

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETROTÉCNICA
CURSO DE ENGENHARIA ELÉTRICA

ALINE POLICARPO

GRACIELLE SILVÉRIO DE LIMA

**IMPLEMENTAÇÃO DE FILTROS DIGITAIS UTILIZANDO
RASPBERRY PI E MATLAB/SIMULINK**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2019

ALINE POLICARPO
GRACIELLE SILVÉRIO DE LIMA

**IMPLEMENTAÇÃO DE FILTROS DIGITAIS UTILIZANDO
RASPBERRY PI E MATLAB/SIMULINK**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 2, do Curso de Engenharia Elétrica do Departamento Acadêmico de Eletrotécnica – DAELT – da Universidade Tecnológica Federal do Paraná (UTFPR), como requisito parcial para obtenção do título de Engenheira eletricista.

Orientador: Prof. Glauber Gomes de Oliveira Brante

CURITIBA
2019

Aline Policarpo
Gracielle Silvério de Lima

Implementação de Filtros Digitais Utilizando Raspberry Pi e Matlab/Simulink

Este Trabalho de Conclusão de Curso de Graduação foi julgado e aprovado como requisito parcial para a obtenção do Título de Engenheiro Eletricista do curso de Engenharia Elétrica do Departamento Acadêmico de Eletrotécnica (DAELT) da Universidade Tecnológica Federal do Paraná (UTFPR).

Curitiba, 27 de junho de 2019.

Prof. Emerson Rigoni, Dr.
Coordenador de Curso
Engenharia Elétrica

Profª. Annemarlen Gehrke Castagna, Mestre
Responsável pelos Trabalhos de Conclusão de Curso
de Engenharia Elétrica do DAELT

ORIENTAÇÃO

Glauber Gomes de Oliveira Brante, Dr.
Universidade Tecnológica Federal do Paraná
Orientador

BANCA EXAMINADORA

Glauber Gomes de Oliveira Brante, Dr.
Universidade Tecnológica Federal do Paraná

Ohara Keraususkas Rayel
Universidade Tecnológica Federal do Paraná

Thiago Passarin
Universidade Tecnológica Federal do Paraná

A folha de aprovação assinada encontra-se na Coordenação do Curso de Engenharia Elétrica.

RESUMO

LIMA, Gracielle S.; POLICARPO, Aline. IMPLEMENTAÇÃO DE FILTROS DIGITAIS UTILIZANDO RASPBERRY PI E MATLAB/SIMULINK, 2019, 69 f. Trabalho de conclusão de Curso (Graduação – Curso de Engenharia Engenharia Elétrica). Universidade Tecnológica Federal do Paraná, Curitiba, 2019.

Este trabalho apresenta as etapas do desenvolvimento de um projeto de implementação de filtros digitais utilizando o *hardware* Raspberry Pi e o *software* Matlab/Simulink, a fim de utilizar esse recurso como uma forma de aprendizagem para a fixação de conceitos teóricos em disciplinas acadêmicas relacionadas a Sinais e Sistemas, sem a necessidade de conhecimentos de programação. Serão abordadas as etapas de configurações necessárias para o *hardware* e o *software* utilizado, bem como testes de bancada para a implementação de filtros dos tipos FIR e IIR. As implementações testadas foram apresentadas na disciplina de Sinais e Sistemas 2, na UTFPR, e avaliada pelos discentes da disciplina. Por fim, um equalizador de frequências foi construído a partir da utilização de vários blocos de programação.

Palavras chave: Filtros digitais. Sinais e Sistemas. Recurso didático. Design de filtros.

ABSTRACT

LIMA, Gracielle S.; POLICARPO, Aline. IMPLEMENTATION OF DIGITAL FILTERS USING RASPBERRY PI AND MATLAB / SIMULINK, 2019, 2019, 69 f .Trabalho de conclusão de Curso (Graduação – Curso de Engenharia Engenharia Elétrica). Universidade Tecnológica Federal do Paraná, Curitiba, 2019.

This project presents the steps of the development of a digital filters implementation project using Raspberry Pi and Matlab / Simulink, in order to use this resource as a learning form for the establishment of theoretical concepts in academic disciplines related to Signals and Systems, without the need for programming knowledge. It will cover the necessary configuration steps for the hardware and software used, as well as bench tests for the implementation of FIR and IIR filters. The tested implementations were presented in the discipline of Signals and Systems 2, in UTFPR, and evaluated by the discipline students. Finally, a frequency equalizer was constructed from the use of several programming blocks.

Key words: Digital filters. Signals and Systems. Didactic resource. Filter design.

LISTA DE FIGURAS

Figura 1 - Diagrama de blocos com os equipamentos a serem utilizados para a implementação de filtro digital com Raspberry Pi	12
Figura 2 - Conversão de Sinais	15
Figura 3 - Processo de amostragem.....	16
Figura 4 - Ilustração do efeito aliasing.....	16
Figura 5 - Resposta em frequência para um filtro passa-baixas ideal.	17
Figura 6- Resposta de Módulos Ideais - a) filtro passa-faixa; b) filtro rejeita-faixa.	18
Figura 7 - Diagrama de blocos de um filtro FIR de ordem 3.	19
Figura 8 – Resposta ao módulo de filtro rejeita-faixa pelo método de função-janela.....	20
Figura 9 - Formas de Janelas no domínio do tempo.....	21
Figura 10 – Detalhe da transformada de Fourier com M=80 das janelas: Retangular, Triangular, Hamming , Hann e Blackman	22
Figura 11 - Diagrama de Blocos para Filtro IIR de ordem 1	23
Figura 12 - Ganho em dB para um filtro passa-baixas.....	24
Figura 13 - Banda de passagem dos filtros IIR	26
Figura 14 - Faixa de transição e banda de rejeição de filtros IIR.....	27
Figura 15 - Blocos de um programa no Matlab/Simulink.....	29
Figura 16 – Ligações em bancada.	30
Figura 17 - Configurações de bloco de design de Filtro tipo Janela Retangular.....	31
Figura 18 – Sinais de onda senoidal de entrada e saída para filtro passa-baixa do tipo FIR - Janela Retangular	31
Figura 19 – Magnitude x Frequência para filtros do tipo FIR de ordem 80, passa-baixa, Fc=400Hz. ...	32
Figura 20 – Detalhe do bloco de design para implementação de Filtro IIR.....	34
Figura 21 - Resposta do filtro Butterworth	35
Figura 22 - Resposta do filtro Chebyshev 1	36
Figura 23 - Resposta do filtro Chebyshev 2.....	37
Figura 24 - Banda de rejeição do filtro Chebyshev 2.....	38
Figura 25 - Resposta do filtro Elíptico.....	38
Figura 26 – Magnitude x Frequência para filtros pelo método de Janela Retangular de ordens 16 e 80.	41
Figura 27 - Magnitude x Frequência para filtros pelo método de Janela Retangular e Hamming de ordem 80.....	41
Figura 28 – Entrada e saída de experiemnto com filtro passa-altas.	42
Figura 29 - Onda	43
Figura 30 - Diagrama de um Equalizador.	46
Figura 31 - Faixas de frequências de cada bloco de filtro.....	47
Figura 32 - Ajuste de ganhos em faixas de frequência de um equalizador.	47
Figura 33 - Instalação do pacote de suporte do <i>Raspberry Pi</i>	53
Figura 34 - Janela de setup das configurações do <i>Raspberry Pi</i> no <i>Matlab</i>	53
Figura 35- Bloco de captura de audio e tela de configuração	55
Figura 36 - Bloco de saída de áudio e tela de configuração.....	55
Figura 37 - Primeiros passos para fazer download do programa	56
Figura 38 - Acesso às configurações de <i>hardware</i>	57
Figura 39– Configuração de <i>Hardware</i> – Parâmetros de placa.	58
Figura 40– Configuração de <i>Hardware</i> – Opções de construção.....	59
Figura 41 - Diagrama de blocos do Simulink.....	61
Figura 42 - Diagrama de blocos com os equipamentos utilizados para a implementação de filtro digital com Raspberry Pi	62
Figura 43 - Configuração de bloco de design para Experimento 1	63
Figura 44 - Diagrama de blocos com os equipamentos utilizados para Experimento 3.....	66

SUMÁRIO

1	INTRODUÇÃO	7
1.1	TEMA.....	7
1.1.1	Delimitação do Tema.....	8
1.2	PROBLEMA E PREMISSAS	8
1.3	OBJETIVOS	9
1.3.1	Objetivo Geral	9
1.4	JUSTIFICATIVA	10
1.5	PROCEDIMENTOS METODOLÓGICOS	11
1.6	ESTRUTURA DO TRABALHO	13
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	AMOSTRAGEM.....	15
2.2	FILTROS IDEAIS	16
2.3	FILTROS FIR.....	18
2.4	FILTROS IIR.....	22
2.4.1	Aproximações para filtros IIR	24
3	DESIGN DE FILTROS DIGITAIS UTILIZANDO HARDWARE RASPBERRY PI E SOFTWARE MATLAB-SIMULINK	28
3.1	DESCRIÇÃO DO SISTEMA IMPLEMENTADO	28
3.2	IMPLEMENTAÇÕES DE FILTROS DIGITAIS	29
3.2.1	Exemplo de Implementação.....	30
3.2.2	Implementação de filtros tipo FIR	32
3.2.3	Implementação de filtros tipo IIR.....	34
4	APLICAÇÕES PRÁTICAS DO SISTEMA IMPLEMENTADO	40
4.1	UTILIZAÇÃO DE <i>DESIGN</i> DE FILTROS CRIADO COMO RECURSO DIDÁTICO	40
4.1.1	Demonstração de experimentos na disciplina de Sinais e Sistemas 2	40
4.1.2	Avaliação dos discentes sobre utilização de <i>design</i> de filtro como recurso didático.....	43
4.2	CONSTRUÇÃO DE UM EQUALIZADOR A PARTIR DE BLOCOS DE PROGRAMAÇÃO	45
5	CONSIDERAÇÕES FINAIS	49
	REFERÊNCIAS	50
	APÊNDICE A – Configurações do sistema	52
A.1	Instalação de sistema operacional do Raspberry Pi via Matlab-Simulink.....	52
A.2	Configuração do adaptador de áudio.....	54
A.3	Configuração de blocos no Matlab-Simulink.....	54
A.4	<i>Download</i> de programa no Raspberry Pi.....	56
	APÊNDICE B – Demonstração de implementação de filtros através de experimentos práticos ..	60
B.1	Objetivo.....	60
B.2	Materiais e Equipamentos	60
B.3	Ambientação.....	61
B.4	Experimento 1: Implementação Dos Filtros Passa-Baixas Dos Tipos FIR E IIR	63
B.5	Experimento 2: Implementação dos Filtros FIR Passa-Altas.....	65
B.6	Experimento 3: Implementação dos Filtros Para Sinais Reais	66
	APÊNDICE C – Ficha de avaliação da demonstração de implementação de filtros através de experimentos práticos	68

1 INTRODUÇÃO

1.1 TEMA

A fixação de conceitos relacionados a processamento digital de sinais é notavelmente um desafio para os cursos de graduação da área. Em Engenharia Elétrica e outras engenharias similares, os alunos são primeiramente submetidos a disciplinas introdutórias de sinais contínuos no tempo e então, passam por disciplinas mais aplicadas, com foco em controle, processamento digital de sinais, entre outras. Para estas últimas disciplinas, o desempenho dos discentes é uma preocupação no meio acadêmico (SIMORI, Mario, ABURDENE, Maurice, 2016).

A complexidade dos conceitos teóricos e a dificuldade de absorção desses conceitos por parte dos discentes gera a necessidade de buscar recursos didáticos para facilitar o processo de aprendizagem nas disciplinas acadêmicas de processamento digital de sinais.

Nesse sentido, há a disponibilidade de vários recursos para auxiliar no processo de aprendizagem, desde videoaulas desenvolvidas pelo MIT (Instituto de Tecnologia de Massachusetts – EUA) na década de 1980, até cursos disponíveis em plataformas *online*, tutoriais práticos, entre outros. (BARAN, Thomas A. et al., 2016)

Contudo, nota-se a necessidade de explorar atividades *hands-on*, isto é, atividades práticas que demonstram de maneira mais visual e aplicada os conceitos teóricos da disciplina, visto que uma das possíveis razões para a dificuldade de aprendizagem está na falta de experiência com sinais e sistemas. (SIMORI, Mario, ABURDENE, Maurice, 2016)

Há várias formas de utilizar recursos para atividades que estimulem a prática com a manipulação de sinais e sistemas. Uma boa forma é a utilização do *software* Matlab e de *hardwares* para processamento digital de sinais, visto que são inerentes a implementações de sinais discretos e envolvem a criação de códigos de programação, o que aborda tanto conceitos teóricos de processamento de sinais quanto experiências práticas de implementação.

1.1.1 Delimitação do Tema

Para a implementação de um sistema a fim de processar sinais, é necessário um *hardware* que faça a leitura do sinal analógico, converta-o em digital, processe via *software* e retorne ao *hardware* para converter o sinal de saída de digital para analógico novamente. Utilizando um Raspberry Pi como *hardware* e a ferramenta Simulink do Matlab para desenvolver o *software* do sistema, é possível gerar, codificar, modular, filtrar, demodular e transmitir sinais.

Neste projeto foi desenvolvido um sistema baseado na utilização de um Raspberry Pi e Matlab/Simulink, para processamento de sinais com finalidade educacional, cujo o foco é a construção de filtros digitais. Os filtros digitais básicos são do tipo passa-baixa, passa-alta, passa-faixa ou rejeita-faixa. Os principais métodos de filtragem de sinais digitais são *Infinite Impulse Response* (IIR) e *Finite Impulse Response* (FIR), sendo que ambos serão implementados.

Um exemplo de aplicação da implementação desses filtros é um equalizador digital, que foi desenvolvido ao fim desse projeto combinando os projetos de filtros previamente implementados com aplicação de áudio utilizando o Raspberry Pi.

1.2 PROBLEMA E PREMISSAS

Na matemática, é possível transcrever funções do domínio do tempo para o domínio da frequência, utilizando a transformada de Fourier, para se resolver problemas mais facilmente. Essa estratégia é adotada em várias disciplinas básicas de Engenharia Elétrica como Circuitos Elétricos, Sinais e Sistemas, Sistemas de Controle, por exemplo.

Para isso, compreender as relações entre a frequência e domínio do tempo é fundamental. Um dos primeiros contatos com as aplicações da mudança de domínios se dá na disciplina de Sinais e Sistemas. Porém, a falta de experiências práticas dos conceitos de sinais e sistemas é uma razão que explica a dificuldade do aprendizado da disciplina. (SIMONI, Mario, ABURDENE, Maurice, 2016).

Verificar os princípios visualmente em suas aplicações ajuda a construir as bases da teoria, porém quando se trata de algo abstrato como em sinais e sistemas é difícil de tornar os

conceitos visuais.

Para enfatizar os conceitos teóricos da disciplina de Sinais e Sistemas, ou outras relacionadas ao processamento de sinais, são utilizados alguns *softwares* como MATLAB, Spice, Mathematica, Mathcad. Porém, mesmo com a utilização desses recursos, os estudantes ainda têm dificuldade em tais cursos, uma vez que os vêem como apenas matemática e teoria (ABURDENE, Maurice F., NEPAL, Kundal, 2011).

A utilização de um recurso prático e de simples entendimento pode auxiliar na fixação desses conceitos teóricos. A implementação de filtros digitais utilizando o Raspberry Pi e o Matlab é uma opção para geração e análise de sinais em frequência e domínio do tempo, o que pode ser utilizado como ferramenta didática.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Desenvolver uma plataforma para uso educacional baseada em Raspberry Pi e *software* Matlab/Simulink, para a aplicação de filtros digitais em sinais captados pelo *hardware*.

1.3.2 Objetivos Específicos

As principais etapas que foram realizadas para atingir os objetivos deste trabalho foram:

- Levantar os tipos de filtros digitais;
- Levantar os métodos de filtragem de sinais digitais;
- Comunicar Raspberry Pi com o Matlab;
- Utilizar interface com o usuário utilizando Matlab Simulink, possibilitando a alteração de tipo e parâmetros de filtro;

- Exibir a resposta do sistema projetado via Matlab e ainda disponibilizar o sinal analógico da resposta na saída do Raspberry Pi;
- Desenvolver roteiro para prática de *design* de filtros para processar o sinal de entrada conforme os parâmetros configurados pelo usuário;
- Avaliar a utilização em aulas da disciplina de Sinais e Sistemas 2.
- Desenvolver equalizador a partir do *design* de filtros previamente desenvolvidos;

1.4 JUSTIFICATIVA

Na grade curricular de Engenharia Elétrica da UTFPR, o ensino sobre filtros faz parte da disciplina de Sinais e Sistemas de forma matemática, o que pode tornar a área pouco atrativa para os estudantes. Somente em Sinais e Sistemas 2, que é disciplina optativa, é possível maior contato com o funcionamento dos filtros e dos métodos de filtragem. Porém, esse contato é feito através de simulações computacionais, o que torna o conteúdo menos abstrato, mas ainda não permite fazer operações com sinais em tempo real. O entendimento de filtros de sinais é de grande importância na área de Engenharia Elétrica, pois possibilita remover ruídos em sistemas de telecomunicações, reduzir harmônicas indesejadas em sistemas de potência, remover frequências indesejadas no processamento de sinais de sensores, entre outras aplicações.

A implementação de filtros digitais utilizando o Raspberry Pi e o Matlab facilita a aprendizagem da aplicação de filtros em sinais, uma vez que fornecerá uma interface visual, possibilitando verificar e alterar os parâmetros de filtros, visualizar os sinais de entrada e saída, e ainda fornece uma saída analógica do sinal processado.

As principais vantagens que esse projeto oferecerá são o baixo custo associado, sendo o valor do *hardware* inferior a R\$200 (duzentos reais) por unidade, além de também possibilitar ao professor aplicar uma atividade prática sobre o tema na qual o aluno não precisa ter conhecimentos prévios de programação e de arquitetura de *hardware*, otimizando o tempo de aula para o desenvolvimento da atividade relacionada ao estudo dos filtros digitais.

Além disso, a utilização do Raspberry Pi permite a realização da filtragem sem a utilização simultânea do Matlab, o que permite o desenvolvimento de projeto de disciplina e trabalhos de conclusão de curso com essa ferramenta.

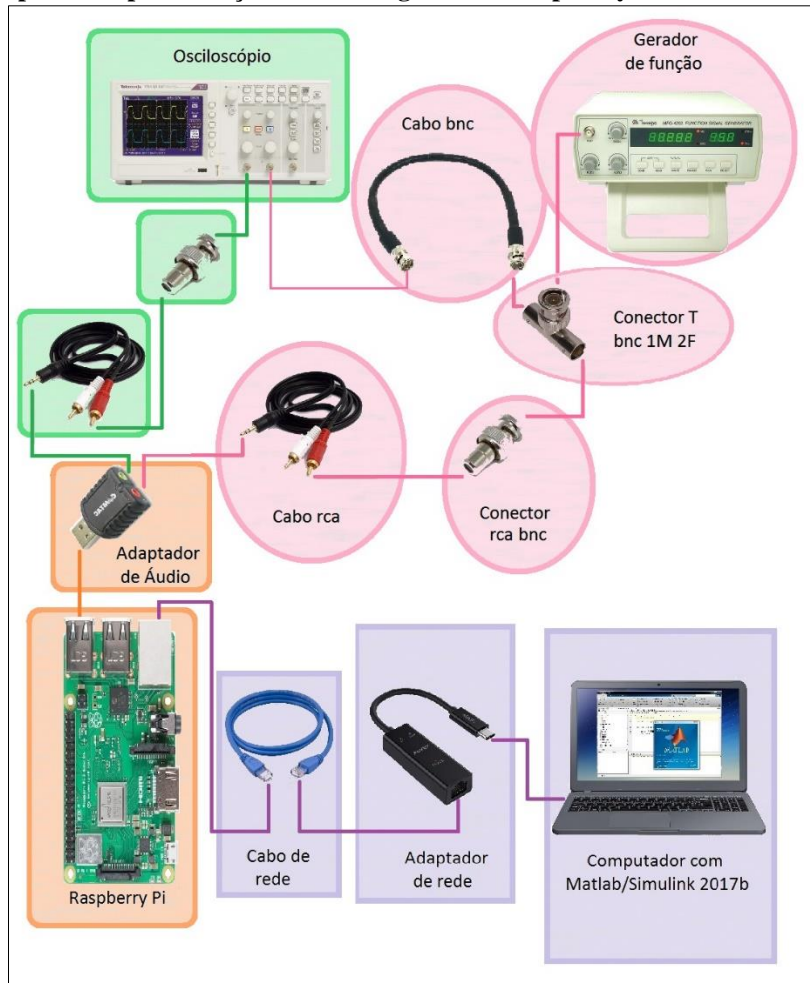
1.5 PROCEDIMENTOS METODOLÓGICOS

Para a execução desse trabalho foram necessários os seguintes equipamentos:

- Raspberry Pi 3 Model B+;
- Cartão de memória micro SD de 8Gb ou mais;
- Cabo de rede;
- Cabo USB micro;
- Fonte de alimentação 5V com saída USB;
- Adaptador de áudio USB (interface de áudio);
- Conectores RCA-BNC;
- Cabos RCA
- Cabo BNC macho-macho
- Plug de áudio;
- Adaptador *USB-Ethernet*.

Além destes componentes, também são necessários um computador com o *software* MATLAB/Simulink versão 2017b ou mais recente, um osciloscópio e um gerador de função, equipamentos disponíveis para empréstimo para utilização em laboratório. A forma com que esses equipamentos são integrados para a implementação de um filtro digital pode ser observada na Figura 1.

Figura 1 - Diagrama de blocos com os equipamentos a serem utilizados para a implementação de filtro digital com Raspberry Pi



Fonte – Adaptado de PASOLIN, Gianni et. all (2017).

A primeira fase deste trabalho consistiu na pesquisa sobre a utilização e o funcionamento do *hardware* Raspberry Pi. Durante essa etapa, o objetivo foi entender como deve ser feita a configuração do *hardware* para a comunicação com o computador, instalação do adaptador de áudio, além da instalação dos Add-Ons dos pacotes de suporte do Raspberry Pi e configurações necessárias no software MATLAB.

Na segunda fase foi feita a montagem do sistema apresentado na Figura 1, isto é, utilizando um adaptador de rede e cabo de rede foi configurada uma rede entre o Raspberry Pi e o computador, o que possibilitou a comunicação entre os dois dispositivos. O adaptador de áudio *USB* serviu como conversor analógico-digital, no qual os sinais de entrada foram adquiridos do gerador de função e a saída pode ser observada pelo osciloscópio. Utilizando uma derivação em T na saída do gerador de funções, foi possível conectar o sinal de entrada do sistema no outro canal do osciloscópio, possibilitando fazer uma comparação entre os sinais de entrada e saída do sistema.

A terceira fase desse trabalho consistiu na implementação de filtros digitais no

Raspberry Pi através do uso do bloco de *design* de filtros do Matlab/Simulink. Assim, foi possível configurar os parâmetros de filtro e processar os sinais obtidos na entrada da estrutura de *hardware* anteriormente preparada, fornecendo sinais de saída já filtrados que puderam ser visualizados no osciloscópio. A partir disto, foi desenvolvido um roteiro de experimento para que os conceitos básicos de filtros digitais pudessem ser observados na prática.

A quarta fase desse trabalho foi a apresentação do projeto realizado como um recurso didático para alunos da disciplina de Sinais e Sistemas 2 na UTFPR, bem como a avaliação dos alunos submetidos a esse recurso.

Por fim, na quinta fase desse trabalho foi desenvolvido um equalizador de frequências através de um arranjo de blocos de *design* de filtro associados a blocos de ganhos. O programa foi implementado na mesma estrutura de hardware anterior, e pode-se observar diferentes ganhos e atenuações, conforme configurado, para diferentes faixas de frequências.

1.6 ESTRUTURA DO TRABALHO

Este trabalho consistirá em 4 capítulos conforme descrição a seguir.

Capítulo 1: Introdução

Neste capítulo, o trabalho é introduzido, especificando o tema a ser abordado, bem como os objetivos a serem desenvolvidos.

Capítulo 2: Fundamentação Teórica

Neste capítulo é feita uma abordagem sobre as pesquisas realizadas sobre filtros digitais, utilizando-se de referências bibliográficas para a compreensão do funcionamento da plataforma.

Capítulo 3: *Design* de filtros digitais utilizando *hardware* Raspberry Pi e *software* Matlab-Simulink

Neste capítulo são apresentados brevemente o Raspberry Pi e Matlab/Simulink, bem como a descrição da implementação dos filtros, além de comentários sobre a utilização prática dos filtros escolhidos.

Capítulo 4: APLICAÇÕES PRÁTICAS DO SISTEMA IMPLEMENTADO

Neste capítulo é apresentada a maneira de como os filtros digitais desenvolvidos foram utilizados como recurso didático na disciplina de Sinais e Sistemas 2, na UTFPR, além

de mostrar a avaliação dos discentes sobre a utilização desse recurso.

Por fim, este mesmo capítulo mostra as etapas de desenvolvimento e testes de um equalizador a partir dos filtros previamente implementados.

Capítulo 5: Considerações Finais

Neste capítulo são apresentadas as análises sobre a utilização do projeto realizado como recurso didático, bem como considerações sobre os testes realizados com as implementações de filtros digitais e sobre o equalizador de frequências construído.

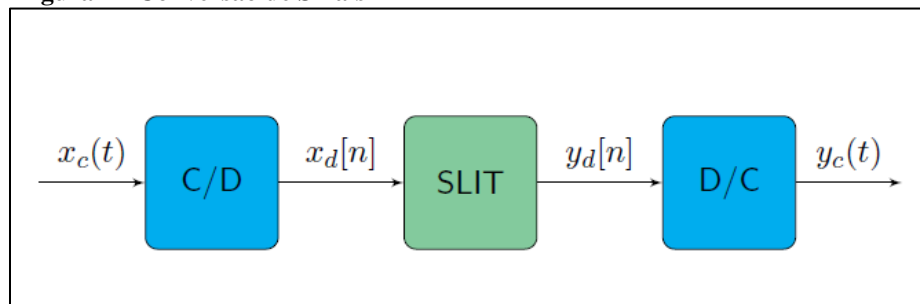
2 FUNDAMENTAÇÃO TEÓRICA

2.1 AMOSTRAGEM

Os sinais digitais e discretos no tempo e na amplitude, são os tipos de sinais que um computador pode interpretar, os quais são representações de sinais analógicos e contínuos no tempo, sendo estes últimos a maioria dos sinais fornecidos pelo mundo real (WEEKS, 2007).

Seja um sistema linear e invariante no tempo (SLIT), que deverá receber os sinais vindos do mundo real, $x(t)$, processá-los e então gerar em uma saída um sinal $y(t)$, no mesmo formato do sinal de entrada. Para isto, será necessário a utilização de conversores. O primeiro, transforma o sinal analógico em digital e o segundo, a partir de um sinal digital, o reconstrói em um sinal analógico como mostra a Figura 2.

Figura 2 - Conversão de Sinais

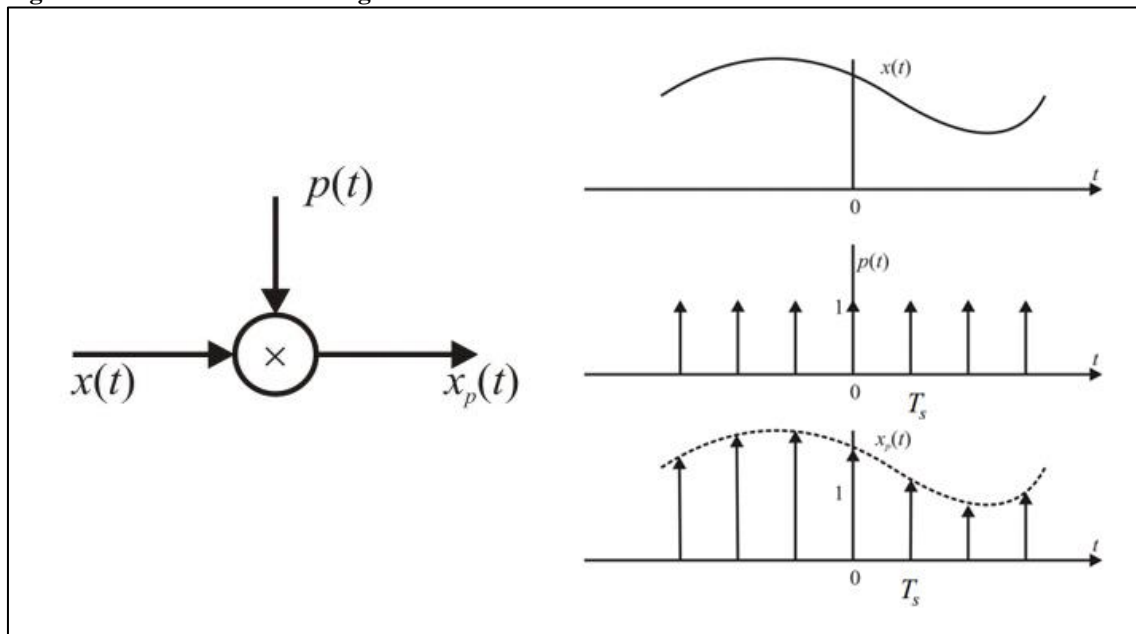


Fonte: Adaptado de Oppenheim, 1997.

Para a conversão contínuo/digital (bloco C/D) são necessários os processos de amostragem e quantização. No processo de amostragem, a variável independente, que é contínua no tempo, será discretizada e no processo de quantização a amplitude do sinal será aproximada para valores discretos. Resumidamente, a quantização é um arredondamento do valor da amplitude, que depende da resolução do dispositivo de conversão. Este processo pode gerar erros, que dependem do sinal.

A amostragem, isto é, a transformação de um sinal de tempo contínuo em discreto, pode ser feita através da multiplicação de um sinal $x(t)$ por um trem de impulsos $p(t)$. Os pulsos são dados em intervalos de tempo T_s , chamado de período de amostragem, e a frequência de amostragem é dada por $F_s = 1/T_s$. O resultado é em um sinal $x_p(t)$, como mostra a Figura 3.

Figura 3 - Processo de amostragem

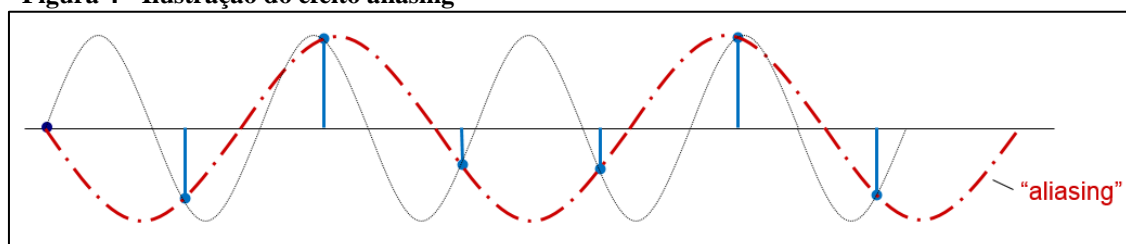


Fonte: Adaptado de OPPENHEIM, 1997.

Para a reconstrução de um sinal amostrado é necessário um filtro passa-baixas para retirar as novas frequências geradas. Para isso, deve existir um espaçamento mínimo entre cada cópia do espectro original. Esta é uma importante regra da amostragem: a frequência de amostragem (ω_s) deve ser no mínimo maior que o dobro da frequência do sinal, ou seja, $\omega_s > 2\omega_{\text{máx}}$.

Se esta condição não for satisfeita, estas componentes do espectro irão se sobrepor e algumas informações serão perdidas. Este fenômeno é conhecido como *aliasing* e se isto ocorrer o sinal reconstruído terá erros e não será uma representação do sinal de original. O resultado disto é exemplificado na Figura 4.

Figura 4 - Ilustração do efeito aliasing



Fonte: Autoria própria.

2.2 FILTROS IDEAIS

A manipulação de sinais é frequentemente utilizada para diversas aplicações, como

um sinal de áudio, por exemplo, que pode ser equalizado ou ter seu volume aumentado. Os filtros têm como função manipular sinais de modo necessário à aplicação destes (OPPENHEIM; WILLSKY,1996).

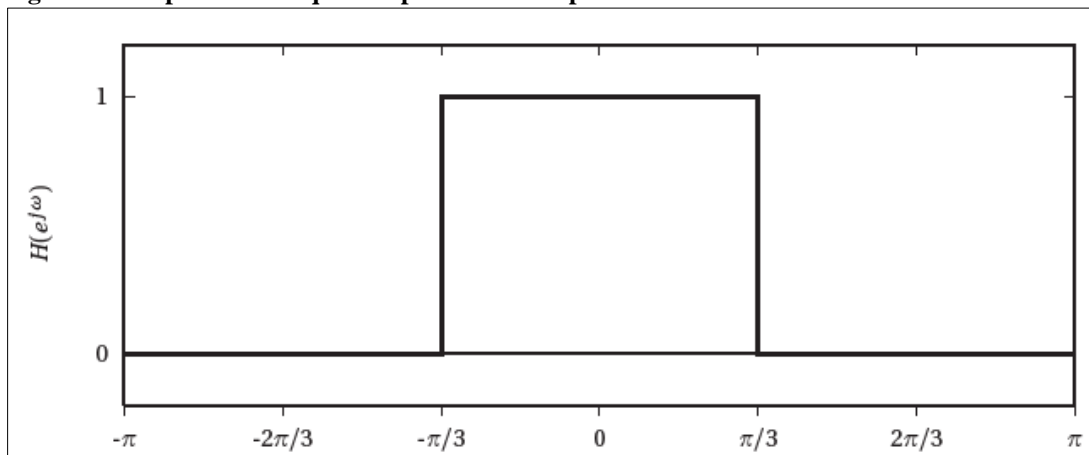
Uma maneira de entender a função dos filtros é pela representação dos filtros ideais. Segundo Prandoni e Vetterli (2008), filtros ideais são abstrações teóricas que capturam a essência da filtragem básica operação, mas que não são realizáveis na prática. Para Diniz, Silva e Netto (2014), a implementação de um filtro se baseia em sua função de transferência $H(z)$ da seguinte forma:

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n}. \quad (1)$$

Um filtro ideal passa-baixas pode ser caracterizado por um filtro que retira de um sinal todo o conteúdo de frequência superior à frequência de corte (ω_c) e mantém para este sinal todo o conteúdo de frequência inferior à frequência de corte. (PRANDONI; VETTERLI, 2008).

A Figura 5 representa a resposta em frequência de um filtro passa-baixas ideal com $\omega_c = \pi/3$ e exemplifica a equação anterior.

Figura 5 - Resposta em frequência para um filtro passa-baixas ideal.



Fonte: (PRANDONI; VETTERLI, 2008).

Um filtro ideal passa-altas é um filtro caracterizado por fazer o contrário de um filtro passa-baixas, ou seja, um filtro passa-altas elimina todo o conteúdo de frequência inferior à frequência de corte e permite a passagem de todo o conteúdo de frequência superior a ela.

Portanto, a resposta em frequência para esse filtro pode ser definida por:

$$H(e^{j\omega}) = \begin{cases} 0 & |\omega| \leq \omega_c \\ 1 & \text{caso contrário} \end{cases} \quad (2)$$

Um filtro ideal passa-faixa é aquele que permite a passagem de conteúdo de um

intervalo de frequência, definido por dois valores de frequência limite, e não elimina o conteúdo para frequências inferiores e superiores a esse intervalo. Um filtro ideal passa-faixa é ilustrado na Figura 6 (a) e pode ser definido pela seguinte equação:

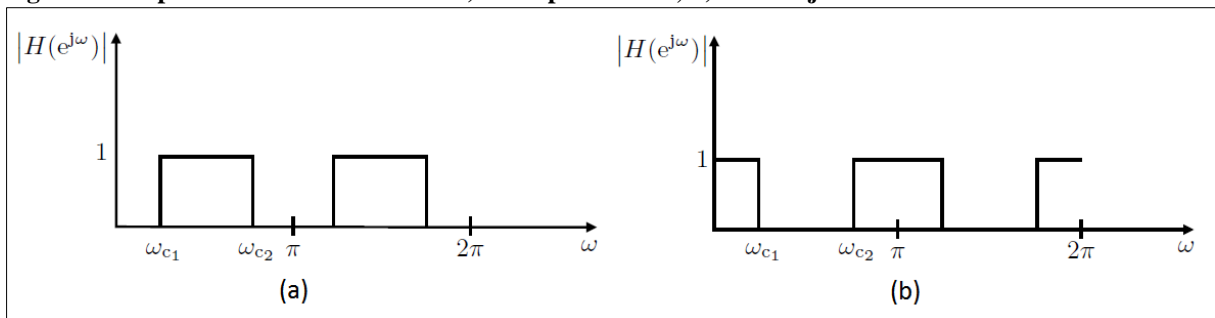
$$H(e^{j\omega}) = \begin{cases} 1 & \omega_{c1} \leq \omega \leq \omega_{c2} \\ 0 & \text{caso contrário} \end{cases} \quad (3)$$

Já um filtro ideal rejeita-faixa tem o funcionamento contrário ao de um passa-faixa, como é ilustrado na Figura 6 (b). Portanto, a resposta ao módulo de um filtro rejeita-faixa pode ser dada por:

$$|H(e^{j\omega})| = \begin{cases} 1 & 0 \leq |\omega| \leq \omega_{c1} \\ 0 & \omega_{c1} < |\omega| \leq \omega_{c2} \\ 1 & \omega_{c2} < |\omega| \leq \pi \end{cases} \quad (4)$$

considerando ω_{c1} e ω_{c2} as frequências que limitam a faixa a ser rejeitada (DINIZ; SILVA; NETTO, 2014).

Figura 6- Resposta de Módulos Ideais - a) filtro passa-faixa; b) filtro rejeita-faixa.



Fonte: Diniz, Silva e Netto (2014).

2.3 FILTROS FIR

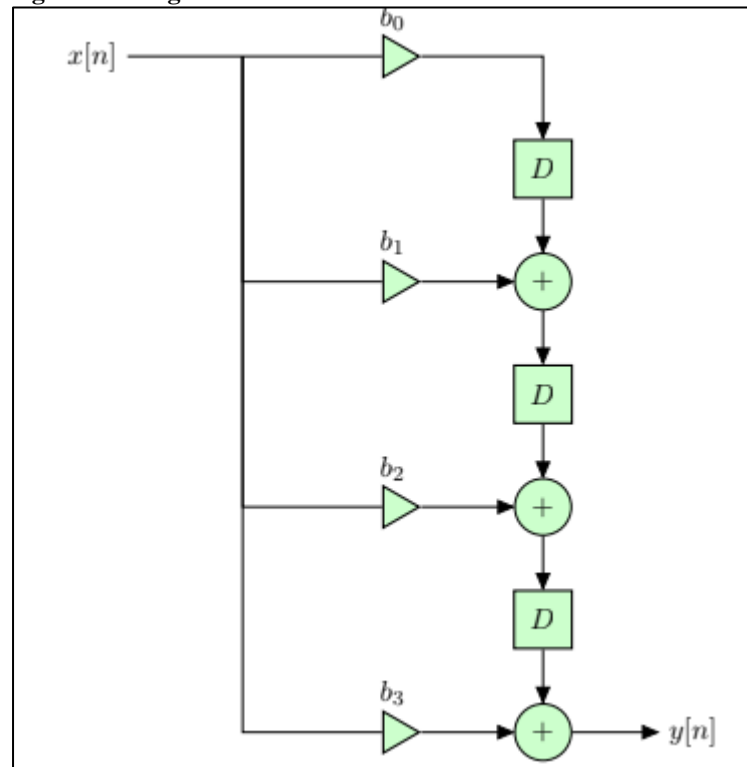
Um filtro FIR pode ser definido como um filtro de resposta ao impulso de duração finita (DINIZ; SILVA; NETTO, 2014). A função de transferência de um filtro FIR pode ser dada por:

$$H(z) = \sum_{n=0}^M b_n z^{-n}, \quad (5)$$

sendo M a ordem do e filtro b_n os seus coeficientes.

Na Figura 7 pode-se observar um diagrama de blocos para um filtro FIR de ordem 3.

Figura 7 - Diagrama de blocos de um filtro FIR de ordem 3.



Fonte: Adaptado de OPPENHEIM, 1997.

Os projetos de filtros FIR utilizam técnicas baseadas na aproximação direta da resposta ao impulso desejada do sistema de tempo discreto. Neste trabalho, serão brevemente abordados alguns tipos de métodos com funções-janela para aproximação de filtros FIR.

Para os filtros ideais citados na Seção 2.2, as respostas ao impulso têm infinitos coeficientes, levando a filtros não executáveis. Uma alternativa é definir uma sequência auxiliar $h'(n)$ de comprimento finito que resulte num filtro de ordem M . Isto é obtido multiplicando a resposta ao impulso $h(n)$ por uma função-janela $\omega(n)$, projetada de forma a introduzir um mínimo de desvio em relação à resposta na frequência ideal. Os coeficientes da resposta ao impulso resultante $h'(n)$ se tornam

$$h'(n) = h(n) \cdot \omega(n). \quad (7)$$

Considerando o domínio da frequência, a multiplicação da equação anterior corresponde a uma convolução, representada na seguinte equação:

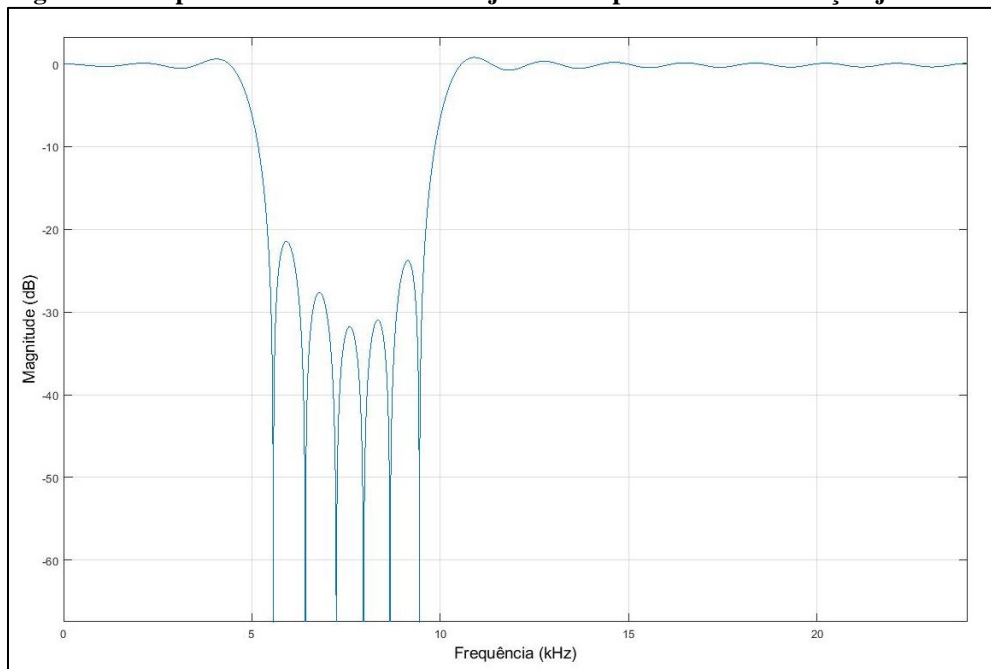
$$H'(e^{j\omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(e^{j\omega'}) W(e^{j(\omega-\omega')}) d\omega', \quad (8)$$

onde $H(e^{j\omega})$ corresponde à resposta na frequência do filtro ideal e $W(e^{j\omega})$ à função-janela.

Nota-se então, que uma janela interessante seria uma sequência de comprimentos finitos cuja resposta em frequência quando convoluída com resposta na frequência ideal produziria menor distorção possível (DINIZ; SILVA; NETTO, 2014).

Na ilustra-se Figura 8 um filtro rejeita-faixa pelo método de Janela Retangular de ordem $M=50$ com frequências de corte de 5 kHz e 10 kHz e frequência de amostragem de 48 kHz. Nota-se uma ondulação de elevada amplitude próxima às extremidades da faixa de passagem, comumente chamadas de oscilações de Gibbs. Elas são resultantes do truncamento dos coeficientes e não se reduzem ainda que a ordem M do filtro seja aumentada em muitas vezes. Esse fenômeno se deve à lenta convergência da série de Fourier quando se aproxima se está a um ponto de descontinuidade da função $h(n)$ (DINIZ; SILVA; NETTO, 2014).

Figura 8 – Resposta ao módulo de filtro rejeita-faixa pelo método de função-janela.



Fonte: Adaptado de Diniz, Silva e Netto (2014).

Existem várias possibilidades de janelas para este método, algumas das mais comuns são definidas pelas seguintes equações:

Retangular

$$\omega_r(n) = \begin{cases} 1, & |n| \leq M/2 \\ 0, & |n| > M/2 \end{cases} \quad (9)$$

Triangular ou Janela de Barlett

$$\omega_{tB}(n) = \begin{cases} \frac{2n}{M}, & |n| \leq \frac{M}{2}, M \text{ par} \\ 2 - \frac{2n}{M}, & \frac{M}{2} < n \leq M \\ 0, & \text{caso contrário} \end{cases} \quad (10)$$

Hann e Hamming

$$\omega_H(n) = \begin{cases} \alpha + (1 + \alpha) \cos\left(\frac{2\pi n}{M}\right), & |n| \leq M/2 \\ 0, & \text{caso contrário} \end{cases} \quad (11)$$

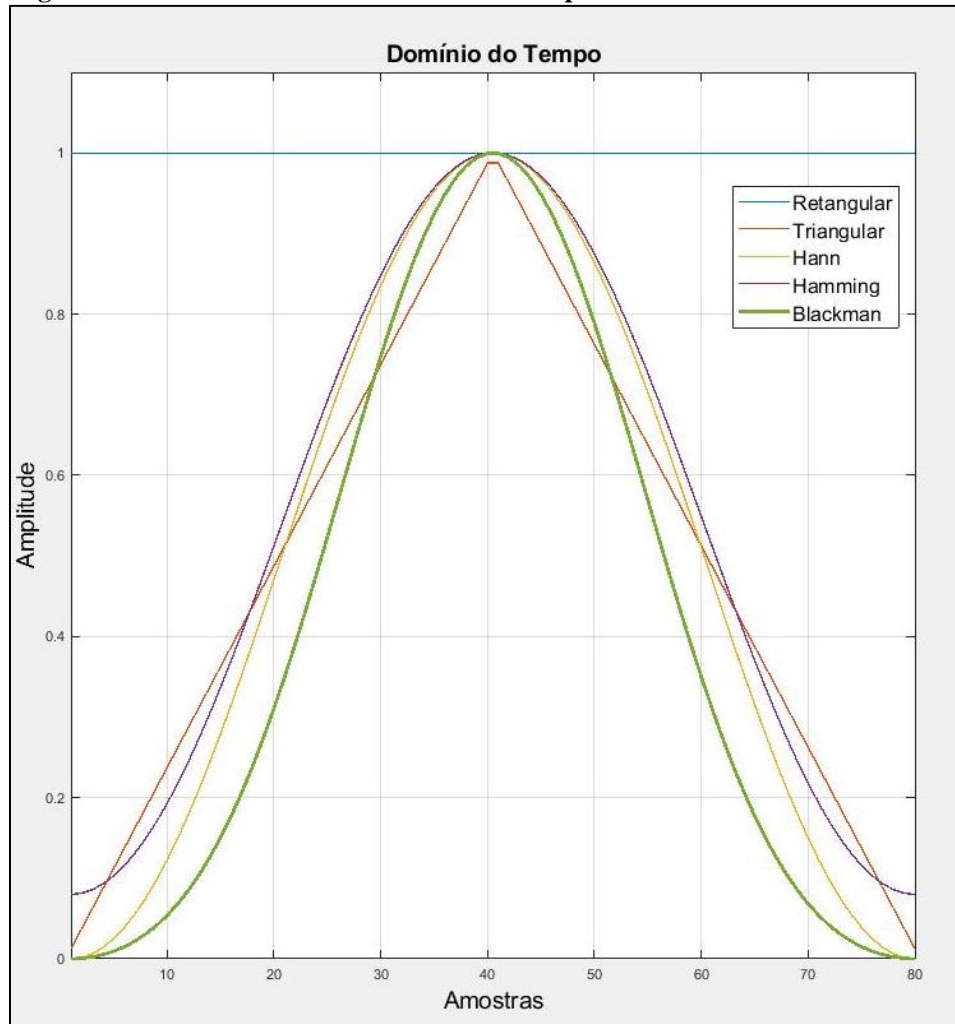
onde na Janela de Hann $\alpha=0,5$ e na janela de Hamming $\alpha=0,54$.

Blackman

$$\omega_B(n) = \begin{cases} 0,42 + 0,5\cos\left(\frac{2\pi n}{M}\right) + 0,08\cos\left(\frac{4\pi n}{M}\right), & |n| \leq \frac{M}{2} \\ 0, & \text{caso contrário} \end{cases} \quad (12)$$

As diferenças entre as janelas podem ser observadas na Figura 9.

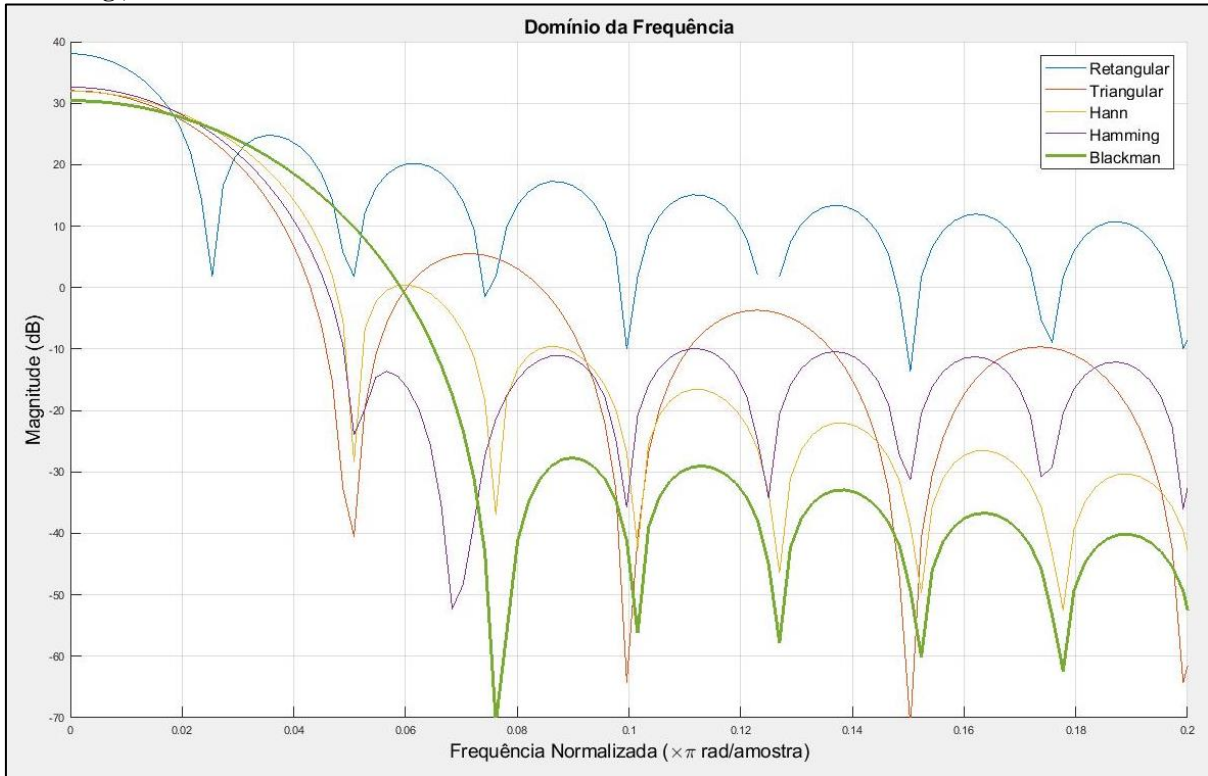
Figura 9 - Formas de Janelas no domínio do tempo



Fonte: Autoria própria.

Para uma função $20 \log_{10} |W(e^{j\omega})|$ e $M=80$ são mostrados os gráficos da Figura 10 para cada uma dessas janelas. (OPPENHEIM, 1997).

Figura 10 – Detalhe da transformada de Fourier com $M=80$ das janelas: Retangular, Triangular, Hamming, Hann e Blackman



Fonte: Autoria própria.

A janela retangular deverá gerar transições mais abruptas do que as janelas triangulares, de Hann, de Hamming e Blackman. Nesta a ordem, as ondulações de passagem são cada vez menores e a atenuação da faixa de rejeição cada vez maior. Porém é na ordem inversa que se tem as menores faixas de transições.

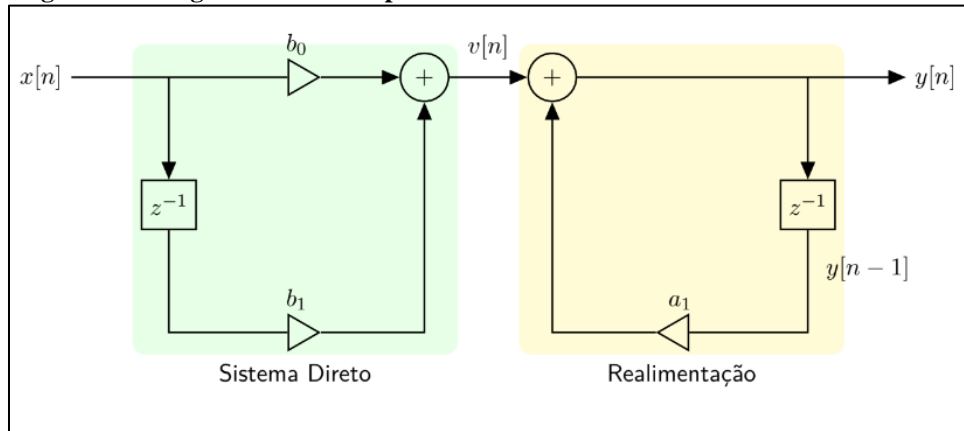
2.4 FILTROS IIR

Filtros IIR podem ser definidos como filtros com resposta ao impulso de duração infinita. Estes filtros são capazes de aproximar uma resposta em frequência com um número de multiplicações inferior aos filtros FIR, portanto são mais indicados para aplicações mais práticas, tais como processamento de sinais em tempo-real (DINIZ; SILVA; NETTO, 2014). Estes filtros se caracterizam pela realimentação, o que os diferencia dos filtros FIR. A equação geral da saída do sistema é dada por:

$$y[n] = \sum_{l=1}^N a_l y[n-l] + \sum_{k=0}^M b_k x[n-k] \quad (13)$$

onde a_l representa os coeficientes de realimentação e N a ordem do sistema de realimentação e b_k representa os coeficientes do sistema direto e M a ordem do sistema direto. A Figura 11 ilustra um diagrama para um filtro IIR de ordem 1.

Figura 11 - Diagrama de Blocos para Filtro IIR de ordem 1

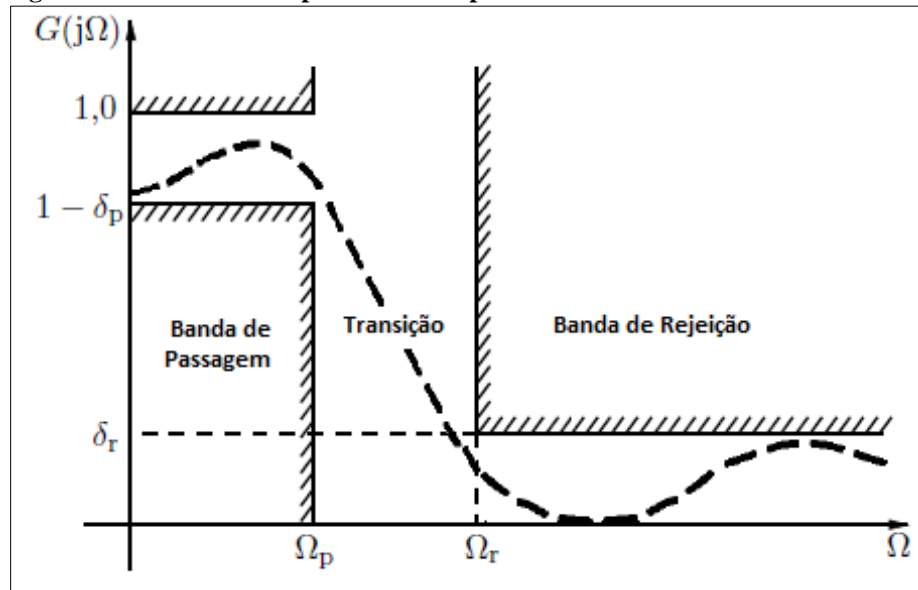


Fonte: Adaptado de Diniz, Silva e Netto (2014).

A ordem N do sistema de realimentação, em geral, define a ordem do filtro, pois M é muito menor que N . Pode-se considerar nestes casos que $M=N$ e que os coeficientes b_{M+1} até b_N sejam iguais a zero.

Segundo Diniz, Silva e Netto, um filtro é usualmente especificado através de uma região do plano $\Omega \times H(j\Omega)$, onde sua resposta na frequência tem que estar contida. A Figura 12 mostra este plano para um filtro passa-baixas.

Figura 12 - Ganho em dB para um filtro passa-baixas



Fonte: Adaptado de Diniz, Silva e Netto (2014).

As frequências Ω_p e Ω_r denotam, respectivamente, as frequências das extremidades da faixa de passagem e da faixa de rejeição. A região de frequências compreendidas entre estes é a chamada faixa de transição, para a qual não é fornecida qualquer especificação.

No eixo vertical, encontram-se os ganhos G para cada frequência. Este ganho pode apresentar ondulações, cujas amplitudes máximas na faixa de passagem e na faixa de rejeição são denotadas por δ_p e δ_r , respectivamente.

O ganho na banda de passagem G_p é dado por:

$$G_p = 20 * \log_{10}(1 - \delta_p). \quad (14)$$

Em termos de atenuação da banda de passagem (A_p),

$$A_p = -G_p. \quad (15)$$

O ganho da banda de rejeição (G_r) é dado por:

$$G_r = 20 * \log_{10}(\delta_r), \quad (16)$$

e a atenuação da banda de rejeição é:

$$A_r = -G_r. \quad (17)$$

Idealmente, um filtro passa-baixas tem ganho unitário na banda de passagem e nulo na banda de rejeição

2.4.1 Aproximações para filtros IIR

Os tipos de filtros IIR abordados neste trabalho são aproximações de filtros analógicos. Estas aproximações possuem frequência normalizada, ou seja, as frequências analógicas (Ω) são escalonadas em função da frequência máxima da banda de passagem (Ω_p):

$$\Omega' = \frac{\Omega}{\Omega_p}, \quad (18)$$

onde Ω' denota uma frequência analógica normalizada. O filtro original pode ser obtido pela operação reversa.

As aproximações de filtros analógicos mais usuais para filtros IIR são as de Butterworth, Chebyshev e Elíptica. As atenuações dessas aproximações podem ser descritas pela equação seguinte:

$$|A(j\Omega')|^2 = 1 + |E(j\Omega')|^2, \quad (19)$$

sendo $A(j\Omega')$ a função de atenuação desejada e $E(j\Omega')$ é um polinômio característico da aproximação utilizada.

Para a aproximação de Butterworth, a atenuação aumenta de acordo com a frequência, de forma lenta na faixa de passagem e rapidamente na faixa de rejeição.

O polinômio característico para essa aproximação pode ser descrito pela seguinte equação

$$E(\Omega') = \varepsilon(j\Omega')^N, \quad (20)$$

sendo ε uma constante e N a ordem do filtro.

Neste tipo de aproximação, caso se deseje aumentar a atenuação da faixa de rejeição ou diminuir a faixa de transição, é preciso aumentar a ordem do filtro. Se comparados com outros filtros IIR da mesma ordem, este filtro possui uma faixa de transição maior, porém sem ondulações na banda de passagem.

As aproximações de Chebyshev possuem transições mais íngremes para as faixas de passagem e rejeição do que as aproximações de Butterworth, pois os filtros de Chebyshev, em grande parte, requerem funções de transferência de menores ordens. Os filtros de Chebyshev de Tipo 1 possuem maior ondulação na faixa de passagem, enquanto os filtros de Tipo 2 apresentam ondulações na faixa de rejeição.

O polinômio característico para a aproximação de Chebyshev Tipo 1 pode ser descrito pela seguinte equação

$$E(\Omega') = \varepsilon Cn(\Omega'), \quad (21)$$

sendo $Cn(\Omega')$ a função de Chebyshev de ordem N , dada por:

$$Cn(\Omega') = \begin{cases} \cos(n \cos^{-1}\Omega'), & 0 \leq \Omega' \leq 1 \\ \cosh(n \cosh^{-1}\Omega'), & \Omega' > 1 \end{cases}. \quad (22)$$

O polinômio característico para a aproximação de Chebyshev Tipo 2 pode ser descrito pela seguinte equação

$$E(\Omega') = \frac{Cn(\Omega's)}{Cn(\frac{\Omega's}{\Omega'})}. \quad (23)$$

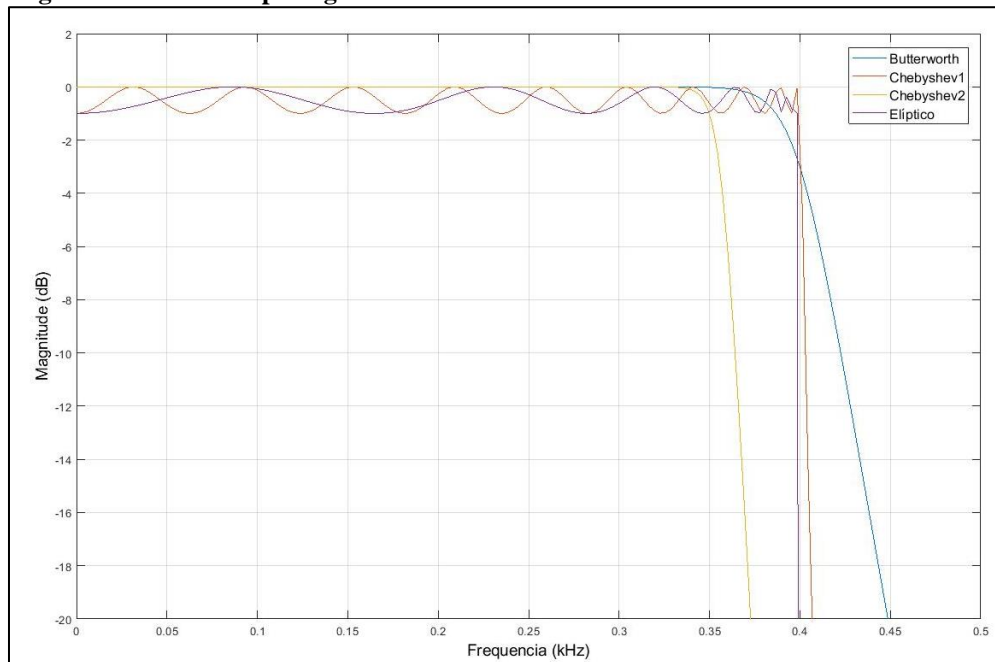
Já a aproximação elíptica caracteriza-se pela ondulação ajustável para as faixas de passagem e de rejeição, bem como pelo maior decaimento quando comparada às demais aproximações, sendo $Rn(\Omega')$ uma função racional elíptica.

O polinômio característico para essa aproximação pode ser descrito pela equação

$$E(\Omega') = \varepsilon Rn(\Omega'), \quad (24)$$

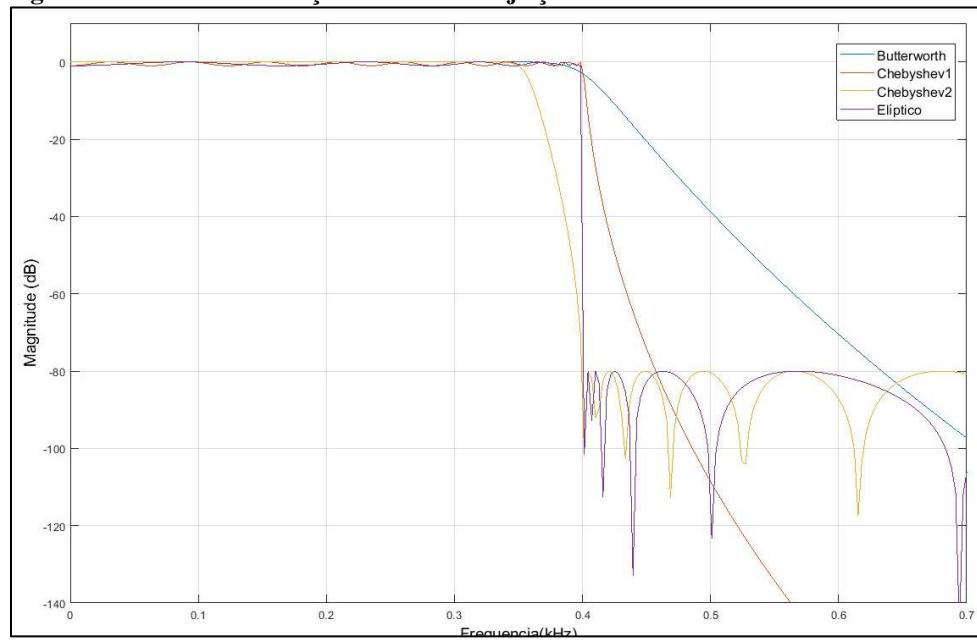
A Figura 13 mostra o detalhe da banda de passagem de um filtro passa-baixas de 400Hz de ordem 20 para cada tipo de filtro IIR citado acima.

Figura 13 - Banda de passagem dos filtros IIR



Fonte: Autoria própria.

Nota-se a existência de oscilações na banda de passagem tanto para o filtro do tipo Chebyshev 1 quanto para o do tipo Elíptico enquanto os do tipo Butterworth e Chebyshev2 têm ganhos quase que constantes. Na frequência de corte é possível verificar que o filtro do tipo Elíptico decai de maneira mais íngreme que os demais, enquanto o Butterworth se mostra mais lento. No filtro do tipo Chebyshev 2, configurou-se uma atenuação de banda de rejeição de 80dB, o que resultou numa banda de transição tal que a atenuação do sinal começa muito antes dos demais filtros, afim de garantir a atenuação configurada em 400Hz. A Figura 14 mostra a diferença nos tamanhos da faixa de transição, sendo a do filtro Butterworth a maior, e também as oscilações da banda de rejeição dos filtros do tipo Chebyshev 2 e Elíptico.

Figura 14 - Faixa de transição e banda de rejeição de filtros IIR

Fonte: Autoria própria.

3 DESIGN DE FILTROS DIGITAIS UTILIZANDO *HARDWARE* RASPBERRY PI E *SOFTWARE* MATLAB-SIMULINK

3.1 DESCRIÇÃO DO SISTEMA IMPLEMENTADO

O Raspberry Pi é um microcomputador em uma única placa com dimensões próximas a de um cartão de crédito, capaz de realizar tarefas básicas como acessar a internet via um *browser*, reproduzir vídeos, criar e editar documentos, etc. (ARTHUR, James)

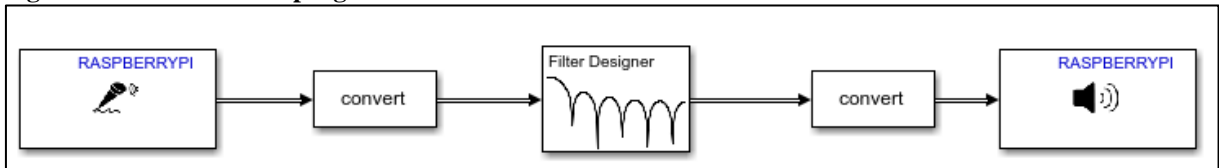
Como este trabalho tem como foco o desenvolvimento de filtros digitais utilizando os métodos de filtragem FIR e IIR, para a implementação desses métodos, o *software* Matlab foi utilizado, sobretudo com a ferramenta Simulink. Como ferramenta para modelagem, o Simulink oferece um editor gráfico para construção de modelos como diagramas de blocos, permitindo simples construção como se estivesse esboçando um sistema como um simples desenho. Além disso, uma biblioteca é disponibilizada com vários blocos, tais como fontes, componentes lineares e não-lineares. Caso seja necessário, ainda é permitida ao usuário a criação de um bloco específico para atender às necessidades de cada modelagem.

O sistema implementado consiste na utilização de blocos de programa para *design* de filtro digital utilizando o Matlab. Através deste bloco, o usuário pode configurar:

- Seleção do tipo de filtro: passa-baixas, passa-altas, passa-faixa e ou rejeita-faixa;
- Seleção das faixas de frequência: frequência de corte, faixa de transição;
- Ordem do filtro;
- Atenuações de bandas de passagem e rejeição.

O sistema conta com blocos de programa já existentes: *ALSA Audio Capture* e *ALSA Audio Playback* da biblioteca RASPBERRYPI que fazem a interface com o mundo real e o bloco *convert* de conversão de tipo de dado (Figura 15).

Figura 15 - Blocos de um programa no Matlab/Simulink



Fonte: Autoria Própria

As configurações dos filtros são feitas nos blocos de *Design* de Filtro do Simulink. A partir do diagrama criado, é gerado um programa no Raspberry Pi, que faz o processamento dos sinais. Pode-se alterar alguns parâmetros de forma *online* via Simulink, porém é possível trabalhar apenas com o Raspberry Pi se não forem necessárias alterações.

No Apêndice A estão descritos os passos para configurar e comunicar os dispositivos, sendo estes o sistema operacional do Raspberry Pi, o adaptador de áudio USB e Matlab/Simulink, partindo do pressuposto que já se tenha instalado no computador o Matlab com Simulink na versão 2017b ou mais recente e que todos os equipamentos citados anteriormente estejam disponíveis.

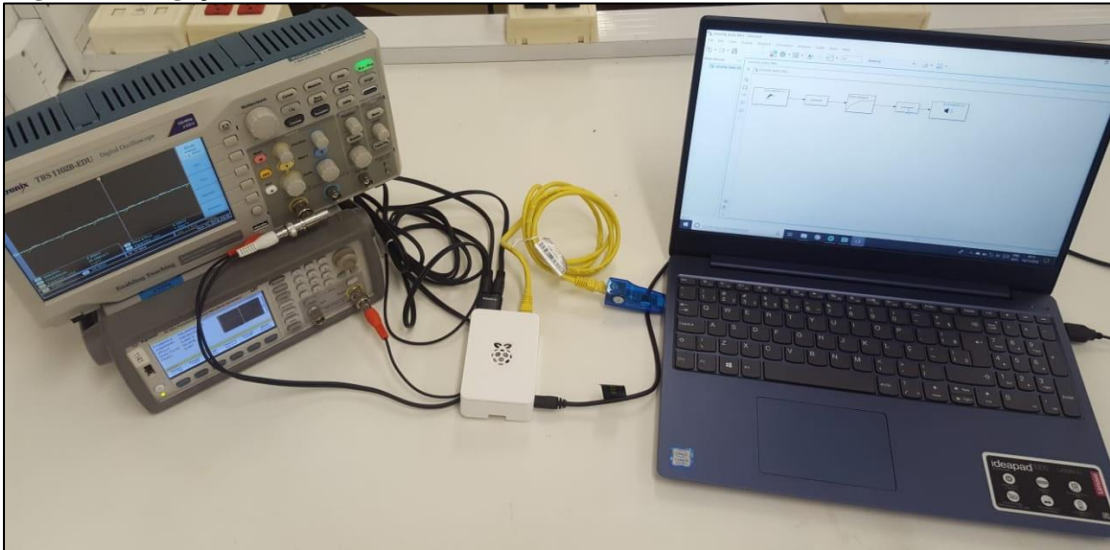
Após as configurações necessárias, foram feitas implementações para filtros FIR e IIR utilizando os métodos de filtragem abordados no Capítulo 2. A partir dessas implementações, foi feita uma demonstração em sala de aula, na disciplina de Sinais e Sistemas 2, na UTFPR, ministrada pelo Prof. Glauber Gomes de Oliveira Brante. Em seguida, utilizando os blocos de programação implementados, foi construído um equalizador de frequências.

As etapas de implementação dos filtros digitais, a demonstração em sala de aula e a construção do equalizador a partir da combinação de blocos de programação são abordados a seguir.

3.2 IMPLEMENTAÇÕES DE FILTROS DIGITAIS

A partir da Figura 1, foi possível conectar todos os dispositivos em bancada, como mostra a Figura 16, para que testes fossem realizados.

Figura 16 – Ligações em bancada.



Fonte: Autoria Própria.

Após realizar as conexões dos dispositivos foram realizadas simulações com a finalidade de testar tanto a comunicação entre dispositivos, funcionalidade do programa gerado pelo Matlab/Simulink e do *hardware*.

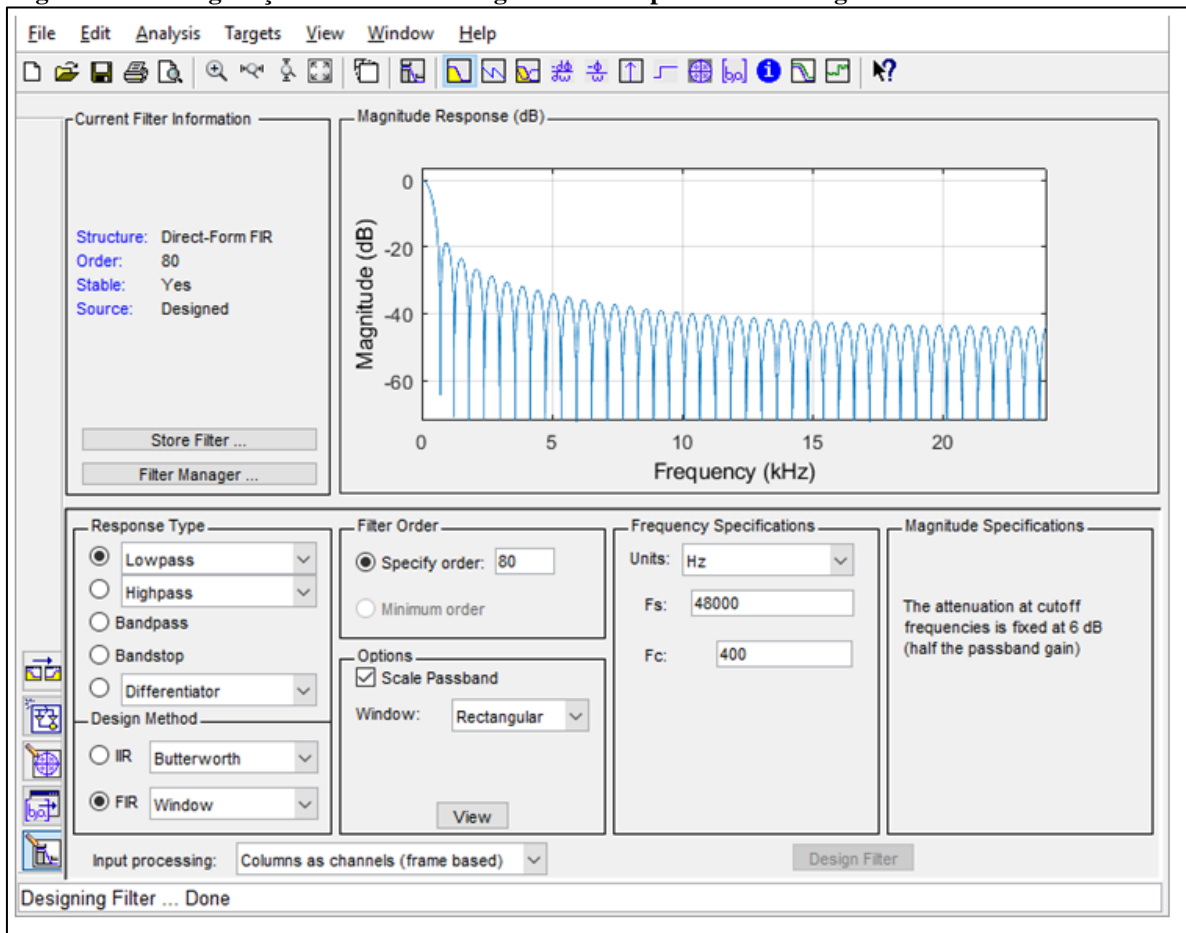
Em seguida, no bloco de *design* de filtro da biblioteca *DSP System Toolbox* do Simulink, foram configurados e testados os filtros de passa-baixa, passa-alta, passa-faixa e rejeita-faixa utilizando os métodos de filtragem abordados no Capítulo 2.

3.2.1 Exemplo de Implementação

Para os filtros FIR, foram implementados os métodos de Janela Retangular, Janela Triangular, Hann, Hamming e Blackman. Para os filtros IIR, foram implementados as aproximações de Butterworth e Chebyshev tipo 1 e tipo 2.

Como exemplo, está descrita a seguir uma implementação de filtro FIR para o método de Janela Retangular de ordem 80. As configurações do bloco de *design* para um filtro passa-baixa do tipo Janela Retangular pode ser observado na Figura 17. Nota-se que a frequência de corte para o filtro configurado é de 400 Hz e a frequência de amostragem é de 48 kHz, que foi a mesma utilizada para todos os filtros implementados.

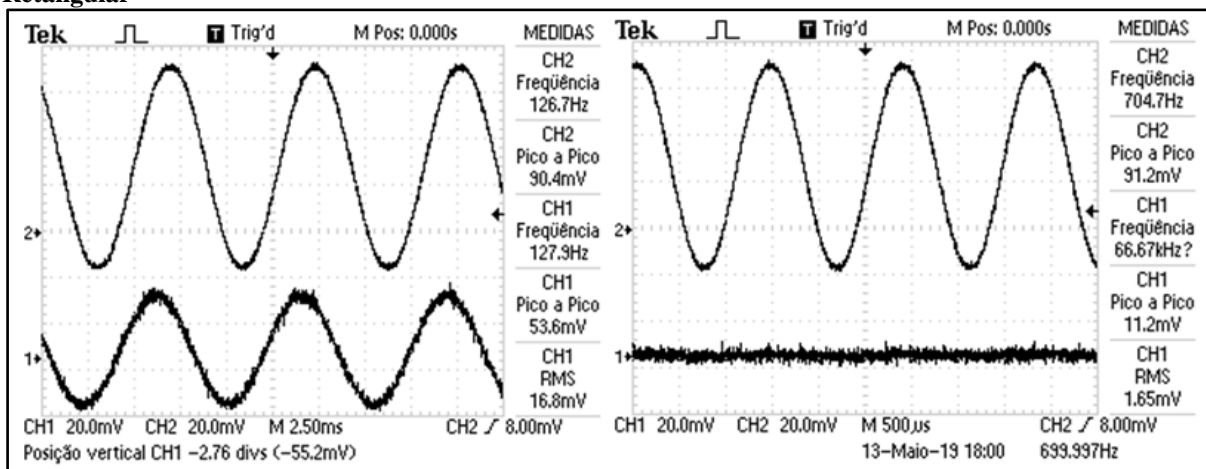
Figura 17 - Configurações de bloco de design de Filtro tipo Janela Retangular



Fonte: Autoria própria

Na Figura 18 pode-se observar os sinais de entrada e sinais filtrados para ondas senoidais, sendo o primeiro um sinal de frequência de 126,7 Hz e o segundo um sinal de frequência de 704,7 Hz.

Figura 18 – Sinais de onda senoidal de entrada e saída para filtro passa-baixa do tipo FIR - Janela Retangular



Fonte: Autoria própria

Nota-se que, como a frequência de corte adotada foi de 400Hz, para o sinal de

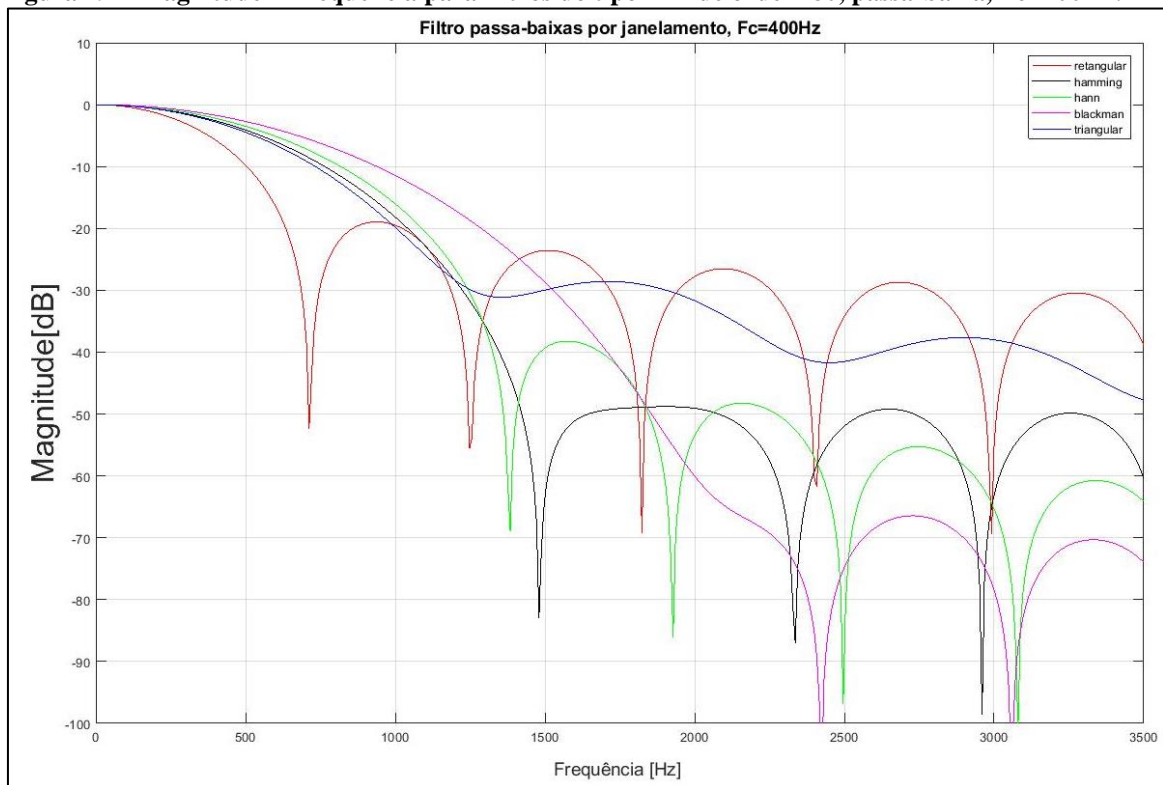
126,7 Hz, é perceptível um sinal de saída de mesmo formato de onda que a entrada. Já para o sinal de 704,7 Hz, observa-se que o sinal foi atenuado de maneira que não se pode observar na saída a frequência do sinal entrada, o que é esperado para o filtro passa-baixa adotado.

3.2.2 Implementação de filtros tipo FIR

A partir de implementações para filtros FIR, foi possível verificar o comportamento de cada tipo de método por função-janela, bem como a variação no comportamento do filtro para diferentes ordens adotadas.

Um exemplo das variações do comportamento dos filtros pode ser observada na Figura 19, que ilustra o gráfico de Magnitude x Frequência para filtros passa-baixas, com frequência de corte de 400Hz, implementados para todos os métodos de filtro FIR em ordem 80.

Figura 19 – Magnitude x Frequência para filtros do tipo FIR de ordem 80, passa-baixa, $F_c=400\text{Hz}$.



Fonte: Autoria própria.

A partir do gráfico, escolheu-se algumas frequências que apresentassem as principais diferenças entre os filtros para verificar a saída do sistema. A Tabela 1, mostra as frequências escolhidas e a amplitude da saída para cada tipo de filtro FIR ao se aplicar uma onda senoidal

de tensão pico-a-pico de 184mV.

Tabela 1 - Amplitudes de sinais filtrados por janelamento

Frequência de entrada [Hz]	Retangular n=16		Retangular = 80		Triangular n=80	
	Vpp [mV]	Vrms [mV]	Vpp [mV]	Vrms [mV]	Vpp [mV]	Vrms [mV]
400	244	86	128		184	63
712	232	80	8	0	88	29,5
984	212	73	32	9,5	32	9,7
1248	184	64,5	8	0	16	3,45
1384	172	59	16	5	16	3,2
1480	164	56,8	20	5,76	16	12
1824	124	43	8	0	16	3,6
2416	60	19,7	8	0	8	1
Frequência de entrada [Hz]	Hamming		Hann		Blackman	
	Vpp [mV]	Vrms [mV]	Vpp [mV]	Vrms [mV]	Vpp [mV]	Vrms [mV]
400	192	65,7	200	68,7	208	72,8
712	96	33	112	37,8	136	46,1
984	40	11,6	48	15	72	24,5
1248	12	2,95	12	3,3	32	9,9
1384	8	0,7	8	0	24	6
1480	8	0	8	1	16	3,7
1824	8	0	8	0	8	0
2416	8	0	8	0	8	0

O gráfico tem destacado os melhores valores de atenuação para as frequências acima da frequência de corte. Comparando os dados da Tabela 1 e da Figura 19, tem-se que o filtro do tipo janela retangular apresentou a melhor atenuação para uma frequência mais próxima da frequência de corte, porém ao aumentar a frequência do sinal de entrada, reapareceram alguns sinais atenuados na saída. O filtro do tipo triangular não obteve em nenhum momento a mesma atenuação atingida pelos demais filtros até a frequência máxima testada de 2416Hz. Como esperado, as janelas de Hann, Hamming e Blackman possuem, nesta ordem, uma atenuação mais lenta do que a janela retangular, mas por outro lado apresentam uma atenuação efetiva a partir de certa frequência, o que não é possível no filtro de janela retangular da mesma ordem para este sinal de entrada. O filtro de ordem mais baixa é o que possui a maior faixa de transição sendo que a primeira atenuação eficaz do sinal ocorre em uma frequência maior do que

3.2.3 Implementação de filtros tipo IIR

Nas implementações de filtros IIR pode-se verificar algumas diferenças nas configurações do bloco de *design*. Para estes filtros, não é necessário informar a ordem do filtro, pois, caso se deseje trabalhar com a ordem mínima, devem apenas ser informados os valores para frequência de passagem (F_{pass}) e frequência de rejeição (F_{stop}) e a atenuação de banda de passagem (A_{pass}) e rejeição (A_{stop}), conforme a Figura 20.

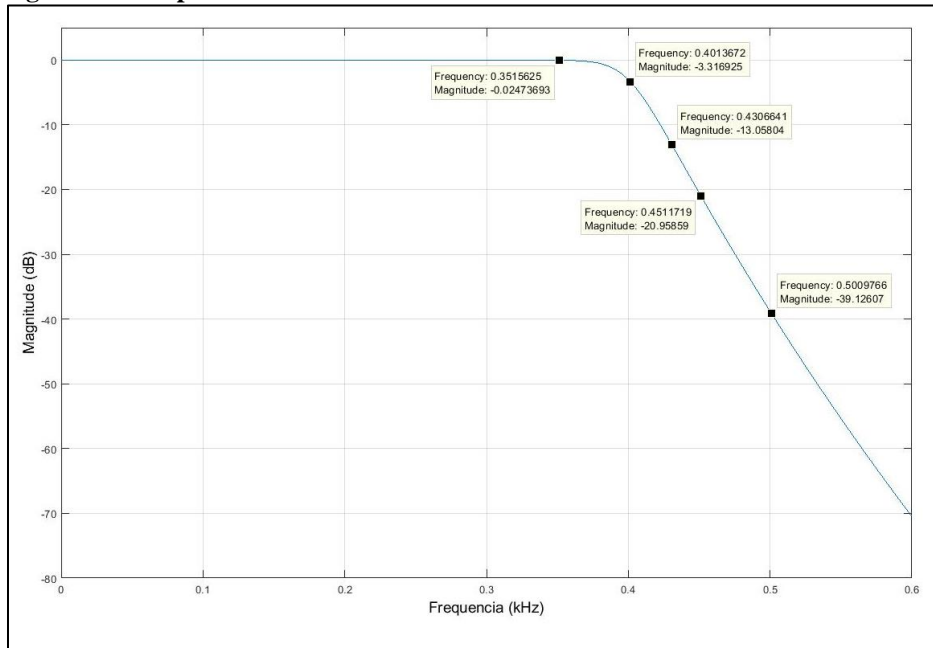
Figura 20 – Detalhe do bloco de design para implementação de Filtro IIR

Response Type	Filter Order	Frequency Specifications	Magnitude Specifications
<input checked="" type="radio"/> Lowpass <input type="radio"/> Highpass <input type="radio"/> Bandpass <input type="radio"/> Bandstop <input type="radio"/> Differentiator	<input type="radio"/> Specify order: 80 <input checked="" type="radio"/> Minimum order	Units: Hz Fs: 48000 Fpass: 400 Fstop: 500	Units: dB Apass: 1 Astop: 80
Design Method: <input checked="" type="radio"/> IIR Butterworth <input type="radio"/> FIR Least-squares	Options: Match exactly: passband		
Input processing: Columns as channels (frame based)		Design Filter	

Fonte: Autoria própria

Assim como na implementação dos filtros FIR, para cada tipo de filtro IIR foram escolhidas algumas frequências com base nas suas respostas em frequência que pudessem comprovar suas principais características. Todos os filtros implementados tinham a frequência de corte igual à 400Hz e ordem 20.

Para o filtro Butterworth, a Figura 21 apresenta a resposta na frequência do filtro projetados e marca alguns pontos escolhidos para verificar uma atenuação mais lenta se comparada com os demais filtros IIR.

Figura 21 - Resposta do filtro Butterworth

Fonte: Autoria própria

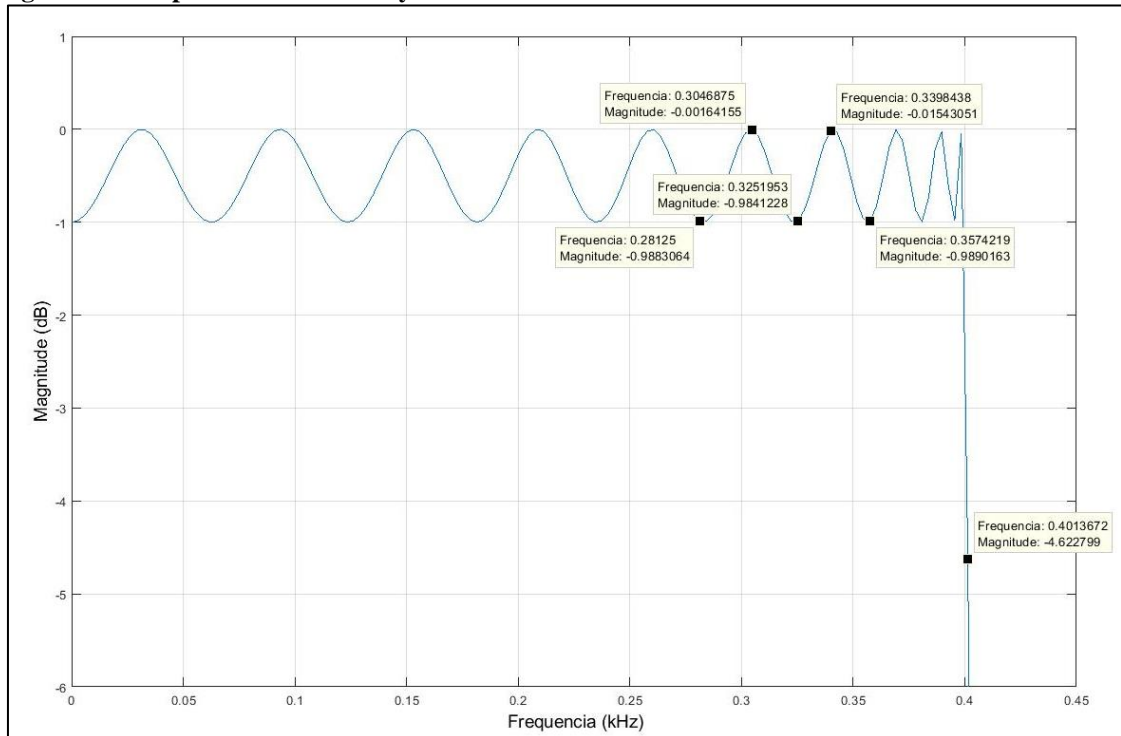
A partir destas frequências, sinais senoidais de tensão pico-a-pico de 180mV foram gerados na entrada do sistema e o valor das ondas da saída são mostrados na Tabela 2.

Tabela 2 – Valores

Freq [Hz}	Vpp [mV]	Mag [dB]
350	176	-0,1952
400	124	-3,23702
430	46	-11,8503
450	20	-19,0849
≥500	8	-27,0437

Para o filtro Chebyshev do tipo 1, buscou-se pelas oscilações que ocorrem na banda de passagem que são características deste tipo de filtro, estas limitadas à 1dB, como mostra a Figura 22.

Figura 22 - Resposta do filtro Chebyshev 1



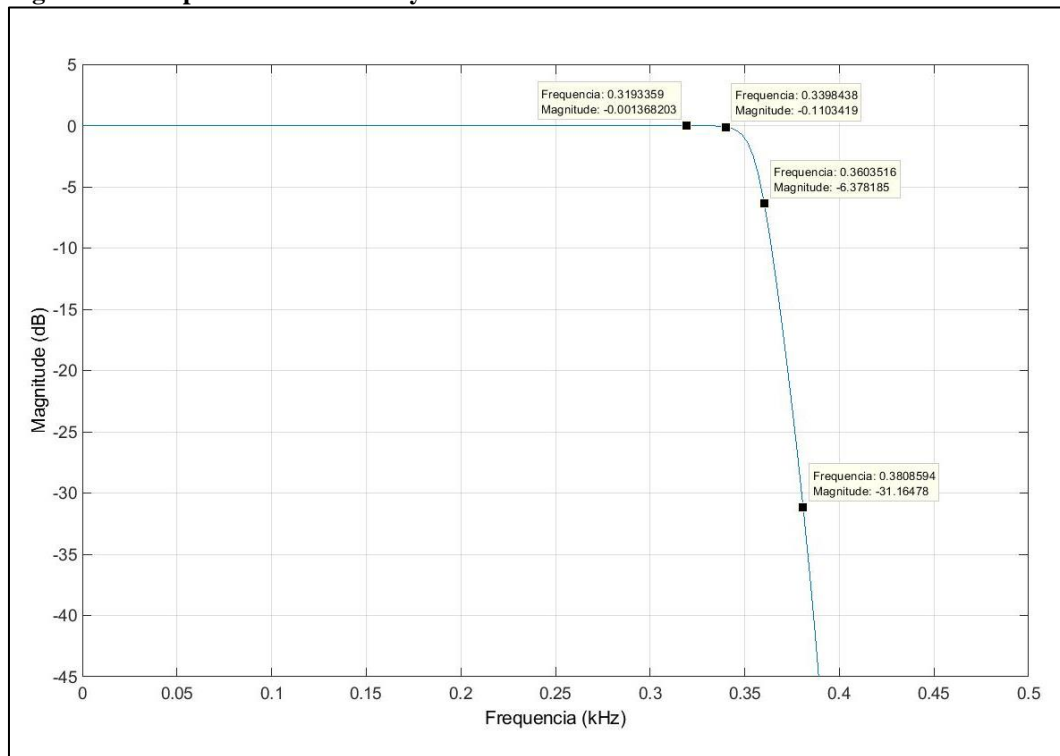
Fonte: Autoria própria

Ao aplicar ondas senoidais com tensão pico-a-pico de 184mV nas frequências onde ocorrem alguns picos sequenciais, obteve-se os resultados da Tabela 3. Obteve-se então uma oscilação na magnitude das frequências da banda de passagem, como esperado.

Tabela 3 - Valores Chebyshev 1

Freq [Hz]	Vpp [mV]	Mag (dB)
280	148	-1,89112215
305	180	-0,19090636
324	156	-1,43386449
340	180	-0,19090636
356	156	-1,43386449
370	176	-0,3861031
400	164	-0,9994795
406	24	-17,6921316
≥418	8	-27,2345567

Diferentemente do filtro Chebyshev do tipo 1, o filtro Chebyshev tipo 2 não deve possuir oscilações na banda de passagem, como mostra a Figura 23.

Figura 23 - Resposta do filtro Chebyshev 2

Fonte: Autoria própria

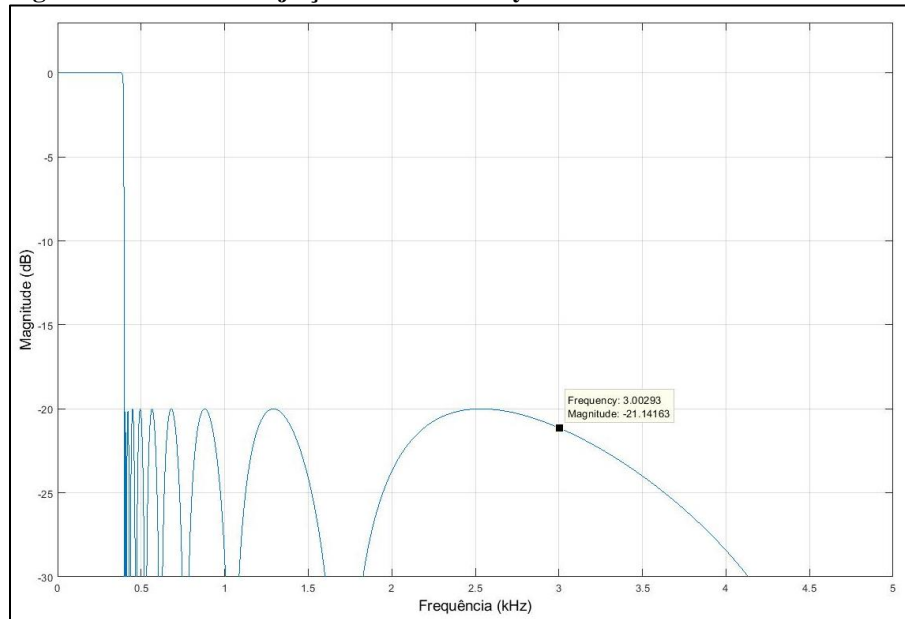
Além disso, observa-se a atenuação das frequências menores que 400Hz. Isso se deve à grande atenuação da banda de rejeição configurada (80dB). Os resultados obtidos são apresentados na Tabela 4.

Tabela 4 - Valores Chebyshev 2

Freq [Hz]	Vpp [mV]	Mag [dB]
320	180	-0,19091
340	176	-0,3861
360	96	-5,65093
380	12	-23,7127
≥400	8	-27,2346

A partir de 400Hz o sinal da saída do sistema não apresenta a frequência do sinal de entrada pelo filtro. Para este tipo de filtro, ocorrem oscilações na banda de rejeição que não podem ser observadas para uma atenuação tão elevada da banda de rejeição. Para verificar a existências destas oscilações, diminuiu-se essa atenuação para 20 dB, e a resposta do filtro é conforme a Figura 24.

Figura 24 - Banda de rejeição do filtro Chebyshev 2

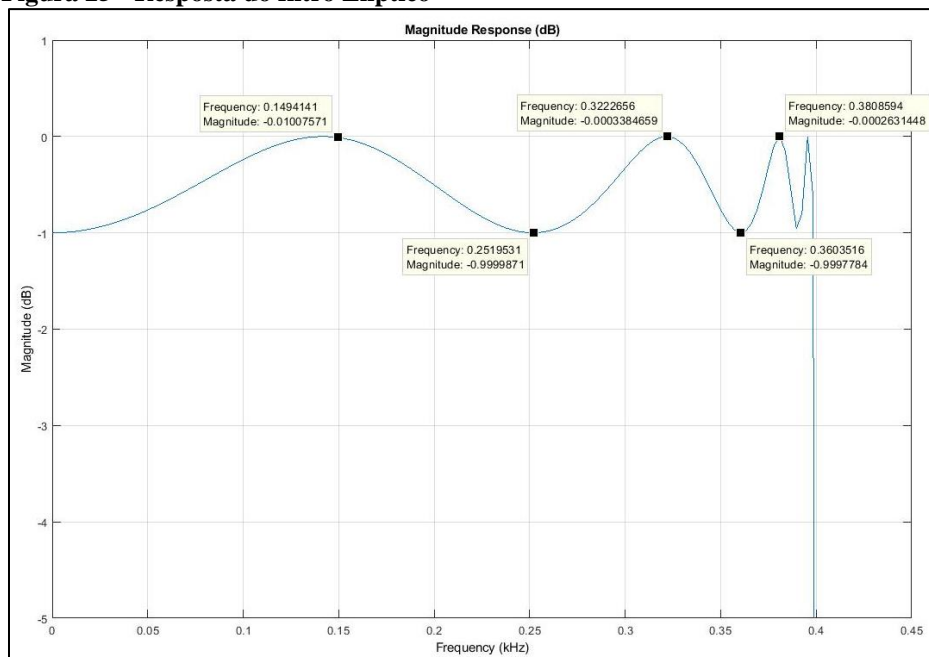


Fonte: Autoria própria

Para verificar a existência de um sinal na saída, aplicou-se uma onda de 3 kHz e observou-se uma onda de amplitude de 10mV na saída, ou seja, o sinal foi atenuado em aproximadamente 19,3 dB, próximo ao sugerido pela resposta gráfica.

Para o filtro do tipo Elíptico, tem-se oscilações tanto na banda de passagem como na banda de rejeição, porém a faixa de transição é a menor de todos os filtros implementados, conforme Figura 25.

Figura 25 - Resposta do filtro Elíptico



Fonte: Autoria própria

Aplicou-se as frequências marcadas e obteve-se a resposta conforme a Tabela 5. Pelos resultados obtidos, observa-se a ocorrência das oscilações na banda de passagem, porém como a atenuação da banda de rejeição configurada era muito elevada (40 dB), todas as frequências acima de 400Hz foram eliminadas neste caso

Tabela 5 - Valores Elíptico

Freq [Hz}	Vpp [mV]	Mag [dB]
150	176	0
250	156	-1,04776
322	172	-0,19968
360	124	-3,04182
380	172	-0,19968
400	8	-26,8485
>400	8	-26,8485

A partir desses experimentos, foi possível verificar as particularidades de cada filtro implementado: a lenta atenuação do filtro Butterworth, as oscilações da banda de passagem dos filtros Chebyshev tipo 1 e Elíptico, que diferem entre si pelo número de oscilações, as oscilações da banda de rejeição do filtro chebyshev tipo 2 e a atenuação das frequências antes da frequência de parada.

4 APLICAÇÕES PRÁTICAS DO SISTEMA IMPLEMENTADO

4.1 UTILIZAÇÃO DE *DESIGN* DE FILTROS CRIADO COMO RECURSO DIDÁTICO

Na disciplina de Sinais e Sistemas 2, da UTFPR, ministrada pelo Prof. Glauber Gomes de Oliveira Brante, foram apresentadas algumas implementações dos filtros FIR e IIR, a fim de disponibilizar aos alunos uma experiência prática de métodos de filtragem através do Matlab/Simulink e Raspberry Pi.

Durante as atividades práticas, foi solicitado aos alunos que avaliassem o recurso utilizado a fim de compreender sobre a vantagem didática em se utilizar recursos *hands-on* em disciplinas de Sinais e Sistemas.

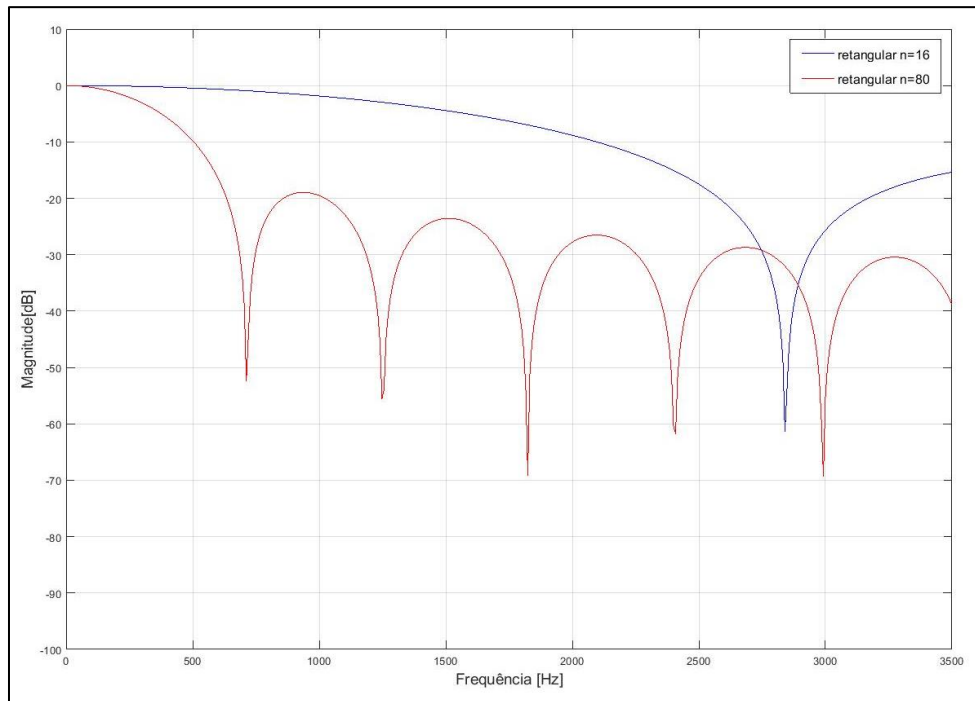
4.1.1 Demonstração de experimentos na disciplina de Sinais e Sistemas 2

Nesta demonstração foram realizados três experimentos, com montagem dos equipamentos similar à Figura 1, sendo os dois primeiros utilizando sinais determinísticos, ondas senoidal e quadrada, e o terceiro utilizando sinais de áudio. O roteiro da demonstração utilizado em sala de aula está disposto no Apêndice B.

O primeiro experimento consistiu na implementação de filtros FIR passa-baixas pelos métodos de Janela Retangular, ordem 16 e 80, Janela de Hamming de ordem 80 e de um filtro IIR pelo método de Butterworth de ordem mínima.

Neste experimento, pode-se observar diferenças significantes na saída de filtros para ordens diferentes, conforme a Figura 26, que mostra o comportamento de filtro pelo método de Janela Retangular de ordem 16 e de ordem 80.

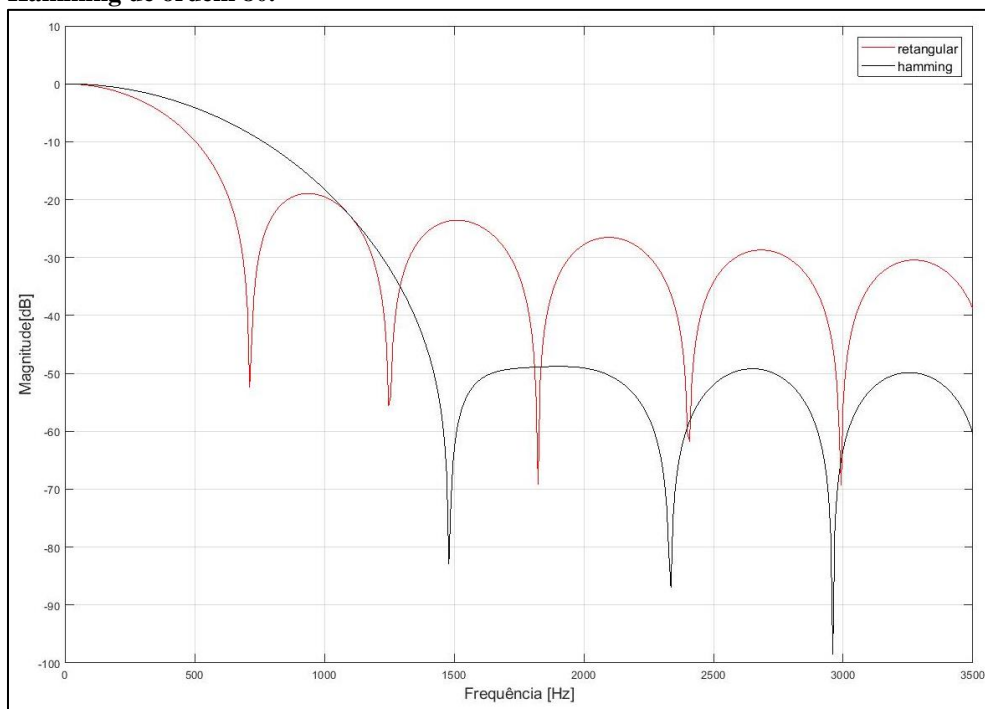
Figura 26 – Magnitude x Frequência para filtros pelo método de Janela Retangular de ordens 16 e 80.



Fonte: Autoria própria

Em seguida, com a implementação de um filtro pelo método de janela de Hamming de ordem 80, pode-se observar as diferenças para os dois métodos de mesma ordem. O gráfico de Magnitude x Frequência para os dois métodos pode ser observado na Figura 27.

Figura 27 - Magnitude x Frequência para filtros pelo método de Janela Retangular e Hamming de ordem 80.

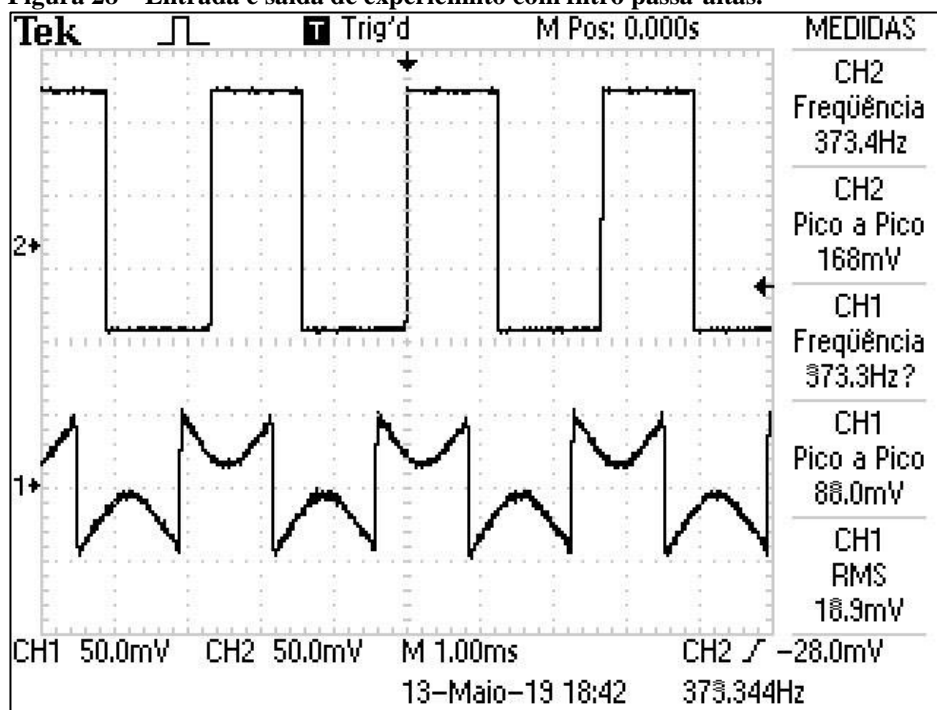


Fonte: Autoria própria.

Por fim, com a implementação de um filtro IIR de ordem mínima, pode-se observar que, para este tipo de filtro, a atenuação do sinal de saída acontece de maneira muito mais abrupta quando comparado aos métodos implementados para filtro IIR.

No segundo experimento, foram implementados filtros passa-altas para ondas senoidais e ondas quadradas pelo método de Janela Retangular de ordem 80, o que possibilitou então observar as diferenças no comportamento dos sinais de saída. As ondas senoidais foram quase que totalmente atenuadas para sinais com frequência abaixo da frequência de corte, enquanto as ondas quadradas abaixo da frequência de corte apresentavam atenuação nas frequências mais baixas do que a frequência de corte. Como a frequência de corte projetada foi de 400Hz, uma onda quadrada de frequência igual à 373,4Hz teve sua componente fundamental atenuada, como pode-se observar na Figura 28.

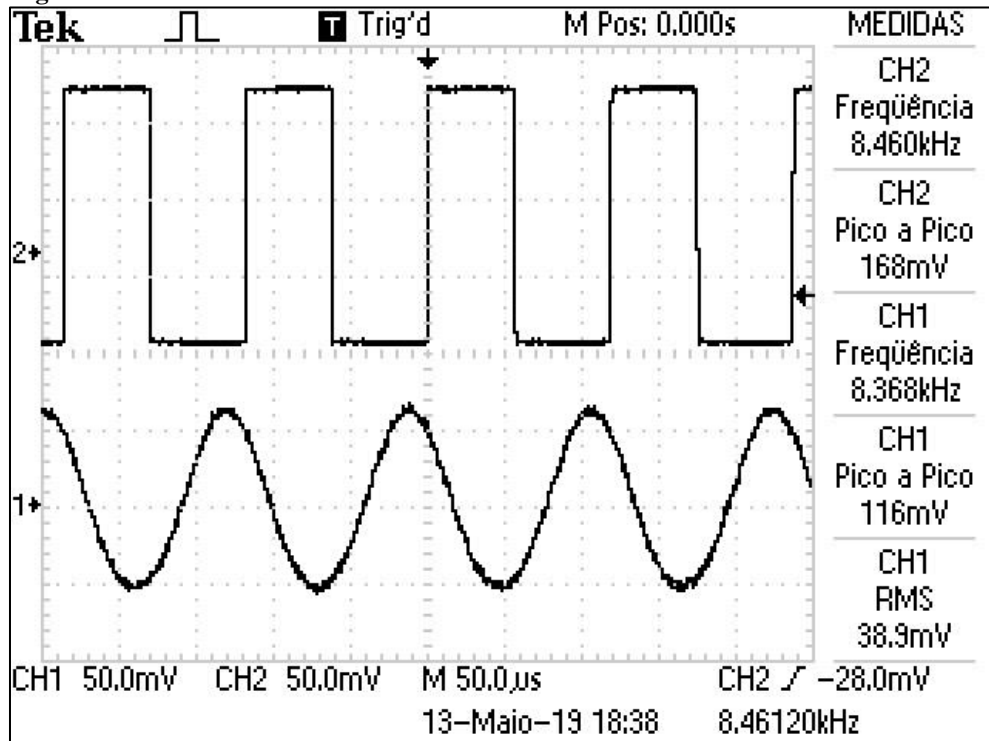
Figura 28 – Entrada e saída de experimento com filtro passa-altas.



Fonte: Autoria própria.

Neste mesmo experimento, pode-se verificar que há limitações do *hardware* com relação às frequências que podem ser reproduzidas na saída, que são frequências maiores ou iguais à 24 kHz. Para uma onda quadrada de frequência de 8460 Hz, foi possível observar apenas a onda senoidal da frequência fundamental na saída do sistema, pois as demais componentes têm frequência acima do limite do *hardware*, como mostra a Figura 29.

Figura 29 - Onda



Fonte: Autoria própria.

No terceiro experimento, foram implementados filtros passa-baixa e passa-alta para sinais de áudios. Para isso, foi conectado na entrada do adaptador de áudio um *smartphone* para a reprodução de uma música. Na saída do adaptador, foi conectada uma caixa de som, para que os alunos pudessem observar que, para o filtro passa-baixa, no áudio reproduzido na saída do filtro, os graves da música foram muito mais perceptíveis que os agudos, enquanto que para o filtro passa-alta, os agudos foram mais notáveis.

4.1.2 Avaliação dos discentes sobre utilização de *design* de filtro como recurso didático

Durante a demonstração dos experimentos práticos foi solicitado aos alunos que preenchessem uma ficha de avaliação, que pode ser observada no Apêndice C.

Ao todos foram feitas 7 avaliações, sendo de seis discentes da disciplina de Sinais e Sistemas 2 e mais uma feita pelo Prof. Glauber Gomes de Oliveira Brante, que ministra a disciplina. Foi solicitado aos presentes na aula que avaliassem, com notas de 1 a 10 afirmações sobre os experimentos realizados. A média das notas para cada afirmação podem ser observadas na Tabela 6.

Tabela 6 - Média das avaliações da demonstração de implementação de filtros digitais na disciplina de Sinais e Sistemas 2

	Afirmação	Média
1.	Experimento 1: A diferença da ordem de Filtros FIR fica claro ao final deste experimento.	9,57
2.	Experimento 1: A diferença das respostas dos sistemas devido a mudança do tipo de janela fica claro neste experimento.	9,57
3.	Experimento 1 e 2: A composição de uma onda quadrada fica clara nestes experimentos.	8,57
4.	Experimento 1: A resposta mais rápida de um filtro IIR fica claro neste experimento.	8,71
5.	Experimento 1: O funcionamento de um filtro do tipo passa-baixas foi verificado com clareza neste experimento.	9,71
6.	Experimento 2: O funcionamento de um filtro do tipo passa-altas foi verificado com clareza neste experimento.	8,71
7.	Experimento 2: O limite do hardware foi verificado com clareza neste experimento.	7,71
8.	Sobre a didática no ensino de disciplinas relacionadas a Sinais e Sistemas, você acredita que aulas práticas podem ajudar na fixação de conceitos teóricos.	9,57
9.	De maneira geral, o recurso utilizado nesta aula prática facilitou a fixação de conceitos teóricos relacionados à disciplina de Sinais e Sistemas 2.	9,29

Fonte: Autoria Própria

Nota-se que os experimentos tiveram boas avaliações por parte dos discentes e do professor que ministrava a aula, sobretudo para os experimentos que envolviam a fixação de conceitos sobre filtros FIR e IIR.

Contudo, os experimentos que tinham como objetivo mostrar as diferenças do comportamento de sinais determinísticos, isto é, a saída do filtro para ondas quadradas e senoidais tiveram uma avaliação mais baixa, que pode ser observada na média da afirmação 3 na Tabela 6.

Além disso, a parte do Experimento 2 que tinha como objetivo apresentar aos alunos sobre o comportamento das ondas quadradas levando em consideração os limites do *hardware* utilizado não foi verificado com muita clareza, visto a afirmação 7 da Tabela 6 teve a menor média de avaliação.

Após as avaliações das afirmações da Tabela 6, foi solicitado aos alunos que fizessem comentários com sugestões sobre a demonstração realizada. Nestes comentários, um dos alunos sugeriu que fossem abordados conceitos teóricos de Séries de Fourier para ondas quadradas e senoidais, a fim de facilitar a compreensão dos experimentos que relacionavam conceitos de componentes para ondas quadradas e senoidais.

Outra sugestão feita por um aluno foi a utilização de aplicativos para *smartphones* capazes de gerar sinais determinísticos com frequência controlada, o que eliminaria a necessidade de se utilizar o gerador de funções, tornando o experimento mais simples e com menor custo de equipamentos. Além disso, um aluno sugeriu que fosse utilizada a porta GPIO do Raspberry Pi para as entradas e saídas dos sinais de áudio.

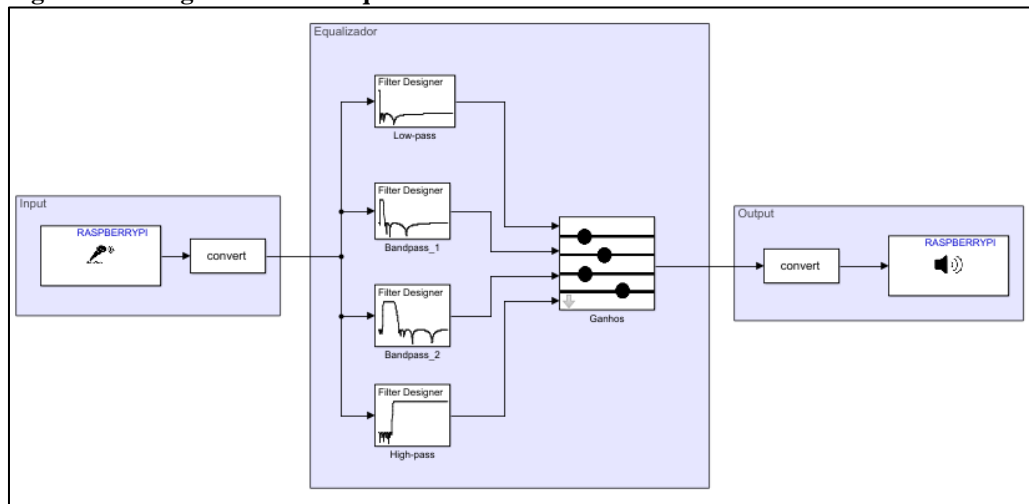
Para o Prof. Glauber, as avaliações seriam melhores se fossem feitas em 5 níveis, como, por exemplo, Péssimo, Ruim, Regular, Bom e Muito bom, a fim de facilitar a avaliação pelos alunos no momento das atividades práticas. Outro ponto comentado pelo professor foi que o segundo experimento seria interessante apenas para atividades individuais, e que, como foi apenas realizada como uma demonstração, não foi interessante para a fixação de conceitos teóricos. Além disso, o professor explica que os experimentos para onda quadrada são de difícil compreensão, sobretudo para explicar os limites do *hardware*. Portanto poderiam ser utilizadas apenas ondas senoidais no primeiro e segundo experimento.

No terceiro experimento, utilizando uma música como sinal de entrada e verificando o áudio após passar por filtros passa-baixas e passa-altas notou-se um grande interesse e interação dos alunos com o experimento realizado. Neste mesmo experimento, houve um problema com a caixa de som utilizada para a reprodução na saída do Raspberry Pi, que reproduziu com muito ruído os graves da música, dificultando a percepção da reprodução do áudio na saída do filtro passa-baixas.

4.2 CONSTRUÇÃO DE UM EQUALIZADOR A PARTIR DE BLOCOS DE PROGRAMAÇÃO

Na última etapa deste trabalho um equalizador de frequências foi implementado, ou seja, um sistema que combina filtros passa-baixas, passa-faixas e passa-altas. O diagrama construído no Simulink é mostrado na Figura 30.

Figura 30 - Diagrama de um Equalizador.

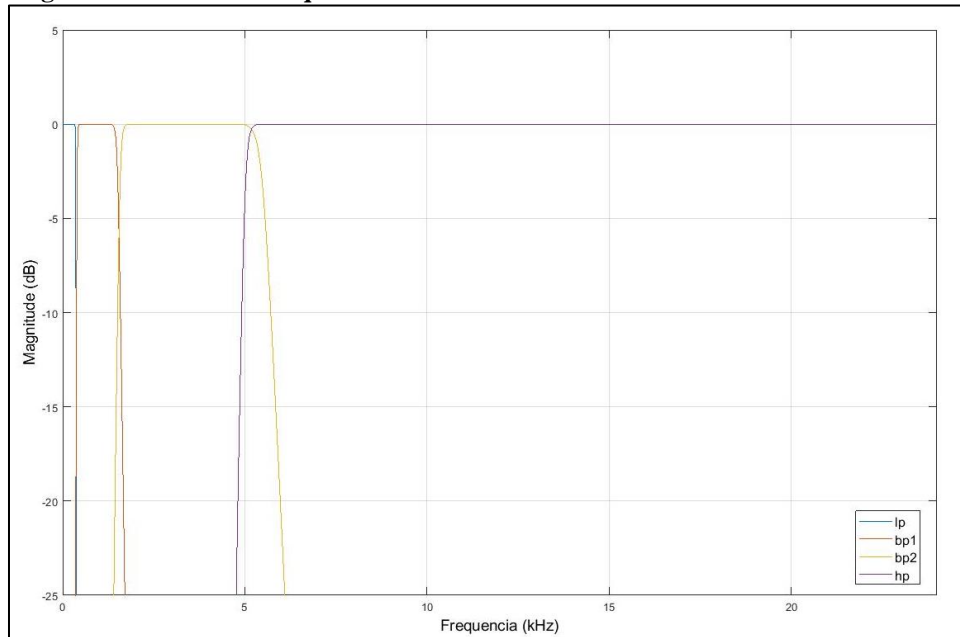


Fonte: Autoria própria.

Como os filtros FIR possuem ordem elevada, isto gera um atraso grande na saída do sistema. Além disso, a faixa de transição é maior do que dos filtros IIR. Desta forma, os filtros FIR não são indicados para aplicações em tempo-real, como um equalizador de frequências. Portanto, o método de filtragem escolhido foi o de Chebychev 2 de ordem 20, pois não apresenta ondulações na banda de passagem, as ondulações da banda de rejeição não podem ser observadas devido a atenuação elevada na banda de rejeição e a faixa de transição é pequena.

Para os experimentos realizados, o filtro passa baixa foi configurado com uma frequência de corte de 400Hz, o primeiro passa faixa com frequências de corte de 300Hz e 2000Hz, o segundo passa faixa com frequências de corte de 1200Hz e 7000Hz e o passa-altas com frequência de 4500Hz. A Figura 31 mostra a resposta da Magnitude em função da frequência os filtros implementados.

Figura 31 - Faixas de frequências de cada bloco de filtro

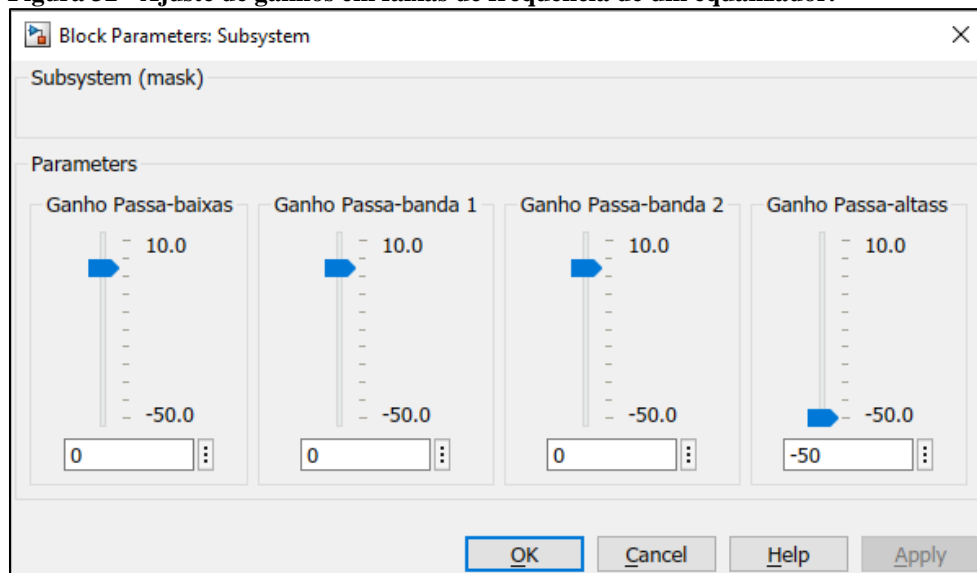


Fonte: Autoria própria.

Devido à grande atenuação da banda de rejeição configurada (80 dB), foi necessário fazer a sobreposição das frequências.

O diagrama mostra que além dos filtros, o equalizador também tem um bloco de ganhos, isto é, pode-se variar os ganhos de cada faixa de frequência filtrada. O bloco é um subsistema que agrupa 4 blocos de ganho e recebe uma máscara na qual se pode alterar os valores de ganhos de 10 a -50 dB, como mostra a Figura 32.

Figura 32 - Ajuste de ganhos em faixas de frequência de um equalizador.



Fonte: Autoria própria

Pode-se então variar a frequência do sinal de entrada, e conforme o ganho ajustado foi observado a alteração na amplitude do sinal de saída para as diferentes faixas de

frequências trabalhadas.

5 CONSIDERAÇÕES FINAIS

Neste trabalho foi possível verificar que as dificuldades de fixação de conhecimentos teóricos em disciplinas relacionadas a Sinais e Sistemas podem ser amenizadas com a utilização de recursos que possibilitem uma visualização prática do comportamento de sinais determinísticos e de áudio, bem como permitem a manipulação desses sinais utilizando a implementação de filtros digitais.

A utilização de um sistema formado pelo *hardware* Raspberry Pi e o *software* Matlab/Simulink, permitiu uma simples configuração de *design* de filtros, o que pode servir como um recurso didático para atividades *hands-on* durante as aulas, mesmo sem que o discente tenha conhecimentos de programação.

Analisando as avaliações dos discentes da disciplina de Sinais e Sistemas 2, da UTFPR, foi possível verificar que este recurso é eficaz para a exemplificação dos conceitos teóricos da disciplina, tanto para o entendimento do comportamento de sinais determinísticos quanto para as configurações dos filtros digitais abordados.

Além disso, partir dos testes realizados para as implementações de filtros digitais, foi possível observar as características para os métodos de filtragem de filtros dos tipos FIR e IIR, que foram objeto de estudo no levantamento da fundamentação teórica neste trabalho.

Por fim, com o conhecimento de como configurar os blocos de programação para *design* de filtros foi possível, com a construção de um equalizador de frequências, verificar o comportamento de vários filtros digitais conectados ao mesmo diagrama de blocos no Matlab/Simulink, bem como verificar os sinais de saída para esse sistema.

REFERÊNCIAS

ABURDENE, Maurice F., NEPAL, Kundal. Wow! Linear Systems and Signal Processing is fun! 41st ASEE/IEEE Frontiers in Education Conference 2011.

ARTHUR, James, Raspberry Pi: The complete guide to Raspberry Pi for beginners, including projects, tips, tricks, and programming

DINIZ, P. S. R, SILVA E. A. B., NETTO, S. L., Processamento Digital de Sinais: Projeto e Análise de Sistemas, 2 ed., Bookman, 2014.

INSTITUTO DE ELETRÔNICA E CIÊNCIAS DA COMPUTAÇÃO. Disponível em: <www.edi.lv/media/uploads/UserFiles/dasp-web/sec-5.htm> Acesso em 18/05/2018

JOAQUIM, Marcelo B. Sistemas Lineares e Invariantes. Disponível em <www.cear.ufpb.br/juan/wp-content/uploads/2016/08/Aula-3-Sistemas-Lineares-e-Invariantes.pdf> Acesso em 02/05/2018.

JOAQUIM, Marcelo B. Teorema de Amostragem. Disponível em <www.cear.ufpb.br/juan/wp-content/uploads/2016/08/Aula-5-Teorema-de-Amostragem.pdf> Acesso em 15/05/2018.

MITOLA, J. The software radio architecture, IEEE Communications Magazine, vol. 33, Maio 1995.

PASOLIN, G., BAZZI, A. ZABINI, F. A Raspberry Pi-Based Platform for Signal Processing Education, IEEE Signal Processing Magazine, julho 2017.

OPPENHEIM, A. V.; SCHAFER, R. W. Discret-Time signal Processing, 2.ed, Prentice-Hall, Upper Saddle River, NJ, EUA, 1997

RASPBERRY PI FOUNDATION, Raspberry Pi model B. Disponível em <www.raspberrypi.org/products/raspberry-pi-3-model-b/> Acesso em 04/02/2018

SIMONI, Mario, ABURDENE, Maurice. Lessons Learned from Implementing Application-Oriented Hands-On Activities for Continuous-Time Signal Processing Courses, IEEE Signal Processing Magazine, Julho 2016.

SILVA, W. S.,CORDEIRO, J. R. S; MACEDO, D. F., VIEIRA, M. A. M., VIEIRA, L. F. M., NOGUEIRA ,J. M. S., Introdução a Rádios Definidos por Software com aplicações em GNU Radio.

STEWART, B; BARBE, K; ATKINSON D; CROCKETT, L. Software Defined Radio using Matlab & Simulink and the RTL-SDR. Publicado por Strathclyde Academic Media, 2015

WEEKS, Michael, Digital Signal Processing Using Matlab and Wavelets. Publicado por Infinity Science Press LLC, 1. Ed. 2007

APÊNDICE A – Configurações do sistema

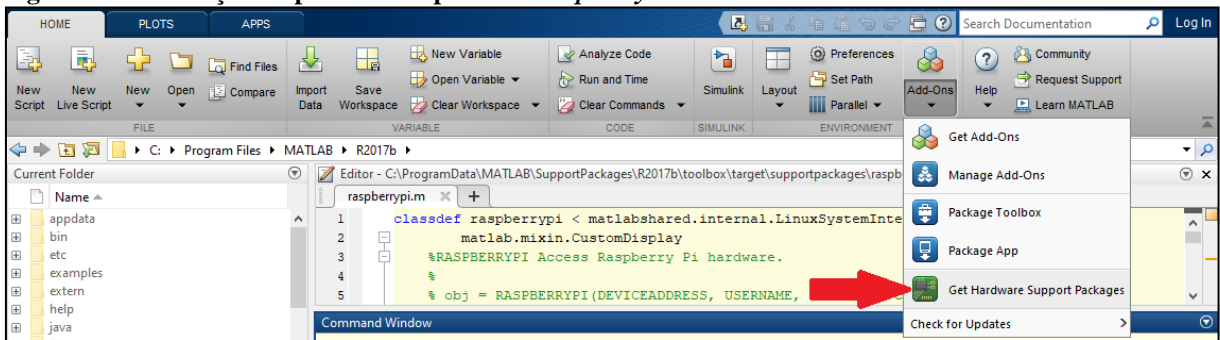
Nesta secção serão descritos os passos para configurar e comunicar os dispositivos, sendo estes o sistema operacional do Raspberry Pi, o adaptador de áudio USB e Matlab-Simulink, partindo do pressuposto que já se tenha instalado no computador o Matlab com Simulink na versão 2017b ou mais recente e que todos os equipamentos citados anteriormente estejam disponíveis.

A.1 Instalação de sistema operacional do Raspberry Pi via Matlab-Simulink

Segundo PASOLIN (2017), as seguintes bibliotecas do Matlab precisam estar instaladas:

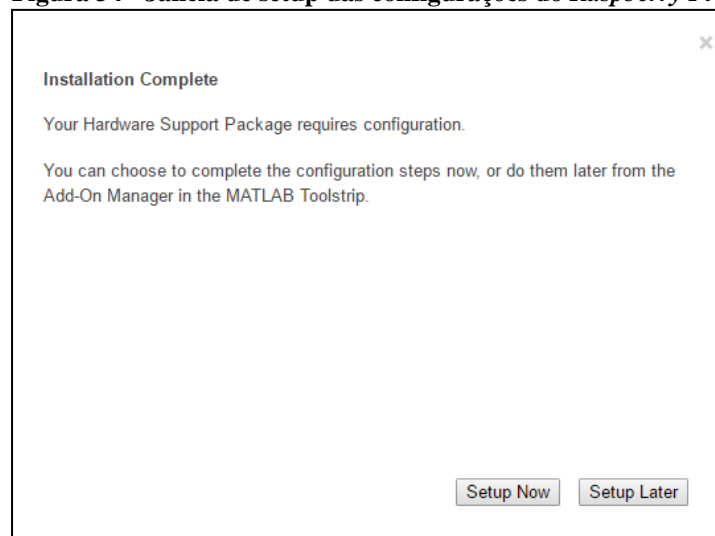
- Communications System Toolbox;
- DSP System Toolbox;
- Data Acquisition Toolbox;
- Fixed-Point Designer;
- Instrument Control Toolbox;
- Signal Processing Toolbox.

Além disso, para fazer *download* e instalar os pacotes de suporte do Raspberry Pi, é necessário ter uma conta e fazer login no Matlab. Não será descrito o processo de registro de conta no Matlab, apenas de *download* e instalação do pacote de *hardware*. Para fazer o *download* dos pacotes de suporte de *hardware*, inicializar o programa e na aba “HOME” clicar no botão Add-Ons e em seguida clicar em “Get Hardware Support Packages”, como mostra a Figura 33.

Figura 33 - Instalação do pacote de suporte do *Raspberry Pi*

Fonte: Autoria Própria

Na janela que abrir, procurar por “Simulink Support Package for Raspberry Pi *Hardware*” e clicar em instalar. Após aceitar os termos de compromisso, o pacote de suporte do Simulink e do Matlab serão instalados e também será feito *download* do sistema operacional Raspbian pré-configurado para esta aplicação. Após a instalação, uma janela de configuração aparecerá conforme a Figura 34.

Figura 34 - Janela de setup das configurações do *Raspberry Pi* no *Matlab*

Fonte: Autoria Própria

Clicar em “*Setup Now*” e seguir as orientações para a gravação do sistema operacional no cartão microSD.

Nas configurações de rede, escolher a conexão direta ao computador.

Depois da gravação, remover o cartão microSD do computador e inserir no Raspberry Pi. O cabo de rede deve ser conectado diretamente entre o computador e o Raspberry Pi. O adaptador de áudio USB deve ser plugado antes dos próximos passos. Ligar o Raspberry Pi na fonte 5V e realizar os testes de comunicação conforme será indicado no Matlab.

A.2 Configuração do adaptador de áudio

Após a instalação e teste de comunicação, é necessário configurar o adaptador de áudio. Isto é feito via terminal, que pode ser aberto pelo Matlab com os comandos:

```
h = raspi()
openShell(h)
```

Primeiramente, é necessário desabilitar a saída de áudio do próprio Raspberry Pi. Para isto, é necessário alterar um arquivo que pode ser aberto pelo comando:

```
sudo nano /etc/modprobe.d/raspi-blacklist.conf
```

Dentro deste arquivo, inserir o seguinte código:

```
blacklist snd_bcm2835
```

Depois de salvar as alterações e sair do arquivo, é necessário tornar o adaptador de áudio USB como dispositivo padrão no arquivo aberto pelo comando:

```
sudo nano /lib/modprobe.d/aliases.conf
```

Dentro deste outro arquivo, inserir “#” antes da linha que contém o código **options snd-usb-audio index=-2**

Após as alterações, deve-se reiniciar o Raspberry Pi, o que pode ser feito pelo comando **sudo reboot** no Matlab.

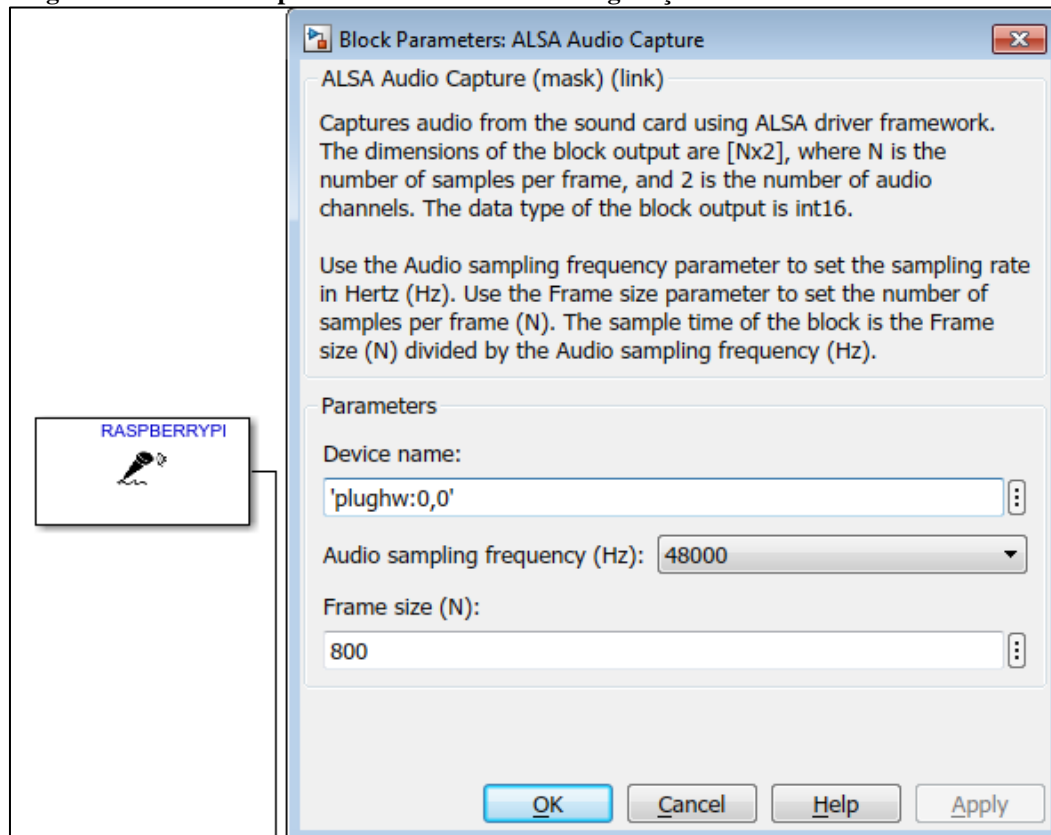
Após reiniciar o sistema, pode-se verificar se as alterações foram realizadas, testando a saída de som do adaptador de áudio. Para isto, deve-se plugar um fone de ouvido no adaptador de áudio USB, abrir novamente o terminal via Matlab e digitar o comando **aplay /usr/share/sounds/alsa/Front_Center.wav**. Se for possível escutar no fone de ouvido o áudio, a configuração foi feita com sucesso.

A.3 Configuração de blocos no Matlab-Simulink

Com todas as configurações prontas, pode-se montar esquemas via Simulink que podem ser compilados e enviados em forma de programa para o Raspberry Pi o que executará.

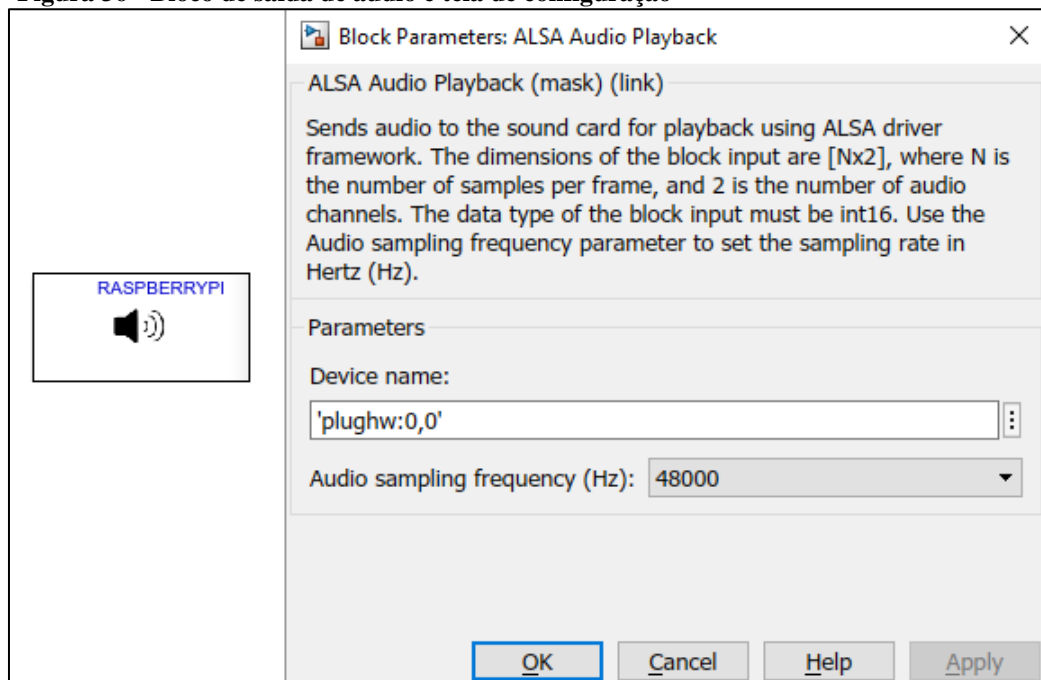
A biblioteca RASPBERRYPI contém 2 blocos que serão utilizados para este projeto: “*ALSA Audio Capture*” e “*ALSA Audio Playback*” cujas configurações são apresentadas na Figura 35 e na Figura 36, respectivamente.

Figura 35- Bloco de captura de áudio e tela de configuração



Fonte: Autoria Própria

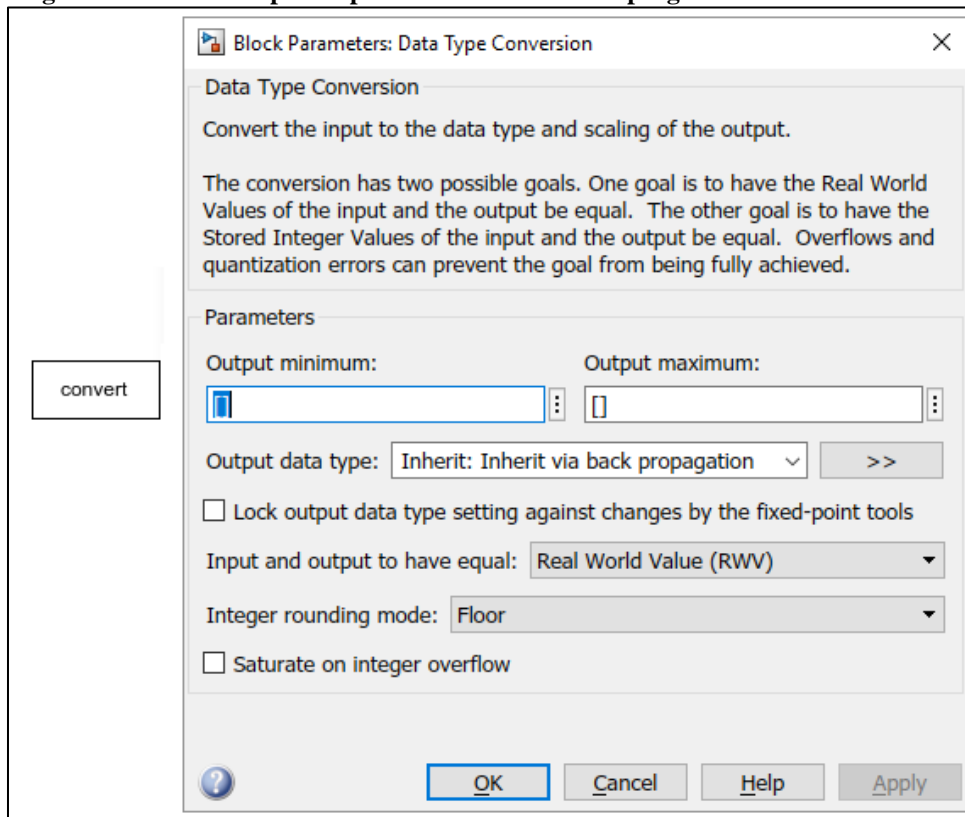
Figura 36 - Bloco de saída de áudio e tela de configuração



Fonte: Autoria Própria

Além destes dois blocos, o bloco de conversão de tipo de variável “*Convert*” da biblioteca *DSP Toolbox* deve ser configurado conforme a Figura 37.

Figura 37 - Primeiros passos para fazer download do programa

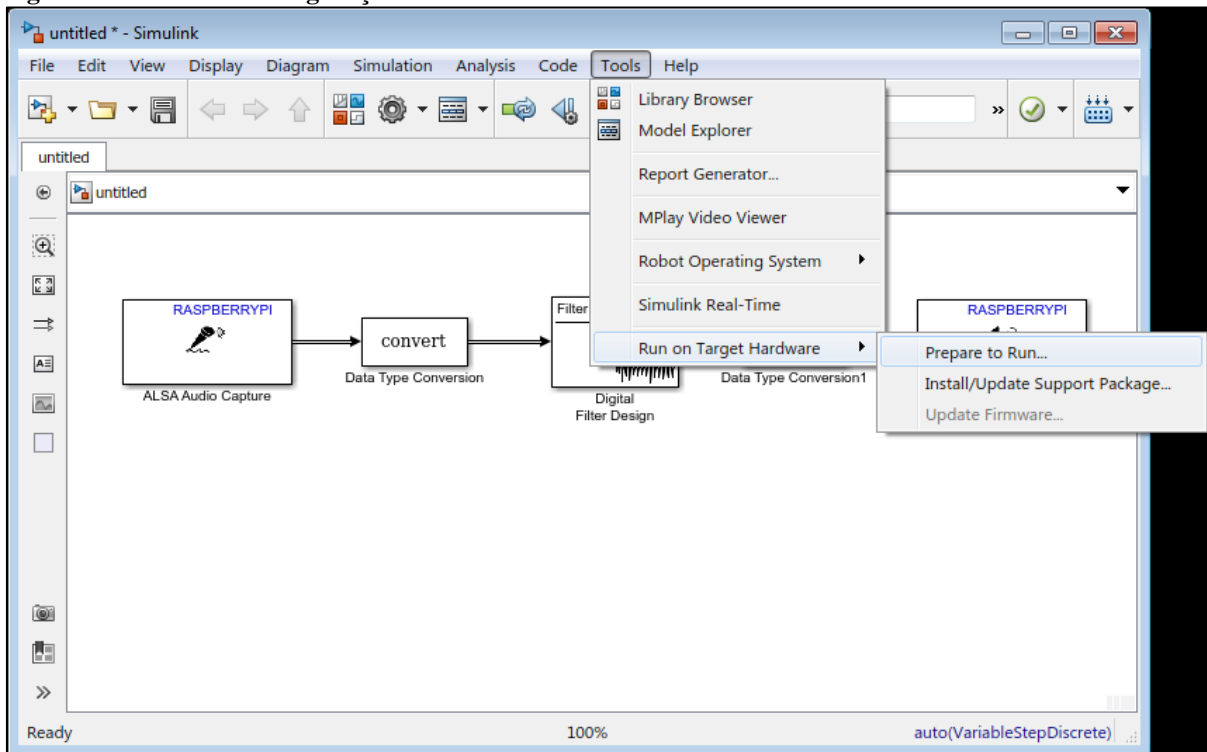


Fonte: Autoria Própria.

A.4 Download de programa no Raspberry Pi

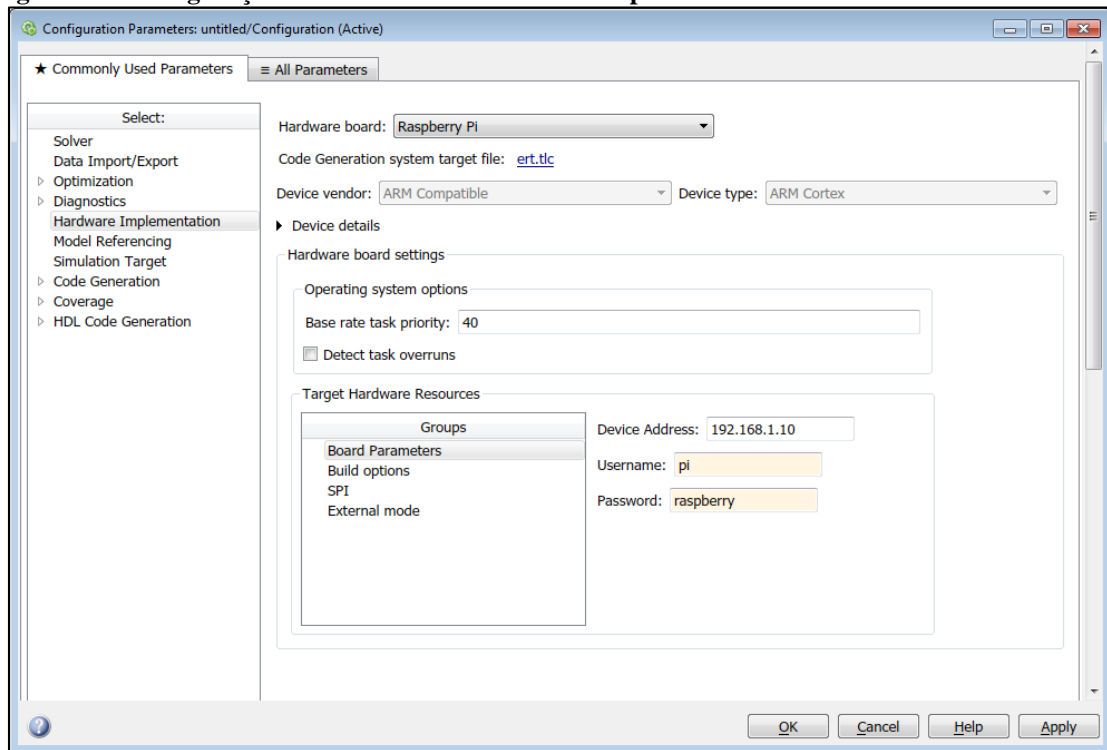
Após terminada a programação no MATLAB/Simulink, é necessário compilar o programa e fazer *download* do mesmo no Raspberry Pi. Para a primeira vez que se faz isto, é necessário fazer a configuração do *hardware* do Raspberry Pi. Para acessar este menu do Simulink, clicar em “*Tool>Run on Target Hardware>Prepare to Run...*”, conforme indicado na Figura 38.

Figura 38 - Acesso às configurações de *hardware*.



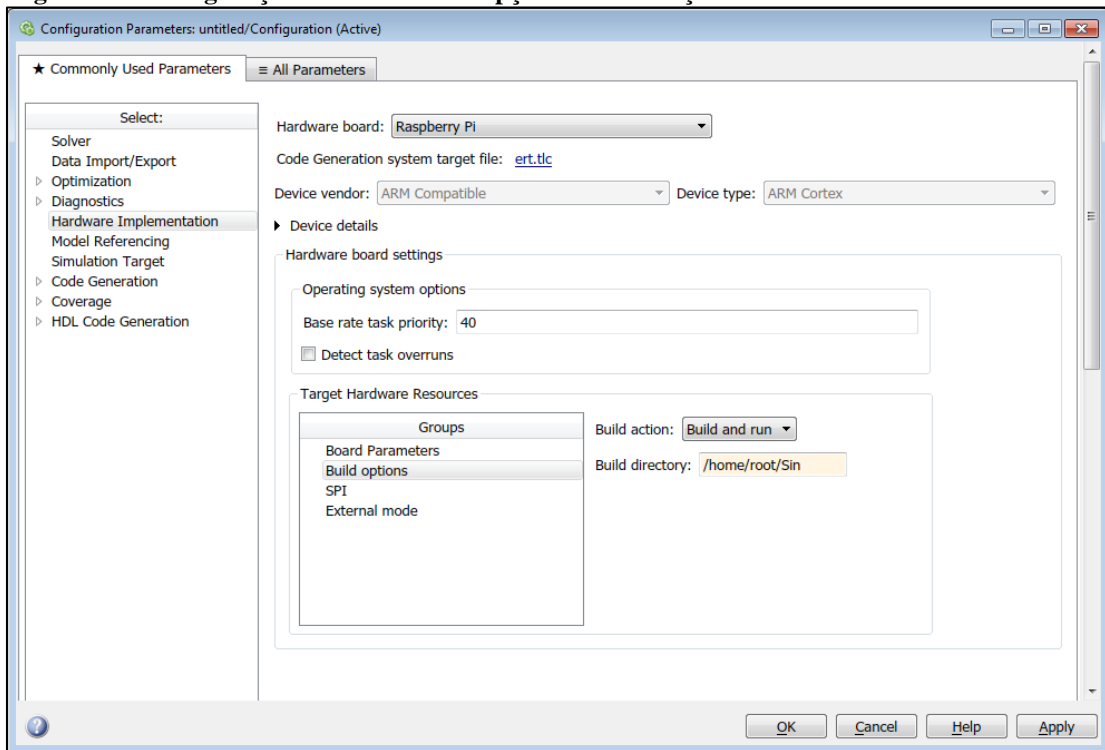
Fonte: Autoria Própria.

Na nova tela, Figura 39, conferir se os dados da placa do Raspberry Pi estão de acordo com o que foi configurado previamente na etapa de instalação do sistema operacional. Caso haja alguma divergência, trocar o dado errado nesta tela.

Figura 39– Configuração de *Hardware* – Parâmetros de placa.

Fonte: Autoria Própria.

Com os dados de placas corretos, o próximo passo é indicar o local onde os programas serão armazenados no Raspberry Pi. Clicando em “*Build options*“, o caminho desta pasta deve ser indicado no campo “*Build directory*“, como, por exemplo, na Figura 40.

Figura 40– Configuração de *Hardware* – Opções de construção

Fonte: Autoria Própria.

O último passo é enfim compilar e fazer o *download* do programa. Para isto, basta clicar em “*Deploy to hardware*” na barra de ferramentas do *Simulink*. Se não houver nenhum problema no programa ou de configuração de dispositivo, para iniciar a execução do programa basta clicar em “*Run*” na barra de ferramentas.

APÊNDICE B – Demonstração de implementação de filtros através de experimentos práticos

B.1 Objetivo

O objetivo desta atividade é verificar na prática conceitos de Processamento Digital de Sinais, utilizando Raspberry Pi e Matlab/ Simulink sem a necessidade de conhecimentos de programação.

B.2 Materiais e Equipamentos

Para esta atividade são necessários:

Tabela 7 – Materiais e Equipamentos

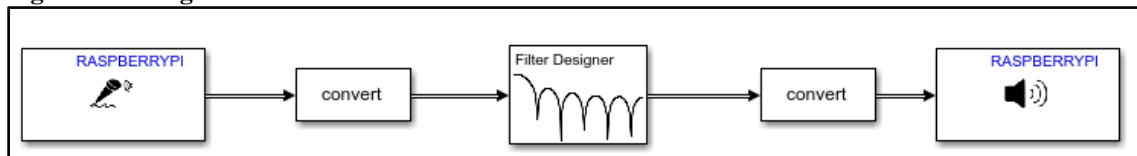
Item	Quantidade	Descrição
1	1	Raspberry Pi3 Model B+
2	1	Computador com Matlab/Simulink
3	1	Osciloscópio
4	1	Gerador de funções
5	1	Adaptador de áudio USB
6	1	Cabo de rede
7	1	Adaptador de rede USB
8	1	Cabo bnc macho-macho
9	2	Cabo rca
10	2	Conector rca-bnc
11	1	Conector T bnc 1 macho - 2 fêmeas
12	1	Cabo auxiliar p2
13	1	Microfone
14	1	Caixa de som/fone de ouvido

B.3 Ambientação

B.3.1 Software

A ferramenta utilizada para as configurações dos parâmetros de processamento e dos filtros é o Simulink/Matlab. A Figura 41 mostra o diagrama de blocos utilizado.

Figura 41 - Diagrama de blocos do Simulink

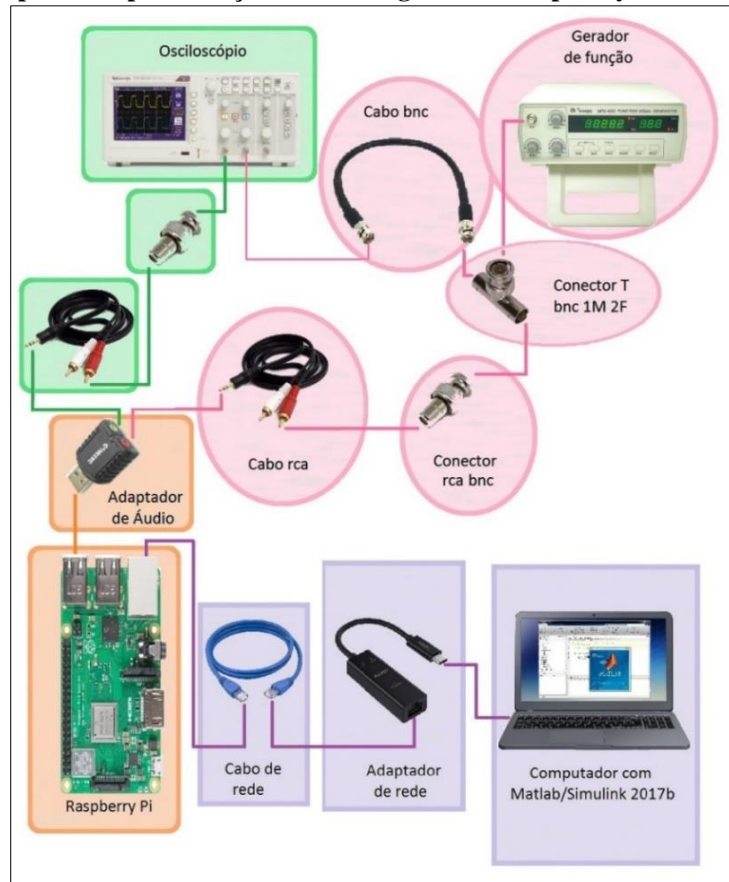


A partir destes blocos, vamos descarregar o programa no Raspberry Pi, clicando no ícone de “Deploy to hardware” (📡), que fará o processamento dos dados de entrada e saída do sistema. Para iniciar a aplicação, clica-se em “Run” (▶).

B.3.2 Conexão dos equipamentos

Para as primeiras atividades, os equipamentos são conectados conforme a Figura 42.

Figura 42 - Diagrama de blocos com os equipamentos utilizados para a implementação de filtro digital com Raspberry Pi



Fonte: Adaptado de PASOLIN, Gianni et. all (2017).

B.3.3 Sinais

Para o melhor entendimento do funcionamento, primeiramente utilizaremos sinais determinísticos, pois assim é possível verificar se os resultados obtidos são iguais aos esperados.

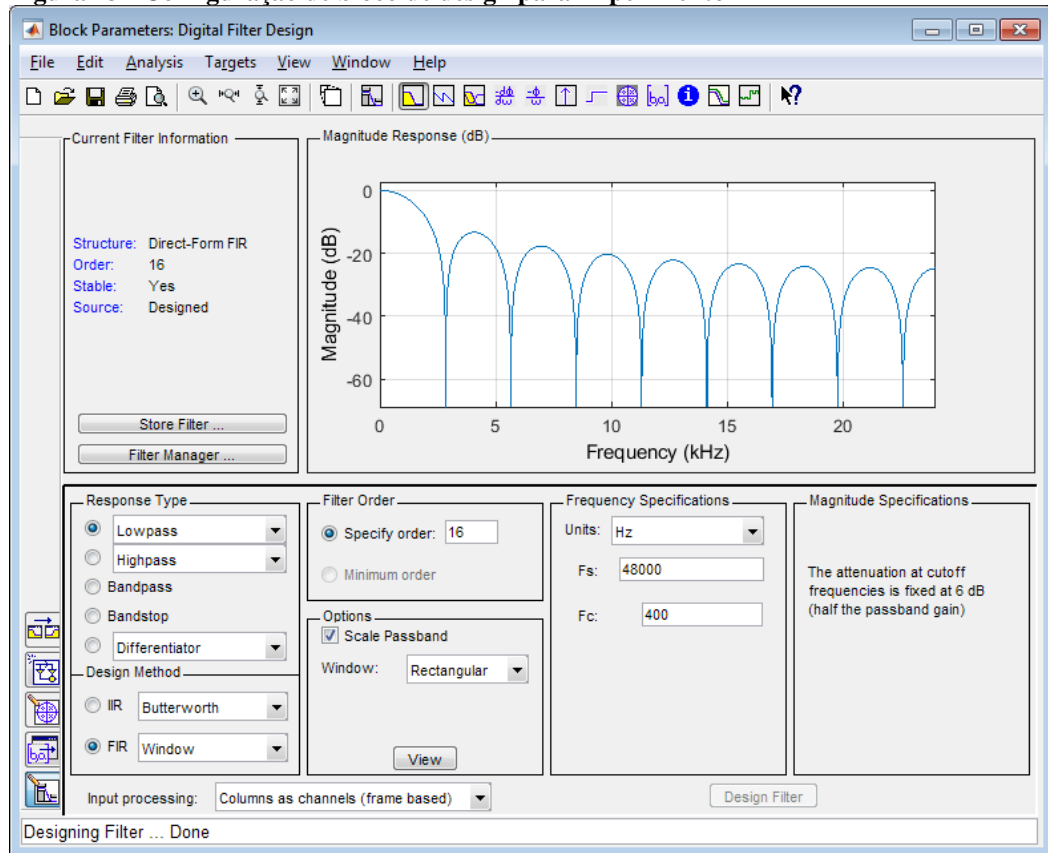
Os sinais determinístico escolhidos para a verificação dos conceitos de filtros são a onda senoidal – que possui apenas uma componente na série de Fourier referente à sua frequência fundamental – e a onda quadrada com $t_h = t_l = T/2$ – que possui apenas componentes ímpares da série de Fourier.

Além dos sinais determinísticos, sinais reais de áudio serão utilizados para que se possa trabalhar com sinais reais.

B.4 Experimento 1: Implementação Dos Filtros Passa-Baixas Dos Tipos FIR E IIR

Nesta atividade, será possível visualizar como são configurados os filtros dos tipos FIR e IIR no bloco de *design* de filtros.

Figura 43 - Configuração de bloco de design para Experimento 1



Passo 1: Altere os parâmetros do bloco de design para configurar um filtro FIR, passa-baixas, pelo método de Janela Retangular, ordem 16, $F_c=400\text{Hz}$.

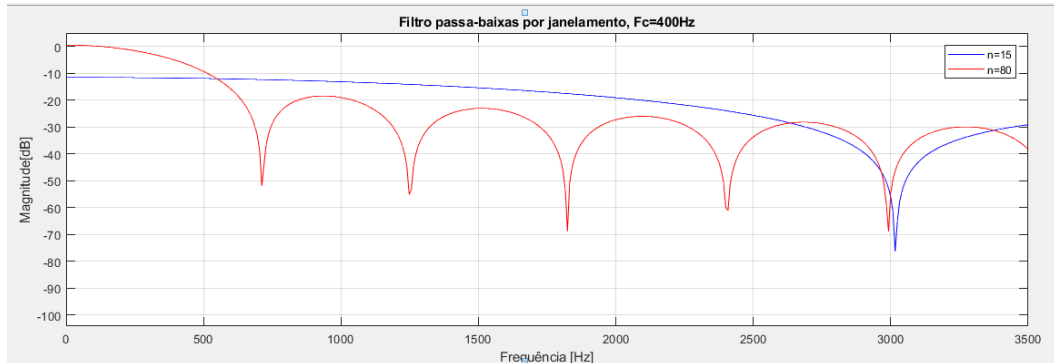
Passo 2: Gere uma onda senoidal de 300Hz e aumente a frequência para verificar como o sinal é atenuado. Verifique o comportamento do filtro.

Note que devido à baixa ordem do filtro, este torna-se ineficiente pois os sinais com frequência muito maior que 400Hz continuam a ser reproduzidos na saída sem muita atenuação.

Passo 3: Configure um filtro FIR, passa-baixas, pelo método de Janela Retangular, ordem 80, $F_c=400\text{Hz}$.

Passo 4: Verifique o comportamento do filtro para a mesma onda.

Devido à alta ordem do filtro, os sinais com frequência maior que 400Hz têm uma atenuação expressiva na saída. Esta é uma característica dos filtros FIR, ou seja, possuem uma ordem alta para que satisfaçam as exigências do sistema.

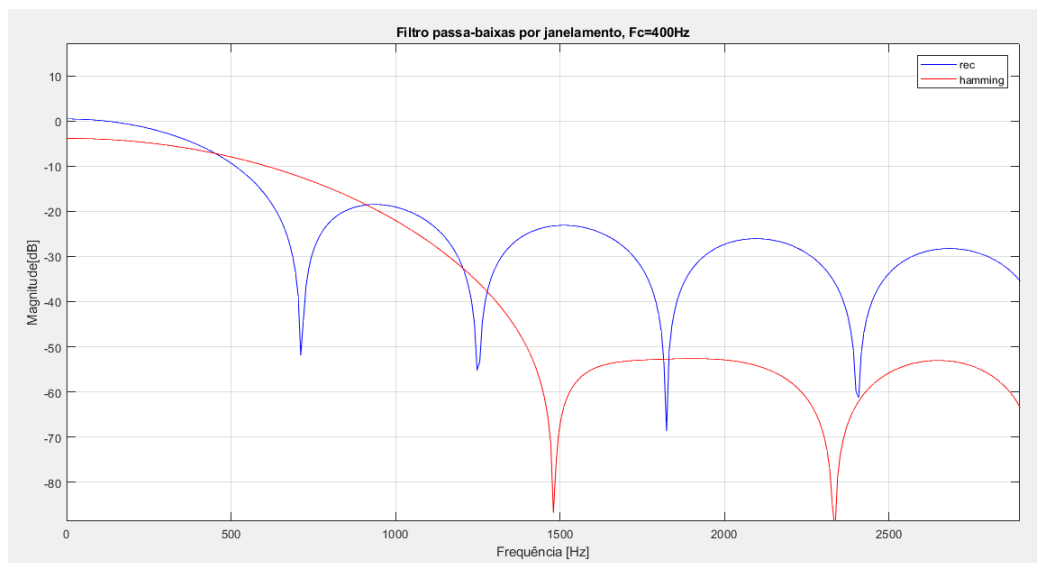


Nota-se que, para até 400Hz, o sinal de entrada todo o sinal é passado para a saída com uma pequena atenuação. A partir de 400Hz, as frequências são atenuadas cada vez mais, até aproximadamente 700Hz.

Passo 5: Configure um filtro FIR, passa-baixas, pelo método de Janela de Hamming, ordem 80, $F_c=400\text{Hz}$.

Passo 6: Verifique o comportamento do filtro para a mesma onda.

Verifique que para cada tipo de janela, o gráfico da resposta da Magnitude x Frequência se altera e mostra as características de cada método. Compare a resposta do filtro projetado neste experimento com a resposta do experimento anterior. Verifique o quão rápido ocorre o corte das frequências superiores a 400Hz e também o quanto essas frequências são atenuadas.



Passo 7: Gere uma onda quadrada de 100 Hz e observe as componentes que passam pelo filtro.

Passo 8: Aumente a frequência do sinal de entrada e verifique o comportamento do sinal de saída.

Nota-se que para os sinais entre 150Hz e 400Hz, apenas a componente fundamental será observada no sinal da saída. Para os sinais acima de 400Hz, o sinal de saída será atenuado.

Passo 9: Configure um filtro IIR, passa-baixas, $F_c=400\text{Hz}$, pelo método de Butterworth de ordem mínima.

Passo 10: Verifique o comportamento do filtro para a mesma onda senoidal.

Observe que a atenuação das frequências acima da frequência de corte sofrem uma atenuação muito mais abrupta se comparada aos filtros FIR, ainda que utilizando a ordem mínima.

B.5 Experimento 2: Implementação dos Filtros FIR Passa-Altas

Neste experimento, poderá se observar as diferenças de comportamento de sinais determinísticos na saída dos filtros implementados. Os sinais utilizados serão onda quadrada e senoidal.

Passo 1: Configure um filtro FIR pelo método de Janela Retangular, ordem 80, passa-alta, $F_c=400\text{Hz}$.

Passo 2: Gere uma onda senoidal de 600Hz e baixe a frequência. Verifique o comportamento do filtro.

Passo 3: Gere uma onda quadrada de 600Hz e baixe a frequência. Verifique o comportamento do filtro.

Para uma onda senoidal em frequências abaixo da frequência de corte, o sinal de saída será bastante atenuado. Para uma onda quadrada abaixo da frequência de corte, componentes de ordem superior à frequência de corte poderão ser observadas na saída. Contudo, estas serão limitadas pelo *hardware*.

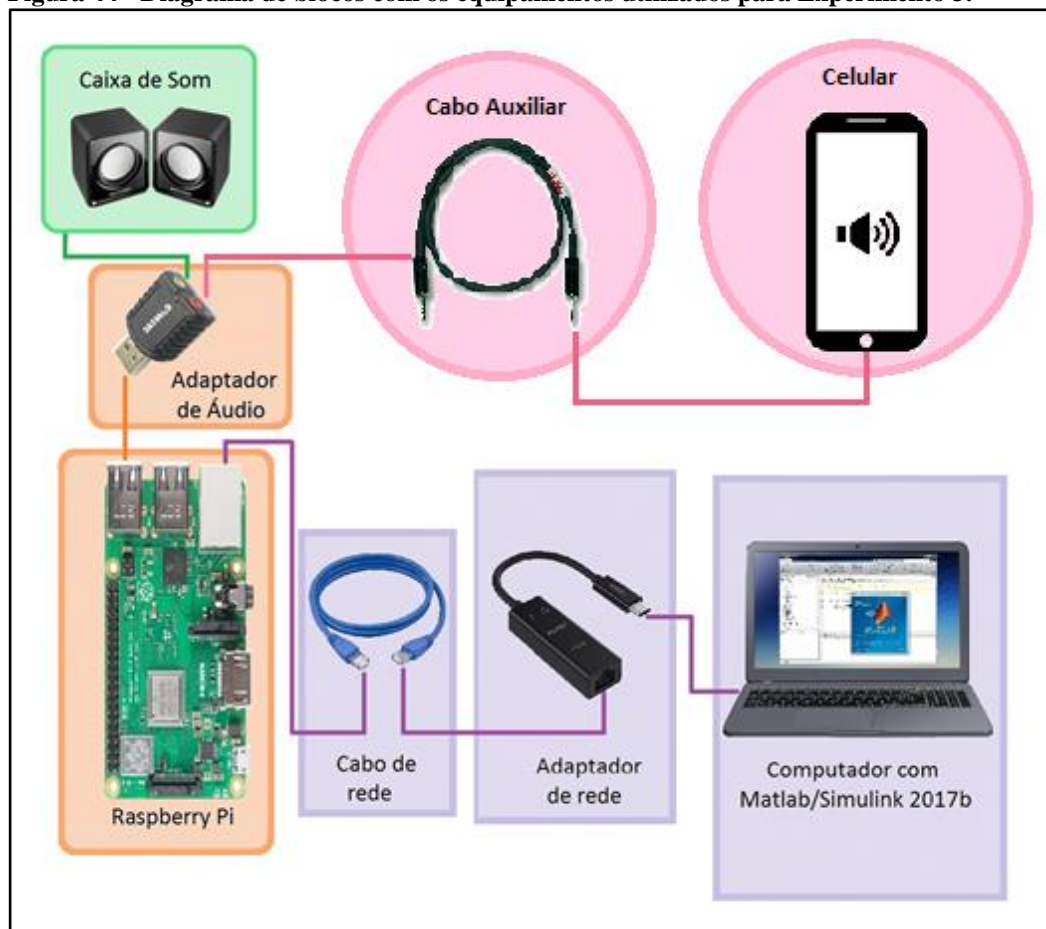
Passo 4: Gere uma onda quadrada de 8kHz e verifique o sinal de saída

Nota-se que apenas a componente fundamental poderá ser observada, pois o *hardware* é limitado a 24kHz.

B.6 Experimento 3: Implementação dos Filtros Para Sinais Reais

Para estes experimentos, na entrada de microfone do adaptador de áudio, utiliza-se um cabo auxiliar conectando um celular tocando arquivo de áudio e na saída uma caixa de som.

Figura 44 - Diagrama de blocos com os equipamentos utilizados para Experimento 3.



Passo 1: Configure um filtro IIR pelo método de Butterworth, ordem mínima, passa-baixa, $F_{pass}=300\text{Hz}$ e $F_{stop}=350\text{Hz}$.

Passo 2: Reproduza a música *My Own Business* na entrada do Raspberry Pi.

Passo 3: Sem pausar a música, altere os parâmetros do filtro para um passa-altas, $F_{stop}=650\text{Hz}$ e $F_{pass}=700\text{Hz}$.

Passo 4: Note as diferenças na reprodução da música para cada filtro.

APÊNDICE C – Ficha de avaliação da demonstração de implementação de filtros através de experimentos práticos

A partir de avaliação de 1 a 10, selecione como você classifica as seguintes afirmações sobre a aula ministrada.

1. Experimento 1: A diferença da ordem de Filtros FIR fica claro ao final deste experimento.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

2. Experimento 1: A diferença das respostas dos sistemas devido a mudança do tipo de janela fica claro neste experimento.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

3. Experimento 1 e 2: A composição de uma onda quadrada fica clara nestes experimentos.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

4. Experimento 1: A resposta mais rápida de um filtro IIR fica claro neste experimento.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

5. Experimento 1: O funcionamento de um filtro do tipo passa-baixa foi verificado com clareza neste experimento.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

6. Experimento 2: O funcionamento de um filtro do tipo passa-alta foi verificado com clareza neste experimento.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

7. Experimento 2: O limite do hardware foi verificado com clareza neste experimento.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

8. Sobre a didática no ensino de disciplinas relacionadas a Sinais e Sistemas, você acredita que aulas práticas podem ajudar na fixação de conceitos teóricos.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

9. De maneira geral, o recurso utilizado nesta aula prática facilitou a fixação de conceitos teóricos relacionados à disciplina de Sinais e Sistemas 2.

01	02	03	04	05	06	07	08	09	10
----	----	----	----	----	----	----	----	----	----

Comente sobre a aula realizada e sobre a utilização de recursos práticos em disciplinas de Sinais e Sistemas:
