

PLATAFORMA DIDÁTICA PARA PROCESSAMENTO DIGITAL DE SINAIS: APLICAÇÕES EM FILTROS DIGITAIS EM TEMPO REAL

EDUCATIONAL PLATFORM FOR REAL-TIME DIGITAL SIGNAL PROCESSING: PRACTICAL APPLICATIONS OF DIGITAL FILTERS

Wesley Becari,¹ Rodrigo B. dos Santos,² André B. Carlos,² Rafael A. Biliatto,² Henrique E. M. Peres³

DOI: 10.5935/2236-0158.20170004

RESUMO

As principais plataformas de desenvolvimento de processamento digital de sinais são as FPGAs (*Field-programmable gate arrays*) e os DSPs (*Digital Signal Processors*). Entretanto, seu uso normalmente fica restrito a ambientes industriais e de pesquisa que têm condições de financiar tais tecnologias. Através do conceito de DSC (*Digital Signal Controllers*), microcontroladores acrescidos de operações em ponto flutuante, instruções matemáticas e periféricos, tornou-se possível a redução do custo desses microprocessadores, e facilitou-se a operação e o desenvolvimento de projetos e experimentos didáticos. Este trabalho apresenta uma plataforma didática de baixo custo para o desenvolvimento de experiências de Processamento Digital de Sinais (PDS). Para tanto, são apresentados o conteúdo teórico e prático do desenvolvimento de filtros digitais, métodos de caracterização dos filtros digitais desenvolvidos e, ao final, é realizada uma discussão do uso da plataforma em sala de aula.

Palavras-chave: Ensino-aprendizagem; tecnologias de ensino; processamento digital de sinais; filtros digitais.

ABSTRACT

The main platforms for digital signal processing in real-time are the FPGAs (Field-programmable Gate Arrays) and the DSPs (Digital Signal Processors). However, these platforms are usually restricted to industrial and research environments that can finance such technologies. Through the concept of DSC (Digital Signal Controllers), microcontrollers with floating point and mathematical operations can reduce the cost of microprocessors, as well as facilitating the development of digital signal processing projects and the access to educational experiments. This paper proposes a low cost educational platform for developing a DSP laboratory. We present theoretical and practical aspects of the digital filter development, characterization methods and discussion about the use of this platform in the classroom.

Keywords: Teaching-learning; teaching technologies; digital signal processing; digital filtering.

1 Pesquisador, Departamento de Engenharia de Sistemas Eletrônicos, Escola Politécnica da Universidade de São Paulo; Professor, Faculdade de Ciências da Fundação Instituto Tecnológico de Osasco; wesley@lme.usp.br

2 Engenheiros, Faculdade de Ciências – Fundação Instituto Tecnológico de Osasco; rodrigobrito8618@gmail.com, dredicarlo@gmail.com e rafaelbiliatto@gmail.com

3 Pesquisador, Departamento de Engenharia de Sistemas Eletrônicos, Escola Politécnica da Universidade de São Paulo; Professor, Faculdade de Ciências da Fundação Instituto Tecnológico de Osasco, hperes@lme.usp.br.

INTRODUÇÃO

Filtros são sistemas lineares e invariantes no tempo, cuja função é selecionar ou rejeitar (atenuar) faixas de frequências determinadas. Eles podem ser classificados em analógicos ou digitais.

Filtros analógicos são circuitos eletroeletrônicos implementados utilizando-se componentes discretos: resistores, capacitores e indutores para o caso dos filtros passivos, e, adicionalmente, amplificadores operacionais para o caso de filtros ativos. Esses filtros são objeto de estudo dos alunos de Engenharia Elétrica nos cursos de Circuitos Elétricos e Eletrônicos, nos quais se faz a análise e síntese dos filtros para diferentes topologias de circuitos.

Por sua vez, os filtros digitais são discutidos, principalmente, nos últimos anos do curso de Engenharia Elétrica, na disciplina “Processamento Digital de Sinais” (PDS), subsequente à “Sinais e Sistemas”. Em algumas instituições, o próprio curso de PDS nada mais é do que a extensão do curso de Sinais e Sistemas para Tempo Discreto, cabendo ao professor apresentar aos alunos os conceitos de “sinais e sistemas em tempo discreto”, “série de Fourier em tempo discreto”, “transformada de Fourier em tempo discreto”, e “transformada discreta de Fourier”.

Dessa forma, é usual que não se dê ênfase ao projeto de filtros digitais. Além disso, quando é apresentado esse conteúdo, não é oferecida ao aluno a parte prática, seja através do projeto de filtros digitais para processamento *off-line*, no qual o processamento ocorre com dados previamente coletados e armazenados, ou através de processamento em tempo real (*real-time*), no qual os dados são amostrados, processados e reconstruídos quase que instantaneamente (SMITH, 1999; KUO *et al.*, 2013; REAY, 2012).

Nesse sentido, o ensino de PDS demanda que sejam desenvolvidas plataformas para ensino-aprendizagem, que permitam aos alunos

um contato com a prática de engenharia e que proporcione uma experiência com aplicações reais de processamento de áudio, vídeo e comunicações (SMITH, 1999), visto que no parecer CNE/CES nº 1.362/2001 (2002), que trata das diretrizes nacionais para os cursos de engenharia, destacam-se as competências e habilidades de projetar e conduzir experimentos e interpretar resultados; identificar, formular e resolver problemas de engenharia.

Como já mencionado, tanto os DSPs como as FPGAs são sistemas eletrônicos que possuem um alto custo, somados à alta complexidade de operação e aprendizado. Dessa maneira, torna-se oneroso seu uso em ambientes de ensino. Diante dessas dificuldades, os DSCs (*Digital Signal Controllers*) apresentaram-se como alternativa, já que consistem em microcontroladores com blocos em *hardware* para operações em ponto fixo e flutuante, e instruções otimizadas para PDS. Tais características permitem a construção de aplicações em um microprocessador de baixo custo e que possui, internamente, periféricos fundamentais, como os conversores analógico-digitais (A/D) e conversores digital-analógicos (D/A).

A partir da perspectiva de uma lacuna em aplicações reais e práticas de processamento digital de sinais, este trabalho apresenta os resultados do desenvolvimento de uma plataforma didática de baixo custo para projeto e implementação de filtros digitais em tempo real.

Primeiramente, são apresentadas as metodologias de projeto de filtros digitais. Em seguida, é descrito o desenvolvimento da plataforma didática para aquisição e condicionamento de sinais elétricos, ambientes para programação, instrumentação para caracterização de filtros e fluxograma de projetos. Como resultados, são descritas a caracterização dos filtros projetados e a análise do desempenho dos códigos. Por fim, é destacado o uso da plataforma em sala de aula através de uma proposta com enfoque mais científico e

investigativo, no qual os alunos são estimulados a desenvolver projetos a partir das ferramentas estudadas e diante de um problema, e também formular procedimentos e desenvolver capacidade de análise dos resultados.

FILTROS DIGITAIS

Filtros digitais utilizam processadores digitais para realizar cálculos numéricos sobre os valores de sinais amostrados. Suas aplicações usuais estão no processamento de sinais de áudio e vídeo. No entanto, pode ser usado em qualquer aplicação que requeira o processamento em tempo real, como controle e automação, condicionamento de sinais, comunicações, entre outras (SMITH, 1999).

O projeto de um filtro é realizado em três etapas: especificação dos requisitos de projeto; análise das aproximações utilizando sistemas discretos causais; e implementação do sistema. A classificação básica de filtros digitais se dá quanto à sua resposta ao impulso: filtros FIR (*Finite Impulse Response*) e filtros IIR (*Infinite Impulse Response*) (OPPENHEIM, SCHAFER, 2010) (PROAKIS; MANOLAKIS, 1996).

Projeto de filtros FIR

Os filtros FIR são empregados quando o projeto requer um filtro com fase linear. Essa característica é importante para aplicação e processamentos de áudio e imagem, bem como transmissão de dados, posto que evita distorções no sinal reconstruído. Esses filtros são realizados de modo não recursivo, ou seja, não possuem realimentação, dessa forma são sempre estáveis. Neles os efeitos de precisão finita e dos erros de quantização são menos severos. Contudo demandam uma elevada ordem para obter os mesmos resultados comparativos ao um filtro IIR. Um filtro de ordem M é descrito pela equação de diferenças (1):

$$y[n] = b_0x[n] + b_1x[n-1] + \dots + b_Mx[n-M+1] \\ = \sum_{k=0}^{M-1} b_kx[n-k] \quad , \quad (1)$$

no qual b_k são os coeficientes do filtro FIR, $x[n]$ e $y[n]$ são, respectivamente, a entrada e a saída do filtro. O projeto desses filtros se resume em encontrar os coeficientes da resposta ao impulso que satisfaçam um determinado conjunto de especificações da resposta em frequência.

Os principais métodos de projeto de filtros FIR são: janelamento, amostragem em frequência, projeto iterativo baseado em restrições ótimas, entre outros. Nos cursos de PDS, são desenvolvidas as equações do projeto de filtros FIR por transformada de Fourier, seguidas pelo método do janelamento.

Projeto de filtros FIR por janelas

Como já mencionado, o projeto de filtros FIR por janelas é baseado diretamente na aproximação de uma resposta em uma frequência desejada com condição de fase linear. Essa resposta é escolhida com base nos filtros seletivos projetados no domínio da frequência. Esses filtros são sempre não causais e com resposta ao impulso infinita.

Para tornar esses filtros realizáveis, isto é, causais, trunca-se sua resposta ao impulso. Dessa forma, é obtida uma função aproximada para cada tipo de filtro (passa baixas, passa altas, rejeita faixas e passa faixas). Para um filtro passa baixas, a solução é dada pela seguinte equação:

$$h_d[n] = \frac{\text{sen} \left[2\pi f_c \left(n - \frac{(M-1)}{2} \right) \right]}{\left[n - \frac{(M-1)}{2} \right] \pi}, \quad 0 \leq n \leq M-1 \quad (2)$$

no qual M corresponde à ordem correspondente ao filtro e f_c à frequência de corte (OPPENHEIM; SCHAFER, 2010).

Os coeficientes $h_d[n]$ correspondem à aproximação da resposta em frequência desejada com duração finita. Tal solução é equivalente à multiplicação da resposta impulsiva ideal por uma janela retangular de duração finita $w[n]$.

$$w_{ret}[n] = 1, \text{ para } 0 \leq n \leq M-1 \quad (3)$$

Com a determinação dos coeficientes de $h[n]$, determina-se o filtro FIR.

$$h[n] = h_d[n] \cdot w[n] \quad (4)$$

Como a multiplicação no domínio do tempo é equivalente à convolução no domínio da frequência, o espectro de um sinal janelado é a convolução do espectro do sinal original com o espectro da janela. Para uma janela retangular, a convolução resulta em um filtro com ondulações tanto na banda de passagem quanto na banda de rejeição.

A fim de evitar as ondulações, diversas janelas foram propostas para o projeto de filtros FIR. Algumas das mais utilizadas são: *Bartlett* (triangular), *Hanning*, *Hamming*, *Kaiser*, *Blackman*, entre outras (TAN; JIANG, 2013). Em específico, a função da janela *Hamming* é descrita por:

$$w_{ham}[n] = 0,54 - 0,46 \cos\left(\frac{2\pi n}{M-1}\right), \quad 0 \leq n \leq M-1, \quad (5)$$

A ordem do filtro é calculada a partir dos requisitos de transição entre a banda de passagem e a banda de rejeição, considerando a largura da faixa de transição normalizada $\Delta f_{norm} = |f_s - f_i|/f_a$ no qual f_p corresponde à frequência de corte inferior, f_s à frequência de corte superior, e f_a à frequência de amostragem. Para a janela *Hamming*, por exemplo, a ordem é obtida por $\Delta f = 3,3/N$. Dessa janela, tem-se a máxima ondulação na faixa de passagem (A_p) de 0,0194 dB e a máxima atenuação na faixa de rejeição (A_s) igual a 54,3 dB.

Projeto de filtros IIR

Os filtros IIR são recursivos, ou seja, os valores de saída dependem dos valores de entrada atuais e anteriores e de saída anteriores. Sua resposta ao impulso possui duração infinita; tal característica permite que eles sejam projetados a partir das aproximações de filtros analógicos (TAN; JIANG, 2013).

Em comparação com os filtros FIR, os filtros IIR apresentam menor número de coeficientes para um mesmo requisito, sendo assim processados mais rapidamente e com maior eficiência no uso da memória do microprocessador. Contudo, a implementação possui maior complexidade em comparação com os filtros FIR, pois, ao possuir polos na função de transferência, os filtros podem, em alguns casos, ser instáveis. Além disso, os filtros IIR são mais suscetíveis a erros de quantização.

A equação de diferenças a seguir corresponde à saída do filtro digital tipo IIR:

$$y[n] = \sum_{k=0}^M b_k x[n-k] - \sum_{l=1}^N a_l y[n-l], \quad (6)$$

no qual b_k e a_k são os coeficientes do filtro, M corresponde à ordem do filtro e N corresponde ao grau do numerador de função do filtro.

Os filtros IIR são realizados, principalmente, a partir do projeto de filtros analógicos, que são padronizados e tabelados. As principais topologias são: *Butterworth*, *Chebyshev I e II*, *Bessel*, e *Elíptico*. Uma vez projetados, os filtros analógicos devem ser realizados no tempo discreto. As três principais técnicas para transformar um filtro de tempo contínuo em tempo discreto são: aproximação de derivadas da equação diferencial, invariância na resposta ao impulso e a transformação bilinear. Todas as técnicas atuam sobre a função de transferência do filtro, mapeando o plano s da transformada de Laplace para o plano da transformada z .

É usual apresentar aos alunos o método da transformação bilinear, pois não há limitação em relação ao tipo de filtro. De maneira direta, permite que a função de transferência do filtro analógico seja reescrita como uma função de transferência digital, levando em consideração a igualdade (7):

$$s = \frac{2}{T_a} \left[\frac{1 - z^{-1}}{1 + z^{-1}} \right], \quad (7)$$

no qual T_a corresponde a tempo (taxa) de amostragem do sinal. A função de transferência do filtro digital será dada por (8):

$$H(z) = H_a \Big|_{s = \frac{2}{T_a} \frac{z-1}{z+1}}, \quad (8)$$

no qual H_a corresponde à função de transferência do filtro analógico e H à função em tempo discreto.

Em relação aos protótipos de filtros analógicos, o projeto de um filtro passa baixas com aproximação *Butterworth* tem o módulo ao quadrado da sua função de transferência definido como:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \left(\frac{\Omega}{\Omega_c}\right)^{2N}} = \frac{1}{1 + \left(\frac{-s^2}{\Omega_c^2}\right)^N}, \quad (9)$$

no qual Ω_c é a frequência de corte do filtro analógico e N é sua ordem.

A aproximação *Butterworth* é caracterizada por uma resposta em amplitude que é maximamente plana na faixa de passagem e decrescente na banda de rejeição (OPPENHEIM; SCHAFER, 2010; PROAKIS; MANOLAKIS, 1996). Os polos da função de transferência, p_k , que satisfazem o projeto do filtro *Butterworth*, são descritos por:

$$p_k = \Omega_c e^{j\pi/2} e^{j(2k+1)\pi/2N},$$

$$\text{para } k = 0, 1, \dots, N-1, \quad (10)$$

Nota-se que esses polos estão localizados no semiplano esquerdo do plano s , em pontos regularmente espaçados em um círculo de raio Ω_c .

A partir do filtro passa baixas, é possível obter qualquer tipo de filtro (passa altas, passa faixas, entre outros). Esse procedimento é realizado com base em transformações em frequências efetuadas a partir da função de transferência projetado.

MATERIAIS E MÉTODOS

Projeto da plataforma didática

O núcleo da plataforma didática foi implementado com o *kit* de desenvolvimento de baixo custo STM32F4 Discovery, que contém o microcontrolador ARM Cortex™-M4 STM32F407VG, com frequência de *clock* de 168 MHz e diversos periféricos configuráveis, destacando-se dois conversores A/D e D/A de 12 *bits* de resolução.

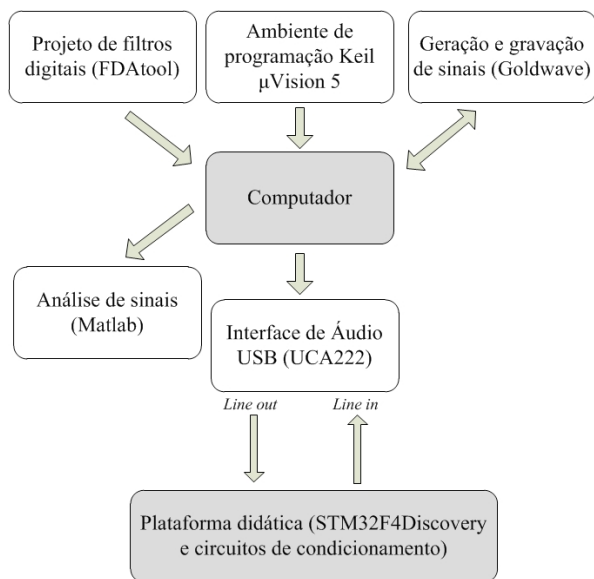
Os microprocessadores Cortex-M4 de 32 bits possuem características essenciais para o processamento digital de sinais (YIU, 2013). Seu microprocessador RISC (*Reduced Instruction Set Computing*), com arquitetura Harvard, apresenta uma lista de instruções de ciclo único, como a função MAC (*Multiply and Accumulate*), além de funções aritméticas SIMD (*Single Instruction, Multiple Data*) e unidade de ponto flutuante. Seu desempenho é de 1,25 DMIPS/MHz (*Dhrystone Million Instructions Per Second/MHz*), comparável aos principais microprocessadores atuais.

A Figura 1 apresenta o fluxograma de desenvolvimento utilizando a plataforma. Ini-

cialmente, são realizados a especificação e o projeto dos filtros digitais, considerando as equações analíticas ou ferramentas matemáticas do Matlab, em especial, o *Toolbox Fdatool*. A programação e prototipação do microcontrolador é realizada em linguagem ANSI C, através da versão gratuita do Keil μ Vision5. Para a gravação e geração de sinais arbitrários, é utilizado o *software* GoldWave.

A interface de áudio Behringer UCA222 é responsável pela geração de sinais editados no computador e pela aquisição de sinais oriundos da plataforma didática. Essa interface de áudio flexível possui dois canais de entrada (*line-in* RCA), dois canais de saída (*line-out* RCA), frequência de amostragem de 44.100 amostras/s e resolução de 16 bits.

Figura 1 – Fluxograma de desenvolvimento de filtros digitais em tempo real.

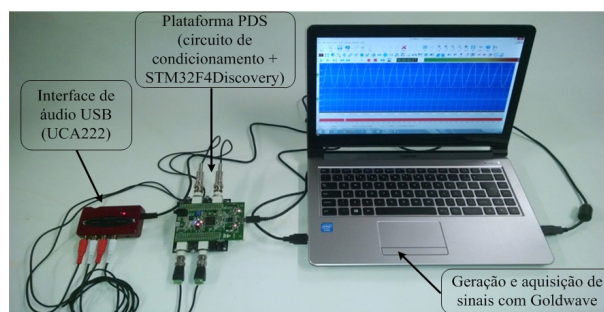


A plataforma didática realiza o condicionamento e amostragem do sinal gerado pela interface de áudio (*line out*). O dado amostrado é processado pelo algoritmo do filtro previamente programado. Após a execução, o microcontrolador reconstrói o valor de saída utilizando o seu conversor D/A. O sinal de saída é condicionado, amostrado pelo canal de entrada (*line-in*) da interface de áudio e armazenado através do *software* GoldWave. Nessa

aquisição, é possível tanto efetuar a análise do sinal no próprio *software*, como salvar os dados em arquivos PCM-WAV (extensão *.wav*) normalizados entre -1 e 1. Esses dados são facilmente manipulados por programas matemáticos, como é o caso do Matlab.

A Figura 2 apresenta os instrumentos constitutivos da plataforma PDS implementada, em conjunto com a interface de áudio e o computador.

Figura 2 – Plataforma PDS com circuitos de condicionamento, interface de áudio e software de geração e aquisição de sinais.



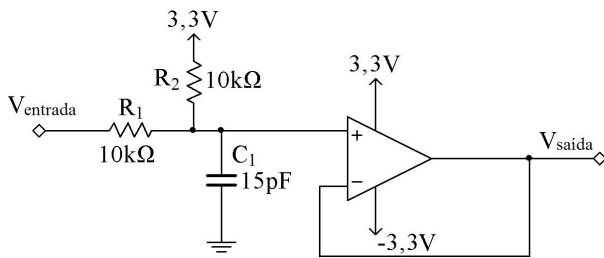
Ressalta-se a ausência de osciloscópios, geradores de sinais e fontes de alimentação, equipamentos que, normalmente, têm um custo elevado. Com isso, as experiências são otimizadas, focando no seu conteúdo, evitando tempo desperdido com polarizações e conexões entre componentes, além de eliminar possíveis ruídos.

Circuitos de condicionamento

Comumente, os conversores A/D e D/A internos a microcontroladores são unipolares, ou seja, operam somente com tensões de entrada e saída positivas, no caso do STM32F4 Discovery com tensões entre 0 e +3,3. Dessa maneira, foi projetada e fabricada uma placa de circuito impressa com três principais circuitos de condicionamento.

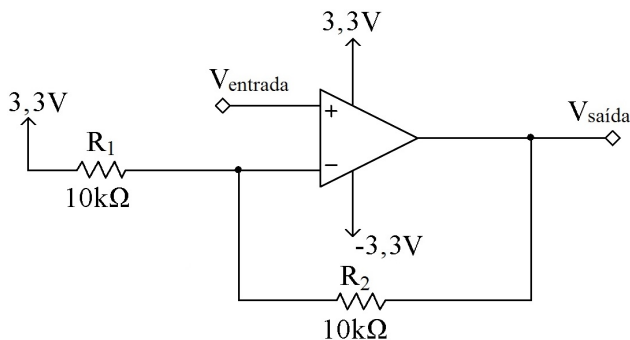
Primeiramente, para expandir a faixa de tensões na entrada, foi necessário um circuito para elevar a tensão, capaz de converter sinais bipolares em unipolares, ou seja, de -3,3 a +3,3 V para 0 a +3,3 V, como demonstrado na Figura 3.

Figura 3 – Circuito grameador de tensão, conversor de tensão bipolar para unipolar.



Outro circuito, conforme figura 4, foi adicionado a saída, garantindo que a tensão unipolar de 0 a +3,3V da saída do D/A fosse convertida em um faixa de tensão bipolar de -3,3V a +3,3V. Dessa forma, um sinal de áudio (-1V a +1V) pode ser amostrado e reconstruído corretamente.

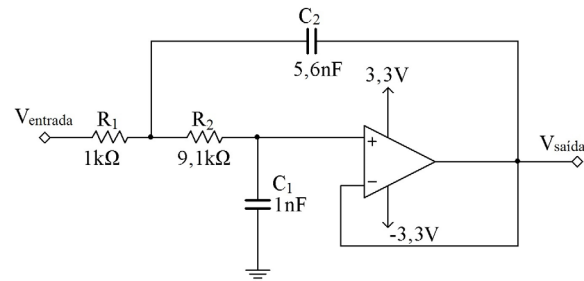
Figura 4 – Circuito conversor de tensão unipolar para bipolar.



Como se pretendeu trabalhar com sinais de áudio na frequência de amostragem de 44.100 amostras/s (qualidade usualmente utilizada em áudios MPEG-1, VCD, SVCD e MP3), foram idealizados filtros antirrebatiamento (*anti-aliasing*) e de reconstrução, junto aos conversores A/D e D/A, respectivamente. Esses filtros passa baixas analógicos foram projetados com frequência de corte em 22 kHz, utilizando topologia *Sallen-Key* e aproximação *Butterworth*. Essa frequência é obtida de acordo com o teorema de *Nyquist*.

A Figura 5 apresenta o esquema elétrico do filtro e os respectivos valores dos capacitores e resistores.

Figura 5 – Filtro passa baixas com topologia *Sallen-Key* e aproximação *Butterworth*, frequência de corte em 22 kHz.



A Figura 6 apresenta o diagrama em blocos dos circuitos de condicionamento e filtragem integrados ao *kit STM32F4 Discovery*. Por sua vez, a Figura 7 apresenta o protótipo do circuito de condicionamento. Esse é conectado diretamente ao *kit* de desenvolvimento *STM32F4 Discovery*.

Figura 6 – Diagrama em blocos dos circuitos de condicionamento integrados ao *kit STM32F4 Discovery*.

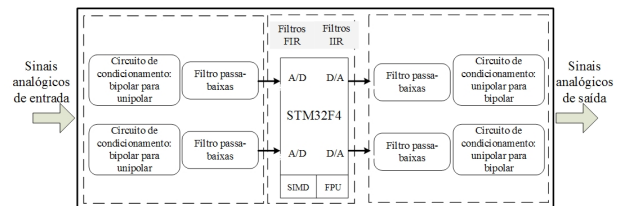
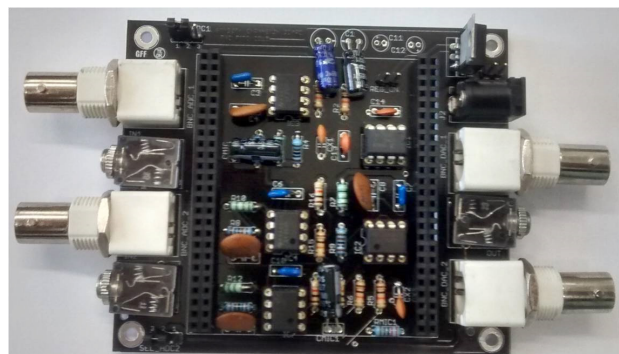


Figura 7 – Placa para condicionamento de sinais e interface com o *kit STM32F4 Discovery*.

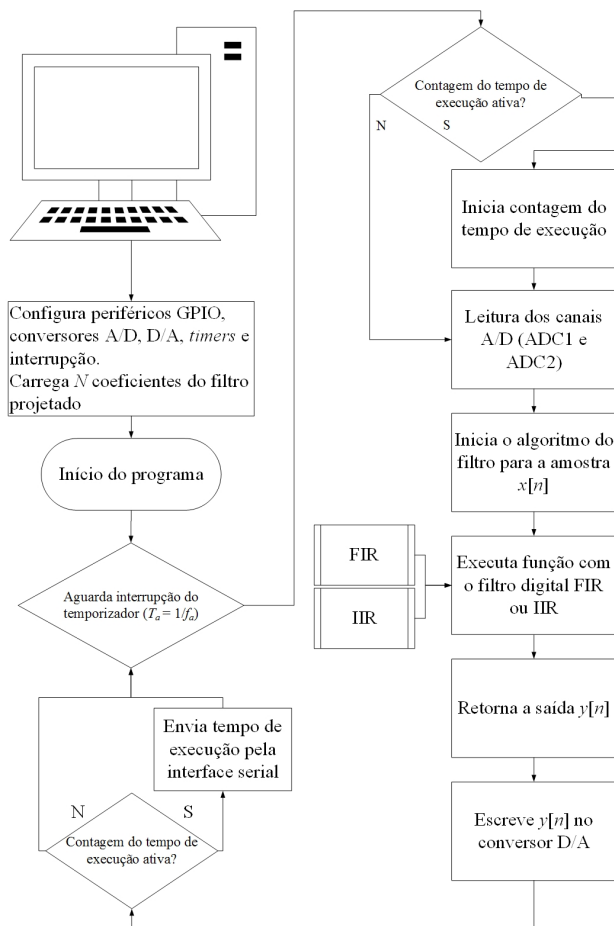


Fluxograma de execução do firmware

Para a execução do algoritmo, foi elaborada uma estrutura básica de código (protótipo) que executa as principais funções de configuração dos periféricos, controle e in-

terface do microcontrolador ARM Cortex-M4. A diferenciação entre os códigos se dá pela chamada do algoritmo do filtro, que contém os dados do filtro a ser executado, o número de coeficientes, o valor dos coeficientes com ponto flutuante e o algoritmo do referido filtro (FIR ou IIR). A Figura 8 mostra o fluxograma da estrutura básica do código.

Figura 8 – Fluxograma de execução do firmware contendo o filtro digital projetado.



Da Figura 8, nota-se que o microcontrolador amostra o sinal, processa o dado de entrada e atualiza o valor de saída no conversor D/A a cada intervalo de tempo correspondente ao tempo de amostragem. Esse procedimento é conhecido como *sample-based processing* (WELCH *et al.*, 2012). Nele o fluxo de dados de saída ocorre à medida que os dados são amostrados, oferecendo a menor latência entre as amostras de entrada e saída. Esse

procedimento ocorre dentro das interrupções preconfiguradas no periférico de temporização do microcontrolador.

O tempo de execução do algoritmo pode ser medido através de outro temporizador, que inicia sua contagem junto ao algoritmo do filtro. Após a finalização do bloco correspondente ao filtro, o microcontrolador envia o tempo de execução (parâmetro de desempenho) através de uma interface serial.

Os algoritmos foram desenvolvidos a partir das suas respectivas equações de diferenças, também conhecidos como algoritmos de “força bruta”. Neles o resultado de um filtro FIR é a soma das entradas anteriores, multiplicadas por coeficientes previamente calculados. Já um filtro IIR é a soma das entradas e saídas anteriores, multiplicadas também por coeficientes previamente projetados. A cada nova amostra, calcula-se a resposta do filtro e atualiza-se o *buffer* de entrada. Por meio de implementação direta, os alunos conseguem fazer uma associação direta entre a teoria dos projetos e sua execução na prática.

RESULTADOS E DISCUSSÃO

Os resultados obtidos da plataforma didática de desenvolvimento de filtros digitais em tempo real são descritos através da resposta em frequência dos filtros FIR e IIR desenvolvidos e da análise do desempenho do código implementado.

A resposta em frequência dos filtros pode ser obtida através de diferentes métodos: resposta impulsiva na entrada do filtro, varredura em frequência de sinais senoidais e análise através de ruído branco. Pode-se realizar a caracterização através de ruído branco, pois esse ruído possui densidade espectral de energia constante em todo o espectro de frequências (LATHI, 2007). Dessa forma, ao ser utilizado como entrada de um filtro digital, a saída corresponderá à resposta em frequência do filtro.

Para testar os algoritmos desenvolvidos, gera-se a função ruído branco durante um período de um minuto no ambiente GoldWave. Esse sinal gerado pela interface de áudio USB é amostrado, processado pela plataforma PDS e reconstruído. Por fim, é gravado pelo próprio GoldWave.

A análise da resposta do filtro é realizada através da densidade espectral de potência (*Power Spectral Density* – PSD). Essa função define a potência por unidade de banda em função da frequência, em outras palavras, a potência média por unidade de banda. Essa função fornece somente a informação de magnitude, não considerando a fase. Opta-se pelo uso dessa função nas medidas de sistemas práticos, pois mesmo sinais sem representação matemática podem ser analisados.

Esse recurso facilita a validação dos códigos desenvolvidos, uma vez que não é necessária a utilização de nenhum outro *hardware* como analisador de redes ou analisador de resposta em frequência, somente a análise através de *software* matemático.

Resposta em frequência dos filtros FIR

Foram projetados e analisados dois filtros digitais FIR: um filtro passa baixas de 10^a ordem, e um filtro passa baixas de 100^a ordem.

O primeiro, é um filtro FIR passa baixas projetado através de uma janela *Hamming*, com faixa de transição (Δf) de 14,5 kHz, frequência de corte de 10 kHz, e frequência de amostragem de 44.100 amostras/s. Os coeficientes foram calculados através de (2), (4) e (5) e descritos por: $b_0 = h[0] = 0,0007$, $b_1 = h[1] = -0,0163$, $b_2 = h[2] = -0,0238$, $b_3 = h[3] = 0,1371$, $b_4 = h[4] = 0,4023$, $b_5 = h[5] = 0,4023$, $b_6 = h[6] = 0,1371$, $b_7 = h[7] = -0,0238$, $b_8 = h[8] = -0,0163$, $b_9 = h[9] = 0,0007$.

O segundo filtro, com os mesmos requisitos de frequência de corte, contudo, é de 100^a ordem. As figuras 9 e 10 apresentam a

resposta em frequência do primeiro e segundo filtros, respectivamente.

Figura 9 – Resposta em frequência do filtro passa baixas FIR, 10^a ordem e frequência de corte 10 kHz.

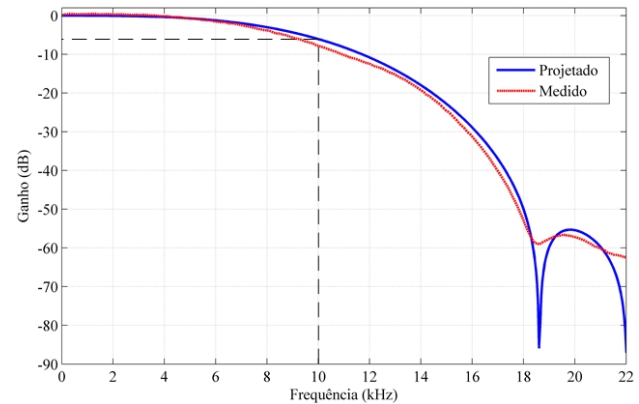
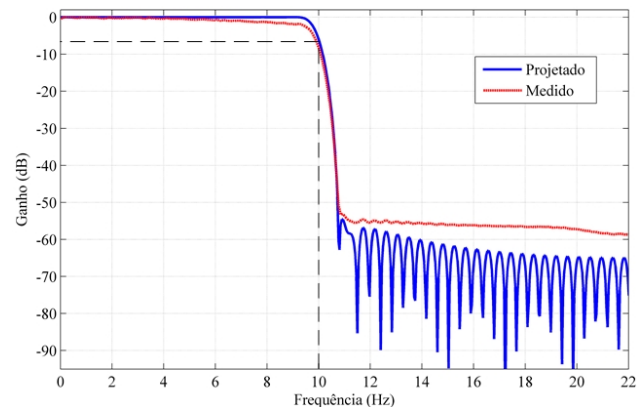


Figura 10 – Resposta em frequência do filtro passa baixas FIR, 100^a ordem e frequência de corte 10 kHz.



As figuras apresentam a resposta em frequência dos filtros FIR passa baixas projetados junto com a resposta dos filtros caracterizados. Nota-se grau de concordância entre os valores medidos e projetados. Além disso, fica evidente que quanto maior a ordem do filtro, mais próximo ele fica da resposta ideal.

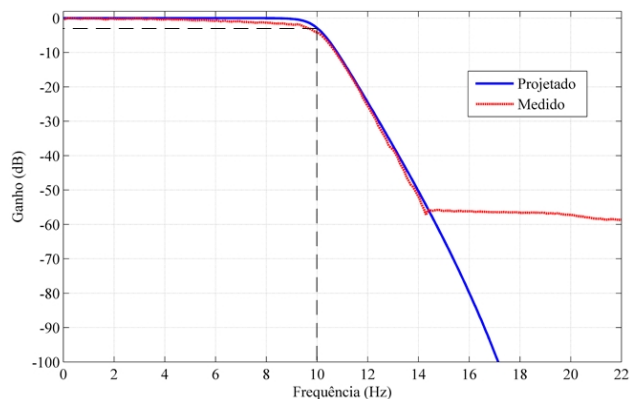
Para valores menores do que -52 dB, os lóbulos da resposta medida passam a ser atenuadas, devido ao efeito de quantização dos conversores A/D e D/A e da equalização realizada pela configuração da interface de áudio USB.

Resposta em frequência dos filtros IIR projetados

A título de comparação com os filtros FIR, foi projetado um filtro IIR de 10ª ordem passa baixas com frequência de corte em 10 kHz e aproximação *Butterworth*. O projeto desse filtro possui os seguintes coeficientes: $b_0 = 0,0013$, $b_1 = 0,0135$, $b_2 = 0,0605$, $b_3 = 0,1614$, $b_4 = 0,2825$, $b_5 = 0,3390$, $b_6 = 0,2825$, $b_7 = 0,1614$, $b_8 = 0,0605$, $b_9 = 0,0135$, $b_{10} = 0,0013$, $a_1 = -0,9260$, $a_2 = 1,7027$, $a_3 = -1,0215$, $a_4 = 0,8284$, $a_5 = -0,3160$, $a_6 = 0,1344$, $a_7 = -0,0298$, $a_8 = 0,0062$, $a_9 = -0,0006$ e $a_{10} = 0$.

Obteve-se a resposta em frequência de acordo com a Figura 11.

Figura 11 – Resposta em frequência do filtro passa baixas IIR, 10ª ordem e frequência de corte em 10 kHz.



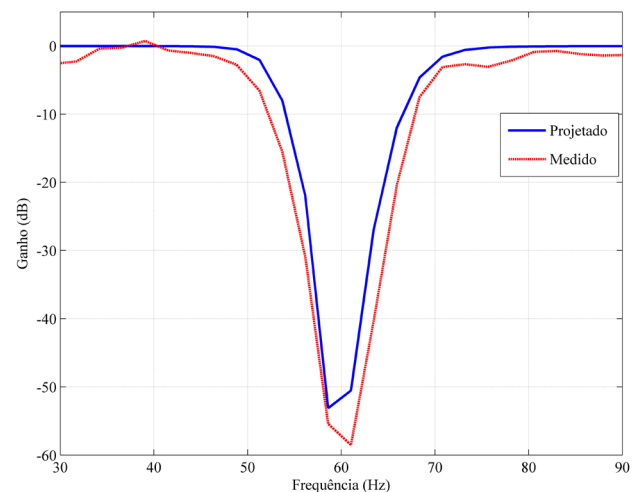
Novamente é possível verificar um alto grau de concordância entre os valores medidos e projetados. Fica claro que um filtro IIR de menor ordem já atenderia os requisitos que foram impostos para o filtro de 10ª ordem FIR.

Outro exemplo de projeto amplamente utilizado é o filtro digital rejeita faixa com frequências de rejeição em torno de 60 Hz. Foi implementado esse filtro para uma aproximação *Butterworth* de 6ª ordem, com frequência de amostragem 10.000 amostras/s, frequência de corte superior em 62 Hz e frequência de corte inferior em 58 Hz. Obteve-se, utilizando ferramentas matemáticas, os seguintes coefi-

cientes: $b_0 = 0,9975$, $b_1 = -5,9807$, $b_2 = 14,9454$, $b_3 = -19,9243$, $b_4 = 14,9454$, $b_5 = -5,9807$, $b_6 = 0,9975$, $a_1 = -5,9907$, $a_2 = 14,9579$, $a_3 = -19,9243$, $a_4 = 14,9328$, $a_5 = -5,9707$ e $a_6 = 0,9950$

A Figura 12 apresenta a resposta em frequência desse filtro. Nota-se que a banda de rejeição medida acompanha de forma satisfatória o resultado teórico. Além disso, o filtro implementado é estável.

Figura 12 – Resposta em frequência do filtro rejeita faixa IIR, 10ª ordem e frequência de corte em 1 kHz.



Como exemplo prático e didático, o filtro IIR foi utilizado para cancelamento de um ruído de 60 Hz sobre um sinal de ECG (eletrocardiograma). A Figura 13, adiante, apresenta o sinal no domínio do tempo com interferência do ruído, enquanto a Figura 14 demonstra o sinal ECG filtrado com o uso da plataforma.

Figura 13 – Sinal de ECG amostrado com ruído de 60 Hz.

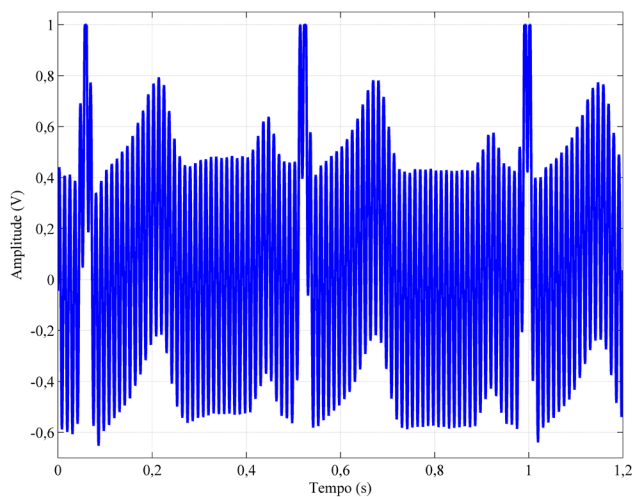
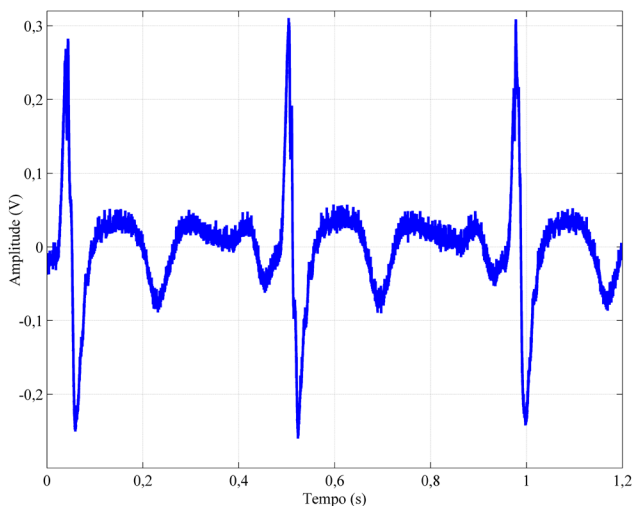


Figura 14 – Sinal ECG com filtro rejeita faixa.



Desempenho dos códigos e análise comparativa

O desempenho dos filtros digitais pode ser obtido através do seu custo de execução. Esse parâmetro possibilita analisar a complexidade do algoritmo mediante seu tempo de execução. Além disso, permite analisar a implementação dos filtros para a posterior otimização.

A Tabela 1 apresenta, comparativamente, o tempo de execução dos filtros digitais projetados. A plataforma DSP foi desenvolvida para operar com uma frequência de amostragem em 44.100 amostras/s, portanto o tempo máximo permitido para execução do algorit-

mo é o intervalo entre duas amostragens consecutivas, aproximadamente, 22,73 μ s.

Tabela 1 – Comparativo de tempo de execução dos filtros digitais projetados e caracterizados.

	Filtro FIR passa baixas 10ª ordem	Filtro FIR passa baixas 100ª ordem	Filtro IIR passa baixas 10ª ordem	Filtro IIR rejeita faixas 6ª Ordem
Tempo de execução (μ s)	1,98	12,82	1,05	1,06

Para os filtros passa baixas, nota-se que o filtro IIR apresenta menor tempo de execução, como é esperado. Esse custo, extremamente baixo, garante ao filtro IIR melhor resposta, comparado a um filtro FIR com mesma ordem, mostrando-se mais adequado para aplicações em tempo real.

Para os filtros FIR, foram realizados testes, além dos apresentados, com diferentes ordens e mesmos requisitos de projeto, no caso, frequência de corte de 10 kHz. Foram medidos tempos para: 6ª ordem (1,29 μ s), 50ª ordem (7,52 μ s), 130ª ordem (16,43 μ s), 150ª ordem (19,45 μ s), 160ª ordem (20,71 μ s) e 170ª ordem (22,10 μ s). O algoritmo FIR de 100ª ordem levou, aproximadamente, o dobro do tempo que um mesmo projeto com metade dos coeficientes. Nota-se, para um aumento do número de coeficientes, um crescimento linear do seu tempo de execução.

Através desses resultados, é demonstrada a principal limitação dos filtros digitais, que reside na eficiência do código implementado, somada à velocidade em que o microprocessador executa as operações. Assim, destaca-se para os alunos a necessidade de contínua aperfeiçoamento dos algoritmos, juntamente com o desenvolvimento de novas tecnologias de *hardware* para melhorar a eficiência do processamento de operações matemáticas complexas.

A plataforma didática em sala de aula

O planejamento do uso da plataforma didática foi realizado através de duas tarefas e duas atividades em laboratório. Primeiramente-

te, é apresentada a teoria acerca dos filtros FIR e, em específico, o projeto de filtros através da técnica de janelamento. É requisitado a grupos de dois alunos que projetem, como tarefa, dois filtros FIR, de acordo com determinados requisitos. Solicita-se que eles calcularem os coeficientes para pequenas ordens de filtros (até 10^a ordem). Além disso, sugere-se a implementação de *scripts* que realizem o cálculo dos coeficientes.

Esses valores são utilizados no ambiente de laboratório. Na primeira aula prática, são demonstrados o ambiente de desenvolvimento integrado, a plataforma PDS e o fluxo de projeto a ser realizado. Os alunos são levados a inserir os coeficientes previamente calculados e, posteriormente, são instigados a analisar o resultado na ferramenta matemática. Esse procedimento demanda mediação do professor que, nas intervenções, deve orientar não respondendo a todas as questões, mas sim fazendo com que os alunos indaguem a metodologia e os usos da plataforma.

O mesmo procedimento deve ser realizado em relação aos filtros IIR. É necessário conhecimento da teoria e dos métodos de projeto de filtros analógicos e sua posterior conversão em filtro digital. Para isso, o *software* Matlab é ferramenta fundamental, pois permite aos alunos o projeto de filtros IIR com rapidez. Tal qual no filtro FIR, a preparação para a aula de laboratório consiste em projetar dois filtros diferentes de pequenas ordens. Os coeficientes calculados auxiliam no projeto do filtro IIR em tempo real. No laboratório, os alunos são convidados a conduzir a experiência, tal como realizada anteriormente. Além disso, a própria aplicação do algoritmo do filtro pode ser desenvolvida pelos alunos, de acordo com o tempo e perspectiva do professor.

Ressalta-se que os alunos também devem discutir os métodos de análise de desempenho dos filtros. No caso, optou-se por avaliar o desempenho através do tempo de execução, que é uma medida indireta do nú-

mero de ciclos de *clock* necessários para a execução do código. Além disso, podem-se avaliar outras implementações dos filtros, como: uso de *buffer* circular; uso das formas diretas I e II, bem como realizações paralelas ou em cascata; uso de cascatas de filtros biquadráticos; entre outros. Temas como o uso de ponto fixo ou ponto flutuante também podem ser discutidos e comentados pelo professor.

Além disso, pode-se aproveitar o uso da plataforma para trabalhar questões relativas à amostragem, em especial: erros de quantização dos conversores, taxa de amostragem e rebatimento.

Cabe ressaltar que a proposta é realizada com um roteiro mínimo, no qual é solicitado o projeto dos filtros e se disponibiliza as ferramentas de *hardware* e *software*. Essa dependência mínima dos roteiros estimula a maior participação do aluno na pesquisa bibliográfica, na formulação da atividade, na aquisição de dados e na discussão dos resultados. Dessa forma, tenta-se limitar o trabalho mecanizado oriundo do uso rígido de roteiros (LEITE; ESTEVES, 2005).

Espera-se que, para além da resolução de um problema aplicado a uma situação prática de engenharia, os alunos compreendam os procedimentos de: especificação de requisitos, projeto, prototipação, caracterização e análise de desempenho de filtros digitais em tempo real. Nesse sentido, cabe aos alunos mostrarem-se capazes de interpretar os resultados e validar a metodologia proposta, através dos conhecimentos matemáticos relacionados.

CONCLUSÕES

Este trabalho apresenta uma proposta de plataforma didática de baixo custo para desenvolvimento de filtros digitais em tempo real. O trabalho foi estruturado de forma a conciliar a teoria básica de projeto de filtros digitais, somados a exemplos práticos de prototipação e caracterização dos filtros digitais.

Nessa proposta, foi utilizado um *kit* de desenvolvimento no lugar de osciloscópios, fontes, geradores de função e circuitos de processamento de sinais.

O *kit*, com o microcontrolador ARM Cortex-M4 STM32F407VG, demonstrou ser capaz de atender às expectativas de projeto e execução dos algoritmos de forma satisfatória.

A plataforma possibilita inúmeras abordagens no tocante à disciplina de PDS, pois é um sistema flexível e reprogramável. Todos os filtros projetados foram desenvolvidos a partir das suas respectivas equações de diferenças. Também foi proposta uma metodologia para análise de algoritmos através do tempo de execução do código inerente ao algoritmo do filtro.

Por último, é apresentado um roteiro que pode ser utilizado em sala de aula para nortear o uso da plataforma em duas experiências de laboratório. Dessa forma, pretende-se incentivar que escolas de tecnologia e engenharia ofereçam cursos práticos de PDS, fazendo, assim, com que os alunos tenham experiências de aplicações reais e maior contato com o uso de processadores digitais de sinal.

AGRADECIMENTOS

Os autores agradecem à Faculdade de Ciências da Fundação Instituto Tecnológico de Osasco (FAC-FITO).

REFERÊNCIAS

- KUO, S. M.; LEE, B. H.; TIAN, W. **Real-Time Digital Signal Processing**: fundamentals, implementations and applications, New York, US, Wiley, 2013.
- LATHI, B. P. **Sistemas e sinais digitais lineares**. 2. ed. Bookman, 2007.
- LEITE, L.; ESTEVES, E. Análise crítica de atividades laboratoriais: um estudo envolvendo estudantes de graduação. **Revista Electrónica de Enseñanza de las Ciencias**, v. 4, n. 1, 2005.
- OPPENHEIM, A. V.; SCHAFER, R. W. **Discrete-time Signal Processing**, 3. ed. Prentice-Hall, 2010.
- CNE/CES n.º 1.362/2001. Parecer despacho do ministro em 22/2/2002. Brasília, DF: **Diário Oficial da União** de 25 de fevereiro de 2002, Seção 1, p. 17.
- PROAKIS, J. G.; MANOLAKIS G. D. **Digital signal processing**: principles, algorithms, and applications. 3. ed. Prentice Hall, 1996.
- REAY, D. S. **Digital Signal Processing and applications with the OMAP-L138 eXperimenter**, New York, US, Wiley, 2012.
- SMITH, S. W. **The scientist and engineer's guide to Digital Signal Processing**. 2. ed. California Technical Publishing, 1999.
- TAN, L.; JIANG J. **Digital Signal Processing**: fundamentals and applications, 2. ed. Elsevier, 2013.
- WELCH, T. B.; CAMERON, H. G.; WRIGHT, M. G. **Real-Time Digital Signal Processing from MATLAB® to C with the TMS320C6x**. 2. ed. CRC Press, 2012.
- YIU, J. **The definitive guide to the ARM Cortex-M3 and Cortex-M4 Processors**, 3. ed. Newnes, 2013.

DADOS DOS AUTORES



Wesley Becari – Engenheiro e mestre em Engenharia Elétrica pela Escola Politécnica da USP. É pesquisador nos grupos de Micro-ondas e Optoeletrônica (gMO) e de Sensores Integráveis e Microsistemas (gSIM), ambos do Dep. Eng. Sistemas Eletrônicos – PSI, Escola Politécnica da USP. Também foi professor no curso de Engenharia Elétrica na Faculdade de Ciências da Fundação Instituto Tecnológico de Osasco – FAC-FITO. Realiza pesquisas nas áreas de: processamento digital de sinais; sistemas embarcados; dispositivos e circuitos de micro-ondas e RF; e sensores de gases e líquidos. Também atua no desenvolvimento de metodologias de ensino para engenharia.



Rodrigo Brito dos Santos – possui graduação em Engenharia Elétrica pela Faculdade de Ciências da Fundação Instituto Tecnológico de Osasco. Possui experiência na área de Engenharia Elétrica, com ênfase em projetos de painéis elétricos e sistemas de controle para máquinas e equipamentos.



André Batista Carlos – possui graduação em Engenharia Elétrica pela Faculdade de Ciências da Fundação Instituto Tecnológico de Osasco.



Rafael Augusto Biliatto – possui graduação em Engenharia Elétrica pela Faculdade de Ciências da Fundação Instituto Tecnológico de Osasco.



Henrique E. M. Peres – Doutor e mestre em Engenharia Elétrica, ambos pela Escola Politécnica da USP. Também é Bacharel em Física pelo Instituto de Física da USP. É pesquisador no grupo de Sensores Integráveis e Microssistemas (gSIM), no Dep. Eng. Sistemas Eletrônicos – PSI, Escola Politécnica da USP. Também foi professor no curso de Engenharia Elétrica na Faculdade de Ciências da Fundação Instituto Tecnológico da Osasco – FAC-FITO. Realiza pesquisa na área de microeletrônica com ênfase em materiais e processos, MEMS, sensores integráveis, sensores de gases e líquidos. Também atua no desenvolvimento de metodologias de ensino para engenharia.