

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO**

Milton Brenner Machado Matoso

**DESENVOLVIMENTO DE UM KIT DIDÁTICO
COMPOSTO POR UM CONVERSOR CC-CC BUCK
PARA ENSINO DE ELETRÔNICA DE POTÊNCIA E
CONTROLE DIGITAL DE SISTEMAS**

Santa Maria, RS, Brasil

2024

Milton Brenner Machado Matoso

**DESENVOLVIMENTO DE UM KIT DIDÁTICO COMPOSTO POR UM
CONVERSOR CC-CC BUCK PARA ENSINO DE ELETRÔNICA DE POTÊNCIA E
CONTROLE DIGITAL DE SISTEMAS**

Trabalho de Conclusão de Curso apresentado
ao Curso de Engenharia de Controle e
Automação, da Universidade Federal de Santa
Maria (UFSM, RS), como requisito parcial
para obtenção do título de Engenheiro de
Controle e Automação.

Orientador: Prof. Dr. Rafael Concatto Beltrame

Santa Maria, RS
2024

Milton Brenner Machado Matoso

**DESENVOLVIMENTO DE UM KIT DIDÁTICO COMPOSTO POR UM
CONVERSOR CC-CC BUCK PARA ENSINO DE ELETRÔNICA DE POTÊNCIA E
CONTROLE DIGITAL DE SISTEMAS**

Trabalho de Conclusão de Curso apresentado
ao Curso de Engenharia de Controle e
Automação, da Universidade Federal de Santa
Maria (UFSM, RS), como requisito parcial
para obtenção do título de Engenheiro de
Controle e Automação.

Aprovada em 20 de junho de 2024.

Rafael Concatto Beltrame, Dr. (UFSM)
(Presidente/Orientador)

Jorge Rodrigo Massing, Dr. (UFSM)

Lucas Vizzotto Bellinaso, Dr. (UFSM)

Santa Maria, RS
2024

DEDICATÓRIA

Aos meus pais, Beatriz e Milton.

RESUMO

DESENVOLVIMENTO DE UM KIT DIDÁTICO COMPOSTO POR UM CONVERSOR CC-CC BUCK PARA ENSINO DE ELETRÔNICA DE POTÊNCIA E CONTROLE DIGITAL DE SISTEMAS

AUTOR: MILTON BRENNER MACHADO MATOSO

ORIENTADOR: RAFAEL CONCATTO BELTRAME

Este Trabalho de Conclusão de Curso apresenta o desenvolvimento de um *kit* didático para ensino em Eletrônica de Potência e Controle Digital de Sistemas composto por um conversor CC-CC rebaixador *Buck*, uma placa de instrumentação e controle e o circuito de *gate-driver* para acionamento da chave semicondutora (MOSFET) do conversor. O objetivo deste trabalho é desenvolver um material didático que melhore a formação de futuros engenheiros, facilitando a consolidação do seu conhecimento durante as disciplinas mencionadas anteriormente. O desenvolvimento deste projeto envolve o estudo da topologia do conversor *Buck*, bem como o dimensionamento de seus componentes. Foi elaborado o esquemático do circuito com os componentes escolhidos e o *layout* para a produção das placas de circuito impresso (PCI). Após o teste da parte eletrônica do projeto são projetadas as malhas de controle da topologia estudada, com o desenvolvimento dos modelos matemáticos das duas malhas (corrente e tensão). A validação destes modelos é realizada com a utilização dos softwares MATLAB® e PSIM®. Após a validação é realizada a implementação das malhas de controle no conversor implementado e os resultados obtidos são comparados com os simulados.

Palavras-chave: Controle Digital, Eletrônica de Potência, Ensino, PCI, Engenharia, *Buck*, *kit* didático.

ABSTRACT

DESIGN AND IMPLEMENTATION OF AN EDUCATIONAL KIT WITH A DC-DC STEP-DOWN CONVERTER BUCK FOR TEACHING IN POWER ELECTRONICS AND DIGITAL CONTROL

AUTHOR: MILTON BRENNER MACHADO MATOSO

ADVISOR: RAFAEL CONCATTO BELTRAME

This assignment presents the development of an educational kit for Power Electronics and Digital Control formed by a DC-DC step-down converter (Buck), a control and instrumentation board and a gate-driver board to trigger the semiconductor switch (MOSFET) that controls the converter. This assignment has its focus the development of a educational kit that improves the quality of the future engineers by presenting the students with practical classes instead of the standard ones. This project starts studying the topology and operation of the Buck converter so the components can be calculated. Next step is the schematic of the project with the chosen components and the development of the layout for the printed circuit boards (PCB). After we test and make sure all the electronics of the project are working accordingly with the mentioned topology, we start the control design for the two control loops (current and voltage). This design starts with the mathematical model of the chosen control loop. The model has to be validated and it's used the MATLAB® and PSIM® software for that. After the validation, the control loop is implemented in the implemented PCB and the results are compared with simulated ones.

Key words: Digital Control, Power Electronics, Education, PCB, Engineering, Buck, Educational Kit.

LISTA DE FIGURAS

Figura 1 - Conversores Buck, Boost e Buck-Boost da Texas Instruments®	16
Figura 2 - Conversor <i>Buck</i> Scientech Technologies®	16
Figura 3 - Banco de ensaios Exsto Tecnologia.....	17
Figura 4 - Conversor <i>Buck</i>	18
Figura 5 - Medidas de tensão e corrente no conversor <i>Buck</i>	20
Figura 6 - Primeira etapa de operação do conversor <i>Buck</i>	21
Figura 7 - Segunda etapa de operação do conversor <i>Buck</i>	22
Figura 8 - Tensão e corrente no indutor nas duas etapas de operação.....	23
Figura 9 - Formas de onda de tensão e corrente no diodo e chave.....	25
Figura 10 - Formas de onda no capacitor	27
Figura 11 - Esquemático conversor <i>Buck</i> desenvolvido no <i>software Kicad</i>	33
Figura 12 - <i>Layout</i> da PCI do conversor no Kicad	36
Figura 13 - Visão 3D da da PCI do conversor no Kicad	36
Figura 14 - Adesivo silkscreen colado no conversor.....	37
Figura 15 - Conversor <i>Buck</i> implementado	38
Figura 16 - Esquemático do circuito de <i>gate-driver</i> desenvolvido no Kicad	40
Figura 17 - <i>Layout</i> do circuito de <i>gate-driver</i> no Kicad.....	42
Figura 18 - Visualização 3D do circuito de <i>gate-driver</i> no <i>Kicad</i>	43
Figura 19 – Circuito de <i>gate-driver</i> fabricado	43
Figura 20 – Circuito de instrumentação de corrente.....	45
Figura 21 – Circuito de instrumentação de tensão	45
Figura 22 - Esquemático da placa de instrumentação e controle	47
Figura 23 - <i>Layout</i> da PCI de instrumentação e controle	50
Figura 24 - Visualização 3D da PCI de instrumentação e controle.....	50
Figura 25 - PCI de instrumentação e controle implementada	51
Figura 26 - Malha de controle	52
Figura 27 - Circuito equivalente para análise da malha de tensão	57
Figura 28 – Modelo de corrente no PSIM®	59
Figura 29 - Validação modelo de corrente	59
Figura 30 - Modelo de tensão no PSIM®	60
Figura 31 - Validação modelo de tensão	60

Figura 32 - Diagrama de Bode da malha de corrente em malha aberta compensada	63
Figura 33 - Diagrama de Bode da malha de tensão em malha aberta compensada.....	63
Figura 34 - Implementação das malhas de controle no PSIM®	65
Figura 35 – Simulação do conversor operando em malha fechada no PSIM®	66
Figura 36 - Ações de controle e erros	66
Figura 37 - <i>Kit</i> didático completo montado.....	67
Figura 38 - Leituras da instrumentação de corrente	68
Figura 39 - Leituras da instrumentação de tensão	69
Figura 40 – Tensão de entrada e saída do conversor.....	70
Figura 41 – Corrente média no indutor	72
Figura 42 - <i>Ripple</i> indutor L_1	73
Figura 43 - <i>Ripple</i> indutor L_2	73
Figura 44 - <i>Ripple</i> indutor L_3	74
Figura 45 - <i>Ripple</i> de tensão C_1	75
Figura 46 - Ripple de tensão C_2	76
Figura 47 - <i>Ripple</i> de tensão C_3	76
Figura 48 - Formas de onda de tensão e corrente da chave	78
Figura 49 – Formas de onda de tensão e corrente do diodo	79
Figura 50 - Simulação de um degrau na razão-cíclica.....	80
Figura 51 - Degrau na razão-cíclica no conversor.....	81
Figura 52 - Simulação degrau na referência de corrente	82
Figura 53 - Degrau na referência do controlador de corrente no conversor	82
Figura 54 - Simulação de um degrau na referência de tensão	83
Figura 55 - Degrau na referência de tensão no conversor	84
Figura 56 - Simulação de um degrau na carga	85
Figura 57 - Degrau de carga no conversor	85
Figura 58 – Esquemático da fonte de alimentação auxiliar	90
Figura 59 - <i>Layout</i> da fonte de alimentação auxiliar	92
Figura 60 - Visualização 3D da fonte auxiliar.....	93
Figura 61 - Fonte de alimentação auxiliar fabricada	94
Figura 62 - Foto da fonte de alimentação aberta	94

LISTA DE TABELAS

Tabela 1 - Especificações do conversor <i>Buck</i>	30
Tabela 2 - Parâmetros do conversor <i>Buck</i>	30
Tabela 3 - Componentes selecionados.....	31
Tabela 4 - Lista de componentes conversor <i>Buck</i>	33
Tabela 5 - Lista de componentes do circuito de <i>gate-driver</i>	41
Tabela 6 - Componentes da PCI de instrumentação e controle	48
Tabela 7 - Valores das constantes da equação de diferenças discretas	64
Tabela 8 – <i>Ripple</i> dos indutores do conversor.....	74
Tabela 9 - <i>Ripple</i> dos capacitores do conversor	77
Tabela 10 - Componentes fonte de alimentação auxiliar	91

SUMÁRIO

CAPÍTULO 1 INTRODUÇÃO	12
1.1 Objetivos.....	13
1.1.1 Objetivos gerais	13
1.1.2 Objetivos específicos	13
CAPÍTULO 2 REVISÃO DA LITERATURA	14
2.1 Propostas de metodologia de ensino e aprendizagem.....	14
2.2 Soluções comerciais.....	15
2.2.1 <i>Texas Instruments®</i>	15
2.2.2 Scientech Technologies®	16
2.2.3 Exsto Tecnologia	17
2.3 Limitações das soluções comerciais	17
2.4 Conversor Buck	18
CAPÍTULO 3 ANÁLISE DE FUNCIONAMENTO DO CONVERSOR BUCK.....	20
3.1 Considerações iniciais.....	20
3.2 Primeira etapa	21
3.3 Segunda etapa	22
3.4 Formas de onda.....	23
CAPÍTULO 4 DIMENSIONAMENTO DOS COMPONENTES	26
4.1 Projeto do capacitor de filtro	26
4.2 Projeto do indutor de filtro.....	27
4.3 Dimensionamento da chave (MOSFET) e do diodo.....	28
4.4 Especificações do conversor <i>Buck</i>	29
CAPÍTULO 5 IMPLEMENTAÇÃO DO CONVERSOR.....	32
5.1 Esquemático do conversor <i>Buck</i>	32
5.2 Lista de componentes.....	33
5.3 Layout desenvolvido no <i>software Kicad</i>	35
5.4 Conversor <i>Buck</i> implementado	37
CAPÍTULO 6 PROJETO DO CIRCUITO DE GATE-DRIVER	39
6.1 Esquemático do circuito de <i>gate-driver</i>	39
6.2 Lista de componentes do circuito de <i>gate-driver</i>	40
6.3 Layout do circuito de <i>gate-driver</i> no <i>software Kicad</i>	42
6.4 Círculo de <i>gate-driver</i> implementado.....	43
CAPÍTULO 7 PROJETO DO CIRCUITO DE INSTRUMENTAÇÃO.....	44
7.1 Dimensionamento da instrumentação de corrente e tensão	44

7.2	Esquemático da placa de instrumentação e controle.....	45
7.3	Lista de componentes da PCI de instrumentação e controle.....	48
7.4	<i>Layout</i> da PCI de instrumentação e controle no <i>software</i> Kicad	49
7.5	PCI de instrumentação e controle implementada.....	51
CAPÍTULO 8 MODELAGEM DINÂMICA DO CONVERSOR.....		52
8.1	Modelagem da planta para controle da corrente no indutor de filtro.....	53
8.1.1	Modelo médio em espaço de estados.....	53
8.1.2	Modelo CA de pequenos sinais	55
8.2	Modelagem da planta para controle da tensão no capacitor de filtro.....	56
CAPÍTULO 9 PROJETO DO CONTROLE EM MALHA FECHADA		58
9.1	Validação dos modelos	58
9.2	Projeto do sistema de controle em malha fechada	61
9.3	Implementação das malhas de controle	65
9.3.1	Implementação no PSIM	65
9.3.2	Implementação no microcontrolador	66
CAPÍTULO 10 RESULTADOS EXPERIMENTAIS		67
10.1	Resultados das instrumentações de corrente e tensão	68
10.2	Resultados em malha aberta	69
10.2.1	Tensão de entrada e saída	70
10.2.2	Corrente no indutor.....	71
10.2.3	<i>Ripples</i> da tensão de saída	75
10.2.4	Tensão e corrente sobre a chave	77
10.2.5	Tensão e corrente sobre o diodo	78
10.3	Resultados em malha fechada.....	79
10.3.1	Degrau na razão-cíclica	80
10.3.2	Degrau na referência de corrente	81
10.3.3	Degrau na referência de tensão	83
10.3.4	Degrau de carga	84
CAPÍTULO 11 CONCLUSÃO.....		86
CAPÍTULO 12 REFERÊNCIAS BIBLIOGRÁFICAS		87
APÊNDICE A – PROJETO DA FONTE AUXILIAR.....		90
12.1	Esquemático da fonte auxiliar.....	90
12.2	Lista de componentes da fonte auxiliar	91
12.3	<i>Layout</i> da fonte auxiliar	92
12.4	Fonte de alimentação auxiliar fabricada	93
APÊNDICE B – SCRIPT EM MATLAB® PARA O PROJETO DOS CONTROLADORES.....		95
APÊNDICE C - SCRIPT EM C PARA A SIMULAÇÃO NO PSIM®		98

APÊNDICE D – *SCRIPT EM C IMPLEMENTADO NO MICROCONTROLADOR.....* 100

CAPÍTULO 1 INTRODUÇÃO

O sistema de ensino atual utiliza, em sua maioria, uma forma expositiva para a apresentação de conteúdos, porém, cada vez, mais há contestação deste método. Devido às diversas maneiras que um ser humano se sente confortável para aprender, a variabilidade dos meios utilizados para ensino deve ser considerada, podendo-se utilizar uma alternância de metodologias para que o aluno atinja maior aprendizado.

A melhora nos métodos de ensino da engenharia permite uma melhora na qualificação dos profissionais formados. Atualmente, além do método expositivo, há aulas experimentais em laboratório. Normalmente, após as aulas expositivas explicando os experimentos a serem realizados, os alunos são levados a um laboratório que possua a capacidade de realização do experimento em questão. No caso da Engenharia Elétrica e áreas correlatas, circuitos elétricos são um foco muito importante no aprendizado dos estudantes, onde a explicação do funcionamento do circuito precede sua simulação que, por sua vez precede a montagem em laboratório. ANG (1996) atesta que este método auxilia a solidificação do conteúdo pelos alunos, ao menos no tópico explorado por este autor (conversores de potência em regime transitório e permanente).

Para facilitar o aprendizado dos estudantes, especificamente na disciplina de Eletrônica de Potência, há diversos softwares utilizados para a simulação dos circuitos e também existem *kits* didáticos comerciais para auxílio nas aulas experimentais. A integração desse *hardware* e *software* tem efeito positivo na retenção de conhecimento pelos alunos.

Por essas razões, propõe-se neste trabalho o desenvolvimento de um *kit* didático para contribuir com o ensino em Eletrônica de Potência e Controle Digital de Sistemas. No caso deste trabalho foi escolhido o conversor CC-CC rebaixador não isolado (*Buck*) para ser desenvolvido.

1.1 Objetivos

1.1.1 Objetivos gerais

Este trabalho tem como objetivo o projeto e a implementação de um módulo didático para ensino de Eletrônica de Potência e Controle Digital de Sistemas, composto por uma placa de circuito impresso do conversor CC-CC rebaixador (*Buck*) e circuitos auxiliares (*hardware*), bem como um código básico implementado em microcontrolador (*software*). Além disso, serão propostos experimentos para emprego em aulas práticas.

1.1.2 Objetivos específicos

O trabalho possui os seguintes objetivos específicos:

1. Desenvolvimento do projeto e execução das seguintes placas de circuito impresso necessárias ao módulo:
 - (a) Conversor CC-CC *Buck*;
 - (b) *Gate-driver*;
 - (c) Instrumentação e controle;
 - (d) Fonte de alimentação auxiliar.
2. Proposição de experimentos para o módulo que facilitem o aprendizado dos alunos em Eletrônica de Potência e Controle de Sistemas.

CAPÍTULO 2 REVISÃO DA LITERATURA

Neste capítulo será feita a revisão bibliográfica. Estarão presentes diferentes soluções para o ensino de Eletrônica de Potência, seja metodologia de ensino ou soluções comerciais. Por fim, será abordado o funcionamento de um conversor CC-CC rebaixador (*Buck*).

2.1 Propostas de metodologia de ensino e aprendizagem

Nesta seção serão abordadas diferentes metodologias que autores utilizam para a melhoria do ensino em Eletrônica de Potência, assim chegando no objetivo de melhor formação dos profissionais da área.

MAX et al. (2009) propõem o método CDIO (do inglês *Conceive, Design, Implement and Operate*). Este método começa com a apresentação do problema e objetivos a serem atingidos, após isso é realizado o desenvolvimento do projeto, bem como sua implementação e operação. O aluno, assim, experiencia a resolução de um problema de maneira mais próxima à realizada no setor produtivo, pois, no caso da Engenharia, a implementação dos projetos envolve o emprego de componentes comerciais, os quais dificilmente são iguais aos simulados ou teóricos para aquele projeto.

Para facilitar o aprendizado dos estudantes em Eletrônica de Potência, há diversos *softwares* utilizados para a simulação dos circuitos e também existem *kits* didáticos comerciais para auxílio nas aulas experimentais. Alguns trabalhos desenvolvidos na área são de JIMENEZ-MARTINEZ et al. (2005), que implementam uma plataforma configurável integrando *hardware* e *software*, a qual é capaz de, alterando parâmetros de conversores configurados nela, analisar seu funcionamento.

SUSIN (2000) e FERREIRA MACHADO; ABRITTA COSTA; DO CARMO (2012) propõem laboratórios modulares, permitindo a prática de técnicas de identificação e controle em Eletrônica de Potência em diversas topologias de conversores. PEREZ et al. (2008) desenvolvem *kits* didáticos em Eletrônica de Potência com baixa tensão de saída, mas com alta corrente, permitindo o acionamento de motores de indução trifásicos. Por fim, AL BASTAMI; AVESTRUZ; LEEB (2014) propõem o desenvolvimento de um *kit* para ensino de modelagem e controle em Eletrônica de Potência.

POMILIO (2020) traz a abordagem da Unicamp (Universidade Estadual de Campinas) para o ensino da Eletrônica de Potência. Este autor aborda a análise de circuitos, seu projeto e simulações (para saber os resultados esperados deles), e, por fim, os experimentos realizados nas aulas para avaliá-los. Esta metodologia vem sendo empregada por melhorar a retenção do conhecimento pelos alunos.

2.2 Soluções comerciais

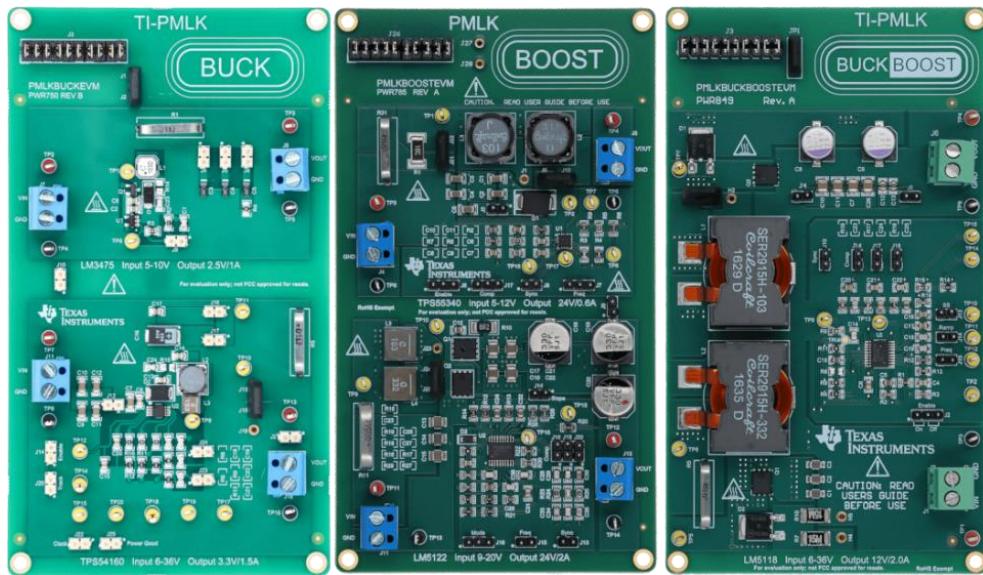
Há diversas soluções comerciais em *hardware* para auxiliar no ensino de Eletrônica de Potência. Têm-se exemplos de fabricantes que disponibilizam *kits* e bancadas didáticas de laboratório nesta área.

2.2.1 Texas Instruments®

A *Texas Instruments*® é uma gigante mundial da eletrônica, tem em sua carta de produtos dispositivos semicondutores, microcontroladores e uma extensa variedade de produtos voltados à eletrônica, com mais de 40 mil patentes registradas.

Esta empresa possui *kits* para Eletrônica de Potência de diversos conversores. Tem ainda o programa conhecido como *Power Management Lab Kit* (PMLK), que possui em sua carta diversas topologias de conversores CC-CC como, por exemplo, o *Buck*, o *Boost* e o *Buck-Boost*. Esses *kits* possuem um custo de U\$ 49,00, U\$ 89,00 e U\$ 89,00, respectivamente (TEXAS INSTRUMENTS®, 2023). A Figura 1 mostra estes conversores.

Figura 1 - Conversores Buck, Boost e Buck-Boost da Texas Instruments®



Fonte: (TEXAS INSTRUMENTS®, 2023)

2.2.2 Scientech Technologies®

A Scientech Technologies® é uma empresa de origem Indiana com mais de 35 anos no mercado fornecendo soluções no setor de eletrônica. Oferece mais de 550 produtos em 76 países do mundo. Esta empresa dispõe de diversos produtos para a aprendizagem em Eletrônica de Potência, possuindo também os módulos das topologias de conversores mencionadas anteriormente (*Buck, Boost e Buck-Boost*).

O custo do conversor *Buck* desta empresa é em torno de U\$ 200,00. Entretanto, esse produto tem como ponto negativo ser uma “caixa fechada”, onde os alunos não tem acesso aos componentes do conversor, a Figura 2 mostra detalhes do produto.

Figura 2 - Conversor *Buck* Scientech Technologies®



Fonte: (SCIENTECH TECHNOLOGIES).

2.2.3 Exsto Tecnologia

A Exsto Tecnologia é uma empresa brasileira com base em Minas Gerais. Seu banco de ensaios é utilizado para montagem de retificadores bifásicos e trifásicos, controlados ou não, circuitos de controle de ângulo de disparo de tiristores (SCR – *Silicon Controled Rectifier*) e um inversor trifásico didático. Esse banco de ensaios é composto por diversos módulos montados em um *rack* de dimensões 575 mm x 426 mm x 400 mm. O valor unitário deste produto é de R\$19.183,73. O *rack* pode ser visualizado na Figura 3.

Figura 3 - Banco de ensaios Exsto Tecnologia



Fonte: (EXSTO TECNOLOGIA, 2016).

2.3 Limitações das soluções comerciais

As soluções comerciais apresentadas apresentam alguns pontos negativos, como sua alta complexidade ou abstração dificultando o aprendizado do aluno, nas soluções de maior complexidade se torna mais complicado o aprendizado do aluno dos conceitos básicos ensinados em sala de aula, as de alta abstração aumentam o distanciamento entre o material de ensino e a maneira que projetos dessa área são desenvolvidos na prática.

Outro problema é o alto custo desses materiais didáticos, ou baixa presença no mercado nacional, assim tornando sua obtenção por unidades de ensino públicas muito mais complexa.

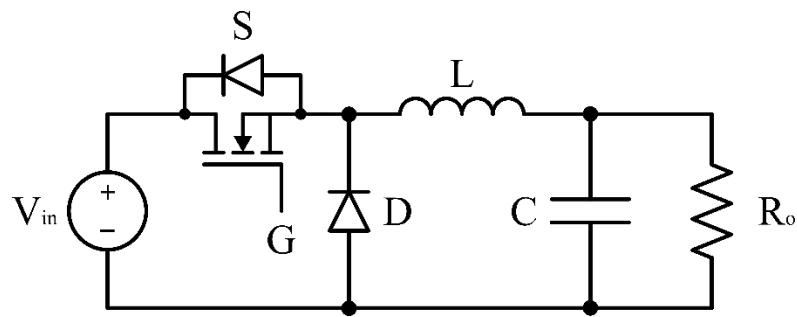
O trabalho desenvolvido tenta realizar o projeto de material didático para o ensino de Eletrônica de Potência e Controle de Sistema de forma a suprir algumas das carências dos materiais didáticos mencionados neste capítulo.

2.4 Conversor Buck

O módulo de potência deste trabalho será um conversor CC-CC rebaixador não isolado (*Buck*). Assim, é necessário revisitar sua topologia e funcionamento, facilitando a compreensão da etapa de projeto do conversor no CAPÍTULO 4. Este conversor tem como característica converter um nível de tensão CC de entrada (V_{in}) em um nível de tensão CC de saída (V_o), sendo o valor de saída inferior ao de entrada, $V_{in} > V_o$.

O conversor tem como elementos básicos uma chave semicondutora, um diodo e um filtro LC, sendo alimentado por uma fonte CC. Esses elementos são dispostos como apresentado na Figura 4. As chaves semicondutoras mais utilizadas são MOSFETs (*Metal Oxide...*) e IGBTs (*Insulated Gate...*), menos comumente utilizadas temos os TBJs (Transistor Bipolar de Junção).

Figura 4 - Conversor *Buck*



Fonte: Próprio autor.

O conversor *Buck* possui as mais diversas aplicações, dentre elas estão sua utilização em carregadores de bateria para celulares, *notebooks* e *power banks*, é utilizado na alimentação de computadores e placas-mães e também em carregadores solares. Sua versatilidade o coloca em diversas áreas como *Systems on a Chip* (SoC) (LI et al., 2022), *Internet of Things* (IoT) (GAO et al., 2022) e aplicações automotivas (BIZZARRI et al., 2021; SHRUD et al., 2009).

Este conversor possui três modos de operação: modo de condução contínua (do inglês *Continuous Conduction Mode*, CCM), modo de condução crítica (do inglês *Boundary Conduction Mode*, BCM) e modo de condução descontínua (do inglês *Discontinuous Conduction Mode*, DCM). Os modos se referem aos diferentes níveis de corrente no indutor ao longo do tempo. No modo CCM a corrente nunca chega a zero, no modo BCM a corrente no indutor é zero em apenas um instante de tempo e no modo DCM ela se mantém em zero por parte do período de chaveamento. O modo descontínuo possui algumas aplicações, entretanto não é interessante seu uso em potências elevadas. Este trabalho considerará apenas o modo de condução contínua, CCM (ERICKSON; MAKSIMOVIĆ, 2001).

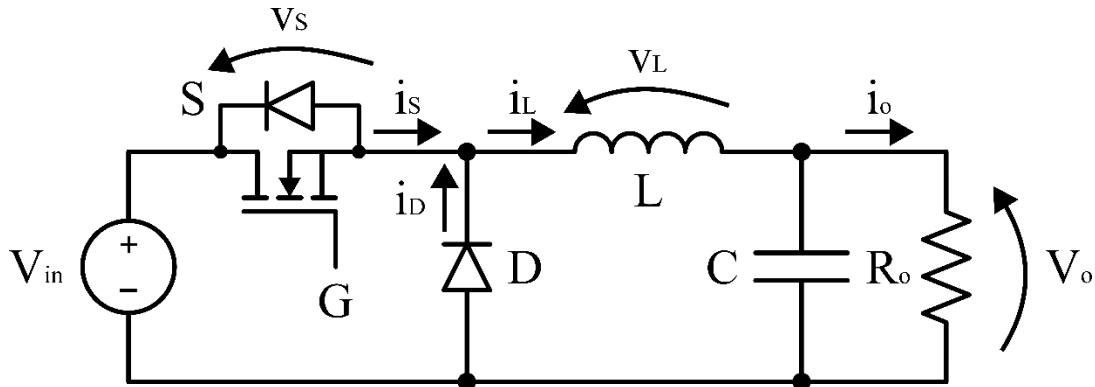
Esta topologia possui apenas um elemento a ser controlado, a chave. Acionada pelo sinal de *gate* ela poderá estar ligada (ON) ou desligada (OFF) e isso gera duas etapas distintas de funcionamento do *Buck*. Na primeira etapa, com a chave ON, ocorrerá o armazenamento de energia no sistema através de seus elementos reativos, enquanto na segunda etapa, com a chave OFF, esses elementos reativos devolverão energia ao sistema. A razão entre o tempo que a chave fica ON e o tempo que ela fica OFF é chamada de razão-cíclica D (do inglês, *duty-cycle*). Nesta topologia a razão-cíclica é dada pelo quociente entre a tensão média de saída e a tensão

$$\text{média de entrada do sistema: } D = \frac{V_o}{V_{in}}.$$

CAPÍTULO 3 ANÁLISE DE FUNCIONAMENTO DO CONVERSOR *BUCK*

Neste capítulo realizar-se-á a análise do conversor *Buck*. Algumas assunções iniciais são necessárias para facilitar esse processo. Serão observadas as duas etapas de operação do conversor em modo CCM e no final serão mostrados os sinais de tensão e corrente nos principais componentes. Todas as medições do conversor são tomadas conforme as orientações dadas na Figura 5.

Figura 5 - Medidas de tensão e corrente no conversor *Buck*



Fonte: Próprio autor.

3.1 Considerações iniciais

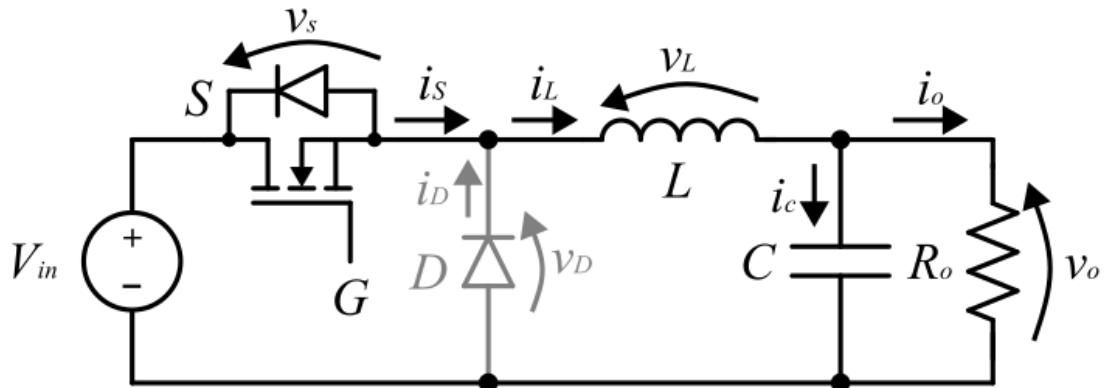
Para a análise das etapas de funcionamento deste conversor, algumas assunções precisam ser feitas, são elas:

1. Operação com frequência de chaveamento constante $f_s = \frac{1}{T_s}$ e modulação de sinal de *gate* por largura de pulso (PWM – *Pulse-Width Modulation*);
2. Operação em modo CCM, com $i_L(t) > 0$ para todo o tempo;
3. Todos os componentes do sistema são ideais, ou seja, não possuem perdas;
4. Conversor rebaixador de tensão, ou seja, $v_o(t) < V_{in}$;
5. Tensão de saída possui pequeno *ripple* (variação), podendo ser aproximada pelo seu valor médio, assim $v_o \equiv V_o$.

3.2 Primeira etapa

A primeira etapa de funcionamento do *Buck* ocorre quando a chave está em condução, ON, nesse momento o diodo está bloqueado, pois se encontra reversamente polarizado, conforme a Figura 6. No modo CCM o diodo opera sempre de maneira complementar a chave: quando ela está conduzindo o diodo está bloqueado, e vice versa. Nesta primeira etapa a chave está em condução, sendo a sua corrente i_s igual à corrente que passa pelo indutor, enquanto isso o diodo está bloqueado, não permitindo a passagem de corrente. Esta etapa ocorre do tempo $t = 0$, até o tempo $t = DT_s$ e sua duração é definida como T_{on} . O sistema é alimentado conforme a Figura 6.

Figura 6 - Primeira etapa de operação do conversor *Buck*



Fonte: Próprio autor.

Aplicando a Lei das Tensões de Kirchoff (LTK) na malha externa desta etapa, tem-se que:

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

Através da equação de braço do indutor pode-se perceber que quando há uma tensão aplicada nele há uma variação na sua corrente:

$$v_L = L \frac{di_L}{dt} \quad (2)$$

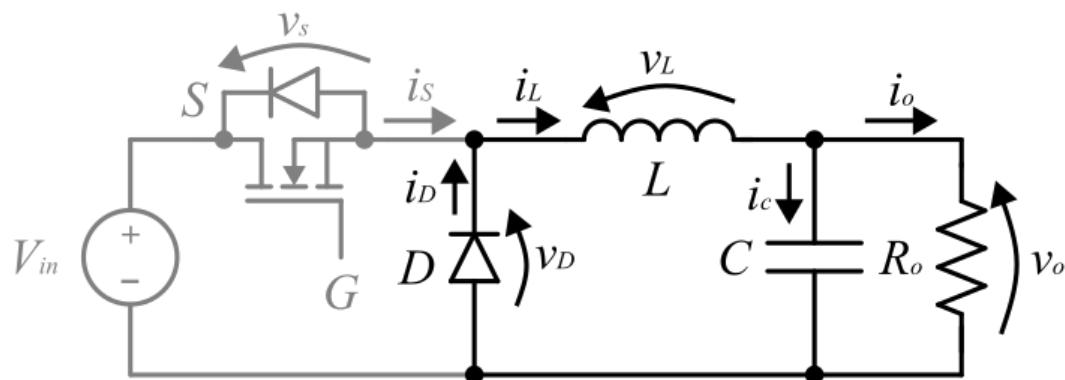
Ao substituir o valor de v_L em (1), isolando $\frac{di_L}{dt}$ e tendo em mente a assunção 4 da seção 3.1, pode-se então inferir que, nesta etapa, a corrente no indutor está crescendo linearmente., conforme a equação (3).

$$\frac{di_L}{dt} = \frac{V_{in} - V_o}{L} > 0 \quad (3)$$

3.3 Segunda etapa

Esta segunda etapa de funcionamento do conversor ocorre quando a chave não está mais em condução, OFF, nesse momento o diodo está em condução. Com o diodo está em condução, a sua corrente i_D é igual a corrente que passa pelo indutor, enquanto isso a chave está bloqueada, não havendo passagem de corrente nela, conforme a Figura 7. Esta etapa ocorre a partir do tempo $t = DT_s$ até $t = T_s$, cuja duração é definida como T_{off} .

Figura 7 - Segunda etapa de operação do conversor Buck



Fonte: Próprio autor.

Aplicando a LTK na malha indicada na Figura 7, encontra-se que nesta etapa $v_L = -V_o$. Aplicando novamente a equação de braço do indutor (4) chega-se na variação de corrente no indutor:

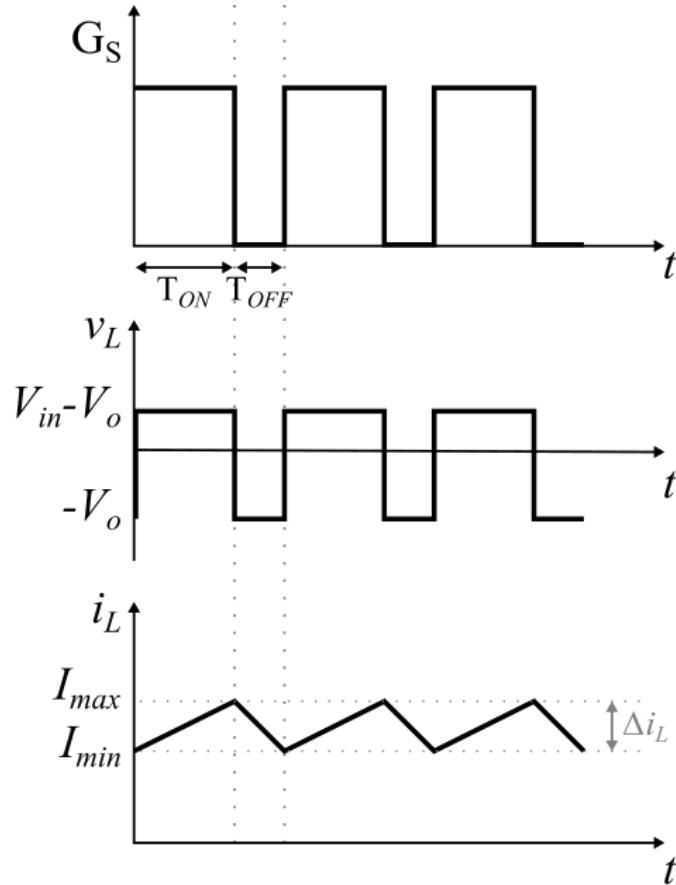
$$\frac{di_L}{dt} = -\frac{V_o}{L} \quad (4)$$

A equação permite inferir que nesta etapa a corrente do indutor decresce linearmente. Ou seja, no modo CCM, a energia que é armazenada no indutor na primeira etapa é devolvida ao sistema na segunda etapa.

3.4 Formas de onda

Para um melhor entendimento das duas etapas de operação de maneira conjunta, se faz necessária a análise das formas de onda do conversor. A Figura 8 apresenta o comportamento tensão v_L e corrente i_L no indutor em relação ao sinal de *gate* da chave (G_s).

Figura 8 - Tensão e corrente no indutor nas duas etapas de operação



Fonte: Próprio autor.

O sinal G_s determina as duas etapas de operação do conversor. Quando ele está alto, durante o período T_{on} , tem-se a primeira etapa, e quando ele está baixo, durante o período T_{off} , tem-se a segunda etapa. Como assume-se que este conversor está operando no modo CCM, a

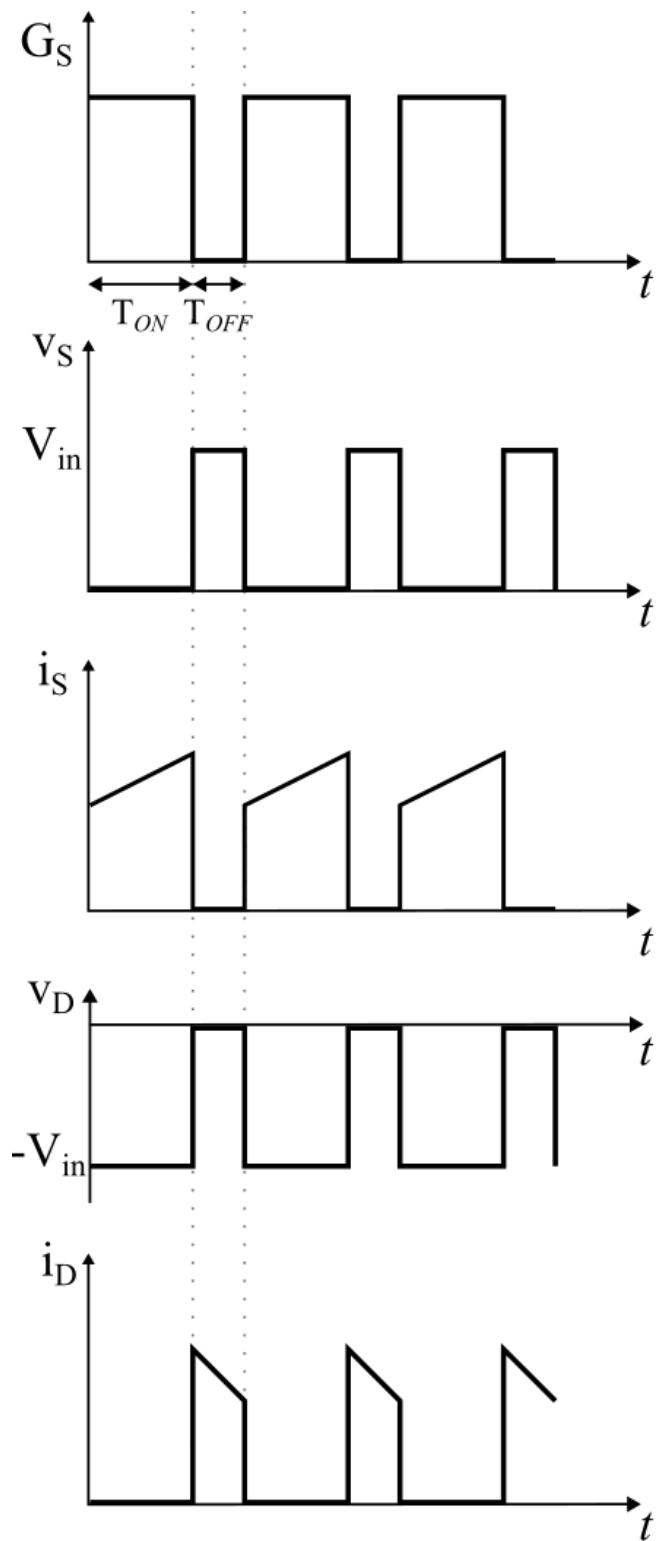
corrente do indutor não chega a 0, então no início do período T_{on} tem-se o valor mínimo de corrente no indutor desta etapa, que será chamado de $I_{\min 1}$. Enquanto o conversor está na primeira etapa o indutor vai ser magnetizado, sua corrente aumentará até o final desta etapa, chegando no valor I_{\max} . Na segunda etapa, o indutor é desmagnetizado pelo período T_{off} e sua corrente varia de I_{\max} ao valor mínimo $I_{\min 2}$, que ocorre no instante T_s .

Em regime permanente, toda a energia que ele armazena na primeira etapa é devolvida ao circuito na segunda etapa, nesse caso temos o balanço de energia no indutor (ERICKSON; MAKSIMOVIĆ, 2001). Isso resulta nos valores mínimos de corrente das duas etapas serem iguais, ou seja, $I_{\min 1} = I_{\min 2}$. Permite assim o cálculo do *ripple* de corrente do indutor como mostra a equação (5):

$$\Delta i_L = I_{\max} - I_{\min} \quad (5)$$

Após esse entendimento das duas etapas de operação e do comportamento da corrente no indutor, deve-se analisar também as formas de onda de tensão e corrente na chave e no diodo. A Figura 9 mostra como estes dois elementos se comportam de maneira complementar.

Figura 9 - Formas de onda de tensão e corrente no diodo e chave



Fonte: Próprio autor.

CAPÍTULO 4 DIMENSIONAMENTO DOS COMPONENTES

Entender o funcionamento do conversor *Buck* permite o dimensionamento dos seus componentes. Os componentes principais a serem dimensionados são: capacitor e indutor de filtro, chave (MOSFET) e diodo. Outras características do conversor projetado dependem de suas especificações, que serão dadas pelo cliente ou engenheiro responsável pelo projeto.

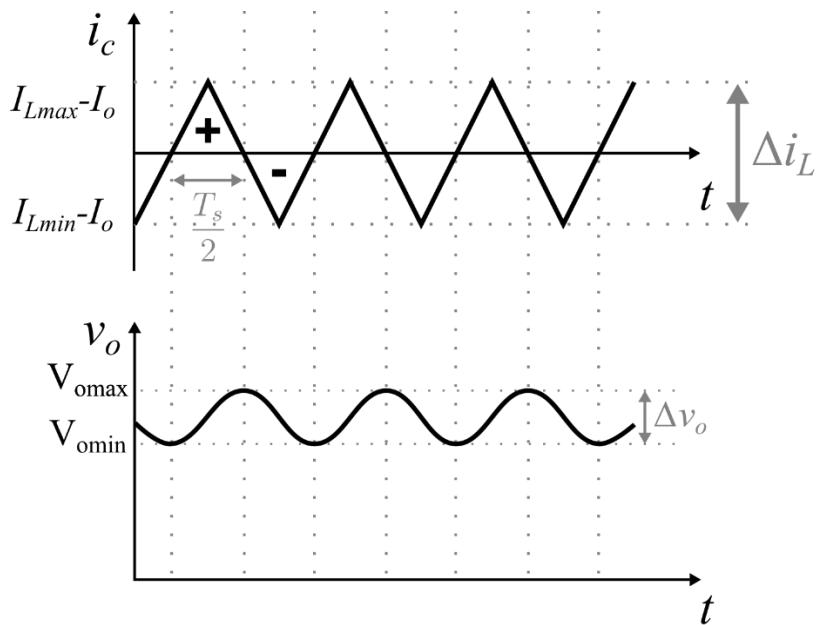
Primeiramente, serão descritas as equações para dimensionar os componentes citados. Após isso, com os valores das especificações definidas pelo autor do projeto, serão calculados esses componentes.

4.1 Projeto do capacitor de filtro

O capacitor de filtro do conversor *Buck* é disposto em paralelo com a resistência de carga, como pode ser visto na Figura 4. Desta forma, ele absorve a parcela em corrente alternada da corrente vinda do indutor, garantindo que apenas a parcela em corrente contínua chegue na carga. A corrente no capacitor segue a equação (6), as formas de onda neste componente são apresentadas na Figura 10. No início deste trabalho a assunção que a tensão de saída V_o possui um *ripple* tão pequeno que pode ser considerada constante, para o cálculo do capacitor de filtro é necessário o *ripple* dessa tensão, portanto neste item do trabalho a tensão de saída não será considerada constante.

$$i_C = C \frac{dv_o}{dt} \quad (6)$$

Figura 10 - Formas de onda no capacitor



Fonte: Próprio autor.

Isolando a tensão no capacitor e integrando em ambos os lados no período de 0 à $T_s/2$, chega-se à seguinte relação:

$$\int_{V_o \text{ min}}^{V_o \text{ max}} dv_o = \frac{1}{C} \int_0^{T_s/2} i_c dt \quad (7)$$

$$\Delta v_o = \frac{1}{C} \left(\frac{T_s}{2} \frac{\Delta i_L}{2} \frac{1}{2} \right) \quad (8)$$

Onde $\Delta v_o = V_{o\text{max}} - V_{o\text{min}}$

Isolando a capacidade e colocando a equação em função da frequência de chaveamento, $f_s = 1/T_s$, a equação se torna:

$$C = \frac{\Delta i_L}{8\Delta v_C f_s} \text{ [F]} \quad (9)$$

4.2 Projeto do indutor de filtro

Para o projeto do indutor de filtro são necessárias as informações do *ripple* admitido de corrente, da razão-cíclica que o conversor vai operar, sua frequência de chaveamento e a tensão

de saída do conversor. A equação de projeto tem por base a segunda etapa de operação do conversor *Buck* em CCM.

Na segunda etapa de operação, como pode ser visto na Figura 8, a corrente começa com o valor de I_{\max} e decresce até o valor de I_{\min} , o período de tempo no qual isso acontece é T_{off} , portanto, parte-se da equação de braço do indutor.

$$v_L = L \frac{di_L}{dt} \quad (10)$$

Isolando di_L e integrando os dois lados da equação (10)

$$\int_{I_{L\max}}^{I_{L\min}} di_L = \frac{1}{L} \int_{T_{on}}^{T_s} v_L dt \quad (11)$$

Resolvendo as integrais e lembrando que na segunda etapa de operação $v_L = -V_o$

$$i_L|_{I_{L\max}}^{I_{L\min}} = \frac{1}{L} (-V_o) t \Big|_{T_{on}}^{T_s} \quad (12)$$

$$I_{L\min} - I_{L\max} = -\frac{V_o}{L} (T_s - T_{on}) \quad (13)$$

Substituindo $I_{L\min} - I_{L\max} = -\Delta i_L$ e $T_s - T_{on} = (1-D)T_s$ na equação (13)

$$\Delta i_L = \frac{V_o}{L} (1-D) T_s \quad (14)$$

Como está sendo calculado o valor do indutor de filtro do conversor, deve-se isolar a variável L , lembrando também que o período de chaveamento é o inverso da frequência,

$T_s = \frac{1}{f_s}$, para relacionar a equação à frequência de chaveamento. Chega-se em:

$$L = \frac{V_o (1-D)}{\Delta i_L f_s} [\text{H}] \quad (15)$$

4.3 Dimensionamento da chave (MOSFET) e do diodo

O dimensionamento do diodo se dá pelo cálculo da corrente média que passa por ele, enquanto o dimensionamento da chave (especificamente, um MOSFET) é através do cálculo da corrente eficaz. O gráfico das correntes através de cada elemento pode ser visto na Figura 9.

. Utilizando o gráfico e as definições de corrente média e eficaz, calcula-se os valores de interesse. Para a corrente média no diodo, tem-se:

$$i_{D_{med}} = \frac{1}{T_s} \int_0^{T_s} i_d(t) dt \quad (16)$$

onde

$$i_{D_{med}} = \frac{1}{T_s} \left(0 \times DT_s + \frac{(I_{L\max} + I_{L\min})(1 - D)T_s}{2} \right) \quad (17)$$

$$I_L = \frac{(I_{L\max} + I_{L\min})}{2} \quad (18)$$

Substituindo (18) em (17) e simplificando:

$$i_{D_{med}} = I_L(1 - D) \quad (19)$$

Da mesma forma, para a chave, tem-se

$$i_{S_{rms}} = \left\{ \frac{1}{T_s} \int_0^{T_s} i_S^2(t) dt \right\}^{1/2} \quad (20)$$

$$i_{S_{rms}} = \left\{ \frac{1}{T_s} \left[\int_0^{DT_s} i_{S_1}^2(t) dt + \int_{DT_s}^{T_s} 0^2(t) dt \right] \right\}^{1/2} \quad (21)$$

onde:

$$i_{S_1} = I_{L\min} + \frac{I_{L\max} - I_{L\min}}{DT_s} t \quad (22)$$

Substituindo (22) em (21) e simplificando, resulta:

$$i_{S_{rms}} = \sqrt{DI_L} \quad (23)$$

4.4 Especificações do conversor *Buck*

A Tabela 1 contém as especificações do conversor a ser projetado. As especificações relacionadas às tensões e potência foram escolhidas para serem compatíveis com os equipamentos disponíveis no NUPEDEE (fontes de alimentação e osciloscópios). Como este projeto é para o uso didático, foram definidos três diferentes valores de *ripple* para corrente e para tensão, aumentando a variedade dos experimentos com o *kit*. Também foram previstos dois valores distintos de carga.

Tabela 1 - Especificações do conversor Buck

Descrição	Especificação
Potência de saída nominal	$P_o = 20 \text{ W}$
Rendimento esperado	$\eta = 0,9$
Tensão média de entrada	$V_{in} = 30 \text{ V}$
Tensão média de saída	$V_o = 15 \text{ V}$
Frequência de chaveamento	$f_s = 10 \text{ kHz}$
<i>Ripple</i> de corrente no indutor	$\Delta I_{L\%} = [20\%; 15\%; 10\%]$
<i>Ripple</i> de tensão no capacitor	$\Delta V_{C\%} = [4,75\%; 1,70\%; 0,50\%]$

Fonte: Próprio autor.

Essas especificações permitem o cálculo dos parâmetros que serão utilizados para o dimensionamento dos componentes do conversor, são eles: potência de saída, corrente média de carga, *ripple* de corrente através do indutor de filtro, *ripple* de tensão através do capacitor de filtro e resistência de carga. A Tabela 2 mostra os parâmetros calculados.

Tabela 2 - Parâmetros do conversor Buck

Descrição	Especificação
Potência de entrada nominal	$P_{in} = 22,22 \text{ W}$
Corrente média de carga	$I_o = 1,36 \text{ A}$
<i>Ripple</i> de corrente no indutor	$\Delta I_L = [0,266; 0,200; 0,133] \text{ A}$
<i>Ripple</i> de tensão no capacitor	$\Delta V_C = [0,710; 0,250; 0,075] \text{ V}$
Resistência de carga	$R_o = 11 \Omega$

Fonte: Próprio autor.

Substituindo essas informações nas equações apresentadas neste capítulo para o dimensionamento dos componentes, chega-se nos valores comerciais dos componentes informados na Tabela 3.

Tabela 3 - Componentes selecionados

Descrição	Especificação
	$L_1 = 2,8 \text{ mH}$
Indutâncias	$L_2 = 3,7 \text{ mH}$
	$L_3 = 5,6 \text{ mH}$
	$C_1 = 4,7 \mu\text{F}$
Capacitâncias	$C_2 = 10 \mu\text{F}$
	$C_3 = 22 \mu\text{F}$
Chave semicondutora	IRF 530 (FREESCALE) ($V_{DS,max} = 100 \text{ V}$, $I_{D,max} = 9 \text{ A}$)
Diodo semicondutor	MUR 820 (FREESCALE) ($V_{AK,max} = 200 \text{ V}$, $I_{F,max} = 8 \text{ A}$)

Fonte: Próprio autor.

Salienta-se que a chave e diodo foram propositalmente superdimensionados, diminuindo o risco de queima por utilização inadequada por parte do usuário.

CAPÍTULO 5 IMPLEMENTAÇÃO DO CONVERSOR

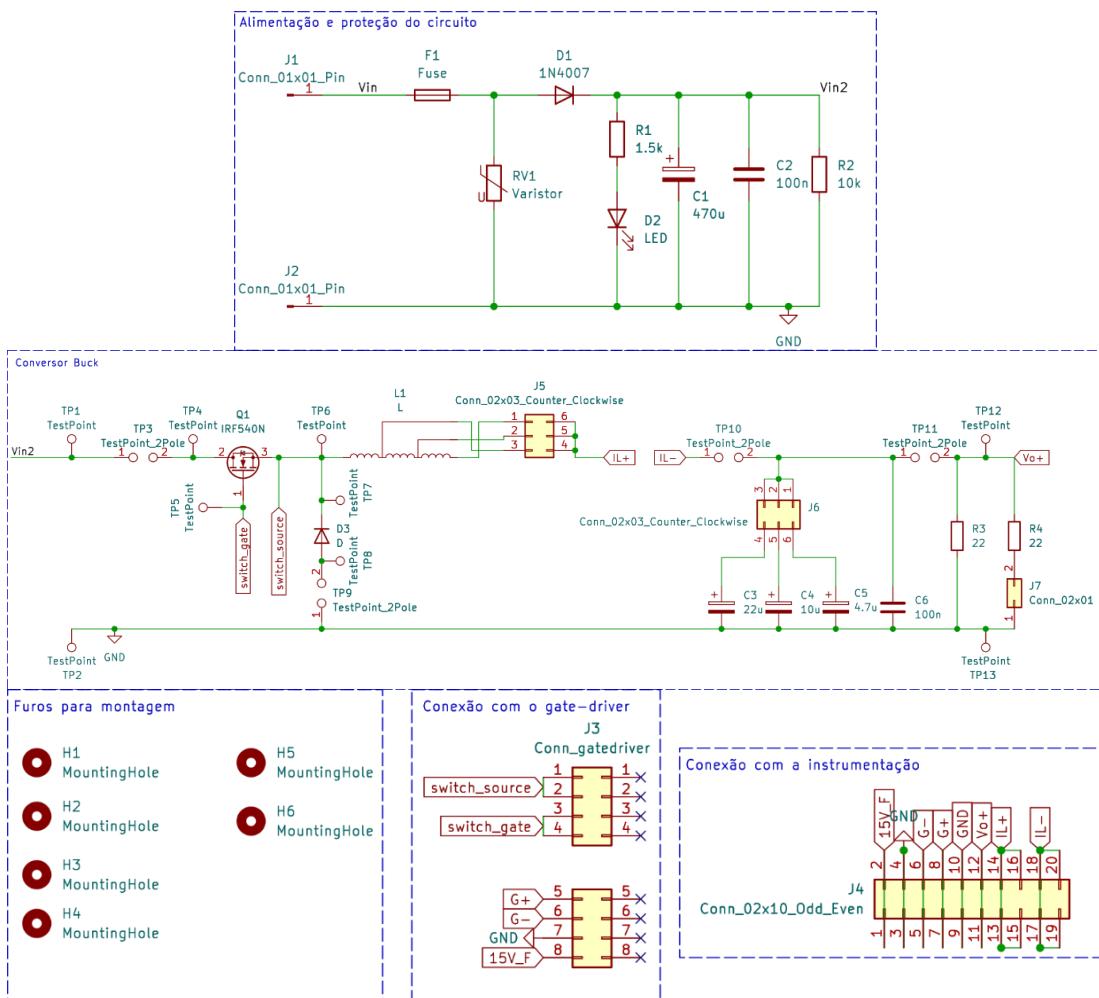
Este capítulo aborda a construção física do conversor. Será desenvolvido um esquemático e *layout* da placa de circuito impresso (PCI) do conversor no *software* KiCad. Com os arquivos gerados pela ferramenta mencionada, a PCI será fresada no laboratório de prototipação do Grupo de Eletrônica de Potência e Controle (GEPOC). Após isso, os componentes serão soldados também nos laboratórios do GEPOC.

5.1 Esquemático do conversor *Buck*

A Figura 11 apresenta o esquemático do conversor *Buck* desenvolvido no *software* Kicad. Este projeto foi realizado com o intuito de facilitar o aprendizado do aluno e permitir diferentes resultados alterando alguns componentes, assim permitindo trabalhos diferentes entre os grupos.

O projeto foi desenvolvido com três opções de valores de capacitância e indutância, e dois valores de resistência. O conversor é alimentado por uma fonte de bancada do laboratório e é conectado a duas PCIs adicionais: (i) o *gate-driver* e (ii) a instrumentação e controle.

Figura 11 - Esquemático conversor Buck desenvolvido no software Kicad



Fonte: Próprio autor.

5.2 Lista de componentes

A Tabela 4 mostra os componentes necessários para a implementação de uma unidade do conversor. Além dos componentes listados, para a leitura do valor de corrente nos pontos de interesse são utilizados pontos de teste de corrente, que são fios de 3 mm de diâmetro distribuídos ao longo do conversor.

Tabela 4 - Lista de componentes conversor *Buck*

(continua)

Nome do componente	Tipo do componente	Tecnologia do componente	Modelo	Faixa de tensão	Faixa de corrente
J1, J2	Conecotor Banana	THT		100 V	10 A

Tabela 4 - Lista de componentes conversor *Buck*

(continuação)

J3	<i>Gate-driver</i> conector	PinHeader	PinHeader 2x2x04 P2.54mm Vertical	100 V	3 A
J4	Conecotor PinHeader para Instrumentação	PinHeader	PinHeader 2x10 P2.54mm Horizontal	100 V	3 A
J5, J6	Conecotor PinHeader	PinHeader	PinHeader 2x06 P2.54mm Vertical	100 V	3 A
J7	Conecotor PinHeader	PinHeader	PinHeader 2x02 P2.54mm Vertical	100 V	3 A
Q1	Chave	TO220	IRF540N	100 V	9 A
F1	Fusível	THT	—	100 V	2 A
D1	Díodo proteção	THT	1N4007	700 V	1 A
D2	LED	THT	LED	100 V	30 mA
D3	Díodo potência	TO220	MUR820	200 V	8 A
C1	Capacitor	Eletrolítico THT	470 µF	63 V	—
C2, C6	Capacitor	Cerâmico THT	100 nF	63 V	—
C3	Capacitor	Eletrolítico THT	22 µF	63 V	—
C4	Capacitor	Eletrolítico THT	10 µF	63 V	—
C5	Capacitor	Eletrolítico THT	4,7 µF	63 V	—
L1, L2, L3	Indutor	Ferrite Modelo EE- 42/20	2,8/3,7/5,6 mH	—	1,847 A AWG 21

Tabela 4 - Lista de componentes conversor *Buck*

(conclusão)

R1	Resistência	THT 0,25 W	1,5kΩ	—	—
R2	Resistência	THT 0,25 W	10kΩ	—	—
R3, R4	Resistência	THT 10 W	22Ω	—	—
RV1	Varistor	THT	—	75V	—
TP1, TP2, TP4, TP5, TP6, TP7, TP8, TP12, TP13	TestPoint Single	THT	—	100 V	—
TP3, TP9, TP10, TP11	TestPoint Double	THT	—	—	60A

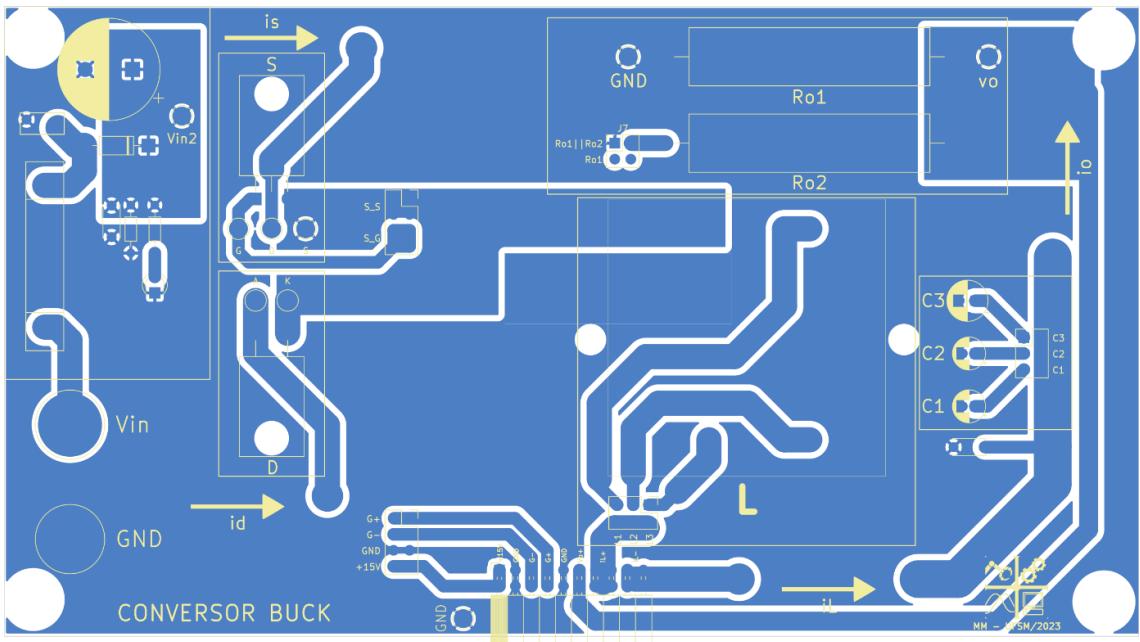
Fonte: Próprio autor.

5.3 Layout desenvolvido no software Kicad

As Figura 12 e Figura 13 mostram o *layout* e visão 3D, respectivamente, do conversor *Buck* projetado no *software* Kicad. As trilhas de alta corrente do conversor foram dimensionadas com as maiores larguras possíveis, sendo utilizadas áreas em duas partes da placa, próximo da entrada dos resistores de carga e da chave.

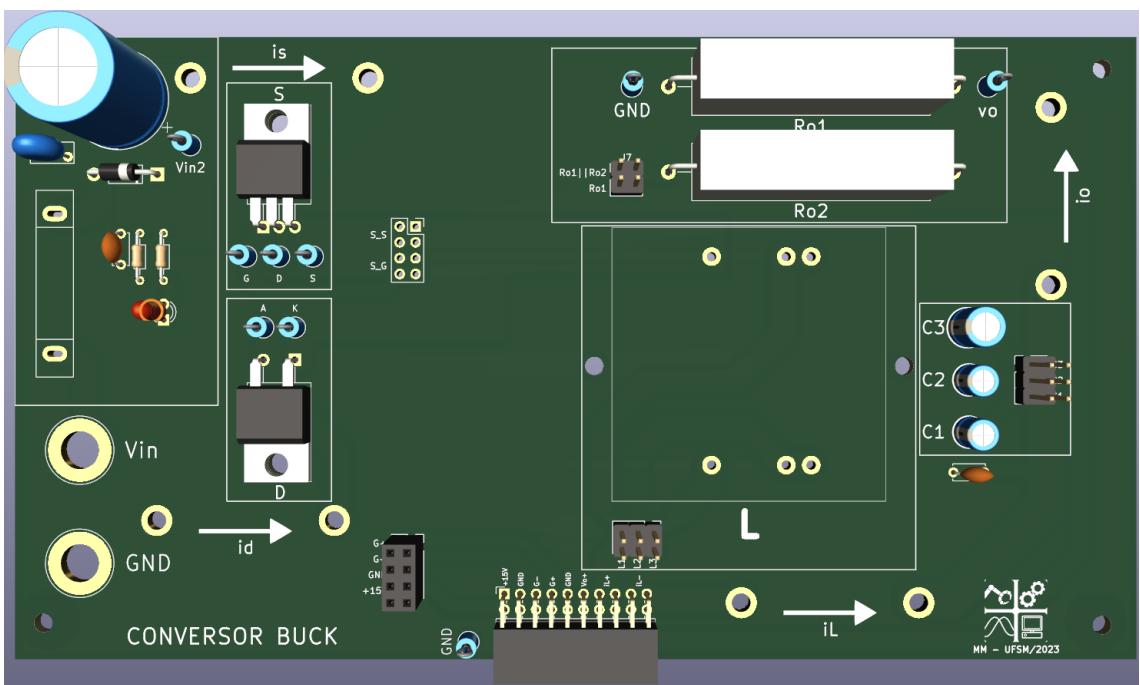
Os componentes foram posicionados e identificados com a intenção de facilitar a compreensão do aluno. Vários pontos de teste foram espalhados pela placa para facilitar o monitoramento das variáveis de interesse, assim facilitando a comparação com as referências teóricas.

Figura 12 - Layout da PCI do conversor no Kicad



Fonte: Próprio autor

Figura 13 - Visão 3D da PCI do conversor no Kicad

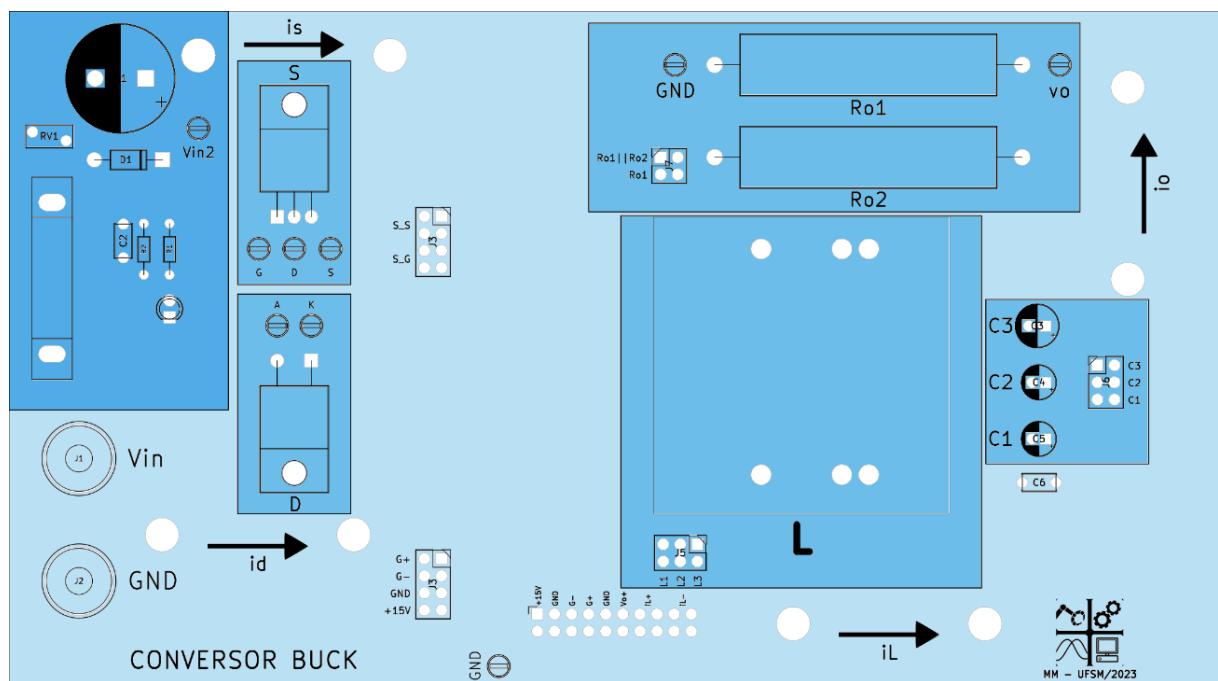


Fonte: Próprio autor

5.4 Conversor Buck implementado

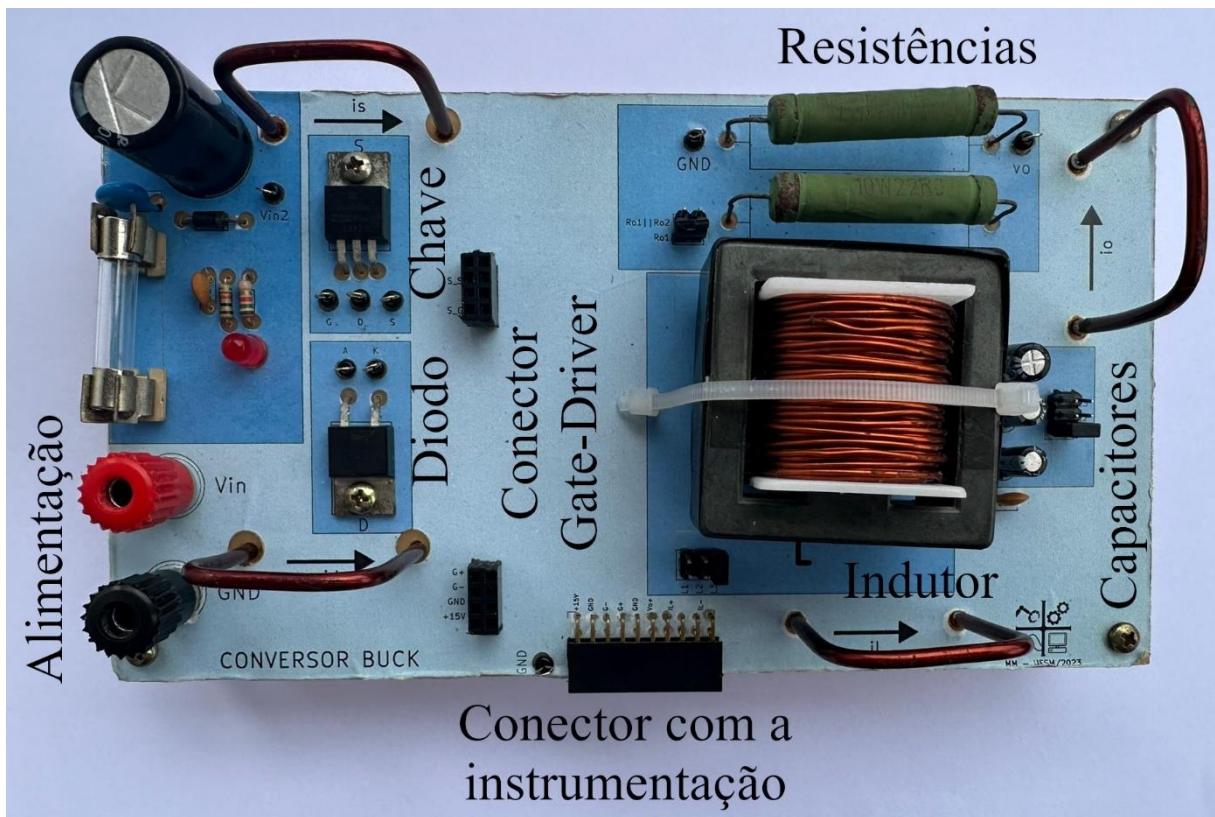
Foi necessário colar um adesivo na PCI do conversor para a identificação dos seus componentes, pois há a limitação da máquina CNC do laboratório que não aplica o *silkscreen*. A Figura 14 mostra este adesivo. A versão final do conversor *Buck* fabricado pode ser vista na Figura 15.

Figura 14 - Adesivo silkscreen colado no conversor



Fonte: Próprio autor.

Figura 15 - Conversor Buck implementado



Fonte: Próprio autor.

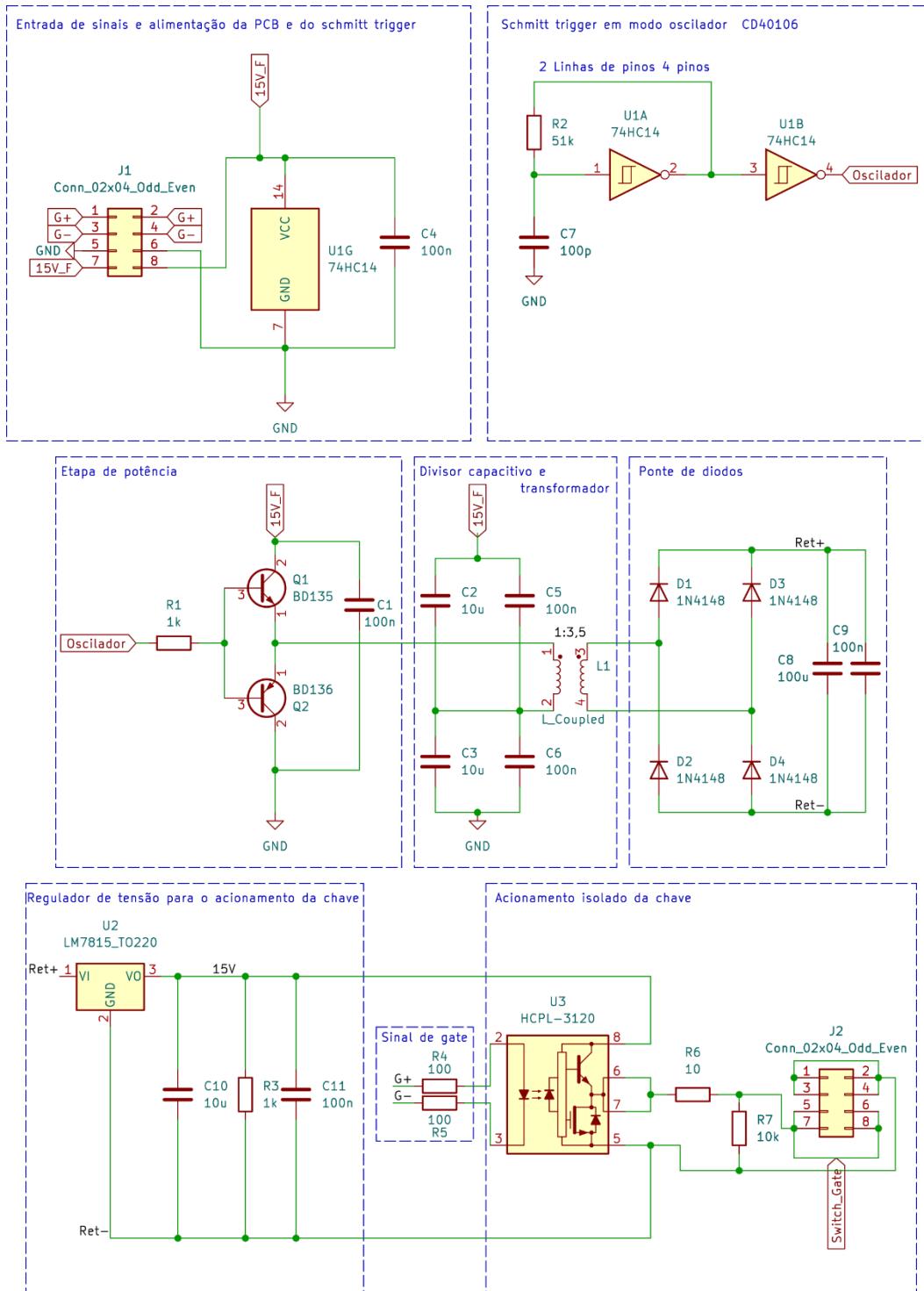
CAPÍTULO 6 PROJETO DO CIRCUITO DE *GATE-DRIVER*

Este capítulo aborda o projeto e a implementação do circuito de *gate-driver*. Este circuito foi projetado para o acionamento da chave IRF 530 utilizada no conversor *Buck*. Através da consulta ao *datasheet* foi decidido que as tensões utilizadas para acionar e desativar a chave serão de 15 V e 0 V, respectivamente.

6.1 Esquemático do circuito de *gate-driver*

Este circuito de *gate-driver* foi projetado para ser isolado, assim podendo ser utilizado em qualquer conversor em projetos que deem sequência à série de conversores utilizados para ensino em Eletrônica de Potência. Por ser um *gate-driver* isolado, foi necessária a utilização de um optoacoplador para separar o sinal lógico de entrada (proveniente do microcontrolador), do sinal de saída (aplicado entre *gate* e *source* do MOSFET), um transformador de alta frequência para isolar a alimentação, um oscilador em modo multivibrador astável para gerar a onda quadrada para o transformador e um regulador de tensão para garantir a tensão de 15 V de acionamento da chave. O esquemático desenvolvido no *software Kicad* é apresentado na Figura 16.

Figura 16 - Esquemático do circuito de *gate-driver* desenvolvido no Kicad



Fonte: Próprio autor.

6.2 Lista de componentes do circuito de *gate-driver*

A Tabela 5 apresenta a lista de componentes necessários para a implementação de uma unidade do circuito de *gate-driver*. O componente selecionado para ser o oscilador é o

comparador *schmitt-trigger* modelo CD40106B (TEXAS INSTRUMENTS®, 2003) e o optoacoplador que isola o sinal de entrada do sinal de acionamento é o HCPL-3120 (HP). Já o transformador projetado foi montado em laboratório com uma relação de espiras de 1:3,5.

Tabela 5 - Lista de componentes do circuito de *gate-driver*

(continua)

Nome do componente	Tipo do componente	Tecnologia do componente	Modelo	Faixa de tensão	Faixa de corrente
U1	<i>Schmitt trigger</i>	DIP14	CD40106	15 V	10 mA (Entrada) 100 mA (Saída)
U2	Regulador de tensão	TO220	LM7815	30 V	1 A
U3	Optoacoplador	DIP8	HCPL-3120	30 V	2 A
Q1	Transistor NPN	TO126	BD135	45 V	1,5 A
Q2	Transistor PNP	TO126	BD136	45 V	1,5 A
C1, C4, C5, C6, C9, C11	Capacitor	Cerâmico THT	100 nF	63 V	—
C2, C3, C10	Capacitor	Eletrolítico THT	10 µF	63 V	—
C7	Capacitor	Cerâmico THT	100 pF	63 V	—
C8	Capacitor	Eletrolítico THT	100 µF	63 V	—
R1, R3	Resistência	THT 0,25 W	1 kΩ	—	—
R2	Resistência	THT	51 k	—	—
R4, R5	Resistência	THT 0,25 W	100 Ω	—	—
R6	Resistência	THT 0,25 W	10 Ω	—	—
R7	Resistência	THT 0,25 W	10 kΩ	—	—
L_Coupled	Transformador	Anel de ferrite A = 15 x B = 8,5 x C = 8mm	1:3,5	—	200 mA

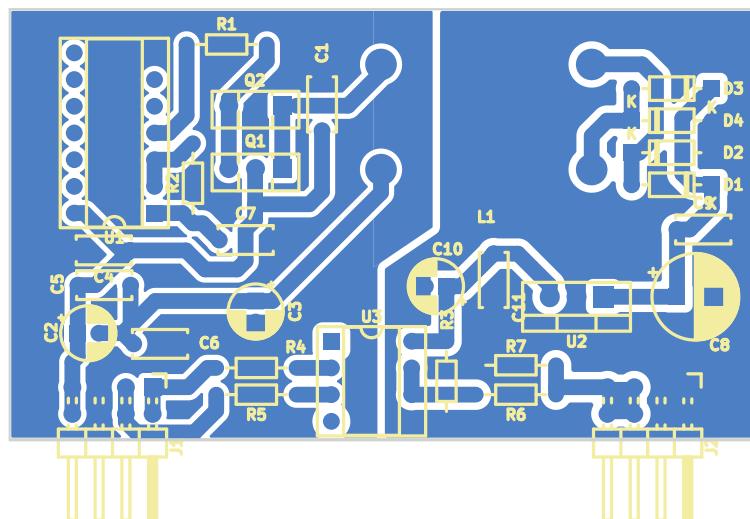
Tabela 5 - Lista de componentes do circuito de *gate-driver*

(conclusão)

J1	Conektor	PinHeader	PinHeader 2x04 P2.54mm Horizontal	–	3A
J2	Conektor	PinHeader	PinHeader 2x04 P2.54mm Horizontal	–	3A

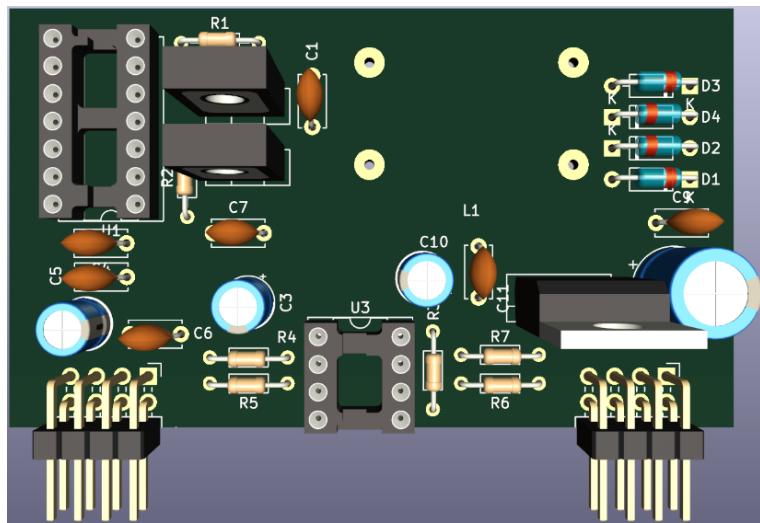
6.3 Layout do circuito de *gate-driver* no software Kicad

As Figura 17 e Figura 18 apresentam, respectivamente, o *layout* do circuito de *gate-driver* desenvolvido e a visualização 3D do mesmo. Como o projeto tem fins didáticos, foram escolhidos componentes THT que facilitam a troca e manutenção por alunos com pouca experiência. Este *gate-driver* também foi projetado para ser compatível com *protoboards*. Os pontos mais importantes no desenvolvimento deste *layout* foram um curto caminho entre o sinal PWM e o acionamento da chave, e que a tensão de acionamento fosse isolada da entrada de energia.

Figura 17 - *Layout* do circuito de *gate-driver* no Kicad

Fonte: Próprio autor.

Figura 18 - Visualização 3D do circuito de *gate-driver* no Kicad

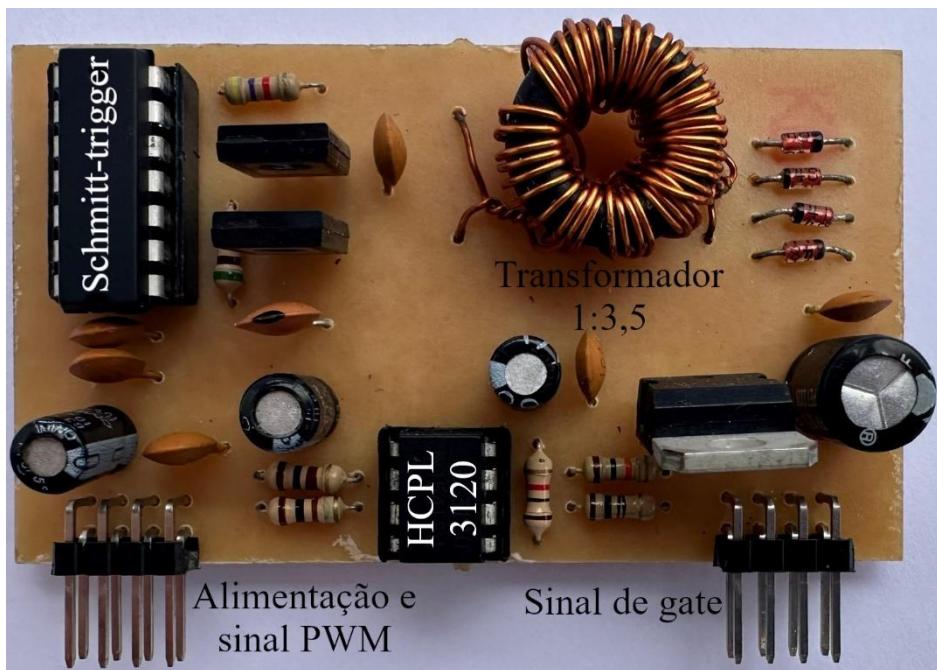


Fonte: Próprio autor.

6.4 Circuito de *gate-driver* implementado

A Figura 19 apresenta o circuito de *gate-driver* fabricado no laboratório de prototipação. A sua compatibilidade com *protoboard* permite a utilização em testes e desenvolvimento com chaves em outras funções.

Figura 19 – Circuito de *gate-driver* fabricado



Fonte: Próprio autor.

CAPÍTULO 7 PROJETO DO CIRCUITO DE INSTRUMENTAÇÃO

Este projeto tem como objetivo o controle da tensão de saída do conversor através das leituras de tensão de saída e corrente do indutor e do controle do sinal de acionamento da chave. Para isso há a escolha do sensor de corrente, microcontrolador para o controle, instrumentação de tensão e corrente e alimentação. Todos esses pontos serão comentados ao longo deste capítulo.

7.1 Dimensionamento da instrumentação de corrente e tensão

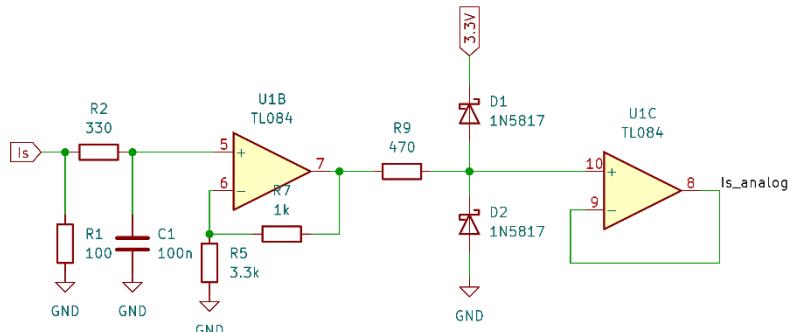
A instrumentação de corrente e tensão tem por objetivo o condicionamento dos sinais de interesse: (i) a corrente do indutor de filtro e (ii) a tensão do capacitor de filtro. Para o projeto dessa instrumentação foram adotados como valores limites de corrente 0 a 5 A e para a tensão 0 a 40 V.

Esses sinais precisam ser condicionados ao microcontrolador escolhido, cujas GPIO (*General Purpose Input Output*) possuem limite de operação em 3,3 V. Para isso são utilizados amplificadores operacionais (amp-ops), dois para cada circuito. O primeiro fará o condicionamento do sinal para os níveis de tensão requisitados pelo microcontrolador e o segundo fará parte de um circuito de proteção, para garantir que os valores de tensão nas portas do microcontrolador não ultrapassem o valor máximo.

A primeira etapa da instrumentação irá: ajustar o ganho das variáveis medidas e a segunda etapa, a proteção. A etapa de proteção do circuito usará diodos *schottky* que terão a tensão máxima controlada por um regulador linear de 3,3 V.

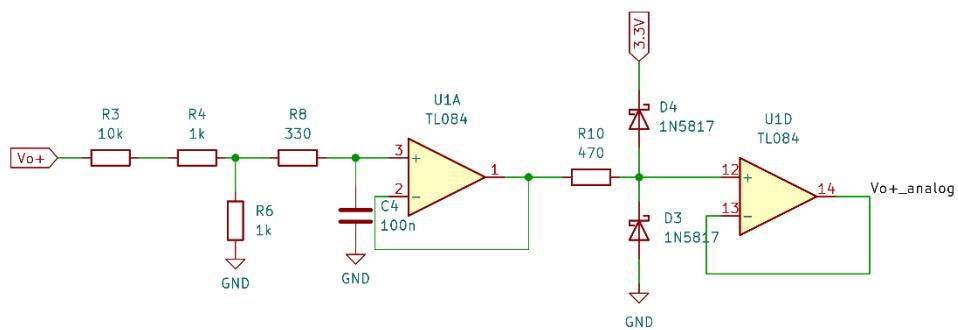
Os circuitos são apresentados nas Figura 20 e Figura 21. O sinal I_s é a saída do sensor de corrente utilizado e a tensão V_o+ a tensão de saída do conversor. A saída dos circuitos de instrumentação, I_s_analog e V_o+_analog são os sinais enviados para a leitura dos ADCs do microcontrolador.

Figura 20 – Circuito de instrumentação de corrente



Fonte: Próprio autor.

Figura 21 – Circuito de instrumentação de tensão



Fonte: Próprio autor.

7.2 Esquemático da placa de instrumentação e controle

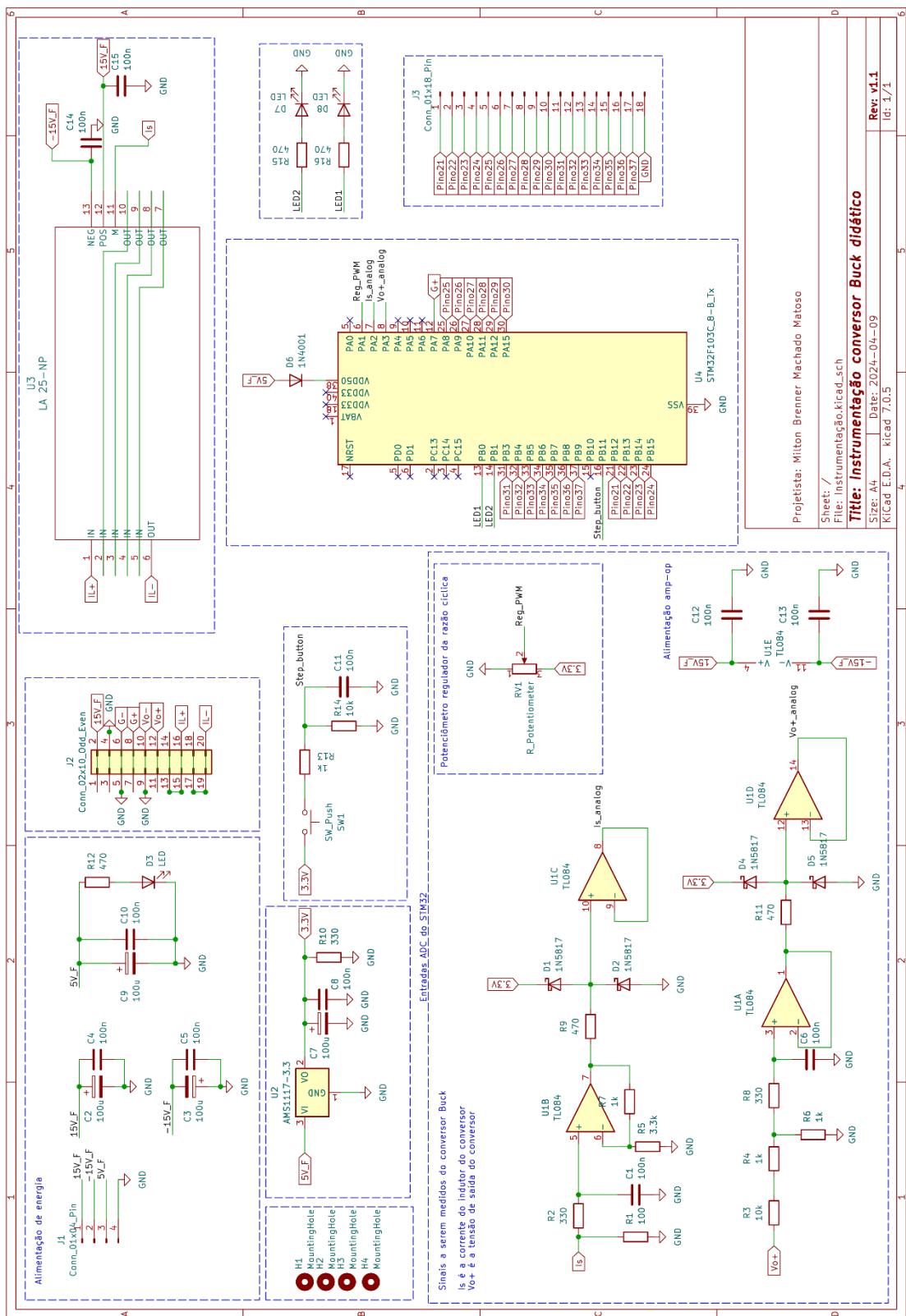
O projeto da placa de instrumentação e controle visa atender algumas especificações. Por exemplo, (i) o controle em malha aberta precisa de um sinal de entrada que altere a razão-cíclica aplicado à chave; (ii) o controle em malha fechada precisa da leitura das variáveis a serem controladas (I_L e V_o); e (iii) a saída da placa de instrumentação deve prover um sinal lógico PWM para comando do circuito de *gate-driver*.

Para cumprir as especificações desses três pontos mencionados, foram escolhidos: (i) potenciômetro; (ii) sensor de corrente efeito hall para a medição de corrente e divisor resistivo para medição de tensão; e (iii) um microcontrolador capaz de fazer a leitura das informações provenientes da instrumentação, cálculo das ações de controle e geração do sinal PWM. O sensor de efeito hall escolhido é o LA25-NP e o microcontrolador escolhido é o STM32F103C8T6, comercialmente conhecido como *bluepill*.

Para alimentar o microcontrolador é necessária uma fonte de alimentação de 3,3 V ou 5 V. Já a alimentação do sensor utilizado e do amp-op é com uma tensão simétrica de +15 V e -15 V. Para isso foi projetada uma fonte de alimentação auxiliar (que é apresentada no Apêndice A – Projeto da fonte auxiliar) com as tensões de +15 V, +5 V, 0 V e -15 V. Além da fonte de alimentação, foram adicionados no circuito dois LEDs a serem acionados pelo microcontrolador, um LED na entrada de energia para mostrar que a placa está energizada e um botão para teste de resposta ao degrau da malha de controle.

Com essas informações em mente, o projeto do circuito de instrumentação e controle é mostrado na Figura 22.

Figura 22 - Esquemático da placa de instrumentação e controle



Fonte: Próprio autor.

7.3 Lista de componentes da PCI de instrumentação e controle

A Tabela 6 apresenta a lista de componentes para a produção de uma unidade da PCI de instrumentação e controle. Os principais componentes deste projeto são o sensor hall de corrente LA25-NP (LEM), o microcontrolador STM32F103C8T6 (STMICROELECTRONICS), o amp-op TL084 (STMICROELECTRONICS) e o regulador de tensão AMS1117-3,3 (ADVANCED MONOLITHIC SYSTEMS).

Tabela 6 - Componentes da PCI de instrumentação e controle

(continua)

Nome do componente	Tipo do componente	Tecnologia do componente	Modelo	Faixa de tensão	Faixa de corrente
J1	Conektor KK	THT	KK 01x04 Vertical	–	3 A–
J2	Conektor PinHeader	PinHeader	PinHeader 2x10 P2.54mm Horizontal	–	3 A–
J3	Conektor PinHeader	PinHeader	PinHeader 1x18 P2.54mm Vertical	–	3 A–
U1	AMPOP	DIP-14	TL084	±30 V	–
U2	Regulador tensão 3.3V	SOT-223-3	AMS1117-3.3	12 V	1 A
U3	Sensor Hall	THT	LA25-NP	±15 V	25 A
U4	Microcontrolador	DIP-40	STM32F103C	5V	400 mA–
D1, D2, D4, D5	Diodo Schottky	THT	1N5817	14 V	1 A
D3, D6, D7	LED	THT	LED	–	30 mA
D8	Diodo	THT	1N4001	35 V	1 A
C1, C4, C5, C6, C8, C10, C11, C12, C13, C14, C15	Capacitor	Cerâmico THT	100 nF	63 V	–
C2, C3, C7, C9	Capacitor	Eletrolítico THT	100 µF	63 V	–

Tabela 6 - Componentes da PCI de instrumentação e controle

(conclusão)

R1	Resistência	THT 0,25 W	100 Ω	—	—
R2, R8, R10	Resistência	THT 0,25 W	330 Ω	—	—
R3, R14	Resistência	THT 0,25 W	10 k Ω	—	—
R4, R6, R7, R13	Resistência	THT 0,25 W	1 k Ω	—	—
R5	Resistência	THT 0,25 W	3,3 k Ω	—	—
R9, R11, R12, R15, R16	Resistência	THT 0,25 W	470 Ω	—	—
RV1	Potenciômetro	THT	0-10 k Ω	—	—
SW1	Botão	THT	—	—	1 A

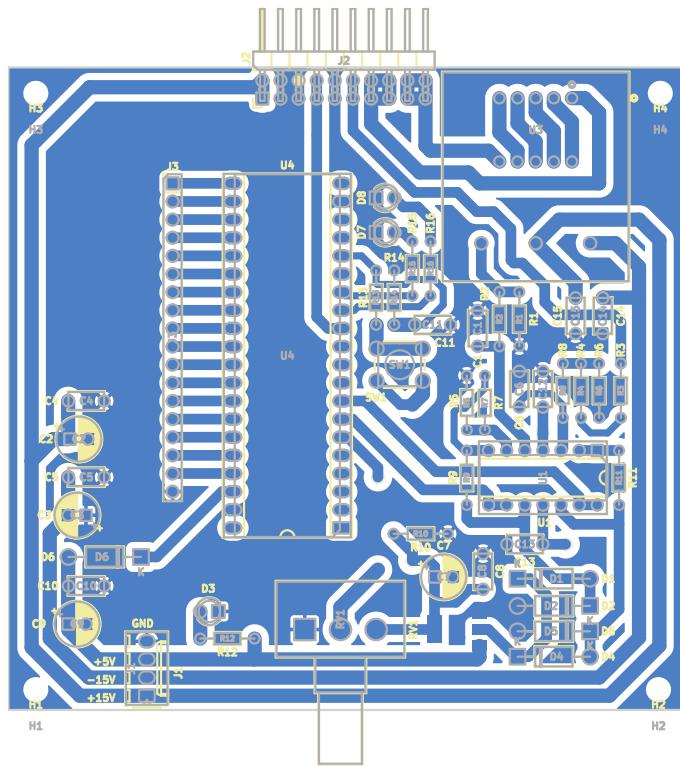
Fonte: Próprio autor.

7.4 Layout da PCI de instrumentação e controle no software Kicad

O *layout* da PCI de instrumentação e controle é representado na Figura 23. Na esquerda, conectado próximo da entrada de energia, estão os capacitores para regular a tensão de entrada, e toda a coluna da direita é responsável pela aquisição e instrumentação do sinal do conversor para o microcontrolador.

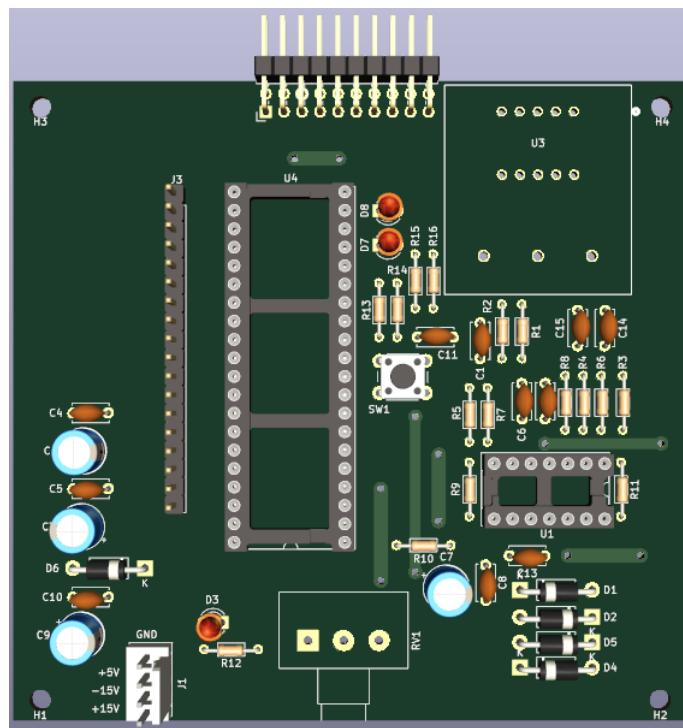
Um desafio deste projeto foi a utilização de apenas uma camada de cobre, sendo necessários alguns cruzamentos pela parte superior. No total foram 6 cruzamentos, sendo que 3 deles foram para melhorar o caminho de retorno da corrente pelo plano de terra da placa. Com o *layout* da placa foi possível também ter uma visualização prévia em 3D, conforme se observa na Figura 24.

Figura 23 - *Layout* da PCI de instrumentação e controle



Fonte: Próprio autor.

Figura 24 - Visualização 3D da PCI de instrumentação e controle

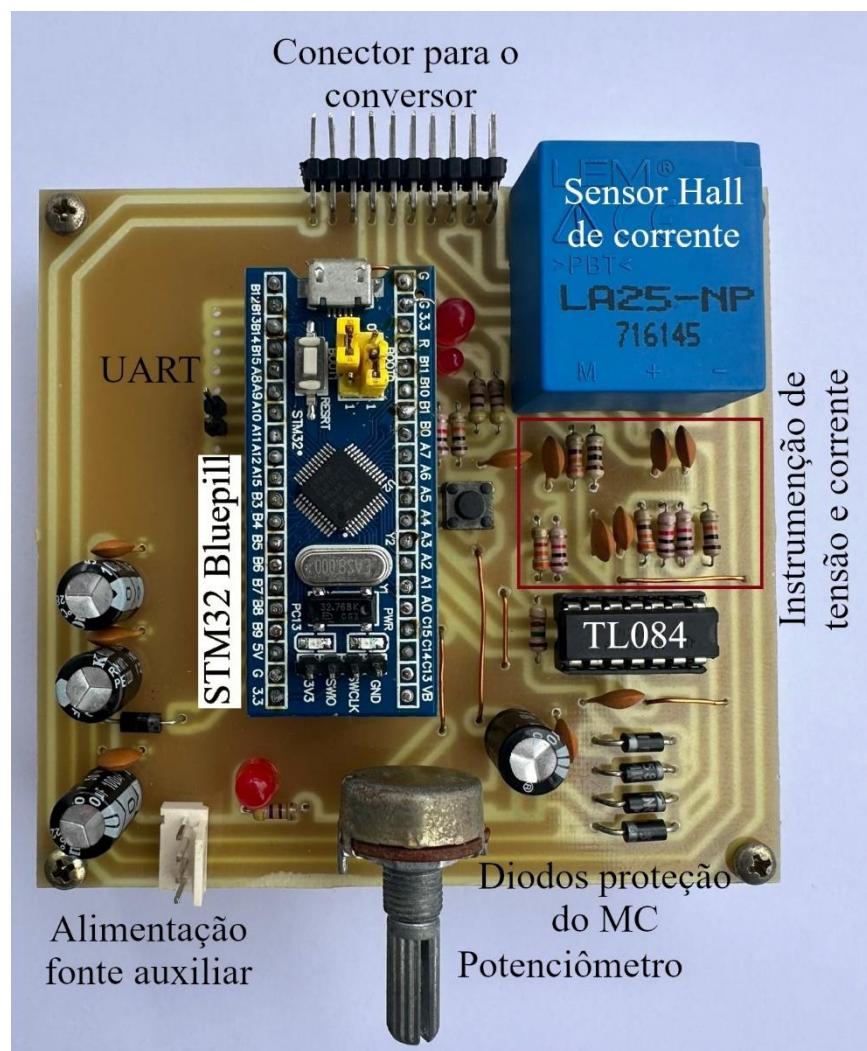


Fonte: Próprio autor.

7.5 PCI de instrumentação e controle implementada

A Figura 25 mostra a PCI de instrumentação e controle implementada. Os cruzamentos pela parte superior da placa podem ser vistos na forma de *jumpers*. A alimentação (conector KK) é conectada na fonte de alimentação auxiliar apresentada no Apêndice A – Projeto da fonte auxiliar.

Figura 25 - PCI de instrumentação e controle implementada

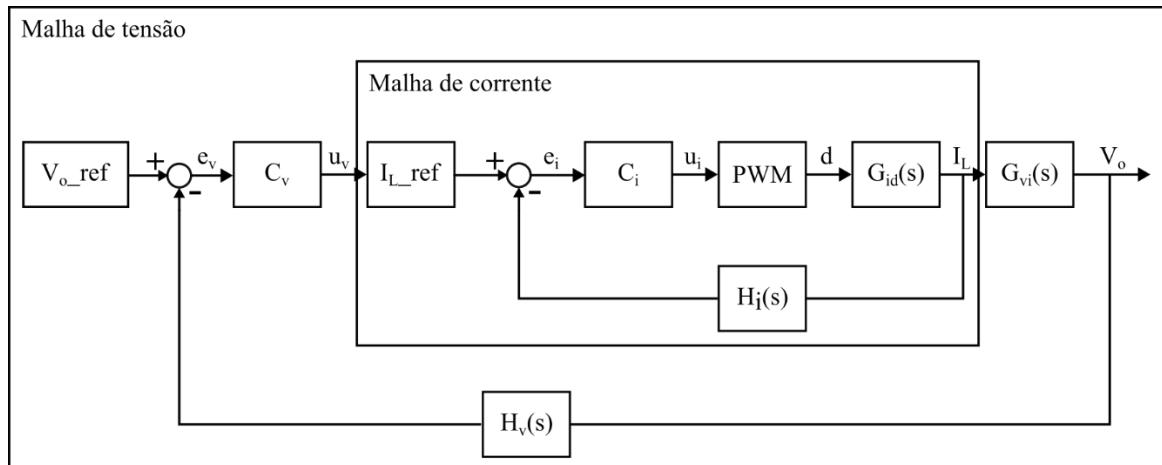


Fonte: Próprio autor.

CAPÍTULO 8 MODELAGEM DINÂMICA DO CONVERSOR

Para o desenvolvimento do sistema de controle, primeiramente é necessário se obter o modelo dinâmico da planta a ser controlada. Este capítulo aborda a modelagem dinâmica do conversor *Buck*. Serão consideradas duas plantas distintas: uma que relaciona a corrente no indutor de filtro com a razão-cíclica $G_{id}(s)$ e outra que relaciona a tensão no capacitor de filtro com a corrente no indutor $G_{vi}(s)$ não diagrama do controle proposto (com duas malhas) pode ser visualizado na Figura 26.

Figura 26 - Malha de controle



Fonte: Próprio autor.

A variável final a ser controlada é a tensão de saída, V_o , e a única ação que temos sobre o conversor é a alteração da razão-cíclica.

Para a obtenção da função de transferência da planta $G_{id}(s)$ apresentada na Figura 26 será utilizada a técnica de modelagem por espaço de estados. Com a malha de corrente implementada, esta será substituída no circuito do conversor por uma fonte de corrente para a obtenção da função de transferência $G_{vi}(s)$. Essa abordagem pode ser adotada em virtude da frequência de corte da malha de tensão (malha externa) ser, pelo menos, uma década menor do que a frequência de corte da malha de corrente (malha interna).

8.1 Modelagem da planta para controle da corrente no indutor de filtro

A obtenção da função de transferência da planta $G_{id}(s)$ utilizará as técnicas de modelagem por espaço de estados e modelo de pequenos sinais CA apresentadas em (ERICKSON; MAKSIMOVIĆ, 2001).

8.1.1 Modelo médio em espaço de estados

A descrição por espaço de estados de um sistema dinâmico é um pilar na teoria de controle moderno, dela deriva-se o modelo de pequenos sinais que será utilizado para obter os modelos das plantas do sistema (ERICKSON; MAKSIMOVIĆ, 2001).

A descrição por espaço de estados é uma forma canônica de escrever equações diferenciais que descrevem um sistema. A parte física das variáveis de estados de um sistema normalmente são associadas com os elementos que armazenam energia, para o caso do conversor *Buck* as variáveis de estado independentes são a corrente do indutor e a tensão do capacitor.

As equações de estado de um sistema podem ser escritas na forma compacta matricial, conforme (24) e (25).

$$\mathbf{K} \frac{d\mathbf{x}(t)}{dt} = \mathbf{Ax}(t) + \mathbf{Bu}(t) \quad (24)$$

$$\mathbf{y}(t) = \mathbf{Hx}(t) + \mathbf{Eu}(t) \quad (25)$$

onde o vetor $\mathbf{x}(t)$ é um vetor contendo todas as variáveis de estado, como a corrente dos indutores, tensão dos capacitores, etc. O vetor de entradas $\mathbf{u}(t)$ contém as variáveis de entrada independentes do sistema, como a tensão de entrada V_{in} . A derivada do vetor de estados é um vetor cujos elementos são iguais às derivadas dos elementos correspondentes do vetor de estados.

A matriz \mathbf{K} é a matriz contendo os valores de capacidade e indutância do circuito. A matriz \mathbf{A} é a matriz de estados e a matriz \mathbf{B} é a matriz de entradas, ambas contêm constantes de proporcionalidade. O vetor $\mathbf{y}(t)$ é o vetor de saídas. A matriz \mathbf{H} é a matriz de saídas e \mathbf{E} é a matriz *feedforward*, as duas também contêm constantes de proporcionalidade.

No caso do conversor *Buck*, ele possui apenas duas variáveis de estado: a corrente do indutor e a tensão do capacitor. O circuito possui apenas uma variável de entrada independente,

V_{in} . A saída do sistema é a tensão de saída, V_o . Assim a matriz \mathbf{K} e os vetores $\mathbf{u}(t)$ e $\mathbf{y}(t)$ ficam:

$$\mathbf{K} = \begin{pmatrix} C & 0 \\ 0 & L \end{pmatrix}, \quad \mathbf{u}(t) = [V_{in}], \quad \mathbf{y}(t) = [V_o]$$

As matrizes **A**, **B**, **H** e **E** terão uma especificidade na hora de serem calculadas, pois apresentam diferentes valores para cada etapa de operação do conversor. As matrizes da primeira etapa serão chamadas de \mathbf{A}_1 , \mathbf{B}_1 , \mathbf{H}_1 e \mathbf{E}_1 , e as matrizes da segunda etapa de \mathbf{A}_2 , \mathbf{B}_2 , \mathbf{H}_2 e \mathbf{E}_2 .

Na primeira etapa de operação do conversor, cujo intervalo de duração é definido por $0 < t < DT_s$, tem-se as seguintes equações diferenciais:

$$L \frac{di_L(t)}{dt} = V_{in} - V_o \quad (26)$$

$$C \frac{dv_c(t)}{dt} = -v_o(t) \frac{1}{R_o} \quad (27)$$

Logo, as matrizes ficam da forma:

$$\mathbf{A}_1 = \begin{pmatrix} 0 & -1 \\ 1 & -\frac{1}{R_o} \end{pmatrix}, \quad \mathbf{B}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \mathbf{H}_1 = (1 \ 0), \quad \mathbf{E}_1 = (0);$$

Na segunda etapa de operação do conversor, cujo intervalo de operação é definido por $DT_s < t < T_s$, tem-se as seguintes equações diferenciais:

$$L \frac{di_L}{dt} = -v_o(t) \quad (28)$$

$$C \frac{dv_c(t)}{dt} = i_L(t) - \frac{v_o(t)}{R_o} \quad (29)$$

Logo, as matrizes ficam da forma:

$$\mathbf{A}_2 = \begin{pmatrix} 0 & -1 \\ 1 & -\frac{1}{R_o} \end{pmatrix}, \quad \mathbf{B}_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \mathbf{H}_2 = (1 \ 0), \quad \mathbf{E}_2 = (0);$$

Considerando a operação em regime permanente, as matrizes médias são dadas por:

$$\mathbf{A} = \mathbf{A}_1 \times D + \mathbf{A}_2 \times (1-D) \quad (30)$$

$$\mathbf{B} = \mathbf{B}_1 \times D + \mathbf{B}_2 \times (1-D) \quad (31)$$

$$\mathbf{H} = \mathbf{H}_1 \times D + \mathbf{H}_2 \times (1-D) \quad (32)$$

$$\mathbf{E} = \mathbf{E}_1 \times D + \mathbf{E}_2 \times (1-D) \quad (33)$$

Resultando em:

$$\mathbf{A} = \begin{pmatrix} 0 & -1 \\ 1 & -\frac{1}{R_o} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} D \\ 0 \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad \mathbf{E} = \begin{pmatrix} 0 \end{pmatrix};$$

Dado que as frequências naturais do conversor são muito inferiores à frequência de chaveamento, o vetor de estados de equilíbrio $\mathbf{X} = (I_L \ V_o)^T$ e o vetor de saídas de equilíbrio $\mathbf{Y} = (I_L)$ podem ser obtidos através de:

$$\mathbf{X} = -\mathbf{A}^{-1}\mathbf{B}\mathbf{U} \quad (34)$$

$$\mathbf{Y} = (-\mathbf{H}\mathbf{A}^{-1}\mathbf{B} + \mathbf{E})\mathbf{U} \quad (35)$$

Assim:

$$\mathbf{X} = \begin{pmatrix} DV_{in} \\ R_o \\ DV_{in} \end{pmatrix}, \quad \mathbf{Y} = \begin{pmatrix} DV_{in} \\ R_o \end{pmatrix} \quad (36)$$

8.1.2 Modelo CA de pequenos sinais

Após a determinação das equações de equilíbrio, determinar-se-á o modelo CA de pequenos sinais. De acordo com (ERICKSON; MAKSIMOVIĆ, 2001), as equações que descrevem o modelo linear de pequenos sinais são expressas nas equações (37) e (38), onde $\hat{\mathbf{u}}_p(t) = [\hat{\mathbf{u}}(t) \ \hat{d}(t)]^T$. Chama-se atenção para o fato que $\hat{\mathbf{u}}(t)$ e $\hat{d}(t)$ são pequenas variações CA no sinal de entrada e razão-cíclica, e $\hat{\mathbf{x}}(t)$ e $\hat{\mathbf{y}}(t)$ são as pequenas variações CA resultantes nos vetores de estado e saída. Para que o modelo linear das equações mencionadas seja válido, é assumido que os valores quiescentes são muito maiores que os sinais de perturbação, ou seja, $\|\mathbf{U}\| \gg \|\hat{\mathbf{u}}(t)\|$, $\|D\| \gg \|\hat{d}(t)\|$, $\|\mathbf{X}\| \gg \|\hat{\mathbf{x}}(t)\|$ e $\|\mathbf{Y}\| \gg \|\hat{\mathbf{y}}(t)\|$.

$$\frac{d\hat{\mathbf{x}}(t)}{dt} = \mathbf{A}_p^{ccm} \hat{\mathbf{x}}(t) + \mathbf{B}_p^{ccm} \hat{\mathbf{u}}_p(t) \quad (37)$$

$$\hat{\mathbf{y}}(t) = \mathbf{H}_p^{ccm} \hat{\mathbf{x}}(t) + \mathbf{E}_p^{ccm} \hat{\mathbf{u}}_p(t) \quad (38)$$

Onde

$$\mathbf{A}_p^{ccm} = \mathbf{K}^{-1} \mathbf{A} \quad (39)$$

$$\mathbf{B}_p^{ccm} = \mathbf{K}^{-1} [\mathbf{B} - (\mathbf{A}_1 - \mathbf{A}_2) \mathbf{X} + (\mathbf{B}_1 - \mathbf{B}_2) \mathbf{U}] \quad (40)$$

$$\mathbf{H}_p^{ccm} = \mathbf{H} \quad (41)$$

$$\mathbf{E}_p^{ccm} = [\mathbf{E} - (\mathbf{H}_1 - \mathbf{H}_2) \mathbf{X} + (\mathbf{E}_1 - \mathbf{E}_2) \mathbf{U}] \quad (42)$$

A função de transferência do conversor pode ser obtida aplicando-se a Transformada de Laplace nas equações (37) e (38), resultando em:

$$\hat{\mathbf{y}}(s) = \mathbf{H}_p^{ccm} (sI_{2x2} - \mathbf{A}_p^{ccm}) \mathbf{B}_p^{ccm} \hat{\mathbf{u}}_p(s) \quad (43)$$

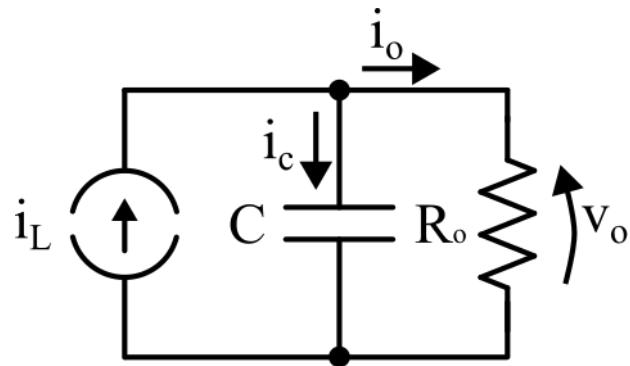
Aplicando para o caso do conversor *Buck*, chega-se à função de transferência que correlaciona a corrente do indutor à razão-cíclica:

$$G_{id}(s) = \frac{\hat{i}_L(s)}{\hat{d}(s)} = \frac{V_{in}}{L} \frac{s + \frac{1}{CR_o}}{s^2 + \frac{s}{CR_o} + \frac{1}{CL}} \quad (44)$$

8.2 Modelagem da planta para controle da tensão no capacitor de filtro

Uma vez projetada a malha de controla da corrente (malha interna), do ponto de vista da malha de tensão (malha externa, mais lenta) a corrente no indutor pode ser considerada constante. Assim é feita uma alteração no circuito para considerar, a partir de agora, o indutor como uma fonte de corrente, conforme o circuito apresentado na Figura 27.

Figura 27 - Circuito equivalente para análise da malha de tensão



Fonte: Próprio autor.

Através da aplicação da lei de Lei das Correntes de Kirchoff nesse circuito, obtém-se a seguinte equação:

$$i_L = i_c + i_o \quad (45)$$

$$i_L(t) = C \frac{dv_o(t)}{dt} + \frac{v_o(t)}{R_o} \quad (46)$$

Após aplicar a Transformada de Laplace em (46), resulta

$$i_L(s) = C s v_o(s) + \frac{v_o(s)}{R_o} \quad (47)$$

Para correlacionar a tensão de saída com a corrente no indutor, aplica-se álgebra e chega-se em:

$$G_{vi}(s) = \frac{\hat{v}_o(s)}{\hat{i}_L(s)} = \frac{1}{Cs + \frac{1}{R_o}} \quad (48)$$

CAPÍTULO 9 PROJETO DO CONTROLE EM MALHA FECHADA

Aqui serão apresentadas as etapas desenvolvidas no projeto do controle do conversor. Primeiramente, é necessária a validação dos modelos do conversor. Para isso serão utilizados os softwares PSIM® e MATLAB®.

9.1 Validação dos modelos

O Apêndice B – *Script* em MATLAB® para o projeto dos controladores do trabalho apresenta um *script* em MATLAB® que apresenta a resolução numérica dos modelos das plantas no capítulo anterior. Esses modelos são inseridos na simulação do PSIM®. Para o projeto dos controladores foram escolhidos os parâmetros do conversor, a saber: a menor capacitância, a maior indutância e a menor resistência de carga.

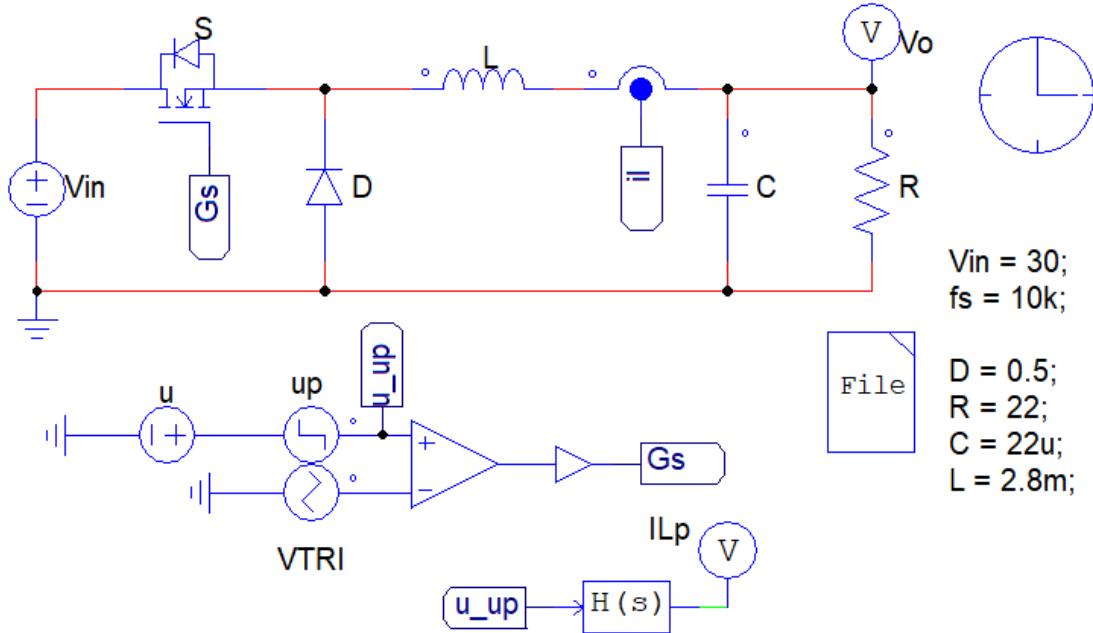
Assim, estas são as funções de transferência resultantes:

$$G_{id} = \frac{5357s + 5,181 \times 10^7}{s^2 + 9671s + 3,799 \times 10^7}$$

$$G_{vi} = \frac{2,128 \times 10^5}{s + 9671}$$

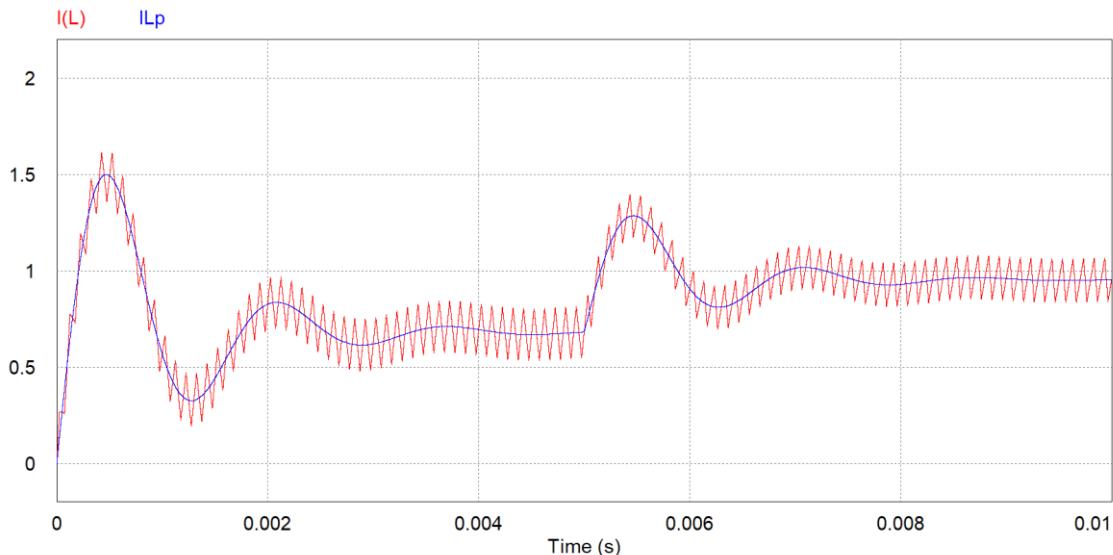
Essas plantas são simuladas no software PSIM®, conforme a Figura 28. As plantas são implementadas dentro das funções de transferência $H(s)$. A Figura 29 mostra o gráfico do modelo de corrente contra o resultado simulado. Pode-se observar os valores são próximos, provando que este modelo é válido. As condições iniciais são nulas e aplica-se um aumento na referência no instante de tempo 5ms.

Figura 28 – Modelo de corrente no PSIM®



Fonte: Próprio autor.

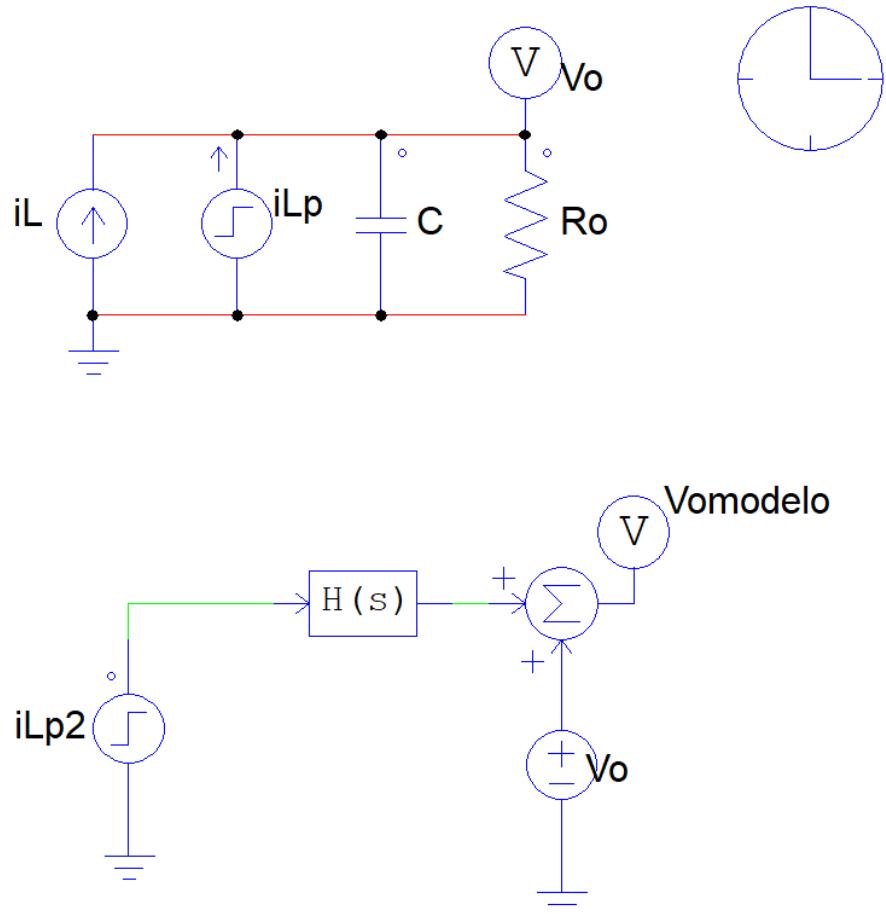
Figura 29 - Validação modelo de corrente



Fonte: Próprio autor.

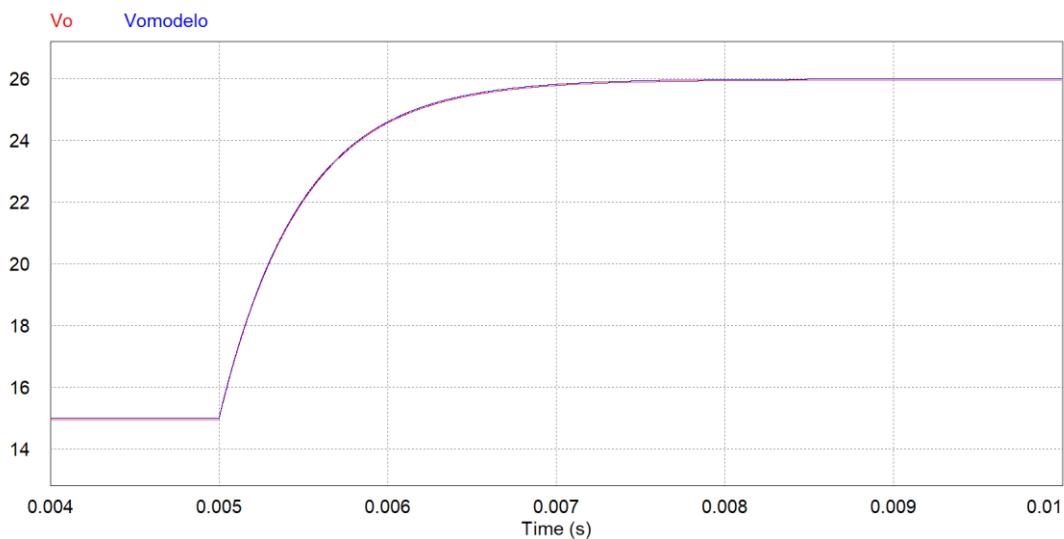
De maneira análoga foi realizada a validação do modelo de tensão do conversor, simulando a planta G_{vi} no PSIM®. Essa planta é validada utilizando o circuito descrito na Figura 27. A Figura 30 mostra o modelo no PSIM® e a Figura 31 apresenta o resultado da validação, foi aplicado um aumento na referência no mesmo instante de tempo 5ms

Figura 30 - Modelo de tensão no PSIM®



Fonte: Próprio autor.

Figura 31 - Validação modelo de tensão



Fonte: Próprio autor.

9.2 Projeto do sistema de controle em malha fechada

Após o modelo validado, é feito o projeto do controle em malha fechada, conforme a Figura 26 mostra, sendo projetados os controladores $C_{iLd}(s)$ e $C_{vi}(s)$. O *software* escolhido para esse projeto é o MATLAB®, utilizando o *script* apresentado no Apêndice B – *Script* em MATLAB® para o projeto dos controladores, após isso será simulado no PSIM® antes da sua implementação no microcontrolador. Ressalta-se que a variável f_{clk} presente no *script* (linha 39) possui dois valores, o de 16×10^6 Hz para o desenvolvimento e validação no PSIM e o de 72×10^6 Hz que é o valor do *clock* do microcontrolador utilizado neste trabalho. Os valores diferentes resultam em controladores diferentes. A ação de controle de corrente age diretamente sobre o sinal PWM, assim ela deve ser ajustada ao tamanho da triangular, no caso da simulação a contagem do sinal ($cont_{max}$) vai até 800 e na *bluepill* 3600. As plantas serão calculadas utilizando os valores de $C = 4,7 \mu F$, $L = 5,6 mH$ e $R = 22 \Omega$, conforme o *script* utilizado.

Para adicionar o ganho do modulador, multiplica-se a planta por $\frac{1}{cont_{max}}$, portanto:

$$G_{iLd}M = G_{iLd} \times \frac{1}{cont_{max}} \quad (49)$$

Após isso foi adicionado o atraso, utilizando a aproximação de Padé, que utiliza o valor do período de amostragem T_a . Para $G_{iLd}M$ o valor é de $\frac{1}{f_s}$ e para G_{vi} é $\frac{1}{10f_s}$. A aproximação de Padé é dada por:

$$G_{atraso} = \frac{-s + \frac{2}{T_a}}{s + \frac{2}{T_a}} \quad (50)$$

As plantas resultantes, para a frequência de *clock* de 72×10^6 Hz após adicionar o atraso, são:

$$G_{iLd}Ma = \frac{-1,488s^2 + 4,513 \times 10^4 s + 5,757 \times 10^8}{s^3 + 4,967 \times 10^4 s^2 + 4,248 \times 10^8 s + 1,52 \times 10^{12}}$$

$$G_{via} = \frac{-2,128 \times 10^5 s + 8,511 \times 10^9}{s^2 + 4,967 \times 10^4 s + 3,868 \times 10^8}$$

Para os controladores, serão projetados inicialmente 2 controladores proporcionais integrais (PIs) com um zero, a posição do zero desses controladores será dada por $\frac{2\pi f_c}{20}$ sendo f_c a frequência de cruzamento desejada para o controlador.

$$C_{pi} = \frac{s + \frac{2\pi f_c}{20}}{s} \quad (51)$$

As frequências de cruzamento das duas malhas foram decididas com referência na frequência de chaveamento, colocando a malha de corrente uma década abaixo da frequência de amostragem (chaveamento) e a malha de tensão uma década abaixo da frequência de corte da malha de corrente:

$$f_{c_{id}} = \frac{f_s}{10} \quad f_{c_{vi}} = \frac{f_{c_{id}}}{10} \quad (52)$$

Assim, sabendo que a frequência de chaveamento do conversor é de 10 kHz, as frequências de cruzamento ficam $f_{c_{id}} = 1000$ Hz e $f_{c_{vi}} = 100$ Hz, os controladores proporcionais-integrais das malhas de corrente e tensão se apresentam como:

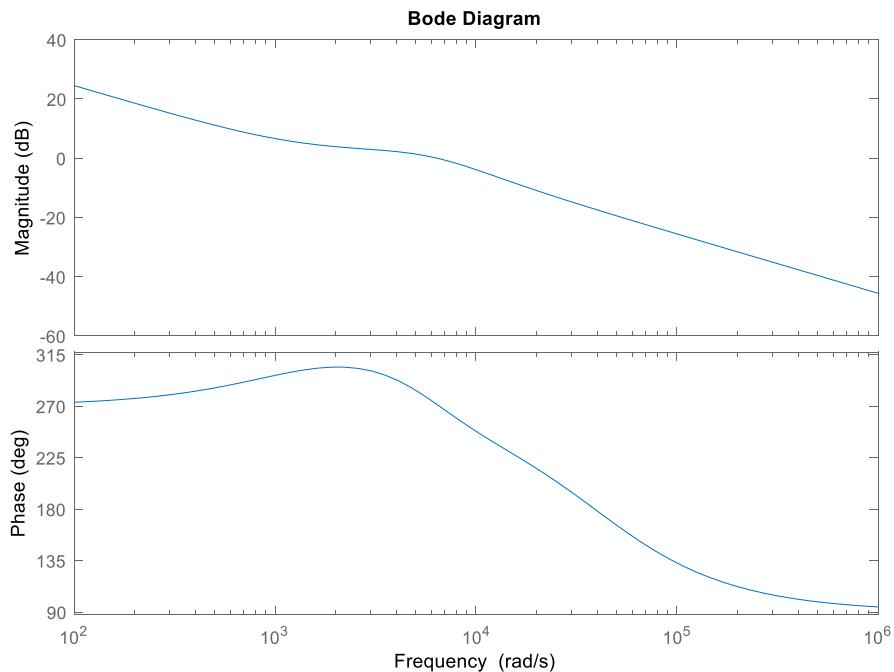
$$C_{pii} = \frac{s + 1257}{s} \text{ e } C_{piv} = \frac{s + 125,7}{s}$$

Após isso, foi projetado o ganho utilizando a ferramenta *sisotool* do MATLAB®. Como as margens de fase dos dois controladores ficaram acima de 45° não foi necessário a inclusão de um compensador PD. Os controladores finais projetados são dados por:

$$C_{pii2} = \frac{3531s + 4,437 \times 10^6}{s} \text{ e } C_{piv2} = \frac{0,04468s + 5,615}{s}$$

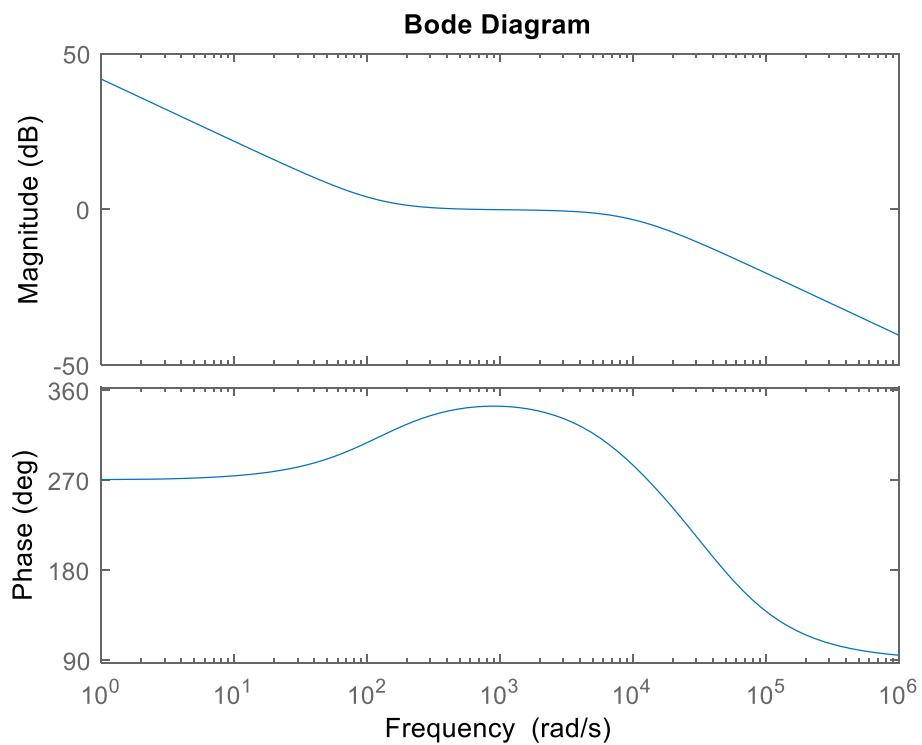
As Figura 32 e Figura 33 mostram o diagrama de Bode das malhas de corrente e tensão projetadas para o microcontrolador.

Figura 32 - Diagrama de Bode da malha de corrente em malha aberta compensada



Fonte: Próprio autor.

Figura 33 - Diagrama de Bode da malha de tensão em malha aberta compensada



Fonte: Próprio autor.

Por último é feita a discretização desses controladores através do método Tustin para que se possa fazer a extração dos coeficientes que serão utilizados na equação de diferenças discretas. Essa parte do projeto pode ser observada detalhadamente no Apêndice B – *Script* em MATLAB® para o projeto dos controladores.

A implementação no microcontrolador faz necessária programar a equação de diferenças discretas. Esta equação relaciona a ação de controle do controlador com o erro medido no período de amostragem anterior.

Como ambas as malhas apresentam um controlador PI de mesma ordem, a equação de diferenças discretas possui o mesmo formato:

$$\frac{u(z)}{e(z)} = \frac{a_1 z + a_2}{b_1 z + b_2} \quad (53)$$

Para implementar o atraso na equação, deve-se multiplicar por $\frac{1}{z}$, obtendo:

$$\frac{u(z)}{e(z)} = \frac{a_1 z + a_2}{b_1 z^2 + b_2 z} \quad (54)$$

$$(b_1 z^2 u + b_2 z u = a_1 z e + a_2 e) z^{-2} \quad (55)$$

$$b_1 u + b_2 z^{-1} u = a_1 z^{-1} e + a_2 z^{-2} e \quad (56)$$

Nos casos dos dois controladores, $b_1 = 1$ e $b_2 = -1$. Assim:

$$u(k) = u(k-1) + a_1 e(k-1) + a_2 e(k-2) \quad (57)$$

Agora precisa-se apenas extrair os valores de a_1 e a_2 de cada controlador do *script* em MATLAB®. Para a simulação no PSIM® e implementação no microcontrolador foram utilizados os valores apresentados na Tabela 7.

Tabela 7 - Valores das constantes da equação de diferenças discretas

		PSIM®	STM32
Malha corrente	a_1	313.569362420081	3.64182649500560e+003
	a_2	-303.870637579919	-3.41997350499440e+003
Malha tensão	a_1	48.2156570163295e-003	44.8243789261330e-003
	a_2	-46.7243429836705e-003	-44.5436210738670e-003

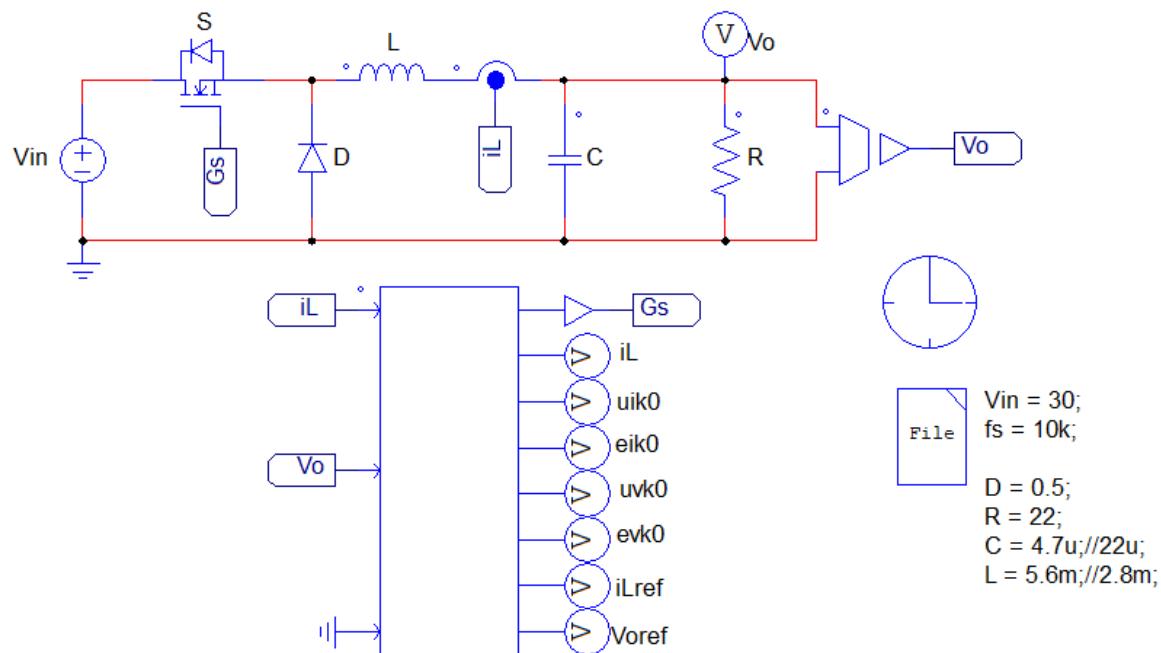
Fonte: Próprio autor.

9.3 Implementação das malhas de controle

9.3.1 Implementação no PSIM

O controle deste projeto foi implementado primeiramente no PSIM® utilizando o bloco C. A Figura 34 mostra como foi montado o circuito no *software* e o código implementado dentro do bloco pode ser visto no Apêndice C - *Script* em C para a simulação no PSIM® deste trabalho.

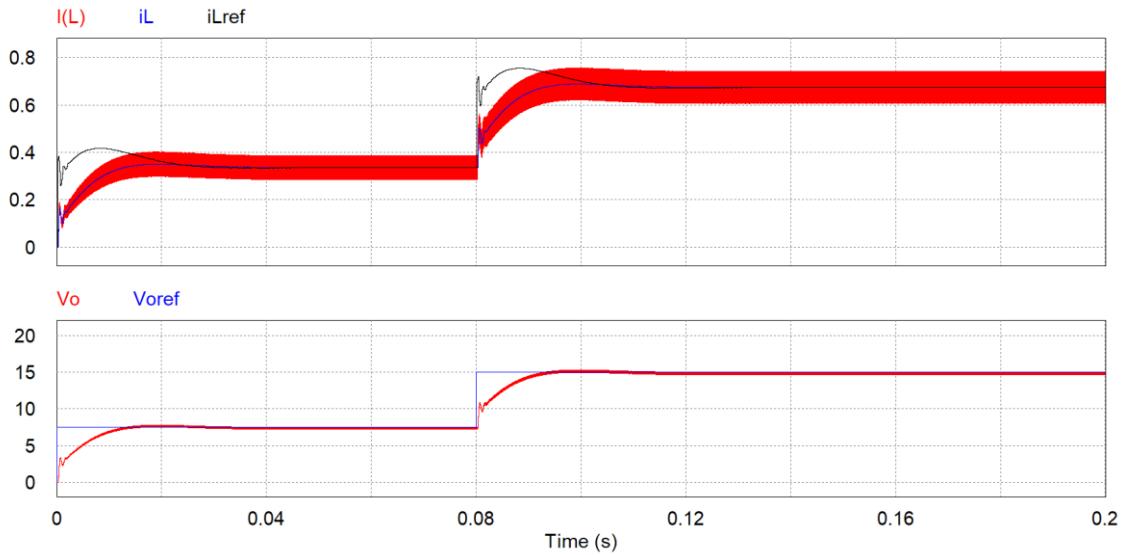
Figura 34 - Implementação das malhas de controle no PSIM®



Fonte: Próprio autor.

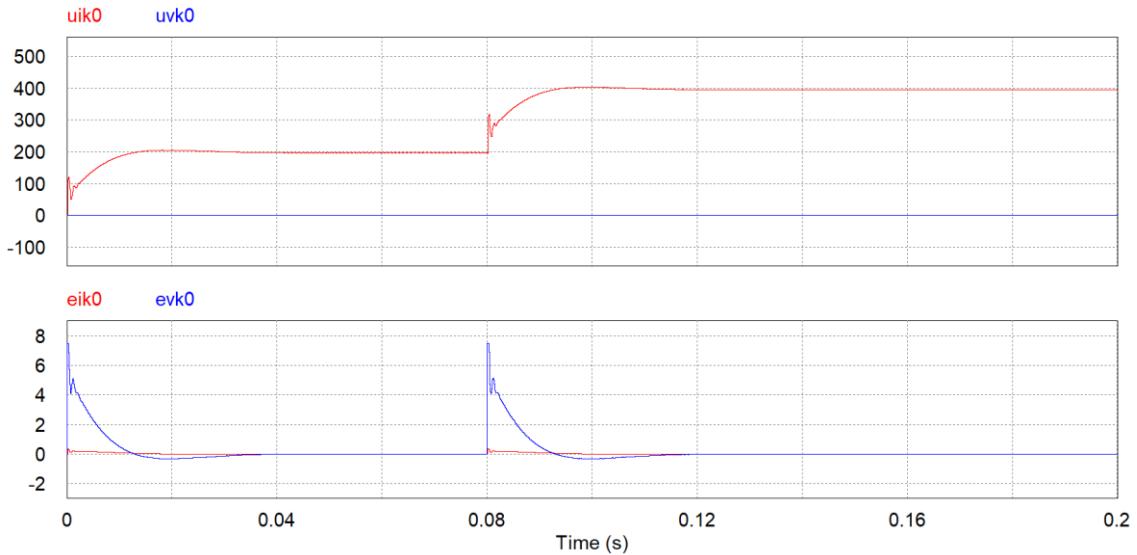
A Figura 35 apresenta os gráficos das variáveis a serem controladas e suas referências, enquanto a Figura 36 mostra as ações de controle e erros monitorados na simulação para a validação da estratégia proposta.

Figura 35 – Simulação do conversor operando em malha fechada no PSIM®



Fonte: Próprio autor

Figura 36 - Ações de controle e erros



Fonte: Próprio autor.

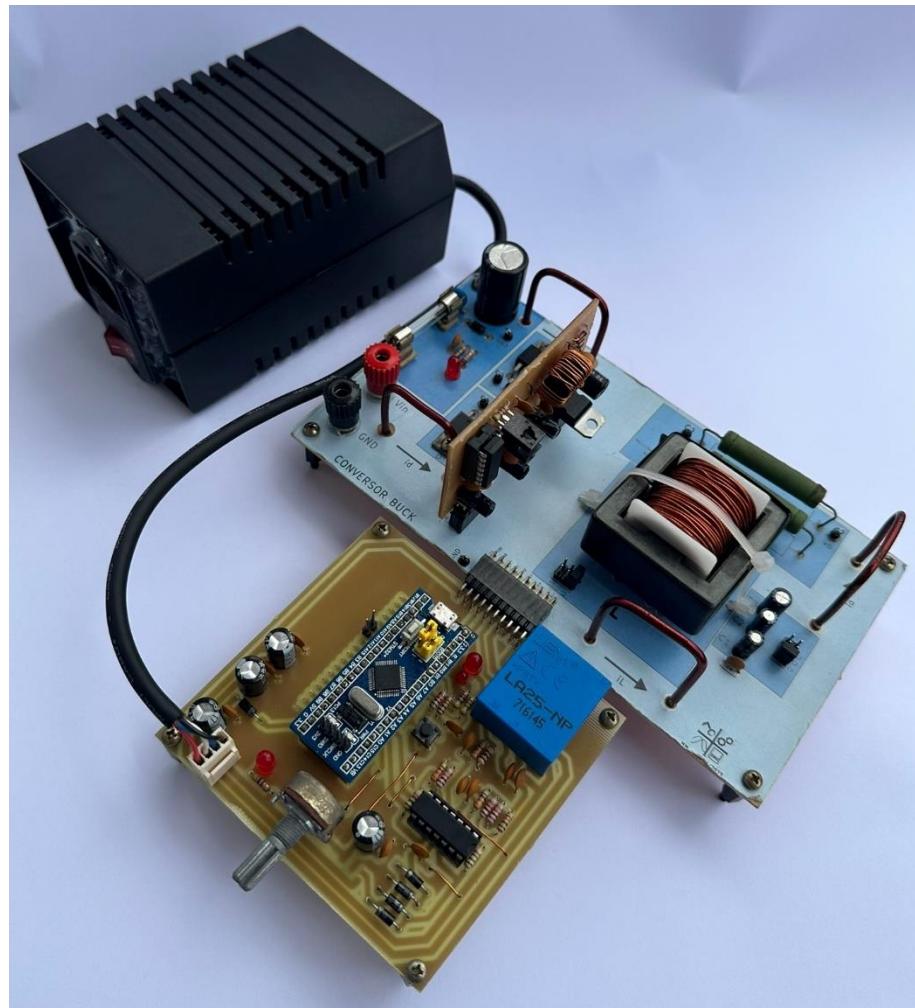
9.3.2 Implementação no microcontrolador

A implementação do controle em malha fechada no microcontrolador foi realizada utilizando o *software* CubeIDE. O código final implementado pode ser visto no Apêndice D – *Script* em C implementado no Microcontrolador. Os resultados serão apresentados no próximo capítulo.

CAPÍTULO 10 RESULTADOS EXPERIMENTAIS

Esta seção apresenta os resultados do *kit* didático desenvolvido. Os resultados experimentais serão separados em resultados das instrumentações de corrente e tensão, resultados em malha aberta e resultados em malha fechada. A Figura 37 mostra o *kit* didático completo montado, no qual serão executados os experimentos. Os experimentos propostos para o aprendizado dos alunos são os mesmos experimentos deste capítulo, seja com o conversor funcionando em malha aberta ou malha fechada.

Figura 37 - *Kit* didático completo montado

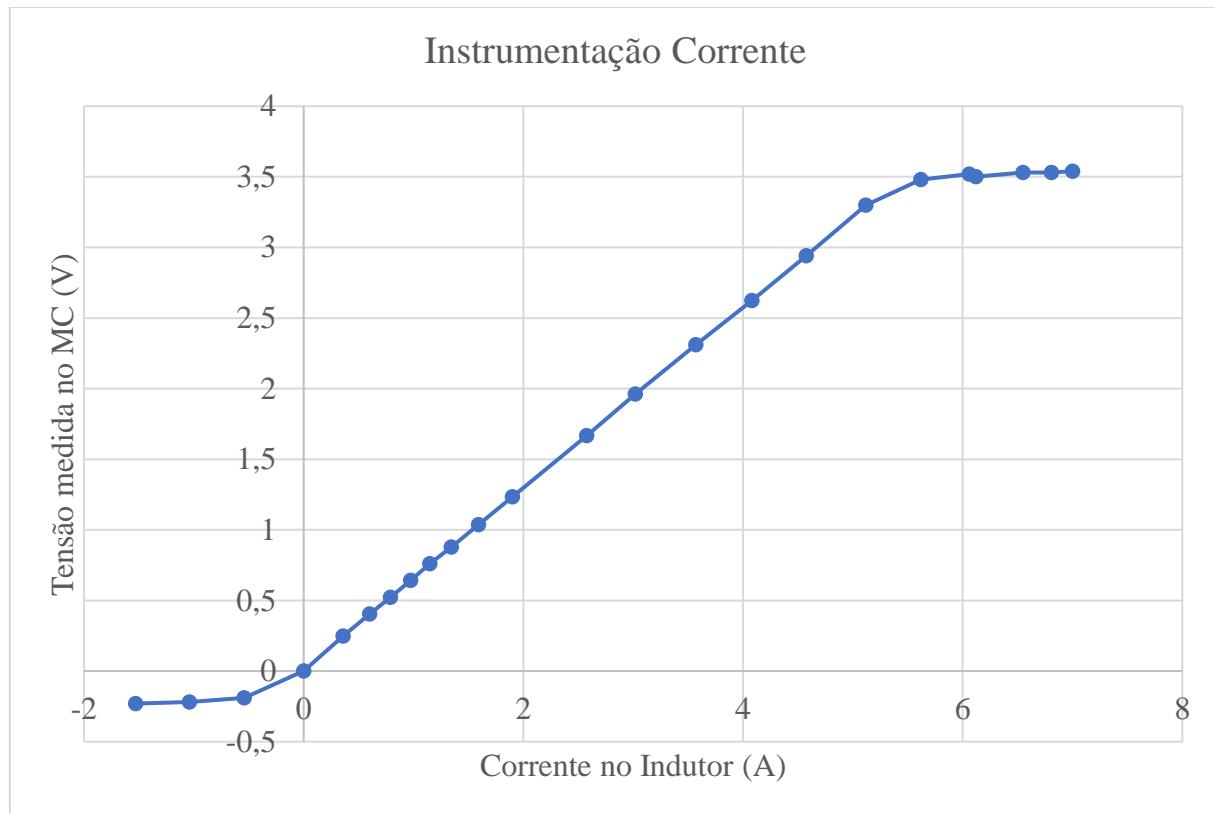


Fonte: Próprio autor

10.1 Resultados das instrumentações de corrente e tensão

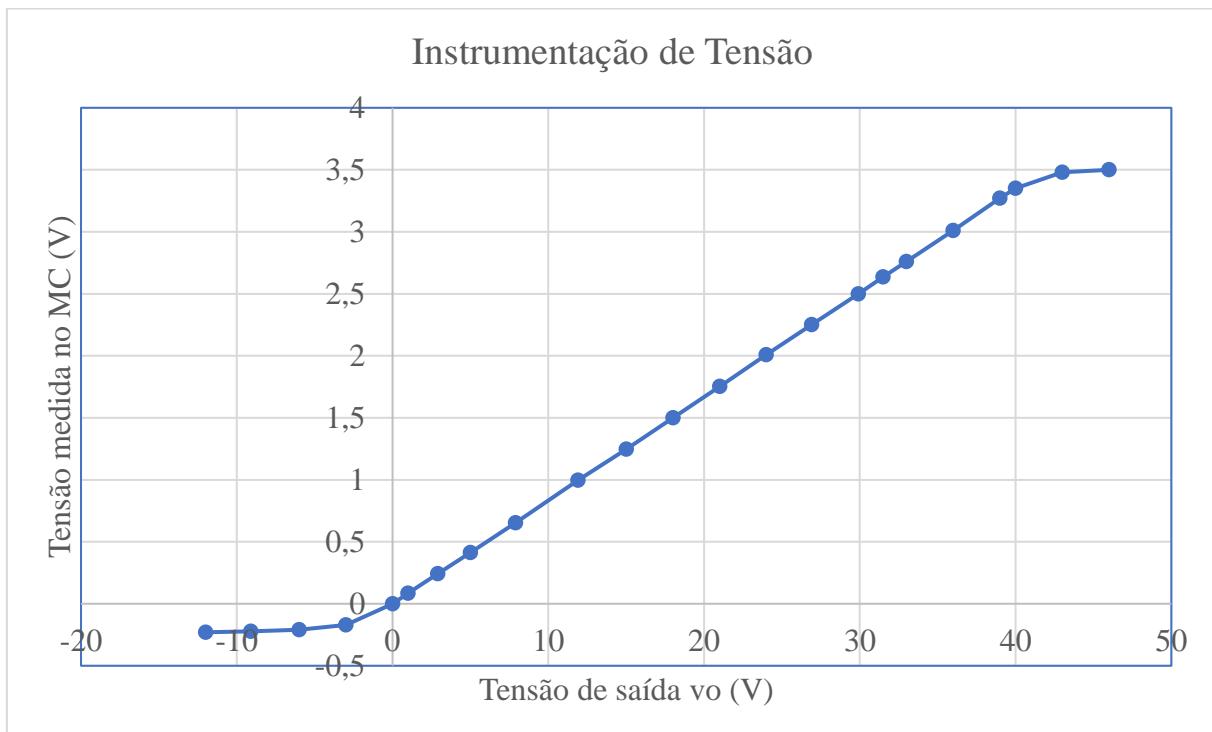
Esta seção apresentará os resultados das leituras de tensão no pino de entrada do microcontrolador (MC), para verificar se elas atenderam os requisitos de projeto mencionados na seção 7.1. É um passo muito importante para o funcionamento do projeto, visto que caso tenham sido mal dimensionados ou não estiver operando corretamente, o controle em malha fechada não funcionará. Lembrando que os limites de leitura para corrente calculados são 0 a 5 amperes, tendo a leitura de 3,3V na entrada do MC quando a corrente possui o valor de 5 A e os limites de tensão devem ser de 0 a 40 volts, tendo a leitura de 3,3 V na entrada do MC quando a tensão de saída é 40 V. Os resultados das leituras de corrente e tensão podem ser observados nas Figura 38 e Figura 39, respectivamente.

Figura 38 - Leituras da instrumentação de corrente



Fonte: Próprio autor.

Figura 39 - Leituras da instrumentação de tensão



Fonte: Próprio autor.

Na instrumentação de corrente, o crescimento do sinal no pino do MC é linear na região entre 0 V e 3,3 V, sendo que a leitura de 3,3 V corresponde à corrente de 5,12 A. Ao atingir este valor, os diodos de segurança entram em ação, não deixando a tensão neste pino passar de 3,54 V quando a corrente no circuito é 7 A. Quando aplicada uma corrente reversa de -1,53 A, a leitura chega a -0,23 V, nível seguro para o microcontrolador.

De maneira análoga, a instrumentação de tensão opera de maneira linear na faixa de 0 V a 40 V, sendo que a leitura da tensão de saída de 40 V envia 3,34 V para o pino do MC. Acima deste valor os diodos de segurança entram em operação, mantendo a tensão no MC em 3,52 V para uma tensão de saída de 50 V. Quando aplicada uma tensão reversa, os diodos mantêm a tensão em -0,23 V no MC para uma leitura de -12 V na tensão de saída, o que representa um nível seguro.

10.2 Resultados em malha aberta

Os resultados do conversor operando em malha aberta. Esses resultados são os mais importantes para a disciplina de Eletrônica de Potência pois estão relacionados apenas com a

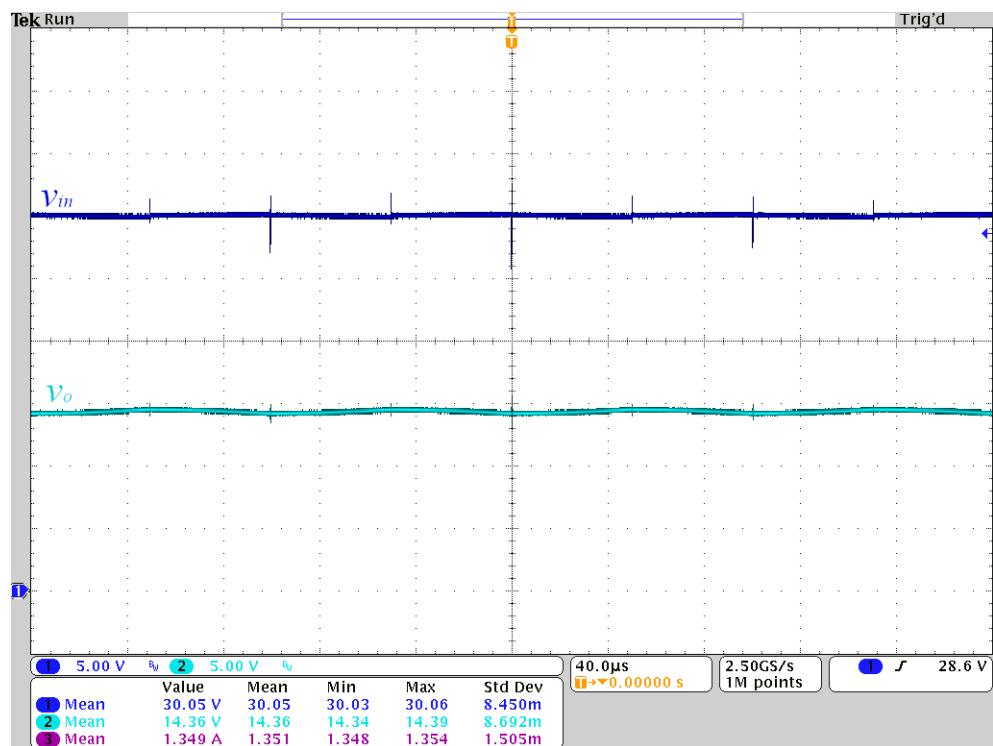
topologia do conversor em si. Serão apresentados os resultados de 5 leituras em potência nominal com sinal PWM em 50%, sendo eles:

1. Tensão de entrada e tensão de saída;
2. Corrente no indutor e seu *ripple* para os 3 valores de indutância;
3. *Ripples* da tensão de saída para os 3 valores de capacitância;
4. Tensão e corrente na chave;
5. Tensão e corrente no diodo.

10.2.1 Tensão de entrada e saída

Este resultado apresenta a tensão de saída do conversor para as condições nominais de tensão de entrada 30 V e razão-cíclica em 50%. A Figura 40 apresenta as leituras realizadas em laboratório. A medição do sinal de entrada foi realizada no ponto de teste V_{in2} e do sinal de saída em V_o . As medições foram realizadas com a carga nominal $R_o = 11 \Omega$, maior capacidade $C_s = 22 \mu F$ e menor indutância $L_i = 2,8 mH$.

Figura 40 – Tensão de entrada e saída do conversor



Fonte: Próprio autor.

Este conversor foi projetado para, operando em condições nominais, a relação entre a tensão de saída e entrada seja igual à razão-cíclica aplicada. Neste caso, como a razão-cíclica é 50% a relação se apresentaria como:

$$\frac{V_o}{V_{in}} = 0,5 \quad (58)$$

Como o projeto foi realizado considerando componentes ideais, o resultado com componentes reais não se apresenta igual ao teórico. No caso deste conversor, para as condições nominais, têm-se a tensão de saída $V_o = 14,36 V$.

$$\frac{V_o}{V_{in}} = \frac{14,36}{30,05} = 0,4779 \quad (59)$$

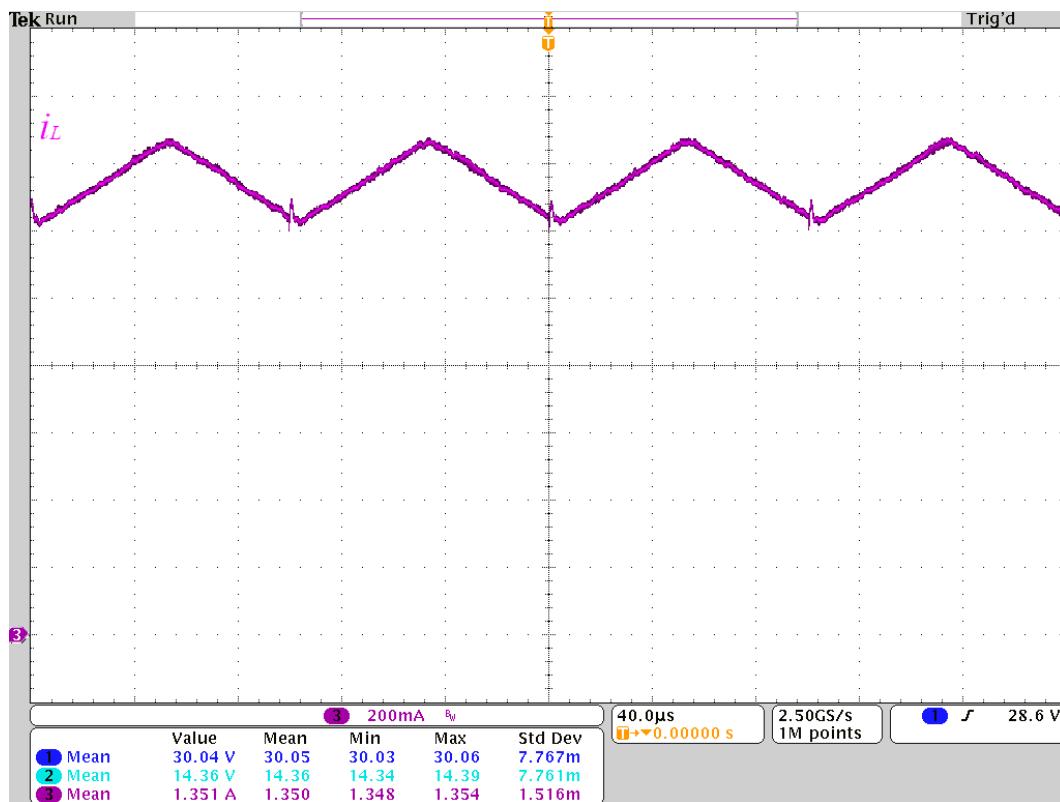
O erro do projeto, relacionando os componentes reais pelos ideais fica em:

$$erro = \left(1 - \frac{0,4779}{0,5} \right) \times 100 = 4,42\% \quad (60)$$

10.2.2 Corrente no indutor

Este resultado apresenta a corrente média no indutor e os *ripples* para cada uma das opções de indutância do projeto. As leituras de corrente são realizadas no ponto de teste de corrente i_L . Como o experimento anterior, ele é realizado com tensão de entrada nominal, razão cíclica de 50%, maior capacidade $C_3 = 22 \mu F$ e carga nominal $R_o = 11 \Omega$. A Figura 41 apresenta o valor instantâneo da corrente no indutor, tendo seu valor médio medido através do osciloscópio.

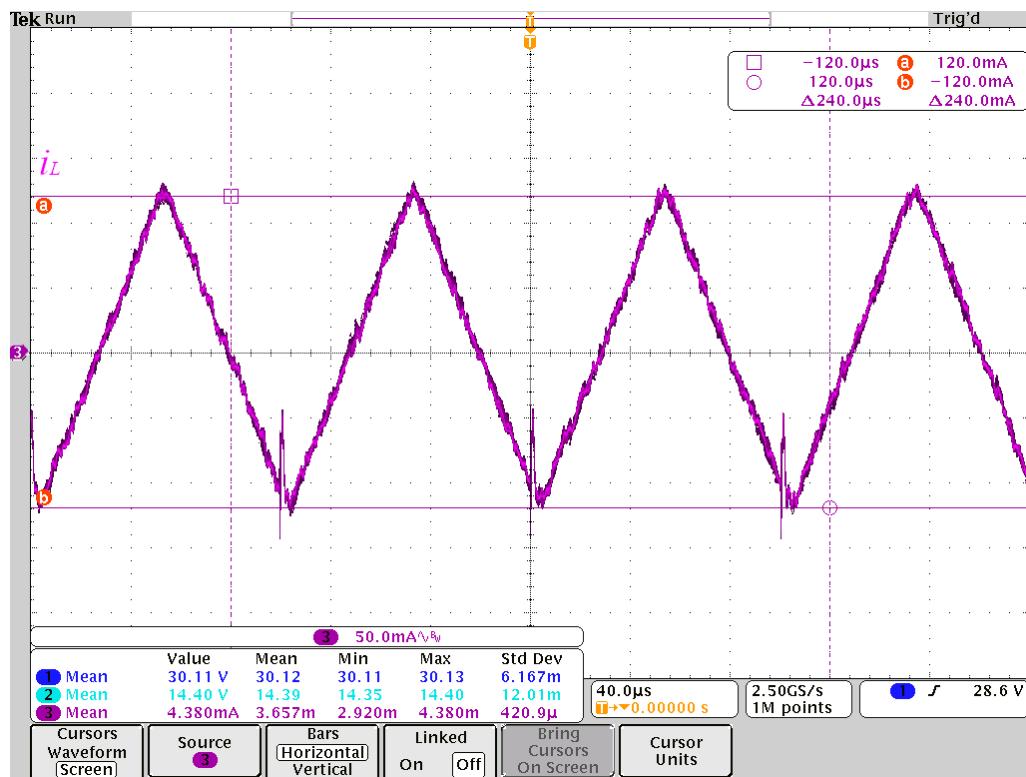
Figura 41 – Corrente média no indutor



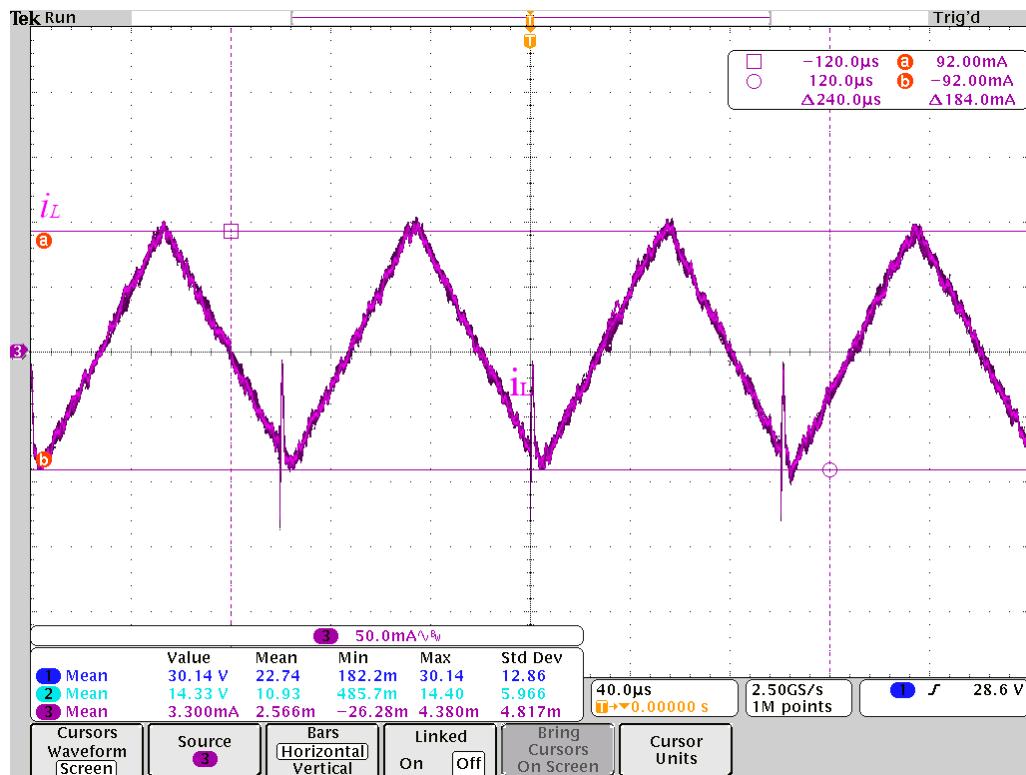
Fonte: Próprio autor.

O comportamento do conversor implementado se apresenta exatamente como a Figura 8 previa, tendo o pico do triângulo no meio da forma de onda devido a utilização da razão-cíclica de 0,5. A corrente ideal esperada na saída seria de 1,36 A, valor muito próximo do medido.

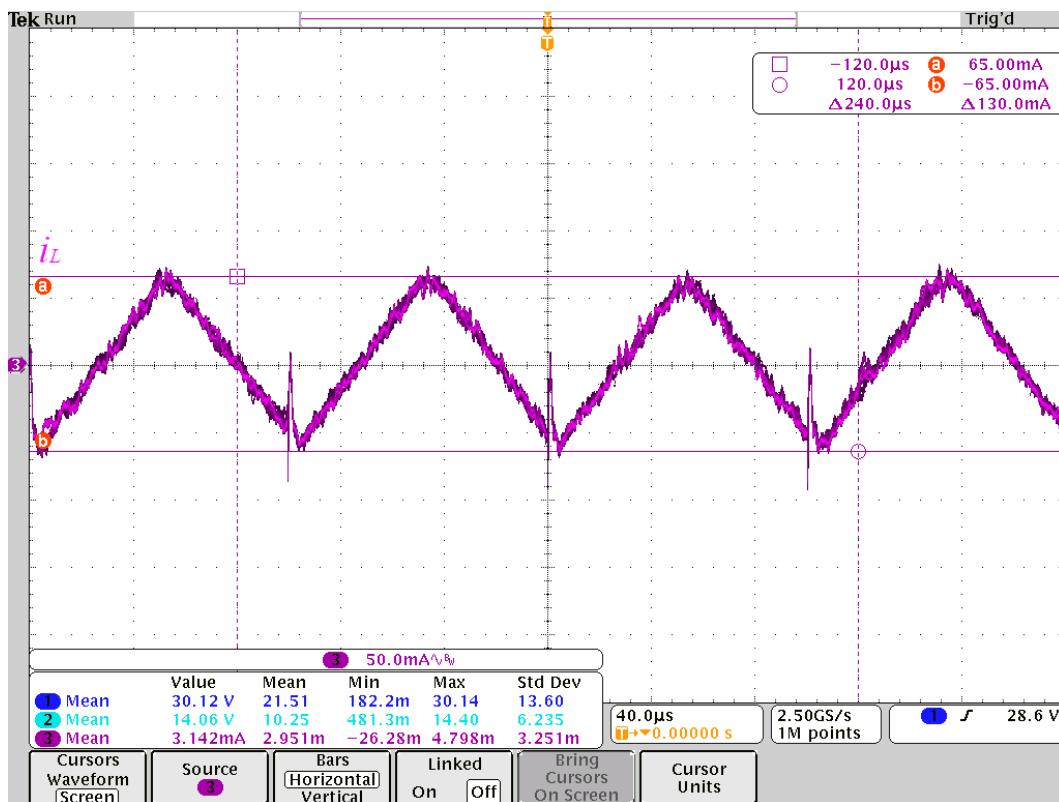
Para os experimentos realizados pelos alunos, valores diferentes de *ripple* são importantes. As Figura 42, Figura 43 e Figura 44 apresentam, respectivamente, os *ripples* dos indutores $L_1 = 2,8\text{ mH}$, $L_2 = 3,7\text{ mH}$ e $L_3 = 5,6\text{ mH}$ do conversor.

Figura 42 - Ripple indutor L_1 

Fonte: Próprio autor.

Figura 43 - Ripple indutor L_2 

Fonte: Próprio autor.

Figura 44 - Ripple indutor L_3 

Fonte: Próprio autor.

Os valores de *ripple* esperados diminuem conforme se aumenta a indutância, então para o indutor L_1 têm-se os maiores valores e o indutor L_3 apresenta os menores. O cálculo do *ripple* percentual é dado por:

$$\text{ripple}(\%) = \frac{\Delta i_L}{I_L} \times 100 \quad (61)$$

Utilizando a Equação (61) para o cálculo e o valor da corrente média apresentada na Figura 41, chega-se nos *ripples* para os indutores do conversor apresentados na Tabela 8.

Tabela 8 – Ripple dos indutores do conversor

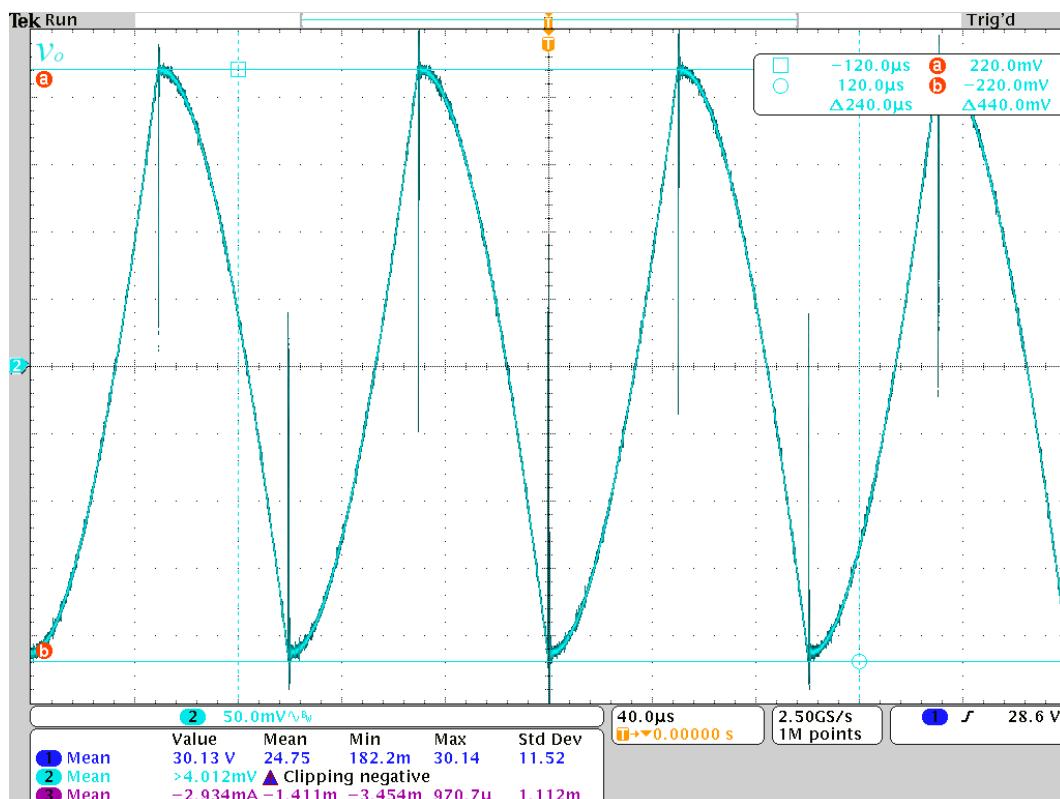
Indutor	Ripple	Ripple projetado
L_1	17,76%	20%
L_2	13,62%	15%
L_3	9,62%	10%

Fonte: Próprio autor.

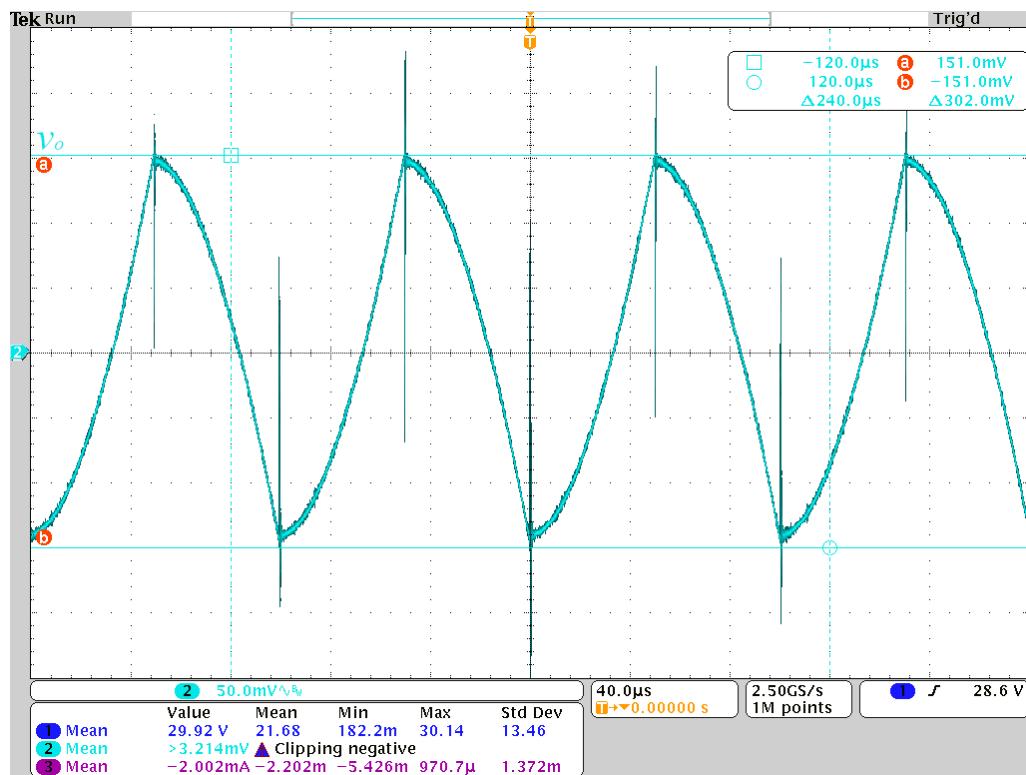
10.2.3 Ripples da tensão de saída

Este resultado apresenta os *ripples* para cada uma das opções de capacitância do projeto. As leituras de tensão são realizadas no ponto de teste V_o . Como o experimento anterior, ele é realizado com tensão de entrada nominal, razão-cíclica de 50% e carga nominal, entretanto, o valor de indutância é o maior (L_3). O valor médio da tensão de saída é apresentado na Figura 40 e os *ripples* serão calculados com esse valor. As Figura 45, Figura 46 e Figura 47 apresentam os *ripples* de tensão medidos para os capacitores $C_1 = 4,7 \mu F$, $C_2 = 10 \mu F$ e $C_3 = 22 \mu F$, respectivamente.

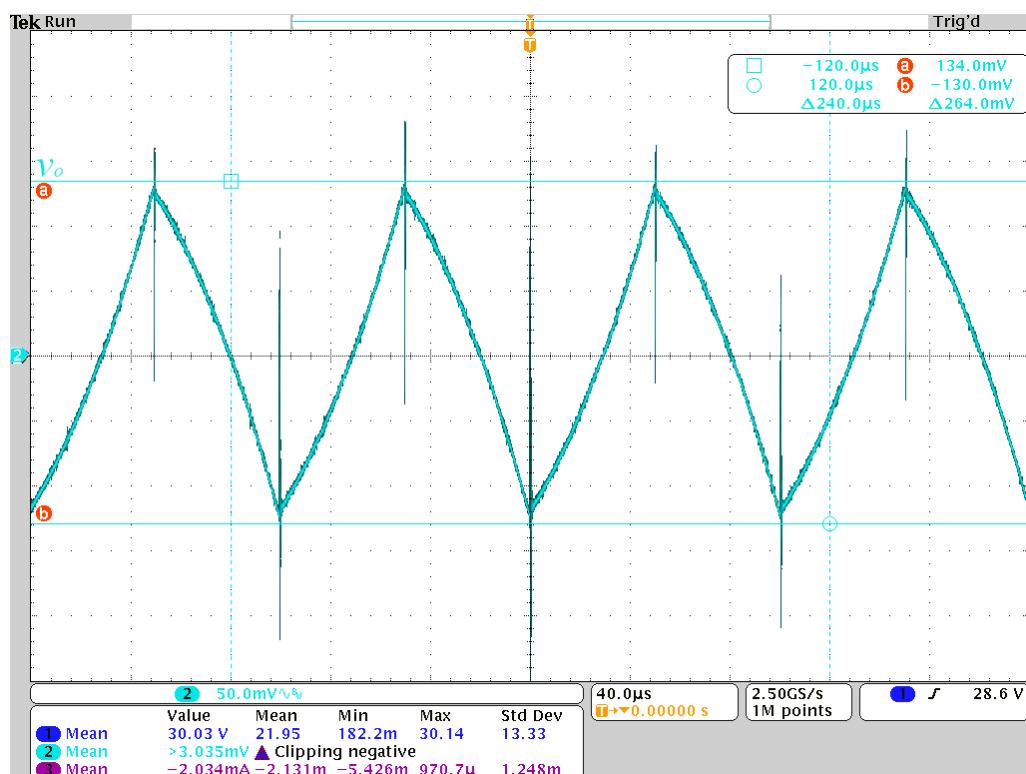
Figura 45 - Ripple de tensão C_1



Fonte: Próprio autor.

Figura 46 - Ripple de tensão C_2 

Fonte: Próprio autor.

Figura 47 - Ripple de tensão C_3 

Fonte: Próprio autor.

Os valores de *ripple*, igualmente ao caso dos indutores, diminuem conforme se aumenta a capacidade. O cálculo do *ripple* percentual é dado por:

$$\text{ripple}(\%) = \frac{\Delta V_o}{V_o} \times 100 \quad (62)$$

Utilizando a Equação (62) para o cálculo e o valor da tensão média apresentada na Figura 40, chega-se nos valores dos *ripples* para os capacitores do conversor apresentados na Tabela 9.

Tabela 9 - *Ripple* dos capacitores do conversor

Capacitor	<i>Ripple</i>	<i>Ripple</i> projetado
C_1	3,06%	4,75%
C_2	2,1%	1,70%
C_3	1,84%	0,50%

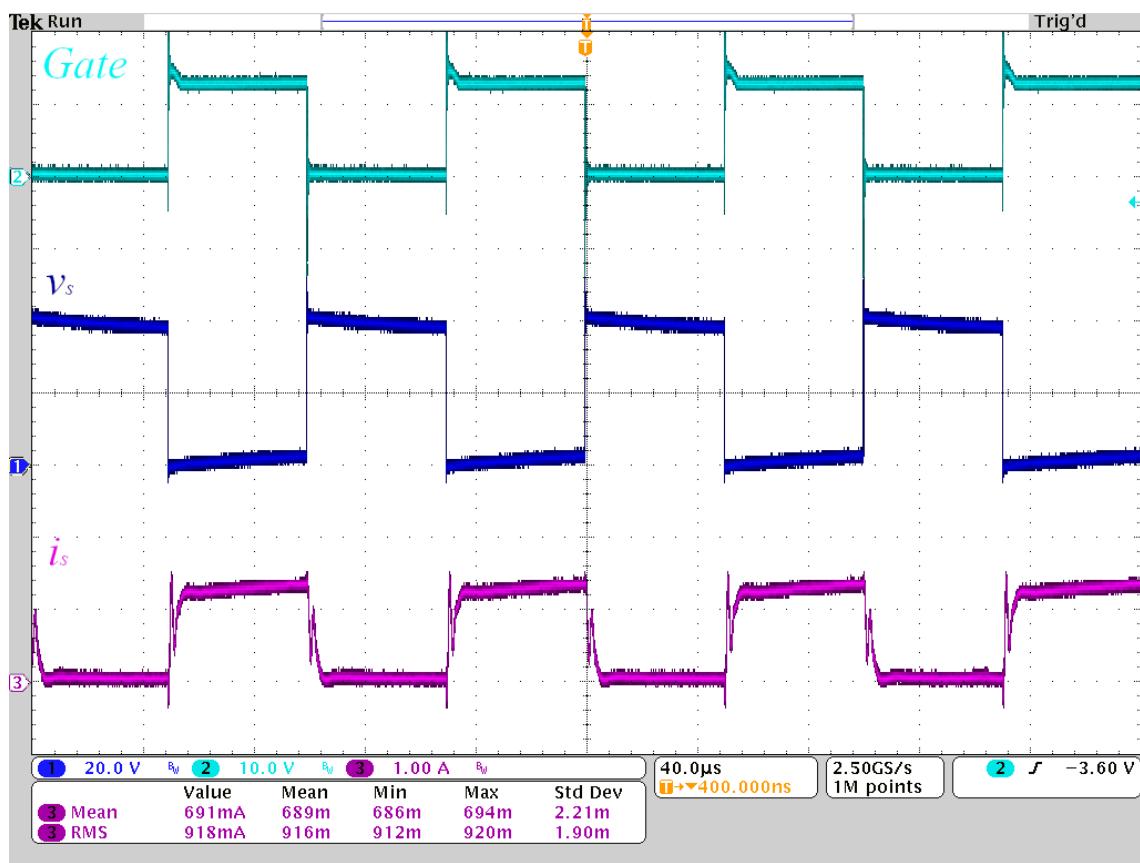
Fonte: Próprio autor.

Um grande fator para o valor do *ripple* real possuir um erro maior na tensão de saída se dá que eles foram calculados com a tensão de saída em 15 V, enquanto as medições do conversor deram a tensão de saída como 14,36 V, aumentando a diferença. Além disso, foram desconsideradas na análise a resistência série equivalente (SER) dos capacitores eletrolíticos empregados.

10.2.4 Tensão e corrente sobre a chave

Esse é o resultado das formas de onda de tensão e corrente na chave. Como já mencionado na parte teórica, é esperado que, quando o sinal de *gate* for positivo (durante o intervalo T_{on}), a chave entre em condução levando a tensão sobre ela a zero e permitindo a passagem de corrente. A corrente neste componente equivale à corrente no indutor no período T_{on} . Este comportamento pode ser visto claramente na Figura 48, onde o canal 2 do osciloscópio representa o sinal de *gate*, o canal 1 a tensão sobre a chave e o canal 3 representa a corrente sobre a chave.

Figura 48 - Formas de onda de tensão e corrente da chave



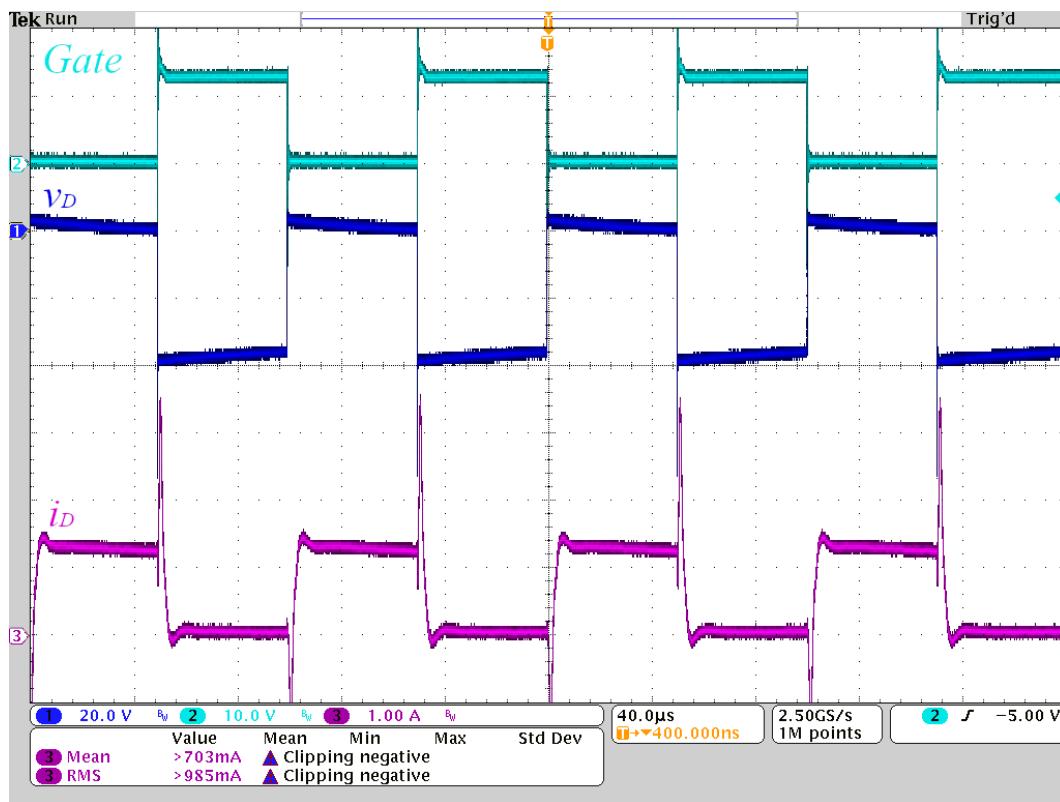
Fonte: Próprio autor.

Observa-se que quando o sinal de *gate* é positivo, a chave entra em condução e a tensão sobre ela vai a zero, nesta etapa a corrente que passa pela chave é igual a i_L . Quando o sinal de *gate* é zero, a chave não conduz, a tensão sobre ela é igual V_{in} e a corrente é zero.

10.2.5 Tensão e corrente sobre o diodo

Este é o resultado que apresenta as formas de onda de tensão e corrente no diodo. Como o diodo opera de maneira complementar à chave, ele entra em condução quando o sinal de *gate* é zero (durante o período de tempo T_{off}). Durante o período T_{off} a tensão sobre o diodo é zero e a corrente equivale à corrente do indutor no mesmo período. A Figura 49 apresenta este comportamento. O canal 2 do osciloscópio representa o sinal de *gate*, o canal 1 a tensão sobre o diodo e o canal 3 representa a corrente sobre o diodo.

Figura 49 – Formas de onda de tensão e corrente do diodo



Fonte: Próprio autor.

De maneira complementar à chave, o diodo entra em condução quando o sinal de *gate* é zero. Nesta etapa a queda de tensão sobre ele é zero e a corrente sobre ele é igual a i_L . Quando o sinal de *gate* é positivo, o diodo está reversamente polarizado com uma tensão igual a $-V_{in}$, assim a corrente passando por ele é zero.

10.3 Resultados em malha fechada

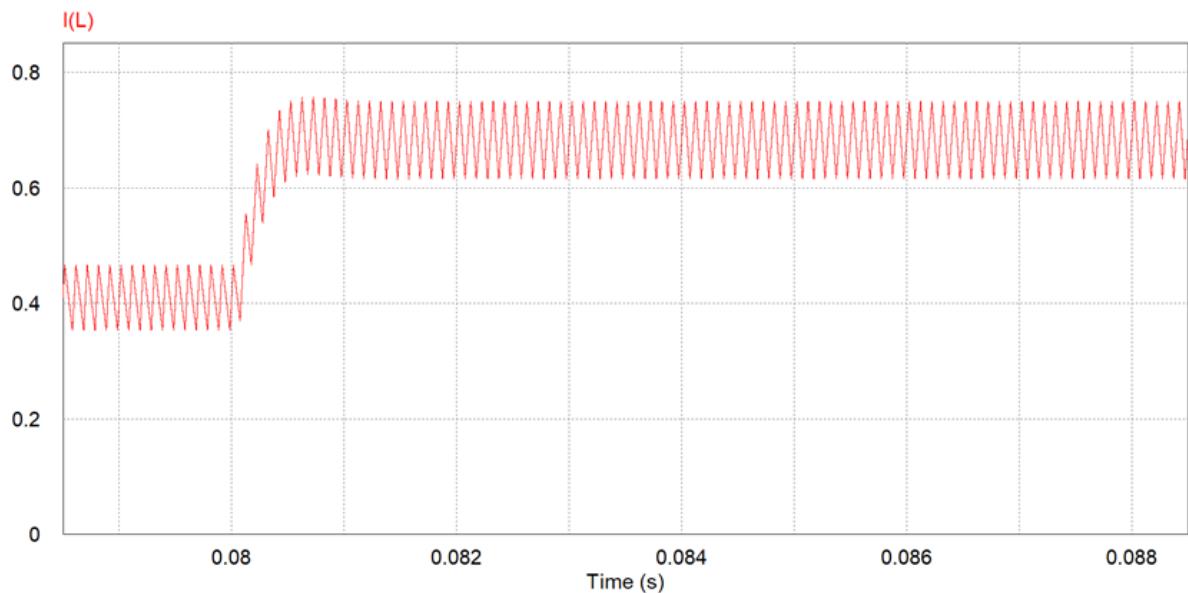
Os resultados do conversor operando em malha fechada, ou seja, com o microcontrolador atuando sobre a razão-cíclica, são os mais importantes para as disciplinas de Controle de Sistemas, pois estão relacionados apenas com a programação da ação de controle sobre o conversor. Serão apresentados os resultados de 4 experimentos, todos eles utilizando os valores de $C = 4,7 \mu F$, $L = 5,6 mH$ e $R = 22 \Omega$:

1. Degrau na razão-cíclica;
2. Degrau na referência de corrente em malha fechada;
3. Degrau na referência de tensão em malha fechada;
4. Degrau de carga em malha fechada.

10.3.1 Degrau na razão-cíclica

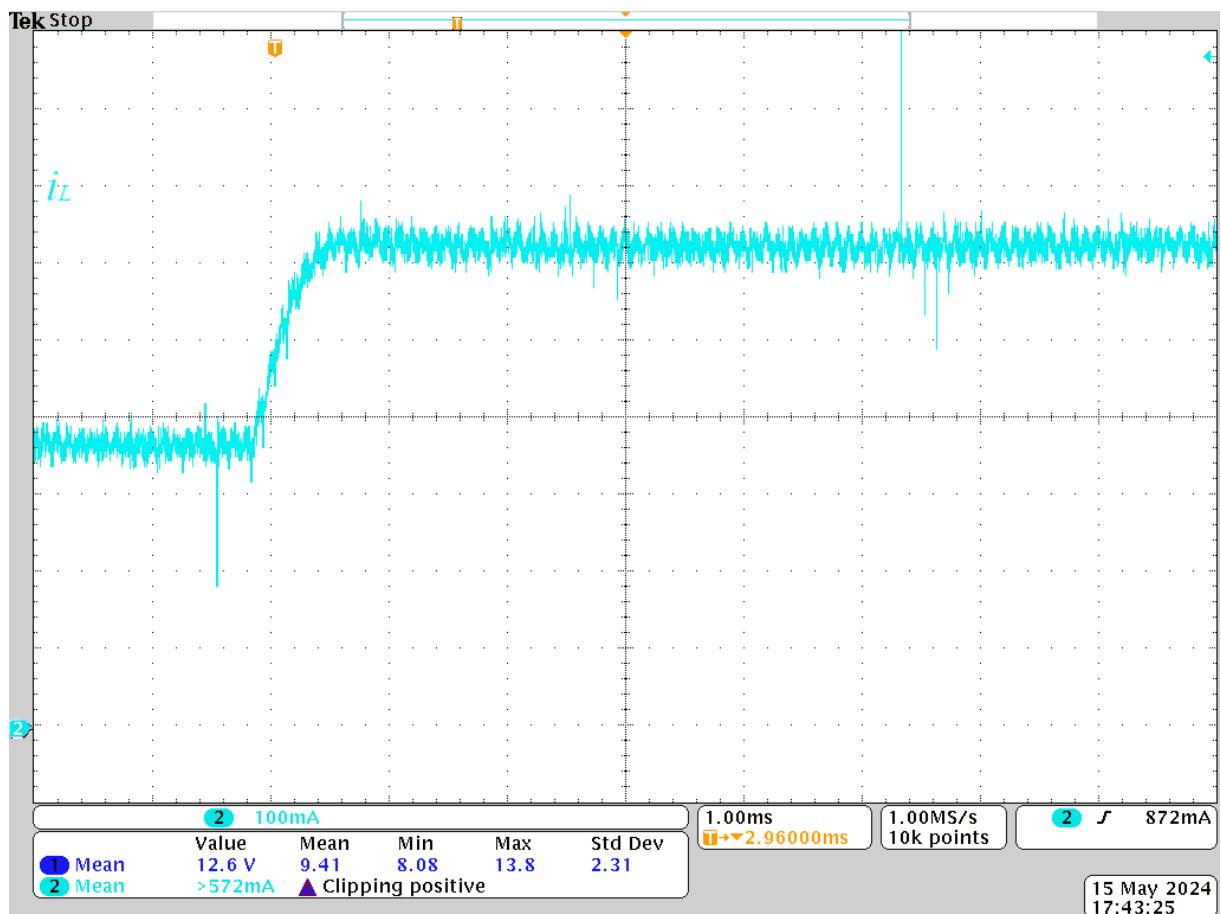
Esse experimento serve para a validação do comportamento da planta. Este experimento consistirá de colocar a razão-cíclica no valor de referência de 0,3 e, após acionado o botão presente na placa de instrumentação, será aplicado um degrau na razão-cíclica, elevando o seu valor para 0,5. A Figura 50 mostra o resultado simulado deste degrau. O resultado experimental, observando a corrente se elevando de um valor inicial aproximado de 0,360 A para o de aproximadamente 0,620 A, pode ser visto na Figura 51.

Figura 50 - Simulação de um degrau na razão-cíclica



Fonte: Próprio autor.

Figura 51 - Degrau na razão-cíclica no conversor

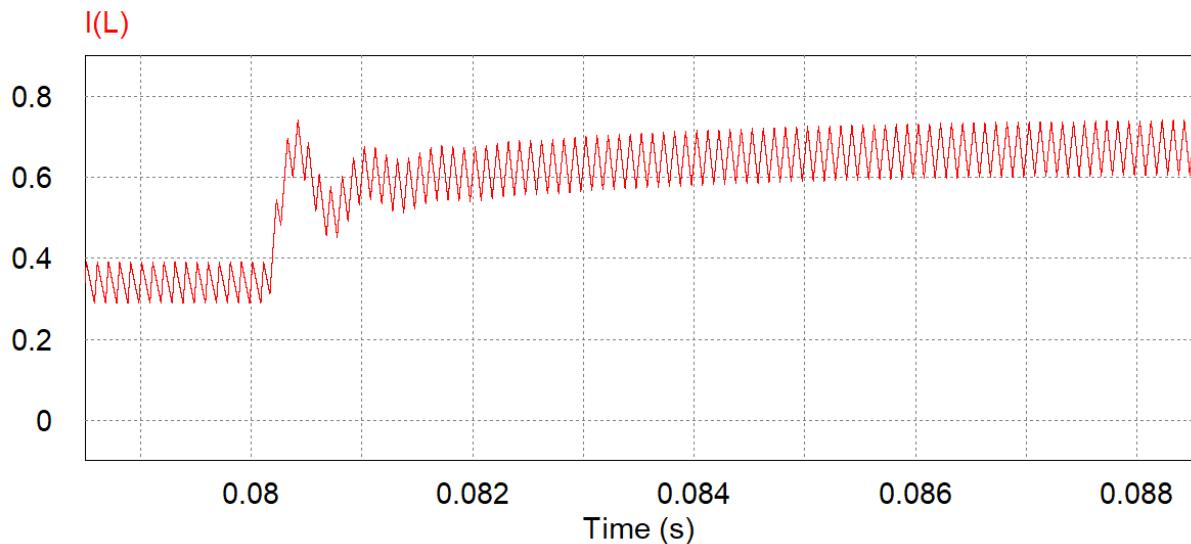


Fonte: Próprio autor.

10.3.2 Degrau na referência de corrente

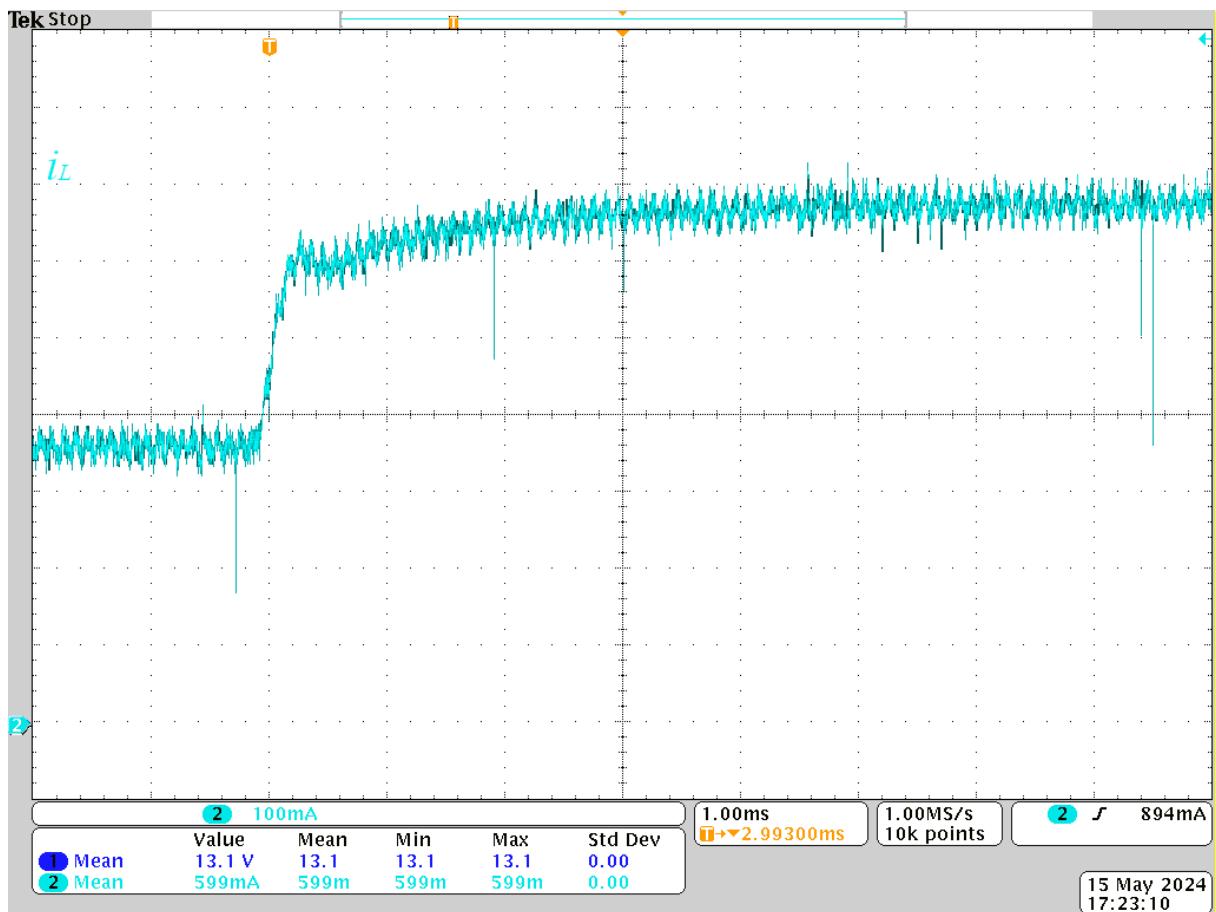
Este experimento serve para a validação do controlador de corrente projetado e será realizado apenas com o controlador de corrente da malha de controle implementado. A referência inicial será o valor de 0,340 A, 50% da corrente nominal e, após pressionado o botão da placa de instrumentação, a referência irá a 0,680 A, 100% da corrente nominal (considerando apenas um resistor de carga). A Figura 52 apresenta o resultado da simulação e Figura 53 apresenta o resultado do controlador implementado. Pode-se observar que a corrente parte de um valor aproximado de 0,350 A para o valor aproximado de 0,680 A.

Figura 52 - Simulação degrau na referência de corrente



Fonte: Próprio Autor

Figura 53 - Degrau na referência do controlador de corrente no conversor

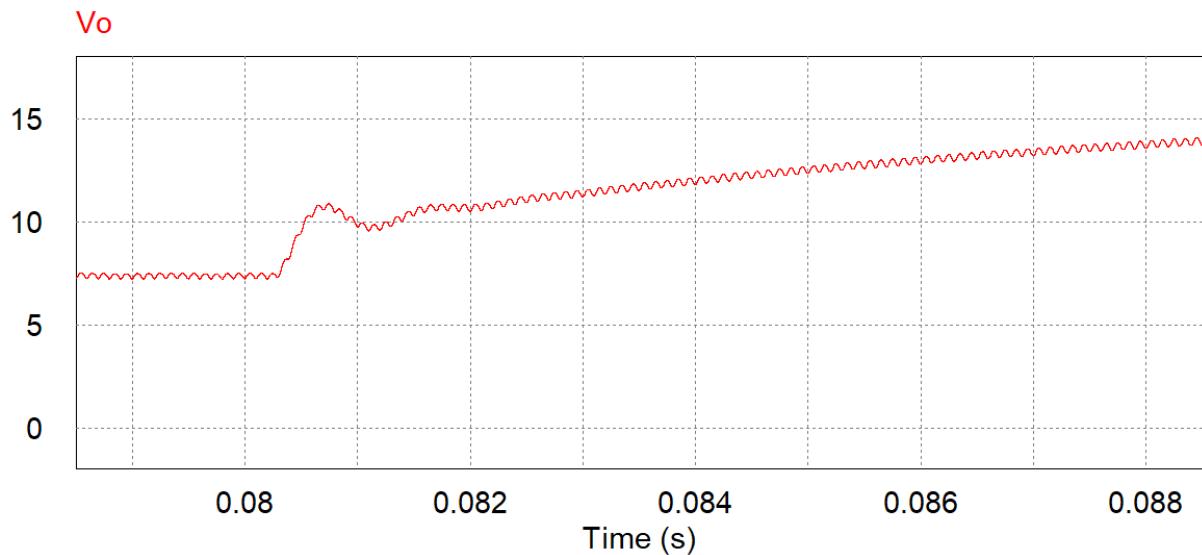


Fonte: Próprio autor.

10.3.3 Degrau na referência de tensão

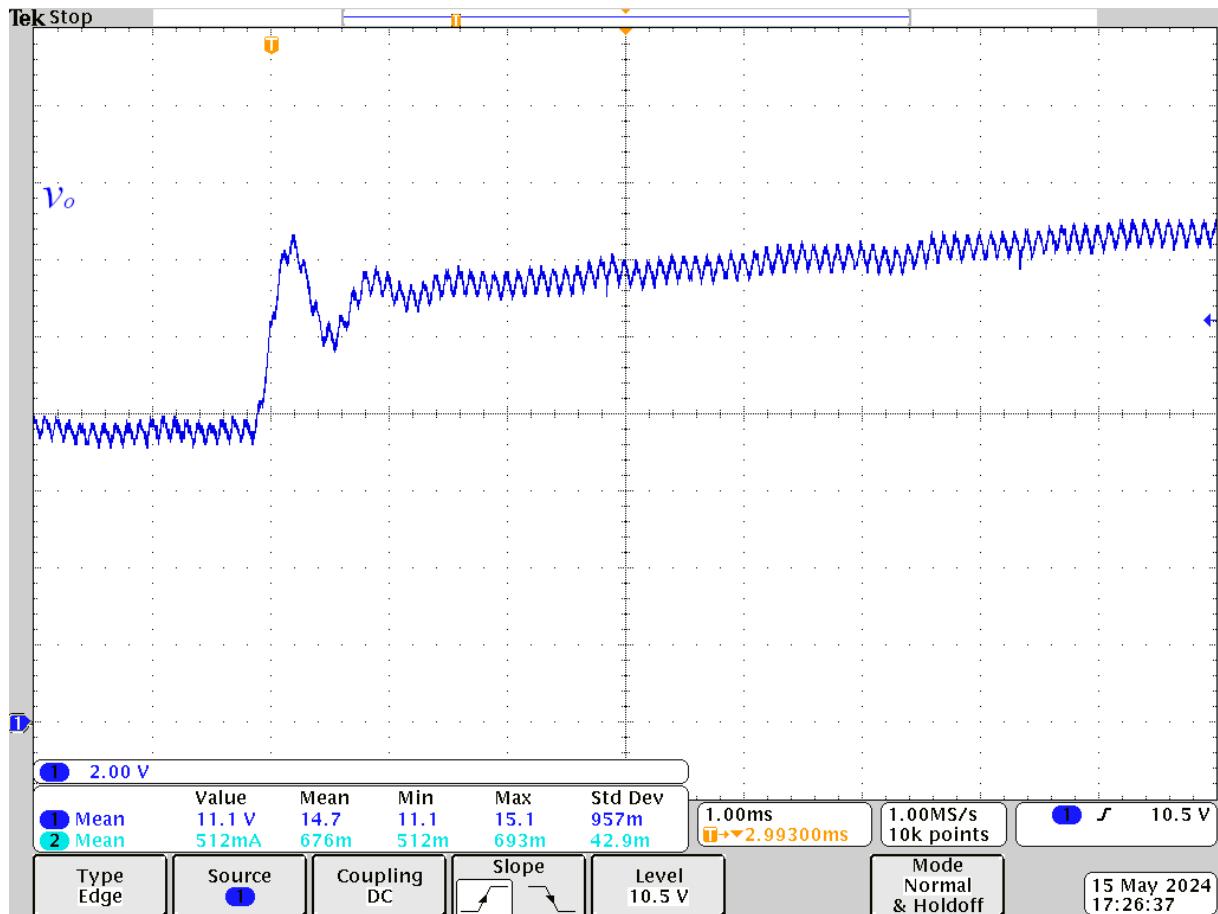
Este é o experimento responsável pela validação da malha de controle completa, a de tensão em cascata com a de corrente, onde a ação de controle da malha de tensão é a entrada da malha de corrente, como pode ser visto na Figura 26. Este experimento tem como referência inicial 7,5 V e, ao pressionar o botão da placa de instrumentação, a referência irá a 15 V. A Figura 54 mostra o resultado da simulação de um degrau na referência de tensão e a Figura 55 apresenta o resultado da malha de controle completa implementada. Pode-se ver que o conversor parte de um valor aproximado de 7,5 V para o valor aproximado de 14,8 V.

Figura 54 - Simulação de um degrau na referência de tensão



Fonte: Próprio autor.

Figura 55 - Degrau na referência de tensão no conversor

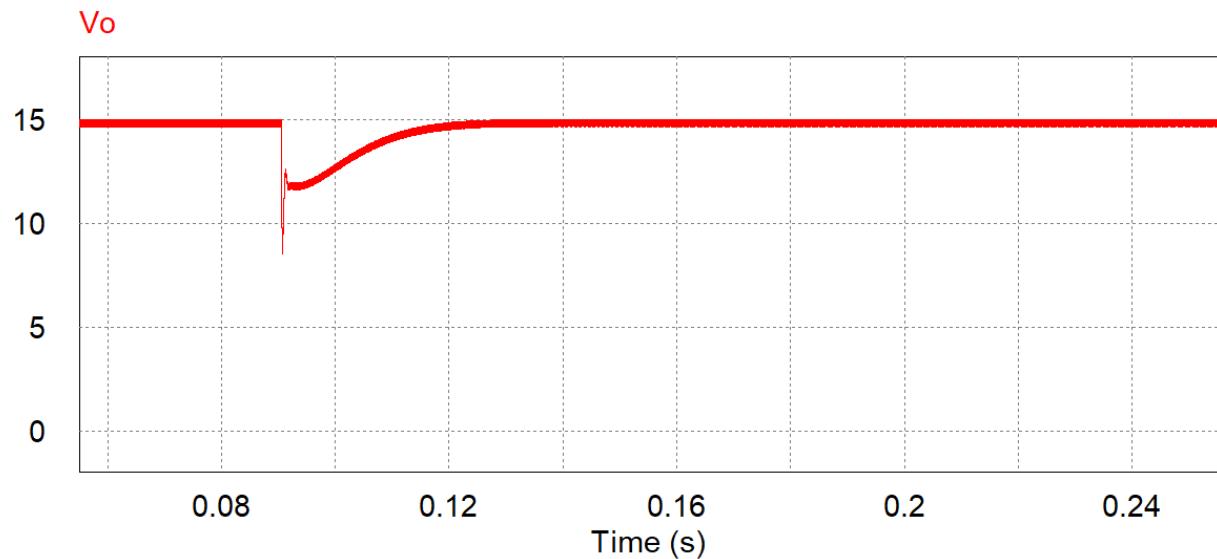


Fonte: Próprio autor.

10.3.4 Degrau de carga

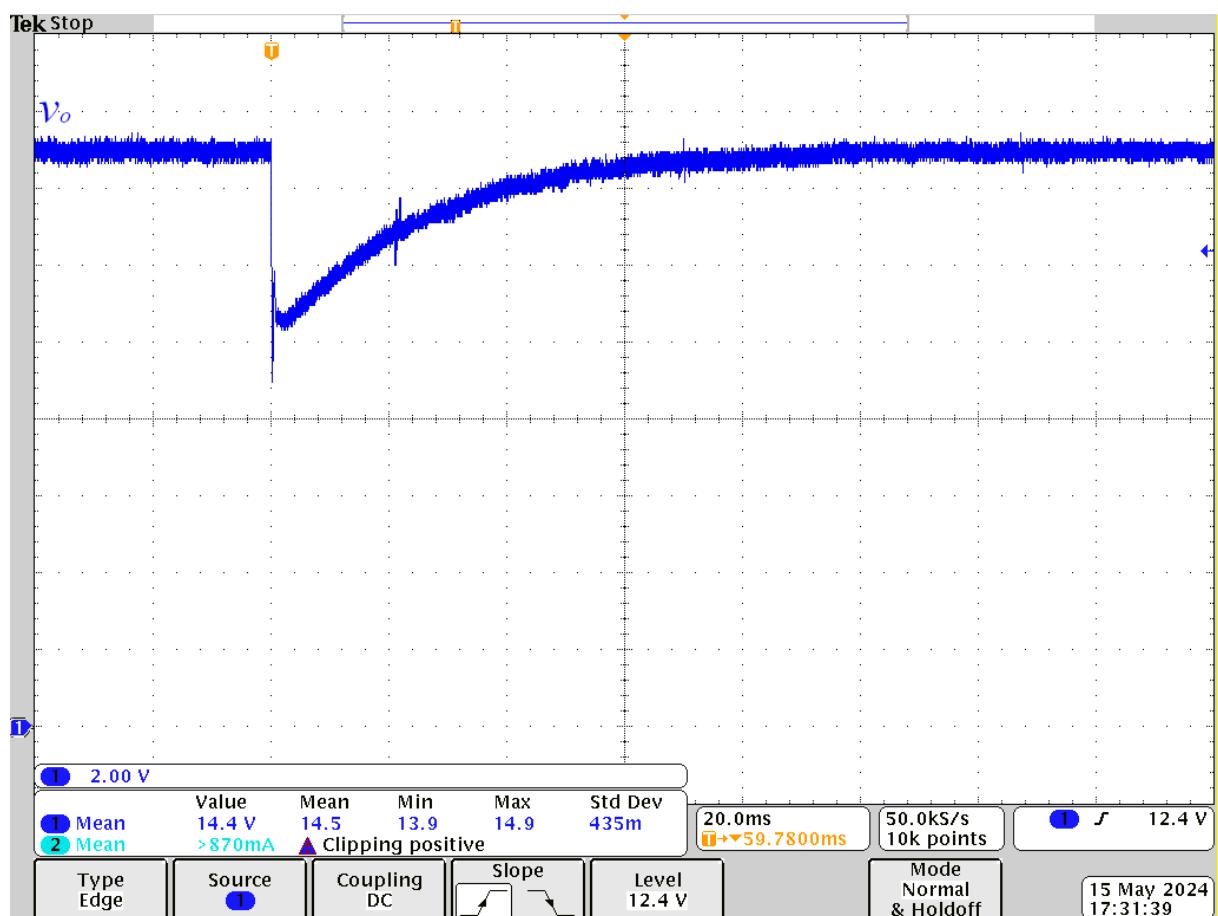
Este experimento avalia a robustez da malha de controle projetada para variação na carga. Ele funcionará da seguinte maneira: primeiramente o conversor estará funcionando com a referência de tensão em 15 V em meia carga ($50\% P_o \quad 22 \Omega$), será adicionado o *jumper* colocando as resistências de carga do conversor em paralelo, assim operando em plena carga ($100\% P_o \quad 11 \Omega$). A Figura 56 apresenta o resultado simulado de um degrau de carga no conversor e a Figura 57 apresenta o comportamento da tensão de saída neste experimento, partindo do valor de aproximadamente 15 V para um vale de 10,5 V, após isso se aproxima da tensão de referência novamente, próximo do valor de 14,6 V no final da amostragem.

Figura 56 - Simulação de um degrau na carga



Fonte: Próprio autor.

Figura 57 - Degrau de carga no conversor



Fonte: Próprio autor.

CAPÍTULO 11 CONCLUSÃO

Os objetivos deste projeto do desenvolvimento de um *kit* didático para as disciplinas de Eletrônica de Potência e Controle Digital foram atendidos. O *kit* permite a realização de aulas práticas com os experimentos mencionados que melhoram a qualidade das aulas, facilitando a retenção de conhecimento pelos alunos. A partir do próximo semestre o *kit* implementado contribuirá para o aprendizado dos alunos da Universidade Federal de Santa Maria.

Comparado com as soluções comerciais, este trabalho apresenta uma representação mais clara da topologia de estudo, conversor CC-CC *Buck*, com várias opções de configuração de componentes e ensaios práticos. O *layout* das placas projetadas é realizado em *software open source*, facilitando trabalhos futuros pelos alunos.

O *gate-driver* desenvolvido possui a vantagem de possuir compatibilidade com diversas topologias que podem ser adicionadas ao *kit* pela sua característica de ter a saída completamente isolada da entrada. A instrumentação apresenta robustez, atendendo completamente aos requisitos do projeto.

O microcontrolador utilizado na instrumentação, o STM32F103C8T6, possui baixo custo, sua utilização é um pouco mais complexa que um Arduino, mas mais simples que um DSP. É muito mais potente que diversos microcontroladores comerciais pelo mesmo custo, sendo uma ótima escolha para projetos de controle simples.

Há pontos de melhora observados durante a montagem do *kit* e dos experimentos realizados, sendo eles:

1. Sensor de corrente de alto custo;
2. Resistores de carga aquecendo muito em operação em potência nominal;
3. Necessário uma fonte auxiliar simétrica em $\pm 15\text{ V}$ para alimentar 2 componentes (sensor de corrente e amplificador operacional).

Todos os arquivos, códigos e instruções adicionais para a replicação deste trabalho podem ser acessados no link:

<https://github.com/Pythorian/Conversor-Buck-Didatico/>

CAPÍTULO 12 REFERÊNCIAS BIBLIOGRÁFICAS

ADVANCED MONOLITHC SYSTEMS. **AMS111733.** Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/49118/ADMOS/AMS1117.html>>. Acesso em: 30 maio. 2024.

AL BASTAMI, A.; AVESTRUZ, A.-T.; LEEB, S. B. **Teaching modeling, control, and simulation in a modular kit for power electronics.** 2014 IEEE 15th Workshop on Control and Modeling for Power Electronics (COMPEL). **Anais...**2014.

ANG, S. S. A practice-oriented course in switching converters. **IEEE Transactions on Education**, v. 39, n. 1, p. 14–18, 1996.

ANTENOR POMILIO, J. EXPERIMENTAL ACTIVITIES IN POWER ELECTRONICS COURSE: MERGING KNOWLEDGE AND APPLICATION. **Eletrônica de Potência**, v. 25, n. 2, p. 146–153, 30 jun. 2020.

BIZZARRI, F. et al. **A Stability Condition for Constant-On Time Buck Converters Suitable for Automotive Applications.** 2021 IEEE International Symposium on Circuits and Systems (ISCAS). **Anais...**2021.

ERICKSON, R. W. (ROBERT W.; MAKSIMOVIĆ, D. **Fundamentals of power electronics.**

EXSTO TECNOLOGIA. **Banco de Ensaios para Eletrônica de Potência**, 2016.

FERREIRA MACHADO, P. C.; ABRITTA COSTA, I.; DO CARMO, J. M. **DESENVOLVIMENTO DE LABORATÓRIO MODULAR PARA APRIMORAMENTO DE COMPETÊNCIAS E HABILIDADES EM ELETRÔNICA DE POTÊNCIA.**

GAO, Z. et al. A 96% Peak Efficiency Adaptively Controlled PSM Buck Converter With Low-Quiescent Current and Wide Dynamic Range for IoT Applications. **IEEE Solid-State Circuits Letters**, v. 5, p. 276–279, 2022.

HP. **HCPL3120.** Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/82098/HP/HCPL3120.html>>. Acesso em: 30 maio. 2024.

JIMENEZ-MARTINEZ, J. M. et al. A new approach for teaching power electronics converter experiments. **IEEE Transactions on Education**, v. 48, n. 3, p. 513–519, 2005.

LEM. **LA25NP.** Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/81509/LEM/LA25-NP.html>>. Acesso em: 30 maio. 2024.

LI, Z. et al. **A Single-input Dual-output Three-level Buck Converter for SoC Applications.** 2022 IEEE International Conference on Integrated Circuits, Technologies and Applications (ICTA). **Anais...**2022.

MAX, L. et al. Power Electronics Design Laboratory Exercise for Final-Year M.Sc. Students. **IEEE Transactions on Education**, v. 52, n. 4, p. 524–531, 2009.

MOTOROLA. **IRF530.** Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/3043/MOTOROLA/IRF530.html>>. Acesso em: 30 maio. 2024a.

MOTOROLA. **MUR840.** Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/5593/MOTOROLA/MUR820.html>>. Acesso em: 30 maio. 2024b.

PEREZ, D. et al. **Training kit for power electronics teaching.** 2008 34th Annual Conference of IEEE Industrial Electronics. **Anais...**2008.

SCIENTECH TECHNOLOGIES. **Buck Converter.** Disponível em: <<https://www.scientechworld.com/pdf/Scientech-2726.pdf>>. Acesso em: 30 maio. 2024.

SHRUD, MOHAMED. A. et al. **Modeling and simulation of automotive interleaved buck converter.** 2009 44th International Universities Power Engineering Conference (UPEC). **Anais...**2009.

STMICROELECTRONICS. **RM0008 Reference manual STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced Arm ®-based 32-bit MCUs**

- STM32F10xxx Cortex ®-M3 programming manual (PM0056) • STM32F10xxx Flash memory programming manual (PM0075) • STM32F10xxx XL-density Flash memory programming manual (PM0068).. Disponível em: <www.st.com>.

STMICROELECTRONICS. **STM32-base Bluepill.** Disponível em: <<https://stm32-base.org/boards/STM32F103C8T6-Blue-Pill.html>>. Acesso em: 30 maio. 2024a.

STMICROELECTRONICS. **TL084.** Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/25385/STMICROELECTRONICS/TL084.html>>. Acesso em: 30 maio. 2024b.

SUSIN, R. M. **Laboratório de Ensino da Eletrônica de Potência - Uma Experiência Construída Para e Pelos Estudantes.** Porto Alegre, 2000.

TEXAS INSTRUMENTS®. **CD40106B.** Disponível em: <<https://pdf1.alldatasheet.com/datasheet-pdf/view/26839/TI/CD40106.html>>. Acesso em: 30 maio. 2024.

TEXAS INSTRUMENTS®. **Texas Instruments Power to Educate Students.**

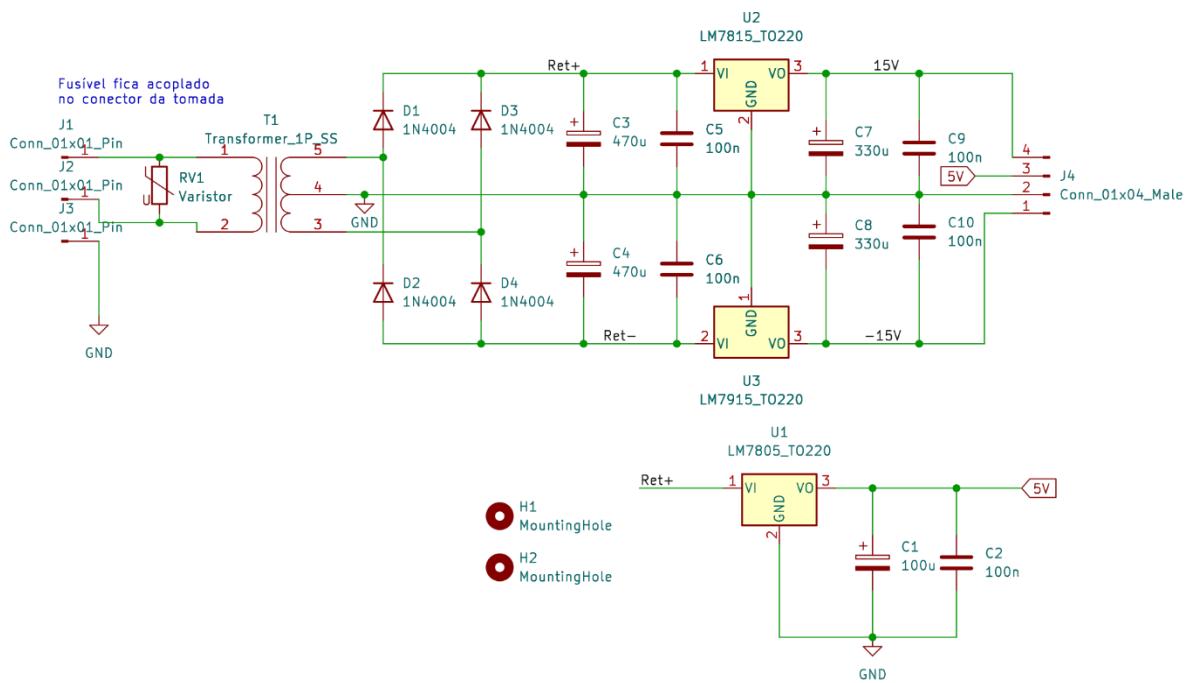
APÊNDICE A – PROJETO DA FONTE AUXILIAR

A placa de instrumentação possui 3 componentes principais, o sensor hall, o amplificador operacional e o microcontrolador. Os componentes selecionados neste projeto necessitam das seguintes tensões de alimentação, +15V, -15V, +5V e 0V. Assim se fez necessário o projeto de uma fonte auxiliar para a alimentação destes componentes.

12.1 Esquemático da fonte auxiliar

Esta fonte precisa de uma saída $\pm 15V$ simétrica, o que torna necessário um transformador com *tap* central. Foram escolhidos os reguladores lineares LM9805, LM7815 e LM 7915 para obter as tensões de saída necessárias. A Figura 58 mostra o esquemático da fonte de alimentação com esses componentes.

Figura 58 – Esquemático da fonte de alimentação auxiliar



Fonte: Próprio autor.

12.2 Lista de componentes da fonte auxiliar

A Tabela 10 apresenta a lista de componentes necessários para a produção da fonte de alimentação auxiliar cujo esquemático foi apresentado. Os componentes mais importantes selecionados são o transformador com *tap* central e os reguladores lineares já mencionados.

Tabela 10 - Componentes fonte de alimentação auxiliar

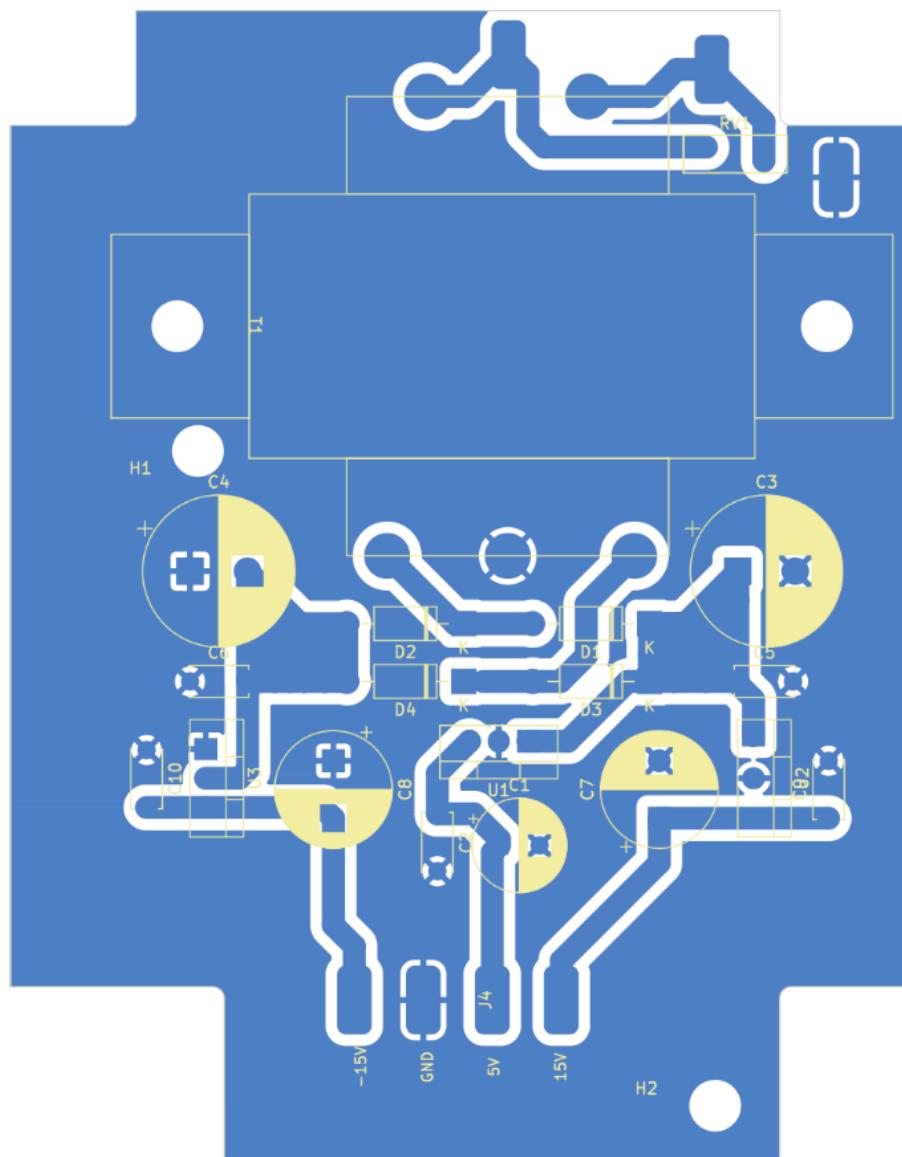
Nome do componente	Tipo do componente	Tecnologia do componente	Modelo	Faixa de tensão	Faixa de corrente
J1; J2; J3	Fio	—	AWG 32	—	1 A
T1	Transformador	Ferrite 60 Hz	15+15 V _{RMS}	220 V RMS	500 mA
U1	Regulador Linear	TO220	LM7805	20 V	1 A
U2	Regulador Linear	TO220	LM7815	30 V	1 A
U3	Regulador Linear	TO220	LM7915	-30 V	1 A
D1; D2; D3; D4	Diodo	THT	1N4007	700 V	1 A
C1	Capacitor	THT Eletrolítico	100 uF	63 V	—
C2; C5; C6; C9; C10	Capacitor	THT Cerâmico	100 nF	63 V	—
C3; C4	Capacitor	THT Eletrolítico	470 uF	63 V	—
C7; C8	Capacitor	THT Eletrolítico	330 uF	63 V	—
RV1	Varistor	THT	—	300 V	—
J4	Conecotor KK	THT	KK 01x04 Vertical	100 V	3 A

Fonte: Próprio autor.

12.3 Layout da fonte auxiliar

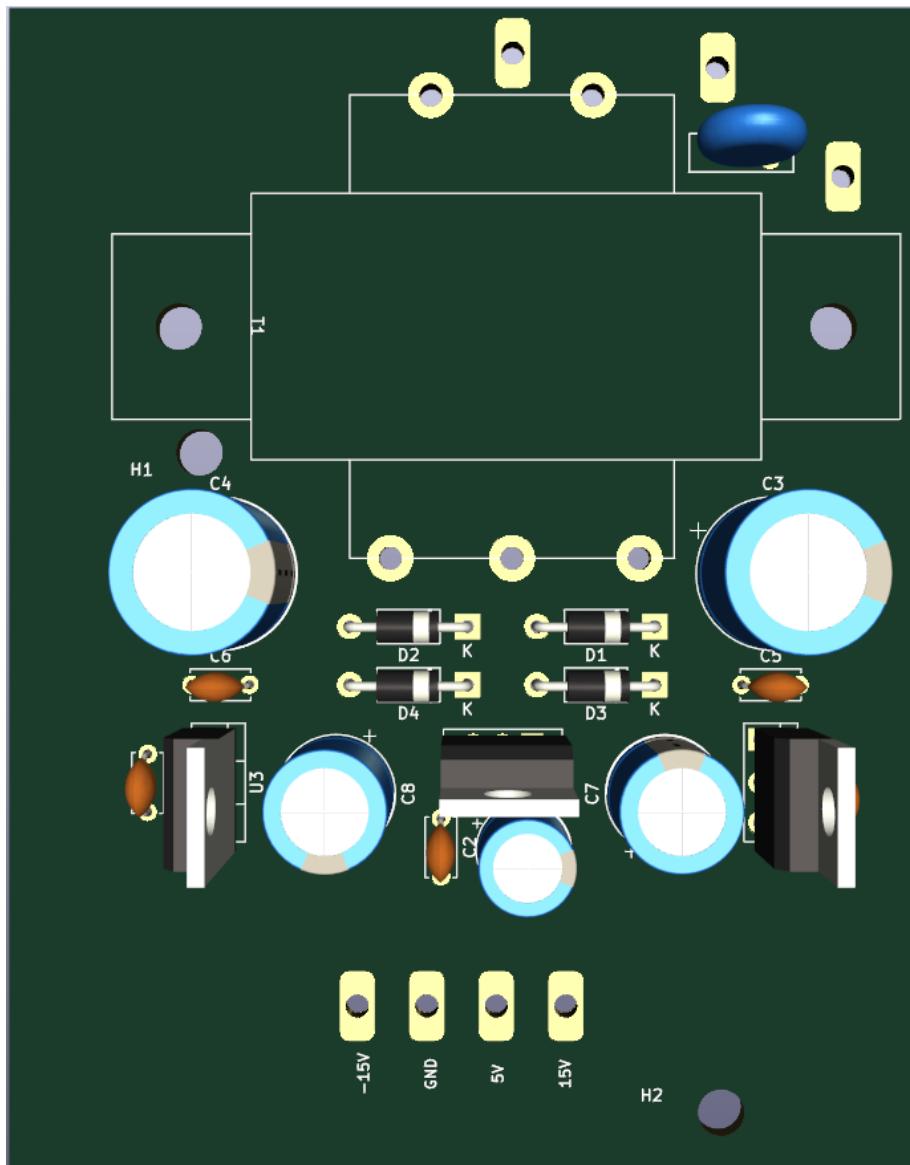
O *layout* da fonte auxiliar foi desenvolvido no *software* Kicad, ele levou em consideração as restrições de espaço dadas pela caixa patola escolhida para a fonte. O projeto do *layout* pode ser visto na Figura 59, e sua visualização 3D pode ser vista na Figura 60.

Figura 59 - *Layout* da fonte de alimentação auxiliar



Fonte: Próprio autor.

Figura 60 - Visualização 3D da fonte auxiliar



Fonte: Próprio autor.

12.4 Fonte de alimentação auxiliar fabricada

As Figura 61 e Figura 62 apresentam a fonte auxiliar fabricada de maneira artesanal, utilizando transferência térmica e banho de percloro de ferro. Ela está anexada à caixa patola com 3 parafusos, sendo um dos parafusos também responsável pela fixação do transformador. Alguns ajustes manuais foram necessários para a inclusão do botão para ligar a fonte junto do fusível e conexão para cabo de alimentação.

Figura 61 - Fonte de alimentação auxiliar fabricada



Fonte: Próprio autor.

Figura 62 - Foto da fonte de alimentação aberta



Fonte: Próprio autor.

APÊNDICE B – SCRIPT EM MATLAB® PARA O PROJETO DOS CONTROLADORES

```
%-----%
%           UFSM - Universidade Federal de Santa Maria %
%           Curso de Engenharia de Controle e Automação %
%           Trabalho de Conclusão de Curso %
%
% Programadores: %
%           Acad. Milton Brenner Machado Matoso %
%           Prof. Rafael Concatto Beltrame %
%
% Versão: 5.0 % 08/05/2024 %
%-----%
%             Descrição do Programa %
%
% Projeto dos controladores das malhas de corrente (interna) e %
% tensão (externa) de um conversor CC-CC Buck. %
%
% v5.0 - Alterada a posição do zero do PI %
% v4.0 - Alterada a frequência de corte do controlador de corrente %
% v3.0 - Projeto do compensador de tensão. Melhor diferenciação das %
%         variáveis das malhas de tensão e corrente. %
% v2.0 - Projeto realizado exclusivamente no plano s (contínuo). %
% v1.0 - Versão inicial. %
%
close all % Fecha todos os gráficos
clear all % Exclui todas as variáveis
clc % Limpa a tela
format long eng % Formato para exibição numérica %

%-----%
% Especificações %
%
Vin = 30; % Tensão de entrada (V)
L = 5.6e-3;%2.8e-3; % Indutância de filtro (H)
C = 4.7e-6;%22e-6; % Capacitância de filtro (F)
Ro = 22; % Resistência de carga (Ohm)
D = 0.5; % Razão-cíclica
fs = 10e3; % Frequência de chaveamento (Hz)
Ts = 1/fs; % Período de chaveamento (s)
fclk = 72e6; %16e6; %64e6; % Frequência de clock (Hz)
Tclk = 1/fclk; % Período de clock (s)

%% -----
% Modelo da Planta (corrente para razão-cíclica) %
%
s = tf('s'); % Variável de Laplace

num = Vin/L * [1 1/(C*Ro)]; % Função de transferência da planta
den = [1 1/(C*Ro) 1/(C*L)];
GiLd = tf(num,den);

cont_max = (fclk/fs)/2; % Valor máximo da triangular
M = 1/cont_max; % Ganho do modulador PWM
GiLdM = M * GiLd; % Adição do ganho do modulador % Planta

%-----%
% Validação no PSIM %
%
time_step = Tclk; % Passo de simulação (s)
total_time = 50*Ts; % Tempo total (s)
```

```

% Condições iniciais
Vo = D*Vin;                                % Tensão média no capacitor (V)
IL = D*Vin/Ro;                               % Corrente média no indutor (A)

%-----
% Modelo do Atraso de Implementação
%-----
% Referência: TCC de Jeferson Calai (pág. 50)

fai = 2*fs;                                  % Frequência de amostragem (Hz)
Tai = 1/fai;                                 % Período de amostragem (s)

Gatrasoi = tf([-1 2/Tai],[1 2/Tai]);        % Função de transferência do atraso
                                                % (Aproximação de Pade)

GiLdMa = GiLdM * Gatrasoi;                  % Adição do atraso de um sample

%-----
% Projeto da Malha de Controle da Corrente
%-----
fci = fai/10;                                % Frequência de cruzamento desejada (Hz)
wci = 2*pi*fci;                             % (rad/s)

%-----
% SISO Tool
%-----
% sisotool

%-----
% Controlador PI
%-----
wp_pii = 0;                                  % Polo (rad/s)
wz_pii = wci/10;                            % Zero (rad/s)
Cpii   = tf([1 wz_pii],[1 wp_pii]);        % Função de transferência

k = 3530.9;                                 % Ganho proporcional (sisotool)
Cpii2 = k * Cpii;                           % Adição do ganho proporcional

ftmf = Cpii2*GiLdMa/(1+(Cpii2*GiLdMa))

%-----
% Discretização
%-----
Gpii_z = c2d(Cpii2, Tai, 'tustin');        % Discretização

[num, den] = tfdata(Gpii_z, 'v');            % Extração dos coeficientes

a1 = num(1)
a2 = num(2)

%% -----
% Modelo da Planta (tensão para corrente)
%-----
num = [1/C];                                % Função de transferência da planta
den = [1 1/(C*Ro)];
Gvi = tf(num, den);

% %-----%
% % Validação no PSIM %
% %-----%
% time_step = Tclk;                         % Passo de simulação (s)
% total_time = 50*Ts;                        % Tempo total (s)
%
% % Condições iniciais
% Vo = D*Vin;                                % Tensão média no capacitor (V)
% IL = D*Vin/Ro;                             % Corrente média no indutor (A)

%-----

```

```

% Modelo do Atraso de Implementação
%-----%
% Referência: TCC de Jeferson Calai (pág. 50)

fav = fai;                                % Frequência de amostragem (Hz)
Tav = 1/fav;                               % Período de amostragem (s)

Gatrasov = tf([-1 2/Tav],[1 2/Tav]);      % Função de transferência do atraso
                                              % (Aproximação de Pade)

Gvia = Gvi * Gatrasov;                     % Adição do atraso de um sample

%-----%
% Projeto da Malha de Controle da Corrente
%-----%
% OBS: A malha de tensão deve ser, pelo menos 10 vezes mais lenta que a
%       malha de corrente
fcv = fci/10;                             % Frequência de cruzamento desejada (Hz)
wcv = 2*pi*fcv;                           % (rad/s)

%-----%
% SISO Tool
%-----%
% sisotool

%-----%
% Controlador PI
%-----%
wp_piv = 0;                                % Polo (rad/s)
wz_piv = wcv/10;                            % Zero (rad/s);
Cpiv   = tf([1 wz_piv],[1 wp_piv]);        % Função de transferência

% OBS: Como a margem de fase ficou em 156°, não foi necessária a
%       inclusão de um compensador PD.

k = 0.044684;                             % Ganho proporcional (sisotool)
Cpiv2 = k * Cpiv;                         % Adição do ganho proporcional

%-----%
% Discretização
%-----%
Gpiv_z = c2d(Cpiv2, Tav, 'tustin');       % Discretização

[num, den] = tfdata(Gpiv_z, 'v');           % Extração dos coeficientes

av1 = num(1)
av2 = num(2)

```

APÊNDICE C - SCRIPT EM C PARA A SIMULAÇÃO NO PSIM®

```

#include <Stdlib.h>
#include <String.h>

int g_nInputNodes=0;
int g_nOutputNodes=0;

int g_nStepCount=0;

static int cont, cont_max, cont_min, step, PWM;

static float uik0, uik1;
static float eik0, eik1, eik2;

static float uvk0, uvk1;
static float evk0, evk1, evk2;

static float iL, iLref, Vo, Voref;

const float ai1 = 313.569362420081;
const float ai2 = -303.870637579919;

const float av1 = 48.2156570163295e-003;
const float av2 = -46.7243429836705e-003;
//-----
g_nStepCount++;

// In case of error, uncomment next two lines. Set *pnError to 1 and copy Error
message to szErrorMsg
///*pnError=1;
//strcpy(szErrorMsg, "Place Error description here.");

//-----
// Triangular
//-----
cont_max = 800;
cont_min = 0;

if (cont >= cont_max) { step= -1; }
if (cont <= cont_min) { step = 1; }
cont = cont + step;

//-----
// Interrupção
//-----
if (cont >= cont_max) {

    //-----//
    // Referencia de tensao
    //-----//
    Voref = 15;

    // Degrau de referencia no instante 80 ms
    //if (t>=0.08) {Voref = 15; }

    //-----//
    // Atualização da ação de controle
    //-----//
    uvk0 = uvk1 + av1*evk1 + av2*evk2;

    //-----//
    // Conversao AD
    //-----//
}

```

```

//-----
// Leitura da tensao
Vo = in[1];

//-----
// Calculo do erro
//-----
evk0 = Voref - Vo;

//-----
// Rotaçao das variaveis
//-----
evk2 = evk1;
evk1 = evk0;
uvk1 = uvk0;

//-----
// Referencia de corrente
//-----
iLref = uvk0;
//iLref = 340e-3;

// Degrau de referencia no instante 80 ms
//if (t>=0.08) {iLref = 680e-3;}

//-----
// Atualizaçao da ação de controle
//-----
uik0 = uik1 + a11*eik1 + a12*eik2;

//-----
// Conversao AD
//-----
// Leitura da corrente
iL = in[0];

//-----
// Calculo do erro
//-----
eik0 = iLref - iL;

//-----
// Rotaçao das variaveis
//-----
eik2 = eik1;
eik1 = eik0;
uik1 = uik0;
}

//-----
// PWM
//-----
if (uik0 >= cont) { PWM = 1; }
else { PWM = 0; }

//-----
// Saidas
//-----
out[0] = PWM;
out[1] = iL;
out[2] = uik0;
out[3] = eik0;
out[4] = uvk0;
out[5] = evk0;
out[6] = iLref;
out[7] = Voref;

```

APÊNDICE D – *SCRIPT EM C IMPLEMENTADO NO MICROCONTROLADOR*

O microcontrolador utilizado no projeto foi o STM32F103C8T6, comumente conhecido como *Bluepill*. Ele possui as vantagens de um grande poder de processamento, alta eficiência energética e maior memória quando comparado com outras soluções de microcontroladores comerciais na mesma faixa de preço. O custo chega a R\$13,00 com entrega em vendedores na Aliexpress. Atenta-se ao fato que ele precisa de uma placa para fazer o *upload* do código para o processador. O nome da placa escolhida e utilizada neste projeto é a STLINK V2, com custo de R\$6,52.

Comparado com microcontroladores mais utilizados nos cursos de engenharia atualmente como o Arduino, a barreira de entrada é um pouco maior. Apesar de existirem bibliotecas que permitam a sua programação diretamente pela porta USB com a Arduino IDE, é recomendado a utilização do *software* de programação do desenvolvedor das placas STM32, chamado de CUBE IDE. O nível de dificuldade de utilização do STM fica entre um DSP e um Arduino.

O clock de processamento conforme o *datasheet* é de 72 MHz, sua alimentação pode ser feita em 5 V ou 3,3 V, possui pinos que são tolerantes a tensão de 5 V, todos os ADCs presentes toleram apenas até 3,3 V. Clique ou toque aqui para inserir o texto.. Para entender melhor as funcionalidades do microcontrolador, é necessário consultar seu manual (STMICROELECTRONICS, 2021). Seus pinos têm até 5 funções diferentes para serem configuradas, e apesar de possuir vários *timers*, eles possuem diferentes limitações, podendo facilmente ocasionar um erro de projeto.

Após instalar o *software* CUBE IDE, cria-se o projeto em que será feito o *upload* na placa. O *software* permite ao usuário fazer a configuração dos pinos do STM32 de maneira gráfica, gerando o código automaticamente para o usuário, e ao atualizar qualquer configuração, todo o código gerado automaticamente é atualizado. Quando gerado o código de configurações de maneira automática, haverá espaços indicados no código para o usuário adicionar o seu próprio. Felizmente, esses espaços próprios não são sobrescritos após atualizar as configurações. As configurações também podem ser feitas de maneira manual pelo usuário no código.

Este projeto foi desenvolvido com as seguintes configurações feitas na interface gráfica do CUBE IDE.

- 1- Iniciar o projeto e selecionar o microcontrolador STM32F103C8T6;
- 2- Configurar o *clock* em 72 MHz;
- 3- Configurar o *timer* TIM3 CHANNEL 2 para a geração do sinal PWM e fazer as interrupções. O pino PA7 é o canal 2 do *timer* 3;
- 4- PA1, PA2 e PA3 são configurados dentro do ADC1, dentro da opção Analog/ADC1 nas configurações;
- 5- PA9 e PA10 são o par utilizado na porta serial para *debug*, e devem ser configurados dentro da categoria *Connectivity USART1*;
- 6- PB0 e PB1 são configurados como GPIO_Output, são os 2 LEDs presentes na placa de instrumentação;
- 7- PB11 é configurado como GPIO_Input, ele é o botão táctil presente na placa de instrumentação.
- 8- Dentro de ADC1, configurar o *trigger* da conversão para o TIM3 e os *ranks* dos ADCs para a leitura, vão ser armazenados na ordem que forem ranqueados;
- 9- No TIM3 *channel 2*, configurar o PWM para *center aligned mode 1, counter period* 3599, para uma frequência de 10kHz com o *clock* do processador em 72MHz;
- 10- Na configuração NVIC, garantir que o TIM3 *global interrupt* está habilitado;
- 11- A função chamada na interrupção do TIM3 no código chama-se: Void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef* htim){ }. Dentro dela está o código que será executado na interrupção;
- 12- Para habilitar a utilização de variáveis do tipo *float*, configurar em: Project – Properties – C/C++ Build/Settings/Tool Settings/MCU Settings – check both boxes. Floating-point ABI Software implementation.

O código presente dentro do arquivo main.c da programação do microcontrolador segue abaixo:

```
/* USER CODE BEGIN Header */
/**
 * @file          : main.c
 * @brief         : Main program body
 */
 * @attention
 *
 * Copyright (c) 2023 STMicroelectronics.
 * All rights reserved.
 *
 * This software is licensed under terms that can be found in the LICENSE file
 * in the root directory of this software component.
 * If no LICENSE file comes with this software, it is provided AS-IS.
 */
```

```
*****
*/
/* USER CODE END Header */
/* Includes ----- */
#include "main.h"

/* Private includes ----- */
/* USER CODE BEGIN Includes */
#include "string.h"
#include "stdlib.h"
#include "math.h"
#include "stdio.h"
/* USER CODE END Includes */

/* Private typedef ----- */
/* USER CODE BEGIN PTD */

/* USER CODE END PTD */

/* Private define ----- */
/* USER CODE BEGIN PD */
#define ADC_BUF_SIZE 3
/* USER CODE END PD */

/* Private macro ----- */
/* USER CODE BEGIN PM */

/* USER CODE END PM */

/* Private variables ----- */
ADC_HandleTypeDef hadc1;
DMA_HandleTypeDef hdma_adc1;

TIM_HandleTypeDef htim3;

UART_HandleTypeDef huart1;

/* USER CODE BEGIN PV */

//-----
// Declaracao de variaveis para a leitura ADC//
//-----
float pot_malha_aberta = 0.0f;
float pot_corrente = 0.0f;
float pot_tensao = 0.0f;
float corrente = 0.0f;
float tensao = 0.0f;
uint16_t ref_pwm = 0;

float ganho_pot_malha_aberta = 0.439560439f;//1800/4095
float ganho_pot_corrente = 0.0001663f; // 0.681A/4095
float ganho_pot_tensao = 3.663e-3f; // 15V/4095
float ganho_corrente = 1.25031e-3f; // 5.12A/4095
float ganho_tensao = 9.76801e-3f; // 40V/4095

float pot_adc = 0.0f;
float corrente_adc = 0.0f;
float tensao_adc = 0.0f;

volatile uint8_t FLAG_READY=0;

volatile uint16_t ADC_buffer[ADC_BUF_SIZE];

//-----Variaveis para comunicacao serial--//
uint16_t SERIAL_DELAY = 1000;
uint8_t SERIAL_TIMES = 0;
//-----//
```

```

//-----//
// Declaracao variaveis controle          //
//-----//
static float uik0, uik1;
static float eik0, eik1;

static float uvk0, uvk1;
static float evk0, evk1;

static float iL, iLref, Vo, Voref;

//Ganhos do compensador de corrente
const float ai1 = 3.64182649500560e+003;
const float ai2 = -3.41997350499440e+003f;

//Ganhos do compensador de tensao
const float av1 = 44.8243789261330e-003f;
const float av2 = -44.5436210738670e-003f;

/* USER CODE END PV */

/* Private function prototypes -----*/
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_DMA_Init(void);
static void MX_ADC1_Init(void);
static void MX_TIM3_Init(void);
static void MX_USART1_UART_Init(void);
/* USER CODE BEGIN PFP */
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim);

/* USER CODE END PFP */

/* Private user code -----*/
/* USER CODE BEGIN 0 */

/* USER CODE END 0 */

/**
 * @brief  The application entry point.
 * @retval int
 */
int main(void)
{
    /* USER CODE BEGIN 1 */

    /* USER CODE END 1 */

    /* MCU Configuration-----*/

    /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
    HAL_Init();

    /* USER CODE BEGIN Init */

    /* USER CODE END Init */

    /* Configure the system clock */
    SystemClock_Config();

    /* USER CODE BEGIN SysInit */

    /* USER CODE END SysInit */

    /* Initialize all configured peripherals */

```

```

MX_GPIO_Init();
MX_DMA_Init();
MX_ADC1_Init();
MX_TIM3_Init();
MX_USART1_UART_Init();
/* USER CODE BEGIN 2 */

//Calibracao do ADC
HAL_ADCEx_Calibration_Start(&hadc1);

//Inicializa o sinal pwm e a interrupcao pelo timer3
HAL_TIM_Base_Start_IT(&htim3); // inicializa a interrupção pelo timer 3
HAL_TIM_PWM_Start(&htim3, TIM_CHANNEL_2); // inicializa o PWM

/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */

while (1)
{
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */
}

/**
 * @brief System Clock Configuration
 * @retval None
 */
void SystemClock_Config(void)
{
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
    RCC_PeriphCLKInitTypeDef PeriphClkInit = {0};

    /** Initializes the RCC Oscillators according to the specified parameters
     * in the RCC_OscInitTypeDef structure.
     */
    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE;
    RCC_OscInitStruct.HSEState = RCC_HSE_ON;
    RCC_OscInitStruct.HSEPredivValue = RCC_HSE_PREDIV_DIV1;
    RCC_OscInitStruct.HSISite = RCC_HSI_ON;
    RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
    RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
    RCC_OscInitStruct.PLL.PLLMUL = RCC_PLL_MUL9;
    if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
    {
        Error_Handler();
    }

    /** Initializes the CPU, AHB and APB buses clocks
     */
    RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                                |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
    RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
    RCC_ClkInitStruct.AHBLKDivider = RCC_SYSCLK_DIV1;
    RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV2;
    RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV1;

    if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_2) != HAL_OK)
    {
        Error_Handler();
    }
}

```

```

PeriphClkInit.PeriphClockSelection = RCC_PERIPHCLK_ADC;
PeriphClkInit.AdcClockSelection = RCC_ADCPCLK2_DIV6;
if (HAL_RCCEx_PeriphCLKConfig(&PeriphClkInit) != HAL_OK)
{
    Error_Handler();
}
}

/**
 * @brief ADC1 Initialization Function
 * @param None
 * @retval None
 */
static void MX_ADC1_Init(void)
{

    /* USER CODE BEGIN ADC1_Init_0 */

    /* USER CODE END ADC1_Init_0 */

    ADC_ChannelConfTypeDef sConfig = {0};

    /* USER CODE BEGIN ADC1_Init_1 */

    /* USER CODE END ADC1_Init_1 */

    /**
     * Common config
     */
    hadc1.Instance = ADC1;
    hadc1.Init.ScanConvMode = ADC_SCAN_ENABLE;
    hadc1.Init.ContinuousConvMode = DISABLE;
    hadc1.Init.DiscontinuousConvMode = DISABLE;
    hadc1.Init.ExternalTrigConv = ADC_EXTERNALTRIGCONV_T3_TRGO;
    hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT;
    hadc1.Init.NbrOfConversion = 3;
    if (HAL_ADC_Init(&hadc1) != HAL_OK)
    {
        Error_Handler();
    }

    /**
     * Configure Regular Channel
     */
    sConfig.Channel = ADC_CHANNEL_1;
    sConfig.Rank = ADC_REGULAR_RANK_1;
    sConfig.SamplingTime = ADC_SAMPLETIME_7CYCLES_5;
    if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
    {
        Error_Handler();
    }

    /**
     * Configure Regular Channel
     */
    sConfig.Channel = ADC_CHANNEL_2;
    sConfig.Rank = ADC_REGULAR_RANK_2;
    if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
    {
        Error_Handler();
    }

    /**
     * Configure Regular Channel
     */
    sConfig.Channel = ADC_CHANNEL_3;
    sConfig.Rank = ADC_REGULAR_RANK_3;
    if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
    {
        Error_Handler();
    }

    /* USER CODE BEGIN ADC1_Init_2 */
}

```

```

/* USER CODE END ADC1_Init 2 */

}

/***
 * @brief TIM3 Initialization Function
 * @param None
 * @retval None
 */
static void MX_TIM3_Init(void)
{
    /* USER CODE BEGIN TIM3_Init 0 */

    /* USER CODE END TIM3_Init 0 */

    TIM_ClockConfigTypeDef sClockSourceConfig = {0};
    TIM_MasterConfigTypeDef sMasterConfig = {0};
    TIM_OC_InitTypeDef sConfigOC = {0};

    /* USER CODE BEGIN TIM3_Init 1 */

    /* USER CODE END TIM3_Init 1 */
    htim3.Instance = TIM3;
    htim3.Init.Prescaler = 0;
    htim3.Init.CounterMode = TIM_COUNTERMODE_CENTERALIGNED1;
    htim3.Init.Period = 3599;
    htim3.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
    htim3.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_ENABLE;
    if (HAL_TIM_Base_Init(&htim3) != HAL_OK)
    {
        Error_Handler();
    }
    sClockSourceConfig.ClockSource = TIM_CLOCKSOURCE_INTERNAL;
    if (HAL_TIM_ConfigClockSource(&htim3, &sClockSourceConfig) != HAL_OK)
    {
        Error_Handler();
    }
    if (HAL_TIM_PWM_Init(&htim3) != HAL_OK)
    {
        Error_Handler();
    }
    sMasterConfig.MasterOutputTrigger = TIM_TRGO_UPDATE;
    sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
    if (HAL_TIMEx_MasterConfigSynchronization(&htim3, &sMasterConfig) != HAL_OK)
    {
        Error_Handler();
    }
    sConfigOC.OCMode = TIM_OCMODE_PWM1;
    sConfigOC.Pulse = 0;
    sConfigOC.OCPolarity = TIM_OCPOLARITY_HIGH;
    sConfigOC.OCFastMode = TIM_OCFAST_ENABLE;
    if (HAL_TIM_PWM_ConfigChannel(&htim3, &sConfigOC, TIM_CHANNEL_2) != HAL_OK)
    {
        Error_Handler();
    }
    /* USER CODE BEGIN TIM3_Init 2 */

    /* USER CODE END TIM3_Init 2 */
    HAL_TIM_MspPostInit(&htim3);
}
/***
 * @brief USART1 Initialization Function
 * @param None
 * @retval None
 */

```

```

/*
static void MX_USART1_UART_Init(void)
{
    /* USER CODE BEGIN USART1_Init 0 */

    /* USER CODE END USART1_Init 0 */

    /* USER CODE BEGIN USART1_Init 1 */

    /* USER CODE END USART1_Init 1 */
    huart1.Instance = USART1;
    huart1.Init.BaudRate = 115200;
    huart1.Init.WordLength = UART_WORDLENGTH_8B;
    huart1.Init.StopBits = UART_STOPBITS_1;
    huart1.Init.Parity = UART_PARITY_NONE;
    huart1.Init.Mode = UART_MODE_TX_RX;
    huart1.Init.HwFlowCtl = UART_HWCONTROL_NONE;
    huart1.Init.OverSampling = UART_OVERSAMPLING_16;
    if (HAL_UART_Init(&huart1) != HAL_OK)
    {
        Error_Handler();
    }
    /* USER CODE BEGIN USART1_Init 2 */

    /* USER CODE END USART1_Init 2 */
}

/**
 * Enable DMA controller clock
 */
static void MX_DMA_Init(void)
{
    /* DMA controller clock enable */
    __HAL_RCC_DMA1_CLK_ENABLE();

    /* DMA interrupt init */
    /* DMA1_Channel1_IRQHandler interrupt configuration */
    HAL_NVIC_SetPriority(DMA1_Channel1_IRQn, 0, 0);
    HAL_NVIC_EnableIRQ(DMA1_Channel1_IRQn);

}

/**
 * @brief GPIO Initialization Function
 * @param None
 * @retval None
 */
static void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStruct = {0};
/* USER CODE BEGIN MX_GPIO_Init_1 */
/* USER CODE END MX_GPIO_Init_1 */

    /* GPIO Ports Clock Enable */
    __HAL_RCC_GPIOC_CLK_ENABLE();
    __HAL_RCC_GPIOD_CLK_ENABLE();
    __HAL_RCC_GPIOA_CLK_ENABLE();
    __HAL_RCC_GPIOB_CLK_ENABLE();

    /*Configure GPIO pin Output Level */
    HAL_GPIO_WritePin(LED1_GPIO_Port, LED1_Pin, GPIO_PIN_RESET);

    /*Configure GPIO pin Output Level */
    HAL_GPIO_WritePin(GPIOB, LED1_5_Pin|LED3_0_Pin, GPIO_PIN_RESET);
}

```

```

/*Configure GPIO pin : LED1_Pin */
GPIO_InitStruct.Pin = LED1_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(LED1_GPIO_Port, &GPIO_InitStruct);

/*Configure GPIO pins : LED1_5_Pin LED3_0_Pin */
GPIO_InitStruct.Pin = LED1_5_Pin|LED3_0_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);

/*Configure GPIO pin : PB11 */
GPIO_InitStruct.Pin = GPIO_PIN_11;
GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
GPIO_InitStruct.Pull = GPIO_NOPULL;
HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);

/* USER CODE BEGIN MX_GPIO_Init_2 */
/* USER CODE END MX_GPIO_Init_2 */
}

/* USER CODE BEGIN 4 */

//-----
//      Funcao que executa a leitura dos ADCs      //
//-----//
void read_ADC()
{
    // Ler ADCs
    FLAG_READY = 0; // Zerar a FLAG de "leitura dos adcs"

    HAL_ADC_Start_DMA(&hadcl, (uint32_t*) ADC_buffer, ADC_BUF_SIZE); // Dispara a
    leitura da leitura do ADC
    //    while(FLAG_READY == 0); // Espera a flag do "leitura dos adcs" ser 1 (leitura
    realizada)

    ref_pwm = ADC_buffer[0];
    pot_adc = ADC_buffer[0];
    corrente_adc = ADC_buffer[1];
    tensao_adc = ADC_buffer[2];

    pot_malha_aberta = pot_adc*ganho_pot_malha_aberta; //multiplica o sinal lido
    pelo pino adc do potenciometro pelo seu ganho para utilizar em malha aberta
    pot_corrente = pot_adc*ganho_pot_corrente; //multiplica o sinal lido pelo pino
    adc do potenciometro pelo seu ganho para utilizar como referencia de corrente
    pot_tensao = pot_adc*ganho_pot_tensao; //multiplica o sinal lido pelo pino adc
    do potenciometro pelo seu ganho para utilizar como referencia de tensao
    corrente = corrente_adc*ganho_corrente; //multiplica o sinal lido pelo pino
    adc da corrente pelo seu ganho definido no inicio do código
    tensao = tensao_adc*ganho_tensao; //multiplica o sinal lido pelo pino adc da
    tensao pelo seu ganho definido no inicio do código
}

// Funcao de debug atraves do protocolo usart para leitura na serial dos valores dos
adcs
void print_MSG()
{
    char mensagemUart[500];
    sprintf(mensagemUart, "%4.2f %4.2f %4.2f \r\n", pot_malha_aberta, corrente,
    tensao);
}

```

```

        HAL_UART_Transmit(&huart1,           (const      uint8_t*)      mensagemUart,
strlen(mensagemUart), 1000);
}

//Função para printar uma variável qualquer na serial como debug através do protocolo
//uart
void debug_msg(float mensagem)
{
    char mensagemUart[500];
    sprintf(mensagemUart, "%4.2f \r\n", mensagem);
    HAL_UART_Transmit(&huart1,           (const      uint8_t*)      mensagemUart,
strlen(mensagemUart), 1000);
}

// 
void HAL_ADC_ConvCpltCallback(ADC_HandleTypeDef* hadc)
{
    FLAG_READY = 1;
}

//Função utilizada para controlar o pwm
void modulacao(float ref){
    TIM3->CCR2 = ref; //Atualiza o registrador do timer 3 que define o duty cycle.
                      //o valor máximo do pwm configurado é 3599, d = ref/3599.
}

void malha_aberta(){
    uint16_t aux2 = 1080;
    if(HAL_GPIO_ReadPin (GPIOB, GPIO_PIN_11)){aux2 = 1800;}
    modulacao(aux2);
    //modulacao(pot_malha_aberta); //Referencia vira o potenciômetro
}

void controle_malha_fechada(){

/*
A declaração das variáveis de controle foi feita no início
do código de forma a economizar tempo de processamento dentro
desta função

*/
//-----
// Malha de tensão
//-----
// Referência de tensão
//-----
Voref = 15.0f;
//Voref = pot_tensão; //A referência vira o potenciômetro

//if(HAL_GPIO_ReadPin (GPIOB, GPIO_PIN_11)){Voref = 15.0f;} //aumenta a ref de
//tensão quando aperta-se o botão presente na placa

//-----
// Conversão AD
//-----
// Leitura da tensão
Vo = tensão;

//-----
// Cálculo do erro
//-----
evk0 = Voref - Vo;

```

```

//debug_msg(evk0);

//-----//
// Atualizaçao da ação de controle      //
//-----//
uvk0 = uvk1 + (av1*evk0) + (av2*evk1);

//-----//
// Rotaçao das variaveis de tensao      //
//-----//
evk1 = evk0;
uvk1 = uvk0;

//-----//
// Malha de corrente                  //
//-----//
// Referencia de corrente            //
//-----//
iLref = uvk0;
//iLref = 0.340f;
//iLref = pot_corrente; // A referencia vira o potenciometro
//if(HAL_GPIO_ReadPin (GPIOB, GPIO_PIN_11)){iLref = 0.680f;} //aumenta a
referencia de corrente quando aperta-se o botao presente na placa

//-----//
// Conversao AD                      //
//-----//
// Leitura da corrente
iL = corrente;
//-----//
// Calculo do erro                   //
//-----//
eik0 = iLref - iL;
//debug_msg(eik0);

//-----//
// Atualizaçao da ação de controle      //
//-----//
uik0 = uik1 + (ail*eik0) + (ai2*eik1);

//Limitar a acao de controle de corrente por seguranca
//if(uik0 > 2520.0f){uik0=2520.0f;}
//if(uik0 < 10.0f){uik0=10.0f;}

//-----//
// Rotaçao das variaveis de corrent      //
//-----//
eik1 = eik0;
uik1 = uik0;

modulacao(uik0);
}

// AQUI EH A FUNCAO CHAMADA NA INTERRUPCAO DO TIMER3
//      |||||||||
//      VVVVVVVVVVVV

void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef* htim)
{
    HAL_GPIO_TogglePin(GPIOB, GPIO_PIN_1);
    //HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, 1);

//-----//
//Leitura dos ADCs e controle//

```

```

//-----
read_ADC(); // Funcao que executa a leitura dos adcs
malha_aberta(); // Funcao que faz o conversor funcionar em malha aberta
//controle_malha_fechada(); // Funcao que faz o conversor funcionar em malha
fechada
//-----

//Funcoes que permitem o debug atraves da porta serial//
//debug_msg(pot);
//print_MSG();
//-----//


// Uma pista visual que o botao foi acionado e que que a referencia do controle
foi alterada//
if(HAL_GPIO_ReadPin (GPIOB, GPIO_PIN_11))
{
    // Aciona o LED quando o botao da placa de instrumentacao é acionado
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_SET);
}
else
{
    // Desliga o LED quando o botao da instrumentacao é desacionado
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_0, GPIO_PIN_RESET);
}
//-----//


//HAL_GPIO_WritePin(GPIOB, GPIO_PIN_1, 0);
}
/* USER CODE END 4 */

/***
 * @brief This function is executed in case of error occurrence.
 * @retval None
 */
void Error_Handler(void)
{
/* USER CODE BEGIN Error_Handler_Debug */
/* User can add his own implementation to report the HAL error return state */
__disable_irq();
while (1)
{
    /*
        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_13, 1);
        HAL_Delay(500);
        HAL_GPIO_WritePin(GPIOC, GPIO_PIN_13, 0);
        HAL_Delay(500);*/
}
/* USER CODE END Error_Handler_Debug */
}

#ifndef USE_FULL_ASSERT
/***
 * @brief Reports the name of the source file and the source line number
 * where the assert_param error has occurred.
 * @param file: pointer to the source file name
 * @param line: assert_param error line source number
 * @retval None
 */
void assert_failed(uint8_t *file, uint32_t line)
{
/* USER CODE BEGIN 6 */
/* User can add his own implementation to report the file name and line number,
   ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */
/* USER CODE END 6 */
}
#endif /* USE_FULL_ASSERT */

```