

Projeto Conversor Buck

1st Welliton Jhonathan Leal Babinski
Departamento de Informática
Universidade Tecnológica Federal do Paraná
Pato Branco, Brasil
welliton@alunos.utfpr.edu.br

2nd Emanuel Cristian de Cesaro
Departamento de Informática
Universidade Tecnológica Federal do Paraná
Pato Branco, Brasil
emanuelcesaro@alunos.utfpr.edu.br

Abstract—Este artigo é referente ao projeto proposto na disciplina de controle digital, tem como objetivo apresentar a modelagem, o projeto, desenvolvimento teórico e implementação prática por meio de simulação de um circuito conversor buck malha aberta e posteriormente de malha fechada realizando o controle com um microcontrolador.

Index Terms—buck converter, control systems, signal processing, microcontrollers

I. INTRODUÇÃO

O conversor buck conhecido como conversor abaixador de tensão, consiste em um gerador de pulsos e um filtro passivo. Ambos se combinam para converter uma tensão de entrada CC¹ aplicada em uma tensão CC de saída com um valor inferior. A técnica de usar uma fonte de impulso e um filtro LC² na saída permite a transferência de energia com alta eficiência. Ou seja, pouca energia é dissipada do próprio conversor. Para regular a tensão de saída, uma fonte de tensão e um amplificador de erro são adicionados ao circuito. [1] A figura 1 apresenta o diagrama de blocos do sistema em malha fechada de um conversor Buck, esse é o escopo com a dimensão global do nosso projeto, o qual é formado por quatro elementos principais, o próprio conversor, um circuito condicionador de sinais, um microcontrolador para realizar o controle do chaveamento e sinal de saída, e por fim um drive para o chaveamento.

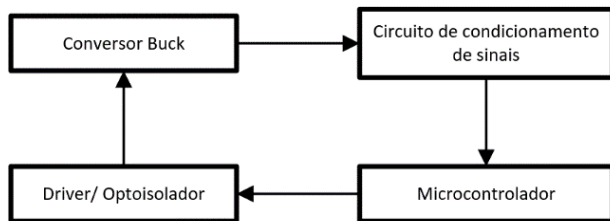


Fig. 1. Diagrama de blocos de um conversor Buck em malha fechada

A figura 2 nos apresenta uma representação por diagrama de blocos do seu sistema em (a) e com é a cadeia de processamento desse sinal de entrada, equanto em (b), esse

sistema é representado utilizando impedâncias.

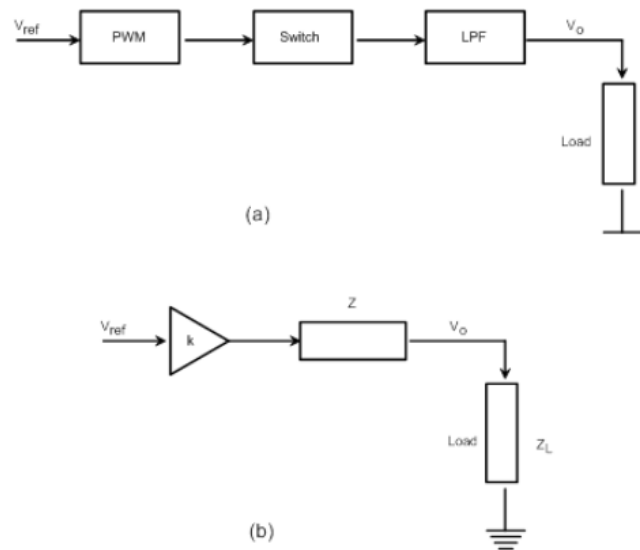


Fig. 2. Diagrama de blocos e fluxo do sinal de um conversor Buck

A figura 3 trás o esquemático do circuito de um conversor Buck que se associa um filtro passa-baixa e posteriormente um comutador. O diodo fornece um caminho para a corrente do indutor quando a chave é aberta e é polarizado reversamente quando a chave está fechada. [1] Entre as vantagens do uso de um conversor Buck estão a boa imunidade a ruído, máxima flexibilidade na escolha de valores de indutor e tipos de capacitores, garantia de estabilidade do sistema, elevado ganho em baixas frequências e garantia de margem de fase. Dentre a grande gama de aplicações deste conversor, podemos citar algumas principais que se destacam, sendo elas:

- sistemas de alimentação de energia;
- fontes controladas para computadores;
- eletroeletrônicos;
- veículos elétricos;
- equipamentos e esteiras industriais;
- acionamento de motores para drones;
- refrigeração;
- processos físico-químicos e de automação como temperatura, fluxo, iluminação, velocidade, posição, entre tantas outras aplicações de precisão;

¹CC: Corrente Contínua

²LC: Indutor e Capacitor

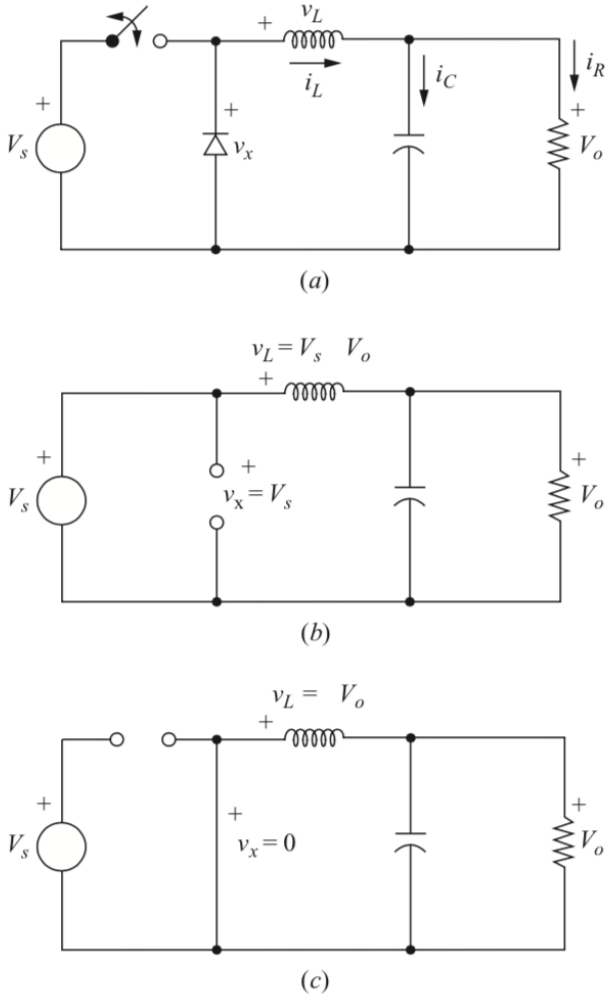


Fig. 3. (a) Circuito conversor Buck; (b) Circuito equivalente para o comutador fechado; (c) Circuito equivalente para o comutador aberto
 Fonte: HART [1]

II. PROJETO MALHA ABERTA

Nesta seção serão apresentados a modelagem e projeto do conversor buck, o projeto do conversor buck, a função de transferência do conversor, os circuitos, simulações e códigos do buck e da etapa de condicionamento de sinais, e componentes necessários para montagem do circuito do físico.

A. MODELAGEM DO CONVERSOR BUCK

Em toda a modelagem do conversor Buck presente nesta seção usou-se como base o desenvolvimento matemático de HART, 2010 [1].

Começamos analisando a figura 3(a) e considerando um filtro passa-baixa ideal, a tensão de saída é a média da tensão de entrada para o filtro. Para a situação do comutador estar fechado (figura 3(b)), a entrada do filtro v_x é igual V_s , quando o comutador está aberto $v_x = 0$. Se o comutador está sendo fechado periodicamente, então o duty cycle D é adicionado a tensão média de entrada, sendo representada por $V_s D$.

Para a análise, supõe-se que o diodo permanece polarizado diretamente durante o tempo que o comutador está aberto, implicando em uma corrente no indutor positiva, ou seja, corrente contínua. Em caso de corrente descontínua a corrente do indutor retorna para zero em cada período. A segunda parte da análise do conversor Buck é examinar a tensão e a corrente no indutor. Esta análise é útil para projetar o filtro e realizar a análise dos circuitos posteriormente. Os conversores Buck e os conversores CC-CC possuem as seguintes propriedades quando operam em regime permanente:

- 1) A corrente do indutor é periódica:

$$i_L(t + T) = i_L(t)$$

- 2) A tensão média do indutor é zero:

$$V_L = \frac{1}{T} \cdot \int_t^{t+T} V_L(\lambda) d\lambda = 0$$

- 3) A corrente média no capacitor é zero:

$$I_C = \frac{1}{T} \cdot \int I_C(\lambda) d\lambda = 0$$

- 4) A potência fornecida pela fonte é a mesma potência fornecida pela carga. Para componentes não ideais, a fonte também fornece perdas.

$$P_s = P_o \text{ (ideal)} \quad P_s = P_o + \text{perdas} \text{ (n\~ao ideal)}$$

Para iniciar a análise do circuito conversor Buck da figura 3, devemos assumir algumas afirmações:

- 1) O circuito está operando em estado estacionário.
- 2) A corrente do indutor é contínua e sempre positiva.
- 3) Considera-se que o capacitor possui um valor elevado, e a tensão de saída é mantida constante para V_o .
- 4) O período de comutação é T , a chave está fechada para o tempo DT e aberta para o tempo $(1 - D)T$.
- 5) Os componentes são considerados ideais.

Primeiramente a análise será feita considerando o circuito com a chave fechada (figura 3(b)) e depois para o circuito com a chave aberta (figura 3(c)).

1) *Análise para a chave fechada:* Quando a chave está fechada no circuito Buck (figura 3(b)) o diodo é polarizado inversamente, e a tensão no indutor é:

$$v_L = V_s - V_o = L \cdot \left(\frac{di_L}{dt} \right)$$

$$\frac{di_L}{dt} = \frac{V_s - V_o}{L}$$

A derivada da corrente é uma constante positiva, a mesma aumenta linearmente. A variação da corrente no indutor é apresentada pelo modelo matemático a seguir:

$$\frac{di_L}{dt} = \frac{\Delta i_L}{\Delta t} = \frac{\Delta i_L}{DT} = \frac{V_s - V_o}{L}$$

$$(\Delta i_L)_{fechado} = \left(\frac{V_s - V_o}{L} \right) DT$$

2) *Análise para a chave aberta:* Quando a chave está aberta o diodo permanece polarizado diretamente, e a tensão no indutor é:

$$v_L = -V_o = L \cdot \left(\frac{di_L}{dt} \right)$$

$$\frac{di_L}{dt} = \frac{-V_o}{L}$$

A derivada da corrente no indutor é uma constante negativa, e a corrente diminui como mostrado na figura 3(c). A mudança de corrente no indutor está demonstrada a seguir:

$$\frac{\Delta i_L}{\Delta t} = \frac{\Delta i_L}{(1-D)T} = \frac{-V_0}{L}$$

$$(\Delta i_L)_{aberto} = \left(\frac{-V_0}{L}\right)(1-D)T$$

A operação em estado estacionário requer que a corrente no indutor no final do ciclo de comutação seja a mesma que no começo, o que significa que a mudança na corrente do indutor ao longo do tempo seja zero.

$$(\Delta i_L)_{fechado} + (\Delta i_L)_{aberto} = 0 \quad (1)$$

Usando as equações para corrente no indutor fechado e aberto, temos:

$$\left(\frac{V_s - V_0}{L}\right)(DT) - \left(\frac{V_0}{L}\right)(1-D)T = 0$$

$$\boxed{V_0 = V_s D} \quad (2)$$

O conversor Buck produz uma tensão de saída menor ou igual à entrada. Como a tensão média do indutor é zero para operação periódica.

$$V_L = (V_s - V_0)DT + (-V_0)(1-D)T = 0$$

A corrente média do indutor deve ser a mesma corrente média na carga do resistor, desde que a corrente média do capacitor deve ser zero para a operação estável:

$$\boxed{I_L = I_R = V_0/R} \quad (3)$$

Desenvolvendo as equações da corrente no indutor máxima e mínima, temos:

$$I_{max} = I_L + \frac{\Delta i_L}{2} = \frac{V_0}{R} + \frac{1}{2} \cdot \left[\frac{V_0}{L} \cdot (1-D)T\right] = V_0 \left(\frac{1}{R} + \frac{1-D}{2Lf}\right)$$

$$I_{min} = I_R - \frac{\Delta i_L}{2} = \frac{V_0}{R} - \frac{1}{2} \cdot \left[\frac{V_0}{L} \cdot (1-D)T\right] = V_0 \left(\frac{1}{R} + \frac{1-D}{2Lf}\right)$$

A última equação encontrada pode ser usada para encontrar o valor do indutor e da frequência de chaveamento. Considerando $I_{min} = 0$, encontramos as seguintes expressões matemáticas:

$$I_{min} = 0 = V_0 \left(\frac{1}{R} - \frac{1-D}{2Lf}\right)$$

$$L_{min} f = \frac{(1-D)R}{2}$$

Para corrente contínua:

$$L_{min} = \frac{(1-D)R}{2f}$$

Onde $f = 1/T$ é a frequência de chaveamento e L_{min} é o valor de indutância mínima para a corrente continua, na prática este valor deve ser maior. Para determinar o valor de indutância específica para um indutor pico a pico, associasse a equação de I_{min} com $V_0 = V_s D$

$$\Delta i_L = \left(\frac{V_s - V_0}{L}\right)DT = \left(\frac{V_s - V_0}{Lf}\right)D = \left(\frac{V_0(1-D)}{Lf}\right)$$

$$\boxed{L = \left(\frac{V_s - V_0}{\Delta i_L f}\right)D = \frac{V_0(1-D)}{\Delta i_L f}} \quad (4)$$

Considerando que os componentes do conversor são ideais, a potência da alimentação deve ser a mesma da potência absorvida pelo resistor de carga.

$$P_S = P_0 = V_s \cdot I_s = V_0 \cdot I_0 = \frac{V_0}{V_s} = \frac{I_s}{I_0}$$

3) *Ripple da tensão de saída:* Para encontrar a corrente no capacitor a seguinte equivalência é considerada: $I_c = i_L - I_R$ Onde a corrente no capacitor é positiva, enquanto o mesmo carrega. Pelas definições de capacitância:

$$Q = C \cdot V_0$$

$$\Delta Q = C \cdot \Delta V_0$$

$$\Delta V_0 = \frac{\Delta Q}{C}$$

Considerando a variação de tensão na carga Q, temos:

$$\Delta V_0 = \frac{T \cdot \Delta I_L}{8}$$

Usando a equação da variação de corrente no indutor, podemos chegar a uma expressão de ondulação (*Ripple*) como uma fração de tensão da saída.

$$\boxed{r_{ripple} = \frac{\Delta V_0}{V_0} = \frac{(1-D)}{8LCf^2}} \quad (5)$$

Isolando o capacitor na equação encontramos a expressão para a capacitância necessária em termos de ondulação de tensão específica:

$$\boxed{C = \frac{(1-D)}{8L \left(\frac{\Delta V_0}{V_0}\right) f^2} = \frac{\Delta I_L}{8f \Delta V_{CR} V_0}} \quad (6)$$

B. PROJETO DO CONVERSOR BUCK

Com base na modelagem matemática apresentada para o conversor em II-A, e usando os parâmetros para o projeto, é possível agora calcularmos os valores para os componentes necessários para a implementação de um circuito real de conversor buck.

TABLE I
PARÂMETROS DO PROJETO

Parâmetro	Variável	Valor
Frequência de chaveamento	f_s	10kHz
Razão cíclica (<i>Duty Cycle</i>)	D	50%
<i>Ripple</i> de tensão	ΔV_C	< 2%
<i>Ripple</i> de corrente	ΔI_L	< 50%
Tensão de entrada	V_s	24V (+/- 6V)
Tensão de saída	V_0	12V+ Sinal Modulado
Potência da carga	P_{R0}	> 12W

- Utilizando a tensão de entrada máxima possível $V_{in} = 30V$, conhecendo a razão cíclica $D = 0.5$, e aplicando na equação 2, encontramos uma tensão de saída igual a:

$$V_0 = V_{in} \cdot D = (30V) \cdot (0,5) = 15V$$

1) Projeto da Indutância Mínima:

- Definindo uma corrente de saída máxima $I_{OutMax} = 1A$ e uma porcentagem máxima pro *Ripple* de corrente de 25%, encontramos um *Ripple* de Corrente igual a:

$$\Delta I_L = L_{IR} \cdot I_{OutMax} = (0,25) \cdot (1A) = 250mA$$

- Se fazendo uso das informações dos passos anteriores, podemos substituir na equação da indutância II-A2 e descobrir seu valor mínimo.

$$L = \left(\frac{V_s - V_0}{\Delta i_L f}\right)D = \left(\frac{30V - 15V}{(250mA)(10Khz)}\right)(0,5) = 3mH$$

2) Projeto da Capacitância Mínima:

- Definindo uma porcentagem do *Ripple* de tensão dentro dos requisitos de 1%, multiplicando pela tensão de saída obtemos o nosso *Ripple* de tensão igual a:

$$\Delta V_C = V_{CR} \cdot V_{out} = (0,01)(15V) = 150mV$$

- Substituindo nossas informações já conhecidas em II-A3, descobrimos o valor mínimo do nosso capacitor.

$$C = \frac{\Delta I_L}{8f\Delta V_{CR}V_o} = \frac{250mA}{8 \cdot (10kHz)(150mV)(15V)} = 20,8333\mu F$$

3) Projeto do Resistor de Potência Mínimo:

- Sabendo que nossa corrente máxima será $I_{OutMax} = 1A$ de acordo com o indutor calculado, podemos calcular o resistor de potência da seguinte forma:

$$R_0 = \frac{V_o}{I_{OutMax}} = \frac{15V}{1A} = 15\Omega$$

4) Componentes Físicos:

Para atender os parâmetros de projeto pré-definidos e pensando em uma projeto prático real, não ideal, a partir dos valores teóricos calculados anteriormente foi possível definir um valores reais de componentes comerciais para ser possível implementarmos o projeto físico do conversor buck.

- O valor teórico mínimo do capacitor era de $20,8333\mu F$, porém na seção II-F, para atingirmos a frequência de corte desejada de $500Hz$, o capacitor necessário foi de $33\mu F$, acima do valor mínimo aceitável, portanto o valor utilizado foi $C = 33\mu F$.
- Como o resistor de carga mínimo para atender uma corrente máxima de $1A$ no caso em que o conversor opera com tensão de entrada $V_{in} = 30V$, obtemos $R = 15\Omega$, porém nesse caso seria necessária uma potência de $15W$ na carga, o resistor comercial escolhido então foi o de $R = 15\Omega$ e $P = 20W$, cumprindo também o requisito de $12W$ e dando uma margem de segurança para a carga aguentar uma corrente de saída de até $1,333A$, evitando o rápido superaquecimento da carga.
- O valor do indutor foi mantido em $L = 3mH$.

TABLE II
COMPONENTES PRÁTICOS DO PROJETO

Parâmetro	Variável	Valores Práticos
Indutor	L	$3mH$
Capacitor	C	$33\mu F$
Resistência de carga	R	15Ω
Potência da carga	P	$20W$

C. FUNÇÃO DE TRANSFERÊNCIA

1) Função de Transferência pelo Domínio da Frequência:

Fazendo a análise do circuito do conversor Buck, apresentado na da figura 9, e sabendo que ela é definida por uma função de transferência de um sistema de segunda ordem, a função de transferência do conversor no domínio da frequência tem o seguinte formato que está apresentado na equação 10.

$$FT_{2order}(s) = \frac{\omega_{LC}^2}{s^2 + 2 \cdot \omega_{LC} \cdot \zeta_{LC} s + \omega_{LC}^2} \quad (7)$$

Em que ω_{LC} é a frequência angular do circuito do conversor Buck, dado em rad/s, e ζ_{LC} é o coeficiente de amortecimento.

De modo a obter os parâmetros da equação 7 em função dos componentes do circuito, tem-se as equações 8 e 9.

$$\omega_{LC} = \frac{1}{\sqrt{LC}} \quad (8)$$

$$\zeta_{LC} = \frac{1}{2R} \sqrt{\frac{L}{C}} \quad (9)$$

Substituindo as equações 8 e 9 na equação 7, e simplificando os termos, obtemos a na função de transferência em função dos componentes do circuito.

Função de transferência obtida pelo domínio da frequência:

$$FT_{2buck} = \frac{\frac{1}{LC}}{(s^2 + \frac{s}{RC} + \frac{1}{LC})} \quad (10)$$

D. Função de Transferência pelo Espaço de Estados

Após analisarmos o circuito, obtemos:

$$v_s = i_L L + v_o$$

$$i_L = \frac{dx_1}{dt}$$

$$i_L = v_o C + \frac{v_o}{R}$$

$$v_o = \frac{dv_o}{dt}$$

Substituindo e isolando as equações,

$$i_L = -\frac{v_o}{L} + \frac{v_s}{L}$$

$$v_o = \frac{i_L}{C} - \frac{v_o}{RC}$$

Colocando as equações no formato do espaço de estados,

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

$$\begin{bmatrix} I_L(s) \\ V_o(s) \end{bmatrix} = \frac{\begin{bmatrix} s + \frac{1}{RC} & \frac{-1}{L} \\ \frac{1}{C} & s \end{bmatrix} \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} V_S(s)}{s^2 + \frac{s}{RC} + \frac{1}{LC}}$$

$$\frac{I_L(s)}{V_S(s)} = \frac{\frac{1}{L} \left(s + \frac{1}{RC} \right)}{\left(s^2 + \frac{s}{RC} + \frac{1}{LC} \right)}$$

$$\frac{V_o(s)}{V_S(s)} = \frac{\frac{1}{LC}}{\left(s^2 + \frac{s}{RC} + \frac{1}{LC} \right)}$$

Função de transferência obtida pelo espaço de estados:

$$FT_{2buck} = \frac{\frac{1}{LC}}{\left(s^2 + \frac{s}{RC} + \frac{1}{LC} \right)} \quad (11)$$

E. Função de Transferência do Conversor Buck

Substituindo o valores dos componentes já calculados na equação 10 ou 11, obtém-se a função de transferência final 12.

$$FT_{Buck}(s) = \frac{9,775 \times 10^6}{s^2 + 2,02 \times 10^3 s + 9,775 \times 10^6} \quad (12)$$

F. PROJETO DOS FILTROS

A frequência de corte do conversor deve ser menor que a frequência do circuito de condicionamento de sinais, assim é possível observar o comportamento da planta sem haver perda de informações no filtro. [1] O circuito buck apresenta nele três componentes que relacionados se comportam como um filtro passa-baixas, e no circuito de condicionamento também temos um circuito passa-baixas sallen-key, o primeiro foi projetado pensando nos componentes do circuito buck, o segundo não tinha dependência de componentes já prontos, segue a baixo os parâmetros calculados para serem atendidos os requisitos de projeto, que no caso são as frequências de cortes desejadas, fator de qualidade e coeficiente de amortecimento.

1) *Frequências de Corte e Fundamental*: Tendo posse das informações como f_s e T , é possível descobrir a frequência fundamental e a de corte do filtro RLC.

$$\begin{aligned} f_s &= 10KHz \\ T &= 40ms \\ f_1 &= \frac{1}{T} = 25Hz \\ f_{LC} &= \sqrt{f_1 f_s} = \sqrt{(25Hz)(10KHz)} = 500Hz \end{aligned}$$

Definindo uma frequência de corte $F_{SK} = 1500Hz$ e verificando se cumpre o requisito de α .

$$\alpha = \frac{f_{SK} = \alpha \sqrt{f_s f_{LC}}}{\sqrt{(10KHz)(500Hz)}} = 0,670820393$$

Como podemos observar α está dentro do intervalo esperado $0,5 < \alpha = 0,67 < 1$ e o ζ_{SK} apresentado na próxima seção também está no intervalo esperado $0,1 < \zeta = 0,33 < 0,4$.

2) Filtro RLC Passa-Baixas de Segunda Ordem:

A figura 4, nos apresenta o circuito característico do filtro RLC.

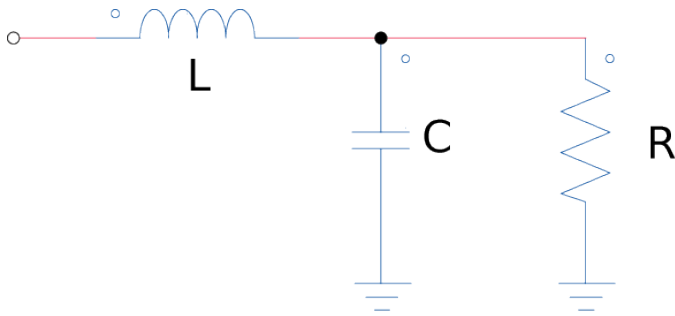


Fig. 4. Circuito do Filtro RLC Passa-Baixas
Fonte: Software PSIM

Parâmetros desejados:

$$F_{LC} = 500Hz \text{ e } \zeta_{LC} = 0,33$$

Componentes:

$$R = 15\Omega, C = 33\mu F \text{ e } L = 3mH$$

Tendo conhecimento desses parâmetros, podemos substituí-los na equação 7 e chegarmos a função de transferência do

TABLE III
PARÂMETROS ATENDIDOS

Parâmetro	Variável	Valor
Frequência de corte	F_{LC}	497,60222634705Hz
Coeficiente de amortecimento	ζ_{LC}	0,32307445639837
Fator de qualidade	Q	1,5476308637148
Margem de fase	pm	54,4° ($f = 626Hz$)

filtro RLC de segunda ordem, que como já era esperado, é a mesma do conversor buck.

$$G_{RLC}(s) = \frac{9775171,0654936}{s^2 + 2020,202020202s + 9775171,0654936} \quad (13)$$

A figura 5 apresenta o diagrama de bode do filtro RLC, definido pela função de transferência 13.

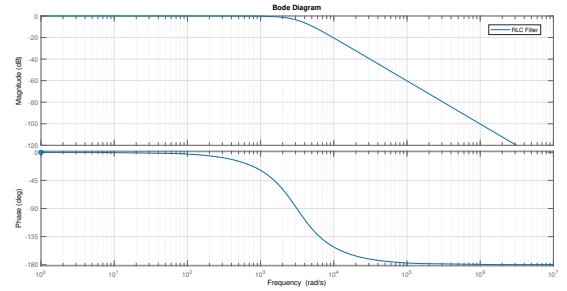


Fig. 5. Resposta do filtro RLC no domínio da frequência e de fase
Fonte: Software MATLAB

3) Filtro Sallen-Key Passa-Baixas de Segunda Ordem:

A figura 6, nos apresenta a topologia utilizada do circuito do filtro Sallen-Key. Parâmetros desejados:

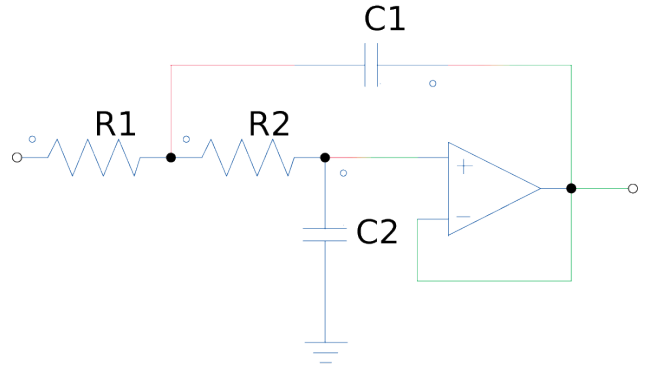


Fig. 6. Circuito do Filtro Sallen-Key Passa-Baixas
Fonte: Software PSIM

$$F_{SK} = 1500Hz \text{ e } \zeta_{SK} = 0,33$$

Componentes:

$$R1 = 47k\Omega, R2 = 22k\Omega, \\ C1 = 0.01\mu F, C2 = 0.001\mu F.$$

TABLE IV
PARÂMETROS ATENDIDOS

Parâmetro	Variável	Valor
Frequência de corte	F_{SK}	1565,1640433668 Hz
Coefficiente de amortecimento	ζ_{SK}	0,33928044236098
Fator de qualidade	Q	1,4737071094361
Margem de fase	pm	57,4° ($f = 1942$ Hz)

Tendo conhecimento desses parâmetros, podemos substituí-los na equação 7 e chegarmos a função de transferência do filtro sallen-key de segunda ordem.

$$G_{FSK}(s) = \frac{96711798,839458}{s^2 + 6673,1141199226s + 96711798,839458} \quad (14)$$

A figura 7 apresenta o diagrama de bode do filtro Sallen-Key, definido pela função de transferência 14.

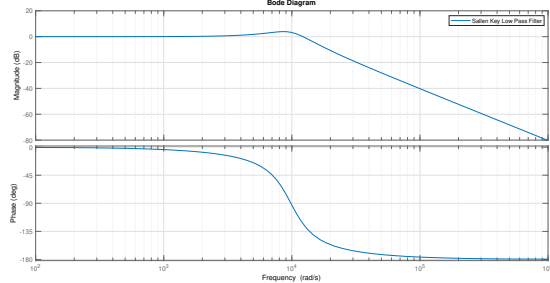


Fig. 7. Resposta do filtro Sallen-Key no domínio da frequência
Fonte: Software MATLAB

4) Resposta e código dos dois filtros:

Na sequência temos o diagrama de bode que modela o comportamento dos dois filtros plotados no mesmo gráfico, bem como o código utilizado para as plotagens.

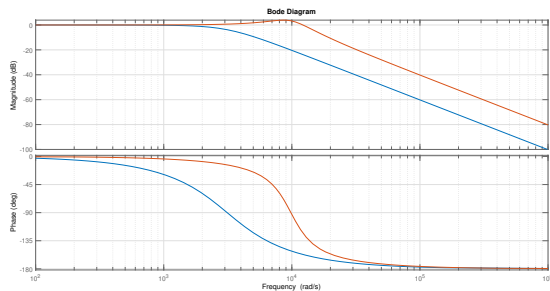


Fig. 8. Resposta dos dois filtros juntos no domínio da frequência e de fase
Fonte: Software MATLAB

```
1 clc; close all;
2 %Filtro RLC
3 num = [9775171.0654936];
4 den = [1 4838.7096774194 9775171.0654936];
5
6 %Filtro Sallen-Key
7 num2 = [96711798.839458];
```

```
8 den2 = [1 6673.1141199226 96711798.839458];
9
10 G = tf(num,den);
11 G2 = tf(num2,den2);
12 bode(G,G2);
13 grid;
```

Listing 1. Resposta dos Filtros

G. CIRCUITOS E SIMULAÇÃO

O nosso circuito buck com os componentes projetados anteriormente encontra-se na figura 9 e na sequência nas figuras 10, 11, 12 e 13, temos respectivamente resposta do circuito para uma entrada de 24V e 30V, também o ripple de tensão para uma saída de $V_0 = 15V$.

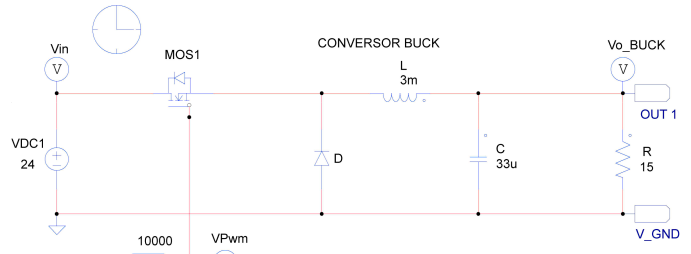


Fig. 9. Circuito conversor Buck
Fonte: Software PSIM

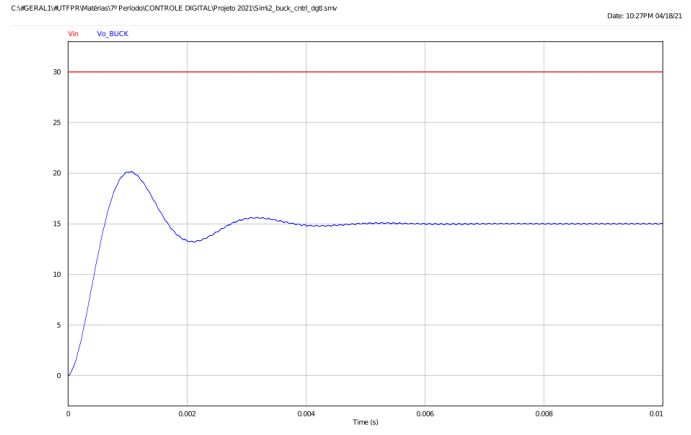


Fig. 10. Vermelho: sinal de entrada do Buck de 30V, Azul: sinal de saída do Buck de 15V
Fonte: Software PSIM

Na Figura 12, podemos observar que o ripple de tensão prático da simulação é $15,0456V - 14,9512 \approx 94,4mV$, pelo menos $50mV$ abaixo do valor $150mV$ que nosso calculo teórico que estimava e sendo aproximadamente $\approx 1\%$ do valor de tensão da saída, ou seja, menor que, e cumprindo o requisito de $< 2\%$ exigido para este parâmetro neste projeto.

H. CONDICIONAMENTO DE SINAIS

Como a tensão de amostragem no conversor buck é muito mais alta que a tensão suportada pelo microcontrolador, faz-se necessário projetar um circuito de condicionamento de sinais

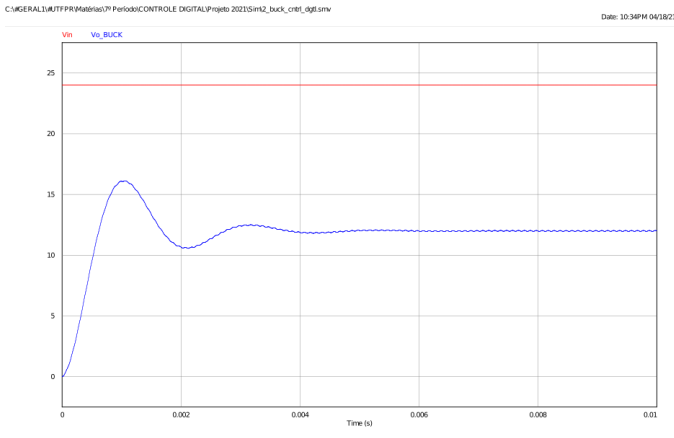


Fig. 11. Vermelho: sinal de entrada do Buck de 24V, Azul: sinal de saída do Buck de 12V
Fonte: *Software PSIM*

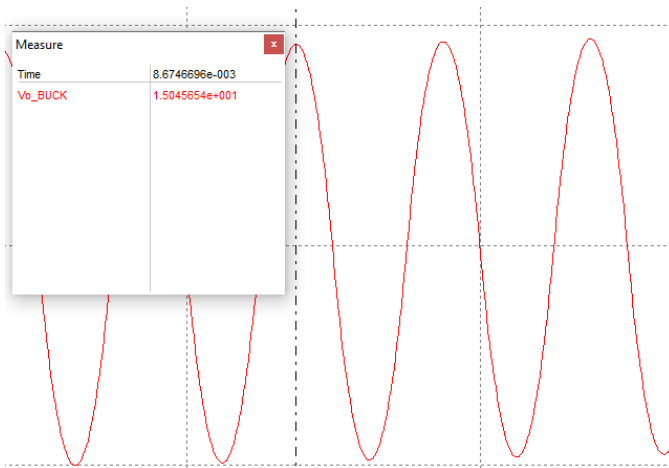


Fig. 12. Parte alta do *Ripple* de tensão no sinal de saída
Fonte: *Software PSIM*

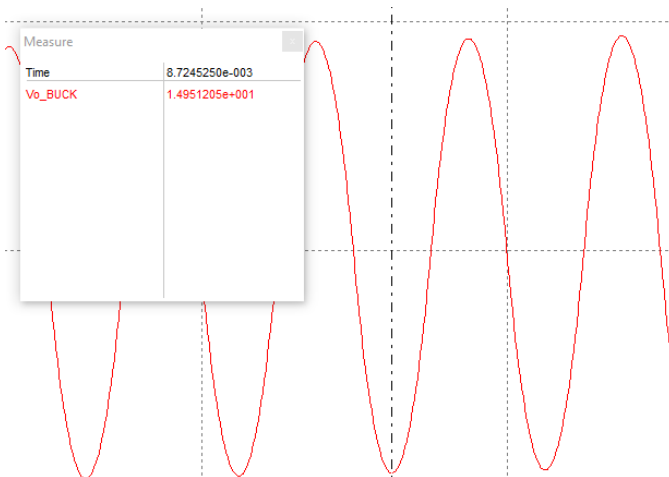


Fig. 13. Parte baixa do *Ripple* de tensão no sinal de saída
Fonte: *Software PSIM*

para que quando os dados de saída forem amostrados, possam ser lidos pelo conversor ADC do microcontrolador sem causar danos ao mesmo. Para realizarmos a leitura da tensão foi projetado um divisor de tensão de forma a limitar a tensão lida pelo microcontrolador em 3V. O divisor de tensão foi projetado da seguinte forma: quando a tensão de saída do conversor for máxima de 30V, a tensão de saída do divisor, que é lida pelo microcontrolador é 3V, consequentemente, quando a tensão de saída do divisor for 12V, a tensão lida pelo micro é 1.2V. Após o divisor de tensão, foi projetado um circuito comparador subtrator, tendo como objetivo comparar a tensão de referência com a tensão na carga do conversor buck. Esse é o sinal de erro na entrada do circuito de controle, qual devido a sua topologia de construção gera um sinal que é utilizado para controlar o que se faz necessário, neste caso a tensão sobre a resistência do conversor.

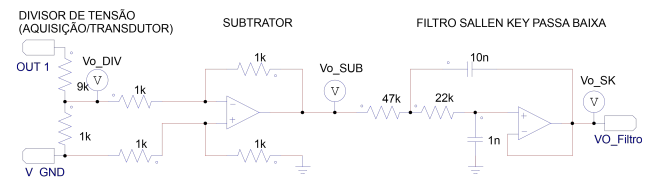


Fig. 14. Circuito de condicionamento de sinais (Parte 1)
Fonte: *Software PSIM*

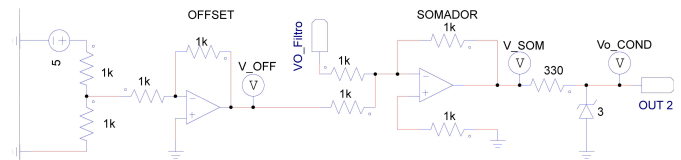


Fig. 15. Circuito de condicionamento de sinais (Parte 2)
Fonte: *Software PSIM*

Na sequência temos as saídas dos sinais de acordo com cada etapa do condicionamento exibida no circuito da figura 14 e 15.

1) Circuito Transdutor e Subtrator:

O Circuito transdutor consiste em um divisor de tensão com relação entre os resistores de 1/10, então um sinal de entrada de 15V se torna um sinal de 10x menor, ele recebe a tensão de saída V_0 do circuito Buck, bem como o seu terra, já o circuito Subtrator na sequência recebe esses dois sinais já com uma diminuição de tensão e os compara para passar adiante apenas o resultado da subtração dos sinais, isso é feito pois não há garantias de que o sinal no terra do buck é exatamente zero, O circuito segue na figura 16.

2) Filtro de Segunda Ordem:

O filtro escolhido foi o padrão mais utilizado chamado de Sallen-Key, seu projeto foi exibido anteriormente, sua função é filtrar as frequências acima de 1565hz para evitar ruídos e perturbações no sinal, observando a figura 19 podemos observar que o ripple que causava uma ondulação de alta frequência no sinal é totalmente filtrado. O Circuito segue na

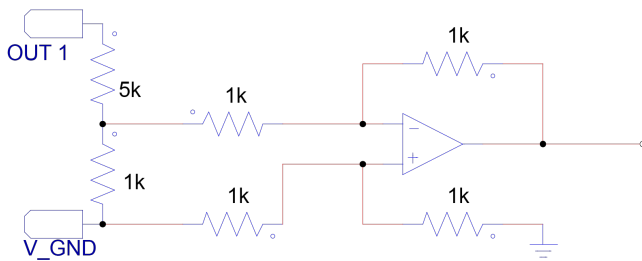


Fig. 16. Transdutor e Subtrator
Fonte: Software PSIM

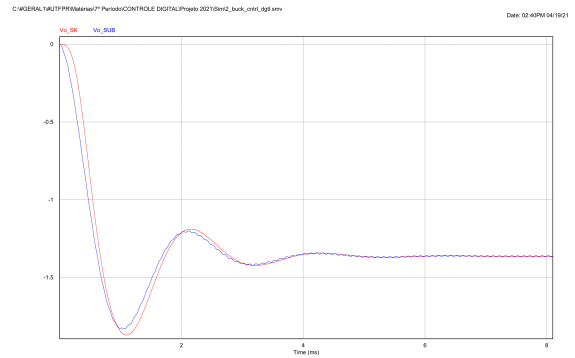


Fig. 19. Azul: sinal de saída do subtrator; Vermelho: sinal de saída do filtro sallen-key;
Fonte: Software PSIM

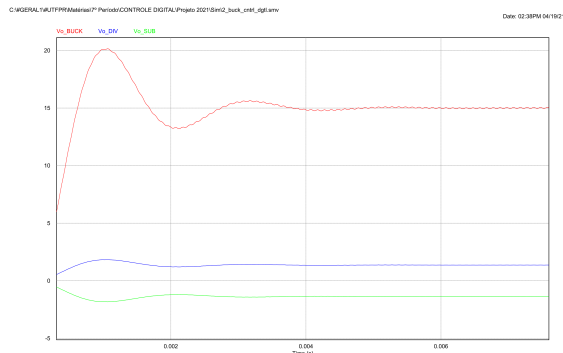


Fig. 17. Vermelho: sinal de saída do buck; Azul: sinal de saída do divisor de tensão; Verde: sinal de saída do subtrator;
Fonte: Software PSIM

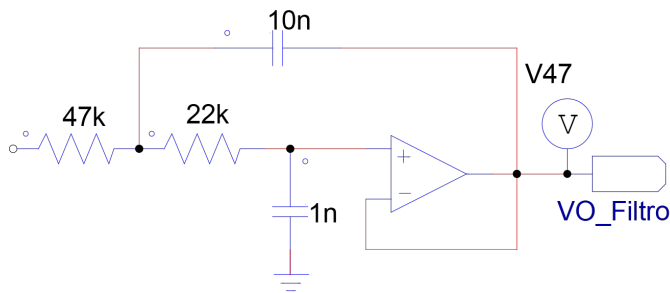


Fig. 18. Filtro Sallen-Key
Fonte: Software PSIM

figura 18. Na sequência temos as saídas dos sinais de acordo com cada etapa do condicionamento exibida no circuito da figura 15.

3) Circuito de Offset e Somador:

Como o valor do sinal após o processo de aquisição e filtragem ficou muito pequeno, é necessário adicionarmos um sinal de *offset* de $\approx 1,6V$ para incrementar esse sinal e deixa-lo mais próximo dos 3.3V recebidos por um microcontrolador, e também para evitar que nosso sinal fique flutuando muito próximo e cruzando o zero, o que pode causar problemas no chaveamento. Pode observar-se na figura 22 que o circuito

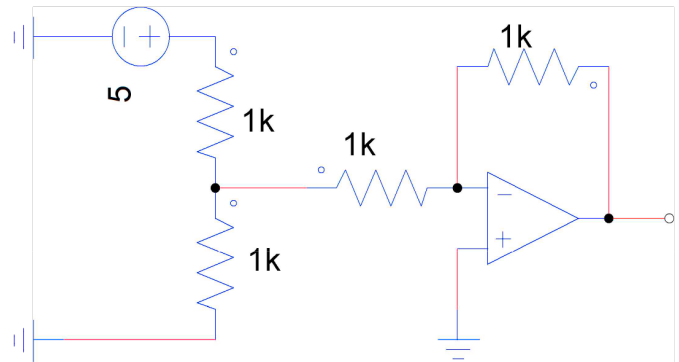


Fig. 20. Circuito de Offset
Fonte: Software PSIM

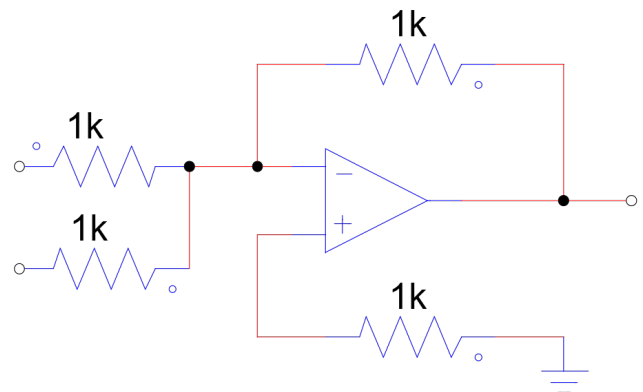


Fig. 21. Circuito Somador
Fonte: Software PSIM

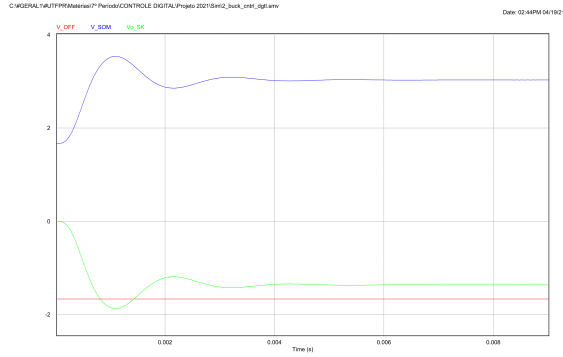


Fig. 22. Verde: sinal de saída de saída do filtro sellen-key; Vermelho: sinal de *offset*; Azul: sinal de saída do somador;
Fonte: *Software PSIM*

4) Circuito Ceifador:

O sinal somado está chegando no valor desejado $3V$, mas ainda existe um problema, o pico antes dele estabilizar está atingindo $3,5V$ e isso pode ser um problema já que nosso micro deve receber no máximo $3.3V$, portanto a alternativa foi adicionar um circuito ceifador com tensão de bloqueio em $3,2V$ para garantirmos a segurança do microcontrolador. Pode se observar a diferença dos sinais e o sinal de saída sendo "ceifado" em azul na figura 24. O circuito segue na figura 23.

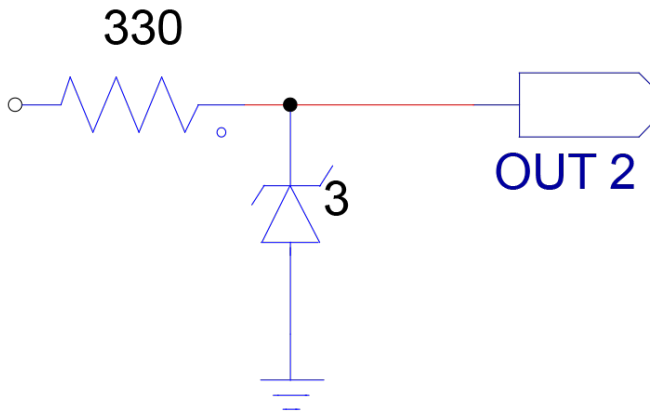


Fig. 23. Circuito Ceifador
Fonte: *Software PSIM*

5) Comparação dos Sinais:

Na Figura 26 é exibida a sobreposição dos sinais de saída do conversor buck, so sinal PWM e da saída do circuito de condicionamento de sinais.

I. PWM

Para geração do sinal PWM, nesta etapa do sistema em malha aberta foi utilizado a ferramenta "*Gating Block*" na

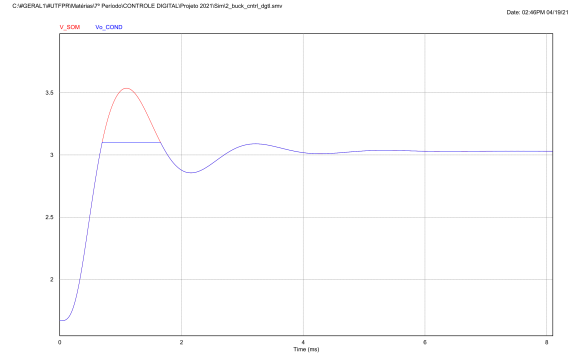


Fig. 24. Para $V_{in} = 30V$, Vermelho: sinal de saída do somador; Azul: sinal de saída do ceifador;
Fonte: *Software PSIM*

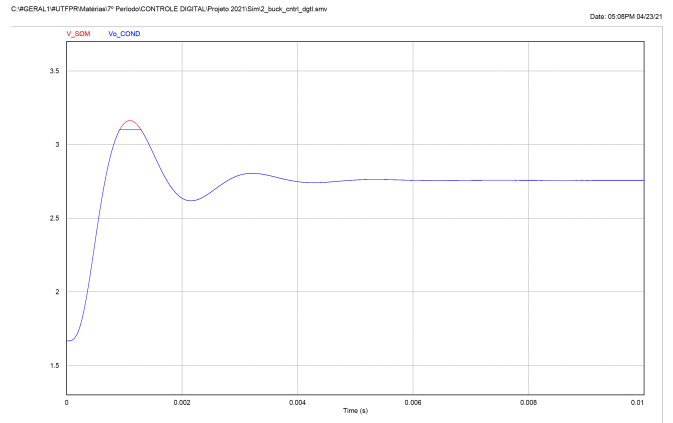


Fig. 25. Para $V_{in} = 24V$, Vermelho: sinal de saída do somador; Azul: sinal de saída do ceifador;
Fonte: *Software PSIM*

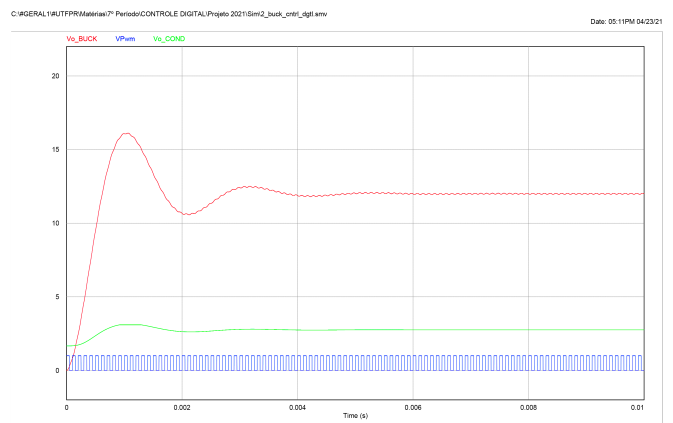


Fig. 26. Sobreposição da saída do conversor, sinal PWM e saída do circuito de condicionamento de sinais;
Fonte: *Software PSIM*

simulação realizada no *Software* PSIM, a mesma foi configurada para gerar um sinal PWM de frequência 10kHz , que é inserido na porta *Gate* do MOSFET, gerando nosso chaveamento contínuo no sinal de entrada no MOSFET, consequentemente alterando a onda de saída de acordo com a razão cíclica de 0.5. O sinal PWM utilizado segue na figura 27.

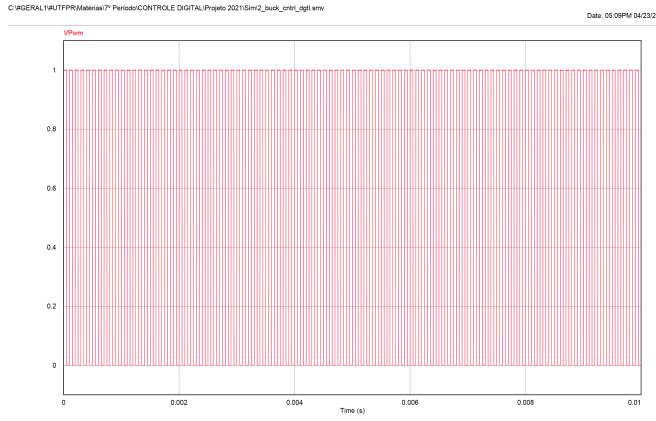


Fig. 27. Sinal PWM Gerado
Fonte: *Software* PSIM

III. PROJETO MALHA FECHADA

Nesta seção serão exibidos os resultados do sistema de controle em malha fechada para o conversor buck, implementados utilizando o MATLAB, nas exibições dos resultados em gráficos, foi utilizado um padrão de cores para destacarmos com clareza cada sinal e suas mudanças ao longo do processo.

A. Geração do Sinal de Referência

Utilizando o software matlab, foi realizada a discretização do sinal de referência, no caso, de uma onda trapezoidal por meio do código abaixo.

```
1 %[...]
2
3 %% Parametros do sinal de referencia
4
5 fs = 10000; %freq de amostragem
6 T0 = 40e-3; %periodo fundamental
7 f0 = 1/T0; %freq fundamental
8 N = fs/f0; %qnt de pontos por amostra
9
10 Aref = 18; %Amplitude maxima da referencia
11 Oref = 6; %Offset minima da referencia
12 Vdif = Aref - Oref;
13 kq = 0;
14
15 %[...]
16
17 for k = 2:N
18
19     % Vetor de tempo disc. para graficos
20     t(k) = k*T;
21     kq = kq + 1;
22
23     % Geracao do sinal de referencia
24     if(k>0 && k<=50)
25         x(k) = Vdif;
26     end
27     if(k>50 && k<=60)
```

```
28         x(k) = x(k-1) + (Oref/10);
29     end
30     if(k>60 && k<=150)
31         x(k) = Aref;
32     end
33     if(k>150 && k<=160)
34         x(k) = x(k-1) - (Oref/10);
35     end
36     if(k>160 && k<=250)
37         x(k) = Vdif;
38     end
39     if(k>250 && k<=260)
40         x(k) = x(k-1) - (Oref/10);
41     end
42     if(k>260 && k<=350)
43         x(k) = Oref;
44     end
45     if(k>350 && k<=360)
46         x(k) = x(k-1) + (Oref/10);
47     end
48     if(k>360 && k<=400)
49         x(k) = Vdif;
50     end
```

Listing 2. Sinal de Referência

Segue abaixo forma de onda gerada através da execução desse código em matlab:

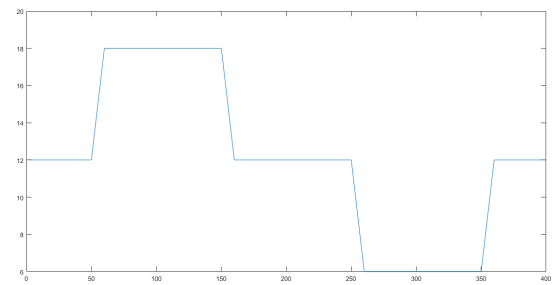


Fig. 28. Sinal de Referência - Onda Trapezoidal
Fonte: *Software* MATLAB

B. Malha de Controle

O controle da forma de onda do conversor sobre a equação de referência é dado pela ação de controle PID (proporcional, integral e derivativo). O código da simulação da malha de controle em MatLab é representado pela figura III-B3.

1) *Ação proporcional*: O sinal de controle aplicado à planta é proporcional a amplitude do valor do sinal de erro. Esta ação pode ser usada para corrigir a resposta transitória ou o erro, não podendo efetuar as duas correções simultaneamente. Com o aumento de K_p , o erro tende a diminuir, porém não se anula. Um aumento expressivo no K_p pode ter como consequência a instabilidade e até mesmo a saturação do sistema.

2) *Ação integral*: Tem como principal objetivo eliminar o erro em regime permanente, O efeito desta ação é pequeno no início, pois o erro acumulado é pequeno, contudo, seu efeito é crescente a medida que o sistema está perto do

regime permanente. Esta ação não é normalmente utilizada isoladamente, uma vez que, sua resposta é lenta e oscilatória.

3) *Ação derivativa:* A ação derivativa produz um sinal de saída que é proporcional a velocidade da variação do erro. A ação derivativa fornece uma correção antecipada do erro, diminuindo o tempo de resposta e melhorando a estabilidade do Sistema.

```
1 %[...]
2 %% Ganhos do controlador
3
4 Kp = 0.1;
5 Ki = 0.125;
6 Kd = 1;
7
8 %[...]
9 % Malha de Controle
10 e(k) = x(k) - ym(k); %erro = referencia + saida
11   adc
12
13 uP(k) = Kp*e(k); %Controle
14   Proporcional
15 uI(k) = uI(k-1) + Ki*e(k); %Controle
16   Integral
17 uD(k) = Kd*(e(k) - e(k-1)); %Controle
18   Derivativo
19 uPID(k) = uP(k) + uI(k) + uD(k); %Controle
20   PID
21
22 d(k) = abs(uPID(k)) * P/E;
23 su = sign(uPID(k));
24 %[...]
```

Listing 3. Malha de Controle

A forma de onde representada pela figura 29 apresenta o contraste entre o sinal de referência e o sinal de controle. Quando temos um erro, a ação de controle age rapidamente para estabilizar o sistema resultado em erro próximo a nulo.

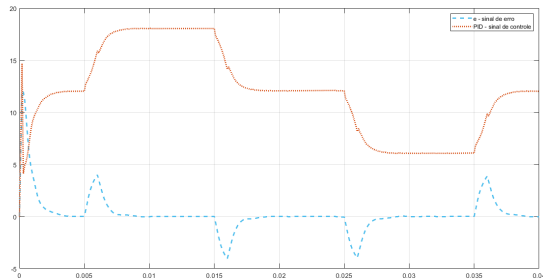


Fig. 29. Vermelho - Sinal de Controle PID, Azul - Sinal de Erro
Fonte: Software MATLAB

Na figura 30 observa-se separadamente as três ações de controle atuando simultaneamente na correção do erro.

Na figura 31 apresenta-se o contraste entre a ação de controle e a saída da planta resultante desta ação.

C. Condicionamento de Sinais

O condicionamento de sinais é realizado em três etapas, a sequência delas são detalhadas, e nas figuras 32 e 33 é exibido o efeito dessas etapas sobre o sinal.

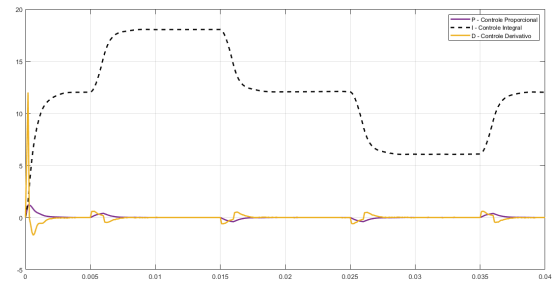


Fig. 30. Roxo - Ação de Controle Proporcional, Caramelo - Ação de Controle Derivativo, Preto - Ação de Controle Integral
Fonte: Software MATLAB

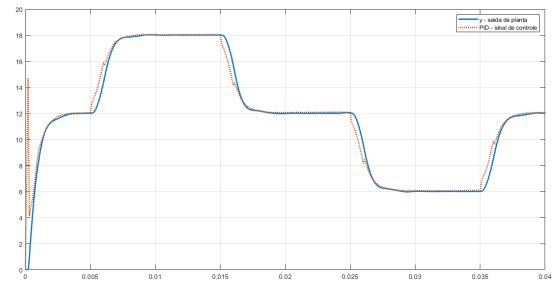


Fig. 31. Azul - Saída da Planta, Y - Sinal de Controle PID
Fonte: Software MATLAB

1) *Etapa 1: Divisor de Tensão:* O divisor resistivo é utilizado para aplicar uma mudança de escala de razão 1/10 no sinal de saída da planta. Dessa forma o sinal pode ser lido pelo conversor ADC.

2) *Etapa 2: Filtro Sallen Key de Segunda Ordem:* O filtro passa-baixas, projetado na seção II-F é aplicado ao sinal, eliminando os ruídos em altas frequências..

3) *Etapa 3: Offset:* A aplicação de um *offset* de 0.5 V no sinal necessita-se em razão de um valor muito próximo de nulo na entrada do conversor ADC

```
1 %[...]
2 % Saída da planta + entrada PWM
3 y(kc+1) = -a1*y(kc) - a0*y(kc-1) + b1*PWM(kc) + b0*
4   PWM(kc-1);
5
6 % Divisor Resistivo
7 yd(kc+1) = y(kc+1)*K;
8
9 % Condicionamento de sinais
10 ys(kc+1) = -as1*ys(kc) - as0*ys(kc-1) + bs1*
11   yd(kc) + bs0*yd(kc-1);
12
13 % Offset
14 yo(kc+1) = ys(kc+1) + offsetCS;
15 %[...]
```

Listing 4. Condicionamento de Sinais

Na figura 32 pode-se observar a razão entre a saída da planta e o circuito de condicionamento de sinais. Já na figura 33 nota-se o offset de 0.5 aplicado ao condicionamento de sinais.

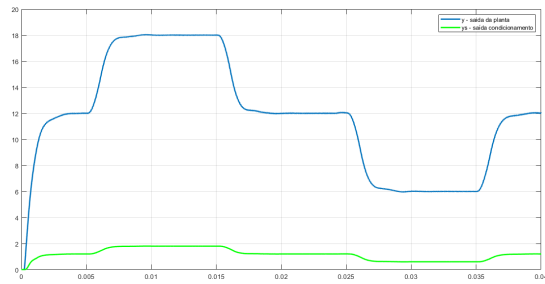


Fig. 32. Azul - Sinal da Planta, Verde - Saída do Condicionamento de Sinais (Filtro SLK)

Fonte: *Software MATLAB*

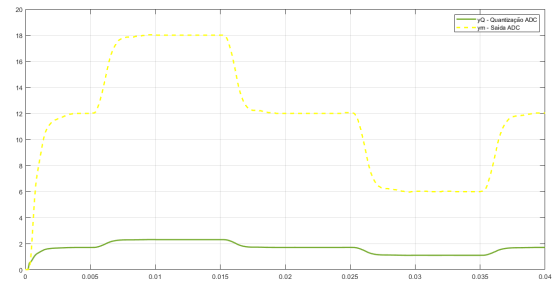


Fig. 34. Verde Musgo - Quantização do ADC, Amarelo - Sinal de Saída do ADC

Fonte: *Software MATLAB*

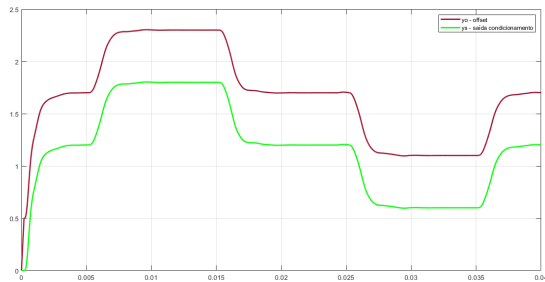


Fig. 33. Bordô - Sinal Com *Offset*, Verde - Saída do Condicionamento de Sinais (Filtro SLK)

Fonte: *Software MATLAB*

D. Conversor ADC

O conversor ADC, representado pelo código em III-D transforma a saída do condicionamento de sinais em um sinal discreto no tempo, quantizado com uma resolução de 10 bits, com 1024 níveis.

```
1 %[...]
2 %Conversor ADC
3 yA(k+1) = yo(kc+1); %
4 Amostragem
5 for i = (kc+1):(kc+1+P)
6     yR(i) = yA(k+1);
7 end
8 yQ(k+1) = round(yR(kc+1)*Q/EAD)/Q*EAD; %
9 Quantizacao
10 ym(k+1) = (yQ(k+1) - offsetCS)/K; % Valor
11 Original
12 %[...]
```

Listing 5. Malha de Controle

E. Geração do sinal PWM

O sinal PWM é modulado de forma que seu *duty cycle* seja controlado, e por meio desse controle da sua frequência, é gerado uma lógica sobre a chave da planta do conversor buck, que torna possível obtermos um sinal muito semelhante ao sinal de referência na saída do conversor. O PWM tem resolução de 8 bits e 256 níveis. Nas figuras 35 e 36 é possível observarmos o PWM mudando a frequência com que alterna para "modelar" assim o sinal de saída desejado.

```
1 %[...]
2 % Geracao do PWM (kc)
3 if cc <= d(k)
4     PWM(kc) = su*E;
5 end
6 if cc > d(k)
7     PWM(kc) = 0;
8 end
9 %[...]
```

Listing 6. Malha de Controle

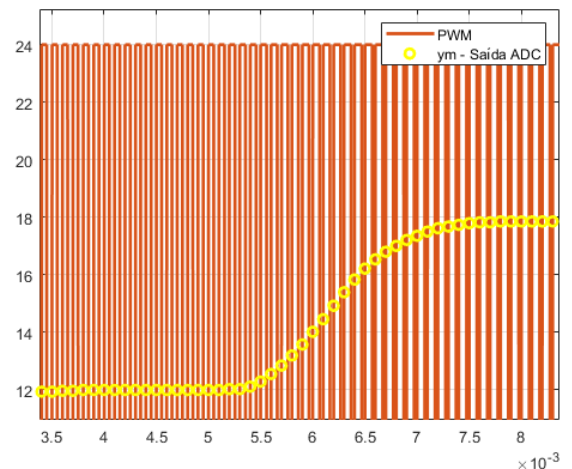


Fig. 35. Laranja - Sinal PWM, Amarelo - Sinal de Saída do ADC

Fonte: *Software MATLAB*

F. Resultado Final

Na imagem 37 é apresentada a comparação mais interessante de sinais, o sinal de referência que era o objetivo deste trabalho, em contraste com os sinais da planta e de saída do ADC, os quais tem resultados muito próximos ao sinal de objetivo cumprindo com o desejado após o processo de fecharmos a malha do conversor buck.

G. Código Completo

Abaixo, o código em III-G apresenta o script inteiro de simulação do sistema de controle do conversor buck em malha fechada no software *MatLab*.

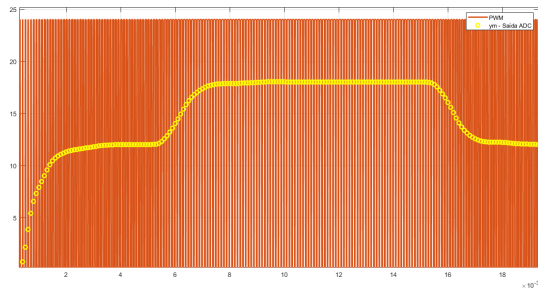


Fig. 36. Laranja - Sinal PWM, Amarelo - Sinal de Saída do ADC
Fonte: Software MATLAB

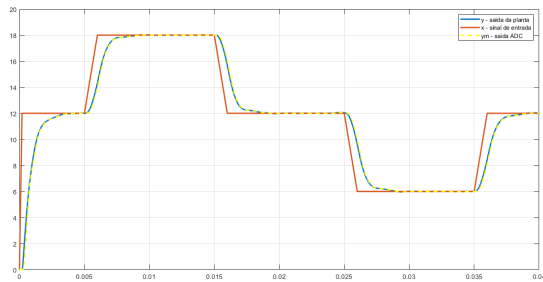


Fig. 37. Vermelho - Sinal de Referência, Azul - Saída da Planta, Amarelo - Saída do ADC
Fonte: Software MATLAB

```

1 %% CONVERSOR BUCK MALHA FECHADA
2
3 clear
4 clc
5 close all
6 T = 0.1e-3;
7 s = tf('s');
8
9 %% Parametros do sinal de referencia
10
11 fs = 10000; %freq de amostragem
12 T0 = 40e-3; %periodo fundamental
13 f0 = 1/T0; %freq fundamental
14 N = fs/f0; %qnt de pontos por amostra
15
16 Aref = 18; %Amplitude maxima da referencia
17 Oref = 6; %Offset minima da referencia
18 Vdif = Aref - Oref;
19 kq = 0;
20
21 %% Parametros do modulador PWM
22
23 E = 24;
24 NbitsPWM = 8;
25 P = 2^NbitsPWM; % num de niveis do PWM
26 Vmax = 30; % Tensao maxima na planta
27
28 %% Definicao dos parametros do conv.A/D
29
30 EAD = 3; %Entrada do conversor A/D
31 NbitsAD = 10; %Num. de bits do conv. A/D
32 Q = 2^NbitsAD; %Num. de niveis conv. A/D
33 offsetCS = 0.5; %Offset do condicionamento
34 K = EAD / Vmax; %Ganho transdut.(div.res.)
35
36 %% Planta e Coeficientes da Equacao de Diferencas

```

```

37
38 %Planta = Filtro Passa Baixas RLC 1 Ordem / BUCK
39
40 % Periodo de amostragem
41 Tc = T/P;
42 %Planta Continua P(s) / Funcao de Transferencia em
43   Tempo Continuo (CTTF)
44 Pc = 9775171.0654936 / (s^2 + 2020.202020202*s +
45   9775171.0654936);
46
47 %Planta Discreta P(z) / Funcao de Transferencia em
48   Tempo Discreto (DTTF)
49 Pd = c2d(Pc,Tc);
50
51 %Coeficientes ED
52 [npd,dpd] = tfdata(Pd,'v')
53 a1 = dpd(2);
54 a0 = dpd(3);
55 b1 = npd(2);
56 b0 = npd(3);
57
58 %% Condicionamento de Sinais e Coeficientes da
59   Equacao de Diferencas
60
61 %Condicionamento = Filtro Passa Baixas Sallen Key 2
62   Ordem
63
64 %Sensor Continuo P(s)
65 Sc = 96711798.839458 / (s^2 + 6673.1141199226*s +
66   96711798.839458);
67
68 %Sensor Discreto
69 Sd = c2d(Sc,Tc);
70
71 %Coeficientes ED
72 [nsd,dsd] = tfdata(Sd,'v')
73 as1 = dsd(2);
74 as0 = dsd(3);
75 bs1 = nsd(2);
76 bs0 = nsd(3);
77
78 %% Ganhos do controlador
79
80 Kp = 0.1;
81 Ki = 0.125;
82 Kd = 1;
83
84 %% Inicializacao dos Vetores
85
86 t = zeros(1,N);
87 tc = zeros(1,N*P);
88 x = zeros(1,N); %referencia
89 e = zeros(1,N); %erro
90 d = zeros(1,N);
91
92 uP = zeros(1,N); %Controle Proporcional
93 uI = zeros(1,N); %Controle Integral
94 uD = zeros(1,N); %Controle Derivativo
95 uPID = zeros(1,N);
96 PWM = zeros(1,N*P);
97
98 y = zeros(1,N*P); %saida planta
99 yd = zeros(1,N*P); %saida divisor resistivo
100 ys = zeros(1,N*P); %saida do condicionamento
101 yo = zeros(1,N*P); %offset na saida
102 ym = zeros(1,N); %saida ADC
103 yA = zeros(1,N);
104 yR = zeros(1,N*P);
105 yQ = zeros(1,N);
106
107 %% Simulacao Conversor Buck
108
109 for k = 2:N
110
111   % Vetor de tempo disc. para graficos
112   t(k) = k*T;

```

```

105 kq = kq + 1;
106
107 % Geracao do sinal de referencial
108 if(k>0 && k<=50)
109     x(k) = Vdif;
110 end
111 if(k>50 && k<=60)
112     x(k) = x(k-1) + (Oref/10);
113 end
114 if(k>60 && k<=150)
115     x(k) = Aref;
116 end
117 if(k>150 && k<=160)
118     x(k) = x(k-1) - (Oref/10);
119 end
120 if(k>160 && k<=250)
121     x(k) = Vdif;
122 end
123 if(k>250 && k<=260)
124     x(k) = x(k-1) - (Oref/10);
125 end
126 if(k>260 && k<=350)
127     x(k) = Oref;
128 end
129 if(k>350 && k<=360)
130     x(k) = x(k-1) + (Oref/10);
131 end
132 if(k>360 && k<=400)
133     x(k) = Vdif;
134 end
135
136 % Malha de Controle
137 e(k) = x(k) - ym(k); %erro = referencia + saida
138 adc
139
140 uP(k) = Kp*e(k); %Controle
141 %Proporcional
142 uI(k) = uI(k-1) + Ki*e(k); %Controle
143 %Integral
144 uD(k) = Kd*(e(k) - e(k-1)); %Controle
145 %Derivativo
146 uPID(k) = uP(k) + uI(k) + uD(k); %Controle
147 PID
148
149 d(k) = abs(uPID(k))*P/E;
150 su = sign(uPID(k));
151
152 % Discretizacao dos sistemas continuos
153 for cc = 1:P
154     % Contador e vetor de tempo do sistema
155     superdiscretizado
156     kc = k*P + cc;
157     tc(kc) = kc*Tc;
158
159     % Geracao do PWM(kc)
160     if cc <= d(k)
161         PWM(kc) = su*E;
162     end
163     if cc > d(k)
164         PWM(kc) = 0;
165     end
166
167     % Saida da planta + entrada PWM
168     y(kc+1) = -a1*y(kc) - a0*y(kc-1) + b1*PWM(kc) + b0*
169     PWM(kc-1);
170
171     % Divisor Resistivo
172     yd(kc+1) = y(kc+1)*K;
173
174     % Condicionamento de sinais
175     ys(kc+1) = -as1*ys(kc) - as0*ys(kc-1) + bs1*
176     yd(kc) + bs0*yd(kc-1);
177
178     % Offset

```

```

171     yo(kc+1) = ys(kc+1) + offsetCS;
172 end
173
174 %Conversor ADC
175 yA(k+1) = yo(kc+1); %
176 %Amostragem
177 for i = (kc+1):(kc+1+P)
178     yR(i) = yA(k+1);
179 end
180
181 yQ(k+1) = round(yR(kc+1)*Q/EAD)/Q+EAD; %
182 %Quantizacao
183 ym(k+1) = (yQ(k+1) - offsetCS)/K; % Valor
184 %Original
185
186 % OCodigo de plotagem dos graficos foi retirado e
187 % esta no arquivo anexo

```

Listing 7. Código Completo

IV. CONSIDERAÇÕES FINAIS

Observando todos os parâmetros pré-definidos para o projeto, todos foram cumpridos, se compararmos alguns, até foram resultados melhores do que os requisitos, a tensão de saída está no valor da metade da tensão de entrada e está de acordo com a razão cíclica de 0,5(50%), o *Ripple* de corrente utilizado foi de 25% sendo metade do requisito de 50%, o *Ripple* de tensão está abaixo até do 1% a qual foi projetado enquanto o requisito era de 2%.

Com relação aos filtros, suas frequências de corte, bem como os coeficientes de amortecimento e a relação de ambos os filtros foram atendidas por valores extremamente próximos que filtraram todas as ondulações e ruídos de alta frequência, permanecendo só nossa fundamental e suas múltiplas. Analisando a saída do nosso conversor buck, tivemos um pouco de *overshoot*, mas o sinal estabiliza rapidamente e no processo de condicionamento de sinais é observado que o sinal de saída que irá para o microcontrolador é controlado e fica estabilizado em aproximadamente 3V, valor que é muito bom. O sinal trapezoidal de referência foi obtido com sucesso, mas por enquanto como foi concluída a etapa do sistema em malha aberta, sua aplicação acontecerá ao adicionarmos o microcontrolador para a etapa de controle PID e um circuito *driver* para fecharmos a malha e obtermos o mesmo na saída do conversor buck a partir de uma lógica de chaveamento.

Por fim, discutindo sobre a implementação do projeto malha fechada feita no *Matlab*, os resultados obtidos na saída da planta e na saída do conversor ADC foram muito próximos ao sinal de referência, após vários testes os parâmetros do controlador PID foram ajustados de forma que o sinal não ficasse oscilatório e pudesse estabilizar rapidamente, os ganhos que forneceram os melhores resultados foram: ganho proporcional = 0,125 e ganho integral = 0,125 e ganho derivativo = 1.

REFERENCES

- [1] Hart, Daniel W , Power electronics, Tata McGraw-Hill Education, 2011.
- [2] Moraes, Caio, Análise do conversor Buck em Condução Contínua, 2018, <https://eletronicadepotencia.com/analise-do-conversor-buck-em-mcc/>, Acessado em 16/04/2021.