

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
GRADUAÇÃO NO CURSO DE ENGENHARIA ELÉTRICA**

**FERRAMENTA COMPUTACIONAL PARA
SÍNTESE DE FILTROS ANALÓGICOS E
DIGITAIS**

TRABALHO DE GRADUAÇÃO

Renan Birck Pinheiro

Santa Maria, RS, Brasil

2015

FERRAMENTA COMPUTACIONAL PARA SÍNTESE DE FILTROS ANALÓGICOS E DIGITAIS

Renan Birck Pinheiro

Trabalho de Graduação apresentado ao curso de Engenharia Elétrica da
Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para
a obtenção do grau de
Engenheiro Eletricista

Orientador: Prof. Dr. Fulano de Tal

Santa Maria, RS, Brasil

2015

Pinheiro, Renan Birck

Ferramenta Computacional para Síntese de Filtros Analógicos e Digitais / por Renan Birck Pinheiro. – 2015.

21 f.: il.; 30 cm.

Orientador: Fulano de Tal

Monografia (Graduação) - Universidade Federal de Santa Maria, Centro de Tecnologia, curso de Engenharia Elétrica, RS, 2015.

1. Filtros eletrônicos. 2. Ferramenta computacional. 3. Projeto Eletrônico. I. de Tal, Fulano. II. Título.

**Universidade Federal de Santa Maria
Centro de Tecnologia
Graduação no curso de Engenharia Elétrica**

A Comissão Examinadora, abaixo assinada,
aprova o Trabalho de Graduação

**FERRAMENTA COMPUTACIONAL PARA SÍNTESE DE FILTROS
ANALÓGICOS E DIGITAIS**

elaborado por
Renan Birck Pinheiro

como requisito parcial para obtenção do grau de
Engenheiro Eletricista

COMISSÃO EXAMINADORA:

Fulano de Tal, Dr.
(Orientador)

Fulano De Tal, Dr. (UFSM)

Sicrano De Tal, Dr. (UFSM)

Santa Maria, XX de junho de 2015.

AGRADECIMENTOS

À minha família e amigos pelo apoio e incentivo durante minha trajetória no curso.

Ao professor Fulano de Tal, por ter me orientado na execução deste trabalho.

Aos meus colegas de trabalho na Chip Inside Tecnologia, que colaboraram na revisão deste trabalho e sugeriram correções e modificações.

“Em algum lugar, alguma coisa incrível está esperando para ser conhecida..”

— CARL SAGAN

RESUMO

Trabalho de Graduação
Graduação no curso de Engenharia Elétrica
Universidade Federal de Santa Maria

FERRAMENTA COMPUTACIONAL PARA SÍNTESE DE FILTROS ANALÓGICOS E DIGITAIS

AUTOR: RENAN BIRCK PINHEIRO

ORIENTADOR: FULANO DE TAL

Local da Defesa e Data: Santa Maria, XX de junho de 2015.

Blá blá blá.

Palavras-chave: Filtros eletrônicos. Ferramenta computacional. Projeto Eletrônico.

ABSTRACT

Undergraduate Final Work
Electrical Engineer
Federal University of Santa Maria

DEVELOPMENT OF A SOFTWARE TOOL FOR ANALOG AND DIGITAL FILTER DESIGN

AUTHOR: RENAN BIRCK PINHEIRO

ADVISOR: FULANO DE TAL

Defense Place and Date: Santa Maria, January XXst, 2015.

This work. . .

Keywords: Foo, Bar, Baz.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ANEXOS

LISTA DE ABREVIATURAS E SIGLAS

DSP	<i>Digital Signal Processing</i>
FFT	<i>Fast Fourier Transform</i>

SUMÁRIO

1 INTRODUÇÃO	14
1.1 Motivação	14
1.2 Objetivos	14
1.3 Estrutura do trabalho	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 Filtros Analógicos	16
2.1.1 Famílias de filtros analógicos	16
2.1.1.1 Filtro de Butterworth	16
2.1.1.2 Filtro de Bessel	16
2.1.1.3 Filtro Elíptico	16
2.1.1.4 Filtro Chebyshev, tipo 1	16
2.1.1.5 Filtro Chebyshev, tipo 2	16
2.1.2 Implementação de Filtros	16
2.1.2.1 Topologia <i>Sallen-Key</i>	16
2.1.2.2 Topologia <i>Multiple Feedback</i>	16
2.1.2.3 Topologia <i>KHN</i>	16
2.2 Filtros Digitais	16
2.2.1 Famílias de filtros digitais	16
2.2.2 Implementação de Filtros	16
2.3 Síntese de Filtros	16
2.4 Ferramentas de desenvolvimento	16
2.4.1 Python	16
2.4.2 NumPy	17
2.4.3 SciPy	17
2.4.4 matplotlib	17
2.4.5 PyQt	17
3 DESENVOLVIMENTO	18
3.1 Estrutura do programa	18
3.2 Metodologia de desenvolvimento	18
3.2.1 TDD (Test Driven Development)	18
3.2.2 Controle de versão	18
4 RESULTADOS	19
5 CONCLUSÃO	20
5.1 Futuras melhorias	20
6 REFERÊNCIAS	21

1 INTRODUÇÃO

Blá blá blá.

1.1 Motivação

As seguintes razões motivaram a escolha do tema e a escrita deste trabalho:

- Blá
- Blá
- Blá

1.2 Objetivos

Os objetivos principais deste trabalho são:

- Blá
- Blá
- Blá

1.3 Estrutura do trabalho

O capítulo 2 irá realizar uma revisão teórica dos conhecimentos empregados nesse trabalho. No capítulo 3 será abordado o processo de desenvolvimento, seguindo-se um capítulo no qual o *software* desenvolvido será demonstrado e discutido. Após, serão apresentadas conclusões e sugestões para futuras melhorias.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Filtros Analógicos

2.1.1 Famílias de filtros analógicos

2.1.1.1 Filtro de Butterworth

2.1.1.2 Filtro de Bessel

2.1.1.3 Filtro Elíptico

2.1.1.4 Filtro Chebyshev, tipo 1

2.1.1.5 Filtro Chebyshev, tipo 2

2.1.2 Implementação de Filtros

2.1.2.1 Topologia *Sallen-Key*

2.1.2.2 Topologia *Multiple Feedback*

2.1.2.3 Topologia *KHN*

2.2 Filtros Digitais

2.2.1 Famílias de filtros digitais

2.2.2 Implementação de Filtros

2.3 Síntese de Filtros

2.4 Ferramentas de desenvolvimento

2.4.1 Python

Python é uma linguagem de programação de alto nível, de propósito geral, cuja sintaxe permite ao programador expressar ideias complexas em poucas linhas de código. Atualmente,

vários softwares proprietários e de código aberto são desenvolvidos nessa linguagem ou a empregam como linguagem de *script*.

Python foi escolhida para o presente trabalho pelo fato de sua simplicidade, aliada à grande disponibilidade de bibliotecas, facilitar o desenvolvimento.

Para o uso dela em aplicações científicas e de engenharia, existe um conjunto de bibliotecas básicas que adicionam funcionalidade similar a de ferramentas como o MATLAB, permitindo o trabalho com números complexos, matrizes, a geração de gráficos e o cálculo de funções especiais, entre outras funcionalidades necessárias.

2.4.2 NumPy

2.4.3 SciPy

2.4.4 matplotlib

2.4.5 PyQt

3 DESENVOLVIMENTO

Esse capítulo irá discorrer sobre as metodologias e procedimentos utilizados no desenvolvimento da ferramenta.

3.1 Estrutura do programa

3.2 Metodologia de desenvolvimento

3.2.1 TDD (Test Driven Development)

TDD (*Test Driven Development*, na literatura em português encontrado como *Desenvolvimento orientado para testes*) é uma metodologia de desenvolvimento de software que blá blá blá.

Considerou-se essa metodologia adequada para o presente trabalho pois ela fornece uma maneira rápida e eficiente de testar os algoritmos e estruturas de dados, além de obrigar o desenvolvedor a pensar em termos de atender requisitos.

Para adicionar-se uma nova funcionalidade, inicialmente escreve-se e executa-se um teste. Se ele executar sem erros, considera-se o desenvolvimento desta funcionalidade completo; caso contrário, o código é desenvolvido e modificado em pequenos passos até que todos os testes passem com sucesso.

Dessa forma, o desenvolvedor tem maior confiança para realizar alterações em um código: se a mudança provocar falha em algum dos testes, sabe-se exatamente o que causou o defeito.

3.2.2 Controle de versão

4 RESULTADOS

Nesse capítulo a ferramenta será demonstrada, comparando-se os resultados obtidos com ela àqueles disponíveis na literatura.

5 CONCLUSÃO

O código-fonte de ferramenta será futuramente disponibilizado, sob licença de software livre, para permitir que seu desenvolvimento seja continuado.

5.1 Futuras melhorias

6 REFERÊNCIAS

MANCINI, R. et al. **Op-Amps for Everyone**. 4a edição. Burlington: Newnes, 2003. 304p.

Python 3.4.3 Documentation. Disponível em <https://docs.python.org/3/>. Acesso em 03 mar. 2015.

SMITH, S. **The Scientist and Engineer's Guide to Digital Signal Processing**. 1a edição. s.l: California Technical Publishers, 1997. 626p. Disponível em <http://www.dspguide.com/>. Acesso em 11 mar. 2015.