

Projeto 2 - Processamento de Sinais Digitais

Daniel Tatsch, Nelson Alves, Schaiana Sonaglio

Engenharia de Telecomunicações

IFSC - Instituto Federal de Santa Catarina, São José, SC

Junho de 2018

1 Introdução

Este relatório apresenta três simulações onde foi aplicada a metodologia de projeto de filtros digitais *FIR* (do inglês *Finite Impulse Response*), que é um filtro digital com resposta ao impulso de duração finita, utilizado em processamento de sinais digitais. Foram implementados os seguintes tipos de filtros: filtro passa baixas utilizando uma janela fixa, filtro passa altas utilizando a janela ajustável de *Kaiser* e um filtro passa faixa utilizando o algoritmo de *Parks-McClellan*.

Para todos os filtros, procurou-se obter a menor ordem possível, para ter uma resposta em frequência que atendesse as especificações.

2 Metodologia

A metodologia aplicada no primeiro exercício foi a seguinte: Primeiramente, foi escolhida uma janela que se adequasse à atenuação na banda de rejeição do filtro. Depois, uma ordem inicial para o filtro foi estimada, realizando-se assim o cálculo dos coeficientes da série de Fourier. Após isso, testou todas as outras janelas disponíveis, a fim de achar a menor ordem possível para o filtro.

No segundo exercício, os cálculos foram realizados de forma mais objetiva, utilizando as funções *kaiserord()* e *kaiser()* do Matlab. Estas funções dão um valor muito próximo ao ideal para as especificações do filtro, porém, com alguns ajustes conseguiu-se uma ordem menor do que o especificado nas funções.

Na última etapa do projeto, o filtro passa-faixa foi projetado com o auxílio das funções *firpmord()* e *firpm()*, que se utiliza para obter a ordem aproximada do filtro desejado, junto com alguns ajustes na magnitude dos *ripples* na banda passante e de bloqueio.

Em todos os exercícios foi utilizada a ferramenta *FDATool* do Matlab para projetar os filtros desejados, a fim de se ter uma aproximação dos possíveis resultados.

3 Implementações e Análises

Nesta seção, serão apresentadas as implementações dos quatro filtros citados na introdução deste documento.

3.1 Filtro 1 - Passa Baixas - Janela de Hamming

As especificações utilizadas no filtro passa baixas estão ilustradas na Tabela 1.

Tabela 1 - Filtro Passa Baixa: especificações

Frequência de amostragem (Fa)	8 KHz
Ganho médio na banda de passagem (Gp)	0 dB
Atenuação máxima na banda de passagem (Ap)	0.5 dB
Atenuação mínima na banda de rejeição (As)	30 dB
Frequência de Passagem (Fp)	10500 Hz
Frequência de Rejeição (Fs)	14000 Hz
Método	Janela fixa de Hamming

Neste filtro, primeiramente foram implementados os coeficientes da série de *Fourier* de filtros ideais, conforme ilustrado na Equação 1, após isso, escolheu-se uma janela inicial, baseando-se na atenuação da banda de rejeição (A_s) - a janela escolhida foi a de *Bartlett-Hanning*, por ser a mais próxima. Após essa escolha inicial da janela, foi realizado o teste com todas as outras janelas, a fim de buscar uma ordem menor que a obtida ($N = 66$). Após este teste, a ordem do filtro foi reduzida para $N = 60$, através da janela de *Hamming*.

$$C_L = \frac{\omega_c}{\pi} \quad \text{para } n = 0 \qquad C_L = \frac{\text{sen}(\omega_c n)}{\pi n} \quad \text{para } |n| > 0 \qquad (1)$$

3.1.1 Análise

Através da implementação do filtro no *software* MATLAB®, os resultados ilustrados na Figura 1 foram obtidos.

Conforme observado na Figura 1, o filtro atendeu todas as especificações propostas, tendo uma atenuação máxima na banda de passagem (A_p) igual a 0.5 dB e uma atenuação máxima na banda de rejeição de 30 dB, além de ter se adequadamente à janela de *Hamming*, não havendo possibilidade de reduzir a ordem $N = 60$.

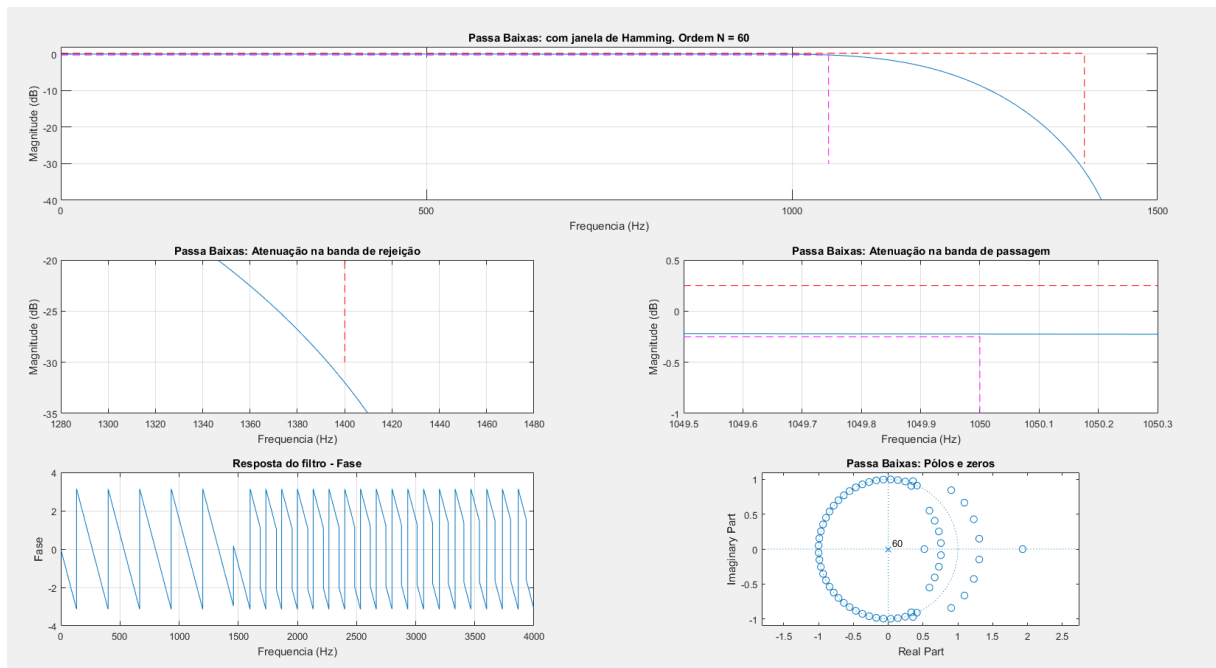


Figura 1 - Filtro passa baixas com janela fixa de Hamming. Fonte: elaboração própria.

3.2 Filtro 2 - Passa Altas com Janela de Kaiser

As especificações utilizadas no filtro passa altas estão ilustradas na Tabela 2.

Tabela 2 - Filtro Passa Alta: especificações

Frequência de amostragem (Fa)	8 KHz
Ganho médio na banda de passagem (Gp)	0 dB
Atenuação máxima na banda de passagem (Ap)	0.5 dB
Atenuação mínima na banda de rejeição (As)	20 dB
Frequência de Passagem (Fp)	1400 Hz
Frequência de Rejeição (Fs)	1050 Hz
Método	Janela ajustável de Kaiser

Neste filtro, primeiramente foram configurados os parâmetros para a utilização da função *kaiserord()*. Ela recebeu um vetor com as respectivas frequências de rejeição e passagem, outro vetor representando as magnitudes correspondentes a essas frequências, um último vetor que determina a amplitude dos *ripples* provenientes do filtro, tanto na banda de rejeição quanto na de passagem e, por fim, também foi necessário informar a frequência de amostragem do sinal.

O vetor que contém as amplitudes dos *ripples*, denominado *devs*, é definido pela Equação 2, onde *As* e *Ap* são as atenuações na banda de rejeição e passagem, respectivamente.

$$devs = \left[10^{\left(\frac{-As}{20}\right)} \quad \frac{10^{\left(\frac{Ap}{20} - 1\right)}}{10^{\left(\frac{Ap}{20} - 1\right)}} \right] = [0.1 \quad 0.0288] \quad (2)$$

Como retorno, obtém-se, através da função *kaiserord()*, a ordem do filtro, as frequências normalizadas, o fator da janela de Kaiser e o tipo do filtro. A partir desses dados foi utilizada a função *fir1()*, que faz o projeto de filtros FIR baseados em janela, retornando os coeficientes do mesmo.

3.2.1 Análise

Conforme especificado na metodologia empregada nesse filtro, a função *kaiserord()* não nos retorna a menor ordem possível para se projetar o filtro, sendo assim, após a primeira análise dos resultados, obteve-se a ordem do filtro igual a 38. Após isso, verificou-se que haviam certas "folgas" nas atenuações de banda de rejeição e passagem. Isso gerou flexibilidade no filtro, possibilitando o dimensionamento das frequências de sobra, para determinar uma ordem menor, de acordo com a Equação 3.

$$n_{ajustado} > n - \frac{F_p - F_s}{f_{sobra_{baixo}} + f_{sobra_{cima}}} = 32 \quad (3)$$

Com esse ajuste de ordem, foi necessário deslocar o sinal cerca de 25 Hz para que não houvesse sobreposição nas especificações do filtro. O resultado final do filtro é apresentado na Figura 2.

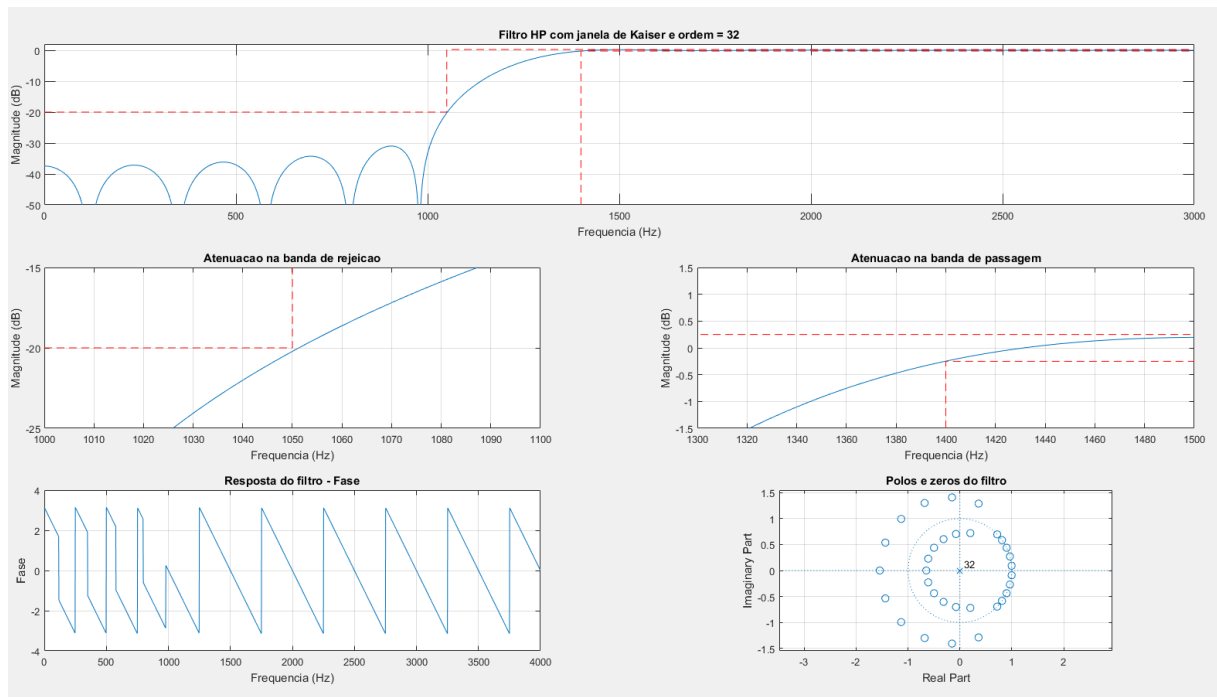


Figura 2 - Filtro passa altas com janela ajustável de Kaiser. Fonte: elaboração própria.

3.3 Filtro 3 - Passa Faixa - algoritmo de Parks-McCleallan

As especificações utilizadas no filtro passa faixa estão na tabela 3.

Tabela 3 - Filtro Passa Faixa: especificações

Frequência de amostragem (Fa)	8 KHz
Ganho médio na banda de passagem (Gp)	0 dB
Atenuação máxima na banda de passagem (Ap)	0.5 dB
Atenuação mínima na banda de rejeição (As)	30 dB
Frequência de Rejeição 1 (Fs1)	693 Hz
Frequência de Passagem 1 (Fp1)	755 Hz
Frequência de Passagem 2(Fp2)	785 Hz
Frequência de Rejeição 2 (Fs2)	847 Hz
Método	Algoritmo de Parks-McCleallan

Neste filtro primeiramente foram configurados os parâmetros para a utilização da função `firpmord()`. Ele é a função na qual irá nos retornar a ordem do filtro, a frequência normalizada, a magnitude das frequências e o peso dos ripples. Para isso devemos configurar os parâmetros de entrada deste mesma função. Para isso utilizamos das seguintes fórmulas para a entrada identificada pelo nome de *dev*, que represente os ripples entre a resposta de frequência.

Para a banda de rejeição:

$$dev = 10^{-\frac{As}{20}} \quad (4)$$

E para a banda de passagem:

$$dev = \frac{10^{\frac{Ap}{20}-1}}{10^{\frac{Ap}{20}}+1} \quad (5)$$

Com os parâmetros da função `firpmord()` ajustadas, a sua saída é usada como parâmetros de entrada para a `firpm()`.

3.3.1 Análise

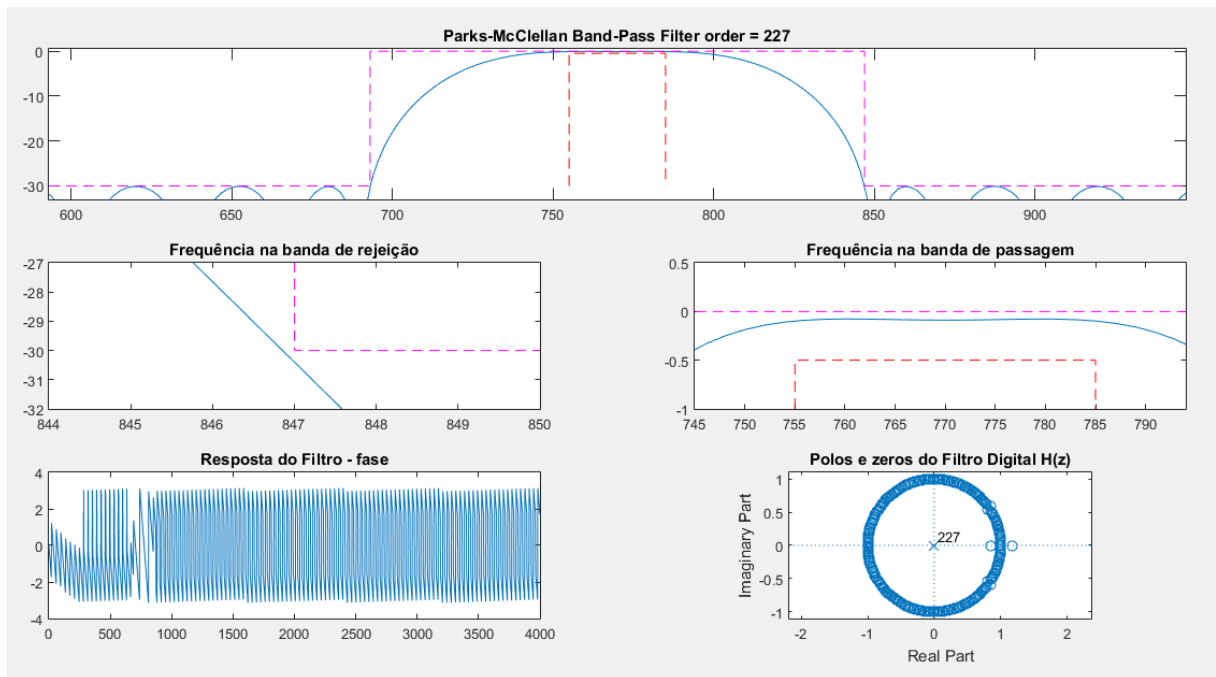


Figura 3 - Filtro passa faixa com algoritmo de Parks-McClellan. Fonte: elaboração própria

Graças ao poder das funções e do algoritmo de Parks-McClellan, o filtro precisou de pequenos ajustes para entrar na máscara delimitada pelas frequências. Nos ajustes, foram alterado os valores da variável *dev* e também a variável '*a*', na qual é responsável pela amplitude da frequência. Não foi necessário ajuste na ordem do filtro, onde foi obtido um filtro de ordem 227. Os resultados obtidos podem ser observados na figura 3.

4 Considerações Finais

Após as três implementações utilizando a metodologia de projeto de filtros digitais *FIR*, notou-se que, devido a sua característica de duração finita, a sua projeção é muito mais simples, utilizando os recursos do software MATLAB®, além de este tipo de filtro possuir fase linear, onde não há distorções do sinal, ele possui atraso de grupo constante. Apesar das vantagens, notou-se que o ajuste do filtro para se adequar às especificações é muito mais complexo.