

ANALISANDO UM CIRCUITO ELETRÔNICO LINEAR EM TEMPO CONTÍNUO E EM TEMPO DISCRETO

Renan S. Duque¹, Fábio A. R. Nascimento², Wilton N. A. Pereira³, Maurício Silveira⁴, Sandro A. Fasolo⁵ e Francisco J. Fraga⁶

Resumo — Este artigo consiste na análise do comportamento de um filtro digital FIR, implementado em hardware e simulado em software, associando-o aos conceitos de sinais e sistemas. A simulação foi feita utilizando o software Mathcad 2001 Professional e a implementação em hardware foi feita através do kit da família Motorola DSP56000.

Abstract — This paper treats the analysis of the behavior of a digital FIR filter, implemented in hardware and simulated in software, associating it to the signals and systems concepts. The simulation was implemented using Mathcad 2001 Professional software and the hardware was implemented using the Motorola DSP56000 family kit.

Index terms — Kit DSP, Causalidade, Motorola DSP56000, Sinais e Sistemas.

INTRODUÇÃO

O circuito a ser implementado em hardware e simulado em software é um filtro digital passa-faixa tipo FIR (Finite Impulse Response), excitado por um sinal digital obtido através da amostragem, na frequência $f_s = 9600$ (Hz), de um outro sinal composto por três senoides contínuas, cada uma com sua frequência particular, assim estipuladas: $f_1 = 223$ (Hz), $f_2 = 1031$ (Hz) e $f_3 = 3419$ (Hz). As Figuras 1 e 2 a seguir ilustram o diagrama simplificado do projeto e a resposta desejada para o filtro passa-faixa, respectivamente.

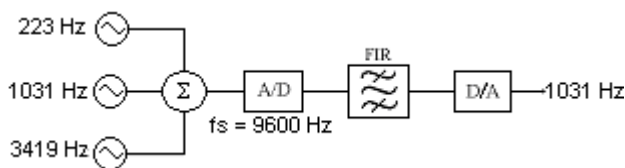


FIGURA. 1
DIAGRAMA EM BLOCOS DO PROJETO



FIGURA. 2
ESPECIFICAÇÕES DO FILTRO PASSA-FAIXA DIGITAL FIR

Utilizando os conceitos de Sinais e Sistemas e contando com um software de apoio em cálculo numérico, podemos facilmente projetar e simular o circuito acima, facilitando a análise de seu comportamento. Para tal, utilizamos o software Mathcad 2001 Professional®. Nele geramos os sinais de entrada do filtro, definimos a sua função de transferência e obtivemos os resultados de saída. Todos os resultados puderam ser numéricos e graficamente analisados, como veremos adiante.

No caso da implementação em hardware, geramos e somamos as três senoides através do software VisSim® e exportamos o sinal resultante para um arquivo tipo WAV, que pode ser executado e enviado para a saída da placa de som do computador. A digitalização do sinal, a filtragem e a conversão de digital para analógico novamente pôde ser obtida com um único circuito, conhecido como Kit DSP; no nosso caso, utilizamos um produzido pela Motorola.

CONCEITUAÇÃO BÁSICA

As razões básicas devido às quais não se pode implementar um filtro ideal, por exemplo passa-baixas, são os fatos de a resposta ao impulso desse filtro não atender ao princípio da causalidade e não ter comprimento finito ao longo do eixo do tempo. Para contornar o problema do comprimento infinito da resposta ao impulso somos compelidos a

¹ Renan S. Duque, Instituto Nacional de Telecomunicações, Santa Rita do Sapucaí, MG, Brazil, renan.duque@inatel.br

² Fábio A. R. Nascimento, Instituto Nacional de Telecomunicações, fabio.nascimento@inatel.br

³ Wilton N. A. Pereira, Instituto Nacional de Telecomunicações, wilton@inatel.br

⁴ Maurício Silveira, Instituto Nacional de Telecomunicações, msilveira@inatel.br

⁵ Sandro A. Fasolo, Instituto Nacional de Telecomunicações, sandro.fasolo@inatel.br

⁶ Francisco J. Fraga, Instituto Nacional de Telecomunicações, fraga@inatel.br

implementar um truncamento em determinado ponto dessa resposta, a prejuízo de uma oscilação, tanto na banda de rejeição, como na banda de passagem da resposta em frequência do filtro. No caso de contornar o problema com o princípio da causalidade, faz-se necessário implementar um atraso na resposta impulsiva temporal, deslocando-a de modo que a extremidade negativa coincida com o ponto de zero do eixo do tempo. Este atraso, causa, forçosamente, uma defasagem linear ao longo do eixo de frequência da resposta do filtro [1].

A Figura 3 ilustra a resposta impulsiva do filtro, truncada e atrasada, devido a necessidade de aproximação para a resposta ideal de um filtro passa-baixas digital.

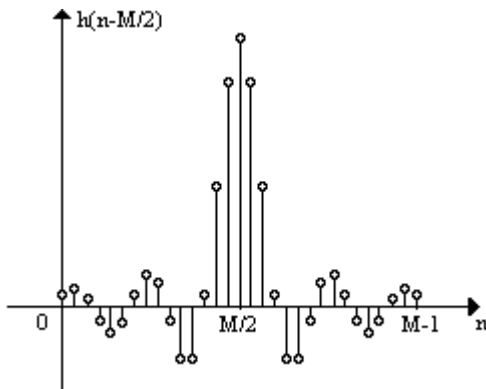


FIGURA. 3
RESPOSTA IMPULSIVA TRUNCADA E ATRASADA DE UM FILTRO DIGITAL
PASSA-BAIXAS REAL

A Figura 4 ilustra o que acontece no módulo e na fase da resposta em frequência do filtro passa-baixas digital quando truncamos e deslocamos as amostras da resposta ao impulso no domínio do tempo. Note-se a ondulação nas bandas de passagem e rejeição, bem como a alteração na fase da resposta em frequência.

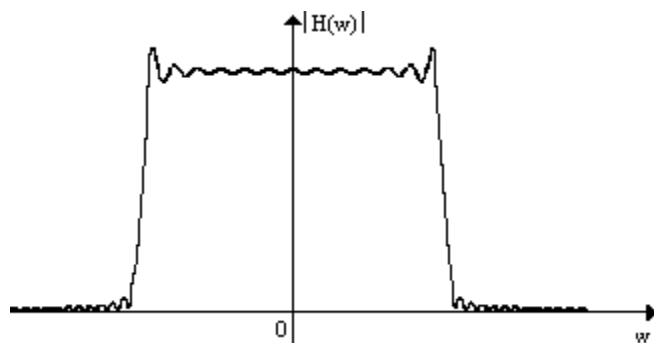


FIGURA. 4
MÓDULO DA RESPOSTA EM FREQUÊNCIA DO FILTRO PASSA-BAIXAS DIGITAL
REAL AO TRUNCAR-SE A RESPOSTA TEMPORAL AO IMPULSO

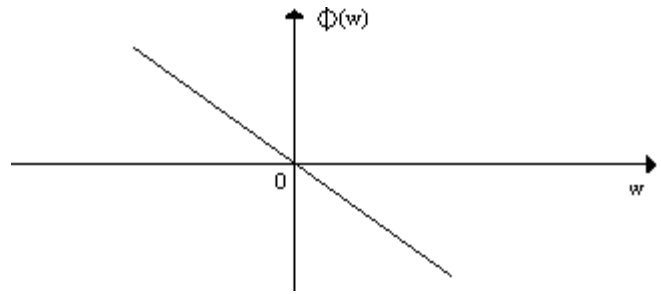


FIGURA. 5
FASE DA RESPOSTA EM FREQUÊNCIA DO FILTRO PASSA-BAIXAS DIGITAL
REAL AO DESLOCAR-SE A RESPOSTA TEMPORAL AO IMPULSO

O objeto principal de nosso estudo é o filtro digital FIR e, por isso, não abordaremos as questões complementares desse projeto, como os conversores A/D e D/A, o somador, a geração das senoides etc., partindo agora para uma abordagem mais detalhada do filtro em si, começando pelo seu equacionamento, tanto no domínio do tempo quanto no da frequência.

No domínio do tempo, o sinal de saída do filtro linear e invariante no tempo é a convolução da sua resposta ao impulso com o sinal de entrada [2], sendo representada por:

$$y[n] = x[n] * h[n] \quad (1)$$

$$y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k] \quad (2)$$

$$y[n] = \sum_{k=-\infty}^{\infty} h_k x[n-k] \quad (3)$$

onde $y[n]$ é o sinal de saída, $x[n]$ é o sinal de entrada e $h[n]$ é a resposta ao impulso do filtro.

No domínio da frequência, o sinal de saída do filtro linear e invariante no tempo é a multiplicação da função de transferência do filtro pelo seu sinal de entrada, na forma:

$$Y(z) = H(z)X(z) \quad (4)$$

$$H(z) = \sum_{k=-\infty}^{\infty} h[k]z^{-k} = \sum_{k=-\infty}^{\infty} h_k z^{-k} \quad (5)$$

$$Y(z) = \sum_{k=-\infty}^{\infty} h_k z^{-k} X(z) = \sum_{k=-\infty}^{\infty} h_k [z^{-k} X(z)], \quad (6)$$

onde $Y(z)$ é o sinal de saída, $X(z)$ é o sinal de entrada e $H(z)$ é a função de transferência do filtro.

Note-se que voltando (6) para o domínio do tempo, obtemos (3), que, neste caso é a própria equação de diferenças do filtro.

Expandindo tal equação de diferenças, temos:

$$y[n] = h_0x[n] + h_1x[n-1] + h_2x[n-2] + \dots, \quad (7)$$

Vimos, porém que a resposta impulsiva é truncada, então, a equação (7) pode ser rescrita por:

$$y[n] = h_0x[n] + h_1x[n-1] + \dots + h_{M-1}x[n-M+1] \quad (8)$$

onde $(M-1)$ é a última amostra da resposta impulsiva $h[n]$, ou h_n . Vemos que essa equação sugere uma implementação direta, conforme ilustra o diagrama em blocos da Figura 6 [1]:

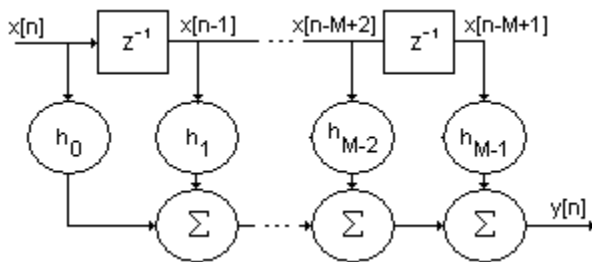


FIGURA. 6

IMPLEMENTAÇÃO DA EQUAÇÃO DE DIFERENÇAS NA FORMA DIRETA

PROJETO

O truncamento da resposta impulsiva temporal do filtro pode ser empregado através de técnicas muito mais elaboradas e vantajosas do que a simples consideração do sinal de resposta dentro de um intervalo de tempo. Tais técnicas são conhecidas como técnicas de janelamento, existindo diversas delas, cada qual, com características próprias, visando uma particularidade no funcionamento e na complexidade do circuito. No nosso caso, aplicaremos a janela de Kaiser, que apresenta a melhor relação entre a atenuação desejada para a banda de rejeição e a ordem do filtro (ou o número de amostras M consideradas na resposta impulsiva do filtro), que determina o tempo de processamento do circuito.

Seja $w[n]$ a função que define a janela de Kaiser [1]:

$$w[n] = \begin{cases} \frac{I_0 \left[\beta \left(1 - \left[\frac{(n-\alpha)/\alpha}{\beta} \right]^2 \right)^{1/2} \right]}{I_0(\beta)} & 0 \leq n \leq M-1 \\ 0 & n < 0, n > M-1 \end{cases} \quad (9)$$

onde:

$$\beta = \begin{cases} 0,1102(A-8,7) & A > 50 \\ 0,5842(A-21)^{0,4} + 0,07886(A-21) & 21 \leq A \leq 50 \\ 0 & A < 21 \end{cases} \quad (10)$$

$$\alpha = (M-1)/2, \quad (11)$$

A = Atenuação entre a banda de passagem e a banda de rejeição,

$$M-1 = \frac{A-7,95}{2,285\Delta w}, \quad (12)$$

Δw = largura da banda de transição, ou seja, a banda entre a frequência de rejeição e a frequência de passagem, em radianos.

Neste ponto já obtemos a função de janelamento $w[n]$ que vai ser multiplicada ponto a ponto pela resposta impulsiva de um filtro passa-faixa ideal, atrasada de $(M-1)/2$ amostras, gerando-se assim, a resposta impulsiva janelada e deslocada do filtro real. Com isto obtemos também a resposta em frequência do filtro real, que é uma aproximação da resposta ideal, pagando-se o preço imposto pelo janelamento e pelo truncamento. A seguir, veremos os resultados das simulações em software desse projeto.

SIMULAÇÃO EM SOFTWARE

Os resultados que obtivemos para $w[n]$ estão ilustrados a seguir:

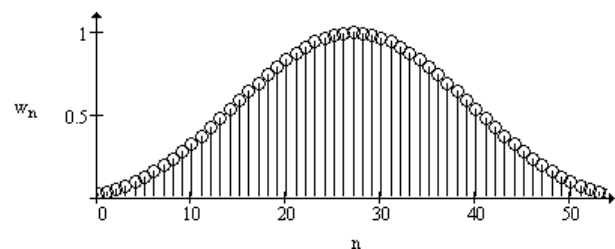


FIGURA. 7

REPRESENTAÇÃO GRÁFICA DA JANELA DE KAISER PARA A=60DB

Entenda-se por h_{d_n} a resposta impulsiva do filtro ideal atrasada, representada na figura 8 por uma linha e h_{pf_n} a resposta impulsiva h_{d_n} multiplicada pela janela de Kaiser $w[n]$, representada também na Figura 8 pelas amostras:

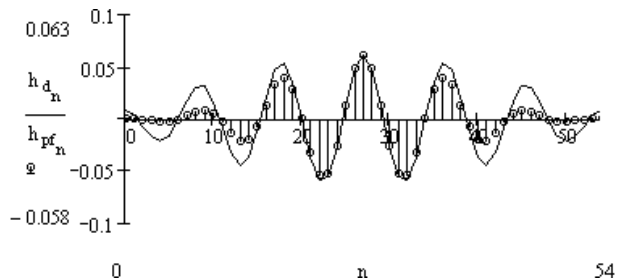


FIGURA. 8

RESPOSTAS IMPULSIVAS ATRASADAS, COM E SEM JANELAMENTO

As figuras 9 a 13 representam a resposta em frequência, o sinal de entrada, no domínio do tempo e da frequência e o sinal de saída, no domínio do tempo e da frequência, respectivamente.

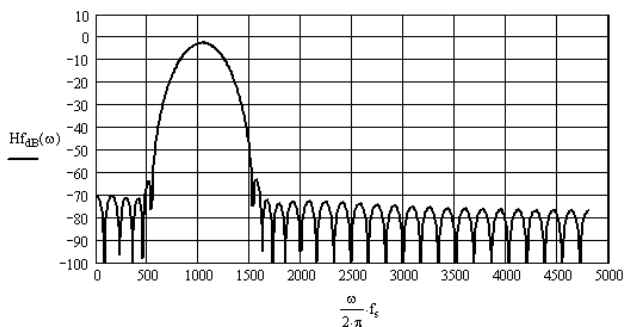


FIGURA. 9

RESPOSTA EM FREQUÊNCIA REAL OBTIDA NA SAÍDA DO FILTRO DIGITAL FIR

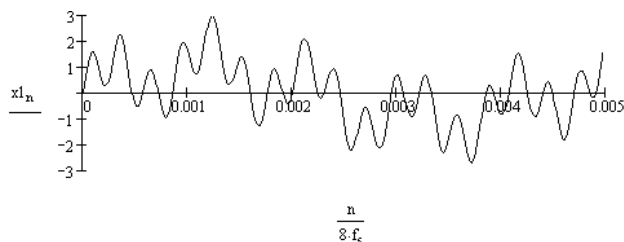


FIGURA. 10

SINAL DE ENTRADA NO DOMÍNIO DO TEMPO

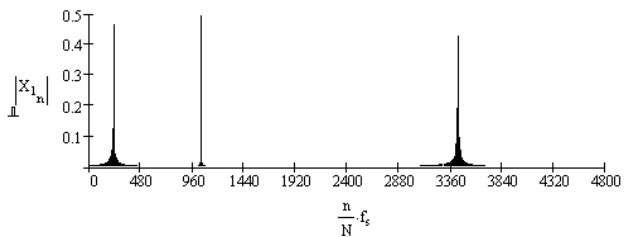


FIGURA. 11

SINAL DE ENTRADA NO DOMÍNIO DA FREQUÊNCIA

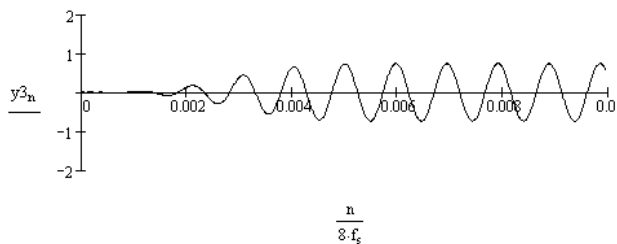


FIGURA. 12

SINAL DE SAÍDA NO DOMÍNIO DO TEMPO

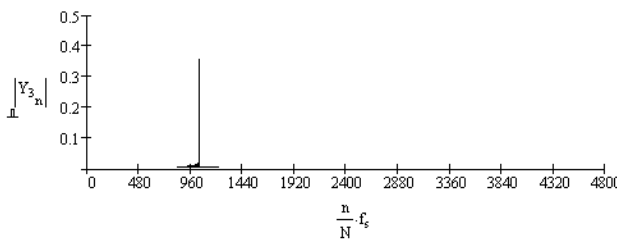


FIGURA. 13

SINAL DE SAÍDA NO DOMÍNIO DA FREQUÊNCIA

IMPLEMENTAÇÃO EM HARDWARE

O diagrama da Figura 6 foi implementado fazendo-se uso de um microprocessador especial para processamento digital de sinais, em forma, basicamente, de instruções *MAC* [3]. A instrução *MAC* (Signed Multiply-Acumulate) executa uma multiplicação entre seus dois primeiros operandos, que são data sources, no nosso caso de 24 bits e adiciona o resultado ao acumulador de destino, no nosso caso de 56 bits. Sendo assim, o processador armazena as M amostras mais recentes do sinal (onde M é a ordem do filtro) em algum lugar da memória Y , por exemplo; armazena todos os coeficientes h_n , em algum lugar da memória X , por exemplo e multiplica cada amostra de entrada pelo seu fator h_n correspondente, acumulando-se o resultado dessas multiplicações no acumulador A , por exemplo. Ao final, temos uma saída $y[n]$, que é a soma acumulada de todas as multiplicações, termo a termo, de h_n pela sua amostra de entrada correspondente, resultando assim no sinal de saída $y[n]$ que é o sinal de entrada $x[n]$ filtrado.

Nos servimos de um kit completo de DSP, com o propósito de pudermos aproveitar ainda a facilidade de utilizar o conversor A/D e D/A do próprio kit, a fim de discretizarmos o sinal de entrada $x(t)$ e obtermos $x[n]$.

As rotinas de inicialização do processador, de comunicação entre os conversores e o processador etc. são fornecidas juntamente com a documentação do Kit e carecem de pequenos ajustes, como por exemplo: sinal stereo ou mono, frequência de amostragem, etc. [3][4][5][7].

A Figura 14 ilustra o diagrama simplificado de um Kit DSP[4].

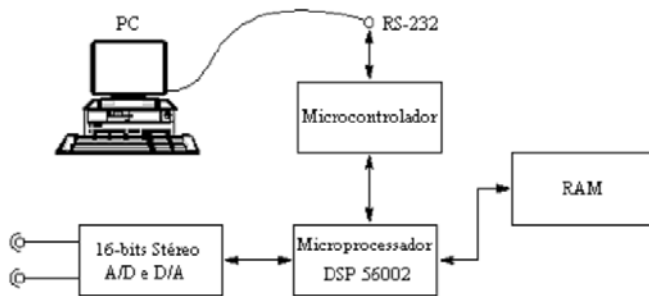


FIGURA. 14
DIAGRAMA SIMPLIFICADO DE UM KIT DSP

As vantagens que encontramos na utilização de um kit DSP, em relação a de um outro hardware convencional são, entre outras, tolerâncias menores (ausência de ajustes) e flexibilidade de implementação de mudanças eventuais no projeto. Uma outra grande vantagem foi a utilização de um conversor *A/D-D/A* já embutido no kit e apropriadamente ajustado ao DSP. Além disso, as rotinas em assembly para comunicação e controle entre o *A/D-D/A* e o DSP são facilmente obtidas na própria documentação do kit. Outra facilidade ainda é o fato de o kit trazer como suporte um programa depurador [5], que facilita o rastreamento de problemas nas rotinas e promove a gravação delas do PC para o DSP, via interface RS-232.

Para fins didáticos seria ideal a implementação de uma via paralela analógica, capaz de realizar a mesma função do filtro FIR. Esta parte do projeto foi idealizada mas não pôde ser concluída e será alvo de um estudo futuro.

CONCLUSÕES

O objetivo fundamental deste artigo é apresentar um projeto de um filtro do ponto de vista educacional e poder complementar a necessidade do aluno com alguns fundamentos teóricos ligados à comunicação digital. Entre estes, podemos destacar a Transformada de Fourier de

Tempo Discreto, Transformada Z, operações inversas, algumas propriedades tais como convolução, deslocamento no tempo e na frequência, linearidade etc. Houve também a necessidade de um estudo apurado de função de transferência, amostragem e equação de diferenças [6]. Nota-se, então, a grande relação entre estes conceitos abordados em um curso de Sinais e Sistemas e os projetos dos filtros digitais FIR.

A simulação em software facilitou sobremaneira a fixação dos conceitos obtidos pelo aluno em sala de aula, visto que é uma aplicação direta, na forma de cálculo numérico. Além disso, propicia ao aluno uma base sólida para implementação do projeto em hardware e comparação de resultados.

O trabalho com o hardware aproximou de maneira altamente positiva os conceitos da teoria à prática. Avaliamos ser esta uma necessidade premente em todas as disciplinas de um programa de pós graduação, pois o aluno sempre encontra uma barreira considerável entre a simulação e a implementação.

AGRADECIMENTOS

Nossos agradecimentos sinceros aos professores Fabiano Valias Carvalho e José Panaro e também às instituições FINATEL e ERICSSON pelo apoio.

BIBLIOGRAFIA

- [1] Fraga J. F. *Apostila da disciplina TP503 do INATEL*, 1ª e 2ª partes.
- [2] Haykin S., Van Veen B. *Signals and Systems*, New York: John Wiley & Sons, 1999. 694 p.
- [3] *DSP56000 24-bit Digital Signal Processor Family Manual*, Austin: Motorola, 1994.
- [4] *DSP56002EVM Quick Start Guide*, Motorola.
- [5] *Motorola DSP Assembler Reference Manual*, Austin: Motorola, 1996.
- [6] Pereira W. N. A. *Notas de aula da disciplina TP502, INATEL*, Transf. Z, Laplace e Fourier.
- [7] *DSP56002 24-bit Digital Signal Processor User's Manual*, Austin: Motorola, 1994.