

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E ELETRÔNICA
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Maria Luiza Fabrício de Oliveira

SIMULAÇÃO EM TEMPO REAL DE CONVERSORES CC-CC EM FPGA

Florianópolis
2022

Maria Luiza Fabrício de Oliveira

SIMULAÇÃO EM TEMPO REAL DE CONVERSORES CC-CC EM FPGA

Trabalho de Conclusão de Curso de Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal de Santa Catarina para a obtenção do título de Bacharel em Engenharia Elétrica.
Orientador: Prof. Samir Ahmad Mussa, Dr
Coorientador: Eng. José Augusto Arbugeri

Florianópolis

2022

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Oliveira, Maria Luiza Fabrício de
SIMULAÇÃO EM TEMPO REAL DE CONVERSORES CC-CC EM FPGA /
Maria Luiza Fabrício de Oliveira ; orientador, Samir Ahmad
Mussa, coorientador, José Augusto Arbuseri, 2022.
74 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro Tecnológico,
Graduação em Engenharia Elétrica, Florianópolis, 2022.

Inclui referências.

1. Engenharia Elétrica. 2. FPGA. 3. controle. 4.
eletrônica de potência. I. Mussa, Samir Ahmad. II.
Arbuseri, José Augusto. III. Universidade Federal de Santa
Catarina. Graduação em Engenharia Elétrica. IV. Título.

Maria Luiza Fabrício de Oliveira

SIMULAÇÃO EM TEMPO REAL DE CONVERSORES CC-CC EM FPGA

Este Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Engenharia Elétrica” e aceito, em sua forma final, pelo Curso de Graduação em Engenharia Elétrica.

Florianópolis, 30 de agosto de 2022.

Prof. Miguel Moreto, Dr.
Coordenador do Curso de Graduação em Engenharia Elétrica

Banca Examinadora:

Prof. Samir Ahmad Mussa, Dr
Orientador
Universidade Federal de Santa Catarina

Prof. Hari Bruno Mohr, Dr
Universidade Federal de Santa Catarina

Eng. Eletric. Edhuardo Francisco Celli Grabovski, Me.
Universidade Federal de Santa Catarina

AGRADECIMENTOS

À minha família por todo apoio incondicional durante os anos deste ciclo de estudo. Agradeço a minha mãe, Graciele Fabrício, minha avó Brunhilda Fabrício, meu avô Biron Batista Fabrício, meu tio e dindo Sandro Fabrício e a minha tia Débora Cristina Torcato por todo o amor e carinho sempre presente na minha vida, por me incentivarem a dar meu melhor e acreditarem em mim. Sou imensamente grata por tudo o que me proporcionaram e diariamente visam proporcionar, e por todas as oportunidades e experiências que me ajudaram a enfrentar.

Agradeço imensamente ao professor Samir Ahmad Mussa por ter sido meu orientador e ter oportunizado no INEP e trabalhar com pesquisa científica. Sua orientação e dedicação foram essenciais para me tornar a profissional que sou hoje. Agradeço também ao José Augusto Arbugeri e ao Eduardo Francisco CelliGrabovski , por todos os ensinamentos, paciência e dedicação.

À meus amigos e amigas que cultivei ao longo dos anos na graduação, em especial para aqueles e aquelas que começaram comigo em agosto de 2016 e até hoje são parte essencial na minha vida. Somente tenho a agradecer todos os momentos que desbravamos juntos, dentro e fora das salas de aula. Agradeço em especial aos colegas que se tornaram amigos e família no decorrer dos anos: Gabriel Germino Martins de Jesus , Lucas Fiamoncini de Souza, Marina Galarça Santa Catarina, Marcello Ferreira Vilela da Silva, Pedro José Teixeira Pagano, Pedro Paulo Santos Gomes da Silva e Vitória Rigoni.

Por fim, agradeço à Universidade Federal de Santa Catarina, ao curso de Graduação em Engenharia Elétrica, seu corpo docente e demais colaboradores pela excelente educação pública, gratuita e de qualidade.

*“Werde, der du bist”
(Friedrich Nietzsche)*

RESUMO

Este documento apresenta o estudo e implementação da metodologia HIL utilizando FPGA . O objetivo é simular uma alternativa ao HIL em tempo real, utilizando conversores eletrônicos de potência como parâmetro de verificação. No cenário atual, um HIL possui alto custo de implementação, além de ser um sistema sofisticado e que necessita a utilização de um software dedicado. Portanto, o objetivo principal deste projeto é fornecer uma alternativa viável para a aplicação através do uso do FPGA. Para fins de demonstração, foram modelados os conversores Buck e Boost. A validação dos resultados dos conversores foi feita via MATLAB e funcionalidades do Quartus. Por fim, para fins didáticos e de pesquisa, algumas formas de onda foram mostradas como resultado dos conversores Buck e Boost emulados em um kit de desenvolvimento FPGA e do controle da tensão do conversor Boost. Os resultados foram lidos no osciloscópio através de um DAC de baixo custo. Este documento apresenta o estudo e implementação da metodologia HIL utilizando FPGA . O objetivo é simular uma alternativa ao HIL em tempo real, utilizando conversores eletrônicos de potência como parâmetro de verificação.

Palavras-chave: FPGA, controle, eletrônica de potência

ABSTRACT

This paper presents the study and implementation of the HIL methodology using FPGA . The goal is to simulate an alternative to HIL in real time, using power electronic converters as a verification parameter. In the current scenario, a HIL has a high implementation cost and is a sophisticated system that requires the use of dedicated software. Therefore, the main goal of this project is to provide a viable alternative application through the use of FPGA. For demonstration purposes, Buck and Boost converters have been modeled. Validation of the converter results was done via MATLAB and Quartus features. Finally, for didactic and research purposes, some waveforms were shown as a result of the Buck and Boost converters emulated in an FPGA development kit and the voltage control of the Boost converter. The results were read on the oscilloscope using a low cost DAC.

Keywords: FPGA, control, power electronics.

LISTA DE FIGURAS

Figura 1 – Diagrama simplificado da aquisição de dados	25
Figura 2 – Fases de operação do conversor Buck em Modo de condução contínua (CCM).	29
Figura 3 – Tensão v_o e Corrente $\langle i_L \rangle_{T_s}$ do conversor Buck	29
Figura 4 – Diagrama de blocos de funções do controle de laço	31
Figura 5 – Fases de operação do conversor Boost em CCM	33
Figura 6 – Tensão v_o e corrente $\langle i_L \rangle_{T_s}$ do conversor Boost	33
Figura 7 – DE10-Standard	37
Figura 8 – Método da aquisição de Dados	38
Figura 9 – Circuito do filtro RC passa baixa	39
Figura 10 – Arquitetura do High Resolution Pulse Width Modulation (HRPWM)	39
Figura 11 – Corrente do indutor do conversor de Buck e tensão do capacitor e período de comutação utilizando o software MATLAB	41
Figura 12 – Conversor Buck em Modelsim	42
Figura 13 – Fluxograma simplificado código de emulação do Conversor Buck	42
Figura 14 – RTL Viewer - Conversor Buck	43
Figura 15 – Valores médios do conversor Buck de tensão (v_o), corrente (i_L) e período de comutação(SW). Em azul a tensão (v_o), em magenta a corrente (i_L) e em ciano o período de comutação (SW).	44
Figura 16 – Corrente do indutor do conversor de Boost e tensão do capacitor e período de comutação utilizando o software MATLAB	45
Figura 17 – Fluxograma simplificado código de emulação do Conversor Boost	46
Figura 18 – RTL Viewer - Boost Malha Aberta	46
Figura 19 – Conversor de Boost malha aberta utilizando a ferramenta para simular.	47
Figura 20 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$ e $D = 0.5$. Trigger ativado pela SW em 1V	48
Figura 21 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$ e $D = 0.5$. Trigger ativado pela SW em 1V	48
Figura 22 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$ e $D = 0.25$. Trigger ativado pela SW em 1V	49

Figura 23 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$ e $D = 0.25$. Trigger ativado pela SW em 1V	50
Figura 24 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$ e $D = 0.75$. Trigger ativado pela SW em 1V	50
Figura 25 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$ e $D = 0.75$. Trigger ativado pela SW em 1V	51
Figura 26 – Diagrama de Bode com $MF = 60$ e $W_c = 2\pi 100$	52
Figura 27 – Diagrama de Bode $MF = 75$ e $W_c = 2\pi 100$	53
Figura 28 – Conversor de Boost malha de tensão fechada utilizando a ferramenta SignalTap para simular.	54
Figura 29 – Fluxograma simplificado código de emulação do Conversor Boost com malha fechada	55
Figura 30 – RTL Viewer - Boost Malha fechada	55
Figura 31 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela SW em 1V. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$	56
Figura 32 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$. Trigger ativado pela SW em 1V. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$	57
Figura 33 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela i_L pela borda de subida. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$	58
Figura 34 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela i_L pela borda de descida. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$	59

Figura 35 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela SW em $1V$. Controle projetado com $MF = 75$ e $W_c = 2\pi 100$	60
Figura 36 – Esquemático da placa de aquisição de dados	70
Figura 37 – Desenho da placa de aquisição de dados	70
Figura 38 – Placa de aquisição de dados antes da solda	71
Figura 39 – Bancada de Testes	72

LISTA DE TABELAS

Tabela 1 – Tabela de Parâmetros do Conversor Buck	41
Tabela 2 – Tabela de Parâmetros do Conversor Boost	45

LISTA DE ABREVIATURAS E SIGLAS

CC	Corrente Contínua
CCM	Modo de condução contínua
DAC	Digital to Analog Converter
EDOs	Equações diferenciais ordinárias
FPGA	Field-programmable gate array
FTLAc	Função de transferência de malha aberta compensada
HDL	Hardware Description Language
HIL	Hardware-in-the-loop
HRPWM	High Resolution Pulse Width Modulation
PI	Proporcional - Integral
PLL	Phase-Locked Loop
PWM	Pulse Width Modulation
RC	Resistor Capacitor
RTL	Register Transfer Level
RTS	Real-time Simulation
VHDL	VHSIC Hardware Description Language

LISTA DE SÍMBOLOS

$S(t)$	Estado do transistor
MF	Margem de Fase

SUMÁRIO

1	INTRODUÇÃO	25
1.1	OBJETIVO	25
1.2	METODOLOGIA	26
2	FUNDAMENTAÇÃO TEÓRICA	27
2.1	SIMULAÇÃO EM TEMPO REAL	27
2.2	HIL	27
2.2.0.1	Comparação entre Typhoon HIL, OPAL-RT e FPGA	28
2.3	CONVERSORES	28
2.3.1	Conversor Buck	28
2.3.1.1	Controle da Corrente	31
2.3.2	Conversor Boost	32
2.4	CONTROLE PI	35
3	DESENVOLVIMENTO	37
3.0.1	Simulações de Conversores	37
3.0.2	FPGA	37
3.0.3	VHDL	38
3.0.4	Placa de Aquisição de Dados	38
3.0.4.1	Filtro RC Passa baixa	38
3.1	PWM DE ALTA RESOLUÇÃO	39
3.2	CONVERSOR BUCK	40
3.2.1	Simulação de Conversor	40
3.2.1.1	MATLAB	40
3.2.1.2	Resultado do Conversor Buck malha aberta na simulação ModelSim	40
3.2.2	Diagrama de blocos do hardware implementado - Conversor Buck	41
3.2.3	Resultado Conversor Buck malha aberta osciloscópio	43
3.3	CONVERSOR BOOST	44
3.3.1	Simulação de Conversor	44
3.3.1.1	MATLAB	44
3.3.2	Diagrama de blocos do hardware implementado do Conversor Boost em malha aberta	45
3.4	SIMULAÇÕES SIGNALTAP	47
3.4.1	Resultado Conversor Boost malha aberta osciloscópio	47
3.5	MODELAGEM DINÂMICA DO CONTROLE DO CONVERSOR BOOST	52
3.5.1	Modelagem do Controle de tensão do Conversor Boost	52
3.6	SIMULAÇÕES SIGNALTAP	54
3.6.1	Diagrama de blocos do hardware implementado- Conversor Boosts Malha fechada	54

3.6.2	Resultado Conversor Boost em malha fechada no osciloscópio .	56
4	CONCLUSÃO	61
	REFERÊNCIAS	63
	APÊNDICE A – CÓDIGOS	66
A.1	CONVERSOR BUCK	66
A.2	CONVERSOR BOOST	66
A.3	CONTROLE DE TENSÃO	67
	APÊNDICE B – PLACA DE AQUISIÇÃO DE DADOS	70
	APÊNDICE C – BANCADA DE TESTES	72

1 INTRODUÇÃO

Nas últimas décadas, o uso de controladores de sinais digitais em conversores estáticos aumentou substancialmente (PENCZEK *et al.*, 2011). Seguindo essa tendência, o ensino de eletrônica de potência para agregar cada vez mais elementos de controle digital em várias áreas da educação. As simulações também são amplamente utilizadas para fins de pesquisa e educação em Engenharia Elétrica.

Na simulação em tempo real de circuitos elétricos, as tensões e correntes do sistema simulado mostram um "relógio real", tempo de resposta equivalente ao do sistema real que está sendo modelado (BÉLANGER; VENNE; PAQUIN, 2010). A simulação de Hardware-in-the-loop (HIL) tem a vantagem de um teste ao vivo, teste de simulação digital, que pode criar o ambiente de teste realista (COLLINS *et al.*, 1998). Os testes são realizados em tempo real, para que seja possível garantir que o modelo funcione na prática. Como os sinais gerados pelo modelo do sistema são implementados no Field-programmable gate array (FPGA), eles podem ser facilmente analisados (PENCZEK *et al.*, 2011). Fig. 1 mostra o diagrama de blocos funcional da aquisição de dados. O FPGA envia os dados digitais para um Digital to Analog Converter (DAC) e, em seguida, é possível medir a corrente e a tensão no osciloscópio. A medida dos sinais no osciloscópio é muito importante, isso garante que o HIL tenha sido projetado corretamente. Também é possível medir indiretamente a corrente e a tensão emuladas no FPGA.

Figura 1 – Diagrama simplificado da aquisição de dados



Fonte: Do autor.

1.1 OBJETIVO

Estudo, desenvolvimento e implementação de algoritmos baseados em estratégias de modulação digital de alta resolução para conversores estáticos de potência. Os algoritmos desenvolvidos destinam-se a geração de pulsos de comandos específicos para aplicação em conversores estáticos de potência.

1.2 METODOLOGIA

A primeira etapa para a realização deste trabalho será a revisão teórica da literatura que discute propostas para o HIL e o estudo de conversores tradicionais, tais como:

- Conversor Buck
- Conversor Boost
- Controle Proporcional - Integral (PI)
- HIL

A base teórica e de pesquisa será em livros e e de artigos, a base se dará em artigos publicados em periódicos e conferências disponíveis no repositório IEEE Xplore.

Na segunda etapa, serão simulados algoritmos nos softwares MatLab e Quartus e então depois de validados, serão simulados em tempo real. Após uma quantidade razoável de cenários observada, com seus pontos altos e baixos encontrados, pretende-se incrementar o algoritmo, para aprimorar seu funcionamento. Ao finalizar esta etapa, foram realizados mais testes e, novamente, aprimorar seu funcionamento

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os principais tópicos estudados e uma explicação sobre o assunto.

- Simulação em tempo real,
- HIL
- Conversor Buck
- Conversor Boost
- Controle PI

2.1 SIMULAÇÃO EM TEMPO REAL

A simulação por computador é parte importante e, muitas vezes, obrigatória no estágio de desenvolvimento de uma nova solução de engenharia. Essa afirmação é especialmente verdadeira na área da Engenharia Elétrica, onde as grandezas físicas envolvidas são capazes de causar danos ao ambiente e ao equipamento em frações de segundo e, em geral, não são diretamente observáveis. Simulações programadas a partir de modelos matemáticos criados para circuitos específicos podem apresentar maior velocidade e fidelidade de resultados, mas exigem um grande tempo de formulação de equações para cada circuito. Um software de simulação de circuitos gerais moderno sacrifica parte da exatidão e velocidade de simulação de modo a permitir uma modelagem muito rápida, onde elementos gráficos que representam os componentes são simplesmente adicionados e interligados (MOCCELINI, 2018).

2.2 HIL

A simulação HIL é uma técnica para realizar testes em nível de sistema de sistemas embarcados de forma abrangente, econômica e repetível. Para desenvolver uma simulação HIL, você precisa de computação e entradas e saídas adequadas hardware, bem como software para realizar a modelagem de simulação em tempo real e operações de entradas e saídas (LEDIN, 1999).

Sua utilização é necessária nesse projeto pois ela possibilita resultados mais fidedignos com a construção física das máquinas elétricas estudadas.

As simulações HIL apresentam um novo conjunto de desafios para os desenvolvedores de modelos de engenharia de tráfego, pois a “correção” de um modelo em tempo real não depende apenas da computação numérica, mas da oportunidade com que o modelo de simulação interage com o controle externo equipamento. (BULLOCK *et al.*, 2004).

2.2.0.1 Comparação entre Typhoon HIL, OPAL-RT e FPGA

O *OPAL-RT* é um x86 e computador mais FPGA que roda um sistema operacional Linux em tempo real. Ele apresenta múltiplos módulos de E/S para testes HIL e um pacote de software com Algoritmos especializados Real-time Simulation (RTS) baseam-se em *Simulink Simscape Power Systems basis*(MOCCELINI, 2018).

Este dispositivo permite alterações na planta desde que solicitadas ao fabricante, o que engessa o processo gera impossibilidade de uma aplicação em várias plantas e ambientes de aprendizado. Seu passo de cálculo é de $200ns$.

O *Typhoon HIL Inc* é um *hardware* amplamente utilizado para simulações de eletrônica de potência, micro-rede e redes de distribuição, sua tecnologia é baseada em FGPA Este dispositivo não permite alterações na planta . Seu passo de cálculo é de $200ns$.

O FPGA possui frequência de $50MHz$ e com isso, seu passo de cálculo é d

O FPGA apresenta grande versatilidade de aplicações, entre elas a possibilidade de alterar a frequência de atualização dos estados de forma a cumprir com requisitos de performance e temporização. Utilizando a frequência de $50MHz$ foi possível obter um passo de cálculo de $20ns$ o qual é menor do que os passos de cálculo presentes no *Typhoon HIL* E *OPAL-RT*, além da possibilidade de mudança de planta a qualquer instante pois todo o modelo é construído internamente.

2.3 CONVERSORES

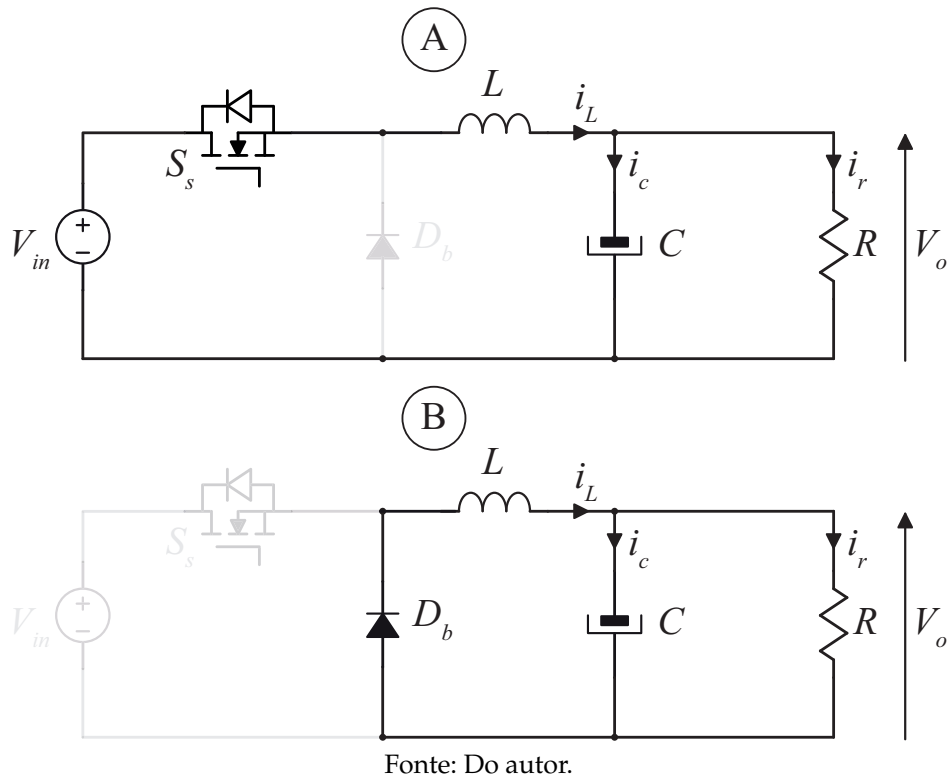
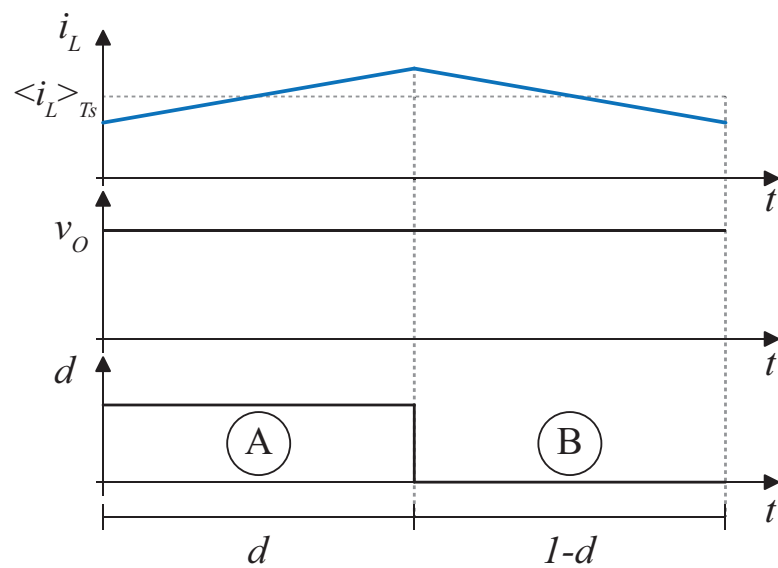
Nesta seção serão abordados os conversores Buck e Boost, estes são conversores clássicos da literatura e que possuem grande aplicabilidade.

2.3.1 Conversor Buck

Essa topologia consiste em um circuito de redução de tensão não isolado. Geralmente usado em baixa tensão Corrente Contínua (CC) em situações em que a diferença de tensão entre a entrada e a saída do conversor não é muito grande. Seu circuito básico e estágio de operação são mostrados na Fig.2. No estágio A, o transistor está fechado, a direção atual passa pelo indutor e alimenta a carga. No estágio B, o transistor está aberto; portanto, o diodo é polarizado e entra em operação. É necessário descrever a operação do sistema e o CCM A Fig. 3 ilustra a Tensão v_o e Corrente $\langle i_L \rangle_{T_s}$ do conversor Buck durante os estágios de operação A e B. Em azul é a corrente no indutor instantâneo (i_L) e tracejada é a corrente média no indutor ($\langle i_L \rangle_{T_s}$).

Em uma primeira abordagem, algumas simplificações são adotadas, os dispositivos semicondutores são considerados interruptores ideais e a resistência dos componentes intrínsecos e os efeitos parasitas são negligenciados (MARTINS; BARBI, 2000). Existem dois estados topológicos em operação do conversor Buck, eles são mostrados

Figura 2 – Fases de operação do conversor Buck em CCM.

Figura 3 – Tensão v_o e Corrente $\langle i_L \rangle_{T_s}$ do conversor Buck

na Fig. 2. No estágio A, o transistor está conduzindo, o fluxo de corrente vai para o indutor.

É possível determinar a aplicação do método da malha de tensão(1).

$$V_{in} - v_L - v_o = 0 \quad (1)$$

A tensão no indutor é dada por:

$$v_L = L \frac{\partial i_L}{\partial t} \quad (2)$$

substituindo (2) em (1).

$$V_{in} - L \frac{\partial i_L}{\partial t} - v_o = 0 \quad (3)$$

No estágio B, o transistor não conduz, então o diodo conduz. É possível determinar a aplicação do método de tensão de malha (4).

$$0 - v_L - v_o = 0 \quad (4)$$

Juntando (3) e (4) em uma equação.

$$L \frac{\partial i_L}{\partial t} = S(t)V_{in} - v_o \quad (5)$$

Onde, $S(t)$ representa o estado do transistor

$$S(t) = \begin{cases} 1, & d = 1 \\ 0, & d = 0 \end{cases} \quad (6)$$

O método de discretização de Euler é um procedimento numérico de primeira ordem para resolver Equações diferenciais ordinárias (EDOs) com um determinado valor inicial.

$$\frac{\partial f(t)}{\partial t} = \lim_{\Delta T \rightarrow 0} \frac{f(t + \Delta T) - f(t)}{\Delta T} \quad (7)$$

Modelo de um ZOH para as entradas e perturbações em (5)

$$\frac{L(i_L[n + 1] - i_L[n])}{\Delta T} = S[n]V_{in} - v_o[n] \quad (8)$$

Finalmente, implementando a discretização de Euler e isolando, a corrente do indutor i_L é:

$$i_L[n + 1] = (S[n]V_{in} - v_o[n]) \frac{\Delta T}{L} + i_L[n] \quad (9)$$

Agora, para determinar a tensão de saída, é possível determinar o método de Nós no nó de saída para determinar (10).

$$i_L = i_c + i_r \quad (10)$$

A corrente do capacitor é dada por :

$$i_c = C \frac{\partial v_C}{\partial t} \quad (11)$$

Substituindo (11) em (10) e aplicando a Lei de Ohm na corrente de saída

$$i_L = C \frac{\partial v_C}{\partial t} + \frac{v_o(t)}{R} \quad (12)$$

Finalmente, implementando a discretização de Euler (7) a tensão no Capacitor é V_o :

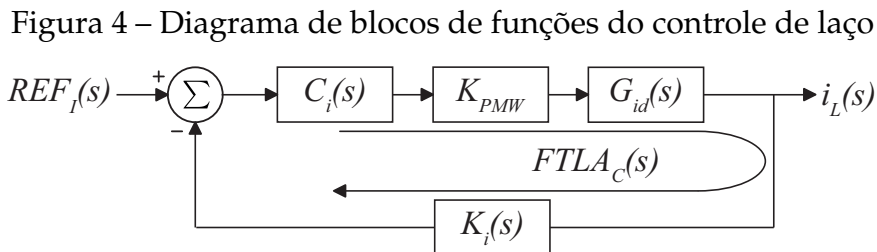
$$\frac{C(v_o[n+1] - v_o[n])}{\Delta T} = i_L[n] - \frac{v_o[n]}{R} \quad (13)$$

simplificando

$$v_o[n+1] = (i_L[n] - \frac{v_o[n]}{R}) \frac{\Delta T}{C} + v_o[n] \quad (14)$$

2.3.1.1 Controle da Corrente

O controle do indutor atual é essencial. Ele deve ter estabilidade no indutor de corrente, pois o conversor deve ser estável. A Fig. 4 mostra o esquema do controle aplicado no conversor Buck.



Fonte: Do autor.

Para o desenvolvimento do controle de laço aberto foi utilizada a função de transferência para controle de corrente no conversor buck é mostrada na equação abaixo.

$$\frac{i_L}{d} = \frac{V_{in}}{sL} \quad (15)$$

A Função de transferência de malha aberta compensada (FTLAc) é obtida pela adição da função de transferência de controlador no laço de controle, como pode ser visto na Fig. 4. Para obter as equações de projeto do controlador atual, supõem-se que um controlador seja projetado para manter o sistema de circuito fechado estável. Portanto, as condições do módulo e do ângulo apresentadas pelas equações (16) e (17) devem ser satisfeitas.

$$|FTLAc(\omega_c)| = |C_i(\omega_c)||FTLAc(\omega_c)| = 1 \quad (16)$$

$$\angle FTLAc(\omega_c) = \angle Ci(\omega_c) + \angle FTLAnc(\omega_c) = 180^\circ \quad (17)$$

No controle de corrente, o valor da frequência de cruzamento zero é geralmente definido dez vezes mais baixo que a frequência de comutação. Portanto, a frequência de comutação é atenuada adequadamente. (ARBUGERI, J. A.; ARBUGERI, C. A.; MUSSA, 2019) Encontrar os parâmetros do controlador em função das condições do módulo e do ângulo requer conhecimento da função de transferência do controlador. Esta função de transferência depende do tipo de controlador usado. Para esta aplicação, será utilizado um controlador de tipo proporcional integral (PI) com a função de transferência definida pela equação (18).

$$Ci(s) = k_s \left(\frac{s + \omega_z}{s} \right) \quad (18)$$

É importante observar que k_s e ω_z são o ganho e a frequência do controlador a ser obtido para o projeto são mostrados abaixo, onde MF é a margem da fase e ω_c é a frequência de cruzamento zero do sistema compensado.

$$k_s = \frac{\omega_c}{\sqrt{\omega_c^2 + \omega_z^2} |FTLA_{nc}(\omega_c)|} \quad (19)$$

$$\omega_z = \frac{\omega_c}{\text{tg}(MF - 90^\circ - \angle FTLA_{nc})} \quad (20)$$

2.3.2 Conversor Boost

O conversor CC-CC elevador de tensão, também conhecido como Conversor Boost, produz um valor médio de tensão de saída superior à corrente média de entrada, enquanto a corrente média de saída é inferior à corrente média de entrada.

É necessário descrever a operação do sistema e o CCM. A Fig. 6 mostra a corrente no indutor i_L e a tensão na carga. Em uma primeira abordagem, algumas simplificações são adotadas, dispositivos semicondutores são considerados comutadores ideais e a resistência dos componentes intrínsecos e os efeitos parasitários são desprezados (MARTINS; BARBI, 2000). Existem dois estados topológicos em operação do conversor Boost, eles são mostrados na Fig. 5. No estágio A, o transistor está conduzindo, o fluxo de corrente vai para o indutor (MARTINS; BARBI, 2000).

Os componentes do conversor Boost são semelhantes ao conversor Buck, sua diferença está presente em como os componentes são organizados e como o fluxo de potência flui. Seu circuito básico e estágio de operação são mostrados na Fig.5. No estágio A, o transistor está fechado, a direção da corrente vai para o indutor e alimenta a carga. No estágio B, o transistor está aberto; portanto, o diodo é polarizado e entra em operação. A Fig. 6 ilustra a Tensão v_o e Corrente $\langle i_L \rangle_{T_s}$ do conversor Boost durante os estágios de operação A e B. Em azul é a corrente no indutor instantâneo (i_L) e tracejada é a corrente média no indutor ($\langle i_L \rangle_{T_s}$).

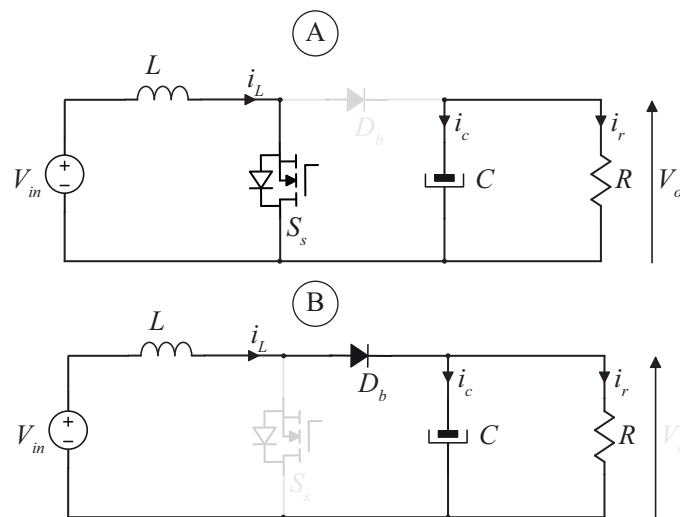


Figura 5 – Fases de operação do conversor Boost em CCM

Fonte: Do autor.

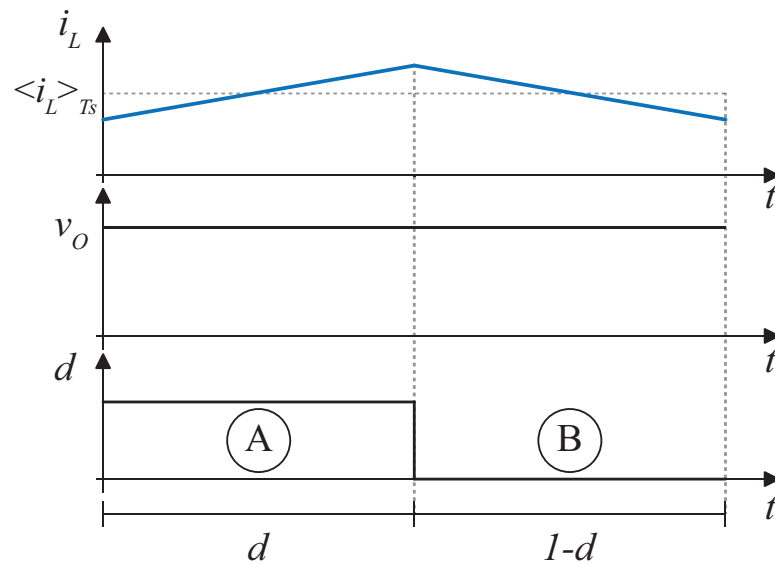


Figura 6 – Tensão v_o e corrente $\langle i_L \rangle_{T_s}$ do conversor Boost

Fonte: Do autor.

Aplicando o método de malha de tensão é possível determinar (21).

$$V_{in} - v_L - v_o = 0 \quad (21)$$

A tensão no indutor é dada por:

$$v_L = L \frac{\partial i_L}{\partial t} \quad (22)$$

Substituindo (22) em (21).

$$V_{in} - L \frac{\partial i_L}{\partial t} - v_o = 0 \quad (23)$$

No estágio B, o transistor não conduz, então o diodo conduz. Aplicando o Método de Tensão de Malha é possível determinar (24).

$$0 - v_L - v_o = 0 \quad (24)$$

Mesclando (23) e (24) na equação:

$$V_{in} - L \frac{\partial i_L}{\partial t} = v_o * (1 - S(t)) \quad (25)$$

onde, $S(t)$ representa o estado do transistor

$$S(t) = \begin{cases} 1, & d = 1 \\ 0, & d = 0 \end{cases} \quad (26)$$

Aplicando o método de discretização de Euler em (25)

$$\frac{L(i_L[n+1] - i_L[n])}{\Delta T} = S[n]V_{in} - v_o[n] \quad (27)$$

Finalmente, implementando o Método de Discretização de Euler a corrente no indutor e isolando i_L é:

$$i_L[n+1] = i_L + \frac{\Delta T}{L} (S[n]V_{in} - v_o[n]) \quad (28)$$

Agora, para determinar a tensão de saída, aplicando o Método do Nós no nó de saída é possível determinar (29).

$$i_L = i_c + i_r \quad (29)$$

A corrente sobre o condensador é dada por:

$$i_c = C \frac{\partial v_C}{\partial t} \quad (30)$$

Substituindo (30) in (10) e aplicando a Lei de Ohm, para a corrente de saída

$$i_L = C \frac{\partial v_C}{\partial t} + \frac{v_o(t)}{R} \quad (31)$$

Finalmente, aplicando o Método de Discretização de Euler, equação (7) a voltagem no capacitor é:

$$\frac{C(v_o[n+1] - v_o[n])}{\Delta T} = i_L[n] - \frac{v_o[n]}{R} \quad (32)$$

então, é possível obter:

$$v_o[n+1] = v_o[n] + (1 - s[n]) \frac{v_o[n]s[n]}{RC} + \frac{\Delta T i_L[n]}{C} - \Delta T s[n] \frac{v_o}{RC} \quad (33)$$

2.4 CONTROLE PI

É uma técnica de controle de processo que combina ação proporcional para que o sinal de erro seja minimizado. O controle PI é, de longe, a maneira mais comum de usar *feedback* em sistemas de engenharia. Aparece em dispositivos simples e em grandes fábricas com milhares de controladores. Os controladores PI aparecem em muitas formas diferentes: como controladores autônomos, como parte de sistemas de controle hierárquicos e distribuídos e embutidos em componentes. (ÅSTRÖM; MURRAY, 2021)

3 DESENVOLVIMENTO

3.0.1 Simulações de Conversores

As simulações são extremamente importantes, através das quais é possível provar e testar hipóteses estudadas, além de aprofundar o conhecimento além da literatura. Para analisar as diferentes plataformas e suas particularidades, serão realizadas simulações no software MATLAB e na VHSIC Hardware Description Language (VHDL), onde serão explorados dois ambientes de simulação, o ModelSim e o SignalTap Logic Analyzer. Os dados são apresentados pelo Embedded Logic Analyzer como formas de onda. Também é usado para depurar e verificar seu design. (QUARTUS. . . , 1999) . Neste projeto essa ferramenta foi amplamente utilizada para fins de verificação dos resultados.

3.0.2 FPGA

Um arranjo de portas programável em campo (FPGA) é um circuito integrado cujo circuito digital interno pode ser projetado usando uma linguagem de descrição de hardware a partir da qual o projetista determina quais lógicas de portas serão sintetizadas no FPGA, bem como suas interconexões (VAHID, 2009). O método de implementação

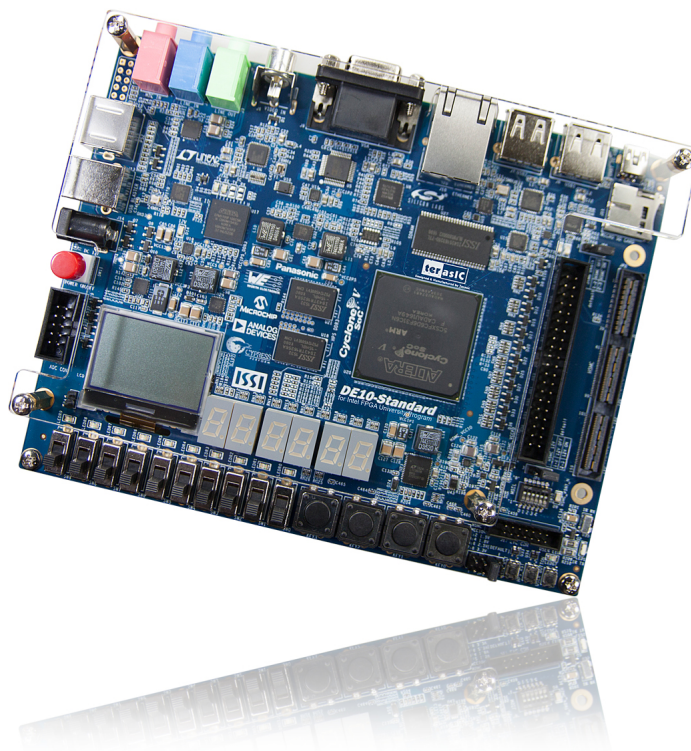


Figura 7 – DE10-Standard

Fonte: www.terasic.com.tw

do FPGA de componentes físicos para simulação ou controle de hardware no circuito

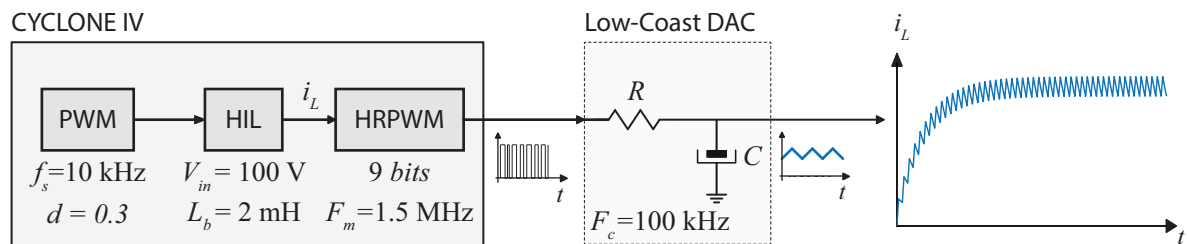
(HIL) é comumente usado, inclusive como uma ferramenta educacional.

3.0.3 VHDL

A Linguagem de Descrição de Hardware de Circuito Integrado de Alta Velocidade (VHDL) é uma linguagem de descrição de hardware usada para descrever sistemas com alguns benefícios, tais como: permite a descrição de um sistema simultâneo e os projetos são polivalentes (NAVABI, 1993). Inicialmente, o sistema foi simulado usando o ModelSim, que é uma ferramenta excelente para códigos VHDL. Um banco de testes foi criado para reduzir os erros antes dos testes na plataforma baseada em FPGA.

A Fig. 8 se refere ao digrama da bancada de testes. Internamente ao FPGA, o Pulse Width Modulation (PWM) de 10 kHz é gerado para o modelo HIL, a corrente do indutor é modulada em um HRPWM para modular o sinal em alta frequência, para que possa ser filtrado usando um filtro Resistor Capacitor (RC), reduzindo custos e simplificando o projeto para avaliação e aprendizado. Para o filtro RC foi projetada e realizada uma placa de aquisição de dados, descrita a seguir.

Figura 8 – Método da aquisição de Dados



Fonte: Do autor.

3.0.4 Placa de Aquisição de Dados

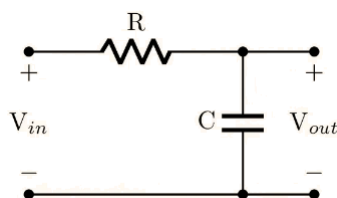
Com o objetivo de uma precisão apurada dos resultados no osciloscópio foi construído para adquirir os dados oriundos do FPGA. O dado que vem do FPGA é um sinal digital e para conseguir ler este em um osciloscópio é necessário transforma-lo em analógico, essa transformação é realizada através de um DAC projetado para a faixa de frequência utilizada. O filtro escolhido para fazer essa conversão foi o Filtro RC Passa baixa.

3.0.4.1 Filtro RC Passa baixa

O filtro passa baixa é um circuito eletrônico que permite a passagem de baixas frequências e atenua a amplitude das frequências maiores que a frequência de corte escolhida. Para esse projeto foi escolhida a frequência de corte de corte $f_c = 10\text{kHz}$. Com base nas metodologias do livro (NOCETI, 2020), concluiu-se que $R = 1\text{k}\Omega$ e $C = 15\text{nF}$.

No Apêndice é possível observar a Fig. 36 que apresenta o esquemático da placa de aquisição de dados, o desenho dela na Fig.38 e também a placa pronta depois da fresa na Fig.37.

Figura 9 – Circuito do filtro RC passa baixa



Fonte: Do autor.

3.1 PWM DE ALTA RESOLUÇÃO

Para obter um PWM com maior número de bits, e por consequência, maior resolução, foi implementada a arquitetura do HRPWM. Esta arquitetura, visível na Fig. 34 consiste em 8 relógios com 45 bits de fase um ao outro e obter um aumento de resolução de 3 bits. Nesta versão, o contador tem 6 bits e o duty-cycle tem 9 bits. (ARBUGERI, J. A.; ARBUGERI, C. A.; MUSSA, 2019).

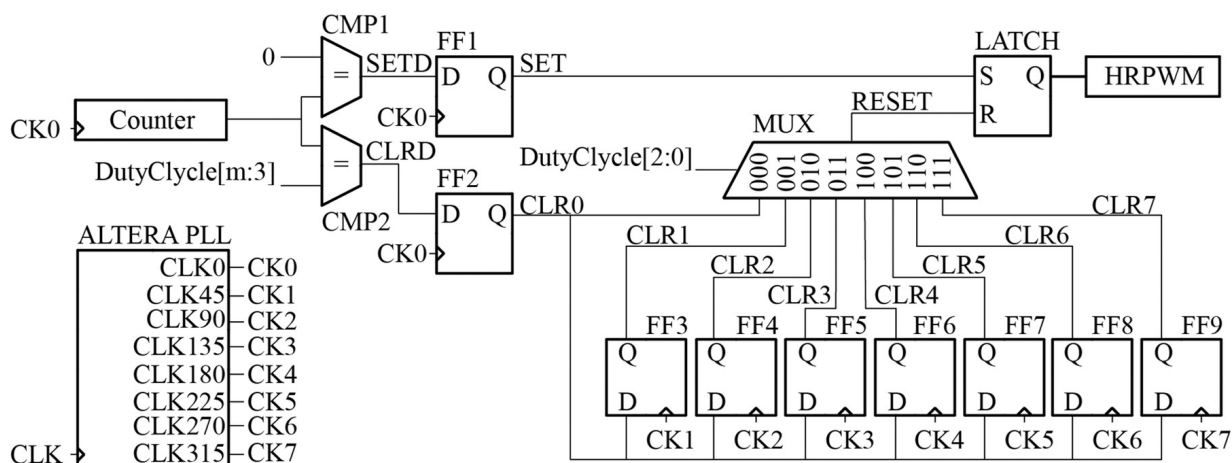
$$Resolução = 2^n \frac{f_{clk}}{f_{pwm}} \quad (34)$$

Onde n é :

$$n = \log_2 M \quad (35)$$

e M é o número do relógio deslocado; esta versão tem oito clocks então n é 3 e a resolução é aumentada em 3 bits.

Figura 10 – Arquitetura do HRPWM



Fonte: (ARBUGERI, J. A.; ARBUGERI, C. A.; MUSSA, 2019)

3.2 CONVERSOR BUCK

3.2.1 Simulação de Conversor

Primeiramente, o modelo foi implementado no MATLAB para sua validação, e em um segundo momento a simulação foi feita no ModelSim para validar a implementação do modelo no FPGA e, em seguida, os resultados experimentais foram obtidos.

3.2.1.1 MATLAB

Usando o MATLAB, foi possível simular os conversores Buck e Boost, bem como visualizar alguns de seus parâmetros. No Algoritmo abaixo, a lógica no código da corrente (i_L) e a tensão (V_c) no indutor são descritas, e i_L é o modelo de corrente comutada e $\langle i_L \rangle$ é o modelo de corrente média.

```
--Conversor Buck
for i=1:length(t)
    if rem(t(i), Ts) < D*Ts
        s(i) = 1;
    else
        s(i) = 0;
    end
end
for i = 1:length(t)-1
    iL(i+1) = iL(i) + h/L * ( -vc(i) + s(i) * Vin);
    vc(i+1) = vc(i) + h/c * ( iL(i) - vc/R );
end
```

A Fig. 11 mostra na parte superior em azul a corrente $\{i_L\}$ e em cinza a corrente média durante o período de comutação $\{i_L\}_{T_s}$. No meio, em azul, a tensão $\{v_c\}$ e em cinza, a corrente média durante o período de comutação $\{v_c\}_{T_s}$. Os parâmetros para o conversor Buck são apresentados na Tab. 1. A corrente no indutor e a tensão no capacitor apresentam o comportamento desejado de acordo com a literatura, portanto a modelagem realizada é válida.

3.2.1.2 Resultado do Conversor Buck malha aberta na simulação ModelSim

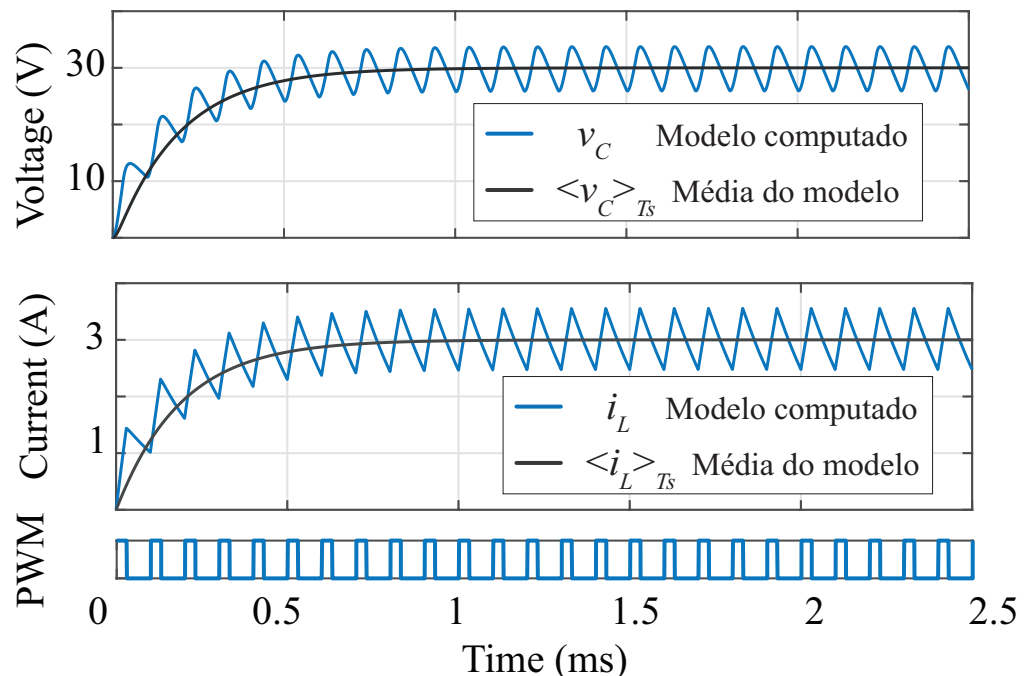
ModelSim é uma ferramenta de verificação e simulação para desenhos VHDL, Verilog, SystemVerilog, e de linguagem mista. É uma excelente aplicação para utilizar durante a aprendizagem de conceitos e práticas Hardware Description Language (HDL). A Fig. 12 mostra o conversor Buck Indutor de Tensão, Corrente e PWM usando a

Tabela 1 – Tabela de Parâmetros do Conversor Buck

Parâmetro	Símbolo	Valor
Resistor	R	10 Ω
Frequência	f_s	10 kHz
Período	T_s	100 μS
Indutor	L	2 mH
Capacitor	C	1 μF
Tensão de Entrada	V_{in}	100 V
Tensão de Saída	V_o	30 V

Fonte: Do autor.

Figura 11 – Corrente do indutor do conversor de Buck e tensão do capacitor e período de comutação utilizando o software MATLAB



Fonte: Do autor.

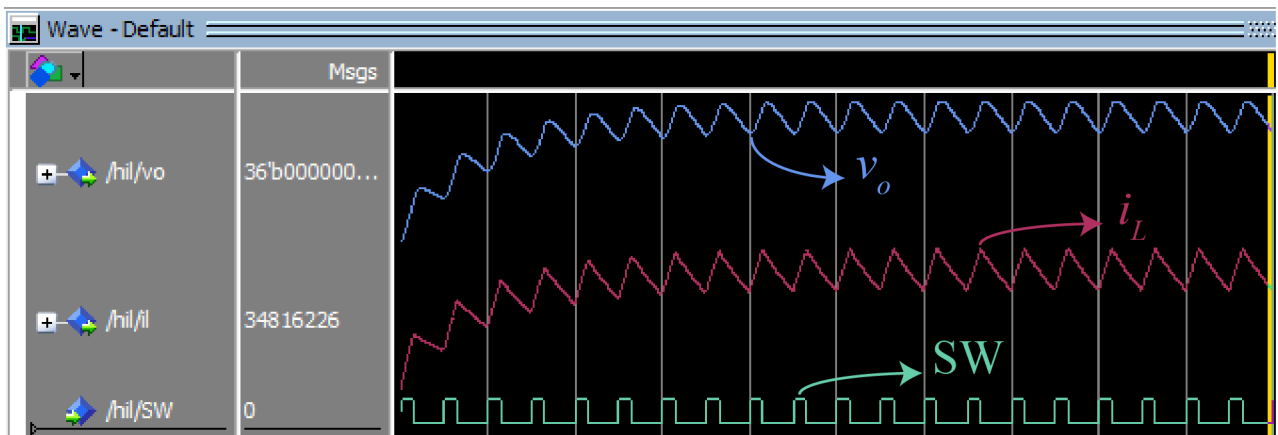
ferramenta Modelsim para simular. Em azul a tensão (v_o), em vermelho a corrente (i_L) e em verde o sinal do período de comutação (SW).

Fonte: Do autor.

3.2.2 Diagrama de blocos do hardware implementado - Conversor Buck

No *software Quartus* ao compilar o é possível obter uma visualização em blocos do funcionamento do código, essa visualização denomina-se *RTL Viewer*. No projeto de circuito integrado, uma descrição da operação de um circuito é chamada de Register Transfer Level (RTL), o comportamento do circuito é baseado no fluxo de sinal - Transferência de dados entre registradores e operações presentes no hardware lógico

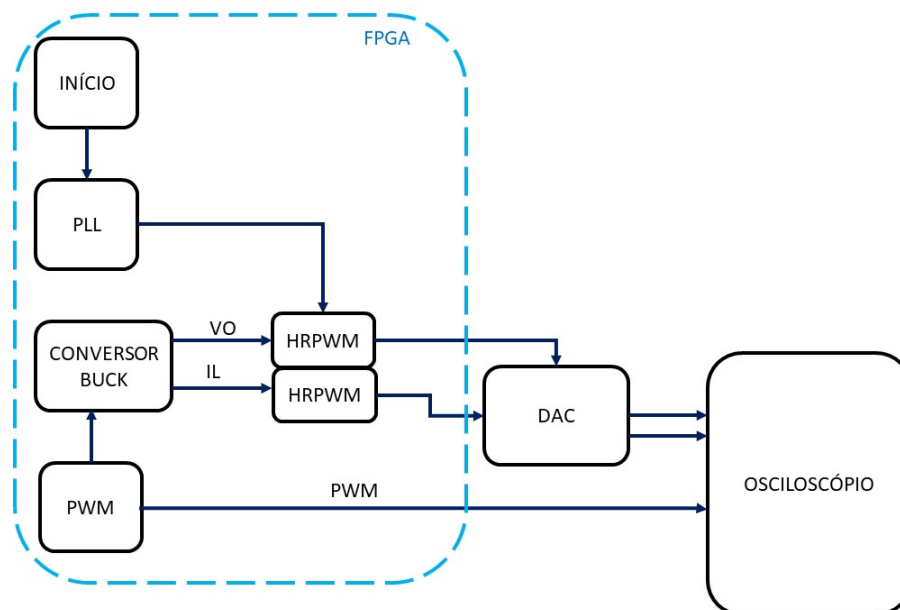
Figura 12 – Conversor Buck em Modelsim



Fonte: Do autor.

executada com estes sinais-. A abstração RTL usada em VHDL representa um circuito de baixo nível do qual as conexões de hardware são derivadas.

Figura 13 – Fluxograma simplificado código de emulação do Conversor Buck

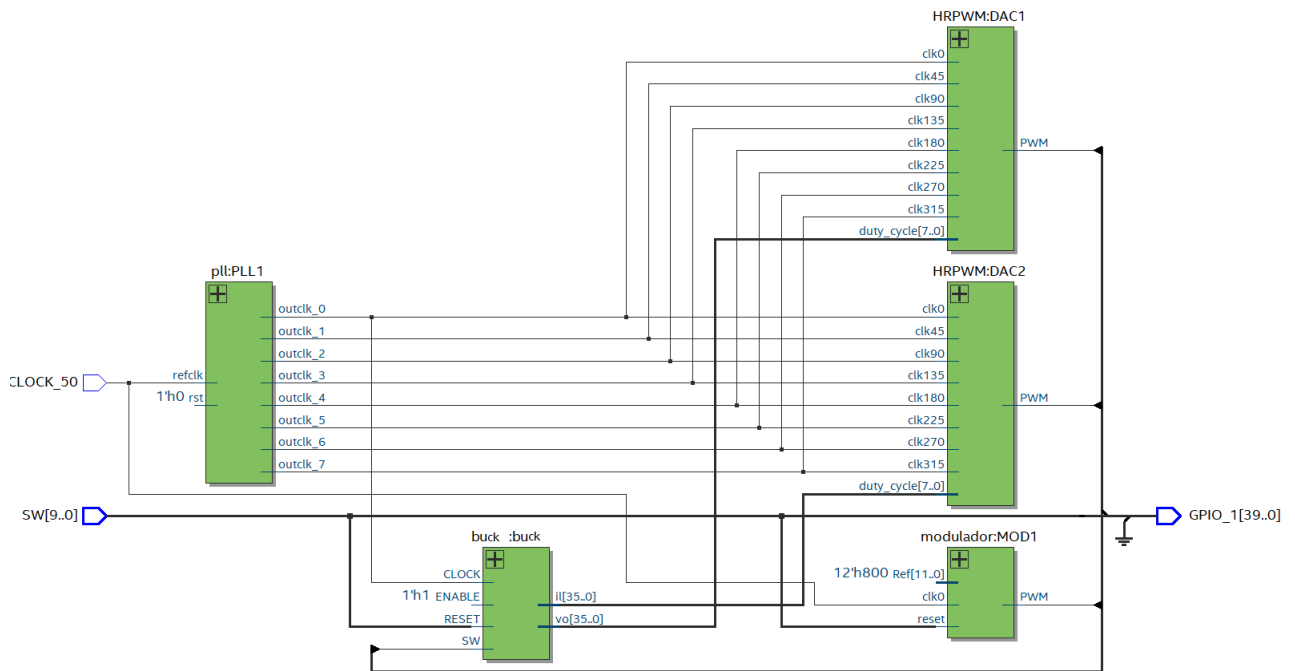


Fonte: Do autor.

A Fig. 14 apresenta a *RTL Viewer* do conversor Buck operando em malha aberta.

O Phase-Locked Loop (PLL) é um é um circuito que faz o sincronismo entre um sinal de referência (entrada) e o sinal do elo de realimentação, o qual atua para manter o sinal de clock em um valor fixo. Neste componente (pll:PLL1), a referência de entrada é de 50MHz e suas 7 saídas são de 250MHz. O componente buck:buck é o próprio conversor Buck emulado. Os componentes HRPWM:DAC1 e HRPWM:DAC2 são os PWMs de alta resolução. O componente modulador:MOD1 é o modulador PWM.

Figura 14 – RTL Viewer - Conversor Buck



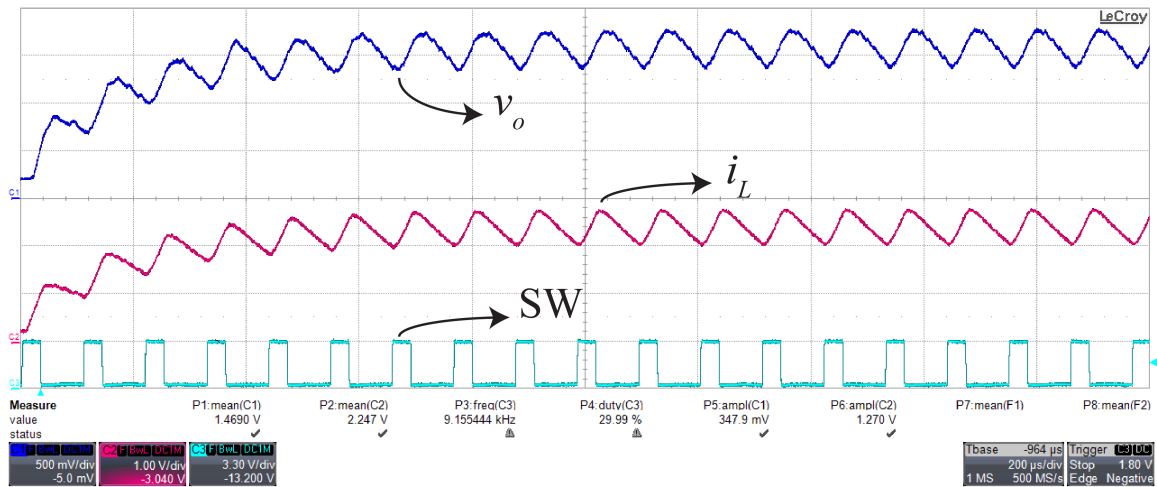
Fonte: Do autor.

3.2.3 Resultado Conversor Buck malha aberta osciloscópio

A Fig. 15 se refere Tensão do indutor do conversor Buck, corrente e período de comutação medidos usando osciloscópio. Em azul a tensão (v_o), em magenta a corrente (i_L) e em ciano o período de comutação (SW). Este resultado foi obtido com o o osciloscópio LeCroy de 200MHz de largura de banda e 4Gs/s de taxa de amostragem. Os canais 1 e 2 apresentam 1V/divisão já o canal 3 apresenta 2V/divisão. A base te tempo é de 200 μ Samples/divisão. O trigger é ativado pelo SW.

O canal 1 apresenta 500mV/divisão, o canal 2 apresenta 1V/divisão já o canal 3 apresenta 3,3V/divisão. A base te tempo é de 200 μ S/divisão

Figura 15 – Valores médios do conversor Buck de tensão (v_o), corrente (i_L) e período de comutação (SW). Em azul a tensão (v_o), em magenta a corrente (i_L) e em ciano o período de comutação (SW).



Fonte: Do autor.

3.3 CONVERSOR BOOST

3.3.1 Simulação de Conversor

3.3.1.1 MATLAB

No Algoritmo a lógica no código da corrente (i_L) e a tensão (V_c) no indutor são descritas, e i_L é o modelo de corrente comutada e $\langle i_L \rangle$ é o modelo de corrente média.

--Conversor Boost

```

for i=1:length(t)
    if rem(t(i), Ts) < D*Ts
        s(i) = 1;
    else
        s(i) = 0;
    end
end
for i = 1:length(t)-1
    il(i+1) = il(i) + (dt/L)*(s(i)*(Vin) + (~s(i))*( Vin- vc(i)));
    vc(i+1) = vc(i) + ~s(i)*(-dt*vc(i)/(R*C) + dt*il(i)/C) + s(i)*dt*(-vc(i)/(R*C));
end

```

A Fig. 16 mostra na parte superior em azul a corrente $\{i_L\}$ e em cinza a corrente média durante o período de comutação $\{i_L\}_{T_s}$. No meio, em azul, a tensão $\{v_c\}$ e em cinza, a corrente média durante o período de comutação $\{v_c\}_{T_s}$.

Os parâmetros para o conversor Boost são apresentados na se refere 2. A corrente no indutor e a tensão no capacitor apresentam o comportamento desejado de acordo com a literatura, portanto a modelagem realizada é válida.

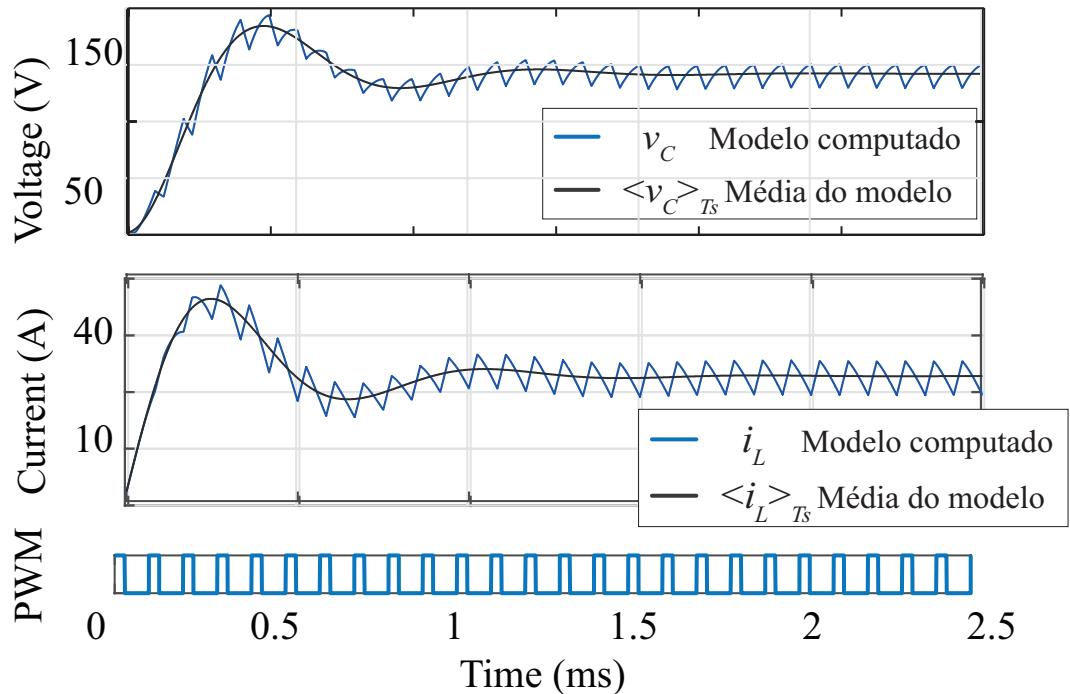


Figura 16 – Corrente do indutor do conversor de Boost e tensão do capacitor e período de comutação utilizando o software MATLAB

Fonte: Do autor.

Tabela 2 – Tabela de Parâmetros do Conversor Boost

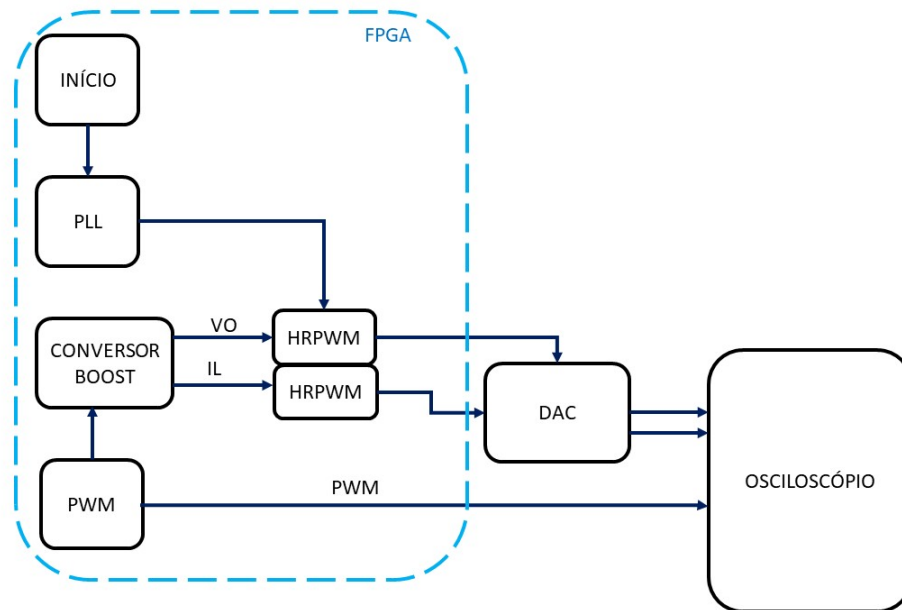
Parâmetro	Símbolo	Valor
Resistor de carga	R	10 Ω
Frequência de comutação	f_s	10 kHz
Período de comutação	T_s	100 μS
Razão Cíclica	D	0.3
Indutor	L	2 mH
Capacitor	C	20 μF
Tensão de Entrada	V_{in}	100 V
Tensão de Saída	V_o	200 V

Fonte: Do autor.

3.3.2 Diagrama de blocos do hardware implementado do Conversor Boost em malha aberta

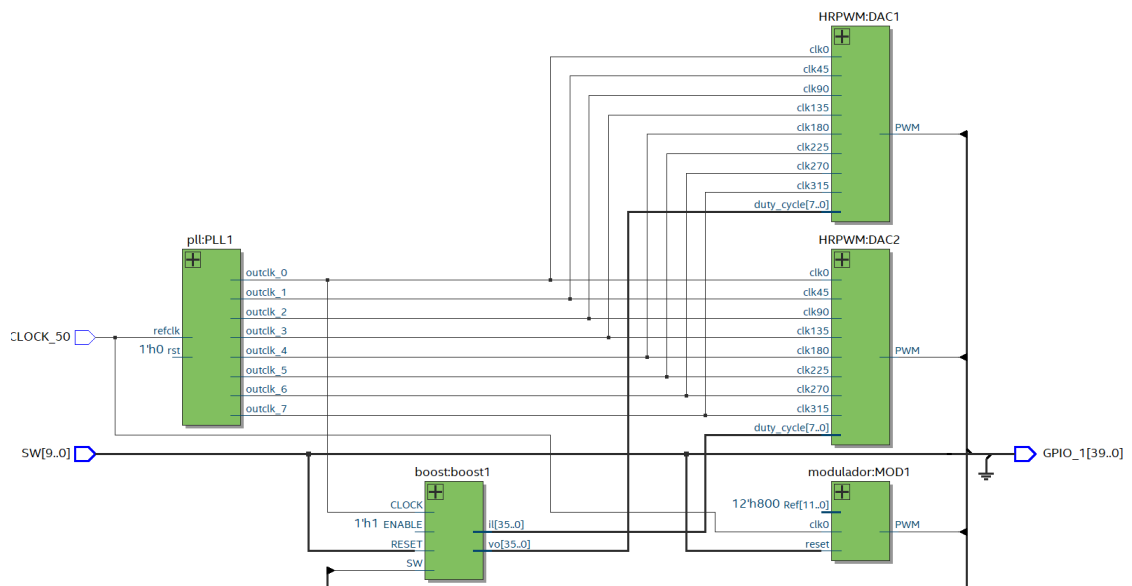
A Fig.18 apresenta a visualização RTL da programação em VHDL do Conversor Boost em malha aberta, nela é possível observar os blocos das funções de cada parte da programação.

Figura 17 – Fluxograma simplificado código de emulação do Conversor Boost



Fonte: Do autor.

Figura 18 – RTL Viewer - Boost Malha Aberta

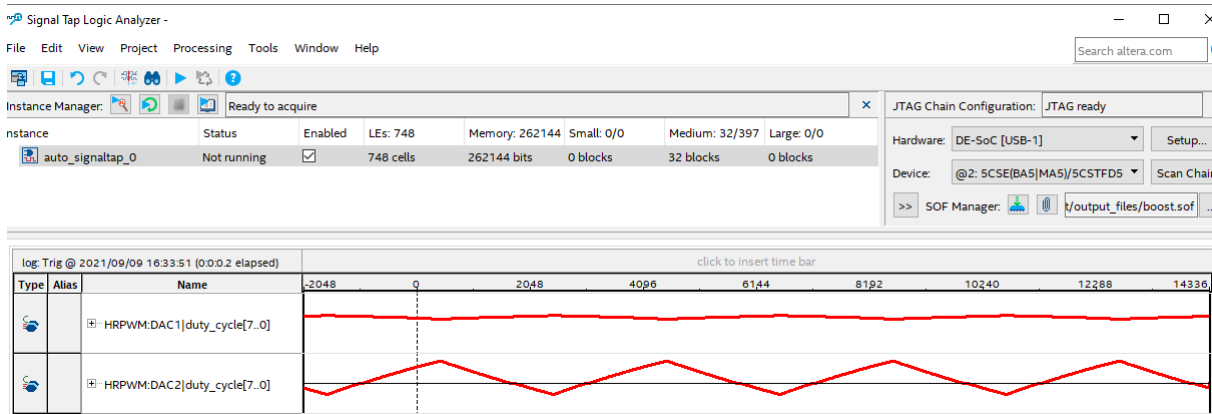


Fonte: Do autor.

O PLL é um é um circuito que faz o sincronismo entre um sinal de referência (entrada) e o sinal do elo de realimentação, o qual atua para manter o sinal de clock em um valor fixo. Neste componente (pll:PLL1), a referencia de entrada é de 50MHz e suas 7 saídas são de 250MHz . O componente boost:boost1 é o próprio conversor Boost emulado. Os componentes HRPWM:DAC1 e HRPWM:DAC2 são o PWMs de alta resolução. O componente modulador:MOD1 é o modulador PWM.

3.4 SIMULAÇÕES SIGNALTAP

Figura 19 – Conversor de Boost malha aberta utilizando a ferramenta para simular.



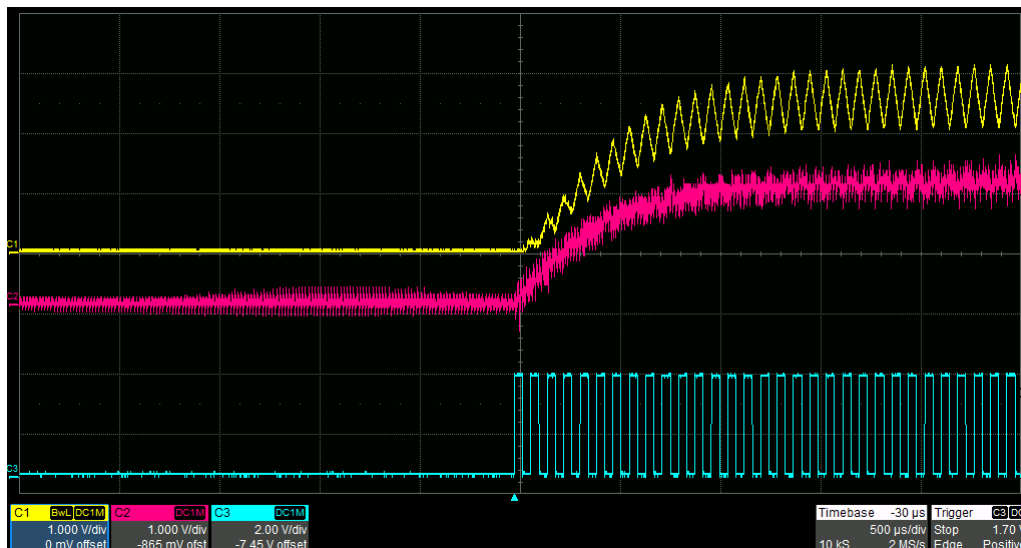
Fonte: Do autor.

3.4.1 Resultado Conversor Boost malha aberta osciloscópio

As Figs.20, 21,22, 23, 22, 25, 24, referem-se Tensão do indutor do conversor Boost, corrente e período de comutação medidos usando osciloscópio. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Todas as Figs.20,21,23,22, 25 e 24 foram adquiridas no osciloscópio LeCroy de 200MHz de largura de banda e 4Gs/s de taxa de amostragem. Os canais 1 e 2 apresentam 1V/divisão já o canal 3 apresenta 2V/divisão. A base de tempo é de 500 μ S/divisão.

A figura 20 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha aberta, nesta simulação seu período de comutação é de $D = 0.5$ e sua tensão de entrada é de $V_{in} = 100V$.

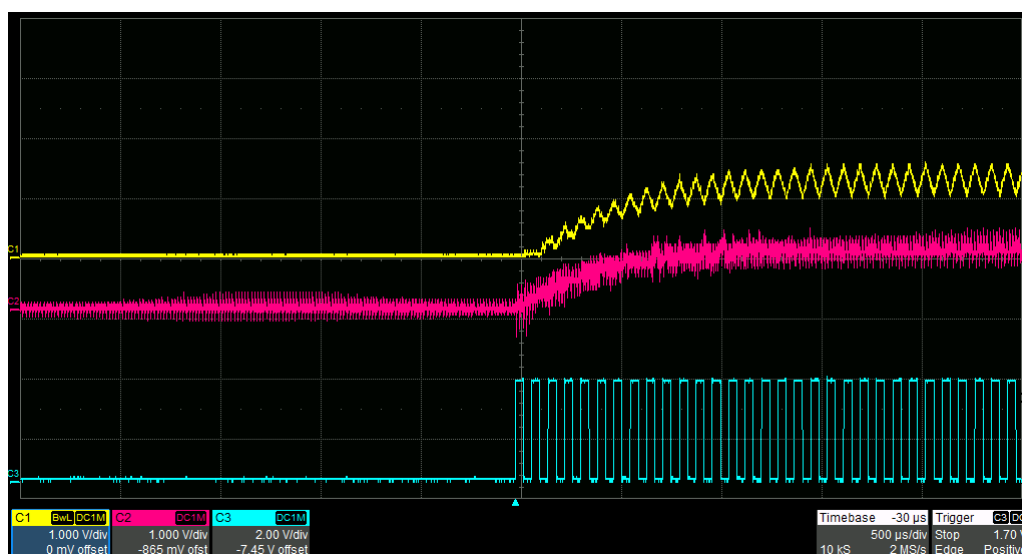
Figura 20 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$ e $D = 0.5$. Trigger ativado pela SW em 1V



Fonte: Do autor.

A figura 21 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha aberta, nesta simulação seu período de comutação é de $D = 0.5$ e sua tensão de entrada é de $V_{in} = 50V$.

Figura 21 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$ e $D = 0.5$. Trigger ativado pela SW em 1V

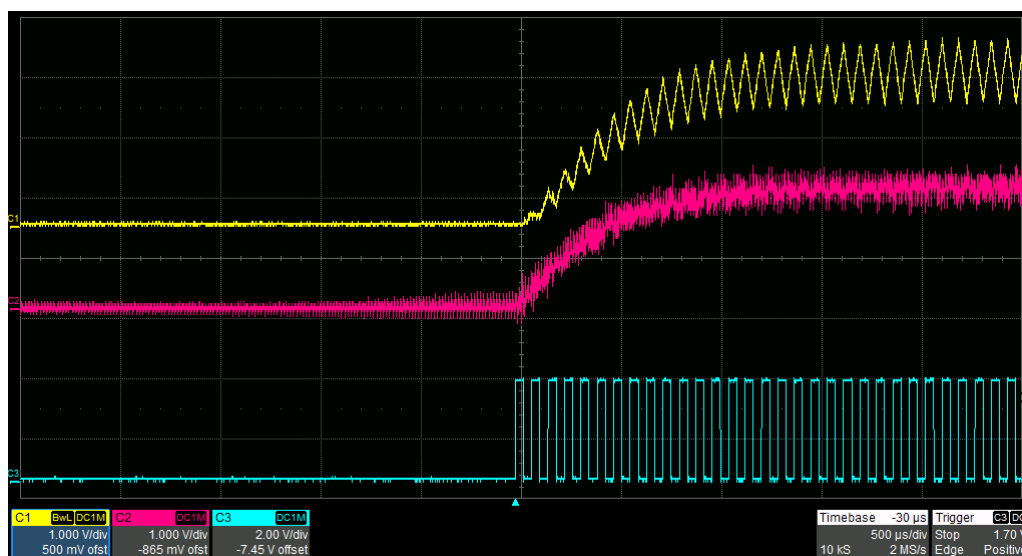


Fonte: Do autor.

A figura 22 se refere ao resultado obtido no osciloscópio da simulação em tempo

real do Conversor Boost em malha aberta, nesta simulação seu período de comutação é de $D = 0.25$ e sua tensão de entrada é de $V_{in} = 100V$.

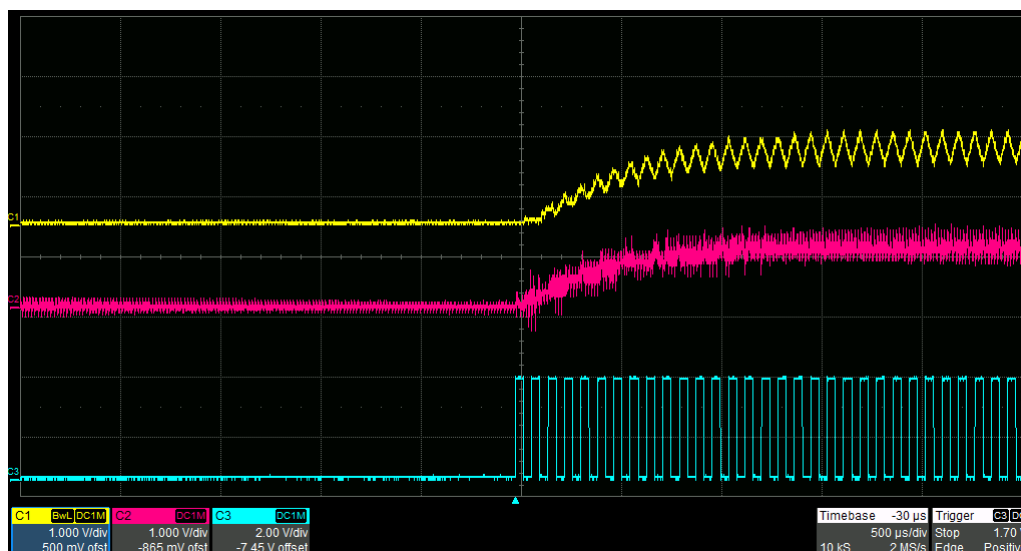
Figura 22 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$ e $D = 0.25$. Trigger ativado pela SW em 1V



Fonte: Do autor.

A figura 23 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha aberta, nesta simulação seu período de comutação é de $D = 0.25$ e sua tensão de entrada é de $V_{in} = 50V$.

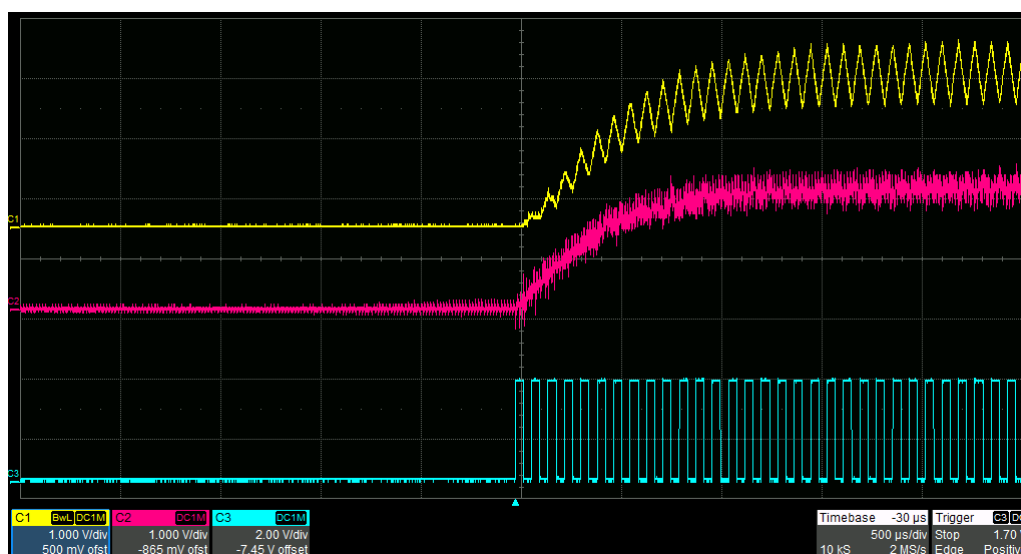
Figura 23 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$ e $D = 0.25$. Trigger ativado pela SW em 1V



Fonte: Do autor.

A figura 24 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha aberta, nesta simulação seu período de comutação é de $D = 0.75$ e sua tensão de entrada é de $V_{in} = 100V$.

Figura 24 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$ e $D = 0.75$. Trigger ativado pela SW em 1V

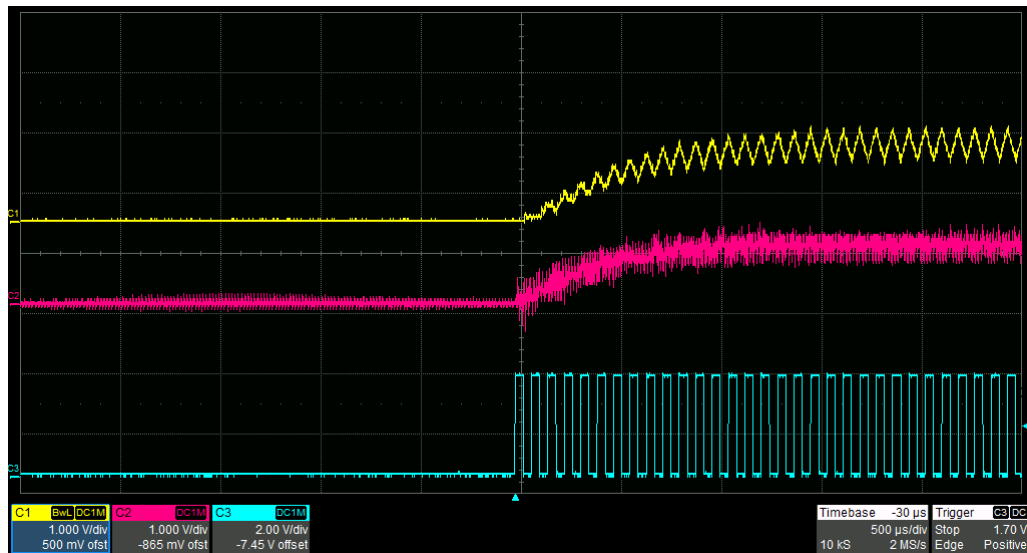


Fonte: Do autor.

A figura 25 se refere ao resultado obtido no osciloscópio da simulação em tempo

real do Conversor Boost em malha aberta, nesta simulação seu período de comutação é de $D = 0.75$ e sua tensão de entrada é de $V_{in} = 5V$.

Figura 25 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$ e $D = 0.75$. Trigger ativado pela SW em 1V



Fonte: Do autor.

3.5 MODELAGEM DINÂMICA DO CONTROLE DO CONVERSOR BOOST

O projeto de modelagem do conversor Boost foi realizado conforme o (ARBUGERI, J. A., 2019). Foi projetado o modelo de grandes sinais e o modelo de pequenos sinais para então ser possível obter a função de transferência de tensão de saída por razão cíclica orientada ao controle (Eq. 36) e a função de transferência de corrente no indutor por razão cíclica orientada ao controle

$$G_{vd} = \frac{vo}{d} = \frac{V_{in}R - sRIL}{s^2RLC + sL + (1 - D)^2R} \quad (36)$$

$$G_{id} = \frac{il}{d} = \frac{V_o(2 + sRC)}{s^2RLC + sL + (1 - D)^2R} \quad (37)$$

3.5.1 Modelagem do Controle de tensão do Conversor Boost

Os parâmetros escolhidos para a modelagem do conversor Boost são apresentados na Tab. 2, a Fig 26 apresenta o Diagrama de Bode da função de transferência. A Eq.38 é referente a função de transferência com os valores desejados. A MF utilizada para este projeto foi de $MF = 60$ e a frequência de corte foi de $W_c = 2\pi 100$

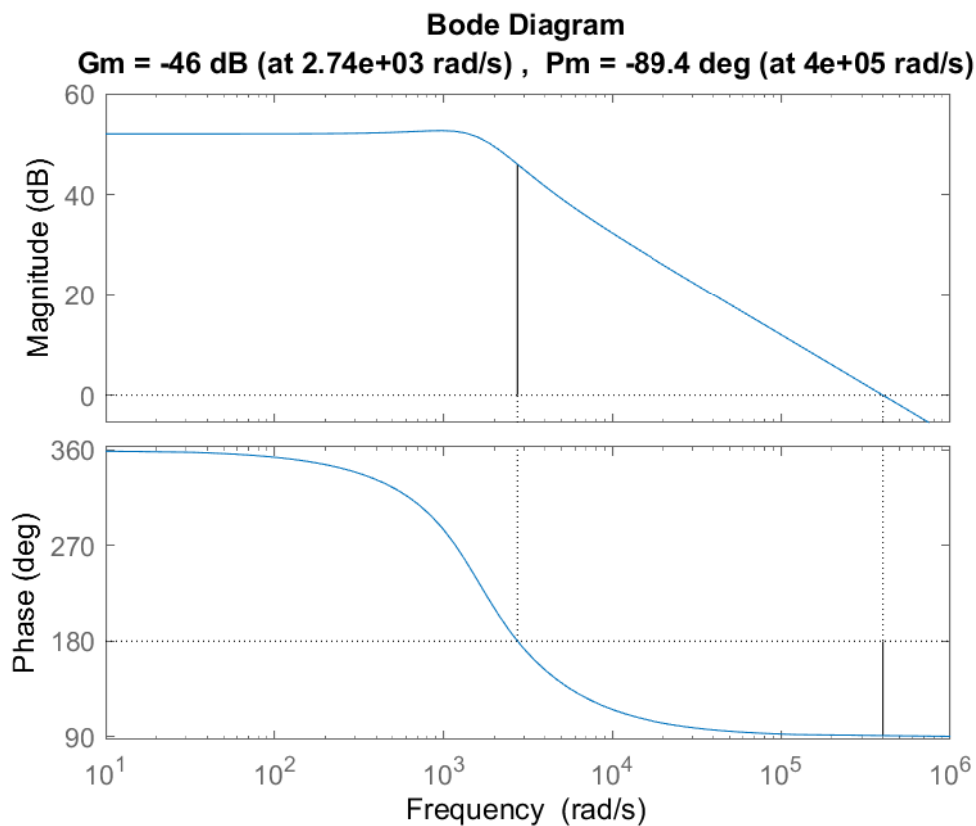


Figura 26 – Diagrama de Bode com $MF = 60$ e $W_c = 2\pi 100$

Fonte: Do autor.

$$G_{vd} = \frac{-0,4s + 1000}{1 \times 10^{-6}s^2 + 2 \times 10^{-3}s + 2,5} \quad (38)$$

Função do PI função de transferência em tempo contínuo

$$PI_i(s) = \frac{6,131 \times 10^{-3}s + 1,444}{s} \quad (39)$$

Aplicando o Método de Tustin na Eq.39 é obtida a função de tempo discreto.

$$PI_i(z) = \frac{6,52 \times 10^{-3} - 5,409 \times 10^{-3}}{z - 1} \quad (40)$$

Utilizando a Solução de Equações de Diferença no domínio Z, a solução de domínio temporal da equação é obtida tomando a sua forma de transferência-Z inversa, com a Eq. 40, é possível obter

$$Y[n] = 6,52 \times 10^{-3}X[n] - 5,409 \times 10^{-3}X[n - 1] + Y[n - 1] \quad (41)$$

Com o objetivo de aferir se o controle era o desejado, foi realizado outro projeto de controle. Foi os mesmos parâmetros do conversor Boost são apresentados na Tab. 2, mas com a utilizada para este projeto foi de $MF = 75$ e a frequência de corte foi de $W_c = 2\pi 100$. A Fig 27 apresenta o Diagrama de Bode da função de transferência.

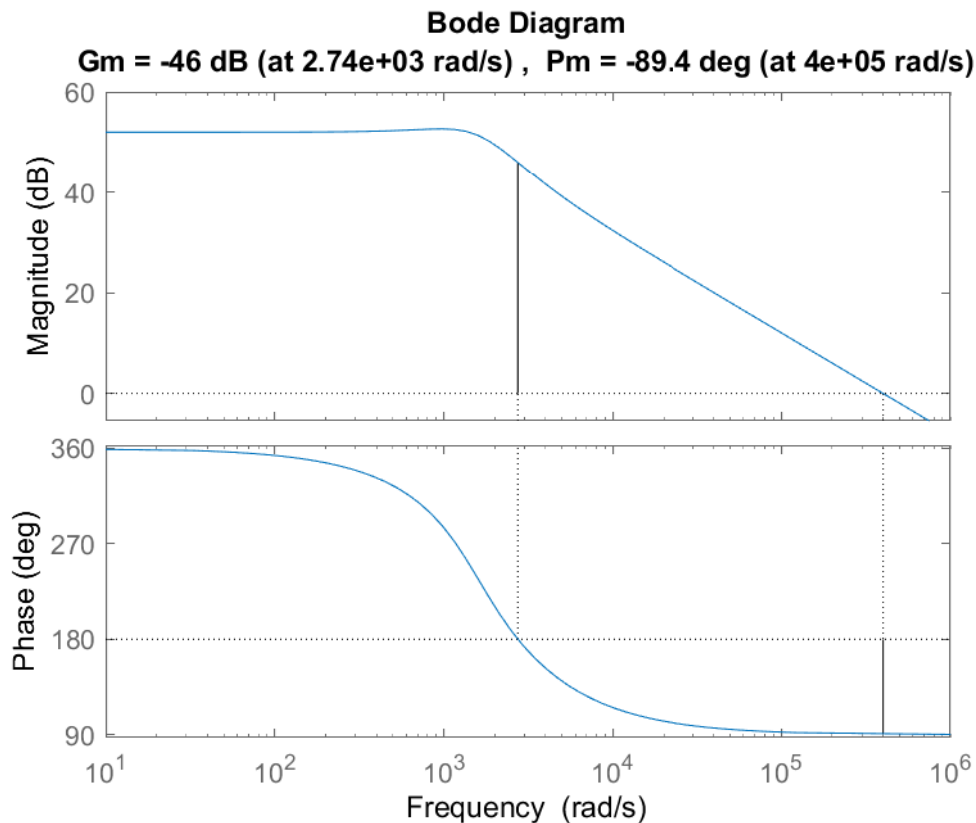


Figura 27 – Diagrama de Bode $MF = 75$ e $W_c = 2\pi 100$

Fonte: Do autor.

Função do PI função de transferência em tempo contínuo

$$PI_i(s) = \frac{1,187 \times 10^{-3}s + 1,295}{s} \quad (42)$$

Aplicando o Método de Tustin na Eq.42 é obtida a função de tempo discreto.

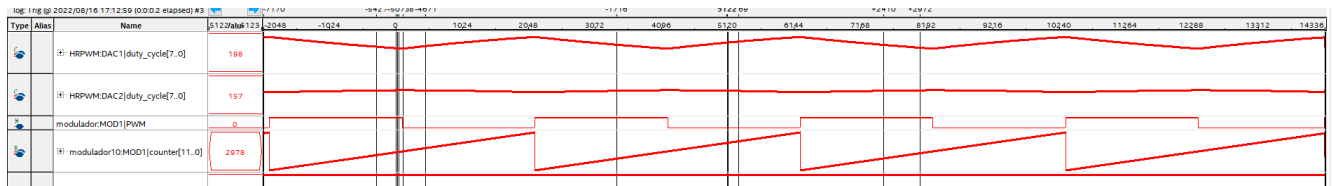
$$PI_i(z) = \frac{1,252 \times 10^{-3}z - 1,22 \times 10^{-3}}{z - 1} \quad (43)$$

Utilizando a Solução de Equações de Diferença no domínio Z, a solução de domínio temporal da equação é obtida tomando a sua forma de transferência-Z inversa, com a Eq. 43

$$Y[n] = 1,252 \times 10^{-3}X[n] - 1,22 \times 10^{-3}X[n - 1] + Y[n - 1] \quad (44)$$

3.6 SIMULAÇÕES SIGNALTAP

Figura 28 – Conversor de Boost malha de tensão fechada utilizando a ferramenta SignalTap para simular.

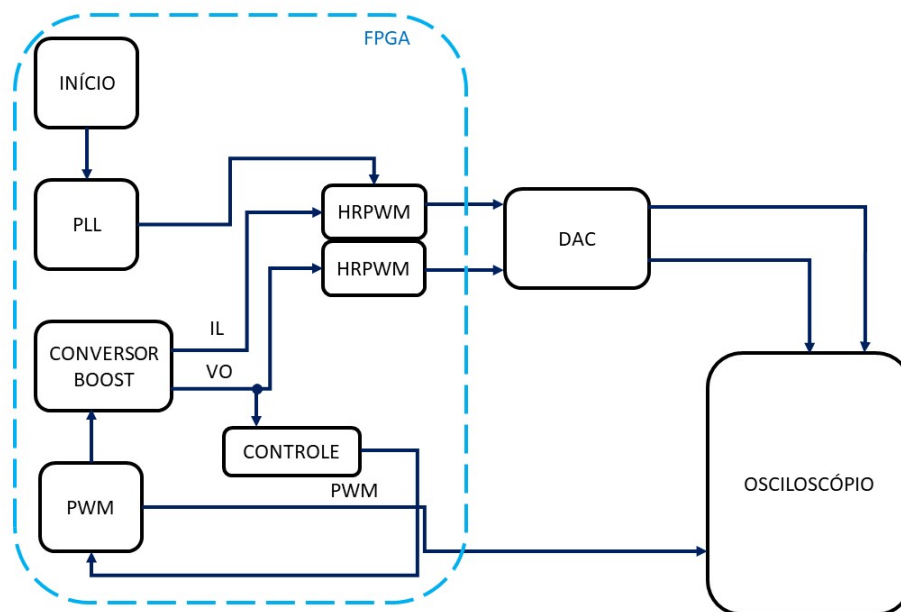


Fonte: Do autor.

3.6.1 Diagrama de blocos do hardware implementado- Conversor Boosts Malha fechada

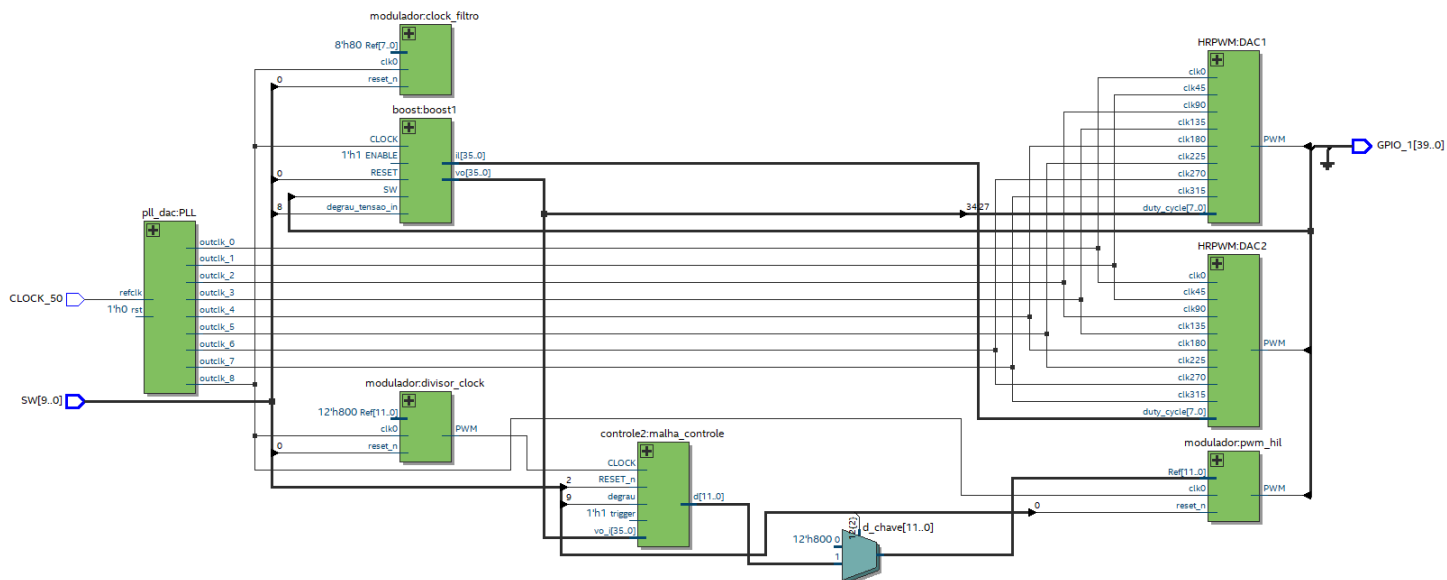
A Fig.30 apresenta a visualização RTL da programação em VHDL do Conversor Boost em malha de controle de tensão fechada, nela é possível observar os blocos das funções de cada parte da programação.

Figura 29 – Fluxograma simplificado código de emulação do Conversor Boost com malha fechada



Fonte: Do autor.

Figura 30 – RTL Viewer - Boost Malha fechada



Fonte: Do autor.

O PLL é um é um circuito que faz o sincronismo entre um sinal de referência (entrada) e o sinal do elo de realimentação, ele atua para manter o sinal de clock em um valor fixo. Neste componente (pll:PLL1), a referencia de entrada é de 50MHz e suas 7 saídas são de 250MHz. O componente boost:boost1 é o próprio conversor Boost emulado. Os componentes HRPWM:DAC1 e HRPWM:DAC2 são o PWMs de alta resolução. O

componente modulador:MOD1 é o modulador PWM. O componente controle2:malha controle é o controle de tensão aplicado.

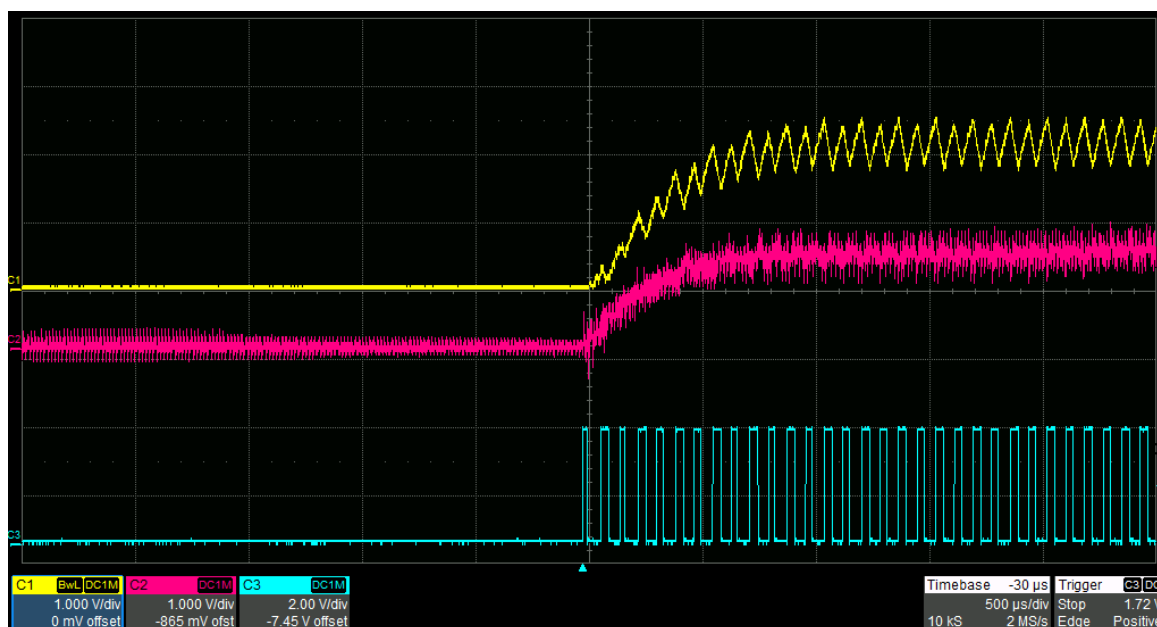
3.6.2 Resultado Conversor Boost em malha fechada no osciloscópio

As Figs.31,32 referem-se Tensão do indutor do conversor Boost, corrente e período de comutação medidos usando osciloscópio. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Com o com o controle de parâmetros $MF = 60$ e $W_c = 2\pi 100$

As Figs. 31,32, 33, 33, 35 desta seção foram adquiridas no osciloscópio LeCroy de 200MHz de largura de banda e 4Gs/s de taxa de amostragem. Os canais 1 e 2 apresentam 1V/divisão já o canal 3 apresenta 2V/divisão. A base te tempo é de 500 μ S/divisão

A figura 31 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha de tensão fechada, nesta simulação sua tensão de entrada é de $V_{in} = 100V$. Este conversor é controlado pela Eq.41 e apresenta $MF = 60$ e $W_c = 2\pi 100$

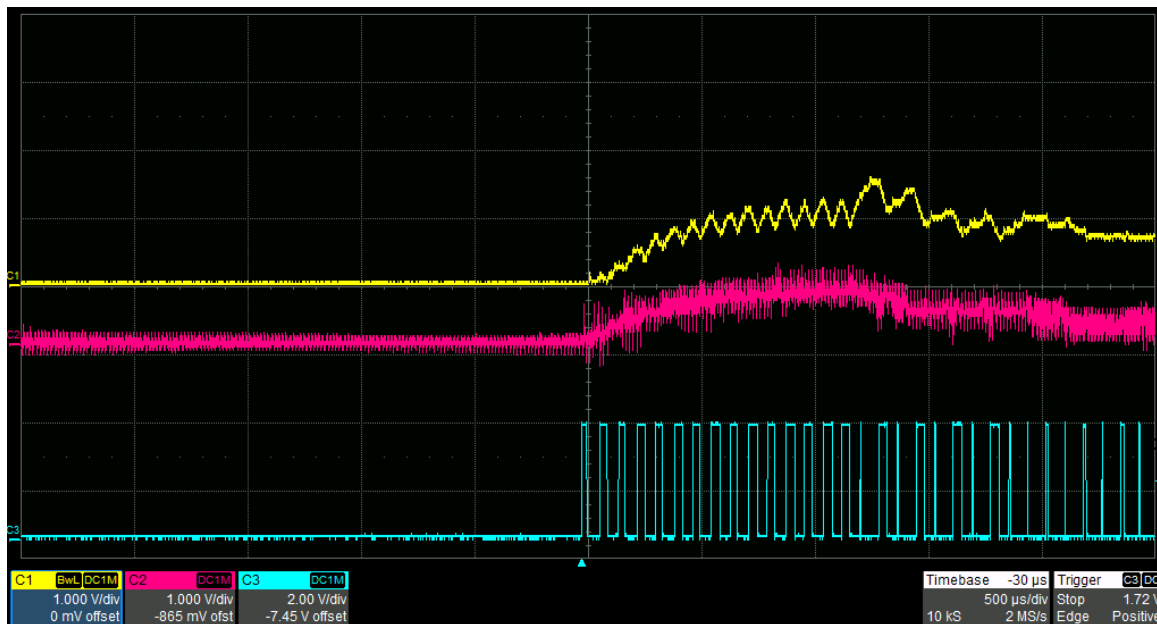
Figura 31 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela SW em 1V. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$



Fonte: Do autor.

Fig. 32 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha de tensão fechada, nesta simulação sua tensão de entrada é de $V_{in} = 50V$. Este conversor é controlado pela Eq.41 e apresenta $MF = 60$ e $W_c = 2\pi 100$.O resultado do controle atuando no conversor foi conforme o esperado.

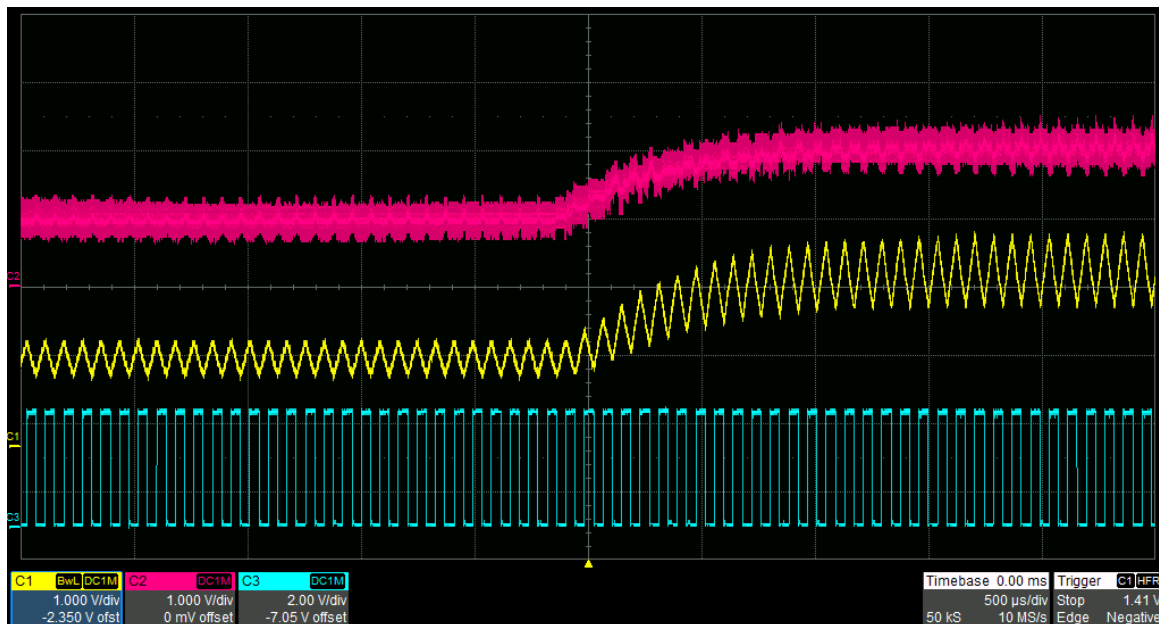
Figura 32 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 50V$. Trigger ativado pela SW em 1V. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$



Fonte: Do autor.

A Fig. 33 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha de tensão fechada, nesta simulação sua tensão de entrada é de $V_{in} = 50V$. Este conversor é controlado pela Eq.41 e apresenta $MF = 60$ e $W_c = 2\pi 100$.

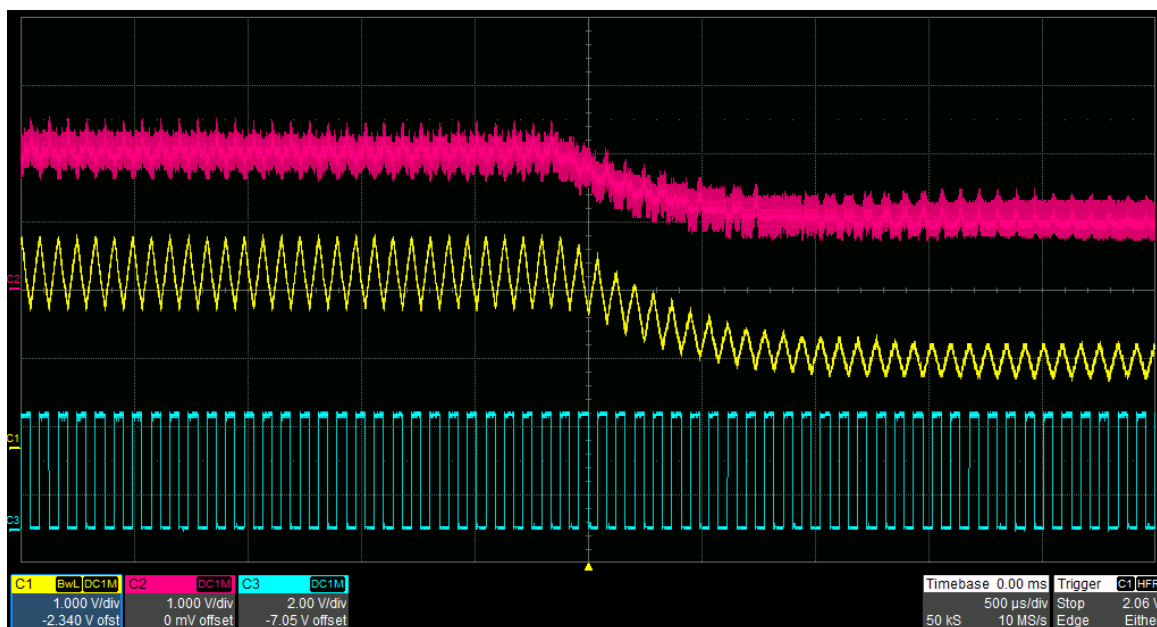
Figura 33 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela i_L pela borda de subida. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$



Fonte: Do autor.

A Fig. 34 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha de tensão fechada, nesta simulação sua tensão de entrada é de $V_{in} = 50V$. Este conversor é controlado pela Eq.41 e apresenta $MF = 60$ e $W_c = 2\pi 100$.

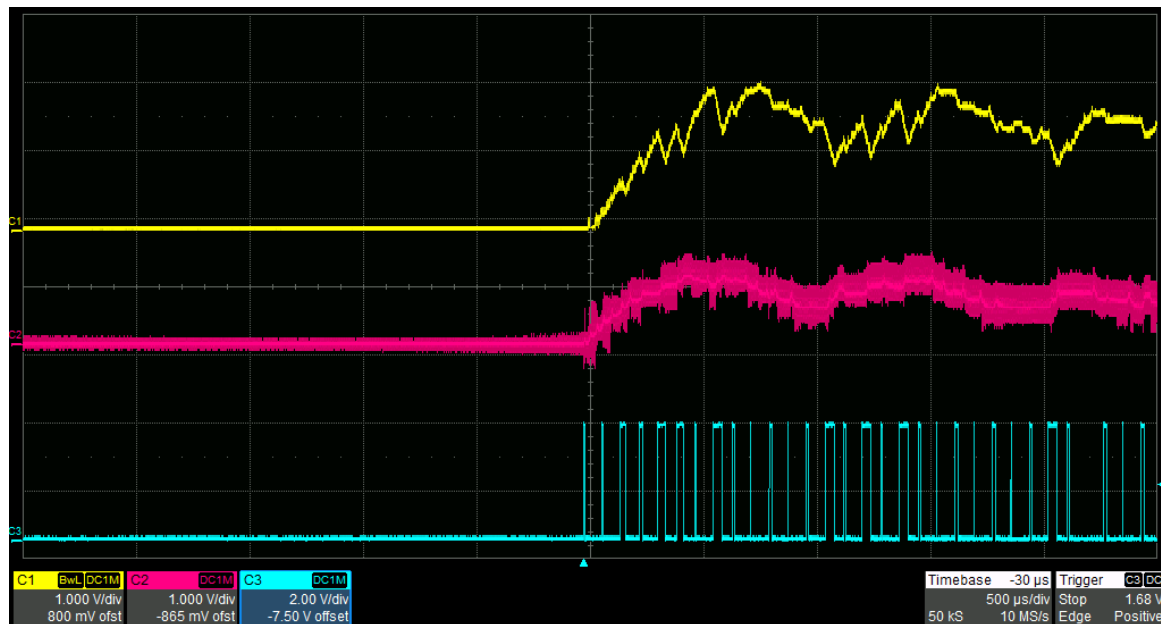
Figura 34 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela i_L pela borda de descida. Controle projetado com $MF = 60$ e $W_c = 2\pi 100$



Fonte: Do autor.

A Fig. 35 se refere ao resultado obtido no osciloscópio da simulação em tempo real do Conversor Boost em malha de tensão fechada, nesta simulação sua tensão de entrada é de $V_{in} = 50V$. Este conversor é controlado pela Eq.44 e apresenta $MF = 75$ e $W_c = 2\pi 100$. Observa-se que o controle com $MF = 60$ e $W_c = 2\pi 100$ funciona de maneira mais adequada em comparação ao com os parâmetros de $MF = 75$ e $W_c = 2\pi 100$

Figura 35 – Tensão, Corrente no indutor e período de comutação do Conversor de Boost. Em amarelo a tensão (v_o), em rosa a corrente (i_L) e em ciano o período de comutação (SW). Esta figura, se refere ao conversor com $V_{in} = 100V$. Trigger ativado pela SW em $1V$. Controle projetado com $MF = 75$ e $W_c = 2\pi 100$



Fonte: Do autor.

4 CONCLUSÃO

As simulações apresentadas são ferramentas extremamente importantes e úteis para o poder de ensino e aprendizagem de eletrônica de potência. Compreensibilidade e mutabilidade são as principais características, portanto, é possível analisar o circuito desejado a partir de vários ângulos. A simulação no software MATLAB apresentou os resultados esperados, validando o modelo apresentado. E a simulação na ferramenta ModelSim valida o código VHDL; finalmente, o resultado do osciloscópio valida os conversores *Buck- em malha aberta-*, *Boost -em malha aberta- e Boost - malha fechada de tensão-* conversor em tempo real.

O modelo pode ter os parâmetros alterados, o que permite ampla aplicabilidade, reduzindo assim o risco de testes reais no ambiente educacional. Para além do ambiente educacional, o modelo permite a fácil substituição de parâmetros para aplicabilidade em testes reais. A implementação da simulação HIL também permite testes de verificação antes da construção física do conversor, o que reduziu os custos do projeto usando o FPGA e um DAC de baixo custo, também foi possível verificar a funcionalidade do conversor.

Com a utilização do FPGA e a frequência de 50MHz seu passo de cálculo é de 20ns o que é menor do que os passos de cálculo presentes no *Typhoon HIL E OPAL-RT*, além da possibilidade de mudança de planta a qualquer instante pois todo o modelo é construído internamente.

REFERÊNCIAS

ALTERA. **Quartus SignalTap User's Guide - SignalTap™ User's Guide**. [S.l.], 1999.

ARBUGERI, José Augusto. **Retificador PFC operando em Alta frequência empregando semicondutores GaN: aspectos de projeto, layout, modulação e controle**. 2019. F. 153. Monografia (Trabalho de Conclusão de Curso (graduação)) – Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia Eletrônica, Florianópolis.

ARBUGERI, José Augusto; ARBUGERI, Cesar Augusto; MUSSA, Samir Ahmad. High-resolution FPGA-pulse width modulation applied to PFC 2 MHz converter using eGaN field effect transistor. **The Journal of Engineering, IET**, v. 2019, n. 17, p. 4205–4208, 2019.

ÅSTRÖM, Karl Johan; MURRAY, Richard M. **Feedback systems: an introduction for scientists and engineers**. [S.l.]: Princeton university press, 2021.

BÉLANGER, Jean; VENNE, P; PAQUIN, Jean-Nicolas. The what, where and why of real-time simulation. **Planet Rt**, v. 1, n. 1, p. 25–29, 2010.

BULLOCK, Darcy; JOHNSON, Brian; WELLS, Richard B; KYTE, Michael; LI, Zhen. Hardware-in-the-loop simulation. **Transportation Research Part C: Emerging Technologies**, Elsevier, v. 12, n. 1, p. 73–89, 2004.

COLLINS, D Louis; ZIJDENBOS, Alex P; KOLLOKIAN, Vasken; SLED, John G; KABANI, Noor J; HOLMES, Colin J; EVANS, Alan C. Design and construction of a realistic digital brain phantom. **IEEE transactions on medical imaging, IEEE**, v. 17, n. 3, p. 463–468, 1998.

LEDIN, Jim A. Hardware-in-the-loop simulation. **Embedded Systems Programming, MILLER FREEMAN INC.**, v. 12, p. 42–62, 1999.

MARTINS, Denizar Cruz; BARBI, Ivo. **Conversores CC-CC Básicos Não Isolados**. [S.l.]: Universidade Federal de Santa Catarina, 2000.

MOCCELINI, Marcos Paulo. **Real-time simulation of power electronics systems and hardware-in-the-loop applications**. 2018. Diss. (Mestrado) – Universidade Federal de Santa Catarina.

NAVABI, Zainalabedin. **VHDL: Analysis and modeling of digital systems**. [S.l.]: McGraw-Hill New York, 1993. v. 2.

NOCETI, Sidnei Filho. **Filtros Seletores de Sinais**. 4. ed. UFSC: Editora da UFSC, dez. 2020. v. 1. ISBN 978853280852.

PENCZEK, Adam; STALA, Robert; STAWIARSKI, Łukasz; SZAREK, Miłosz. Hardware-in-the-Loop FPGA-based simulations of switch-mode converters for research and educational purposes. **Przeład Elektrotechniczny**, v. 87, n. 11, p. 194–200, 2011.

VAHID, Frank. **Sistemas Digitais**. [S.l.]: Bookman Editora, 2009.

Apêndices

APÊNDICE A – CÓDIGOS

A.1 CONVERSOR BUCK

```

process BUCK
  begin
    if switch = '1' then
      p_input <= Vin - Vo;
    else
      p_input <= -Vo;
    end if;
  end process;
process
  begin
    if rst = '0' then
      current <= (others => '0');
      voltage <= (others => '0');
    elsif rising_edge(clk) and en = '1' then
      if current >= 0 then
        --Current
        current <= resize(shift_right(G_N*current
          + H_N*p_input, QN_BASE), DATA_WIDTH);
        --Voltage
        voltage <= voltage + resize(shift_right
          (H_Nc*(current - resize(shift_right
            (H_nr*voltage, QN_BASE);
          else
            current <= (others => '0');
          end if;
        end if;
      end process;

```

A.2 CONVERSOR BOOST

```

process BOOST
  begin
    if switch = '1' then
      p_input <= Vin - Vo;
    else
      p_input <= -Vo;

```

```

        end if;
end process;
process
begin
    if rst = '0' then
        current <= (others => '0');
        voltage <= (others => '0');
    elsif rising_edge(clk) and en = '1' then
        if current >= 0 then
            --Current
current <= resize(shift_right(G_N*current
+ H_N*p_input, QN_BASE), DATA_WIDTH);
        else
            current <= (others => '0');
        end if;
    end if;
    --Voltage
voltage <= voltage + resize(shift_right
(H_Nc*(current - resize(shift_right
H_nr*voltage, QN_BASE));

end process;

```

A.3 CONTROLE DE TENSÃO

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
use ieee.math_real.all;

entity controle2 is
generic
(
DATA_WIDTH           : natural := 36;
QN_BASE              : natural := 16;--27;
N_PWM                : natural := 12;
Q                    : natural := 134217728;
A                    : real := 6.852481920302758e-04;
B                    : real := 5.4089e-4

```

```

);
port
(
CLOCK                : in std_logic;
RESET_n              : in std_logic;
trigger              : in std_logic;
vo_i                 : in signed(DATA_WIDTH-1 downto 0);
d                    : out std_logic_vector(N_PWM-1 downto 0);
degrau               : in std_logic
);
end entity controle2;
architecture rtl of controle2 is
--      Constants
constant ref_Q : signed(DATA_WIDTH-1 downto 0) :=
to_signed(200 * (2 ** QN_BASE) , DATA_WIDTH);

constant A_Q   : signed(DATA_WIDTH-1 downto 0) :=
to_signed( integer(A * real(2 ** QN_BASE)) , DATA_WIDTH);
constant B_Q   : signed(DATA_WIDTH-1 downto 0) :=
to_signed( integer(B * real(2 ** QN_BASE)) , DATA_WIDTH);
-- Signals
signal clk, rst : std_logic;
-- Sinais PI
signal ref_i: signed(DATA_WIDTH-1 downto 0) :=
to_signed(0,DATA_WIDTH);
signal erro      : signed(DATA_WIDTH-1 downto 0) :=
to_signed(0, DATA_WIDTH);
signal x1,x0,y1,y0 : signed(DATA_WIDTH-1 downto 0) :=
to_signed(0, DATA_WIDTH);
signal sat : std_logic := '0';
      begin
          clk <= CLOCK;
          rst <= RESET_n;

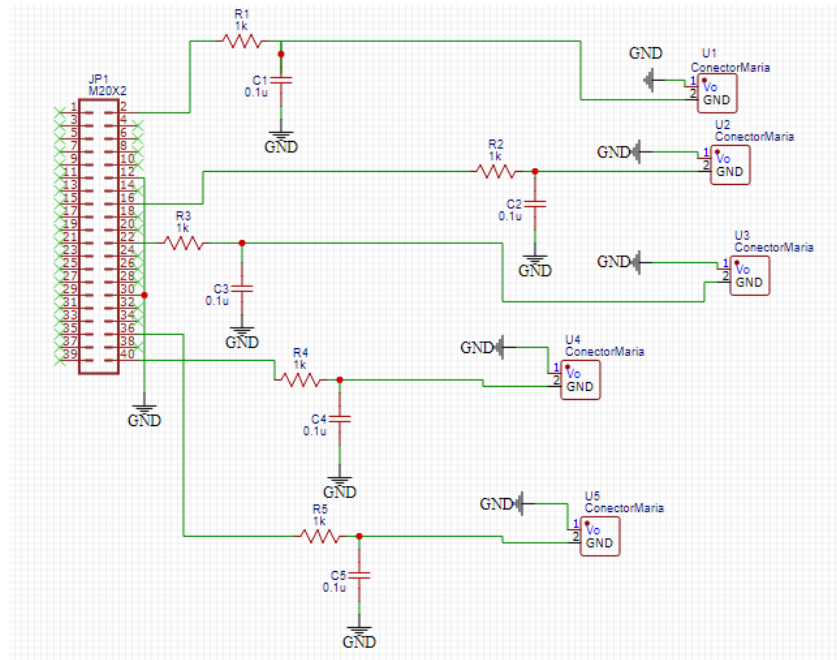
controle_proc : process(clk, rst)
begin
if rst = '0' then --0
erro <= to_signed(0,DATA_WIDTH);
x1 <= to_signed(0,DATA_WIDTH);

```

```
x0 <= to_signed(0,DATA_WIDTH);
Y1 <= to_signed(0,DATA_WIDTH);
y0 <= to_signed(0,DATA_WIDTH);
d <= (others => '0');
elsif rising_edge(clk) and trigger = '1' then
  -- Computa erro
  erro <= ref_Q - shift_right(vo_i,27-16);
  -- PI
  x1 <= x0;
  x0 <= erro;
  y1 <= y0;
  y0 <= resize(shift_right(A_Q * x0, QN_BASE),DATA_WIDTH) -
  resize(shift_right(B_Q * x1, QN_BASE),DATA_WIDTH) + y1;
  -Saida
  d <= std_logic_vector( resize(shift_right(
    (y0, QN_BASE-N_PWM), N_PWM ));
    if y0 < 0 then
  d <= (others => '0');
  sat<= '1';
elsif y0 > 2**N_PWM-1 then
  d <= (others => '1');
    sat <= '1';
else
  d <= std_logic_vector(resize(shift_right(
  y0, QN_BASE-N_PWM), d'length));
  sat <= '0';
end if;
```

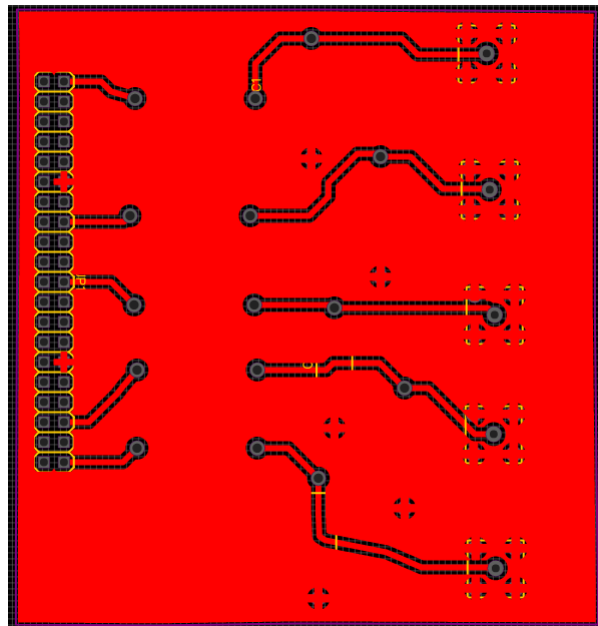
APÊNDICE B – PLACA DE AQUISIÇÃO DE DADOS

Figura 36 – Esquemático da placa de aquisição de dados



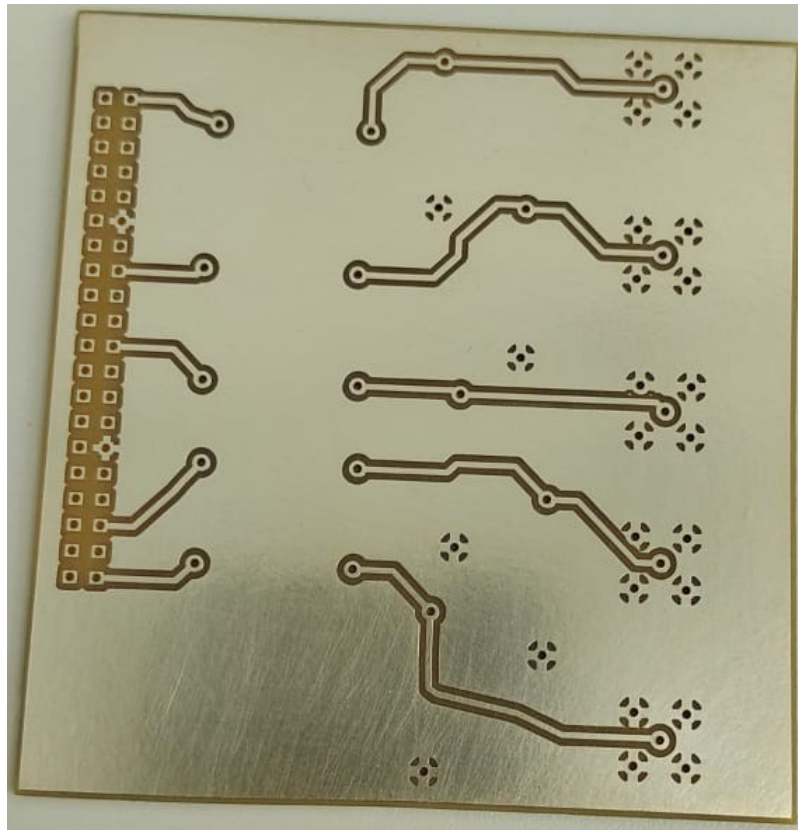
Fonte: Do autor.

Figura 37 – Desenho da placa de aquisição de dados



Fonte: Do autor.

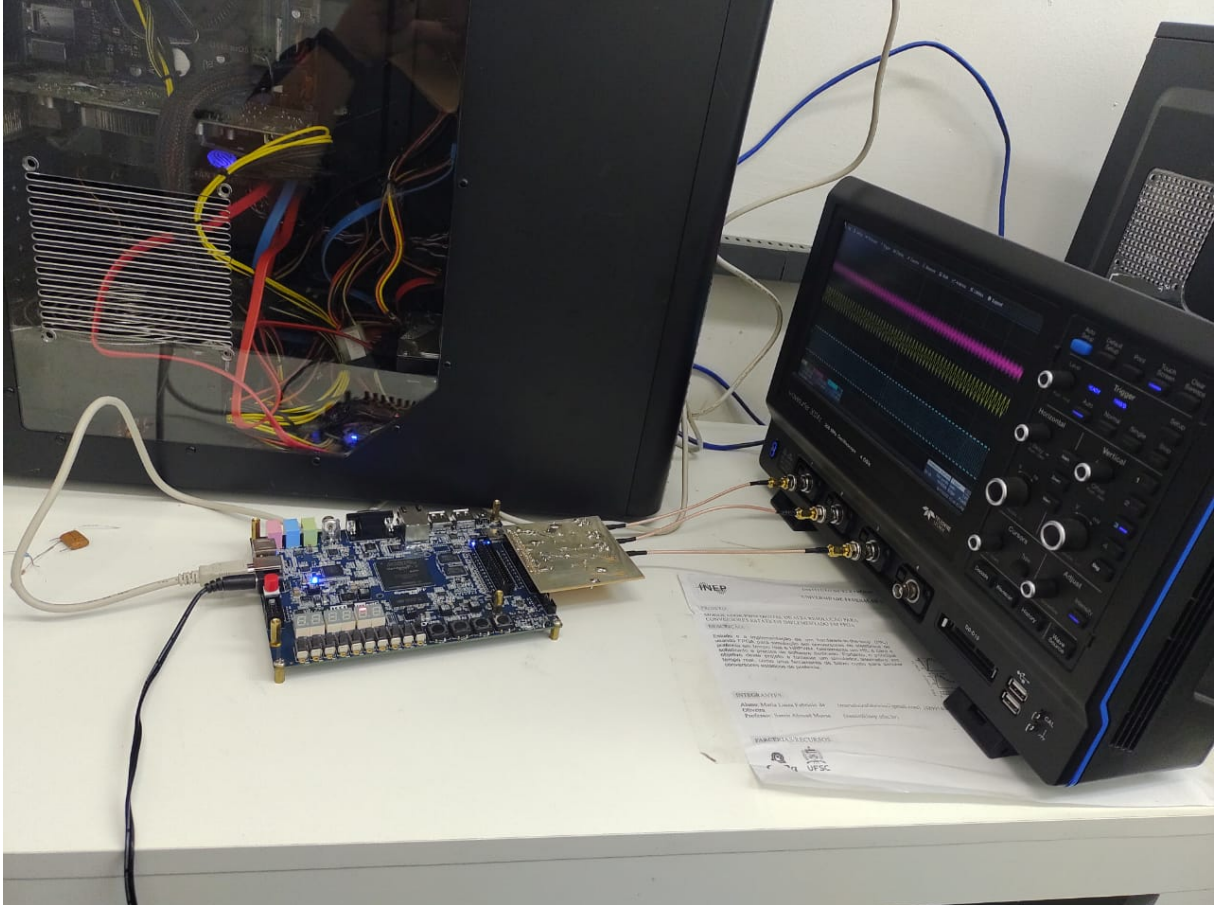
Figura 38 – Placa de aquisição de dados antes da solda



Fonte: Do autor.

APÊNDICE C – BANCADA DE TESTES

Figura 39 – Bancada de Testes



Fonte: Do autor.