

Andrey Gonçalves
Vinicius Luz Souza

Projeto de Filtros Digitais Quantizados IIR e FIR

Brasil
2019, Julho

Introdução

O projeto aqui apresentado surgiu relativamente à disciplina de Processamento de Sinais Digitais ministrada pelo Professor Marcos Moecke do curso de Engenharia de Telecomunicações. O mesmo abrange a quantização de filtros digitais IIR (Infinite Impulse Response) e FIR (Finite Impulse Response), de acordo com os projetos já realizados na primeira etapa da matéria. Em seguida, analisaremos se o filtro está dentro da máscara proposta, sua estabilidade e a quantidade de elementos lógicos necessário para desenvolver o filtro. Como primeiro, foi escolhido utilizar um filtro passa-baixas com duas implementações: IIR do tipo Eliptico e FIR com janela fixa. Como segundo filtro, apresentaremos um passa-altas do tipo IIR com o algoritmo de otimização Butterworth e FIR de janela ajustável. Em seguida será demonstrado um filtro passa-faixa IIR com uso do algoritmo Chebyshev II e um filtro FIR com aproximação de Parks-McClellan. Por fim, demonstraremos o projeto de um filtro rejeita-faixa IIR com algoritmo Chebyshev I e um filtro FIR com aproximação o Parks-McClellan.

Iremos analisar cada filtro e demonstrar as vantagens de cada estrutura, considerando o aspecto de tamanho de hardware e os resultados obtidos em cada implementação. Tais análises serão realizadas através do software MATLAB¹. Com o uso da ferramenta Fdatool², configuram-se as estruturas desejadas e a quantidade de bits dos protótipos. Por fim, exporta-se o filtro quantizado para ferramenta Simulink³, onde são realizadas as simulações com a versão final do filtro, já quantizado.

¹ MATLAB (MATrix LABoratory) trata-se de um software interativo de alta performance voltado para o cálculo numérico. O MATLAB integra análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos em ambiente fácil de usar onde problemas e soluções são expressos somente como eles são escritos matematicamente, ao contrário da programação tradicional.

² Fdatool - Filter Design and Analysis Tool (FDATool), é uma Ferramenta de Análise e Desenho de Filtros utilizada para projetar e analisar filtros rapidamente.

³ Simulink - desenvolvido pela companhia MathWorks, é uma ferramenta para modelagem, simulação e análise de sistemas dinâmicos.

1 Metodologia

Neste capítulo será demonstrado o caminho que percorremos para obter a quantização e seus respectivos valores dos protótipos referências especificados. A metodologia utilizada para quantização dos filtros projetados seguiu os seguintes passos:

1. Inicialmente, compilamos o código de projeto do filtro desejado no *MATLAB*.
2. Em seguida, inicializamos o software *Fdatool* e importamos os coeficientes do filtro.
3. Salvamos os filtros quantizados com 64 e 32 bits.
4. Após salvar os filtros de referência, buscamos reduzir ao máximo a quantidade de bits de quantização, sempre checando se o filtro permanece dentro da máscara. Para isto, exportamos o filtro e executamos o código de checagem da máscara no *MATLAB*. Neste ponto, também verificamos se o filtro permanece estável.
5. Ao reduzir o máximo a quantidade de bits, variamos a estrutura do filtro entre forma direta I, forma direta II, transposta e seções de segunda ordem (SOS), para os filtros IIR e, forma direta, forma direta simétrica e forma transposta, para os filtros FIR, sempre buscando reduzir ainda mais a quantidade de bits. Novamente neste passo, realizamos a checagem da máscara e da estabilidade do filtro.
6. Após chegarmos na quantidade mínima de bits de quantização, realizamos o filtro, exportando-o para a ferramenta *Simulink*.
7. Com o filtro já no *Simulink*, efetuamos a simulação utilizando duas entradas: um ruído branco e uma entrada de *Chirp*, iniciando com a frequência 0 e aumentando até metade da frequência de amostragem.

Obs.1: Para o filtro 2 deste relatório (filtro passa-altas), nos passos 4 e 5, além de reduzirmos ao mínimo a quantidade de bits, variamos também o projeto, para que possamos reduzir ainda mais a quantidade de elementos lógicos do *hardware* final.

Obs.2: Para o filtro 2 FIR deste relatório (filtro passa-altas), após a simulação no *Simulink*, geramos o código para realização do filtro em VHDL¹, bem como um *testbench* para simulação.

Obs.3: Conforme acordado com o professor, passo 7 só foi realizado para os filtros passa-altas FIR e passa-faixas IIR.

¹ VHDL - "VHSIC Hardware Description Language" é uma linguagem usada para facilitar o design de circuitos digitais em CPLDs, FPGAs e ASICs.

2 Desenvolvimento

2.1 Filtro Passa-Baixas

No primeiro projeto será utilizado o filtro passa-baixas IIR com aproximação do tipo Elíptico, em seguida um FIR com Janela Fixa, de acordo com as especificações da tabela abaixo.

Frequência de amostragem	10000 Hz
Frequência de passagem	2800 Hz
Frequência de rejeição	3200 Hz
Atenuação máxima na banda de passagem	1 dB
Atenuação máxima na banda de rejeição	40 dB
Ganho	0 dB

Tabela 1 – Especificação do filtro passa baixas

2.1.1 Filtro IIR do tipo Eliptico

Para desenvolvimento do projeto de quantização do protótipo IIR Passa Baixa, utilizamos o código já implementado no projeto anterior, adicionando a função *sos2tf* (equação abaixo), esta retorna a função transferência que descreve um sistema de tempo discreto *SOS* na seção de segunda ordem com ganho *G*. O cálculo da função *sos2tf* utiliza a convolução para multiplicar todos os polinômios de segunda ordem do numerador e do denominador.

$$[Num, Den] = sos2tf(SOS, G); \quad (2.1)$$

Com o uso do software *Fdatool*, importamos o filtro do *workspace* do *MATLAB* e, utilizando a estrutura de Forma Direta II, quantizamos o filtro com a menor quantidade de bits, levando em conta os parâmetros e máscara dada no início do projeto. O resultado da quantização pode ser vista na imagem abaixo:

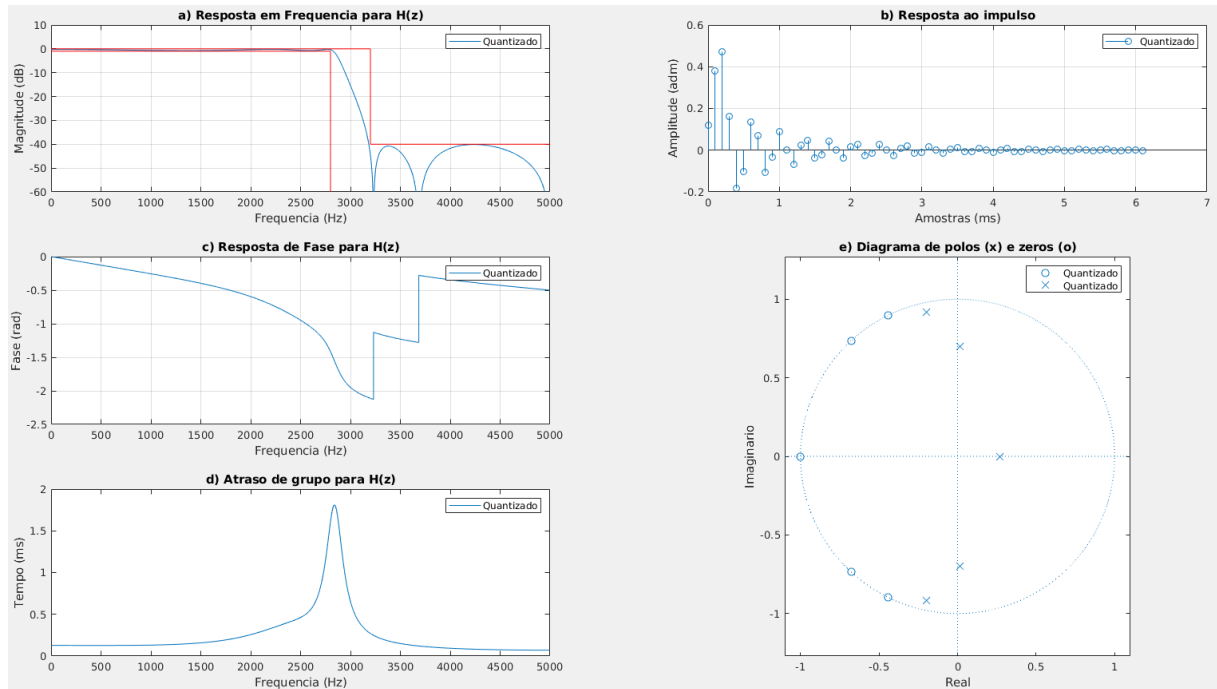


Figura 1 – Visualização geral do filtro quantizado

Pode-se observar, conforme exposto nos gráficos, os resultados obtidos pelo filtro passa baixas IIR com aproximação *Elíptico* quantizado a partir do filtro referência especificado. Consta na figura a resposta em frequência após a transformação bilinear, também a resposta ao impulso, resposta em fase, o atraso de grupo (nota-se que não é constante) e o diagrama de pólos e zeros.

2.1.1.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro IIR passa baixas, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposto uma comparação sobreposta na mesma figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

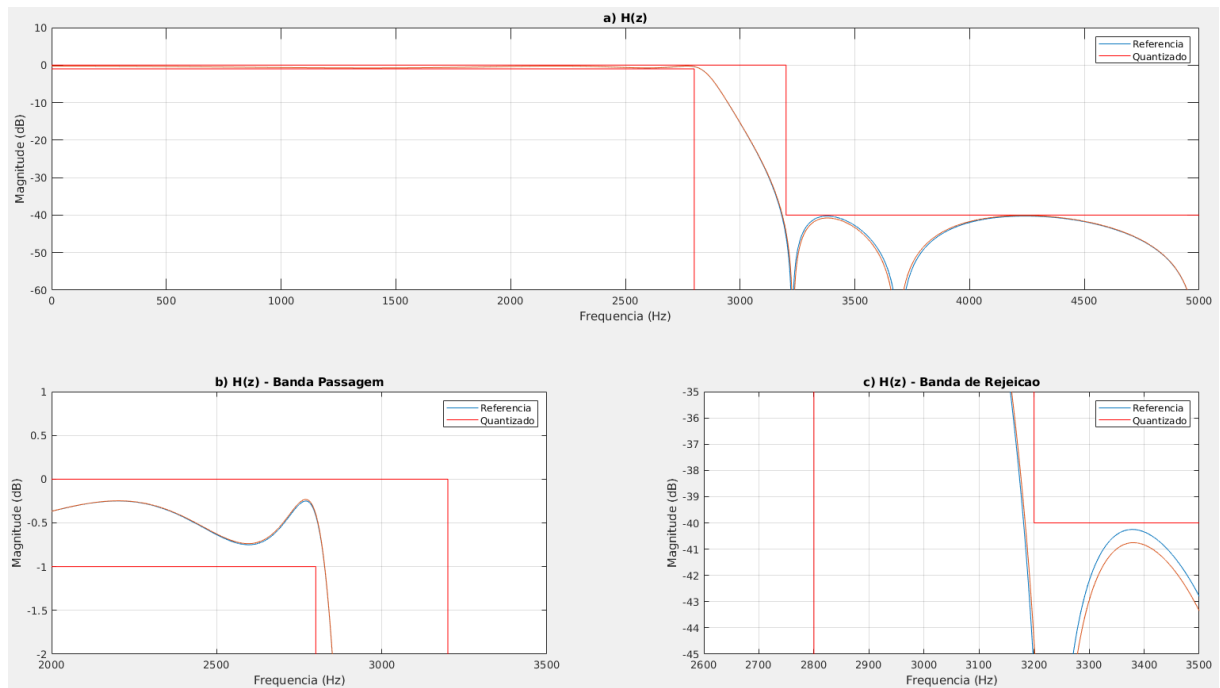


Figura 2 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que o resultado é praticamente o mesmo, o filtro continua estável.

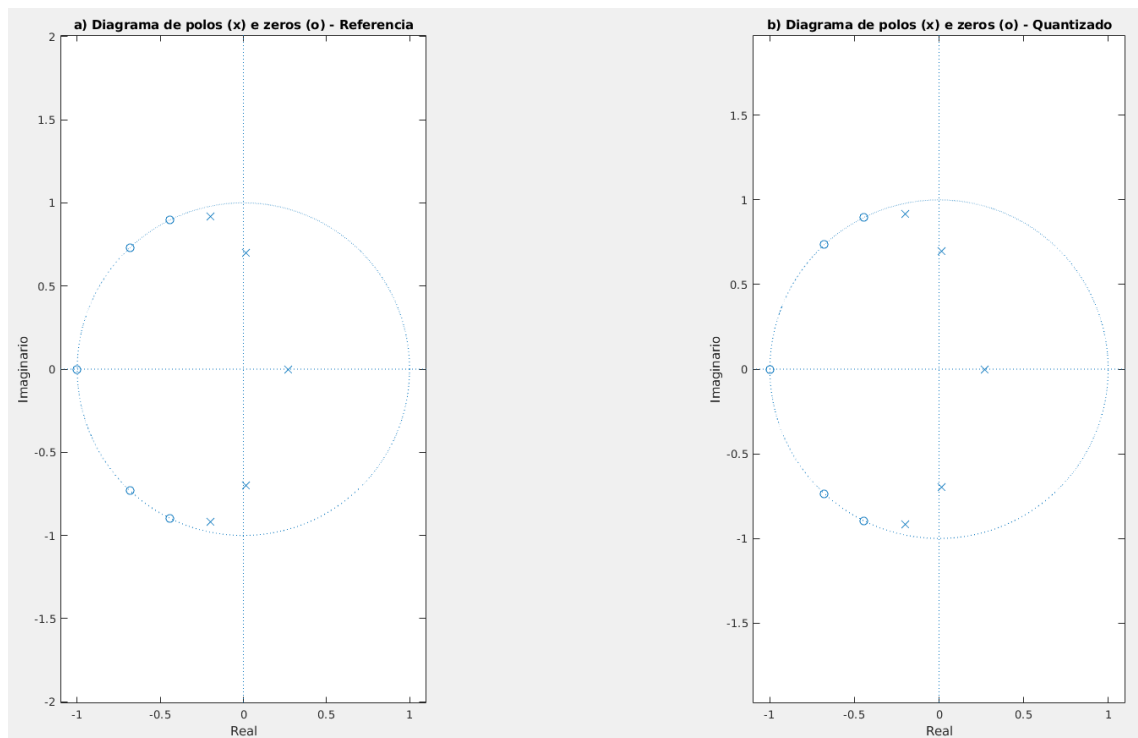


Figura 3 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados para estimativa de elementos de área lógica do filtro referência e do filtro quantizado. O dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*LP IIR Ref*) e o filtro quantizado (*LP IIR Quant*). Em ambos os filtros foram aplicados a estrutura Forma Direta II.

Na tabela é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores, Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.2)$$

A estimativa de quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.3)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
LP IIR Ref	5	64	D.II	11	10	5	2368	2048
LP IIR Quant	5	13	D.II	11	10	5	481	416

Tabela 2 – Comparativo dos resultados obtidos no filtro referência com o quantizado

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de referência (projeto).

$$H(z) = \frac{0.12015 z^5 + 0.37904 z^4 + 0.63184 z^3 + 0.63184 z^2 + 0.37904 z + 0.12015}{z^5 + 0.0058624 z^4 + 1.3167 z^3 - 0.32649 z^2 + 0.44365 z - 0.17764} \quad (2.4)$$

A equação a seguir expressa o filtro quantizado.

$$Hq(z) = \frac{0.12134 z^5 + 0.39331 z^4 + 0.6604 z^3 + 0.6604 z^2 + 0.39331 z + 0.12134}{z^5 + 0.095215 z^4 + 1.2554 z^3 - 0.20068 z^2 + 0.38428 z - 0.11572} \quad (2.5)$$

Este filtro foi reprojetoado para melhor adequação à máscara e possibilitar uma redução no tamanho do hardware. Foram realizados ajustes no ganho e na atenuação de passagem.

2.1.2 Filtro FIR do tipo Janela Fixa

Para desenvolvimento do filtro de janela fixa quantizado, selecionamos o código do projeto anterior que foi utilizado o método de janela de *Hann*, devido a atenuação que precisávamos para a banda de rejeição (40dB). Em seguida, utilizamos o software *Fdatool*, importando o filtro do *workspace* do *MATLAB* e aplicando a estrutura de Forma Direta Simétrica para quantização do filtro com a menor quantidade de bits possível, levando em conta os parâmetros e máscara dada no início do projeto. O resultado da quantização pode ser vista na imagem a seguir:

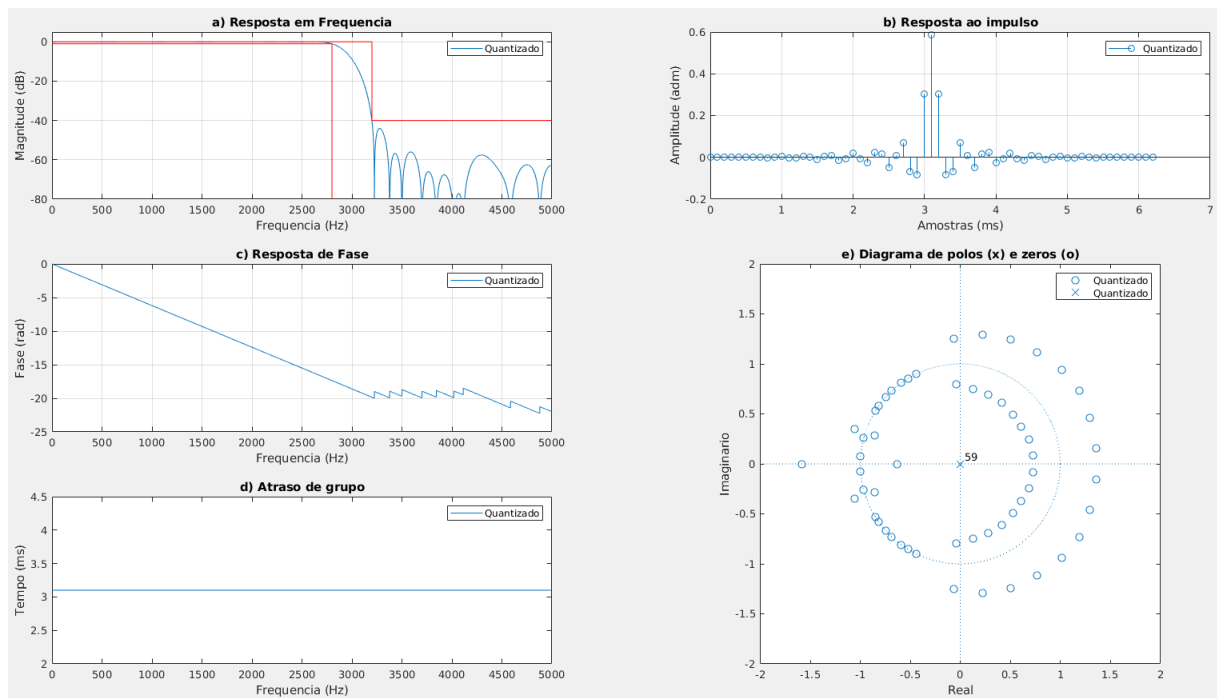


Figura 4 – Visualização geral do filtro quantizado

Na figura anterior, observa-se os resultados obtidos no *MATLAB* pelo filtro passa baixas FIR com aproximação janela de *Hann* quantizado a partir do filtro referência especificado. A resposta em frequência após a transformação bilinear é apresentada na primeira imagem(a) da figura acima, em seguida na segunda imagem(b) da figura é exposto à resposta ao impulso, também pode-se analisar o gráfico de resposta em fase, e o atraso de grupo constante constante de 3,1ms(d) e por fim o diagrama de pólos e zeros (e).

2.1.2.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro FIR passa baixas, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposto uma comparação sobreposta na mesma

figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

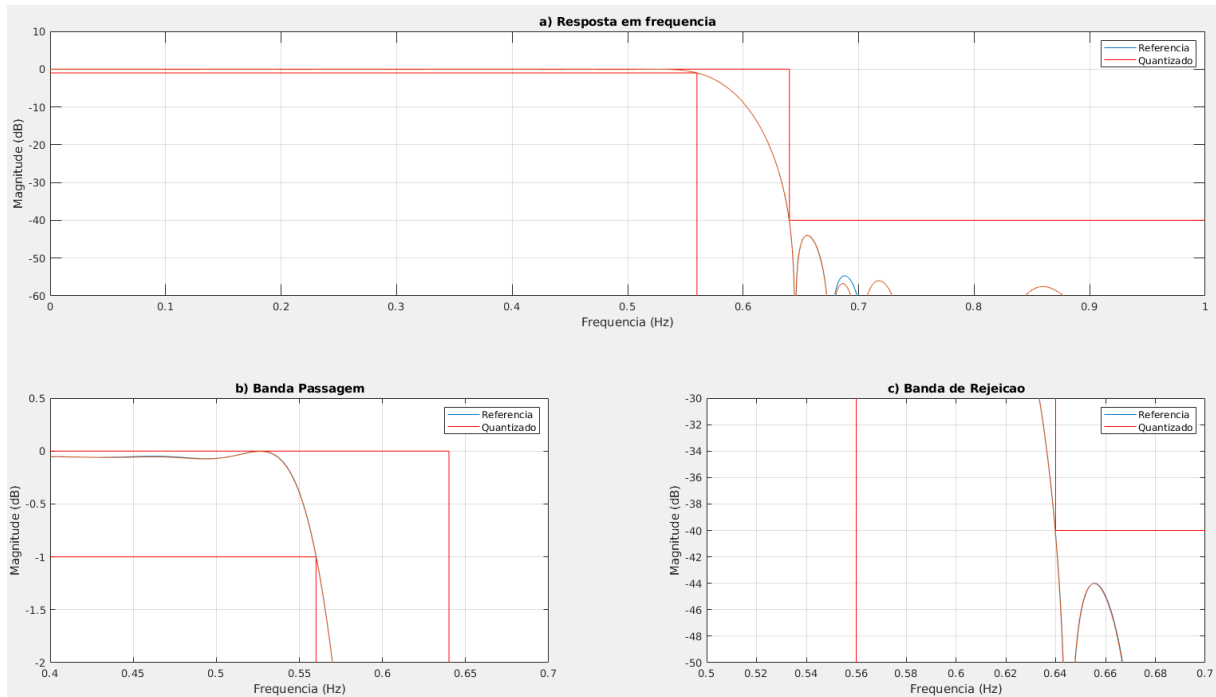


Figura 5 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que há diferença no diagrama de polos e zeros, todavia, a estabilidade do filtro é mantida.

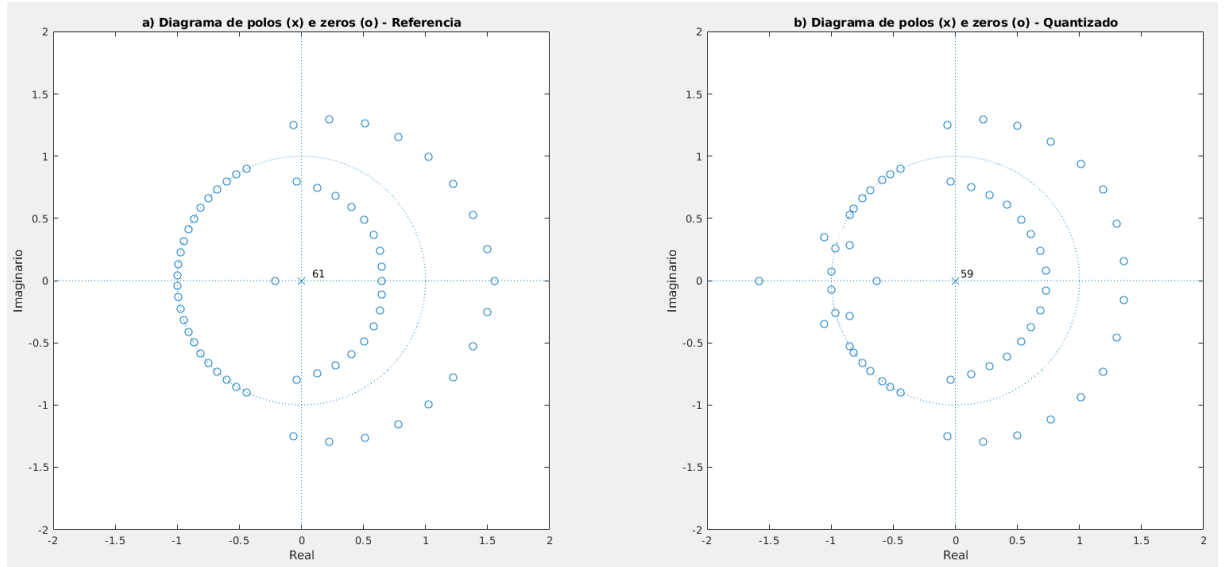


Figura 6 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Na comparação dos polos e zeros, observamos uma diferença em relação ao filtro quantizado. Todavia, a estabilidade do filtro é mantida, preservando o correto funcionamento do filtro.

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados de elementos de área lógica do filtro referência e do filtro quantizado. O dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*LP FIR Ref*) e o filtro quantizado (*LP FIR Quant*). Em ambos os filtros foram aplicados a estrutura Forma Direta Simétrica (*D SYM*).

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
LP FIR Ref	61	64	D SYM	31	61	62	11840	7872
LP FIR Quant	61	13	D SYM	31	61	62	2405	1599

Tabela 3 – Comparativo dos resultados obtidos no filtro referência com o quantizado

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de

referência (projeto).

$$\begin{aligned}
H(z) = & -0.000020176 z^{61} - 0.000047808 z^{60} + 0.00025584 z^{59} - 0.000061323 z^{58} \\
& -0.00069915 z^{57} + 0.00073187 z^{56} + 0.00087503 z^{55} - 0.0020449 z^{54} - 0.000030078 z^{53} \\
& +0.0034134 z^{52} - 0.0023555 z^{51} - 0.0035954 z^{50} + 0.0059638 z^{49} + 0.0011884 z^{48} \\
& -0.0092974 z^{47} + 0.0045269 z^{46} + 0.0098872 z^{45} - 0.012745 z^{44} - 0.0051021 z^{43} \\
& +0.020706 z^{42} - 0.0066978 z^{41} - 0.023896 z^{40} + 0.025145 z^{39} + 0.016629 z^{38} \\
& -0.047478 z^{37} + 0.0082924 z^{36} + 0.069068 z^{35} - 0.067339 z^{34} - 0.084747 z^{33} \\
& +0.30263 z^{32} + 0.58727 z^{31} + 0.30263 z^{30} - 0.084747 z^{29} - 0.067339 z^{28} \\
& +0.069068 z^{27} + 0.0082924 z^{26} - 0.047478 z^{25} + 0.016629 z^{24} + 0.025145 z^{23} \\
& -0.023896 z^{22} - 0.0066978 z^{21} + 0.020706 z^{20} - 0.0051021 z^{19} - 0.012745 z^{18} \\
& +0.0098872 z^{17} + 0.0045269 z^{16} - 0.0092974 z^{15} + 0.0011884 z^{14} + 0.0059638 z^{13} \\
& -0.0035954 z^{12} - 0.0023555 z^{11} + 0.0034134 z^{10} - 0.000030078 z^9 - 0.0020449 z^8 \\
& +0.00087503 z^7 + 0.00073187 z^6 - 0.00069915 z^5 - 0.000061323 z^4 + 0.00025584 z^3 \\
& -0.000047808 z^2 - 0.000020176 z
\end{aligned} \tag{2.6}$$

A equação a seguir expressa o filtro quantizado.

$$\begin{aligned}
Hq(z) = & 0.00024414 z^{59} - 0.00073242 z^{57} + 0.00073242 z^{56} + 0.00097656 z^{55} - 0.0019531 z^{54} \\
& +0.003418 z^{52} - 0.0024414 z^{51} - 0.0036621 z^{50} + 0.0058594 z^{49} + 0.0012207 z^{48} - 0.0092773 z^{47} \\
& +0.0046387 z^{46} + 0.0097656 z^{45} - 0.012695 z^{44} - 0.005127 z^{43} + 0.020752 z^{42} - 0.0065918 z^{41} \\
& -0.023926 z^{40} + 0.025146 z^{39} + 0.016602 z^{38} - 0.047363 z^{37} + 0.0083008 z^{36} + 0.069092 z^{35} \\
& -0.067383 z^{34} - 0.084717 z^{33} + 0.30273 z^{32} + 0.58716 z^{31} + 0.30273 z^{30} - 0.084717 z^{29} \\
& -0.067383 z^{28} + 0.069092 z^{27} + 0.0083008 z^{26} - 0.047363 z^{25} + 0.016602 z^{24} + 0.025146 z^{23} \\
& -0.023926 z^{22} - 0.0065918 z^{21} + 0.020752 z^{20} - 0.005127 z^{19} - 0.012695 z^{18} + 0.0097656 z^{17} \\
& +0.0046387 z^{16} - 0.0092773 z^{15} + 0.0012207 z^{14} + 0.0058594 z^{13} - 0.0036621 z^{12} - 0.0024414 z^{11} \\
& +0.003418 z^{10} - 0.0019531 z^8 + 0.00097656 z^7 + 0.00073242 z^6 - 0.00073242 z^5 + 0.00024414 z^3
\end{aligned} \tag{2.7}$$

2.2 Filtro Passa-Altas

O segundo projeto utiliza-se um filtro Passa Altas IIR com aproximação do tipo *Butterworth*, em seguida um FIR com Janela Ajustável, com as especificações da tabela a seguir.

Frequência de amostragem	10000 Hz
Frequência de passagem	2800 Hz
Frequência de rejeição	3200 Hz
Atenuação máxima na banda de passagem	0.5 dB
Atenuação máxima na banda de rejeição	20 dB
Ganho	0 dB

Tabela 4 – Especificação do filtro Passa Alta

2.2.1 Filtro IIR do tipo Butterworth

No protótipo do filtro Passa Alta do tipo IIR foram especificados os parâmetros da Tabela acima. As equações de cálculo de ordem e de pólos foram baseadas na aproximação *Butterworth*. Este filtro teve seus coeficientes calculados manualmente no primeiro trabalho da disciplina, ou seja, sem a utilização das funções para filtros *Butterworth* do *MATLAB*.

Utilizando o software *Fdatool*, importamos o filtro do *workspace* do *MATLAB*, com uso da estrutura de Forma Direta II. Em seguida, quantizamos o filtro com a menor quantidade de bits sem sair dos parâmetros iniciais de especificação e identificamos a estrutura Forma Direta II com Seções de Segunda Ordem como sendo a de melhor implementação. O resultado da quantização pode ser vista na imagem abaixo:

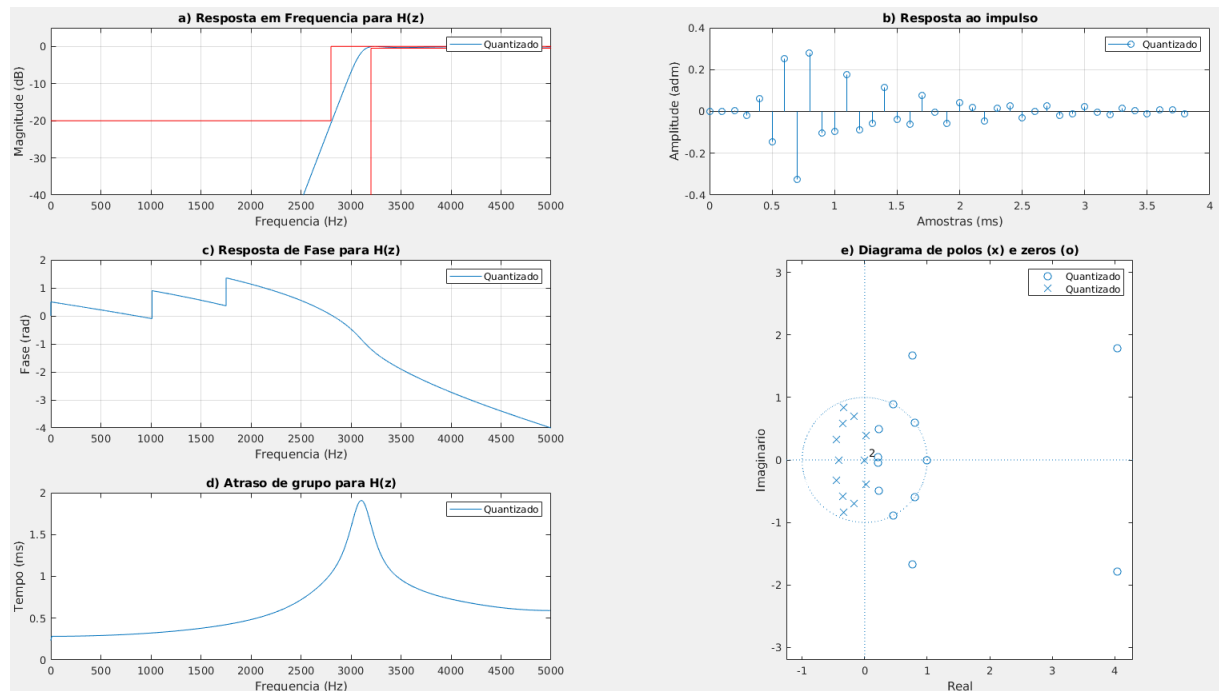


Figura 7 – Visualização geral do filtro quantizado

Pode-se observar, conforme exposto nos gráficos, os resultados obtidos pelo filtro

passa altas IIR com aproximação *Butterworth*, quantizado a partir do filtro referência especificado. Consta na figura a resposta em frequência após a transformação bilinear, também a resposta ao impulso, resposta em fase, o atraso de grupo (nota-se que não é constante) e o diagrama de pólos e zeros.

2.2.1.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro IIR passa altas, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposto uma comparação sobreposta na mesma figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

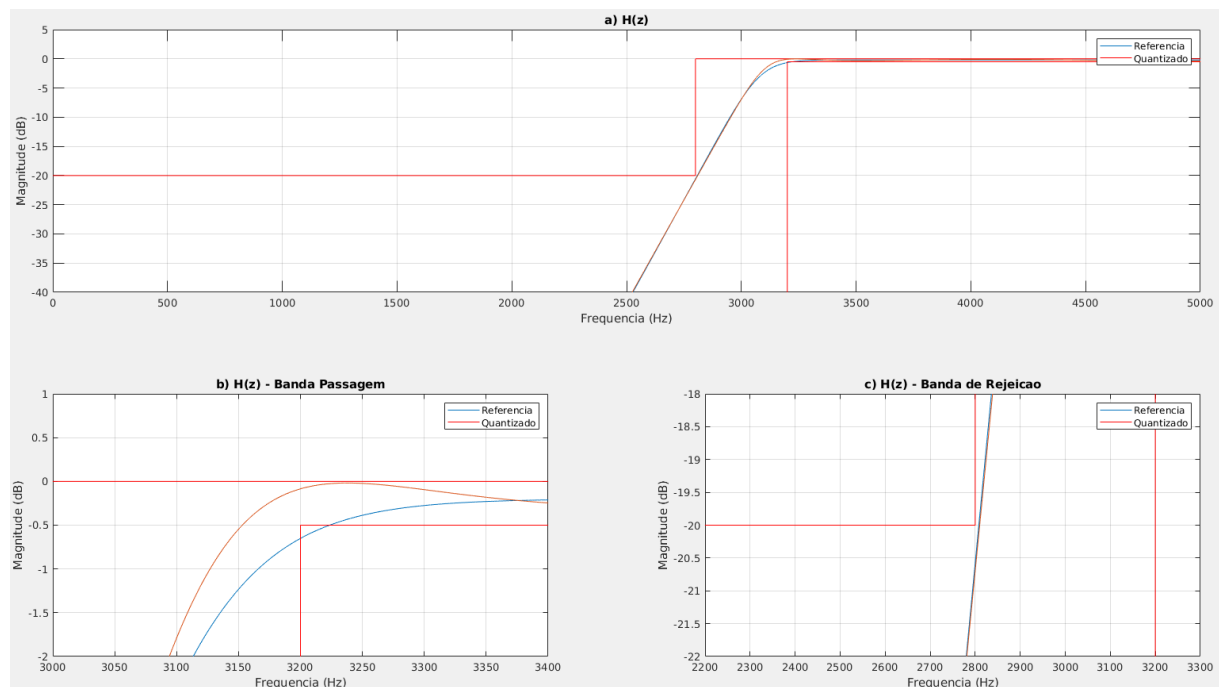


Figura 8 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

Observando a imagem acima, podemos verificar que o filtro quantizado com 64 bits (referência) está fora da máscara de especificação. Este fato aconteceu, devido às alterações realizadas no projeto para minimizar a quantidade de elementos lógicos do

hardware final. Entretanto, após a quantização com 12 bits, notamos que o filtro fica dentro do especificado, atendendo aos requisitos de projeto.

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que há bastante diferença entre os filtro de referência e o filtro quantizado. Todavia, a estabilidade não é perdida.

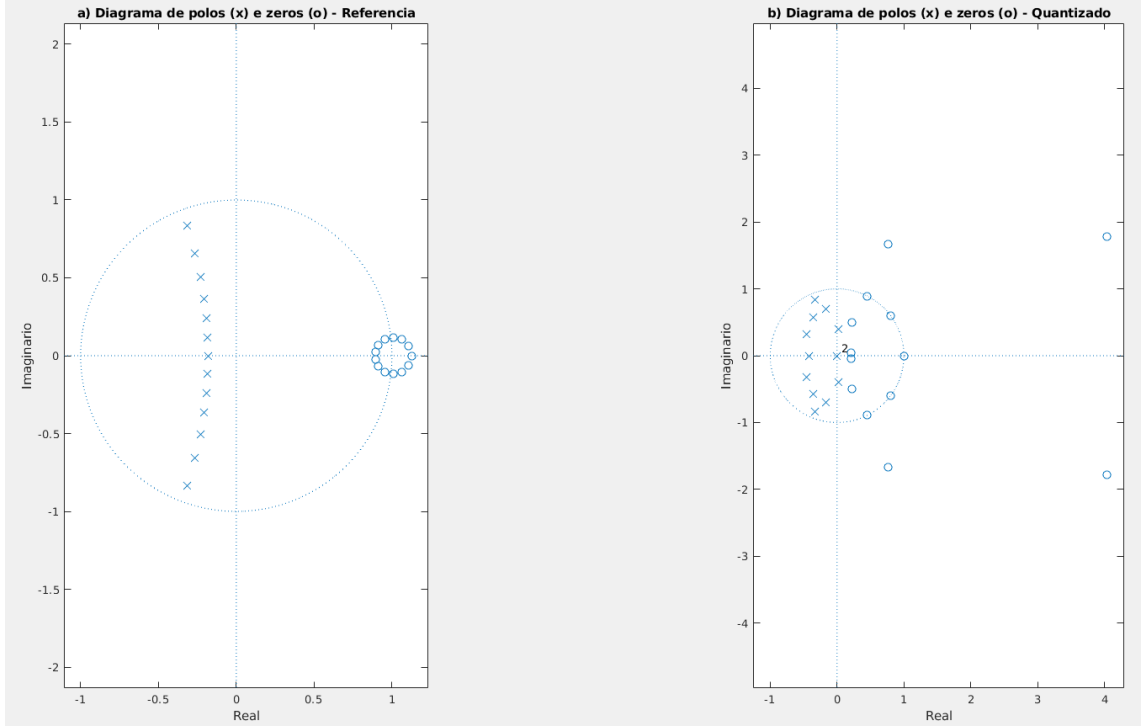


Figura 9 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados de elementos de área lógica do filtro referência e do filtro quantizado. O dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*HP IIR Ref*) e o filtro quantizado (*HP IIR Quant*). Em ambos os filtros foram aplicados a estrutura Forma Direta II de Seção de Segunda Ordem(SOS) (*D.II SOS*).

Na tabela é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores, Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.8)$$

A estimativa de quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.9)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
HP IIR Ref	13	64	D.II SOS	27	26	13	5952	5120
HP IIR Quant	13	12	D.II SOS	25	24	13	1044	888

Tabela 5 – Comparativo dos resultados obtidos no filtro referência com o quantizado

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de referência (projeto anterior).

$$H(z) = \frac{0.000034576 z^{13} - 0.00044948 z^{12} + 0.0026969 z^{11} - 0.0098886 z^{10} + 0.024722 z^9 - 0.0098886 z^3 - 0.0026969 z^2 + 0.00044948 z - 0.000034576}{z^{13} + 2.9853 z^{12} + 5.6831 z^{11} + 7.2858 z^{10} + 7.0854 z^9 + 5.2939 z^8 + 3.1209 z^7 + 1.4468 z^6 + 0.52602 z^5 + 0.14696 z^4 + 0.030594 z^3 + 0.0044727 z^2 + 0.00041083 z + 0.000017848} \quad (2.10)$$

A equação a seguir expressa o filtro quantizado após os ajustes realizados, ou seja, a versão final do filtro para implementação.

$$Hq(z) = \frac{0.000030518 z^{13} - 0.00042725 z^{12} + 0.0026547 z^{11} - 0.0096716 z^{10} + 0.024192 z^9 - 0.043558 z^8 + 0.058062 z^7 - 0.058033 z^6 + 0.043483 z^5 - 0.024104 z^4 + 0.0096025 z^3 - 0.002618 z^2 + 0.00041453 z - 0.000027798}{z^{13} + 2.9834 z^{12} + 5.6812 z^{11} + 7.2813 z^{10} + 7.0822 z^9 + 5.2903 z^8 + 3.12 z^7 + 1.4452 z^6 + 0.52753 z^5 + 0.14857 z^4 + 0.03129 z^3 + 0.0039099 z^2} \quad (2.11)$$

No projeto deste filtro foram realizados ajustes nas frequências de passagem e rejeição, atenuação de passagem e ganho. Todos os ajustes foram realizados com a finalidade de reduzir o tamanho do *hardware* final. Além destes ajustes, tentamos também ir além, ou seja, aumentar a ordem do filtro e buscar reduzir ainda mais a quantidade de bits. Entretanto, a melhor configuração encontrada foi a que expusemos neste relatório.

2.2.2 Filtro FIR do tipo Janela Ajustável

No protótipo do filtro passa-altas do tipo janela ajustável foram especificados os parâmetros iniciais do filtro referência. A melhor aproximação que obtivemos utilizando

uma janela ajustável foi com a de *Kaiser*. Então fizemos uso deste código para fazer a quantização do filtro FIR Passa Altas. Utilizamos o software *Fdatool* para importar o filtro do *workspace* do *MATLAB* e realizar a quantização. A melhor estrutura encontrada após os ajustes foi a de Forma Direta Simétrica com precisão finita, em procura da menor quantidade de bits possível, levando em conta os parâmetros e máscara dada no início do projeto. O resultado da quantização deste filtro pode ser vista na imagem a seguir:

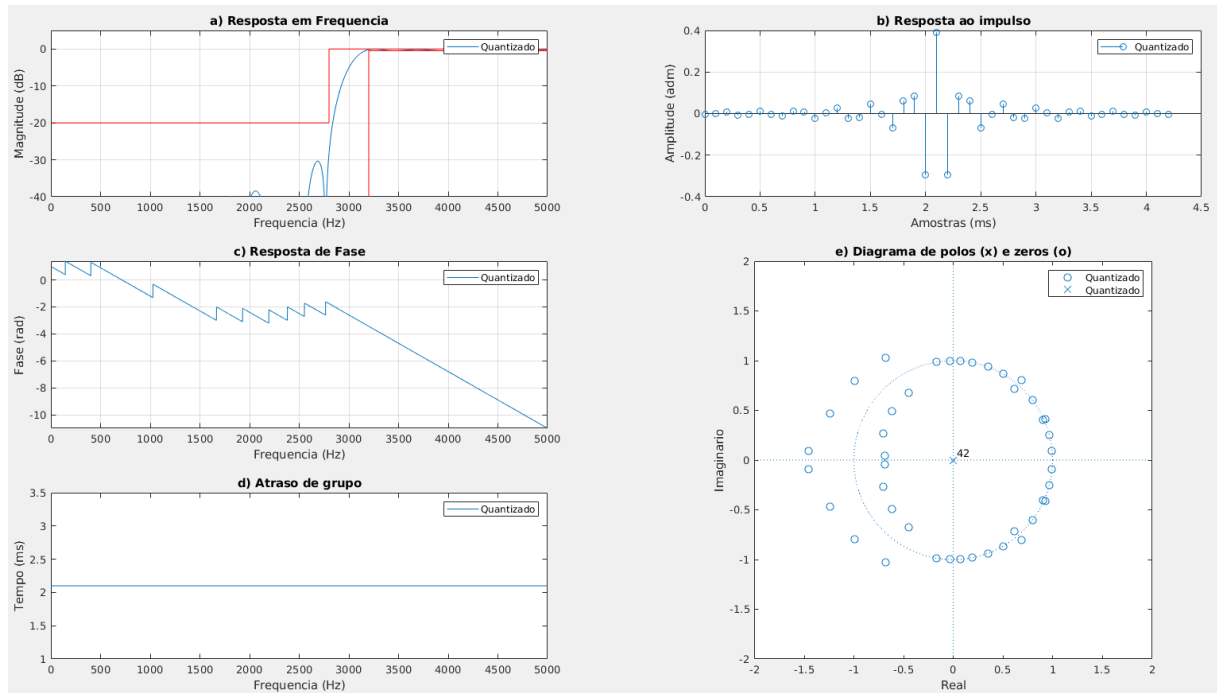


Figura 10 – Visualização geral do filtro quantizado

Pode-se observar, conforme exposto nos gráficos, os resultados obtidos pelo filtro passa altas FIR com aproximação da janela de *Kaiser* quantizado a partir do filtro referência especificado. Consta na figura a resposta em frequência após a transformação bilinear, também a resposta ao impulso, resposta em fase, o atraso de grupo (nota-se que é constante e tem um atraso de 2,1ms) e o diagrama de pólos e zeros.

Após o filtro quantizado no *Fdatool*, exportamos a estrutura para realizar o projeto no *Simulink*. Adicionamos alguns recursos para fazer a simulação, como sinais ruído de entrada, analisador de espectro e *Time Scope*. Os blocos do projeto podem ser visualizados na figura abaixo.

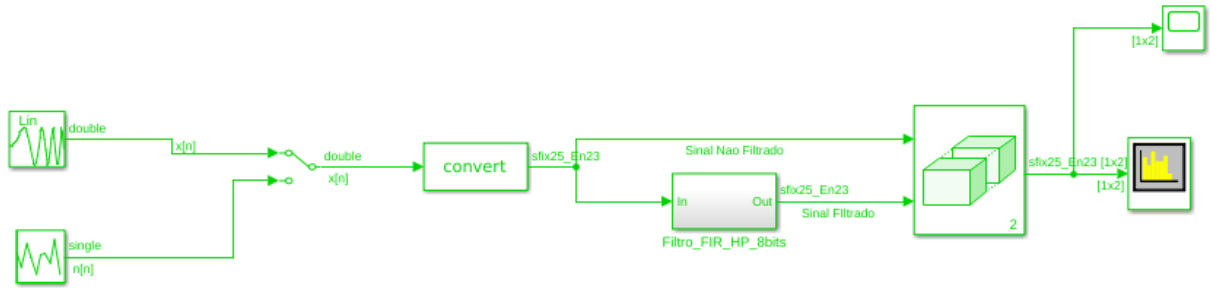


Figura 11 – Circuito de simulação do filtro Passa Altas FIR no Simulink

Na figura a seguir é possível visualizar o resultado da simulação com o sinal de entrada *chirp* no analisador de espectro do software *Simulink*. A cor azul é do sinal filtrado pelo filtro quantizado e o sinal em vermelha é o sinal não filtrado.

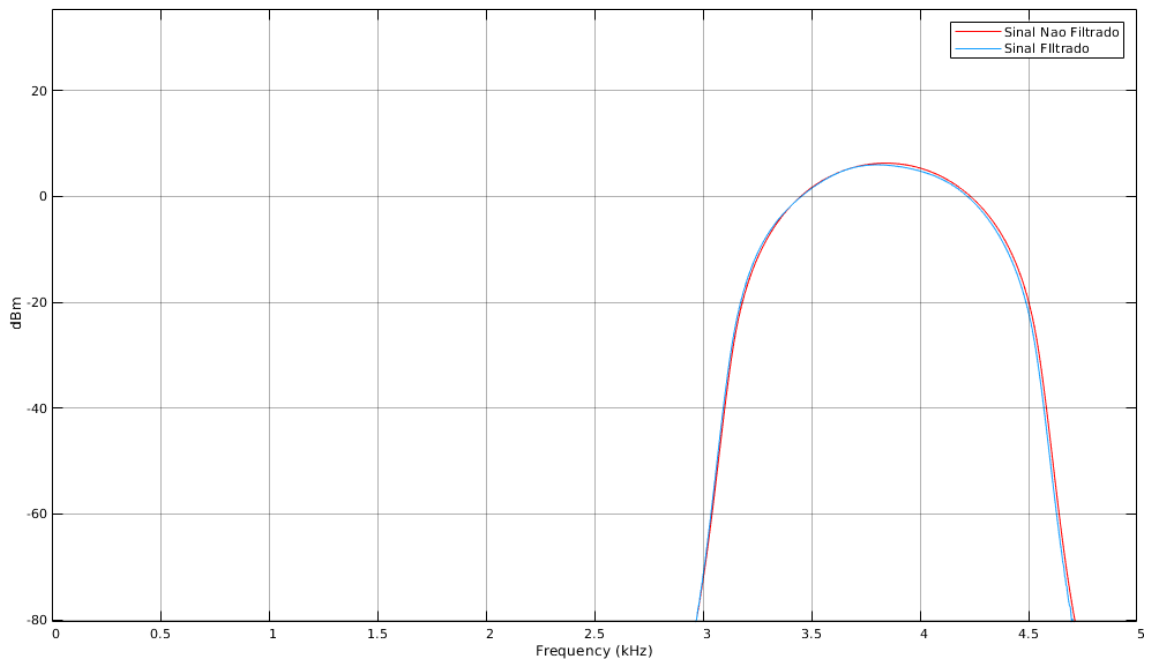


Figura 12 – Resultado da simulação filtro HP no analisador de espectro com chirp na entrada

O resultado obtido no analisador de tempo do *Simulink* da simulação do filtro passa altas com a função de entrada *chirp* pode-se ser verificada abaixo na figura 13. Nota-se que apenas as frequências altas passam, demonstrando o funcionamento do filtro.

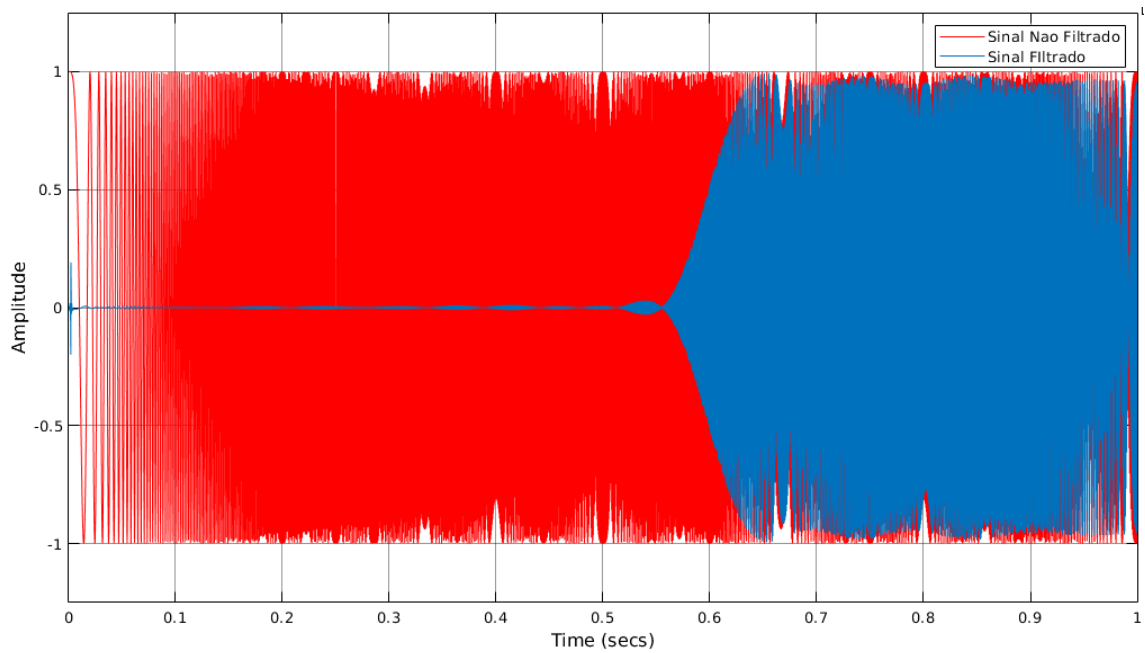


Figura 13 – Resultado da simulação filtro HP no analisador de tempo com chirp na entrada

Aplicamos um ruído branco na entrada do filtro projetado e obtivemos o resultado do sinal no analisador de espectro (figura 14) e no *Time Scope* (figura 15). Pode-se observar no analisador de espectro o resultado do sinal original (não filtrado, cor vermelha) e o sinal filtrado pelo protótipo (cor azul), é notório que o sinal foi filtrado corretamente com as especificações do filtro projetado, ou seja, o sinal passou somente as frequências altas e as baixas foram atenuadas, desta forma, evidenciando o funcionamento do filtro desenvolvido.

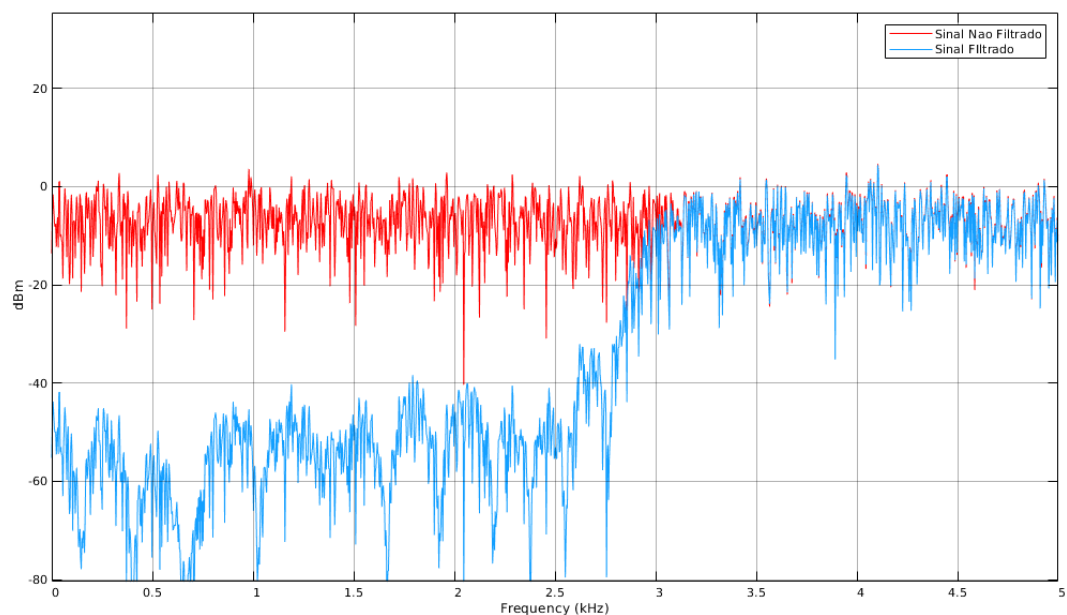


Figura 14 – Resultado da simulação filtro HP no analisador de espectro com ruído branco na entrada

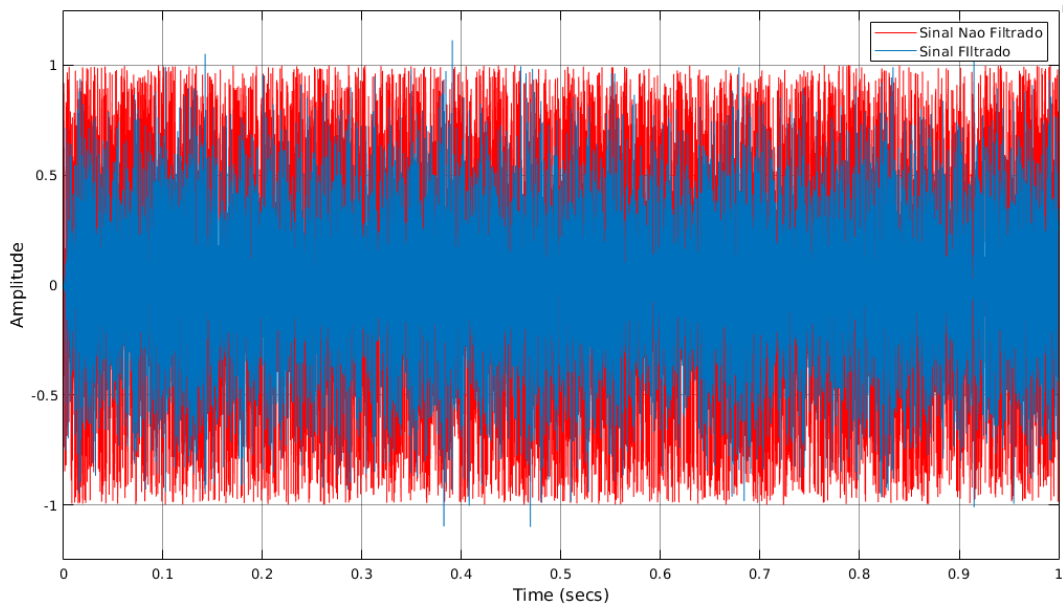


Figura 15 – Resultado da simulação filtro HP no analisador de tempo com ruído branco na entrada

A partir do *Simulink*, geramos o código VHDL que então foi exportado para o *Modelsim* do Quartus, onde foi possível analisar os sinais filtrados pelo protótipo. O testbench expõe o hardware projetado funcionando, ou seja, aqui, temos uma descrição de hardware pronta para ser implementada.

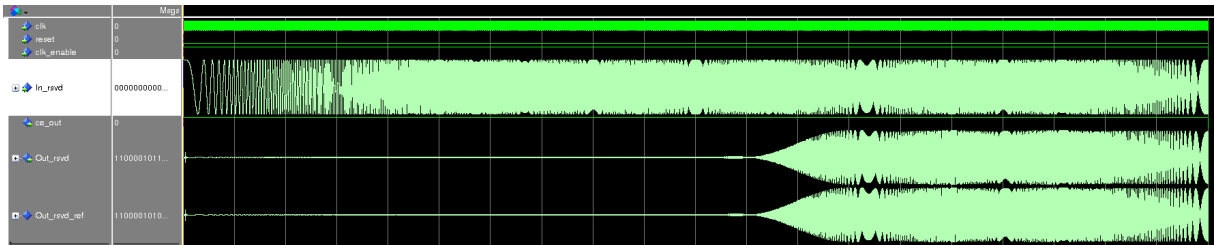


Figura 16 – Resultado do testbench realizado a partir da descrição VHDL gerada

2.2.2.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro FIR passa altas, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposto uma comparação sobreposta na mesma figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a

segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

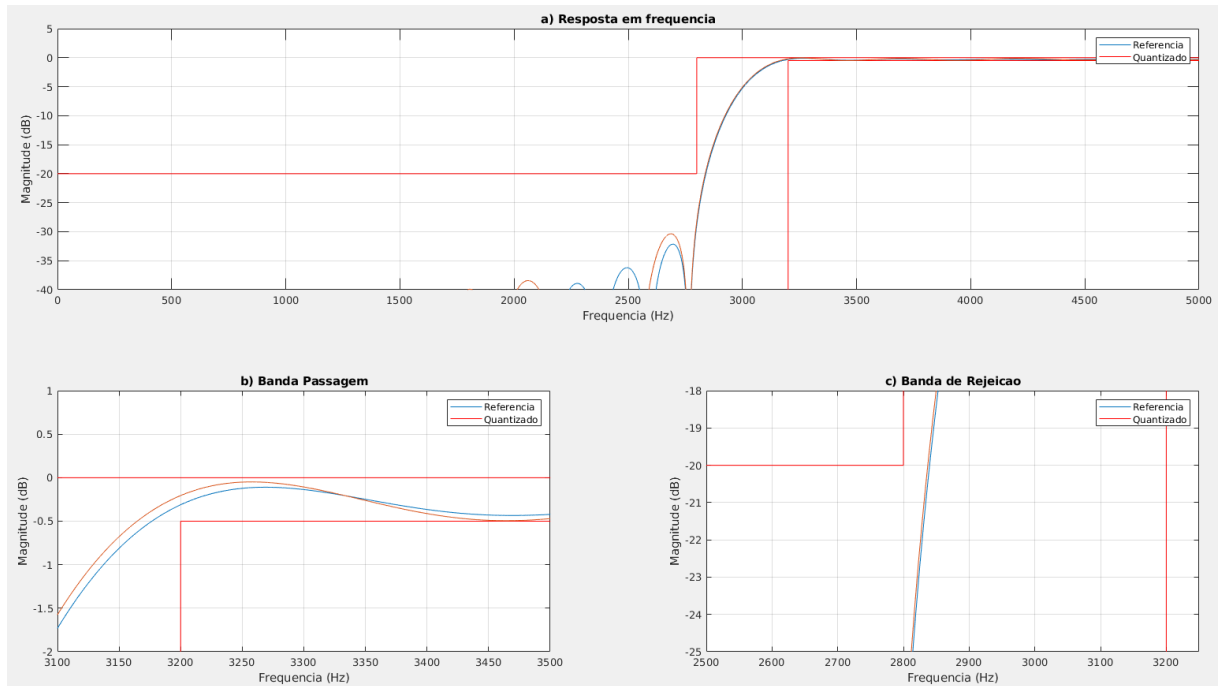


Figura 17 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que o resultado é bastante semelhante e a estabilidade do filtro é mantida.

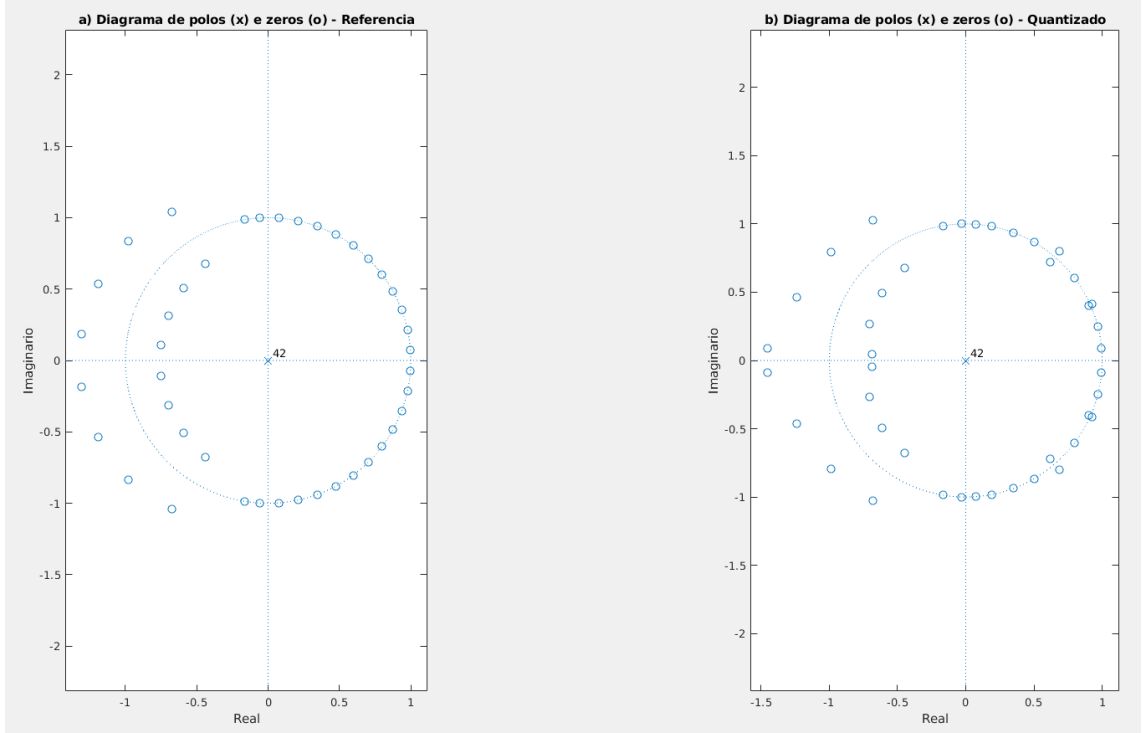


Figura 18 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados de elementos de área lógica do filtro referência e do filtro quantizado. Os dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*HP FIR Ref*) e o filtro quantizado (*HP FIR Quant*). Em ambos os filtros foram aplicados a estrutura Forma Direta Simétrica (*D SYM*).

Na tabela é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores, Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.12)$$

A quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.13)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
HP FIR Ref	38	64	D SYM	20	39	38	7488	5056
HP FIR Quant	42	8	D SYM	22	43	42	1032	696

Tabela 6 – Comparativo dos resultados obtidos no filtro referência com o quantizado

No projeto deste filtro, efetuamos sua realização em *hardware* com descrição VHDL e a quantidade de elementos lógicos gerados pode ser visualizada na figura a seguir.

Multipliers	16
Adders/Subtractors	43
Registers	42
RAMs	0
Multiplexers	1

Figura 19 – Elementos lógicos implementação em VHDL do filtro HP

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de referência (projeto).

$$\begin{aligned}
H(z) = & 0.006197 z^{38} - 0.0063002 z^{37} - 0.004002 z^{36} + 0.011124 z^{35} - 0.0023097 z^{34} \\
& - 0.013023 z^{33} + 0.011878 z^{32} + 0.0089355 z^{31} - 0.021636 z^{30} + 0.0030371 z^{29} \\
& + 0.026617 z^{28} - 0.022518 z^{27} - 0.020697 z^{26} + 0.046463 z^{25} - 0.0035316 z^{24} - 0.069766 z^{23} \\
& + 0.06217 z^{22} + 0.086739 z^{21} - 0.29319 z^{20} + 0.39035 z^{19} - 0.29319 z^{18} + 0.086739 z^{17} \\
& + 0.06217 z^{16} - 0.069766 z^{15} - 0.0035316 z^{14} + 0.046463 z^{13} - 0.020697 z^{12} - 0.022518 z^{11} \\
& + 0.026617 z^{10} + 0.0030371 z^9 - 0.021636 z^8 + 0.0089355 z^7 + 0.011878 z^6 - 0.013023 z^5 \\
& - 0.0023097 z^4 + 0.011124 z^3 - 0.004002 z^2 - 0.0063002 z + 0.006197
\end{aligned}
\tag{2.14}$$

A equação a seguir expressa o filtro quantizado após ajustes no projeto.

$$\begin{aligned}
Hq(z) = & -0.0039062 z^{42} + 0.0078125 z^{40} - 0.0078125 z^{39} - 0.0039062 z^{38} + 0.011719 z^{37} \\
& -0.0039062 z^{36} - 0.011719 z^{35} + 0.011719 z^{34} + 0.0078125 z^{33} - 0.023438 z^{32} \\
& +0.0039062 z^{31} + 0.027344 z^{30} - 0.023438 z^{29} - 0.019531 z^{28} + 0.046875 z^{27} \\
& -0.0039062 z^{26} - 0.070312 z^{25} + 0.0625 z^{24} + 0.085938 z^{23} - 0.29297 z^{22} + 0.39062 z^{21} \\
& -0.29297 z^{20} + 0.085938 z^{19} + 0.0625 z^{18} - 0.070312 z^{17} - 0.0039062 z^{16} + 0.046875 z^{15} \\
& -0.019531 z^{14} - 0.023438 z^{13} + 0.027344 z^{12} + 0.0039062 z^{11} - 0.023438 z^{10} + 0.0078125 z^9 \\
& +0.011719 z^8 - 0.011719 z^7 - 0.0039062 z^6 + 0.011719 z^5 - 0.0039062 z^4 - 0.0078125 z^3 \\
& +0.0078125 z^2 - 0.0039062
\end{aligned}
\tag{2.15}$$

Para esse protótipo nós alteramos o código a fim de procurar a melhor quantização do filtro, de modo que realizamos ajustes na atenuação de rejeição, ganho e ordem, obtendo o melhor resultado da quantização.

Aqui cabe uma observação importante: foi alterada a ordem do filtro, passando de 38 para 42, entretanto através do aumento da ordem, foi possível reduzirmos a quantidade de bits de quantização e, por consequência, conseguimos um *hardware menor*. Ou seja, mesmo com a ordem maior, através de um filtro mais seletivo, projetamos um hardware menor. Além disso conseguimos projetar um filtro do tipo FIR com *hardware* menor que o mesmo filtro de tipo IIR (filtro passa-altas da seção 2.2.1), mesmo com ordem bastante superior (13 do filtro IIR contra 42 do filtro FIR). E, como grande vantagem, ganhamos linearidade na resposta de fase e atraso de grupo constante.

2.3 Filtro Passa-Faixa

A seguir será quantizado um filtro Passa Faixa IIR com aproximação do tipo *Chebyshev II*, em seguida um FIR com algoritmo de *Parks-McClellan*, com as especificações da tabela a seguir.

Frequência de amostragem	10000 Hz
F1	3000 Hz
F2	3200 Hz
F3	3400 Hz
F4	3500 Hz
Atenuação máxima na banda de passagem	2 dB
Atenuação máxima na banda de rejeição	30 dB
Ganho	-10 dB

Tabela 7 – Especificação do filtro Passa Faixa

2.3.1 Filtro IIR do tipo Chebyshev II

Para esse filtro foi permitido o uso das funções do MATLAB no projeto, foi utilizada a função *cheb2ord* para estimar a ordem do filtro, com base nas especificações normalizadas. Em seguida, foi utilizada a função *cheby2* para calcular os coeficientes do filtro, utilizando a ordem gerada na função anterior e os valores da banda de passagem e das bandas rejeição que foram circunstanciadas para este filtro.

Após compilarmos o código de projeto do filtro, utilizando o software *Fdatool*, importamos o filtro do *workspace* do *MATLAB* com uso da estrutura de Forma Direta II de Seção de Segunda Ordem(*SOS*), então quantizamos o filtro com a menor quantidade de bits sem sair parâmetros e máscara dada no início do projeto. O resultado da quantização pode ser vista na imagem abaixo:

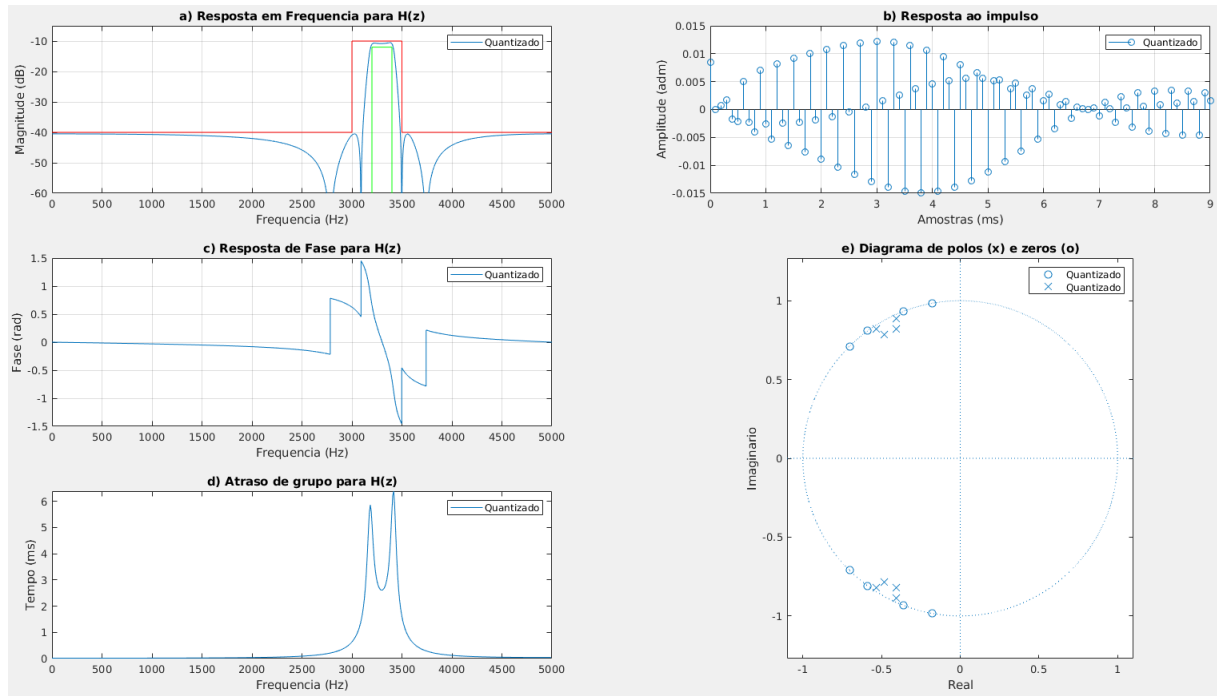


Figura 20 – Visualização geral do filtro quantizado

Pode-se observar, conforme exposto nos gráficos, os resultados obtidos pelo filtro passa faixa IIR com aproximação *Chebyshev II* quantizado a partir do filtro referência especificado. Consta na figura a resposta em frequência após a transformação bilinear, também a resposta ao impulso, resposta em fase, o atraso de grupo (nota-se que não é constante) e o diagrama de pólos e zeros.

Após o filtro quantizado no *Fdatool*, exportamos a estrutura para realizar o projeto no *Simulink*. Adicionamos alguns recursos para fazer a simulação, como sinais ruído de entrada, analisador de espectro e *Time Scope*. Os blocos do projeto podem ser visualizados na figura abaixo

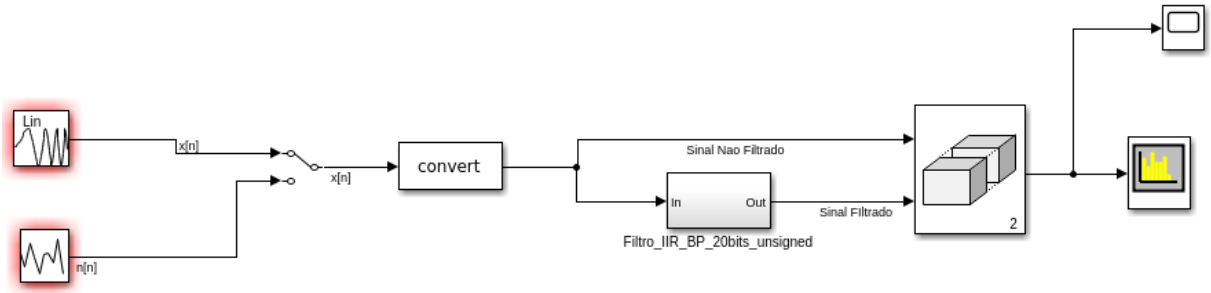


Figura 21 – Circuito de simulação do filtro Passa Faixa IIR no Simulink

Na figura a seguir é possível visualizar o resultado da simulação com o sinal de entrada *chirp* no analisador de espectro do software *Simulink*. A cor azul é do sinal filtrado pelo filtro quantizado e o sinal em vermelha é o sinal não filtrado.

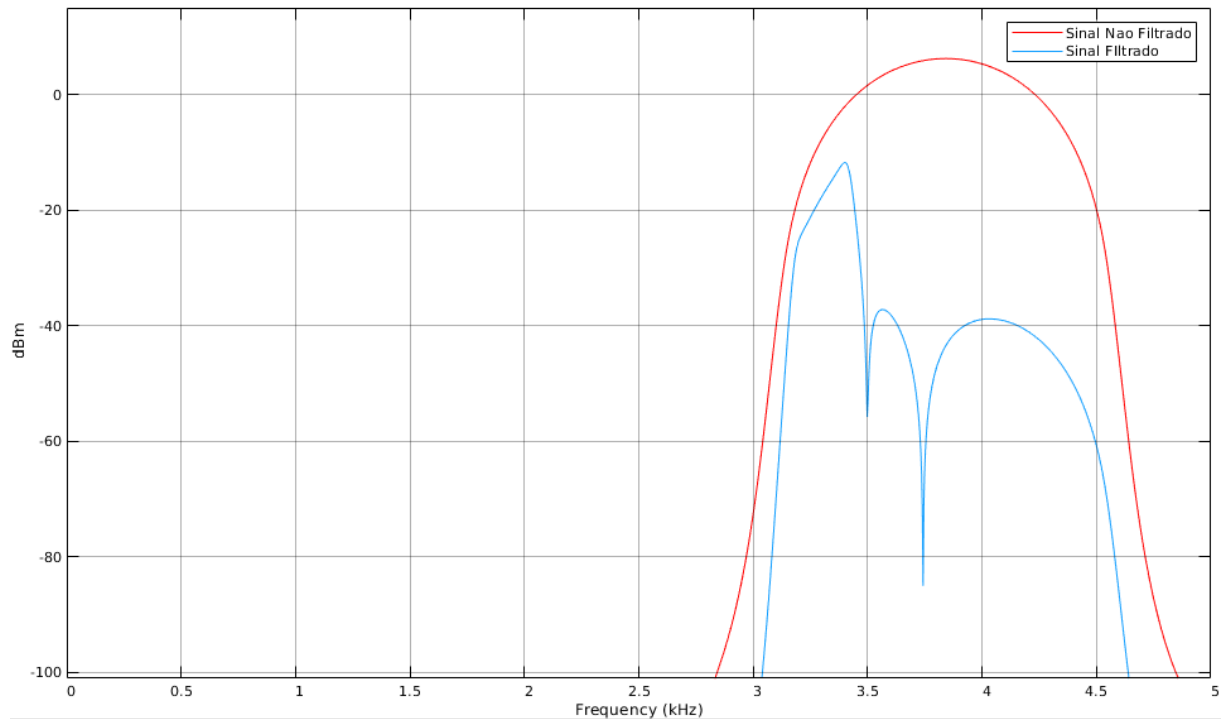


Figura 22 – Resultado da simulação filtro BP no analisador de espectro com chirp na entrada

O resultado obtido no analisador de tempo do *Simulink* da simulação do filtro Passa Faixa com a função de entrada *chirp*, pode-se ser verificada abaixo.

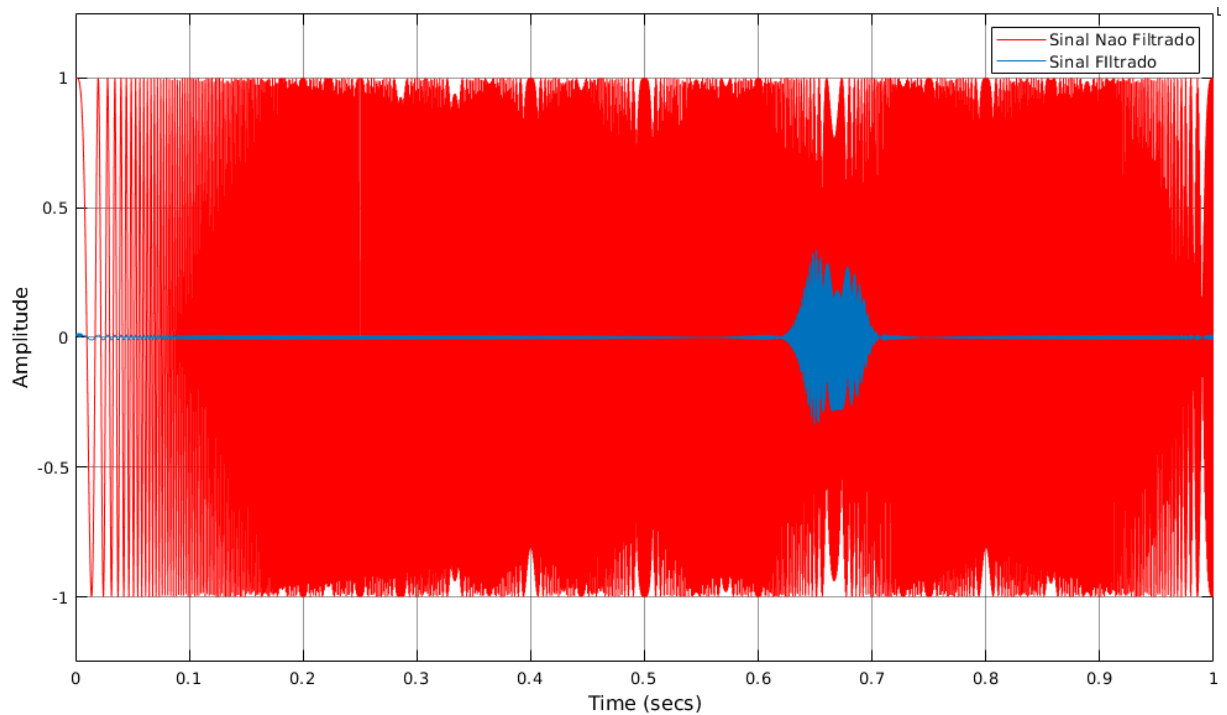


Figura 23 – Resultado da simulação filtro BP no analisador de tempo com chirp na entrada

Após a simulação, nota-se que somente a banda de passagem do filtro está presente na saída do *chirp* e atenuada, conforme foi solicitado na especificação do filtro. Aplicamos um ruído branco na entrada do filtro projetado e obtivemos o resultado do sinal no analisador de espectro e no *Time Scope* respectivamente. Pode-se observar no analisador de espectro o resultado do sinal original (não filtrado, cor vermelha) e o sinal filtrado pelo protótipo (cor azul), é notório que o sinal foi filtrado corretamente neste projeto também, ou seja, o sinal passou na faixa de frequência permitida e as demais frequências foram atenuadas de acordo com a atenuação especificada, sendo assim evidenciando o funcionamento do filtro projetado.

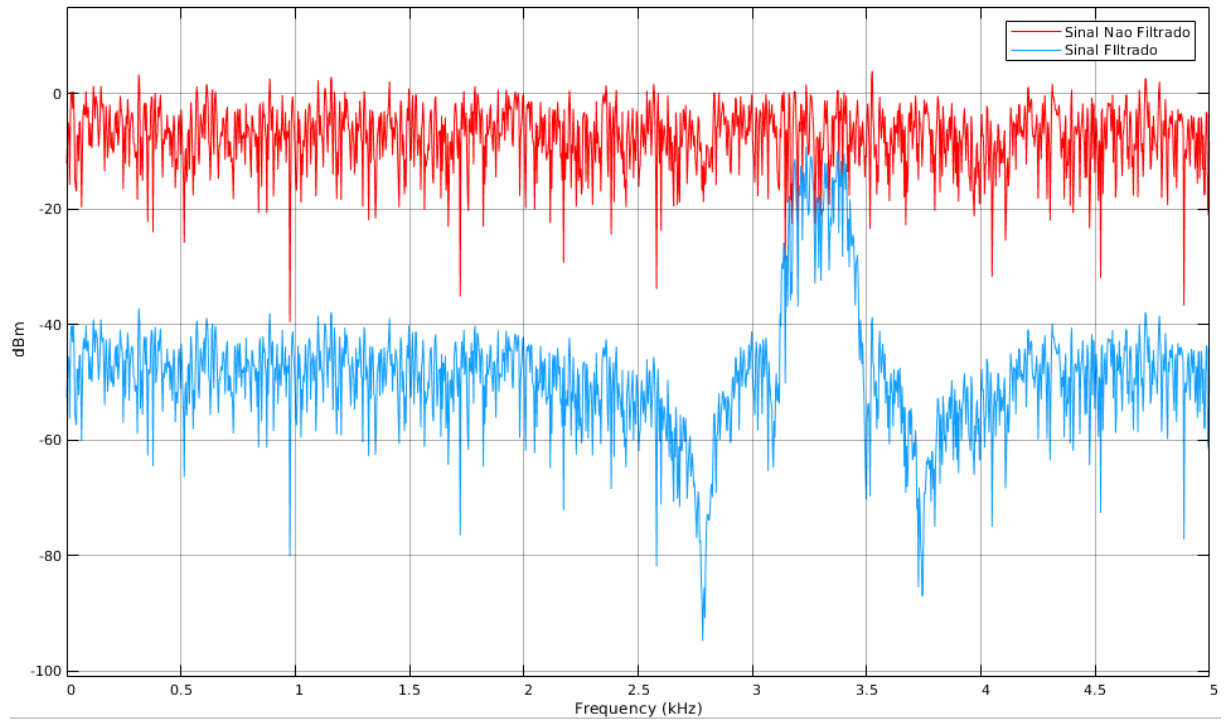


Figura 24 – Resultado da simulação filtro BP no analisador de espectro com ruído branco na entrada

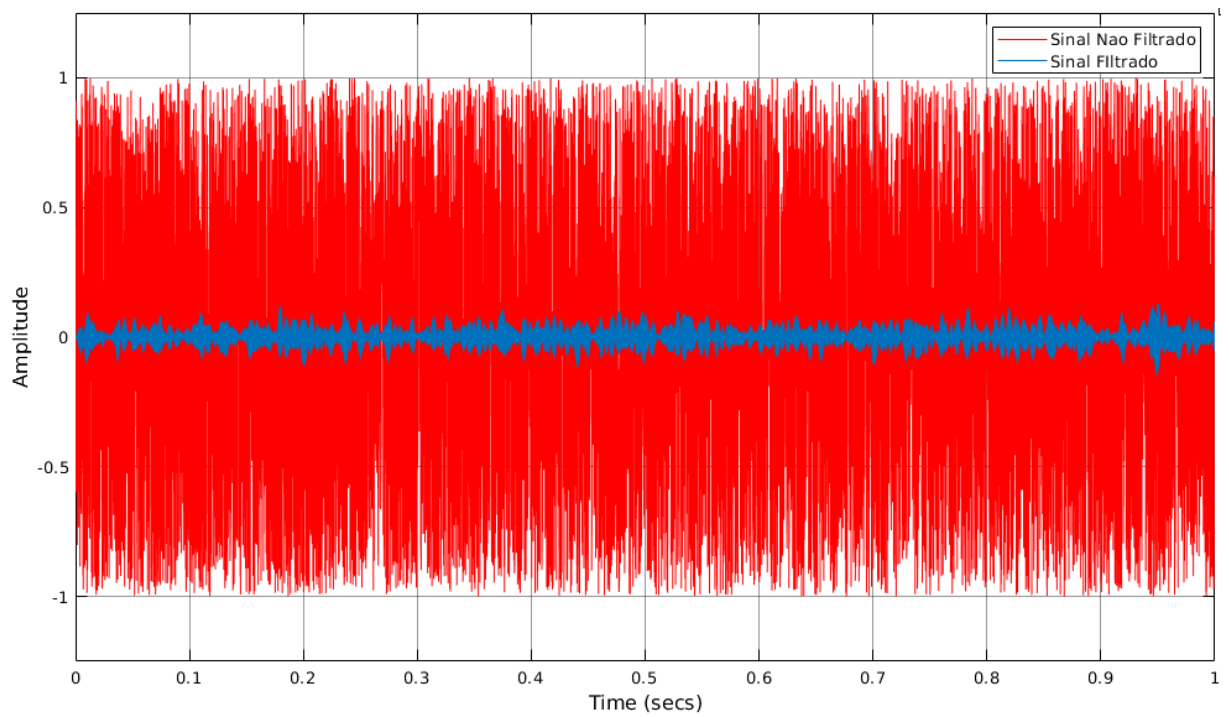


Figura 25 – Resultado da simulação filtro BP no analisador de tempo com ruído branco na entrada

2.3.1.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro IIR passa faixa, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposta uma comparação sobreposta na mesma figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

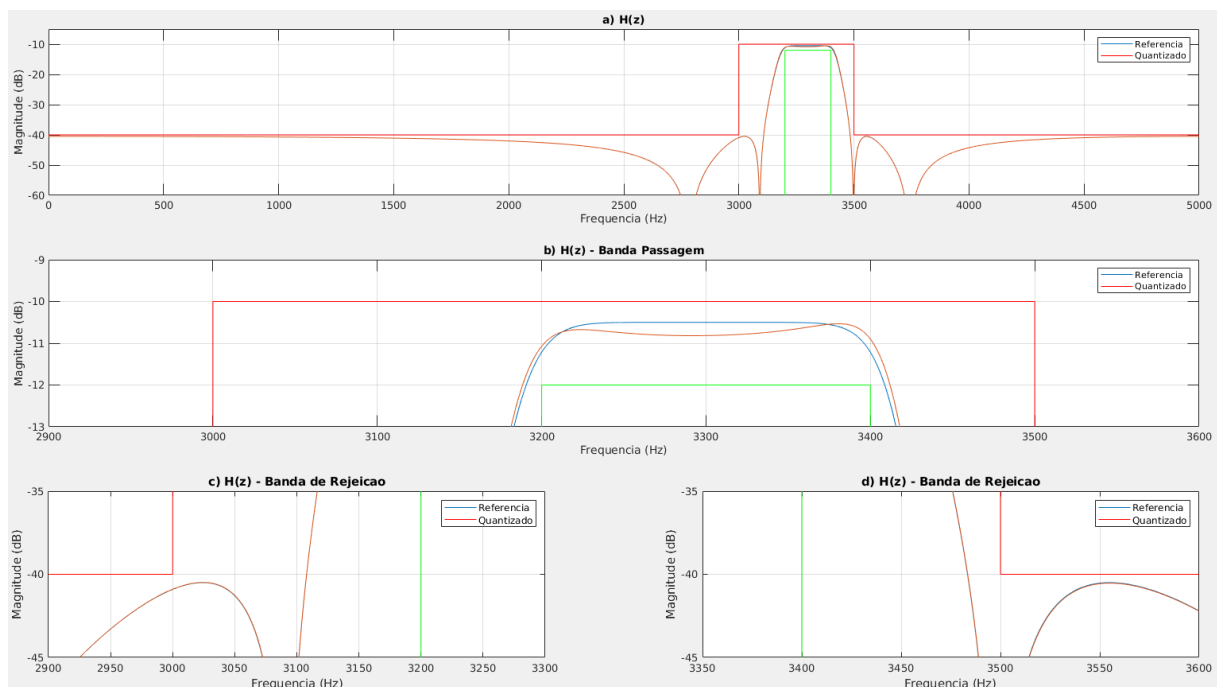


Figura 26 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que o resultado é praticamente o mesmo, mantendo a estabilidade do filtro.

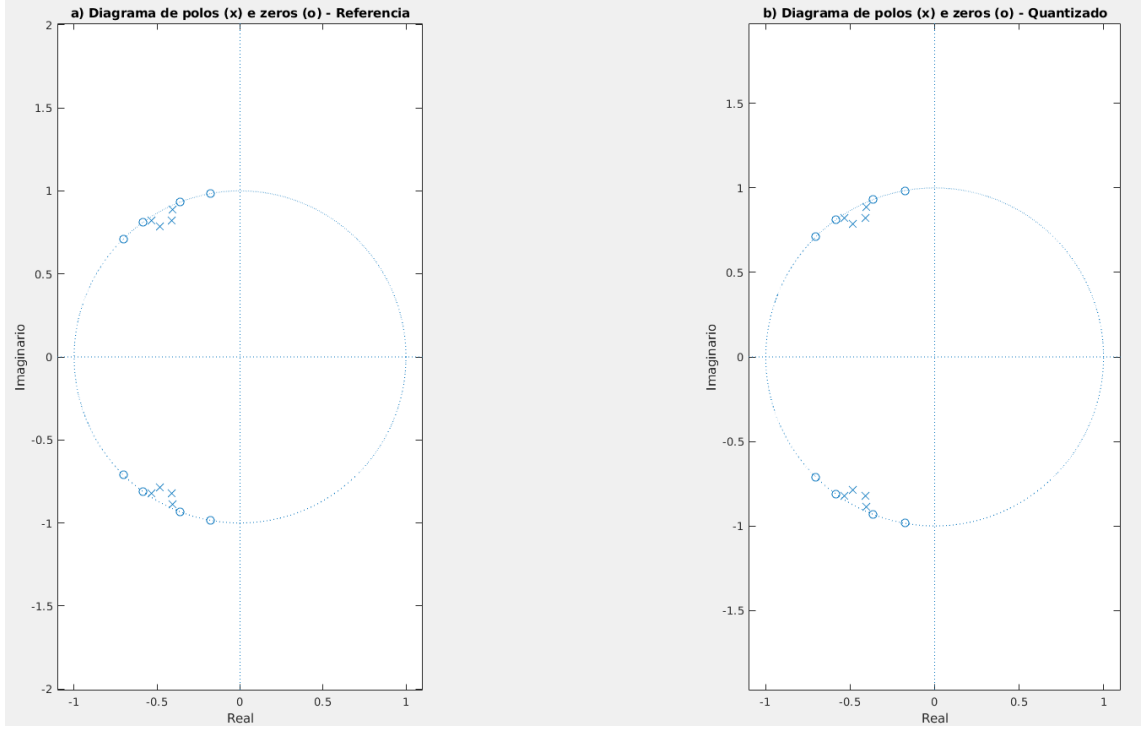


Figura 27 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados de elementos de área lógica do filtro referência e do filtro quantizado. Os dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*BP IIR Ref*) e o filtro quantizado (*BP IIR Quant*). Em ambos os filtros foram aplicados a estrutura Forma Direta II de Seção de Segunda Ordem(SOS) (*D.II SOS*).

Na tabela é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores, Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.16)$$

A estimativa de quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.17)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
BP IIR Ref	8	64	D.II SOS	17	16	8	3712	3200
BP IIR Quant	8	19	D.II SOS	17	16	8	1102	950

Tabela 8 – Comparativo dos resultados obtidos no filtro referência com o quantizado

Neste filtro, foi observado que todos os coeficientes eram todos positivos, possibilitando o uso da representação não sinalizada. Desta forma, como estávamos com 20 bits de quantização, o número de bits foi reduzido para 19, reduzindo ainda mais o tamanho do hardware.

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de referência (projeto).

$$Hq(z) = \frac{0.0090377 z^8 + 0.033078 z^7 + 0.078565 z^6 + 0.12132 z^5 + 0.14288 z^4 + 0.12132 z^3 + 0.078565 z^2 + 0.033078 z + 0.0090377}{z^8 + 3.6593 z^7 + 8.6068 z^6 + 12.918 z^5 + 14.566 z^4 + 11.632 z^3 + 6.9786 z^2 + 2.6712 z + 0.65736} \quad (2.18)$$

A equação a seguir expressa o filtro quantizado.

$$Hq(z) = \frac{0.008532 z^8 + 0.031228 z^7 + 0.07417 z^6 + 0.11454 z^5 + 0.13489 z^4 + 0.11454 z^3 + 0.07417 z^2 + 0.031228 z + 0.008532}{z^8 + 3.6593 z^7 + 8.6068 z^6 + 12.918 z^5 + 14.566 z^4 + 11.632 z^3 + 6.9786 z^2 + 2.6712 z + 0.65736} \quad (2.19)$$

2.3.2 Filtro FIR do tipo Parks-McClellan

Para o filtro FIR utilizou-se o algoritmo de *Parks-McClellan*. Com as funções prontas do *MATLAB* *firpmord* e *firpm*. Para o cálculo de ordem do filtro utiliza-se a função *firpmord* e para montagem do filtro com o algoritmo de *Parks-McClellan* foi utilizada a função *firpm*. Com o filtro implementado importamos ele para o software *Fdatool* para quantização do filtro FIR Passa Faixa utilizando a estrutura de Forma Direta Simétrica, simulamos a quantização diversas vezes para chegar a menor quantidade de bits (precisão finita) sem que o filtro saia das especificações iniciais. O resultado da melhor quantização dentro dos parâmetros especificados pode-se observar na figura abaixo:

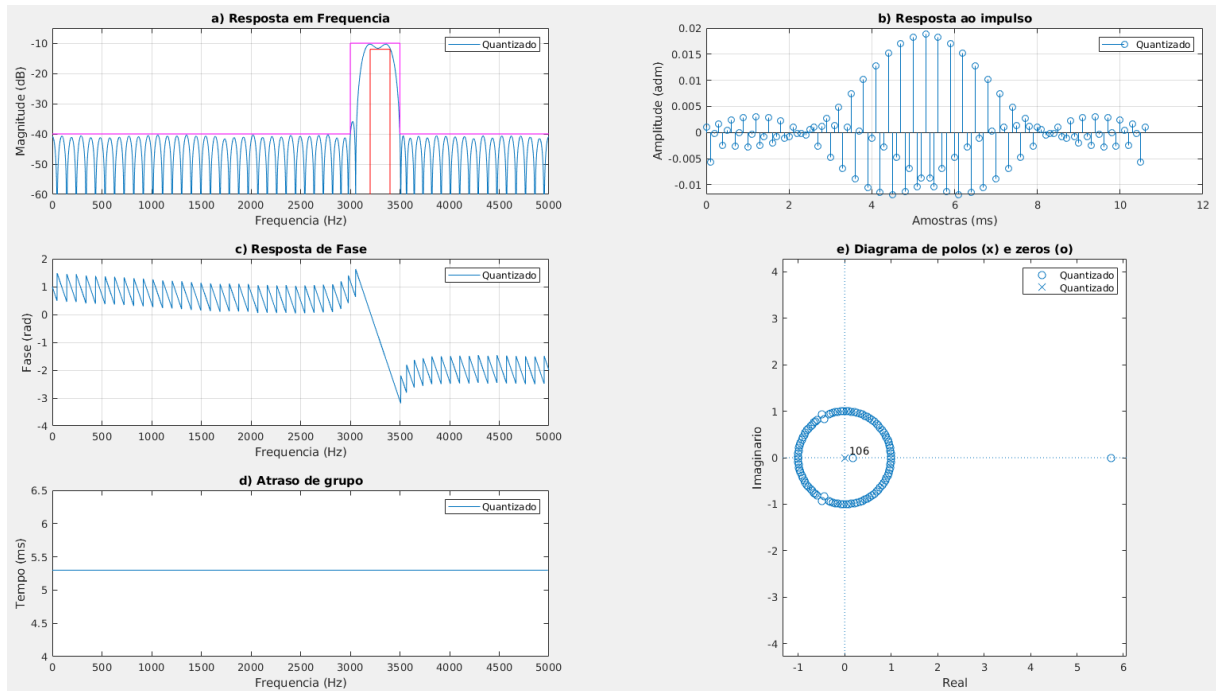


Figura 28 – Visualização geral do filtro quantizado

Pode-se observar, conforme exposto nos gráficos, os resultados obtidos pelo filtro passa faixa FIR com aproximação *Parks–McClellan* quantizado a partir do filtro referência especificado. Consta na figura a resposta em frequência após a transformação bilinear, também a resposta ao impulso, resposta em fase, o atraso de grupo (nota-se tem um atraso constante de 5,3ms) e o diagrama de pólos e zeros.

2.3.2.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro FIR passa faixa, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposta uma comparação sobreposta na mesma figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

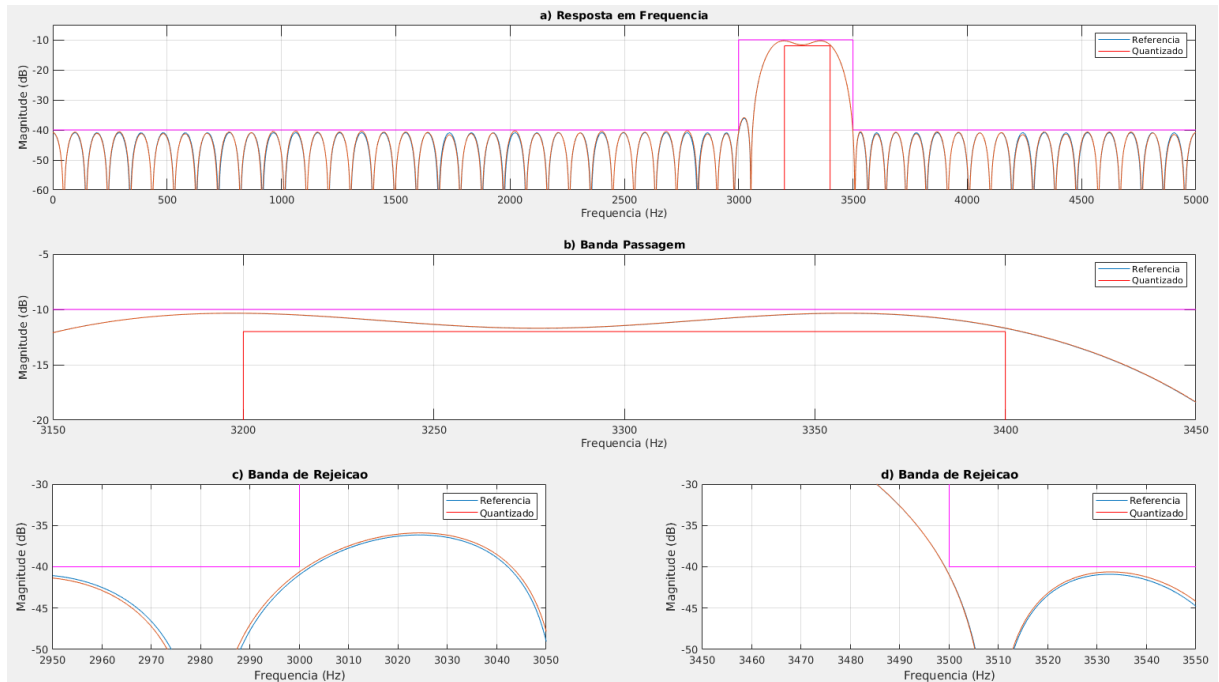


Figura 29 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que o resultado é praticamente o mesmo.

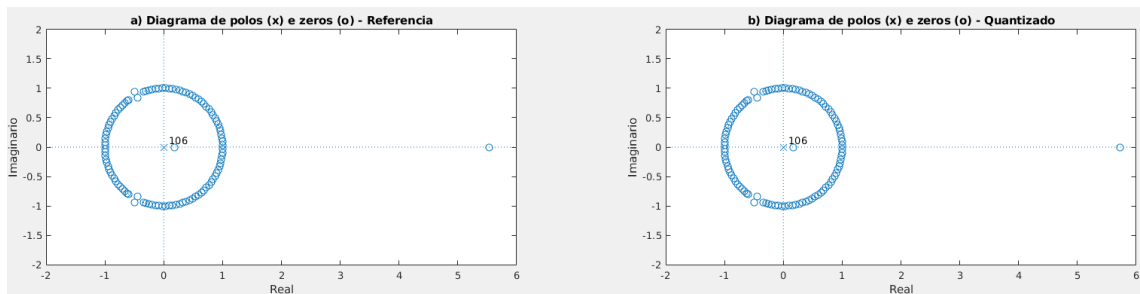


Figura 30 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados de elementos de área lógica do filtro referência e do filtro quantizado. O dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*BP FIR Ref*) e o filtro quantizado (*BP FIR Quant*). Em ambos os filtros foram aplicados a estrutura Forma Direta Simétrica (*D SYM*).

Na tabela é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores,

Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.20)$$

A estimativa de quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.21)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
BP FIR Ref	90	64	D SYM	46	91	90	17472	11712
BP FIR Quant	106	9	D SYM	54	107	106	2889	1935

Tabela 9 – Comparativo dos resultados obtidos no filtro referência com o quantizado

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de referência (projeto).

$$\begin{aligned}
H(z) = & 0.0043007 z^{90} + 0.0020701 z^{89} - 0.0035042 z^{88} + 0.003172 z^{87} - 0.0006195 z^{86} \\
& - 0.0019406 z^{85} + 0.0019498 z^{84} + 0.0005863 z^{83} - 0.0029539 z^{82} + 0.0026043 z^{81} \\
& + 0.00015039 z^{80} - 0.0025858 z^{79} + 0.0024621 z^{78} - 0.00011479 z^{77} - 0.0022371 z^{76} \\
& + 0.0027099 z^{75} - 0.001087 z^{74} - 0.0014505 z^{73} + 0.0031701 z^{72} - 0.0025623 z^{71} \\
& - 0.00058881 z^{70} + 0.004176 z^{69} - 0.0045938 z^{68} + 0.00016955 z^{67} + 0.0058106 z^{66} \\
& - 0.0069711 z^{65} + 0.0005282 z^{64} + 0.008102 z^{63} - 0.0094284 z^{62} + 0.00027034 z^{61} \\
& + 0.010923 z^{60} - 0.011622 z^{59} - 0.00075105 z^{58} + 0.014015 z^{57} - 0.0132 z^{56} \\
& - 0.0025191 z^{55} + 0.017025 z^{54} - 0.013893 z^{53} - 0.0048849 z^{52} + 0.01955 z^{51} \\
& \dots \\
& + 0.00015039 z^{10} + 0.0026043 z^9 - 0.0029539 z^8 + 0.0005863 z^7 + 0.0019498 z^6 \\
& - 0.0019406 z^5 - 0.0006195 z^4 + 0.003172 z^3 - 0.0035042 z^2 + 0.0020701 z + 0.0043007
\end{aligned} \quad (2.22)$$

A equação a seguir expressa o filtro quantizado.

$$\begin{aligned}
Hq(z) = & 0.00097656 z^{106} - 0.0056152 z^{105} - 0.00012207 z^{104} + 0.001709 z^{103} \\
& - 0.0024414 z^{102} + 0.00036621 z^{101} + 0.0024414 z^{100} - 0.0026855 z^{99} + 0.0028076 z^{97} \\
& - 0.0028076 z^{96} - 0.00036621 z^{95} + 0.0030518 z^{94} - 0.0024414 z^{93} - 0.00073242 z^{92} \\
& + 0.0029297 z^{91} - 0.0019531 z^{90} - 0.00085449 z^{89} + 0.0023193 z^{88} - 0.0010986 z^{87} \\
& - 0.00073242 z^{86} + 0.0010986 z^{85} - 0.00024414 z^{84} - 0.00012207 z^{83} - 0.00048828 z^{82} \\
& + 0.00061035 z^{81} + 0.00097656 z^{80} - 0.0025635 z^{79} + 0.0012207 z^{78} + 0.0026855 z^{77} \\
& - 0.0047607 z^{76} + 0.0013428 z^{75} + 0.0048828 z^{74} - 0.006958 z^{73} + 0.0010986 z^{72} \\
& + 0.0074463 z^{71} - 0.0089111 z^{70} + 0.00024414 z^{69} + 0.010132 z^{68} - 0.010498 z^{67} \\
& - 0.0010986 z^{66} + 0.012817 z^{65} - 0.011475 z^{64} - 0.0028076 z^{63} + 0.015259 z^{62} \\
& - 0.011841 z^{61} - 0.0047607 z^{60} + 0.01709 z^{59} - 0.011353 z^{58} - 0.0068359 z^{57} + 0.018311 z^{56} \\
& - 0.010376 z^{55} - 0.0087891 z^{54} + 0.018799 z^{53} - 0.0087891 z^{52} - 0.010376 z^{51} + 0.018311 z^{50} \\
& - 0.0068359 z^{49} - 0.011353 z^{48} + 0.01709 z^{47} - 0.0047607 z^{46} - 0.011841 z^{45} + 0.015259 z^{44} \\
& - 0.0028076 z^{43} - 0.011475 z^{42} + 0.012817 z^{41} - 0.0010986 z^{40} - 0.010498 z^{39} + 0.010132 z^{38} \\
& + 0.00024414 z^{37} - 0.0089111 z^{36} + 0.0074463 z^{35} + 0.0010986 z^{34} - 0.006958 z^{33} \\
& + 0.0048828 z^{32} + 0.0013428 z^{31} - 0.0047607 z^{30} + 0.0026855 z^{29} + 0.0012207 z^{28} \\
& - 0.0025635 z^{27} + 0.00097656 z^{26} + 0.00061035 z^{25} - 0.00048828 z^{24} - 0.00012207 z^{23} \\
& - 0.00024414 z^{22} + 0.0010986 z^{21} - 0.00073242 z^{20} - 0.0010986 z^{19} + 0.0023193 z^{18} \\
& - 0.00085449 z^{17} - 0.0019531 z^{16} + 0.0029297 z^{15} - 0.00073242 z^{14} - 0.0024414 z^{13} \\
& + 0.0030518 z^{12} - 0.00036621 z^{11} - 0.0028076 z^{10} + 0.0028076 z^9 - 0.0026855 z^7 \\
& + 0.0024414 z^6 + 0.00036621 z^5 - 0.0024414 z^4 + 0.001709 z^3 - 0.00012207 z^2 \\
& - 0.0056152 z + 0.00097656
\end{aligned} \tag{2.23}$$

Este filtro foi reprojeto para melhor adequação à máscara e possibilitar uma redução no tamanho do *hardware*. Por este motivo, sua ordem aumentou e, em contrapartida, apesar deste aumento, o tamanho do hardware final estimado diminuiu.

2.4 Filtro Rejeita-Faixa

No último projeto quantizado será utilizamos o filtro rejeita-faixa IIR com aproximação do tipo *Chebyshev I*, em seguida um FIR com algoritmo de *Parks-McClellan*, com as especificações da tabela a seguir.

Frequência de amostragem	8000 Hz
F1	1200 Hz
F2	1250 Hz
F3	1300 Hz
F4	1400 Hz
Atenuação máxima na banda de passagem	0.5 dB
Atenuação máxima na banda de rejeição	60 dB
Ganho	0 dB

Tabela 10 – Especificação do filtro Rejeita Faixa

2.4.1 Filtro IIR do tipo Chebyshev I

Foi utilizada a função do *MATLAB* *cheb1ord* para estimar a ordem do protótipo *Chebyshev tipo 1*, com base nas especificações normalizadas. Em seguida, a função *cheby1* foi utilizada para calcular os coeficientes do filtro, tendo como base a ordem gerada na função anterior e os valores de banda de passagem e rejeição especificados.

Após a compilação do projeto do filtro, importamos os coeficientes para a ferramenta *Fdatool* com uso da estrutura de Forma Direta II de Seção de Segunda Ordem (*SOS*), e então quantizamos o filtro com uma precisão finita com o intuito de encontrar a menor quantidade de bits sem desconsiderar os parâmetros e máscara dada no início do projeto. O resultado da quantização pode ser vista na imagem abaixo:

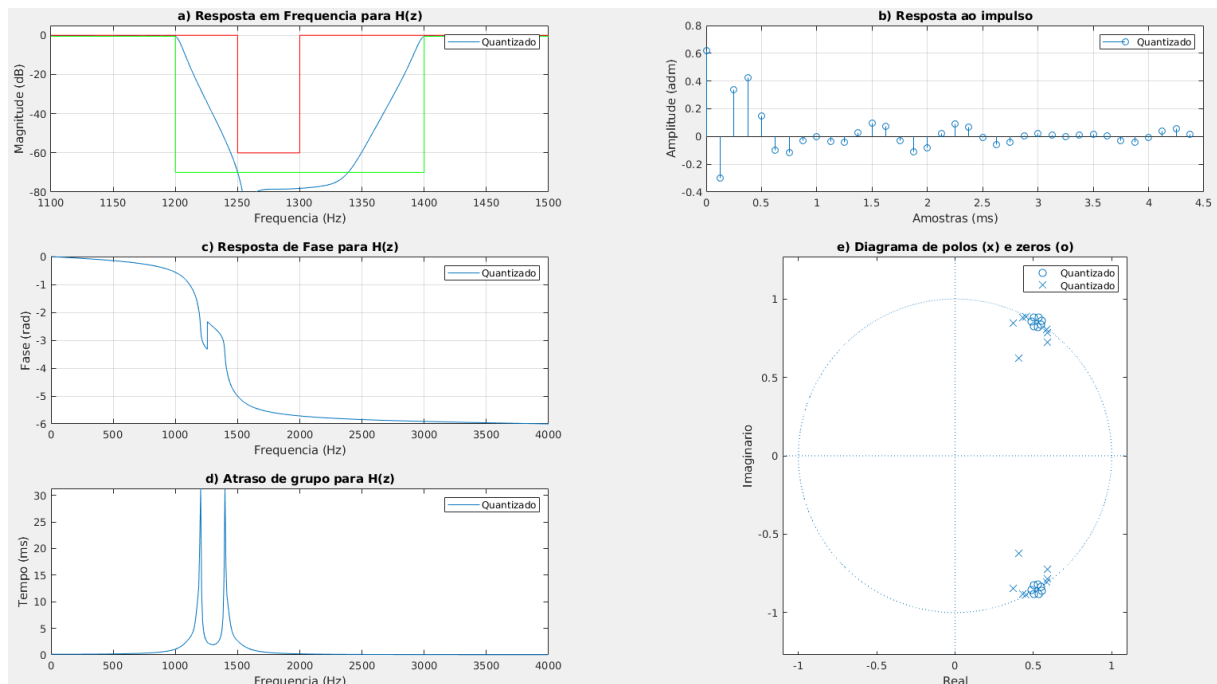


Figura 31 – Visualização geral do filtro quantizado

Pode-se observar, conforme exposto nos gráficos, os resultados obtidos pelo filtro rejeita faixa IIR com aproximação *Chebyshev tipo 1* quantizado a partir do filtro referência especificado. Consta na figura a resposta em frequência após a transformação bilinear, também a resposta ao impulso, resposta em fase, o atraso de grupo (nota-se que não é constante) e o diagrama de pólos e zeros.

2.4.1.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro IIR rejeita faixa, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposto uma comparação sobreposta na mesma figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

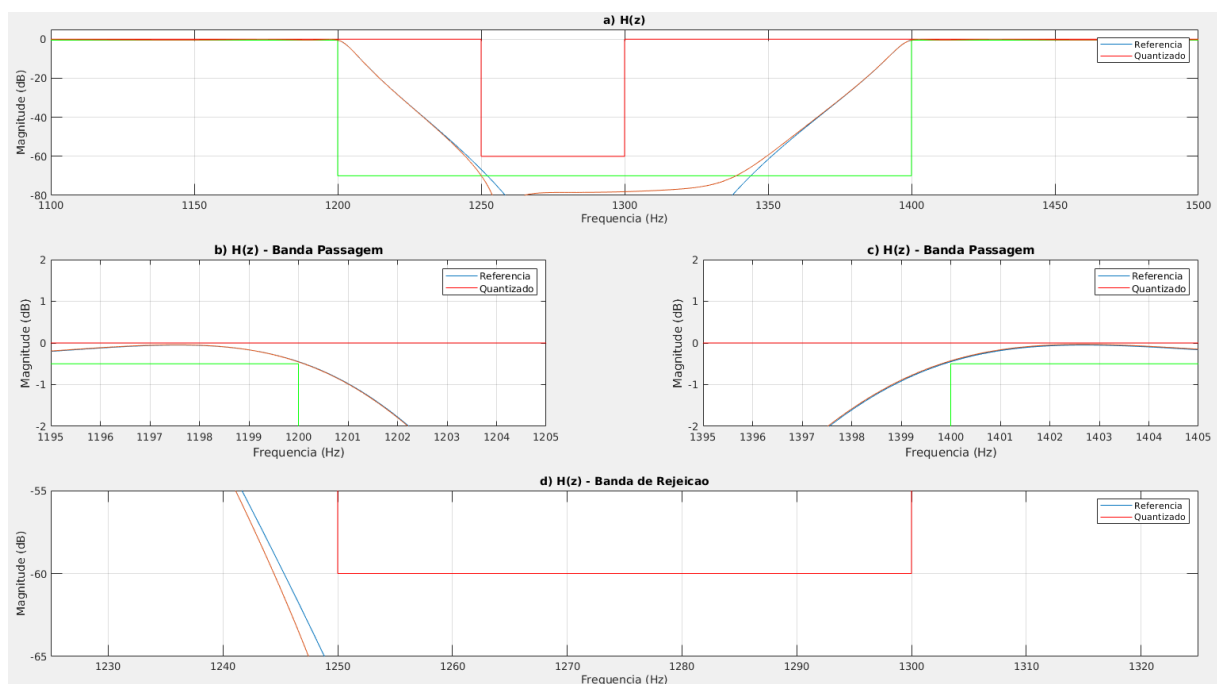


Figura 32 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que há diferença neste diagrama, todavia, a estabilidade do filtro é mantida.

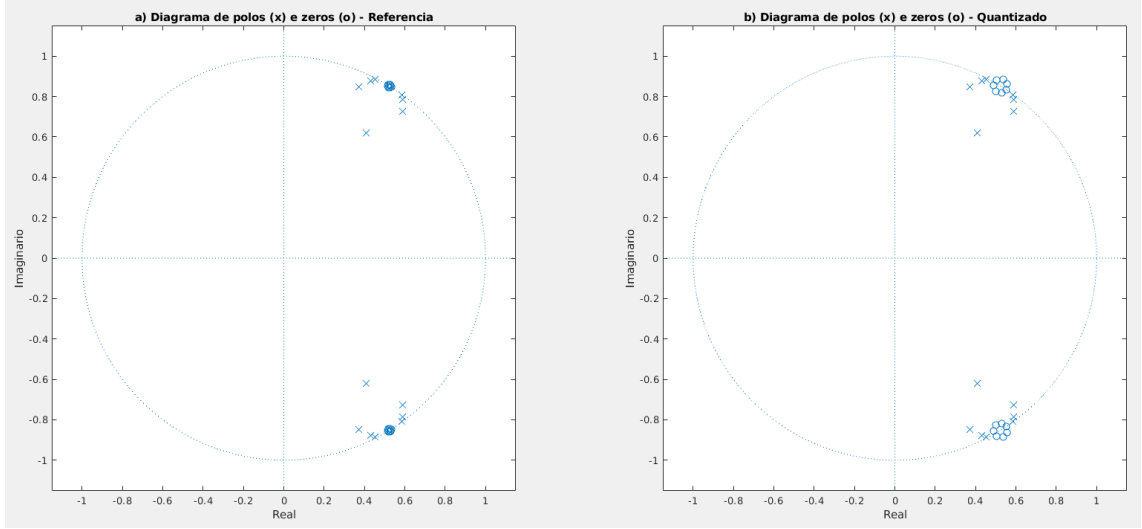


Figura 33 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados de elementos de área lógica do filtro referência e do filtro quantizado. Os dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*BS IIR Ref*) e o filtro quantizado (*BS IIR Quant*). Em ambos os filtros foram aplicados a estrutura Forma Direta II de Seção de Segunda Ordem(SOS) (*D.II SOS*).

Na tabela é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores, Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.24)$$

A estimativa de quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.25)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
BS IIR Ref	14	64	D.II SOS	29	28	14	6400	5504
BS IIR Quant	14	40	D.II SOS	29	28	14	4000	3440

Tabela 11 – Comparativo dos resultados obtidos no filtro referência com o quantizado

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de referência (projeto).

$$H(z) = \frac{0.61703 z^{14} - 4.5275 z^{13} + 18.557 z^{12} - 52.039 z^{11} + 110.22 z^{10} - 183.81 z^9 + 247.92 z^8 - 273.45 z^7 + 247.92 z^6 - 183.81 z^5 + 110.22 z^4 - 52.039 z^3 + 18.557 z^2 - 4.5275 z + 0.61703}{z^{14} - 6.8417 z^{13} + 26.125 z^{12} - 68.263 z^{11} + 134.72 z^{10} - 209.37 z^9 + 263.16 z^8 - 270.46 z^7 + 228.44 z^6 - 157.72 z^5 + 88.021 z^4 - 38.646 z^3 + 12.8 z^2 - 2.8964 z + 0.36518} \quad (2.26)$$

A equação a seguir expressa o filtro quantizado.

$$Hq(z) = \frac{0.61924 z^{14} - 4.5437 z^{13} + 18.623 z^{12} - 52.225 z^{11} + 110.61 z^{10} - 184.47 z^9 + 248.81 z^8 - 274.43 z^7 + 248.81 z^6 - 184.47 z^5 + 110.61 z^4 - 52.225 z^3 + 18.623 z^2 - 4.5437 z + 0.61924}{z^{14} - 6.8495 z^{13} + 26.186 z^{12} - 68.511 z^{11} + 135.39 z^{10} - 210.71 z^9 + 265.25 z^8 - 273.06 z^7 + 231.04 z^6 - 159.83 z^5 + 89.393 z^4 - 39.345 z^3 + 13.068 z^2 - 2.9663 z + 0.37537} \quad (2.27)$$

2.4.2 Filtro FIR do tipo Parks-McClellan

Para o modelo FIR utilizou-se o algoritmo de *Parks-McClellan*. Com as funções prontas do *MATLAB* *firpmord* e *firpm*. Para o cálculo de ordem do filtro utilizou-se a função *firpmord* e para montagem do filtro com o algoritmo de *Parks-McClellan* foi utilizada a função *firpm*. Com o filtro projetado, exportamos os coeficientes para a ferramenta *Fdatool* para que possamos quantiza-lo. Utilizando a estrutura de Forma Direta Simétrica fomos mudando a quantidade de bit a procura da melhor aproximação sem sair dos parâmetros especificados. Ou seja, sem mudar as especificações iniciais do projeto e sem sair da máscara aplicada. Pode-se observar na figura a seguir o resultado do filtro quantizado:

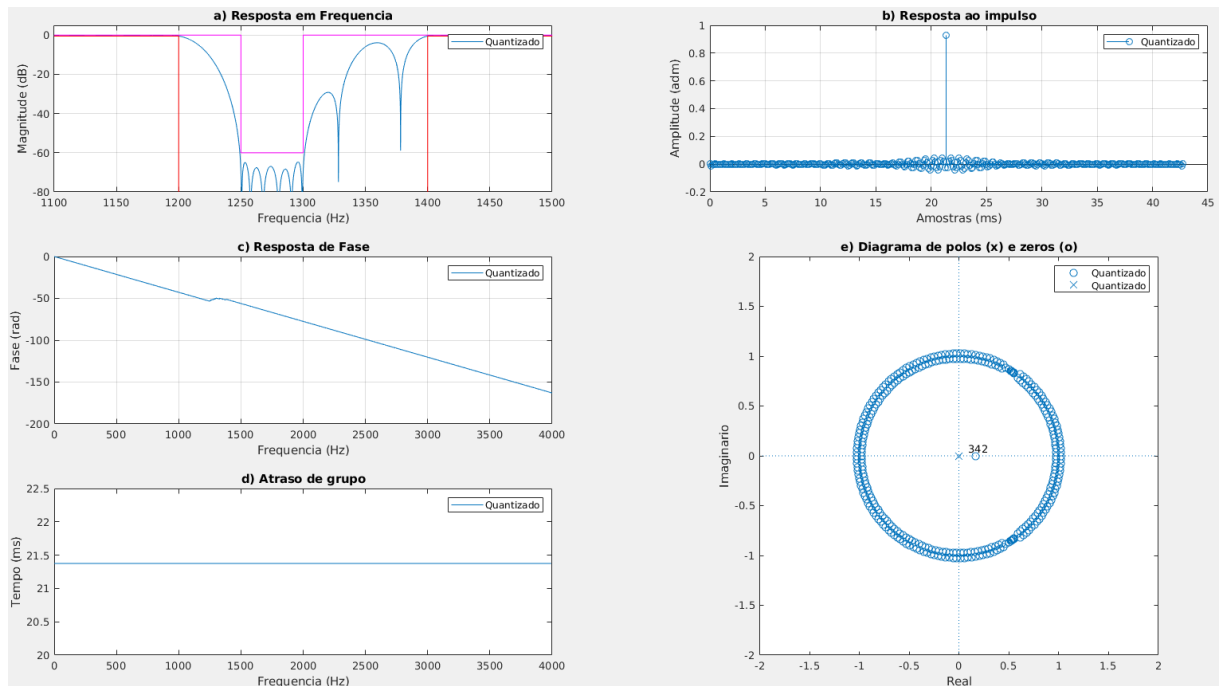


Figura 34 – Visualização geral do filtro quantizado

Pode-se observar, conforme exposto nos gráficos, os resultados obtidos pelo filtro passa faixa FIR com aproximação *Parks–McClellan* quantizado a partir do filtro referência especificado. Consta na figura a resposta em frequência após a transformação bilinear, também a resposta ao impulso, resposta em fase, o atraso de grupo (nota-se tem um atraso constante de 21,4ms) e o diagrama de pólos e zeros.

2.4.2.1 Comparativo entre o filtro referência e o quantizado

Nesta subseção do filtro FIR passa faixa, serão comparados os resultados obtidos na primeira avaliação da disciplina (filtro referência) com o resultado da quantização do filtro. Primeiramente, na figura abaixo é exposto uma comparação sobreposta na mesma figura entre as respostas em frequências do filtro referência (cor azul) e do filtro quantizado (cor vermelha).

Observa-se que foi levado em conta especificações e as máscaras após a quantização do filtro. Ou seja, o filtro quantizado satisfaz as condições iniciais e a máscara proposta. A primeira imagem da figura abaixo é a comparação da resposta em frequência sem zoom, a segunda figura é um zoom na banda de passagem, e a terceira é um zoom na banda de rejeição. O zoom foi realizado para melhor visualização em suas bandas.

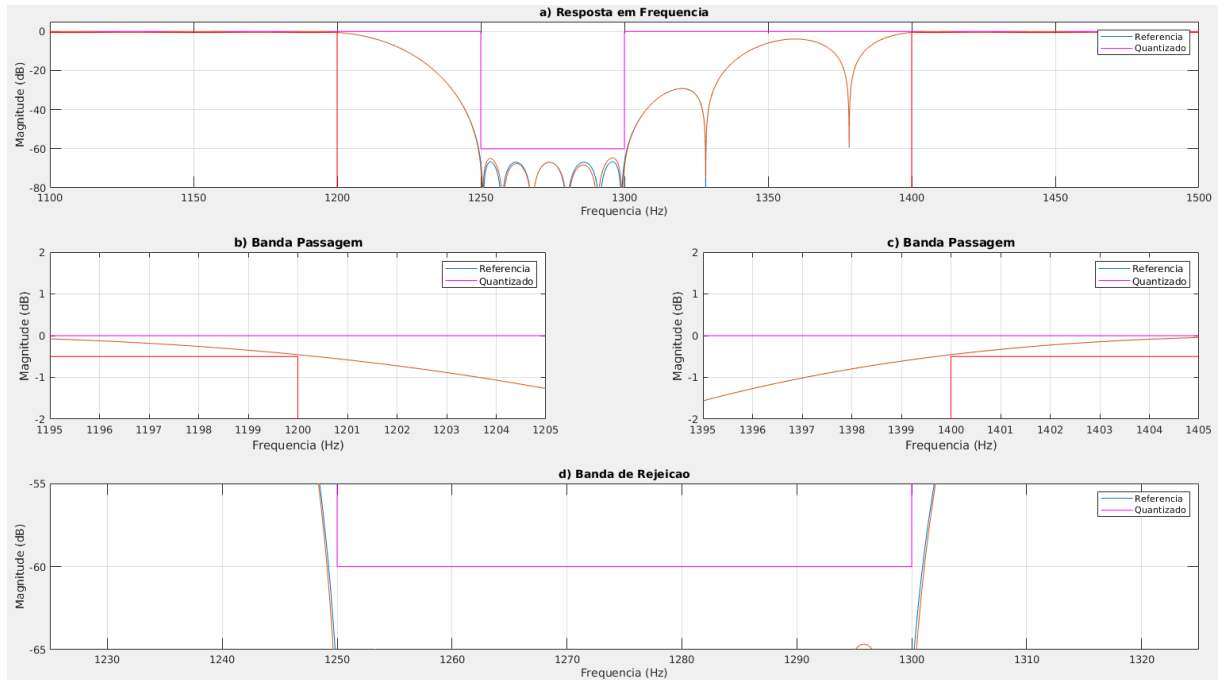


Figura 35 – Resposta em frequência do filtro referência e filtro quantizado sobrepostas

A próxima figura, expõe o diagrama de pólos e zeros do filtro referência e do diagrama após a quantização. Nota-se que o resultado é praticamente o mesmo.

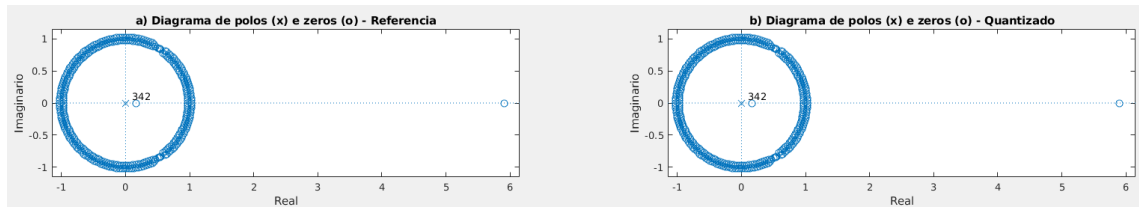


Figura 36 – Comparativo entre os diagramas de pólos e zeros do filtro referência(a) com o filtro quantizado(b)

Por fim, com o uso da ferramenta *Fdatool* recolhemos os dados de elementos de área lógica do filtro referência e do filtro quantizado. Os dados podem ser observados na tabela a seguir, onde foi realizado um comparativo entre o filtro referência (*BS FIR Ref*) e o filtro quantizado (*BS FIR Quant*). Em ambos os filtros foram aplicadas a estrutura Forma Direta Simétrica (*D SYM*).

Na tabela é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores, Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.28)$$

A estimativa de quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.29)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA
BS FIR Ref	342	64	D SYM	172	343	342	65856	43968
BS FIR Quant	342	16	D SYM	172	343	342	16464	10992

Tabela 12 – Comparativo dos resultados obtidos no filtro referência com o quantizado

Por fim, podemos visualizar a diferença entre as equações do filtro, relacionando os coeficientes antes e depois da quantização. A equação a seguir expressa o filtro de referência (projeto).

$$\begin{aligned}
H(z) = & 0.0022206 z^{342} - 0.012654 z^{341} - 0.0028579 z^{340} - 0.000062787 z^{339} + 0.0011442 z^{338} \\
& + 0.00089626 z^{337} - 0.00033143 z^{336} - 0.0012263 z^{335} - 0.00084214 z^{334} \\
& \dots \\
& + 0.00041239 z^9 - 0.00084214 z^8 - 0.0012263 z^7 - 0.00033143 z^6 + 0.00089626 z^5 \\
& + 0.0011442 z^4 - 0.000062787 z^3 - 0.0028579 z^2 - 0.012654 z + 0.0022206
\end{aligned} \quad (2.30)$$

A equação a seguir expressa o filtro quantizado.

$$\begin{aligned}
Hq(z) = & 0.0022278 z^{342} - 0.012665 z^{341} - 0.0028687 z^{340} - 0.000061035 z^{339} \\
& + 0.0011292 z^{338} + 0.00088501 z^{337} - 0.00033569 z^{336} - 0.0012207 z^{335} - 0.00085449 z^{334} \\
& + 0.00042725 z^{333} + 0.0012207 z^{332} + 0.0007019 z^{331} - 0.0005188 z^{330} - 0.0011597 z^{329} \\
& - 0.00048828 z^{328} + 0.0007019 z^{327} + 0.0010681 z^{326} + 0.00021362 z^{325} - 0.00091553 z^{324} \\
& \dots \\
& - 0.00085449 z^8 - 0.0012207 z^7 - 0.00033569 z^6 + 0.00088501 z^5 + 0.0011292 z^4 \\
& - 0.000061035 z^3 - 0.0028687 z^2 - 0.012665 z + 0.0022278
\end{aligned} \quad (2.31)$$

2.5 Comparativo entre os filtros IIR e FIR quantizados

Na tabela a seguir é demonstrada a ordem do filtro, a quantidade mínima de bits possível sem alterar as especificações e também a estrutura utilizada na quantização. Além disso, a quantidade de elementos lógicos resultantes são expostos nas colunas Multiplicadores, Somadores, e atrasos (Delays). Por fim, é apresentada uma estimativa de quantidade de elementos lógicos para os *hardwares* ASIC e FPGA.

Para estimar as unidades de elementos ASIC, foi aplicada a equação a seguir:

$$Asic = (2M + S + D) * N^{\circ}Bits \quad (2.32)$$

A estimativa de quantidade de elementos para FPGA foi obtida a partir da equação abaixo:

$$FPGA = (2M + S) * N^{\circ}Bits \quad (2.33)$$

Filtro	Ordem	NºBits	Estrutura	Multiplicadores	Somadores	Delays	ASIC	FPGA	Atraso de grupo
LP IIR	5	13	D.II	11	10	5	481	416	Não Constante
LP FIR	61	13	D SYM	31	61	62	2405	1599	3,1ms
HP IIR	13	12	D.II SOS	25	24	13	1044	888	Não Constante
HP FIR	42	8	D SYM	22	43	42	1032	696	2,1ms
BP IIR	8	20	D.II SOS	17	16	8	1160	1000	Não Constante
BP FIR	106	9	D SYM	54	107	106	2889	1935	5,3ms
BS IIR	14	40	D.II SOS	29	28	14	4000	3440	Não Constante
BS FIR	342	16	D SYM	172	343	342	16464	10992	21,4ms

Tabela 13 – Comparativo dos resultados obtidos nos filtros quantizados

Podemos observar que, em geral, os filtros IIR obtiveram um número de componentes inferior aos filtros FIR e o seu hardware é significativamente menor. Porém observamos que, proporcionalmente, os filtros IIR dependem de uma quantidade de bits superior para que tenham suas especificações atendidas.

Outra observação importante está no segundo filtro projetado, o filtro passa-altas (HP). Neste caso, vemos que o tamanho estimado do *hardware* final é menor para as duas implementações (ASIC e FPGA) para o filtro FIR. Neste projeto, utilizamos uma janela ajustável (Kaiser) para o filtro FIR, em que foi possível reduzir bastante a ordem e, em contrapartida, utilizamos uma janela para o filtro IIR que, em geral, fornece ordens altas (Butterworth). Por conta disso, foi possível projetarmos um filtro FIR com *hardware* menor que um IIR, mesmo tendo a ordem bem maior. Com isso, além de termos um projeto menor, ganhamos ainda as vantagens inerentes aos filtros FIR: fase linear e atraso de grupo constante.

Conclusão

Este trabalho teve como base o processo de realização de filtros digitais, ou seja, sua implementação final, utilizando elementos de *hardware*. Para este processo, estudamos estruturas de implementação e pudemos aplicá-las aos filtros desenvolvidos. Com isso, observamos como elas são implementadas e analisamos àquelas que ofereciam melhor desempenho.

Para utilização das estruturas estudadas, há a necessidade de quantizarmos os filtros projetados, e desta forma, foi possível constatar a importância do processo de quantização no projeto de filtros digitais. Este processo implica diretamente no tamanho do *hardware* final, fazendo com que um filtro possa até mesmo ser inviabilizado, caso tenha sido mal projetado. Além disso, observamos que, em geral, deve haver um equilíbrio entre a quantização e a ordem do filtro, já que ambos os critérios estão diretamente ligados ao circuito final do projeto. Este equilíbrio deve ocorrer para que tenhamos o projeto ideal, com o menor *hardware* possível.

Sendo assim, foi possível aplicarmos conhecimentos adquiridos ao longo do curso, projetando filtros com base não só numa especificação funcional, mas sim, levando em conta o tamanho físico do circuito, de modo a realizar projetos com o menor *hardware* possível, otimizando a utilização de recursos.

Referências

- [1] SENHOI, B .A.; Introduction to Digital Signal Processing and Filter Design. Índia: Wiley India Pvt. Ltd., 2010
- [2] DINIZ, P. S. R., DA SILVA, E. A. B., e LIMA NETTO, S. Processamento Digital de Sinais: Projeto e Análise de Sistemas. 2. ed. Porto Alegre: Bookman, 2014. 976 p. ISBN 978-8582601235.