor(15);
  lineto(250+x/4,250+(I[x])); }
// salva os pontos da corrente para um arquivo nomeado "pontos da IMPEDANCIA ITERATIVA
exe1 D W1 - W2 2"//
ofstream escreve;
escreve.open("pontos da IMPEDANCIA ITERATIVA exe1 D W1 - W2 2");
escreve<<"\n";
for (x=0; x<=Wo; x++)
    {escreve<<x/4<<" "<< setiosflags (ios::fixed)<< setprecision (5)<<-I[x]<<endl; }
escreve.close();
cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0; }

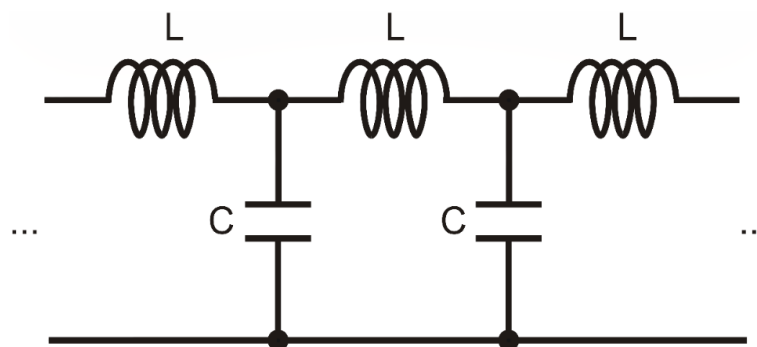
```

Fonte: Própria(2017)

#### 5.4.1 Filtro transmissor de baixa frequência

Para o filtro transmissor de baixa frequência, utilizando os elementos da Figura 14. Assim, teremos um novo filtro, representado na Figura 19.

Figura 19 - Filtro transmissor de baixas frequências



Fonte: Própria(2017)



Estas são as impedâncias para este circuito:

$$\bar{Z}_1 = i\omega L = \bar{Z}_L.$$

E

$$\bar{Z}_2 = \frac{1}{i\omega C} = \bar{Z}_C.$$

onde,

$$\bar{Z}_1 \bar{Z}_2 = L/C \quad (5.46)$$

e:

$$\frac{\bar{Z}_1}{4\bar{Z}_2} = -\frac{1}{4} \omega^2 LC \quad (5.47)$$

ou ainda,

$$\frac{\bar{Z}_1}{4\bar{Z}_2} = -\frac{\omega^2}{\omega_0^2} < 0 \quad (5.48)$$

onde,  $\omega_0$  é expressado em termos da frequência de corte:

$$\omega_0 \equiv \frac{2}{\sqrt{LC}}. \quad (5.49)$$

Assim, a região de passagem (propagação) corresponde a

$$-\frac{\omega^2}{\omega_0^2} > -1 \quad \{ \quad 0 \leq \omega \leq \omega_0. \quad (5.50)$$

Justificando-se o nome (passa as baixas frequências e bloqueia as altas, acima de  $\omega_0$ ), chama-se frequência de corte (é o dobro da frequência ressonância do circuito L-C).

Na região de passagem, o número de onda  $\beta$  é dado por

$$\cos \beta = 1 - 2 \frac{\omega^2}{\omega_0^2} \quad (5.51)$$

de modo que  $\beta$  varia de 0 a  $\pi$  quando  $\omega$  varia de 0 a  $\omega_0$ .

Na região de atenuação ( $\omega > \omega_0$ ), a constante de atenuação  $\alpha$  é dada por

$$\left[ \frac{\bar{Z}_1}{4\bar{Z}_2} = -\frac{\omega^2}{\omega_0^2} < -1, \text{ caso (ii)} \right]$$

E

$$-\text{ch } \alpha = 1 - 2 \frac{\omega^2}{\omega_0^2} \left\{ \text{ch } \alpha = 2 \frac{\omega^2}{\omega_0^2} - 1. \right. \quad (5.52)$$

A impedância interativa é

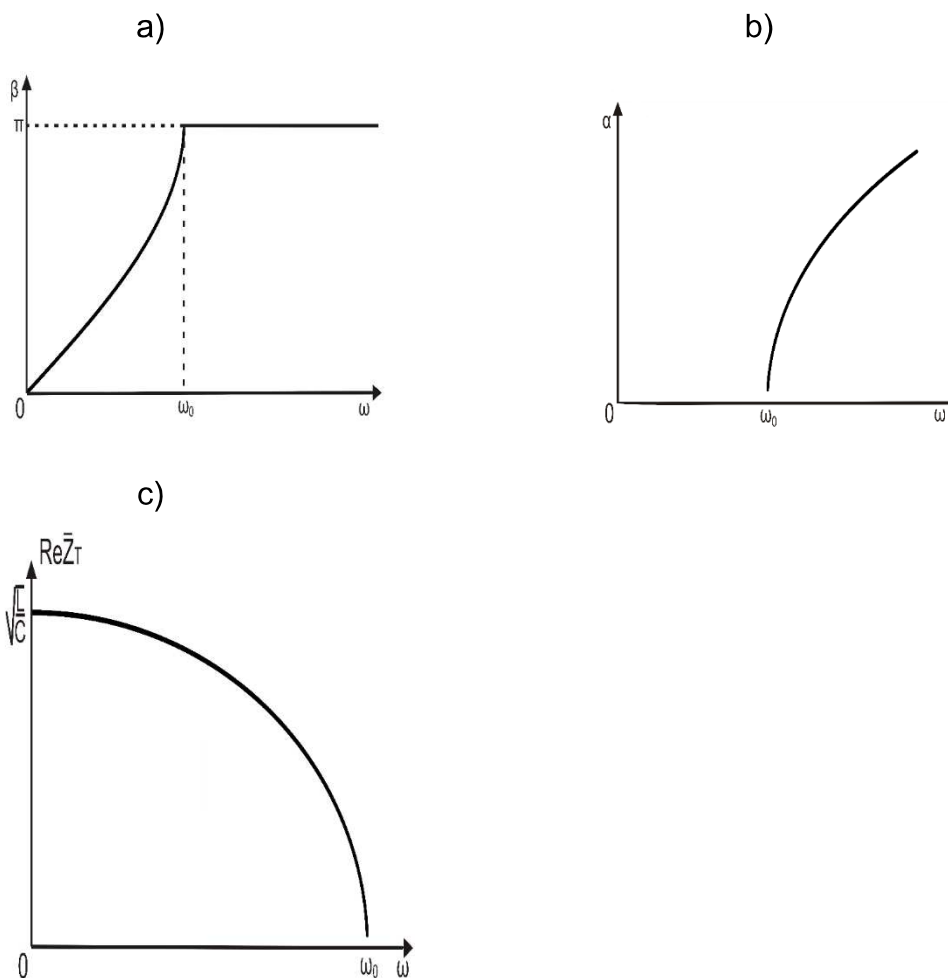
$$\bar{Z}_T = \sqrt{\frac{L}{C} + \frac{1}{4} \omega^2 L^2} = \sqrt{\frac{L}{C}} \sqrt{1 - \frac{\omega^2 CL}{4}} \left\{ \bar{Z}_T = \sqrt{\frac{L}{C} \left(1 - \frac{\omega^2}{\omega_0^2}\right)} \right. \quad (5.53)$$

onde, na banda de passagem  $\bar{Z}_T$  é real. Então, podemos aproximar como uma resistência:

$$R = \sqrt{\frac{L}{C}}$$

quando  $\omega \ll \omega_0$ . Já na banda de atenuação  $\bar{Z}_T$ , teria de comporta-se como uma reatância (imaginária puro), com a dependência da frequência dada pela equação (5.53).

Gráficos 5.10 – a) Ilustra o comportamento de  $\beta$  em função de  $\omega$  b)  $\alpha$  em função de  $\omega$  e c)  $\bar{Z}_T$  em função de  $\omega$



Fonte: Própria(2017)

Tabela 9 - Código que gera o gráfico da defasagem  $\beta$  em função de  $\omega$ . Variáveis substituídas no código foram:  $\omega_0$  por  $W_0$  e  $\beta$  por  $B$

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>
#define KE 1000
using namespace std;

main() {      //criando tela de grafico
              initwindow(700,600);

              setcolor(YELLOW);
              moveto(250,250);
              lineto(250,250);
              moveto(250,250);
              lineto(50,250);
              moveto(250,250);
              lineto(250,250);
              moveto(250,250);
              lineto(500,250);
              moveto(250,250);

              //VARIABLES DO CODIGO Q REPRESENTA O ESPAÇO!
              float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE];
              int E, x, y,n,t, PASSOS;
              float C, L,T, Wo,l0, Pi;

              // DEFININDO OS MAXIMOS E OS TEXTOS!!
              setcolor( WHITE);
              outtextxy(245, 255, "0");

              // Obs que os C L e E sao constantes atribuidas durante o processo.
              cout<<"[----- VAMOS COMECAR-----]"<<endl;
              cout<<"[----- Este programa gera resultados -----]"<<endl;
              cout<<"[----- Resultados possiveis para a defasagem B-----]"<<endl;
              cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;
              cin>>C;
              cout<<endl;

              cin>>L;
              cout<<endl;

              cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
              cout<<endl;
              cout<<" E para o indutor foi "<<L<<endl;
              cout<<endl;
              cout<<"Certo! agora vamos supor uma fonte de energia, escolha a corrente inicial"<<endl;
              cout<<endl;

              cin>>l0;
              cout<<endl;

              cout<<"tudo pronto vamos comecar"<<endl;
              cout<<endl;

```

```

//INICIO DA FUNCAO
// FUNCAO DAS IMPEDANCIAS//
// C 0.01 e L 0.001

    Wo=(2/(sqrt(L*C)));

        for(x=0; x<=KE; x++)
        { Z1[x] = x*L;
          Z2[x] = 1/(x*C); }

//escrever as impedancias Z1 e Z2 na tela
    for(x=0; x<=Wo; x++) {
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
    cout.width();
    cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
    cout.width();
    cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl; }

    // funcao de defasagem beta B
    for (x=0; x<=Wo; x++)
{ B[x] = acos(1-(2*(pow(x*2,2)/pow(Wo,2)))); }

// ESCREVE NA TELA OS PONTOS!!

cout<<"Valor da defasagem em funcao de w "<<endl;
for (x=0; x<=Wo; x++)
{ cout.width();
  cout<<" "<<x<<" "<<" ----> ";
  cout.width();
  cout<<setiosflags (ios::fixed)<< setprecision (10)<<B[x]<<" ";
  cout.width();
  cout<<" "<<endl; }

//DESENHA O GRAFICO

for(x=0;x<=Wo; x++)
{ setcolor(15);
  lineto(250+(x/4),250-(B[x]*20)); }

//Funcao exporta os pontos txt p a pasta "pontos de beta em funcao de w"

ofstream escreve;

escreve.open("pontos de beta em funcao de w");
escreve<<"""\n";

for (x=0; x<=Wo; x++)
    escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (5)<<B[x]<<endl;
escreve.close();
cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0;}

```

Fonte: Própria(2017)

Tabela 10 - Código que gera a constante de atenuação  $\alpha$  em função de  $\omega$ , no código substituímos  $\alpha$  por A

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>

#define KE 1000
using namespace std;

main() {

//criando interface para o grafico
    initwindow(700,600);

    setcolor(YELLOW);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(50,250);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(500,250);

    moveto(250,250);
//VARIÁVEIS DO CÓDIGO Q REPRESENTA O ESPAÇO//
    float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE];
    int E, x, y;
    float C, L,T, Wo,I0, Pi;

    // DEFININDO OS MÁXIMOS E OS TEXTOS!!

    setcolor( WHITE);
    outtextxy(245, 255,"0");

    // Obs que os valores de C e L sao constantes atribuidas durante o processo.

    cout<<"[----- VAMOS COMECAR-----]"<<endl;
    cout<<"[----- Este programa gera resultados -----]"<<endl;
    cout<<"[----- Resultados possiveis para a const de antenucao -----]"<<endl;
    cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;

    cin>>C;
    cout<<endl;

    cin>>L;
    cout<<endl;

    cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
    cout<<endl;
    cout<<" E para o indutor foi "<<L<<endl;
    cout<<endl;

```

```

cout<<"Certo! agora vamos supor uma fonte de energia, escolha a corrente inicial"<<endl;
cout<<endl;

    cin>>I0;

cout<<endl;

    cout<<"tudo pronto vamos comecar"<<endl;
    cout<<endl;

    //INICIO DA FUNCAO
    // FUNCAO DAS IMPEDANCIAS//
    // C 0.001 e L 0.01

    Wo=(2/(sqrt(L*C)));

    for(x=0; x<=KE; x++)
    { Z1[x] = x*L;
      Z2[x] = 1/(x*C); }

//escrever as impedancias Z1 e Z2 na tela//
    for(x=0; x<=KE; x++) {

cout.width();
    cout<<" "<<x<<" "<<" "<<"----> ";
    cout.width();
    cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
    cout.width();
    cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl; }

    // funcao da const de atenuacao alfa A
    for (x=0; x<=Wo; x++)
    { A[x] = acosh((((pow(x*2,2)/pow(Wo,2)))^2)-1); }

// ESCREVE NA TELA OS PONTOS!!
cout<<"Valor da constante de atenuacao ALFA em funcao de w "<<endl;
for (x=0; x<=Wo; x++)
{
    cout.width();
    cout<<" "<<x<<" "<<" ----> ";
    cout.width();
    cout<<setiosflags (ios::fixed)<< setprecision (10)<<A[x]<<" ";
    cout.width();
    cout<<" "<<endl; }

//DESENHA O GRAFICO
for(x=0;x<=Wo; x++) {
  setcolor(15);
  lineto(250+(x/4),250-(A[x]*20)); }

//Funcao exporta os pontos txt para o arquivo "pontos de ALFA em funcao de w"
ofstream escreve;
escreve.open("pontos de ALFA em funcao de w");
escreve<<"\n";

for (x=0; x<=Wo; x++)
    escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (45)<<A[x]<<endl;
escreve.close();

cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0;}

```

Tabela 11 - Código que gera a parte real da impedância total  $\bar{Z}_T$  em função de  $\omega$ , no código substituímos  $\bar{Z}_T$  por ZT

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>

#define KE 1000
using namespace std;

main() {
//criando tela de gráfico//
    initwindow(700,600);
    setcolor(YELLOW);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(50,250);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(500,250);

    moveto(250,250);

//VARIAVEIS DO CODIGO Q REPRESENTA O ESPAÇO!
    float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE],ZT[KE];
    int E, x, y,n,t, PASSOS;
    float C, L,T, Wo,I0, Pi;

    // DEFININDO OS MAXIMOS E OS TEXTOS!!

setcolor( WHITE);
outtextxy(245, 255,"0");

Obs que os C L e E sao constantes atribuidas durante o processo.

    cout<<"[----- VAMOS COMECAR-----]"<<endl;
    cout<<"[----- Este programa gera resultados -----]"<<endl;
    cout<<"[----- Resultados da impedancia total ZT -----]"<<endl;
    cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;

    cin>>C;
cout<<endl;
    cin>>L;
cout<<endl;

    cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
    cout<<endl;
    cout<<" E para o indutor foi "<<L<<endl;

cout<<endl;
cout<<"Certo! agora vamos supor uma fonte de energia, escolha a corrente inicial"<<endl;

```



```

cout<<endl;

        cin>>I0;

cout<<endl;

        cout<<"tudo pronto vamos comecar"<<endl;
        cout<<endl;

        //INICIO DA FUNCAO
        // FUNCAO DAS IMPEDANCIAS//
        // C 0.001 e L 0.01

        Wo=(2/(sqrt(L*C)));

                for(x=0; x<=Wo; x++)
{ Z1[x] = x*L;
  Z2[x] = 1/(x*C); }

//escrever as impedancias Z1 e Z2 na tela

        for(x=0; x<=Wo; x++)
{ cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl; }

        // funcao de impedancia total ZT//
        for (x=0; x<=Wo; x++)
{ ZT[x] = sqrt((L/C)*(1-(pow(x,2)/pow(Wo,2)))); }

// ESCREVE NA TELA OS PONTOS!!
cout<<"Valor da banda de atenuacao ZT em funcao de w "<<endl;
for (x=0; x<=Wo; x++)
{
cout.width();
cout<<" "<<x<<" "<<" ----> ";
cout.width();
cout<<setiosflags (ios::fixed)<< setprecision (10)<<ZT[x]<<" ";
cout.width();
cout<<" "<<endl; }

//DESENHA O GRAFICO
for(x=0;x<=Wo; x++)
{ setcolor(15);
lineto(250+(x/4),250-(ZT[x]*20)); }
//Funcao exporta os pontos txt no arquivo "pontos de ZT em funcao de w"
ofstream escreve;
escreve.open("pontos de ZT em funcao de w");
escreve<<"\n";
for (x=0; x<=Wo; x++)
escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (5)<<ZT[x]<<endl;
escreve.close();
cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0;}

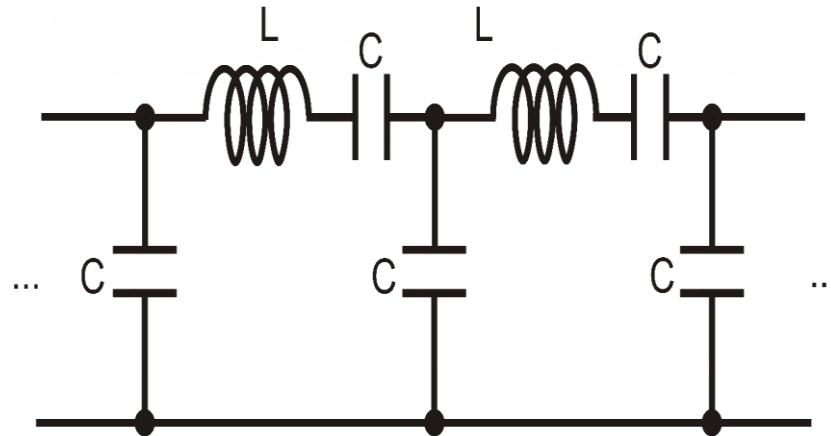
```

Fonte: própria(2017)

### 5.4.2 Filtro transmissor de banda

Na Figura 20 ilustramos um filtro transmissor de banda, onde usamos uma associação mista.

Figura 20 - Filtro transmissor de banda



Fonte: Própria(2017)

As impedâncias para este filtro são:

$$\bar{Z}_1 = i\omega L - \frac{i}{\omega C} = i\omega L \left(1 - \frac{1}{\omega^2 LC}\right) \quad (5.54)$$

E

$$\bar{Z}_2 = -\frac{i}{\omega C} \left\{ \frac{1}{\bar{Z}_2} \right\} = i\omega C \quad (5.55)$$

o que dá

$$\frac{\bar{Z}_1}{4\bar{Z}_2} = -\frac{1}{4} \omega^2 LC \left(1 - \frac{1}{\omega^2 LC}\right) = \frac{1}{4} (1 - \omega^2 LC) \quad (5.56)$$

a equação (5.56) é  $> 0$ , para  $\omega = 0$  [ caso (i) de atenuação ] e  $\rightarrow -\infty$  para  $\omega \rightarrow \infty$  [ caso (ii) de atenuação ]. Assim,

$$\frac{\bar{Z}_1}{4Z_2} < 0 \text{ para } \omega > \omega_1 \equiv \frac{1}{\sqrt{LC}}. \quad (5.57)$$

A equação (5.57) passa pelo valor -1, quando:

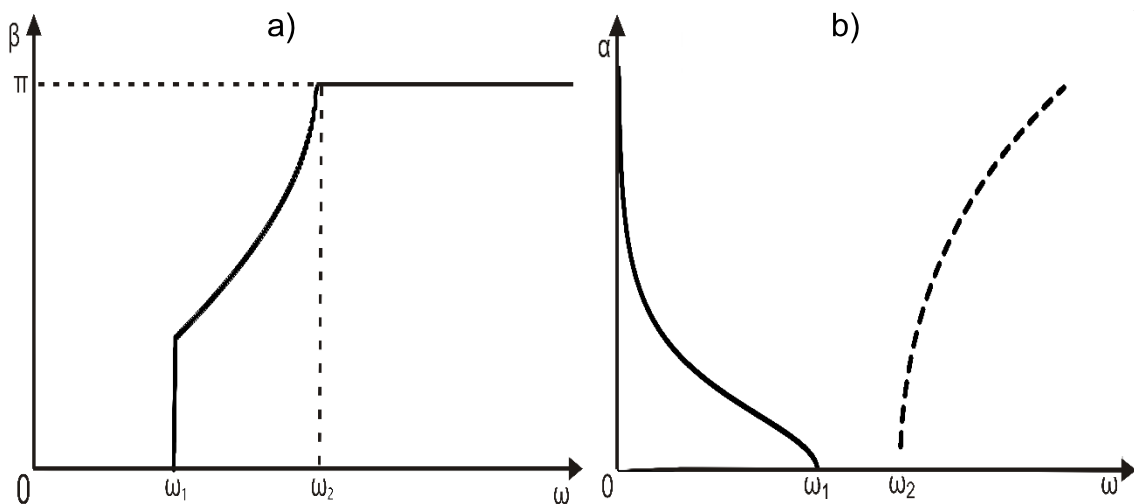
$$\frac{1}{4} (1 - \omega^2 LC) = -1 \{ \omega^2 LC = 4 + 1 = 5 \} \omega = \omega_2 \equiv \frac{\sqrt{5}}{\sqrt{LC}} \quad (5.58)$$

a banda de passagem, nesse caso, são dadas pela equação (5.59)

$$\frac{1}{\sqrt{LC}} \equiv \omega_1 \leq \omega \leq \omega_2 \equiv \frac{\sqrt{5}}{\sqrt{LC}}. \quad (5.59)$$

E as frequências fora desta banda são atenuadas.

Gráficos 5.11 - a) ilustra o comportamento de  $\beta$ , em um domínio de  $\omega$  e b) comportamento de atenuação  $\alpha$  para  $\omega_1$  e  $\omega_2$



Fonte: Própria(2017)

O andamento de  $\beta$  e  $\alpha$  ilustrados no Gráfico 5.11. Notamos que a defasagem permanece igual à  $\pi$ , acima de  $\omega_2$  [caso (ii)].

Tabela 12 – Código da função de defasagem  $\beta$  para as condições de  $\omega_1$  e  $\omega_2$ , substituímos no código  $\omega_1$  por W1 e  $\omega_2$  por W2

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>

#define KE 1000
using namespace std;

main() {

//criando tela de grafico
    initwindow(700,600);

    setcolor(YELLOW);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(50,250);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(500,250);

    moveto(250,250);

//VARIABLES DO CODIGO Q REPRESENTA O ESPAÇO!
    float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE];
    int E, x, y,n,t, PASSOS;
    float C, L,T, Wo,I0, Pi, W1, W2;

    // DEFININDO OS MAXIMOS E OS TEXTOS!!

setcolor( WHITE);
outtextxy(245, 255,"0");

    // Obs que os C L e E sao constantes atribuidas durante o processo.

    cout<<"[----- VAMOS COMECAR-----]"<<endl;
    cout<<"[----- Este programa gera resultados -----]"<<endl;
    cout<<"[----- Resultados possiveis para a defasagem B-----]"<<endl;
    cout<<"[----- Agora escolha os valores para os elementos C e L -----]"<<endl;

    cin>>C;
cout<<endl;

    cin>>L;
cout<<endl;

    cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
    cout<<endl;
    cout<<" E para o indutor foi "<<L<<endl;

```

```

cout<<endl;
        cout<<"tudo pronto vamos comecar"<<endl;
        cout<<endl;

        //INICIO DA FUNCAO
        // FUNCAO DAS IMPEDANCIAS//
        // C 0.001 e L 0.01

        Wo=(2/(sqrt(L*C)));
        W1= (1/(sqrt(L*C)));
        W2= ((sqrt(5))/(sqrt(L*C)));

        for(x=1; x<=KE; x++)

        { Z1[x] = (Wo*L)*(1-(1/(pow(Wo,2)*L*C)));
          Z2[x] = -(1/(Wo*C)); }

//escrever as impedancias Z1 e Z2 na tela

        for(x=1; x<=KE; x++)
{ cout.width();
  cout<<" "<<x<<" "<<" "<<"----> ";
  cout.width();
  cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
  cout.width();
  cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl; }

// funcao de defasagem beta B //
for (x=W1; x<=W2; x++)
{ B[x] = acos(1-(2*(pow(x,2)/pow(W1,2)))); }

// ESCREVE NA TELA OS PONTOS!!

cout<<"Valor da defasagem em funcao de w "<<endl;
for (x=1; x<=KE; x++)
{ cout.width();
  cout<<" "<<x<<" "<<"----> ";
  cout.width();
  cout<<setiosflags (ios::fixed)<< setprecision (10)<<B[x]<<" ";
  cout.width();
  cout<<" "<<endl; }

//DESENHA O GRAFICO
for(x=1;x<=W2; x++)
{ setcolor(15);
  lineto(250+(x/4),250-(B[x])*50); }

//Funcao exporta os pontos txt no arquivo "pontos de beta EX C w1 e w2 em funcao de w"//
ofstream escreve;

  escreve.open("pontos de beta EX C w1 e w2 em funcao de w");
  escreve<<"\n";
  for (x=1; x<=KE; x++)
    escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (5)<<B[x]<<endl;
  escreve.close();
cout<<"[-----CONCLUIDO!-----]"<<endl;
getch();
closegraph();
return 0; }

```

Tabela 13 - Código gera a constante de atenuação  $\alpha$ , para a condição  $\omega_1 \ll \omega$ 

```

#include <graphics.h>
#include <limits.h>
#include <dos.h>
#include <iostream>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <fstream>
#include <iomanip>
#include <string>

#define KE 1000
using namespace std;

main() {
//criando tela de grafico
    initwindow(700,600);

    setcolor(YELLOW);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(50,250);
    moveto(250,250);
    lineto(250,250);
    moveto(250,250);
    lineto(500,250);
    moveto(250,250);

//VARIABLES DO CODIGO Q REPRESENTA O ESPAÇO!
    float W[KE], V[KE], B[KE], Z1[KE], Z2[KE], I[KE], Ph[KE],w[KE], A[KE];
    int E, x, y,n,t, PASSOS;
    float C, L,T, Wo,I0, Pi, W1, W2;

    // DEFININDO OS MAXIMOS E OS TEXTOS!!

setcolor( WHITE);
outtextxy(245, 255,"0");
    cout<<"[----- VAMOS COMECAR-----]"<<endl;
    cout<<"[----- Este programa gera resultados -----]"<<endl;
    cout<<"[----- Resultados possiveis atenuacao alfa W1-----]"<<endl;
    cout<<"[---- Agora escolha os valores para os elementos C e L ----]"<<endl;

        cin>>C;
    cout<<endl;
        cin>>L;
    cout<<endl;

        cout<<"OK os valores escolhidos para o capacitor foi "<<C<<endl;
        cout<<endl;
        cout<<" E para o indutor foi "<<L<<endl;
    cout<<endl;

        cout<<"tudo pronto vamos comecar"<<endl;
        cout<<endl;

//INICIO DA FUNCAO
// FUNCAO DAS IMPEDANCIAS//
// C 0.001 e L 0.01

```

```

        Wo=(2/(sqrt(L*C)));
        W1= (1/(sqrt(L*C)));
        W2= ((sqrt(5))/(sqrt(L*C)));

        for(x=1; x<=KE; x++) {
            Z1[x] = (x*L)*(1-(1/(pow(x,2)*L*C)));
            Z2[x] = -(1/(x*C)); }

//escrever as impedancias Z1 e Z2 na tela//
        for(x=1; x<=KE; x++) {
cout.width();
cout<<" "<<x<<" "<<" "<<"----> ";
        cout.width();
            cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z1[x]<<" ";
            cout.width();
            cout<<setiosflags (ios::fixed)<< setprecision (4)<<Z2[x]<<" "<<endl;
        }
// funcao de alfa para w1
        for (x=1; x<=W1; x++)
            { A[x] = acosh((2*(pow(W1,2)/pow(x,2)))-1); }

// ESCREVE NA TELA OS PONTOS!!

        cout<<"Valor da defasagem em funcao de w "<<endl;
        for (x=1; x<=W1; x++)
            {
                cout.width();
                cout<<" "<<x<<" "<<" ----> ";
                cout.width();
                cout<<setiosflags (ios::fixed)<< setprecision (10)<<A[x]<<" ";
                cout.width();
                cout<<" "<<endl; }

//DESENHA O GRAFICO//
        for(x=1;x<=W1; x++)
        {
            setcolor(15);
            lineto(250+(x/4),250-(A[x]*50)); }
//Funcao exporta os pontos txt no arquivo "pontos de ALFA EX C W1 em funcao de w"
ofstream escreve;
        escreve.open("pontos de ALFA EX C W1 em funcao de w");
        escreve<<"""\n";

        for (x=1; x<=W1; x++)
            escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (5)<<A[x]<<endl;
        escreve.close();
        cout<<"[-----CONCLUIDO!-----]"<<endl;
        getch();
        closegraph();
        return 0; }

```

Fonte: própria(2017)

E para escrever a função  $\alpha$  para a condição  $\omega \ll \omega_2$ , apenas mudamos as condições no laço for do código. Observe a Tabela 14, a seguir.

Tabela 14 - Código  $\omega \gg \omega_2$  em função de  $\alpha$ 

```

// funcao de antenuacao alfa p/ w2 //
  for (x=W2; x<=KE; x++)
    {
      A[x] = acosh((2*(pow(x,2)/pow(W2,2)))-1);
    }
  // ESCREVE NA TELA OS PONTOS!!

      cout<<"Valor da antenuacao em funcao de w "<<endl;
  for (x=W2; x<=KE; x++)
    {
      cout.width();
      cout<<" "<<x<<" "<<" ---> ";
      cout.width();
      cout<<setiosflags (ios::fixed)<< setprecision (10)<<A[x]<<" ";
      cout.width();
      cout<<" "<<endl;
    }

  //DESENHA O GRAFICO

  for(x=W2;x<=KE; x++)

  { setcolor(15);
    lineto(250+(x/4),250-(A[x]*50)); }
  //Funcao exporta os pontos txt no arquivo "pontos de ALFA EX C W2 em funcao de w"

  ofstream escreve;

  escreve.open("pontos de ALFA EX C W2 em funcao de w");
  escreve<<"""\n";

  for (x=W2; x<=KE; x++)
  { escreve<<x<<" "<< setiosflags (ios::fixed)<< setprecision (5)<<A[x]<<endl; }
  escreve.close();

```

Fonte: própria(2017)

## 5.5 Aplicações dos filtros

São dadas conforme a sua classificação. Como citaremos a seguir, existem um número grande de aplicações, tais como:

- Filtros passa baixa - São circuitos RC, RL, CL em série ou não, e podem ter aplicação na seleção de faixas de áudio, eliminando determinadas frequências e possibilitando a saída de áudio puro, tornando o áudio agudo e grave sem ruídos.
- Filtros Bloqueia alta - são filtros de linha, também chamados popularmente de “réguas”. São dispositivos equipados com fusíveis, varistores, capacitores e indutores. O objetivo deste



equipamento é evitar a passagem de altas correntes para os aparelhos nele conectados. Quando isso ocorre, o fusível “queima”, ou seja, corta a energia que alimenta o filtro.

- Num aparelho de som de alta fidelidade, o ruído devido a rotação do motor de um toca-discos, de baixa frequência, perturba o sinal proveniente da gravação, este sinal pode ser eliminado (atenuado) por um filtro transmissor de alta frequência.
- Num cabo telefônico, pode-se usar um filtro transmissor de banda para selecionar somente canais usados para transmissão da voz.

Como já mencionado, as aplicações para esses tipos de filtros são as mais diversas. Elas são criadas a partir de associações de capacitores e indutores, causando no circuito mudanças nas características de transmissão da energia em seus terminais. Contudo, são combinações simples com uma enorme quantidade de opções para a construção desses filtros. Notamos, que boa parte desses filtros nem foram ainda aplicados. Buscamos interpretar este fenômeno através de aplicações computacionais, no intuito de despertar o interesse neste tema de futuros estudantes e ampliar as discursões.

## 6. CONCLUSÃO

Neste trabalho apresentamos uma contribuição direta na construção computacional de dispositivos eletrônicos conhecidos como filtros em baixas frequências. A compreensão desses filtros, bem como a utilização da computação na construção de uma resposta gráfica que identifique a classificação do filtro que, representa uma grande vantagem do ponto de vista tecnológico, pois potencializa a aplicação desses filtros em aplicações tecnológicas diversas. Apresentamos respostas do espectro da frequência dos filtros teóricos apresentado no capítulo 10, do volume 3, de física básica do Moysés, contribuindo para uma melhor visualização da classificação dos filtros.

A linguagem de programação computacional pode ser uma ferramenta extremamente útil a discentes que estejam interessados em ampliar seus conhecimentos, planejando novas formas de aprendizagem, que permitem ampliar seus conhecimentos teóricos, obtidos no decorrer do curso de graduação. Além disso, o discente que aprende a programação computacional, enriquece o seu currículo.

Implementamos diversos códigos, dos quais descrevemos na íntegra neste trabalho de conclusão de curso, possibilitando aos discentes compreender como fazer e visualizar o comportamento do filtro. Com este recurso, podemos visualizar completamente os limites e as condições que Moysés cita para o funcionamento dos filtros CL, LC, tanto para o caso ideal quando para o caso real. Demonstramos também o comportamento da defasagem de corrente e das constantes de atenuação em função da frequência  $\omega$ , para cada filtro e problema investigado.

Por fim, é possível que este trabalho possibilite novas pesquisas em filtros seletores de baixa frequência, bem como: separar o sinal de uma onda modulada, bloquear o sinal de WIFI, ou selecionar faixa de rádio de transmissão.

## BIBLIOGRAFIA

Bóson Treinamento em Tecnologias, **Associação em Série e em Paralelo de Capacitores**. Disponível em: <[http://www.bosontreinamentos.com.br/eletronica/curso-deeletronica/as\\_sociacoes-em-serie-e-em-paralelo-de-capacitores/](http://www.bosontreinamentos.com.br/eletronica/curso-deeletronica/as_sociacoes-em-serie-e-em-paralelo-de-capacitores/)>. Acesso em 26 de agosto de 2017.

Brasil Escola, **Associação de capacitores**. Disponível em: <<http://brasil.escola.uol.com.br/fisica/associacao-capacitores.htm>>. Acesso em 20 de agosto de 2017.

Brasil Escola, **Capacitor**. Disponível em: <<http://brasilecola.uol.com.br/fisica/capacitores.htm>>. Acesso em 20 de agosto de 2017.

DIAS V. F. **Filtros Elétricos**. Disponível em: <[http://www.ufrgs.br/eng04030/Aulas/teoria/cap\\_12/filtelec.htm](http://www.ufrgs.br/eng04030/Aulas/teoria/cap_12/filtelec.htm)>. Acesso em 20 de Abril de 2017.

LIPPMAN, Stanley B.; LAJOIE, José Lajoie, MOO, Barbara E. **C + + Primer**. 4<sup>o</sup> edição. New Jersey: Addison Wesley, 2005.

Mundo da Elétrica, **Tipos de Capacitores**. Disponível em: <<https://www.mundo-da-eletrica.com.br/tipos-de-capacitores/>>. Acesso em 22 de agosto de 2017.

Mundo da Elétrica, **O que é um indutor**. Disponível em: <<https://www.mundo-daeletrica.com.br/o-que-e-um-indutor/1>>. Acesso em 25 de agosto de 2017.

Mundo da Elétrica, **Tipos de indutores**. Disponível em: <<https://www.mundo-daeletrica.com.br/o-que-e-um-indutor/1>>. Acesso em 25 de agosto de 2017.

PRATA, Stephen. **C + + Primer Plus**. 4<sup>o</sup> edição. Nova York: Sams Publishing, 2001.

Ponto Ciência, **Capacitor Eletrolítico**. Disponível em: <<http://www.ponto-ciencia.org.br/experimentos/visualizar/capacitor-eletrolitico/711>>. Acesso em 24 de agosto de 2017.

H. Moysés Nussenzveig. **Curso de Física Básica 3: Eletromagnetismo**. São Paulo: Edgard Blucher, 1997.

H. Moysés Nussenzveig. **Curso de Física Básica 2: Fluidos Oscilações e Ondas Calor**. 4<sup>o</sup> edição. São Paulo: Edgard Blucher, 2002.

H. Moysés Nussenzveig. Circuitos. In: H. Moysés Nussenzveig. **Curso de Física Básica 3: Elementos de circuitos**. São Paulo: Addison Wesley, 2009. p. 181-185.

H. Moysés Nussenzveig. Circuitos. In: \_\_\_\_\_. **Curso de Física Básica 3: Leis de Kirchhoff**. São Paulo: Edgard Blucher, 1997. p. 191-193.

H. Moysés Nussenzveig. Circuito. In: \_\_\_\_\_. **Curso de Física Básica 3: Filtros**. São Paulo: Edgard Blucher, 1997. p. 217-227.

H. Moysés Nussenzveig. Corrente elétrica. In: \_\_\_\_\_. **Curso de Física Básica 3: Conservação de carga e equação da continuidade**. São Paulo: Edgard Blucher, 1997. p. 101-103.

H. Moysés Nussenzveig. Corrente elétrica. In: \_\_\_\_\_. **Curso de Física Básica 3: Lei de Ohm e condutividade**. São Paulo: Edgard Blucher, 1997. p. 103-106.

H. Moysés Nussenzveig. CAPACITANCIA E CAPACITORES. DIÉLETRICOS. In: \_\_\_\_\_. **Curso de Física Básica 3**: Associação de capacitores. São Paulo: Edgard Blucher, 1997. p. 79-80.

H. Moysés Nussenzveig. Ondas. In: \_\_\_\_\_. **Curso de Física Básica 2**: Ondas em uma dimensão. 4ª edição. São Paulo: Edgard Blucher, 2002. p. 99-102.

SEARS E ZEMANSKY. Campo magnético e força magnética. In: SEARS E ZEMANSKY. **FÍSICA III**: Seletores de velocidades. 12ª Edição. São Paulo: PEARSON, 2009. p. 211-217.

SEARS E ZEMANSKY. Fótons, Elétrons e Átomos. In: \_\_\_\_\_. **FÍSICA IV**: Efeito fotoelétrico. 12ª Edição. São Paulo: Edgard Blucher, 1997. p. 189-191.

SEARS E ZEMANSKY. Leis de Kirchhoff. In: \_\_\_\_\_. **FÍSICA II**: Leis dos nós, lei das malhas. 12ª Edição. São Paulo: PEARSON, 2009. p. 173-174.

STROUSTRUP, Bjarne. **O C + + Programming Language**. 3ª edição. New Jersey: Addison Wesley. 1997.

TORRES, Jair Gustavo M. **Curso de Linguagem C**. Ilha Solteira: Faculdade de Engenharia de Ilha Solteira - FEIS – UNESP. Departamento de Engenharia Elétrica, 2006.

## APÊNDICE

### Tutorial C++

Para começarmos a escrever as equações demonstradas no capítulo 10.9 do livro vol. 3, do curso de física básica iii de H. Moysés Nussenzveig, onde desenvolvemos os filtros transmissores em baixa frequência, é necessário ter o código fonte pronto, tanto para a parte de operações matemáticas para as ilustrações gráficas, com esse código pronto, necessariamente precisaremos fazer pequenas modificações para os próximos códigos. Segue as etapas para instalações do compilador DEVE-C++ e das bibliotecas que iram executar as funções, sendo estas bibliotecas instaladas dentro do deve-C++.

### Download da Apostila

Antes de começar a compilar os códigos é necessário aprender o básico de programação em C/C++, várias apostilas estão disponíveis na página de pesquisa google, tanto para abrir ou salvar as apostilas completa, mas é melhor baixar em formato pdf, pronto para leitura off-line, em seguida prepare-se estudando o material.

### Compilador e Editor Dev-C++

Para baixar o compilador do Dev-C++, apenas digite no google “Download do Dev-C++” e aparecera várias opções para baixar este software.

### Lista de Bibliotecas gráficas:

1. windows.h
2. openGL
3. conio 2.0
4. dos.h
5. cmath
6. fstream

7. string
8. graphics.h
9. iomanip
10. iostream

Estas bibliotecas são encontradas em alguns sites de download, o site onde mais encontra opções é “SourceForge.net”, é necessário assistir vídeos no site “you tube” ou sugestões de instalação de bibliotecas no compilador Dev-C++, disponível em “Forum Script Brasil” e em “Clube do Hardware”.