

# Conversor Abaixador com Controle Digital

Diogo Filipe Micali Mota, Lucas Caldeira de Oliveira, Mário Alexandre Rodrigues  
Universidade Tecnológica Federal do Paraná  
Pato Branco, Brasil  
abril de 2021

**Abstract**—Este projeto apresenta um conversor estático abaixador com controle de tensão na carga. As lógicas de acionamento e controle serão implementadas de forma digital usando um microcontrolador. Também serão abordado a modelagem do circuito de condicionamento de sinais e as simulações computacionais.

**Index Terms**—conversor abaixador, conversor Buck, fonte chaveada, controle digital.

## I. INTRODUÇÃO

UM conversor estático abaixador, também chamado de conversor Buck, é um sistema de acionamento eletrônico para conversão de tensão CC/CC que permite controlar a tensão de saída através da comutação de chaves. Em regime estacionário, o Buck permite saídas quase constantes em níveis de tensão que variam entre 0 V e a entrada do circuito conforme o padrão de acionamento das chaves semicondutoras, sendo muito utilizado, por exemplo, em fontes de carregadores, devido à sua alta eficiência.

Este trabalho visa o desenvolvimento de um conversor Buck com acionamento por modulação por largura de pulso (PWM). Para obter uma melhor resposta na saída, será desenvolvido também um controlador implementado digitalmente com um DSP, assim como o circuito de condicionamento de sinais (CCS), composto por sensor, filtragem, ganho e *offset*.

## II. METODOLOGIA

Visando projetar um circuito que atenda aos requisitos de frequência e potência da planta, será adotada a seguinte metodologia de projeto.

A planta será modelada conforme análises por espaço de estados e por função de transferência (seção III), gerando as equações necessárias para as etapas subsequentes. O projeto será abordado na seção IV em três partes: projeto da planta, onde será utilizada a análise de resposta em frequência para determinar as frequências de corte do conversor e do filtro de acordo com as frequências determinadas do sinal de entrada e de amostragem e chaveamento; projeto do controlador, que vai envolver técnicas de controle em tempo discreto para gerar um controlador implementável digitalmente; e projeto do circuito, que consiste no dimensionamento dos componentes eletrônicos do circuito analógico.

Por fim, serão feitas simulações computacionais com os softwares Octave e LTspice, para comparação entre as respostas do modelo matemático e do circuito equivalente e, cujos códigos são apresentados na seção VI, juntamente com o código implementável do microcontrolador.

## III. MODELAGEM

A planta completa do circuito pode ser entendida em 2 partes, conforme a Figura 1: uma etapa de potência, composta pelo conversor Buck e um atuador; e uma etapa de sinal, contendo o circuito de medição e condicionamento e o microcontrolador/DSP (*digital signal processor*) implementando as lógicas de controle e acionamento conforme o sinal de referência na entrada.

Essencialmente, a etapa de potência é a planta em malha aberta, ou seja, o sistema em sua configuração original, contendo um sinal de entrada, uma saída e uma dinâmica natural. Ao medir a saída, calcular o erro e efetuar uma ação de controle (ou seja, ao adicionarmos a etapa de sinais), dizemos que houve uma realimentação no sistema, resultando na planta em malha fechada.

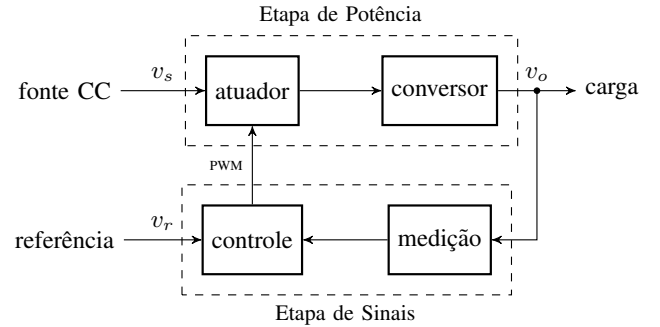


Fig. 1. diagrama simplificado da planta

Além dessa distinção, é possível separar a planta em circuitos analógicos e circuitos digitais. O primeiro contempla o conversor, o atuador, o *drive* de acionamento e a etapa de medição, composta por sensor, filtro e amplificador de instrumentação. Já o segundo é o próprio microcontrolador, que encapsula a lógica de controle e o sinal de acionamento do via PWM conforme a referência.

Considerando que o projeto será feito partindo da resposta em frequência, será feita inicialmente a modelagem matemática dos elementos que contêm dinâmica: o conversor e o filtro. Os demais elementos serão inseridos pontualmente conforme se avança no projeto, à medida em que vão se tornando relevantes para a análise.

### A. Funcionamento do Conversor Buck

Iniciando pelo componente fundamental da planta em malha aberta, o conversor Buck, também chamado de conversor abaixador, é composto de uma fonte de tensão CC  $v_s$  na

entrada, duas chaves  $S$  e  $D$  operando num período  $T$  conforme uma frequência de chaveamento/switching  $f_s = 1/T$ , um indutor  $L$  e um capacitor  $C$  responsáveis por eliminar a alta frequência da tensão chaveada e manter a tensão de saída em um nível constante, e um resistor  $R$  representando a carga.

O funcionamento do circuito ocorre conforme a comutação das chaves, sendo  $dc$  o ciclo de trabalho representando a porcentagem do período em que a fonte de entrada está fornecendo potência ao circuito. Quando  $S$  fecha durante um intervalo de tempo  $t_{on} = dc \cdot T$ , uma corrente circula da fonte  $v_s$  até a carga, carregando o capacitor e o indutor e polarizando inversamente o diodo  $D$ , que abre. Quando  $S$  abre, durante  $t_{off} = (1 - dc) \cdot T$ , imediatamente o diodo é polarizado e passa a conduzir, funcionando como uma chave fechada, e o capacitor e o indutor começam a descarregar. O diodo permanece conduzindo enquanto circular corrente em  $L$  ou até que a chave  $S$  feche novamente.

Na Figura 2 é exemplificado o circuito Buck idealizado, e na Figura 3 temos um circuito real, contendo a tensão de queda no diodo  $v_D$ , resistências da fonte, do indutor e do capacitor  $R_s$ ,  $R_L$  e  $R_C$ .

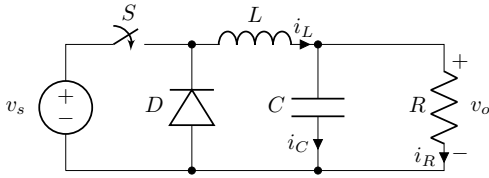


Fig. 2. circuito ideal do conversor Buck

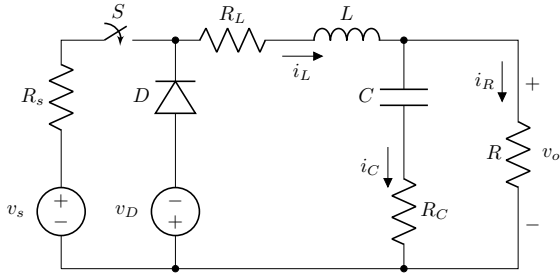


Fig. 3. circuito real do conversor Buck

Será adotado o modo de funcionamento de corrente contínua (CCM), em que a corrente no indutor nunca vai à zero e permite uma modelagem da planta independente da carga – dadas as devidas restrições como potência máxima e baixo ripple de tensão na saída. Ou seja, em CCM as chaves  $S$ ,  $D$  abrem e fecham de forma alternada, e sempre uma delas estará conduzindo corrente.

Para atender a essas condições e garantir que o sistema opere de acordo com a referência – sem atenuar as frequências que compõem o sinal de entrada e suportando a corrente máxima necessária para entregar potência à carga –, será feita uma análise do conversor Buck tanto por espaço de estados quanto por função de transferência. A primeira formulação será utilizada para o dimensionamento dos componentes do circuito, enquanto a segunda servirá para a análise da resposta em frequência.

## B. Modelo do Conversor por Espaço de Estados

Partindo do circuito da Figura 3, vamos modelar a planta do conversor Buck no domínio do tempo por Espaço de Estados, no formato do sistema de equações matriciais

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t) \end{cases}, \quad (1)$$

onde  $\mathbf{x} = [i_L \ v_o]^\top$ ,  $\mathbf{C} = [0 \ 1]$  e  $\mathbf{D} = 0$ .

Quando a chave  $S$  está aberta e  $D$  fechada, em  $t_{on}$ , a dinâmica do sistema é dada por

$$\begin{aligned} \mathbf{A}_{on} &= \begin{bmatrix} -\frac{(R_s+R_L)}{L} & -\frac{1}{L} \\ \frac{R(L-R_C C(R_s+R_L))}{LC(R+R_C)} & -\frac{(L+R_C RC)}{LC(R+R_C)} \end{bmatrix}, \\ \mathbf{B}_{on} &= \begin{bmatrix} \frac{v_s}{L} \\ \frac{R_C R v_s}{L(R+R_C)} \end{bmatrix}, \end{aligned}$$

e depois as chaves  $S$  e  $D$  comutam, em  $t_{off}$ , temos

$$\begin{aligned} \mathbf{A}_{off} &= \begin{bmatrix} -\frac{R_L}{L} & -\frac{1}{L} \\ \frac{R(L-R_L R_C C)}{LC(R+R_C)} & -\frac{(L+R_C RC)}{LC(R+R_C)} \end{bmatrix}, \\ \mathbf{B}_{off} &= \begin{bmatrix} -\frac{v_D}{L} \\ -\frac{R_C R v_D}{L(R+R_C)} \end{bmatrix}. \end{aligned}$$

Definindo as matrizes  $\mathbf{A} = \mathbf{A}_{on} \cdot dc + \mathbf{A}_{off} \cdot (1 - dc)$  e  $\mathbf{B}$  de forma análoga, e simplificando a expressão, temos

$$\mathbf{A} = \begin{bmatrix} -\frac{(dc \cdot R_s + R_L)}{L} & -\frac{1}{L} \\ \frac{R(L-R_C C(dc \cdot R_s + R_L))}{LC(R+R_C)} & -\frac{(L+R_C RC)}{LC(R+R_C)} \end{bmatrix}, \quad (2)$$

$$\mathbf{B} = \begin{bmatrix} \frac{dc \cdot v_s + (1-dc) \cdot v_D}{L} \\ \frac{R_C R (dc \cdot v_s + (1-dc) \cdot v_D)}{L(R+R_C)} \end{bmatrix}. \quad (3)$$

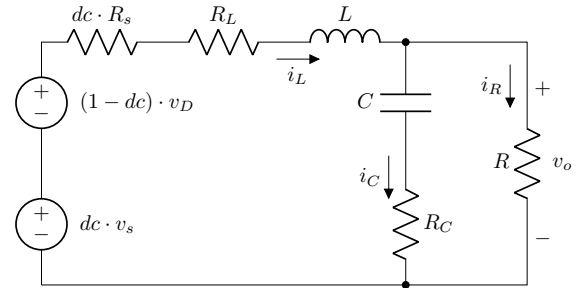


Fig. 4. circuito equivalente do conversor Buck

## C. Modelo do Conversor por Função de Transferência

Uma vez que o circuito ideal pode ser obtido ao zerar as resistências  $R_s$ ,  $R_L$ ,  $R_C$  do circuito real, será usado a formulação mais geral para deduzir a função de transferência. Para tal, considera-se  $v_i(t) = dc \cdot v_s$  e o objetivo é modelar a tensão de saída em função da tensão resultante da comutação das chaves, representada pelo  $v_i(t)$ .

$$G(s) = \mathcal{L} \left\{ \frac{v_o(t)}{dc \cdot v_s} \right\} = \frac{V_o(s)}{V_i(s)} \quad (4)$$

Para se obter as equações para o controle da corrente no indutor e da tensão na carga, pode-se aplicar a transformada de Laplace na equação matricial 1. Neste trabalho, entretanto, será implementado apenas o controle de tensão, então a dedução partirá da transformada do circuito equivalente (Figura 4).

Repare que, por haver  $dc$  multiplicando uma das variáveis de estado, conforme a matriz  $A$  da equação 2, é impossível separar linearmente  $V_o(s)$  de  $dc$ . Problema parecido acontece com a tensão de queda no diodo, uma constante que não aparece multiplicando nenhuma variável de estado. Assim, como aproximação da planta por um modelo linear, se faz necessário considerar  $R_s = 0$  e  $v_D = 0$ .

Resolvendo o sistema de equações resultante, obtém-se

$$G(s) = \frac{\left( \frac{R_C}{L} s + \frac{1}{LC} \right)}{\frac{R+R_C}{R} s^2 + \frac{R_L(R+R_C) + R R_C + L/C}{RL} s + \frac{R+R_L}{RLC}} \quad (5)$$

para o caso real, com as devidas considerações, e

$$G(s) = \frac{\frac{1}{LC}}{s^2 + \frac{1}{RC} s + \frac{1}{LC}} \quad (6)$$

para o caso do conversor ideal.

#### D. Funcionamento do Filtro Sallen-Key

Filtros são circuitos lineares projetados para permitir a passagem de determinadas frequências e atenuar outras, e são baseados em elementos reativos (capacitores e indutores), os quais possuem uma frequência natural de oscilação. A análise do comportamento dos filtros contínuos pode ser caracterizada pela sua resposta em frequência. Neste âmbito, geralmente é utilizada a sua função de transferência no domínio da transformada de Laplace.

A topologia Sallen-Key é uma representação da topologia de filtros eletrônicos utilizada para a implementação de filtros ativos de segunda ordem, que possuem uma maior atenuação às frequências de rejeição. O comportamento desse tipo de filtro passa-baixa é muito parecido com os de primeira ordem, com a diferença de que a variação na queda da amplitude com o aumento da frequência é mais acentuada.

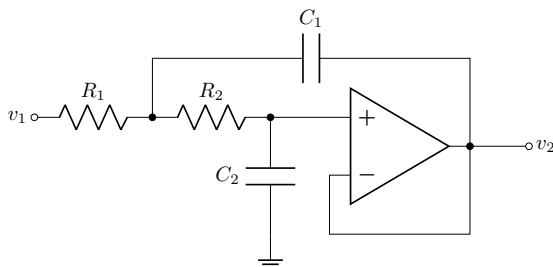


Fig. 5. circuito passa-baixas do filtro Sallen-Key

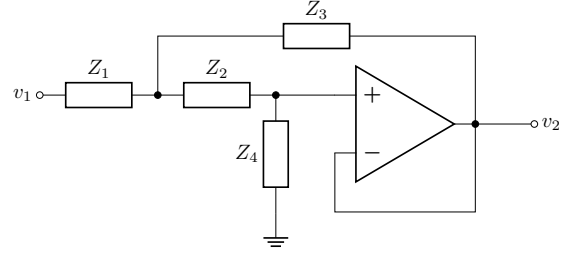


Fig. 6. circuito genérico do filtro Sallen-Key

#### E. Modelo do Filtro por Função de Transferência

A topologia genérica de um filtro Sallen-Key de ganho unitário é implementada através de um amplificador de ganho unitário como mostrado na Figura 6.

A partir da análise desse circuito, tendo como base um amplificador ideal, obtemos a equação 7 para esse circuito passa-baixas genérico.

$$H(s) = \frac{Z_3 Z_4}{Z_1 Z_2 + Z_3 (Z_1 + Z_2) + (Z_3 Z_4)} \quad (7)$$

Substituindo as impedâncias conforme o circuito da Figura 5 e desenvolvendo a equação, obtemos a função de transferência do filtro Sallen-Key, dada por

$$H(s) = \frac{\frac{1}{R_1 R_2 C_1 C_2}}{s^2 + \left( \frac{1}{R_1 C_1} + \frac{1}{R_2 C_1} \right) s + \frac{1}{R_1 R_2 C_1 C_2}}. \quad (8)$$

#### F. Modelo do Controlador PID

Para garantir que a planta siga o sinal de referência com erro nulo e resistência à perturbações, é necessário a adição de um controlador à planta. Para isso, será utilizado um PID (proporcional-integrador-derivativo) projetado em tempo contínuo e discretizado para ser implementado digitalmente.

Um controlador  $C(s) = U(s)/E(s)$  transforma o erro medido  $E(s)$  em uma ação de controle  $U(s)$  para a planta. Um controlador PID clássico é definido pela ação de controle

$$u(t) = K_p \left( e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (9)$$

que, em função de transferência, define-se

$$C_{PID}(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (10)$$

$$C_{PID}(s) = K_p + \frac{K_p}{T_i s} + K_p T_d s \quad (11)$$

$$C_{PID}(s) = K_p + \frac{K_i}{s} + K_d s \quad (12)$$

$$C_{PID}(s) = \frac{K_d s^2 + K_p s + K_i}{s} \quad (13)$$

em que  $K_p, K_i, K_d$  são os ganhos proporcional, integral e derivativo, obtidos a partir dos períodos de integração e derivação  $T_i, T_d$  respectivamente.

Para a escolha dos ganhos, existem diversas técnicas. Foi utilizado o método de ajuste de parâmetros de Ziegler-Nichols

como estimativa inicial, e depois foi feito um ajuste fino dos parâmetros por meio do resultado da simulação.

O método de Ziegler-Nichols consiste em obter o ganho crítico  $K_u$  que leva o sistema atinge à estabilidade crítica; fechar a malha com  $K_u$  e medir o período crítico  $T_u$  das oscilações características do sistema à beira da instabilidade; utilizar  $K_u, T_u$  para dimensionar os ganhos  $K_p, K_i, K_d$ .

Pode-se obter o valor de  $K_u$  por meio do critério de estabilidade de Routh-Hourwitz.

$$K_p = \alpha \cdot K_u \quad (14)$$

$$K_i = \beta \cdot \frac{K_u}{T_u} \quad (15)$$

$$K_d = \gamma \cdot K_u T_u \quad (16)$$

Os valores de  $\alpha, \beta, \gamma$  vêm das tabelas de projeto conforme o método (P/PI/PD/PID Clássico, *Pessen Integral Rule*, pouco *overshoot*). Partiu-se dos valores do PID Clássico 0,6 para  $\alpha$ , 1,2 para  $\beta$  e 0,075 para  $\gamma$ . O ganho

Para facilitar a calibragem, a implementação será feita a partir da equação 12, que representa o controlador como 3 parcelas distintas, e contará com um bloco de saturação no integrador para evitar que o sistema atinja a instabilidade devido ao erro acumulado por muito tempo.

#### G. Modelos em Tempo Discreto

Para permitir a implementação dos elementos de controle e acionamento digitalmente, será feita a discretização do controlador e sua transformação para o formato recursivo.

Também será implementado um filtro digital de segunda ordem, que seguirá as mesmas etapas de projeto do filtro analógico Sallen-Key (com exceção da escolha de componentes) e terá uma etapa adicional de discretização.

Serão utilizados dois métodos para a conversão das funções de transferência de tempo contínuo para tempo discreto: a aproximação *forward*, para o controlador,

$$s = \frac{z - 1}{T_s} \quad (17)$$

e a aproximação bilinear ou trapezoidal, para o filtro,

$$s = \frac{2(z - 1)}{T_s(z + 1)} \quad (18)$$

em que  $z$  é a variável complexa em tempo discreto e  $T_s = f_s^{-1}$  é o período de amostragem.

A conversão de um controlador de tempo discreto  $C(z)$  para a equação recursiva é dada abaixo.

$$C(z) = \frac{U(z)}{E(z)} = \frac{a_n z^n + \dots + a_0}{b_m z^m + \dots + b_0} \quad (19)$$

$$U(z) (b_m z^m + \dots + b_0) = E(z) (a_n z^n + \dots + a_0) \quad (20)$$

$$b_m z^m U(z) + \dots + b_0 U(z) = a_n z^n E(z) + \dots + a_0 E(z) \quad (21)$$

$$b_m u[k + m] + \dots + b_0 u[k] = a_n e[k + n] + \dots + a_0 e[k] \quad (22)$$

$$u[k] + \dots + \frac{b_0}{b_m} u[k - m] = \frac{a_n}{b_m} e[k - m + n] + \dots + \frac{a_0}{b_m} e[k - m] \quad (23)$$

$$u[k] = \frac{a_n}{b_m} e[k + n - m] + \dots + \frac{a_0}{b_m} e[k - m] - \dots - \frac{b_0}{b_m} u[k - m] \quad (24)$$

#### IV. PROJETO

Tanto a função de transferência do conversor Buck ideal,  $G(s)$ , quanto a do filtro Sallen-Key,  $H(s)$ , podem ser escritas na forma genérica de um filtro passa-baixas de segunda ordem, onde  $\omega_0$  representa a frequência de corte,  $\zeta$  o amortecimento e  $K$  o ganho da banda passante.

$$G_{PB}(s) = \frac{K \cdot \omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \quad (25)$$

Repare que, ao considerarmos apenas a componente oscilatória  $s = j\omega$  e tomarmos  $K = 1$ , temos a caracterização do filtro passa-baixas de ganho unitário: para frequências muito maiores que a frequência de corte, a resposta do sistema tende a atenuar o sinal, enquanto que para frequências muito abaixo o ganho é unitário. E para valores próximos à frequência de corte, o ganho é definido pelo amortecimento.

$$|G_{PB}(j\omega)|_{\omega \gg \omega_0} \rightarrow 0 \quad (26)$$

$$|G_{PB}(j\omega)|_{\omega \ll \omega_0} \rightarrow 1 \quad (27)$$

$$|G_{PB}(j\omega)|_{\omega \approx \omega_0} \rightarrow \frac{1}{2\zeta} \quad (28)$$

Uma vez que o funcionamento do circuito final será pautado por duas frequências, a do sinal de referência ( $f_r$ ) e a de amostragem e chaveamento ( $f_s$ ), a primeira etapa do projeto será a análise da resposta em frequência da planta, para o correto dimensionamento da banda passante.

##### A. Seleção da Banda Passante

O sinal de referência desse projeto é uma onda trapezoidal de amplitude de 6 V e nível médio de 12 V, com período de 20 ms, logo  $f_r = 50$  Hz. A frequência de amostragem do conversor A/D e chaveamento do *drive* é de  $f_s = 15$  kHz, resultando em 300 pontos por ciclo, 15 por milissegundo.

Além da frequência fundamental do sinal de referência  $f_r$ , ao fazer a expansão pela série de Fourier temos que esse sinal é composto por várias harmônicas. Definindo a frequência de corte  $f_{cb}$  do conversor Buck através da média geométrica das frequências limite, temos

$$f_{cb} = \sqrt{f_r \cdot f_s} \approx 866 \text{ Hz} . \quad (29)$$

Considerando que para frequências de 0 à  $f_{cb}$  temos um ganho praticamente unitário, faremos uma aproximação do sinal de entrada com as 17 primeiras harmônicas, gerando uma banda passante que vai até 850 Hz.

Para a frequência de corte  $f_{sk}$  do filtro Sallen-Key, foi usado como limites  $f_{cb}$  e  $f_s$ , uma vez que ele deve atenuar apenas as frequências de chaveamento. Foi utilizado como estimativa

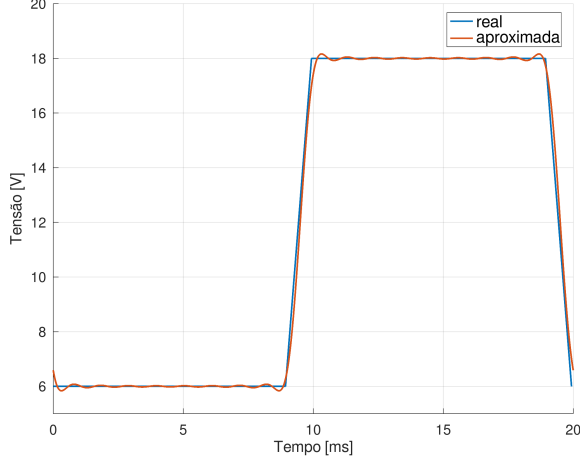


Fig. 7. onda trapezoidal real (azul) e aproximada (vermelha).

a média geométrica ponderada por um  $\alpha \in [0.5, 1]$ . Tomando um  $\alpha$  de 0.5549, temos

$$f_{sk} = \alpha \sqrt{f_{cb} \cdot f_s} \approx 2000 \text{ Hz} . \quad (30)$$

### B. Circuito do Conversor Buck

Das equações 6 e 25, e sabendo que  $\omega = 2\pi f$ , temos as relações para a frequência de corte

$$f_{cb} = \frac{1}{2\pi \sqrt{L_{cb} C_{cb}}} \quad (31)$$

e para o amortecimento do sistema

$$\zeta_{cb} = \frac{1}{2R} \sqrt{\frac{L_{cb}}{C_{cb}}} . \quad (32)$$

Visando uma rápida mudança de fase em torno da frequência de corte, será adotado um amortecimento de  $\zeta_{cb} = 0.46$  como ponto de partida para estimativa. Rearranjando os termos e resolvendo as equações, obtém-se os valores de 2.029 mH para o indutor e 16.65  $\mu$ F para o capacitor.

Aproximando os números para valores comerciais, temos  $L_{cb} = 2 \text{ mH}$  e  $C_{cb} = 2 \times 8.2 \mu\text{F}$ . O amortecimento permanece o mesmo, e a frequência de corte passa a valer  $f_{cb} = 879 \text{ Hz}$ , um pouco maior que a calculada pela média geométrica mas mantendo a qualidade da resposta em frequência.

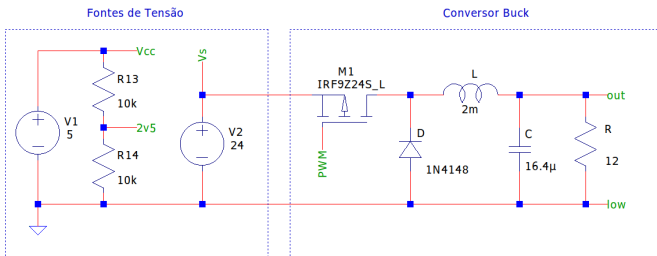


Fig. 8. esquema do conversor abaixador (Buck) com as fontes de tensão de entrada e alimentação dos componentes eletrônicos.

Os critérios para condução em modo contínuo e estabilidade na saída são atendidos conforme às equações de *ripple* de corrente e *ripple* de tensão (deduções na seção VI).

Partindo das equações 54 e 61, o CCM é garantido quando

$$\bar{i}_L - \Delta i_L > 0 \quad (33)$$

$$\frac{\Delta i_L}{\bar{i}_L} < 1 \quad (34)$$

$$\Delta i_{L\%} = \frac{R(1-dc)}{L_{cb} \cdot f_s} < 1 , \quad (35)$$

que nos leva a uma variação percentual de 20% na corrente do indutor escolhido para a tensão média de 12 V e de 15% para os valores máximo e mínimo.

E pela equação 64 podemos deduzir

$$\Delta v_{o\%} = \frac{\Delta v_o}{v_o} = \frac{(1-dc)}{4L_{cb}C_{cb} \cdot f_s^2} , \quad (36)$$

que deve se manter perto de zero para uma boa qualidade na tensão de saída. De fato, com os componentes do conversor a variação percentual na saída é de aproximadamente 1,7%.

### C. Circuito de Condicionamento de Sinais

Para que o conversor Buck seja controlado digitalmente, é necessário que o DSP receba as informações de tensão do conversor. A adequação do sinal medido é feita por meio de um circuito de condicionamento, efetuando a transdução, filtragem, ajustes de ganho e *offset*, e assim transformando o sinal real, na faixa 0 V – 24 V, em um sinal de 0 V – 3.3 V compatível com a entrada do A/D do microcontrolador escolhido (MSP430).

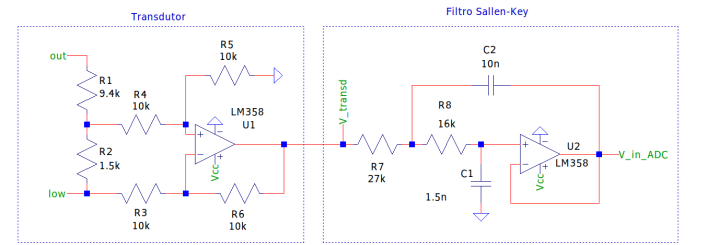


Fig. 9. esquema do circuito de condicionamento de sinais contendo transdutor e filtro passa-baixas de segunda ordem (Sallen-Key).

1) *Etapas de transdução*: será utilizado um divisor resistivo com alta impedância (no mínimo 10 k $\Omega$ ) como sensor e um circuito subtrator de ganho unitário para medição diferencial com remoção de ruído de modo comum. Ao ajustar o ganho do transdutor para  $K_t = \frac{3.3 \text{ V}}{24 \text{ V}} = 0,1375$  e utilizar amplificadores operacionais *rail-to-rail* alimentados na mesma faixa de operação do A/D, teremos uma economia no número de componentes e facilidade de projeto por manter o ganho unitário na maioria das etapas. E uma vez que o sinal a ser medido é esperado para variar entre 6 V e 18 V, o ganho de tensão adotado permite uma leitura de picos de até  $\pm 50\%$  em relação ao nível médio (12 V). Foram adotados os valores dos componentes  $R_{t1} = 9.4 \text{ k}\Omega$  e  $R_{t2} = 1.5 \text{ k}\Omega$ , referentes ao divisor resistivo, e  $R_{t3} = 10 \text{ k}\Omega$  para os resistores do subtrator.

$$v_t = \frac{R_{t2}}{R_{t1} + R_{t2}} \cdot \frac{R_{t3}}{R_{t3}} v_o \quad (37)$$

$$K_t = \frac{\Delta v_t}{\Delta v_o} = \frac{R_{t2}}{R_{t1} + R_{t2}} \quad (38)$$

2) *Etapas de filtragem*: o projeto dos componentes do filtro passa-baixas Sallen-Key de segunda ordem seguiu as equações 8 e 30, que especificam a frequência de corte desejada e a relação entre frequência natural, amortecimento e os componentes do circuito 5. Tomando  $\zeta_{sk} = 0,4$ , temos os valores  $R_{1sk} = 27 \text{ k}\Omega$ ,  $R_{2sk} = 16 \text{ k}\Omega$ ,  $C_{1sk} = 10 \text{ nF}$  e  $C_{2sk} = 1,5 \text{ nF}$ .

3) *Ajuste de ganhos*: devido ao uso de amplificadores *rail-to-rail*, à atenuação de  $K_t$  para a faixa de tensão do A/D e ao fato do sinal medido ser positivo, não é necessário a inserção de um circuito para reajuste de ganho e *offset*.

#### D. Filtro Digital

Para o filtro digital  $F(z)$ , será construído um filtro passa-baixas de segunda ordem de frequência de corte

$$f_{dg} = \alpha \sqrt{f_{sk} \cdot f_s} \approx 3040 \text{ Hz} \quad (39)$$

e amortecimento  $\zeta = 0,2$  (o  $\alpha$  é o mesmo do filtro analógico) e então realizado a discretização pela aproximação trapezoidal (equação 18).

Definindo  $F(z) = Y_f(z)/Y_q(z)$ , onde  $Y_q(z)$  é a saída medida e quantizada, e  $Y_f(z)$  é o respectivo valor filtrado, a equação recursiva que descreve o filtro é

$$\begin{aligned} y_f[k-0] = & +2,441204 \cdot 10^{-1} \cdot y_q[k-0] \\ & +9,764818 \cdot 10^{-1} \cdot y_q[k-1] \\ & +1,464723 \cdot 10^{-1} \cdot y_q[k-2] \\ & +9,764818 \cdot 10^{-1} \cdot y_q[k-3] \\ & +2,441204 \cdot 10^{-1} \cdot y_q[k-4] \\ & -1,283287 \cdot 10^{-1} \cdot y_f[k-1] \\ & -2,597685 \cdot 10^{-1} \cdot y_f[k-2] \\ & -6,696768 \cdot 10^{-1} \cdot y_f[k-3] \\ & -6,931951 \cdot 10^{-1} \cdot y_f[k-4] \end{aligned} \quad (40)$$

#### E. Controlador PID

Ao fechar a malha com  $G(s)$  e  $H(s)$  e aplicar o critério de Routh-Hurwitz, chegou-se no ganho crítico  $K_u = 1,5588$  e respectivo período crítico  $T_u = 750 \text{ ms}$ . Os ganhos do PID foram calibrados conforme os valores abaixo.

$$\alpha = 0,3 \rightarrow K_p = 0,46764 \quad (41)$$

$$\beta = 1,5 \rightarrow K_i = 3117,6 \quad (42)$$

$$\gamma = 0,05 \rightarrow K_d = 5,8455 \cdot 10^{-5} \quad (43)$$

Definindo os controladores proporcional  $C_p(z) = U_p(z)/E(z)$ , integral  $C_i(z) = U_i(z)/E(z)$  e derivativo  $C_d(z) = U_d(z)/E(z)$  de forma que a ação de controle total é gerada pela soma das ações de controle individuais, os

controladores discretizados pela aproximação *forward* são, no formato recursivo,

$$u_p[k-0] = +4,676 \cdot 10^{-1} \cdot e[k-0]; \quad (44)$$

$$u_i[k-0] = +2,078 \cdot 10^{-1} \cdot e[k-0] + u_i[k-1]; \quad (45)$$

$$u_d[k-0] = +8,768 \cdot 10^{-1} (e[k-0] - e[k-1]); \quad (46)$$

#### V. SIMULAÇÃO

Para validar os circuitos projetados, foram feitas análises computacionais com os programas LTspice XII versão 17.0.24.0, para o circuito eletrônico, e GNU Octave versão 6.2.0, para a modelagem matemática. As simulações do circuito utilizaram a diretiva `.tran 0 24m 0 100n`.

##### A. Análise em Malha Aberta

Inicialmente, serão analisados os modelos matemáticos gerados a partir da modelagem do caso ideal por função de transferência, conforme visto na seção III.

A resposta ao degrau das funções de transferências do conversor Buck e dos filtros de segunda ordem, presentes na figura 10, de onde é possível retirar os tempos de subida e acomodação do Sallen-Key (0.11 ms e 1 ms, respectivamente), assim como do Buck (0.65 ms e 2 ms). Além disso, temos o *overshoot* de 20% para o Buck e aproximadamente 25% para o Sallen-Key – números esperados, devido ao baixo amortecimento escolhido na etapa de projeto – e erro nulo em regime permanente. A resposta do filtro  $F(s)$  foi apresentada apenas para efeitos de comparação visual quanto ao seu *overshoot* e tempos de subida e acomodação.

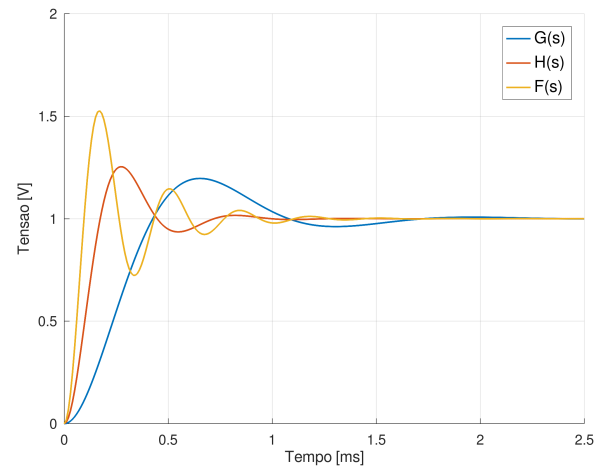


Fig. 10. Resposta ao degrau unitário do conversor Buck e dos filtros de segunda ordem.

Ao analisar a resposta em frequência através dos diagramas de Bode da figura 11, é possível observar que até 100 Hz a defasagem é desprezível e o ganho é unitário (o qual permanece até certa de 1 kHz), enquanto que para a frequência  $f_s = 15 \text{ kHz}$  temos uma boa atenuação.

Já ao simular o circuito, seguindo os esquemáticos das figuras 8 e 9, notou-se um erro de 250 mV para o nível mais

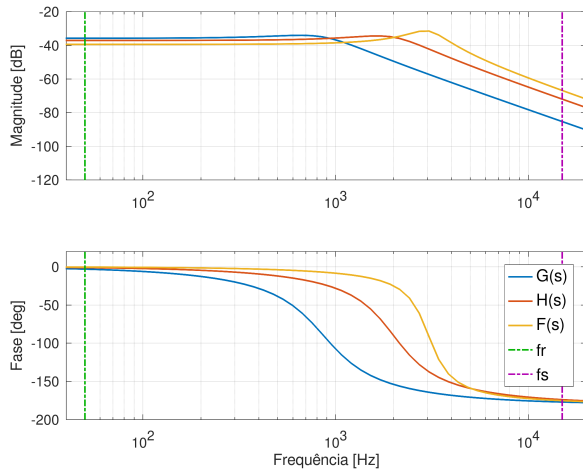


Fig. 11. diagramas de Bode do conversor Buck e filtros de segunda ordem, com frequência do sinal de referência (verde) e de chaveamento (roxo) .

baixo da referência e de 100 mV para o nível mais alto (figura 12). Acredita-se que o erro decorra da desconsideração da tensão de queda no diodo para linearizar o sistema, assim como as demais simplificações. As características transitórias correspondem às esperadas, e na figura 13 vemos que os *ripples* de tensão e corrente são baixos.

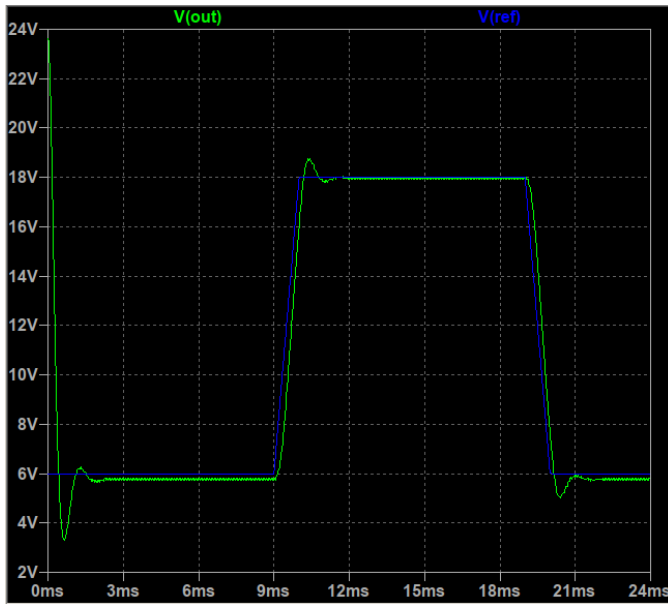


Fig. 12. tensão de saída na carga (verde) e tensão de referência (azul) no conversor Buck, ao longo de quase 1 ciclo e meio. Os sinais estão expressos na escala de 2 V/div e 3 ms/div.

Quanto ao CCS, houve uma correta resposta do circuito desenvolvido. É possível ver na figura 14 que a saída do filtro possui um *overshoot* levemente maior que o medido pelo transdutor, mas para efeitos de controle, um modelo que controle a saída “exagerada” de  $H(s)$  vai controlar bem a planta  $G(s)$ . E ao dar destaque maior ao sinal filtrado, na figura 15, fica nítido a remoção da alta frequência oriunda do chaveamento do MOSFET.

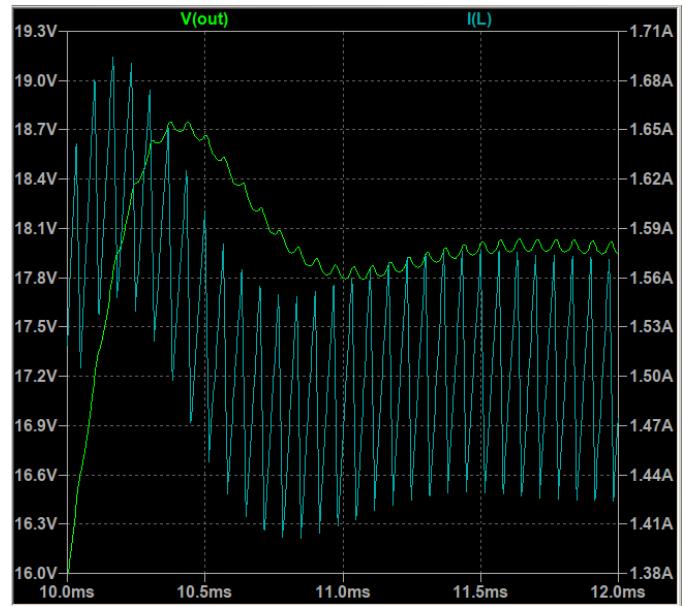


Fig. 13. *ripples* de tensão na carga (verde) e de corrente no indutor (ciano) no conversor Buck. Os sinais estão expressos nas escalas de 0.3 V/div, 0.3 A/div e 0.5 ms/div.

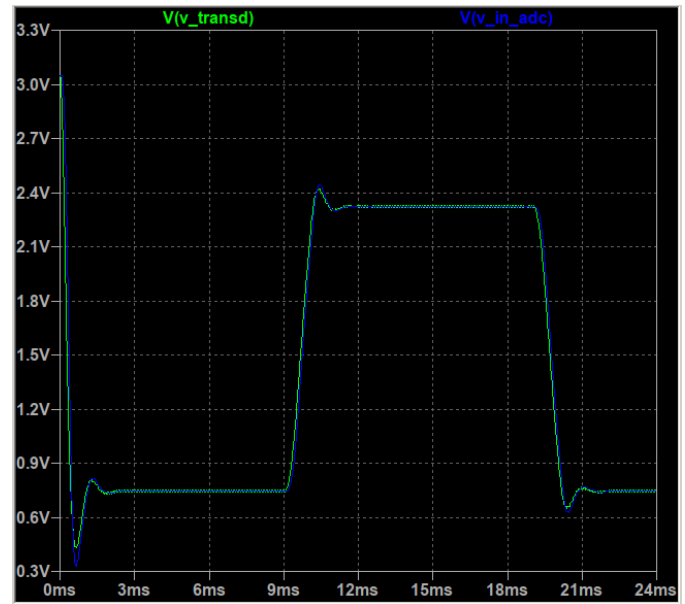


Fig. 14. tensão de saída no transdutor (verde) e tensão filtrada pelo Sallen-Key (azul) no circuito de condicionamento, ao longo de quase 1 ciclo e meio. Os sinais estão expressos nas escalas de 0.3 V/div e 3 ms/div.

### B. Análise em Malha Fechada

Para simular o sistema em malha fechada foi implementado a rotina de controle do DSP, contendo obtenção da entrada do ADC (valor quantizado), mudança de escala, filtragem digital, cômputo do erro, cálculo das ações de controle individuais e total com saturação no integrador ( $\pm v_s$ ) e no controlador completo ( $0e + v_s$ ) e cálculo do *duty cycle*.

Para efetuar a simulação da planta, foi construído o sinal de referência e feita a super-discretização dos elementos da planta (Buck e Sallen-Key) com 50 pontos por amostra e

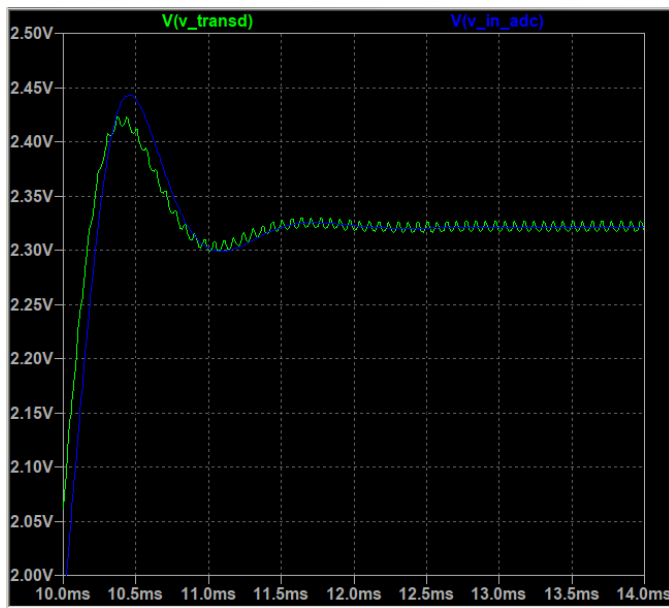


Fig. 15. tensão de saída no transdutor (verde) e tensão filtrada pelo Sallen-Key (azul) no circuito de condicionamento com destaque para a filtragem da banda passante. Os sinais estão expressos nas escalas de 50 mV/div e 0.5 ms/div.

aproximação *forward*. As rotinas de simulação são apresentadas no Apêndice.

Vale destacar que, das funções de transferência discretizadas, a única que utilizou o método trapezoidal foi o filtro digital, pois foi observado que a aproximação *forward* o levava a ter resultados insatisfatórios. Para os demais elementos, a aproximação *forward* foi suficiente e permitiu a simulação sem grande custo computacional.

Abaixo são apresentados os resultados da simulação.

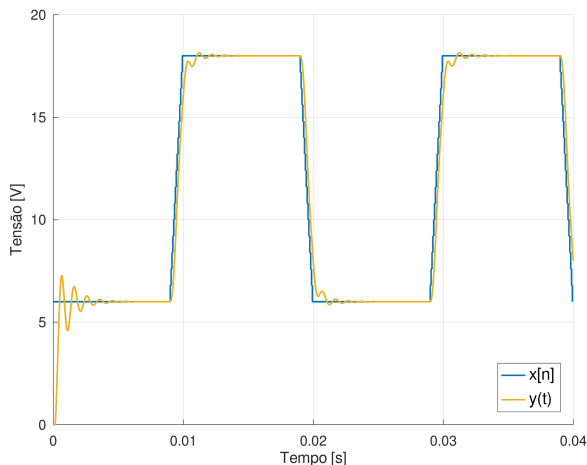


Fig. 16. tensões de entrada ( $u[n]$ ) e saída ( $y(t)$ ).

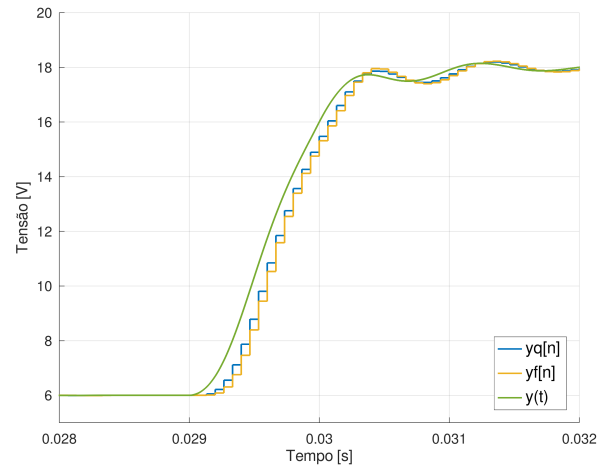


Fig. 17. saída real( $y(t)$ ), quantizada ( $y_q[n]$ ) e filtrada( $y_f[n]$ ).

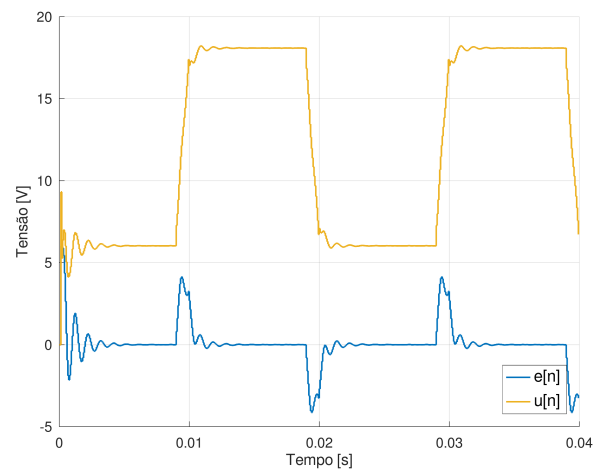


Fig. 18. erro medido ( $e[n]$ ) e ação de controle total ( $u[n]$ ).

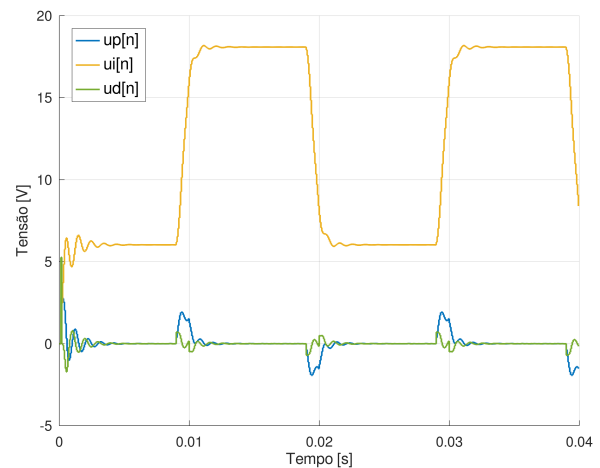


Fig. 19. ações de controle individuais do PID.

## VI. APÊNDICE

Aqui serão apresentados os códigos utilizados para a simulação dos modelos matemáticos e deduções adicionais.



### A. Expansão por Série de Fourier da Onda Trapezoidal

A onda trapezoidal apresentada na Figura 7 foi decomposta em exponenciais complexas e reconstruída com as 17 primeiras harmônicas, incluindo a frequência fundamental. Para efeitos de simplificação, a dedução foi feita na base de milissegundos. O código utilizado é dado abaixo.

```
1 clear;
2 clc;
3
4 T = 20;           % periodo fundamental (ms)
5 dt = 1e-3;       % intervalo entre pontos
6
7 w = 2*pi/T;      % frequencia fundamental
8 t = 0:dt:T-dt;   % array de tempo continuo
9 y = 12 + t*0;    % inicializacao com c0
10
11 n = 0:(20/300):20*(1 - 1/300); % amostras
12 x = [ 6*ones([1,134]) ...
13       (6:(12/15):18) ...
14       18*ones([1,134]) ...
15       (18:(-12/15):6) ];
16
17 for k=1:17
18     % coeficientes de k positivo
19     pk = j*k*w;
20     ck = (18*j*(exp(10*pk)-exp(-10*pk)) ...
21           - 12/(k*w)*(exp(-5*pk) - exp(-4*pk)) ...
22           * (exp(5*pk) - exp(-5*pk))) / (2*pi*k);
23     y = y + ck*exp(pk*t);
24     % coeficientes de k negativo
25     pk = -j*k*w;
26     ck = (18*j*(exp(10*pk) - exp(-10*pk)) ...
27           - 12/(-k*w)*(exp(-5*pk) - exp(-4*pk)) ...
28           * (exp(5*pk) - exp(-5*pk))) / (2*pi*-k);
29     y = y + ck*exp(pk*t);
30 end
31
32 % plot das curvas
33 set(0, 'defaultlinelength', 1.5);
34 fig = figure('position',[200 100 600 600]);
35 axis([0 20 5 20]);
36 xlabel('Tempo [ms]');
37 ylabel('Tensao [V]');
38 grid();
39 hold on;
40
41 plot(n,x, t,y);
42 lbl = legend('real','aproximada');
43 set(lbl,'fontsize',14);
44 set(gca, 'fontsize', 12);
45
46 print(fig, 'onda_trapezoidal.png', '-r300');
47 hold off;
```

### B. Análise das Plantas em Malha Aberta

Os gráficos de análise das funções de transferência  $G(s)$  e  $H(s)$  da seção V fazem parte de um *script* único. Para facilitar o entendimento, o código será apresentado por partes.

A inicialização, definição das variáveis globais, funções úteis e criação dos modelos por função de transferência  $G(s), H(s), F(s)$  é feita da seguinte forma.

```
1 clear all;           % limpeza das variaveis
2 close all;          % fecha janelas abertas
3 clc;                % limpeza do console
4 pkg load control;
```

```
6 set(0, 'defaultlinelength', 1.5);
7
8 xline = @(xval, varargin) ...
9         line([xval xval], ylim, varargin{:});
10
11 function retval = manual_c2d(G, tsamp, method='f')
12     z = tf('z', tsamp);
13
14     % aproximacao forward/backward/trapezoidal
15     if method=='f'      s = (z-1)/tsamp;
16     elseif method=='b'  s = 1/tsamp * (z-1)/z;
17     else
18         s = 2/tsamp * (z-1)/(z+1);
19     end
20
21     numz = 0;
22     denz = 0;
23     [num den] = tfdata(G,'v');
24
25     for k=1:length(num)
26         numz = numz*s + num(k);
27     end
28     for k=1:length(den)
29         denz = denz*s + den(k);
30     end
31
32     retval = minreal(numz/denz);
33 end
34
35 function discrete_rec_eq(Gz, resolution=3, input='e', output='r')
36     [a b] = tfdata(Gz,'v'); % coeficientes
37     n = length(a)-1;        % ordem do numerador
38     m = length(b)-1;        % ordem do denominador
39
40     recstr = sprintf(['\n' output ' (k-0) = ']);
41     numstr = ['%. ' int2str(resolution) ' e'];
42
43     format = ['...\n'
44               '| numstr ' * ' input ' (k-i)'];
45     for k=0:n
46         if b(k+1) != 0
47             recstr = [recstr sprintf(format, a(k+1)/b(1), k)];
48         end
49     end
50
51     format = ['...\n'
52               '| numstr ' * ' output ' (k-i)'];
53     for k=1:m
54         if b(k+1) != 0
55             recstr = [recstr sprintf(format, -b(k+1)/b(1), k)];
56         end
57     end
58
59     printf('%s;\n', recstr);
60 end
61
62 function retval = sat(value, min_v, max_v)
63     % bloco de saturacao
64     retval = min(max(value, min_v), max_v);
65 end
66
67 f_r = 50;           % frequencia do sinal de entrada
68 f_s = 15000;        % frequencia de chaveamento
```

```
1 f_cb = sqrt(f_r*f_s)
2 w_cb = 2*pi*f_cb;   % freq. natural
3 z_cb = 0.46;        % amortecimento
4 num = [0 0 w_cb^2];
5 den = [1 2*w_cb*z_cb w_cb^2];
6 G = tf(num, den)    % Buck
7
8 alpha = 0.5549;
9 f_sk = alpha*sqrt(f_s*f_cb)
```

```

10 w_sk = 2*pi*f_sk;           % freq. natural
11 z_sk = 0.4;                 % amortecimento
12 num = [0 0 w_sk^2];
13 den = [1 2*w_sk*z_sk w_sk^2];
14 H = tf(num, den)           % Sallen-Key
15
16 f_dg = alpha*sqrt(f_s*f_sk)
17 w_dg = 2*pi*f_dg;           % freq. natural
18 z_dg = 0.2;                 % amortecimento
19 num = [0 0 w_dg^2];
20 den = [1 2*w_dg*z_dg w_dg^2];
21 F = tf(num, den)           % filtro digital

```

Os diagramas de Bode de magnitude e fase da Figura 11 foram gerados com a função pré-definida `bode(...)` e alteradas para melhor legibilidade.

```

1 W = logspace(1, 6, 100);
2 [Mg, Pg, _] = bode(G,W);
3 [Mh, Ph, _] = bode(H,W);
4 [Mf, Pf, _] = bode(F,W);
5 W = W/(2*pi);
6
7 fig = figure();
8 hold on;
9
10 subplot(2,1,[1]);
11 semilogx(W, 20*log10(1/Mg), ...
12          W, 20*log10(1/Mh), ...
13          W, 20*log10(1/Mf));
14 xlim([4e1 2e4]);
15 ylim([-120 -20]);
16 xline(f_r, 'linestyle', '-.', ...
17       'color', [0 0.7 0]);
18 xline(f_s, 'linestyle', '-.', ...
19       'color', [.7 0 .7]);
20 ylabel('Magnitude [dB]');
21 set(gca, 'fontsize', 12);
22 grid on;
23
24 subplot(2,1,2);
25 semilogx(W, Pg, W, Ph, W, Pf);
26 xlim([4e1 2e4]);
27 ylim([-200 20]);
28 xline(f_r, 'linestyle', '-.', ...
29       'color', [0 0.7 0]);
30 xline(f_s, 'linestyle', '-.', ...
31       'color', [.7 0 .7]);
32 ylabel('Fase [deg]');
33 xlabel('Frequencia [Hz]');
34 lbl = legend('G(s)', 'H(s)', ...
35            'F(s)', 'fr', 'fs');
36 set(lbl, 'fontsize', 14);
37 set(gca, 'fontsize', 12);
38 grid on;

```

A exibição da resposta ao degrau unitário, presente na Figura 10, é semelhante ao dos diagramas de Bode.

```

1 tdelta = 1e-5; % passo de tempo
2 tfinal = 2.5e-3; % tempo final
3 [xG, t] = step(G, tfinal, tdelta);
4 [xH, _] = step(H, tfinal, tdelta);
5 [xF, _] = step(F, tfinal, tdelta);
6 t = t*1000; % mudanca de unidade
7
8 fig = figure();
9 hold on;
10 grid on;
11 plot(t, xG);
12 plot(t, xH);
13 plot(t, xF);

```

```

14 ylabel('Tensao [V]');
15 xlabel('Tempo [ms]');
16 lbl = legend('G(s)', 'H(s)', 'F(s)');
17 set(lbl, 'fontsize', 14);
18 set(gca, 'fontsize', 12);

```

A etapa de discretização e super-discretização dos modelos é apresentada abaixo.

```

1 VCC = 3.3;
2 VSOURCE = 24;
3 Ka = VSOURCE; % ganho do acionamento
4 Kt = VCC/VSOURCE; % ganho do transdutor
5
6 NC = 2; % numero de ciclos
7 T = 1/f_r; % periodo da referencia
8 N = f_s/f_r; % amostras por periodo
9 TT = NC*T; % duracao da simulacao
10 NT = NC*N; % amostras na simulacao
11
12 NP = 50; % resolucao da superdiscretizacao
13 dn = T/N; % intervalo entre amostras
14 dt = dn/NP; % intervalo de superdiscretizacao
15 n = 0:dn:(NC*T-dn); % instantes de amostragem
16 t = 0:dt:(NC*T-dt); % tempo superdiscretizado
17
18 % discretizacao e superdiscretizacao
19 Cz = manual_c2d(C, dn); % controlador PID
20 Fz = manual_c2d(F, dn, 't'); % filtro digital
21 Gz = manual_c2d(G*Ka, dt); % conversor buck
22 Hz = manual_c2d(H*Kt, dt); % filtro analogico
23 PIDz = [
24     manual_c2d(tf(Kp), dn) % proporcional
25     manual_c2d(Ki/tf('s'), dn) % integral
26     manual_c2d(Kd*tf('s'), dn) % derivativo
27 ];
28
29 % equacoes recursivas
30 prec = 3; % numero de casas decimais (precisao)
31 discrete_rec_eq(PIDz(1,:), prec, 'e', 'up');
32 discrete_rec_eq(PIDz(2,:), prec, 'e', 'ui');
33 discrete_rec_eq(PIDz(3,:), prec, 'e', 'ud');
34 discrete_rec_eq(Fz, 2*prec, 'yq', 'yf');
35 discrete_rec_eq(Gz, 2*prec, 'dc', 'y');
36 discrete_rec_eq(Hz, 2*prec, 'y', 'ym');

```

Por fim, a rotina de simulação em tempo discreto.

```

1 NBITS = 10;
2 RES_ADC = 2^NBITS;
3
4 % variaveis discretas
5 x = zeros(NT,1); % referencia
6 e = zeros(NT,1); % erro
7 u = zeros(NT,1); % acao de controle
8 up = zeros(NT,1); % acao proporcional
9 ui = zeros(NT,1); % acao integradora
10 ud = zeros(NT,1); % acao derivativa
11 yq = zeros(NT,1); % medicao quantizada
12 yf = zeros(NT,1); % medicao filtrada
13 dc = zeros(NT,1); % duty cycle
14
15 % variaveis pseudo-continuas
16 y = zeros(NT*NP,1); % tensao de saida
17 ym = zeros(NT*NP,1); % saida medida
18
19 % geracao do sinal de referencia
20 for k=1:NT
21     idx = mod(k, N);
22     if idx < (9e-3/dn)
23         x(k) = 6;
24     elseif idx < (10e-3/dn)
25         x(k) = -102 + (12/1e-3)*(idx*dn);

```

```

26 elseif idx < (19e-3/dn)
27     x(k) = 18;
28 else
29     x(k) = +246 - (12/1e-3)*(idx*dn);
30 end
31 end
32
33 %plot(n,x);
34 %ylim([0 24]);
35 %grid on;
36
37 % construcao das curvas
38 for k=3:NT
39     % Etapa de amostragem e quantizacao
40     adc_in = round(ym((k-1)*NP) * RES_ADC);
41     yq(k) = adc_in / (Kt*RES_ADC);
42     % Filtro digital e erro
43     yf(k) = ...
44         +2.441204e-01 * yq(k-0) ...
45         +9.764818e-01 * yq(k-1) ...
46         +1.464723e+00 * yq(k-2) ...
47         +9.764818e-01 * yq(max(k-3,1)) ...
48         +2.441204e-01 * yq(max(k-4,1)) ...
49         -1.283287e+00 * yf(k-1) ...
50         -2.597685e-01 * yf(k-2) ...
51         -6.696768e-01 * yf(max(k-3,1)) ...
52         -6.931951e-01 * yf(max(k-4,1));
53     e(k) = x(k) - yf(k);
54     % Acoes de controle individuais
55     up(k) = +4.676e-01 * e(k);
56     ui(k) = +2.078e-01 * e(k) + ui(k-1);
57     ud(k) = +8.768e-01 * (e(k) - e(k-1));
58     % Acao de controle conjunta com saturacao
59     ui(k) = sat(ui(k), -VSOURCE, +VSOURCE);
60     u(k) = sat(up(k) + ui(k) + ud(k),
61         0, VSOURCE);
62     % Duty cycle
63     dc(k) = u(k)/Ka;
64     % Simulacao da planta real
65     for j=1:NP
66         kc = (k-1)*NP + j;
67         % Resposta da planta
68         y(kc) = ...
69             +1.263309e-03 * dc(k-0) ...
70             +1.993325e+00 * y(kc-1) ...
71             -9.933779e-01 * y(kc-2);
72         % Saida do Sallen-Key
73         ym(kc) = ...
74             +3.860035e-05 * y(kc-0) ...
75             +1.986596e+00 * ym(kc-1) ...
76             -9.868767e-01 * ym(kc-2);
77         % Saturacao do Amp-Op
78         ym(kc) = sat(ym(kc), 0, VCC);
79     end
80 end

```

### C. Modelagem do Conversor por Eletrônica de Potência

Considere o período de chaveamento  $T_s = f_s^{-1}$  e um *duty cycle*  $0 < dc < 1$ , e que o transitório do circuito aconteceu no intervalo  $-\infty < t < 0$ , tal que em  $t \geq 0$  o circuito esteja em regime estacionário.

Analisando o circuito pela Lei das Tensões de Kirchhoff em  $t_{on} = dc \cdot T_s$ , quando a chave  $S$  está fechada e o diodo  $D$  está inversamente polarizado, a tensão no indutor é dada

$$v_L = v_s - v_o, \quad (47)$$

e em  $t_{off} = (1 - dc)T_s$ , quando a chave  $S$  abre, o diodo é diretamente polarizado e começa a conduzir, logo a tensão é apenas

$$v_L = -v_o. \quad (48)$$

Considerando a energia armazenada no indutor dada pela fórmula  $E = \frac{1}{2}Li_L^2$ , no estado estacionário toda a energia excedente armazenada durante  $t_{on}$  é descarregada em  $t_{off}$ . Para manter a situação de equilíbrio

$$\Delta E_{on} = \Delta E_{off} \iff \Delta i_{L_{on}} + \Delta i_{L_{off}} = 0 \quad (49)$$

$$\Rightarrow \frac{1}{L} \int_0^{t_{on}} v_L(t) dt + \frac{1}{L} \int_{t_{on}}^{T_s} v_L(t) dt = 0. \quad (50)$$

Como  $T_s = t_{on} + t_{off}$  e, em regime estacionário, a corrente no indutor é cíclica, ou seja,  $i_L(t) = i_L(t + T_s)$ , a equação 50 pode ser reescrita na forma

$$\frac{1}{L} \int_0^{t_{on}} v_{L_{on}}(t) dt + \frac{1}{L} \int_0^{t_{off}} v_{L_{off}}(t) dt = 0. \quad (51)$$

Substituindo as equações 47 e 48, respectivamente e simplificando em torno de  $dc$ , temos a relação entre a *duty cycle* e as tensões de entrada e saída.

$$\frac{(v_s - v_o)dc \cdot T_s}{L} - \frac{v_o(1 - dc)T_s}{L} = 0 \quad (52)$$

$$dc = \frac{v_o}{v_s} \quad (53)$$

Além disso, visto que na condição de equilíbrio da equação 49 as variações de corrente no indutor são iguais, podemos definir o *ripple* de corrente

$$\Delta i_L = \Delta i_{L_{on}} = -\Delta i_{L_{off}} = \frac{v_o(1 - dc)T_s}{L} \quad (54)$$

Analisando agora pela Leis das Correntes de Kirchhoff, em qualquer instante  $t$  a relação abaixo é válida

$$i_L(t) = i_R(t) + i_C(t) \quad (55)$$

$$\frac{1}{L} \int_{-\infty}^t v_L(\tau) d\tau = \frac{v_o(t)}{R} + i_C(t). \quad (56)$$

No estado estacionário, quando a tensão de saída é praticamente constante e a corrente  $i_L$  é cíclica em torno de um valor médio, temos

$$\int_{-\infty}^t v_L(\tau) d\tau = \int_{-\infty}^0 v_L(\tau) d\tau + \int_0^t v_L(\tau) d\tau \quad (57)$$

$$\int_{-\infty}^t v_L(\tau) d\tau = \bar{i}_L + \int_0^t v_L(\tau) d\tau \quad (58)$$

sendo  $\bar{i}_L$  a corrente média no indutor e  $t = nT + \Delta t$ , onde  $n \in \mathbb{N}$  e  $\Delta t \leq T$ .

Repare que a integral de 0 à  $t$  pode ser quebrada em  $n$  integrais ao longo de um período completo, que resultam em zero, e uma integral de  $nT_s$  à  $nT_s + \Delta t$ , que pode ser calculada no intervalo 0 à  $\Delta t$ .

$$\int_0^t v_L(\tau) d\tau = \int_0^{\Delta t} v_L(\tau) d\tau \quad (59)$$

Substituindo e integrando durante o período  $t_{on}$  em que a chave semicondutora  $S$  está fechada, ou, de forma análoga, durante período  $t_{off}$  em que ela está aberta, nos leva à

$$\bar{i}_L + \int_0^{\Delta t} v_L(\tau) d\tau = \frac{v_o}{R} + i_C \quad (60)$$

$$\overline{i_L} \pm \Delta i_L = \frac{v_o}{R} + i_C \quad (61)$$

de onde saem duas relações: a corrente média no indutor é a corrente do resistor; e a corrente máxima no capacitor é o *ripple* de corrente, cuja média ao longo do período é zero.

Observe que a corrente  $i_C$  varia entre  $-\Delta i_L$  e  $+\Delta i_L$  durante  $t_{on}$  e  $+\Delta i_L$  e  $-\Delta i_L$  durante  $t_{off}$ , passando pelo zero na metade de cada intervalo.

Por fim, ao analisar a corrente do capacitor durante metade do período, conforme a relação da Equação 61, temos

$$C \frac{dv_o}{dt} = \Delta i_L \quad (62)$$

$$\frac{2\Delta v_o}{T_s/2} = \frac{v_o(1-dc)T_s}{LC}, \quad (63)$$

de onde pode-se definir o *ripple* de tensão

$$\Delta v_o = \frac{v_o(1-dc)T_s^2}{4LC}. \quad (64)$$

#### D. Circuito Gerador de PWM

Para efeitos de simulação, foi montado um circuito para gerar o sinal de PWM responsável pelo acionamento do MOSFET (chave  $S$ ), cujo esquemático é apresentado a seguir. O circuito contém um sinal de referência, um divisor resistivo para adequá-lo à faixa de operação dos amplificadores operacionais, uma fonte de onda quadrada, um comparador e um *drive* para aumentar o nível de tensão.

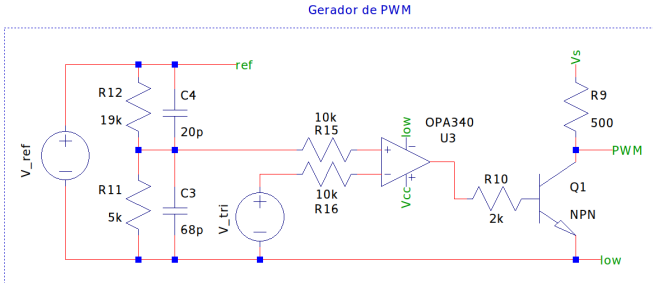


Fig. 20. esquemático do circuito gerador de PWM.