



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

NEI OLIVEIRA DE SOUZA

**INVERSOR DE TENSÃO COM CONTROLADOR DE AÇÃO
REPETITIVA E “PD-FEEDFORWARD” EMBARCADO EM FPGA**

**UBERLÂNDIA
2018**

NEI OLIVEIRA DE SOUZA

**INVERSOR DE TENSÃO COM CONTROLADOR DE AÇÃO
REPETITIVA E “PD-FEEDFORWARD” EMBARCADO EM FPGA**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial à obtenção do título de Mestre em Ciências.

Orientador: Prof. Dr. Ernane Antônio Alves Coelho

**UBERLÂNDIA
2018**

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

S729i
2018

Souza, Nei Oliveira de, 1965-
Inversor de tensão com controlador de ação repetitiva e “PD-FEEDFORWARD” embarcado em FPGA / Nei Oliveira de Souza. - 2018.

91 f. : il.

Orientador: Ernane Antônio Alves Coelho.
Dissertação (mestrado) - Universidade Federal de Uberlândia,
Programa de Pós-Graduação em Engenharia Elétrica.
Disponível em: <http://dx.doi.org/10.14393/ufu.di.2018.285>
Inclui bibliografia.

1. Engenharia elétrica - Teses. 2. Field programmable gate arrays - Teses. 3. Inversores elétricos - Teses. I. Coelho, Ernane Antônio Alves. II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

CDU: 621.3

Maria Salete de Freitas Pinheiro – CRB6/1262

NEI OLIVEIRA DE SOUZA

INVERSOR DE TENSÃO COM CONTROLADOR DE AÇÃO REPETITIVA E “PD-FEEDFORWARD” EMBARCADO EM FPGA

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial à obtenção do título de Mestre em Ciências.

Uberlândia, 02 de março de 2018.

BANCA EXAMINADORA

Prof. Dr. Ernane Antônio Alves Coelho – UFU
Orientador

Prof. Dr. Henrique José Avelar – CEFET MG
Banca externo

Prof. Dr. Gustavo Brito de Lima – UFU
Banca interno

Prof. Dr. Igor Santos Peretta – UFU
Banca interno

Dedicatória

Aos que torcem por mim em todos esses anos – minha família: Adriana, Nei Jr., Mariana e Ana Carolina – e que sofrem as consequências dos meus sucessos ou fracassos.

À minha mãe, por ter me possibilitado a existência e me permitido estudar, com toda a dificuldade que os despossuídos têm.

Aos meus irmãos Otaniel e Auta, e in memoriam ao Neiri, Rosângela e Júlio Souza.

Aos primos Paulo César Firmino e Carlos Antônio de Oliveira.

Aos amigos in memoriam Maria Aparecida Borges Riemslag e Wilhelmus Antonius Maria Riemslag.

Aos colegas de escola que não tiveram como terminar esta jornada da vida e ficaram pelo caminho, não porque morreram, mas por falta de oportunidade. Neste modo como vivemos, a chance de chegar à faculdade é definida pela condição socioeconômica. Assim, os mais abastados serão “doutores” e os outros, tudo, menos doutores. Claro, existem exceções de ambos os lados. Mas a regra é geral. Assim, dedico aos colegas dos primeiros anos de escola e do ensino noturno, que não conheço nenhum que chegou a terminar o ensino fundamental. Mas, neste conjunto, os tenho encontrado pela vida nas mais diversas profissões: Vendedor do Triângulo da Sorte, vendedor de Consórcio Primo Rossi, Eletricista de autos, Faxineiras, Segurança, Pizzaiolo, Feirante, Pintor, Carteiro, Telefonista. E os que morreram: o Luiz, alcoólatra; Gilmar, drogas; Alemão, drogas...

VIVA A MERITOCRACIA!

Agradecimentos

Agradeço a todos que, direta ou indiretamente, contribuiriam para a realização desta pesquisa: das faxineiras, que me propiciaram um ambiente limpo, aos doutores, com quem troquei ideias e que me incentivaram. Provavelmente devo esquecer alguém, já que são muitos e ninguém faz nada sozinho; que me perdoem os esquecidos.

Agradeço em especial a minha esposa Adriana, pelo companheirismo ao longo dos anos, a minhas filhas Mariana e Ana Carolina e meu filho Nei Júnior, e à Conceição Leal, pela compreensão e apoio.

Agradeço à Faculdade de Engenharia Elétrica, na pessoa de seus diretores, pela oportunidade que me foi dada ao longo dos anos, me possibilitando fazer o curso de Engenharia Elétrica e o mestrado. Dentre os diretores, não posso deixar de destacar dois: Renato Alves Pereira, que me deu apoio no curso de graduação, e Marcelo Lynce Ribeiro, que me deu apoio na pós-graduação.

Agradeço aos professores, são todos merecedores, mas em especial ao Prof. João Batista Vieira Júnior, pelo apoio e por acreditar em mim, e ao Prof. Ernane Antônio Alves Coelho, por aceitar ser meu orientador por duas vezes, por me ajudar a superar este grande desafio, por ser humilde, justo e amigo.

Ao amigo Edgard Afonso Lamunier, por me acompanhar ao longo desses anos, me orientado em todas as dificuldades.

Aos técnicos Carlinho, Rubinho, Hélio, Rafael, Hermano e Claudemir.

Às secretárias, em especial à Marcília.

Ao NUPEP, pelo apoio, e a todos os colegas e amigos da pós-graduação, especialmente a Eric Chaves, Fernando Rocha, Henrique Carvalho e Leandro Vilefort.

Perguntas de um operário letrado

Quem construiu Tebas, a das sete portas?
Nos livros vem o nome dos reis,
Mas foram os reis que transportaram as pedras?
Babilónia, tantas vezes destruída,
Quem outras tantas a reconstruiu? Em que casas
Da Lima Dourada moravam seus obreiros?
No dia em que ficou pronta a Muralha da China para onde
Foram os seus pedreiros? A grande Roma
Está cheia de arcos de triunfo. Quem os ergueu? Sobre quem
Triunfaram os Césares? A tão cantada Bizâncio
Só tinha palácios
Para os seus habitantes? Até a legendária Atlântida
Na noite em que o mar a engoliu
Viu afogados gritar por seus escravos.

O jovem Alexandre conquistou as Índias
Sozinho?
César venceu os gauleses.
Nem sequer tinha um cozinheiro ao seu serviço?
Quando a sua armada se afundou Filipe de Espanha
Chorou. E ninguém mais?
Frederico II ganhou a guerra dos sete anos
Quem mais a ganhou?

Em cada página uma vitória.
Quem cozinhava os festins?
Em cada década um grande homem.
Quem pagava as despesas?

Tantas histórias
Quantas perguntas

Bertolt Brecht

Resumo

A pesquisa desenvolvida é sobre a aplicação do uso da FPGA na área de Eletrônica de Potência. Desenvolveu-se um controle para inversor de tensão CC/CA monofásico com filtro de saída LC controlado por PWM senoidal unipolar, tendo como carga um retificador não controlado. O controle implementado, o qual integra as ações de controle repetitiva e PD-Feedforward, foi todo embarcado em FPGA, assim como as funções de geração de tensão de referência, geração de portadora triangular e PWM digital, comparador digital para proteção e operações matemáticas em ponto fixo do tipo multiplica e acumula para as funções digitais. Uma parte das funções foi implementada por meio de *software* com alto nível de abstração, e outra parte foi implementada em baixo nível, através de linguagem de descrição de *hardware*. A pesquisa visa mostrar a viabilidade da aplicação das FPGAs em Eletrônica de Potência, como uma alternativa aos DSPs, para o controle de conversores, realizando o cálculo das leis de controle em ponto fixo, e não somente para a substituição de circuitos digitais. A pesquisa é inédita para o NUPEP (Núcleo de Pesquisa em Eletrônica de Potência) da Faculdade de Engenharia Elétrica da Universidade Federal de Uberlândia. Portanto é uma pesquisa de início da utilização de FPGA na área de Eletrônica de Potência para o NUPEP.

Palavras-chave: FPGA. Controle Repetitivo. Embarcado. System Generator.

Abstract

The research developed herein is based on the applied use of a FPGA to the area of Power Electronics. The authors developed a single-phase DC/CA inverter control with an LC output filter controlled by a unipolar sine wave PWM, which uses as its load a non-controlled rectifier. The implemented control, which integrates the repetitive control actions and PD-Feedforward, was embedded in FPGA, along with the functions of reference voltage generation, triangular carrier generation and digital PWM, digital comparison for protection and mathematical operations in multiple type fixed point and accumulations for digital functions. A part of these functions was implemented through the software using high level abstraction, and the other part was implemented using low level abstraction through the hardware description language. This research study aims at showing the feasibility of applying FPGAs in Power Electronics, as an alternative to DSPs for converter control by performing the calculation for the control laws in fixed point, and not only for the substitution of digital circuits. This is a novel research project here at NUPEP (Núcleo de Pesquisa em Eletrônica de Potência) (Power Electronics research center)) at the electrical engineering faculty of the Federal University of Uberlandia. Therefore, this is an initial research project into the use of FPGA in the area of Power Electronics at NUPEP.

Keywords: FPGA. Repetitive control. Embedded. System Generator.

Lista de Figuras

Figura 1 - Exemplo dos primeiros circuitos programáveis	18
Figura 2 - Conceito de FPGA [10]	19
Figura 3 – Placa FPGA utilizada	23
Figura 4 – Topologia ponte completa	26
Figura 5 – Formas de ondas da modulação senoidal PWM	27
Figura 6 – Formas de ondas de um SPWM Bipolar	28
Figura 7 - Formas de ondas de um SPWM Unipolar	29
Figura 8 - Topologia do controle PID	30
Figura 9 - Princípio do controle Repetitivo	31
Figura 10 - Controle Repetitivo Tradicional	32
Figura 11 - Sistema realimentado	33
Figura 12 - Controlador Feedforward	34
Figura 13 - Diagrama de controle do inversor com ação de controle instantânea e repetitiva [8]	35
Figura 14 - Diagrama de Bode da função $G_c(z)$ para $K_1=-0,175$ e $K_2=-0,011$ e $f_s=5000\text{Hz}$	36
Figura 15 - Transitório de elevação de carga linear	38
Figura 16 - Transitório de redução de carga linear	38
Figura 17 - Desempenho do controlador PD-feedforward para uma carga não linear	39
Figura 18 - Desempenho do controlador PD-feedforward + Repetitivo para uma carga não linear	41
Figura 19 - Gateways	43
Figura 20 - Gateway In	44
Figura 21 - Exemplos de primitivas do System Generator	45
Figura 22 - Exemplo implementado com o bloco Mcode	47
Figura 23 - Exemplo do uso do Black box para implementar a função "sample-and- hold"	47
Figura 24 - Simulação PSIM Controle Repetitivo	49
Figura 25 - Sinal amostrado na saída do controlador <i>PD-feedforward</i> no System generator e Psim	50
Figura 26 - Projeto do controlador Repetitivo e <i>PD-feedforward</i> Simulado	51
Figura 27 - Circuito gerador do sinal de referência	52
Figura 28 - Circuito gerado da forma de onda triangular	53
Figura 29 - Função de transferência do Repetitivo	53
Figura 30 - Equação de diferenças do <i>PD-feedforward</i>	54
Figura 31 - Circuito PWM	54
Figura 32 - Circuito de atraso para aquisição	56
Figura 33 - Imagem da planta	57
Figura 34 - Diagrama de ligação da planta	58

Figura 35 - Placa de aquisição	58
Figura 36 - Estágio de regulação e filtragem.....	59
Figura 37 - Estágio de regulagem: Divisor de tensão gerador do sinal de referência para compensação de <i>offset</i>	60
Figura 38 - Filtro anti-aliasing	60
Figura 39 - Estágio somador e gerado de offset	61
Figura 40 - Placa de adequação de tensão entre FPGA e Gate Driver, conectada. .	62
Figura 41 - Esquema da ligação da carga.....	63
Figura 42 - Retificador não controlado e capacitor de filtro	63
Figura 43 - Resistência de Carga.....	64
Figura 44 - Filtro LC	65
Figura 45 - Módulo SemiKron.....	66
Figura 46 - FPGA e conexão com AD e saída Gate drivers	67
Figura 47 - Sinais de entrada na função Feed Forward no PSIM e System Generator	68
Figura 48 - Sinais de saída na função Feed Forward no PSIM e System Generator	69
Figura 49 - Resposta ao Degrau da função de transferência Feed Forward no PSIM	70
Figura 50 - Resposta ao Degrau da função de transferência Feed Forward no System Generator	70
Figura 51 - Resposta ao degrau no PSIM para a função de transferência Repetitiva	71
Figura 52 - Resposta ao degrau no System Generator para a função de transferência Repetitiva	71
Figura 53 - Resultado da simulação do controle Repetitivo e PD-feedforward no PSIM com tensão de referência e de saída.....	72
Figura 54 - Resultado da simulação no PSIM do controle Repetitivo e <i>PD-feedforward</i> com o sinal de tensão e corrente de saída	72
Figura 55 - Resultado da referência [8] como colocar a fonte?	73
Figura 56 - Resultado da simulação do controle Repetitivo no System Generator com o sinal de referência e tensão de saída.....	74
Figura 57 - Resultado da co-simulação do controle Repetitivo com os sinais de tensão de saída e tensão de referência	75
Figura 58 - Resultado da planta em malha aberta para tensão e corrente de saída.	76
Figura 59 - Resultado em malha fechada, controle Repetitivo, tensão e corrente de saída	77
Figura 60 - Espectro harmônico da corrente	78
Figura 61 - Espectro harmônico da tensão de saída	78

Lista de Tabelas

Tabela 1 - Parâmetros da simulação Feedforward.....	37
Tabela 2 - Parâmetros dos controles Repetitivos.....	40
Tabela 3 - Parâmetros da carga.....	64
Tabela 4 - Parâmetros da Planta.....	67

Lista de Abreviaturas e Símbolos

A	Ampère, unidade de corrente elétrica
A/D	Analógico para Digital
AHDL	<i>Hardware description language</i> desenvolvida pela Altera
AO	Amplificador Operacional
ASIC	<i>Application Specific Integrated Circuit</i>
C(z)	Compensador do ramo direto controlador repetitivo
C(z^{-1})	Representação no formato DSP de C(z)
CA	Corrente alternada
CC	Corrente Continua
CC-CA	Corrente Continua-Corrente Alternada
CLB	<i>Configurable Logic Block</i>
Cr	Ganho do controlador repetitivo
d(z)	Função discreta do distúrbio inerente à planta
DCM	<i>Direct Clock Manager</i>
DSP	<i>Digital Signal Processor</i>
e(z)	Função discreta do erro de tensão
EDA	<i>Electronic Design Automation</i>
FEELT	Faculdade de Engenharia Elétrica
FPGA	<i>Field Programmable Gate Array</i>
Gc(z)	Compensador do controlador de ação instantânea
Gd(z)	Função de transferência do distúrbio
Gff(z)	Função Feedforward do controlador de ação instantânea
Gh(z)	Função de realimentação do controlador de ação instantânea
Gp(z)	Função de transferência da planta
Grp(z^{-1})	Função de transferência do controle Repetitivo (formato DSP)
HIL	<i>Hardware-in-loop</i>
Hz	Hertz, unidade de frequência
I(s)	Corrente de entrada do filtro LC
Io	Corrente de saída do filtro LC
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IGBTS	<i>Insulated Gate Bipolar Transistor</i>
IMC	<i>Internal Model control</i>
JTAG	<i>Joint Test Action Group</i>
LC	<i>Logic Cell</i>
LCD	<i>Liquid Crystal Display</i>
LED	<i>Light Emitting Diode</i>
LUTS	<i>Look-Up Tables</i>
LVTTL	<i>Low Voltage Transistor Transistor Logic</i>
MAC	<i>Multiply-and-accumulate (DSP instruction)</i>
NUPEP	Núcleo de Pesquisa em Eletrônica de Potência

P(z)	Função discreta da planta
PAL	<i>Programmable Logic Device</i>
PD-Feedforward	Proporcional Derivativo <i>Feedforward</i>
PI	Proporcional Integral, controlador
PID	Proporcional Integral derivado, controlador
PV	<i>Photovoltaic</i>
PWM	<i>Pulse Width Modulation</i>
Q(z)	Função do ramo de realimentação do controlador repetitivo
Q(z^{-1})	Representação no formato DSP de Q(z)
r(z)	Função discreta da tensão de referência
S(z)	Função discreta integrante de C(z)
SPWM	<i>Sinusoidal pulse width modulation</i>
THD	<i>Distorção Harmônico Total</i>
Ts	Período de amostragem
UFU	Universidade Federal de Uberlândia
UPS	<i>Uninterruptible Power Supplies</i>
USB	<i>Universal Serial bus</i>
V _o	Tensão de saída do filtro LC
V _{ab}	Tensão de entrada do filtro LC
V _{cr}	Tensão da portadora triangular para o PWM
V _{g1} , V _{g3}	Sinal para o gatilho das chaves IGBTs
VHDL	<i>Very Speed Integrated Circuit, Hardware Description Language</i>
VHDL	<i>VHSIC Hardware Description Language</i>)
VHSIC	<i>Very High Speed Integrated Circuit</i>
V _m	Tensão moduladora para o PWM
V _{ref}	Tensão de referência
Y(z)	Saída de planta
Z _o	Impedância de carga do inversor
Ω	Ohm, unidade de resistência elétrica

SUMÁRIO

1 INTRODUÇÃO	14
1.1 Considerações iniciais	14
1.2 Justificativa	15
1.3 Objetivo	17
2 FPGA	18
2.1 Linguagem de Descrição de Hardware (HDL).....	21
2.2 Fase de síntese para projeto utilizando FPGA.....	22
2.3 Sobre a placa de desenvolvimento utilizada no projeto	23
3 PRINCÍPIOS DOS INVERSORES DE TENSÃO.....	26
3.1 Técnica de modulação	27
3.2 Técnicas de controle	29
3.2.1 Controle PID.....	30
3.2.2 O controlador de Ação Repetitiva	31
3.2.3 Controle <i>Feedforward</i>	33
3.3 Técnicas de controle aplicadas ao Sistema Inversor de tensão	34
3.3.1 Controlador de Ação Instantânea.....	35
3.3.2 Controlador de Ação Repetitiva	39
4 METODOLOGIA	41
4.1 System Generator	42
4.2 VHDL	48
4.3 Construção dos controladores	48
4.4 Detalhes da metodologia	52
4.4.1 Funções de transferências	53
4.4.2 O circuito de ajuste do tempo de aquisição	55
5 DESCRIÇÃO DA PLANTA	57
5.1 Placa de aquisição	58
5.2 Placa de adequação dos sinais da FPGA para os Gate Drivers dos IGBTs	61
5.3 A carga.....	62
5.4 Filtro LC	64
5.5 Módulo da Semikron	65
5.6 FPGA	66

6 RESULTADOS.....	68
6.1 Resposta da Função de Transferência do Repetitivo e PD-feeforward no Sistema	68
6.2 Resposta ao Degrau da Função de Transferência do Repetitivo e PD-feeforward	69
6.3 Resultado da Simulação no PSIM.....	71
6.4 Resultado da Simulação no System Generator	74
6.5 Resultado da co-simulação no System Generator	74
6.6 Resultados experimentais do controle Repetitivo implementado	75
7 CONCLUSÃO E PROPOSTA PARA TRABALHOS FUTUROS	79
REFERÊNCIAS.....	81
APÊNDICE.....	87
APÊNDICE A - Exemplo de funções primitivas do System Generator	87
APÊNDICE B - Código implementado no bloco Black Box	88
ANEXOS	90
ANEXO A - Circuito da placa de aquisição e condicionamento de sinais	90
ANEXO B - Código implementado no bloco Mcode	91

1 INTRODUÇÃO

1.1 Considerações iniciais

O mundo passa atualmente por uma crise energética [1]. O consumo de energia elétrica é cada vez mais elevado, porque mais pessoas estão adquirindo bens que funcionam a partir da eletricidade. Com o aumento da demanda, as indústrias precisam produzir mais bens de consumo e, para isto, necessitam utilizar mais energia elétrica, que é um insumo básico para a indústria de transformação. O petróleo e o carvão mineral são utilizados como fontes de energia e são hoje os principais responsáveis pela maior parte da energia elétrica consumida no mundo. Porém, são recursos minerais e, como tal, não são renováveis [1-2]. Por isso, precisam ser poupadados.

Além da finitude desses recursos, há ainda o inconveniente grave da poluição originada pelo uso dessas fontes de energia e seus derivados. Felizmente, existem outras maneiras de se gerar energia elétrica, que são as fontes renováveis. As mais conhecidas destas fontes são a hidroeletricidade, a energia eólica e a solar. Todas estas formas de energia causam algum tipo de impacto ambiental, porém possuem a vantagem de serem renováveis, ou seja, seu aproveitamento é infinito.

A produção de energia elétrica utilizando a hidroeletricidade provoca alagamentos de grandes áreas pelos reservatórios necessários, comprometendo diversos ecossistemas. A utilização de energia eólica interfere nas rotas das aves migratórias, provoca o aquecimento do solo abaixo das pás e a ocupação de áreas muito extensas, além de possíveis modificações no regime dos ventos após a instalação do parque eólico [3]. A produção de energia solar, por sua vez, requer o uso de painéis fotovoltaicos e causa impacto ambiental com a mineração que é necessária para a obtenção dos metais insumos de sua confecção [4]. Porém, se utilizados nos lugares corretos, esses painéis podem ser uma alternativa muito interessante. A utilização mais racional dos painéis solares é sobre os telhados de residências, fábricas e edifícios. São áreas já impermeabilizadas e, portanto, com ecossistemas degradados, que podem ser aproveitadas sem necessidade de mais espaço alocado para isto.

Diante desse cenário de crise energética, brevemente descrito, é preciso buscar soluções para o problema que se apresenta como um desafio para o mundo

atual. As fontes de energias alternativas, como a eólica e a solar, são opções que podem diminuir o consumo das fontes não renováveis. Mais especificamente a fonte de energia solar, capaz de gerar energia elétrica a partir das irradiações solares, apresenta-se com uma excelente opção, principalmente para o Brasil, que tem grande incidência de raios solares durante quase o ano todo.

Os painéis solares são fontes não despacháveis. Assim, em grande parte das aplicações, os sistemas PV (Painel Voltaico) operam conectados à rede e são chamados sistemas *grid following*. Tais sistemas operam como uma fonte de corrente conectada à rede, sendo que a tensão no ponto de conexão é imposta pela rede. No caso de falta da rede, um algoritmo anti-ilhamento deve desligar o sistema, pois, além de não haver uma malha para regular a tensão, não há como garantir o fornecimento de energia à carga com uma fonte não despachável. Além disso, existem aspectos de segurança para os operadores do sistema que devem ser observados.

Em casos específicos, como nos sistemas distribuídos, ou micro-redes, há necessidade da operação ilhada para permitir o fornecimento ininterrupto de energia. Isto implica a necessidade de um nó formador de rede (*grid forming*) para a regulação da tensão na ausência da rede da concessionária. Este nó deve contemplar uma fonte primária despachável e um inversor com malha reguladora de tensão. Este pode ser, então, um inversor de tensão com controlador de ação *repetitiva* e *PD-feedforward* embarcado em FPGA (*Field Programmable Gate Array*), que é objeto da presente pesquisa.

1.2 Justificativa

O presente trabalho justifica se, de maneira geral, pelo fato de envolver um conversor CC-CA (Corrente Contínua-Corrente Alternada), que está presente em várias aplicações da área de Eletrônica de Potência. E especificamente pela necessidade de iniciar uma cultura para o uso da FPGA em Eletrônica de Potência no Núcleo de Eletrônica de Potência da Faculdade de Engenharia Elétrica da Universidade Federal de Uberlândia (FEELT-UFU), onde a maioria das implementações são realizadas com a utilização dos DSP (Digital Signal Processor). E quando ocorreu o uso de FPGA, sempre limitou-se a substituir circuitos digitais. Ademais, não há trabalhos realizados com a utilização de circuitos complexos

implementados em FPGA, como por exemplo o filtro digital. Neste sentido, este trabalho dá início à formação da cultura de utilizar a FPGA para a implementação de controle na área de Eletrônica de Potência da FEELT-UFU, e se beneficiar dessa tecnologia. Portanto, no presente trabalho são utilizadas as técnicas de controle digital para implementar um controle CC-CA todo embarcado em FPGA.

O controle digital apresenta diversas vantagens em comparação aos controles analógicos, dentre as quais destaca-se algumas:

- Solução programável: modificações, atualizações ou adaptações são realizadas via software;
- Menos sensível ao ruído ambiente;
- Suporta algoritmos avançados de controle;
- Permite as funções de controle adaptativo, não linear e *self-tuning*;
- Admite operação *sensorless*;
- Admite funções especiais: monitoramento, diagnóstico, proteção, etc;
- Capacidade de comunicação: permite a incorporação a um sistema integrado de controle;
- Capacidade de armazenamento de dados flexível [5].

Os sistemas digitais apresentam como desvantagens:

- Requer a conversão de dados (A/D – D/A);
- Métodos de análise e projeto são mais complexos;
- Taxa de amostragem e resolução podem afetar os níveis de rejeição aos distúrbios vinculados à carga;
- Os atrasos computacionais limitam a banda passante do sistema e podem afetar a estabilidade;
- Erros de quantização e truncamento podem afetar a precisão do controle;
- Durante a fase de ajuste, é difícil o acesso à variáveis intermediárias [5];

Na área de Eletrônica de Potência, o processador digital de sinais (DSP – *Digital Signal Processor*) é largamente utilizado para soluções de controle dos

sistemas digitais implementados. Esta opção está relacionada com suas características digitais, alta velocidade de processamento, versatilidade da linguagem de programação (C ou *Assembly*) e com sua característica de trabalhar com ponto flutuante, realizando operações complexas. Entretanto, mesmo com o elevado desempenho dos processadores (inclui os DSP) atuais, capazes de realizar instruções de MAC (*Multiply and accumulate*), as quais aceleram o cálculo de funções digitais, estes possuem apenas uma unidade aritmética e todos os cálculos dos controladores são realizados de forma sequencial.

Uma forma de acelerar o desempenho de um controlador é criar condições para a realização de cálculos em paralelo (simultaneamente) onde os mesmos são possíveis, ou seja, em partes do algoritmo de controle onde não há dependência sequencial de dados. Esta é justamente a vantagem das FPGAs: o processamento paralelo [6-7].

Devido à importância das FPGAs como solução para a implementação de controladores digitais em Eletrônica de Potência, enfatiza-se que este trabalho visa contribuir para a inserção da cultura de utilização das FPGAs no Laboratório de Eletrônica de Potência da FEELT-UFG, como um estudo de caso, em que todas as particularidades das etapas de implementação e seus respectivos desafios são apresentados. Foi adotado um exemplo de aplicação pesquisado na literatura, com a preocupação de focar o uso da FPGA na realização de cálculos, e não somente na implementação de circuitos lógicos.

1.3 Objetivo

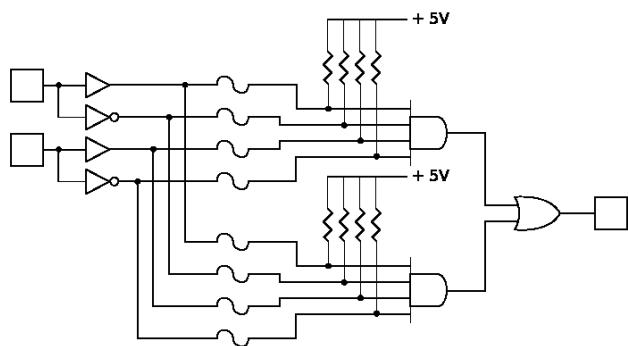
A pesquisa foi realizada com o objetivo de usar o controle digital em FPGA na área de Eletrônica de Potência. Para isso, usou-se como base principal a reprodução dos resultados do controle repetitivo da tese de Leandro Michells [8]. Desta forma, via mera comparação de resultados, foi possível a validação da implementação do controlador em FPGA.

Vale a pena ressaltar que, além das funções de controle em si, outros elementos do sistema são também embarcados em FPGA, como a geração de sinais de referência, o bloco PWM (*Pulse Width Modulation*), a comunicação serial com o conversor A/D (Analógico / Digital) e elementos de proteção (desligamento de chaves no caso de sobrecorrente ou sobretensão na tensão de saída). Ressalte-se

ainda que este será o primeiro projeto completamente embarcado em FPGA desenvolvido no Núcleo de Pesquisa em Eletrônica de Potência (NUPEP) da FEELT-UFU.

2 FPGA

A origem das FPGAs (*Field Programmable Gate Array*) está diretamente associada à evolução da eletrônica em geral, que teve seu início na válvula, depois no transistor, circuitos integrados, processadores e, finalmente, o mais próximo das FPGAs, a PAL (*Programmable Logic Device*). Mas este era um dispositivo que permitia a programação somente uma vez, como se fosse um “fusível”. A Figura 1 [9], abaixo, mostra um exemplo.



Simplified programmable logic device

Figura 1 - Exemplo dos primeiros circuitos programáveis
Fonte: Referência [9]

As PALs disponibilizam somente um conjunto de funções lógicas como portas *AND* e *OR* associadas a um conjunto de “chaves” que permitem configurar apenas funções combinacionais. Esta é uma das diferenças fundamentais em relação às FPGAs, que, além das funções combinacionais, realizam as funções sequenciais, que constituem uma das bases para a realização de qualquer circuito digital.

As FPGAs são conjuntos de circuitos digitais, organizados de uma maneira específica, distribuídos dentro de um mesmo encapsulamento, com diversas possibilidades de ligações entre todos os elementos; são formadas por circuitos com maior complexidade como LUTS (*Look-Up Tables*) e Flip Flop, multiplicadores, entre outros. Na Figura 2, a seguir, temos o conceito de FPGA.

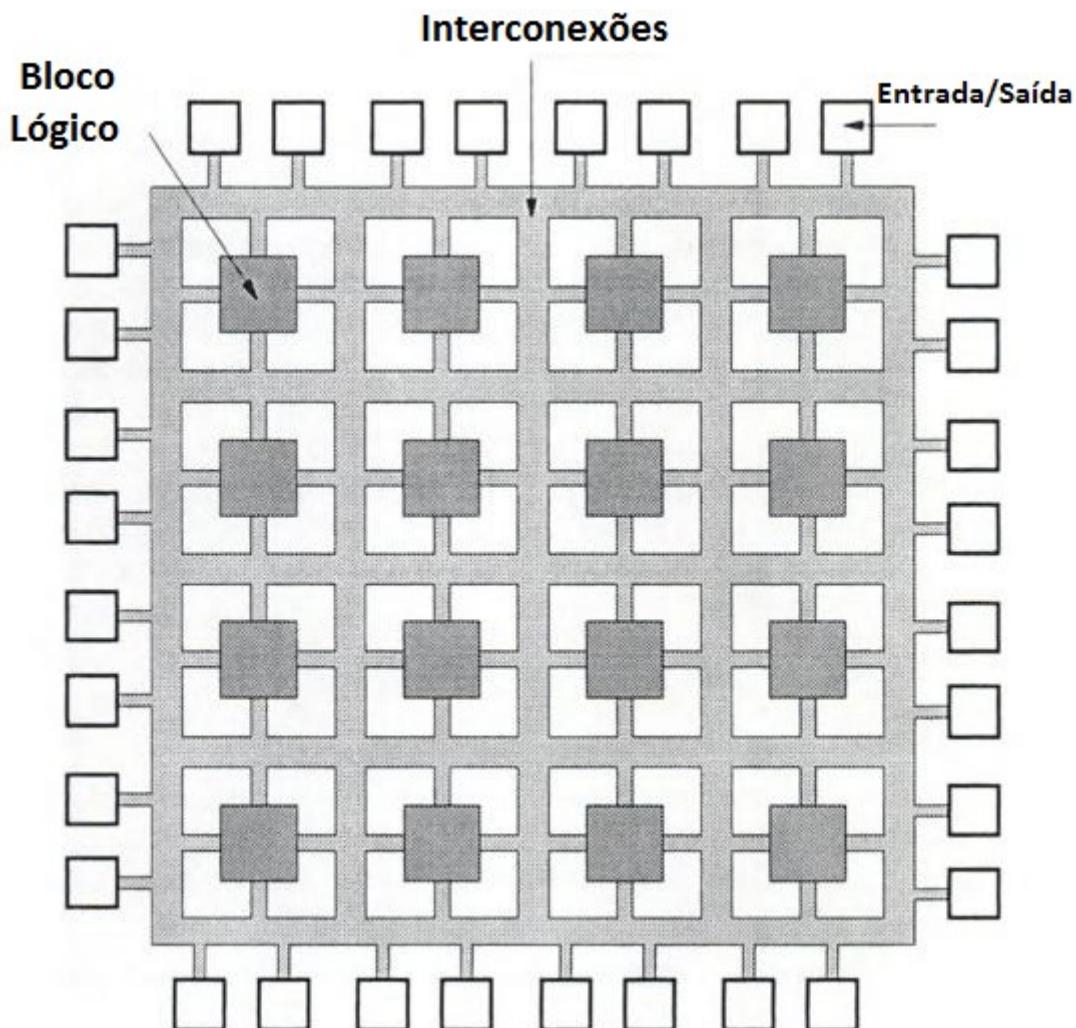


Figura 2 - Conceito de FPGA [10]

Fonte: [https://www.researchgate.net/figure/303773247_fig1_Figura-1-Arquitetura-basica-de-um-FPGA-FPGAs-utilizam-o-conceito-de-Bloco-Logico] Acesso em: 15 dez. 2017

Os conjuntos de circuitos que formam as FPGAs são organizados em blocos. As formas de organização destes blocos são determinadas pelos fabricantes. No caso da Xilinx, o bloco mais elementar é constituído por células lógicas (*Logic Cell* ou LC) formadas por: uma *LUT* de quatro entradas e um *Flip Flop* tipo D; além disso, a LC contém um circuito de transporte (*carry*) para ser utilizado em operações aritméticas; funções lógicas; circuitos de multiplexação para gerenciar o acesso e troca de informações entre as células lógicas [10].

Para aumentar a flexibilidade e melhorar o desempenho, estas células lógicas são combinadas, no caso da Xilinx, em duas e recebem o nome de *slice* (fatia em português), e quatro fatias são agrupadas para formar um Bloco Lógico

Programável (CLB – *Configurable Logic Block*). Junta-se neste processo as estruturas de multiplexação [10].

As FPGAs são formadas também por macro células (*Macro Cell*), além dos elementos citados acima. As macro células são circuitos com funções específicas, colocados junto com os elementos básicos da FPGA para completar, aumentar o desempenho e facilitar a formação dos mais diversos circuitos como: blocos de memórias; gerenciador de clock (DCM); blocos de multiplicadores; blocos de multiplexação da FPGA para entradas e saídas e, por fim, blocos de núcleos de processadores [10].

A primeira FPGA foi a XC2064, construída pela Xilinx em 1985. Era formada por apenas 64 blocos lógicos e cada bloco com duas LUTs e um registrador e um total de 800 portas. Foi produzida com um custo de 55 dólares e com tecnologia 2.0μ . Atualmente, as FPGAs da Xilinx estão sendo feitas com tecnologia de 16nm e 862K de células lógicas [11].

A grande vantagem da FPGA advém da possibilidade de implementar a maioria dos projetos digitais juntando tanto a lógica combinacional quanto a sequencial, combinando os diversos circuitos que fazem parte do conjunto da sua formação. Este fato traz grande liberdade para a implementação de projetos. Por exemplo, se um projeto precisar de somador de 32 bits e a FPGA for formada com blocos somadores de 16 bits, o software responsável pela alocação dos circuitos vai selecionar dois blocos somadores de 16 bits para obter um somador de 32bits. E se precisar de somador de 48 bits, então serão ligados três blocos de somadores de 16 bits para resultar no somador de 48 bits. O importante é que este processo pode ser repetido para todos os componentes da FPGA, registradores, memórias, LUTs, multiplicadores, etc. Vale lembrar que nos processadores atuais, não é possível alterar nenhuma característica de hardware. Isso é uma vantagem das FPGAs [10].

A capacidade de processamento paralelo das FPGAs é outra especificação que faz delas uma referência em processamento. As FPGAs são capazes de realizar o processamento em paralelo de grandes quantidades de multiplicações, somas, deslocamentos de bits, inversão de bits, etc. Essa característica faz das FPGAs um elemento fundamental para aplicações de elevado nível de processamento paralelo e alto grau de confiabilidade [10], já que tudo é realizado em hardware, sem a gerência de nenhum sistema operacional. Neste caso não há ciclo de busca de instrução, nem *Pipeline*, ou *threads*.

Outra vantagem é a facilidade de protipagem de projetos. Fazer um projeto e testar leva muito pouco tempo, se comparado ao desenvolvimento de um chip específico com técnica ASIC (*Application Specific Integrated Circuit*). Os custos dos projetos diminuem muito, tem-se uma redução da duração do ciclo de projeto, ou seja, a reconfiguração. Além do custo em constante queda, há o aumento da velocidade de processamento e da quantidade de componentes [12].

2.1 Linguagem de Descrição de Hardware (HDL)

A implementação de circuitos digitais em FPGA possui, por base, uma linguagem que descreve os componentes que serão implementados. A partir dessa linguagem é possível modelar os circuitos de implementação para uma determinada FPGA (*chip*). Existem várias linguagens para descrição de hardware: VHDL, VERILOG, AHDL (Linguagem desenvolvida pela Altera), Handel-C, SDL, ISP e ABEL [10].

As linguagens para descrição de hardware que mais se destacaram ao longo da história foram VHDL e VERILOG. Para o projeto desenvolvido, utilizou-se o VHDL (*Very Speed Integrated Circuit, Hardware Description Language*). Esta escolha tem por base o fabricante do Kit Spartan3an Xilinx, que usa esta linguagem para a síntese de projeto, e também pela padronização realizada pelo IEEE (Instituto de Engenheiros Eletricistas e Eletrônicos) em 1987 criando o padrão 1076, que se tornou um padrão internacional. Atualmente, a maioria dos ambientes de programação para HDL tem a opção de síntese utilizando o VHDL.

A VHDL possui uma grande quantidade de elementos que permitem modelar circuitos com alto nível de abstração. Abaixo são apresentadas algumas propriedades que justifica sua importância:

- Documentação: a descrição dos componentes por si já constitui uma forma de documentação para o projetista;
- Simulação: permite realizar a simulação funcional e temporal do projeto para verificar o seu correto funcionamento. Simulações que podem ser realizadas por blocos estruturais e comportamentais;
- Simplifica a migração tecnológica: o sistema pode ser facilmente sintetizado em outras tecnologias;

- Reutilização de recursos: permite a construção de bibliotecas (componentes) e módulos (circuitos) para serem reutilizados, o que permite reduzir o tempo de projeto. [10]

A linguagem VHDL possui uma estrutura básica e fundamental para o seu entendimento e utilização. A primeira é a entidade (*entity*), parte do código em que são definidas as conexões do componente com o mundo externo, as portas e os seus tipos de dados. A segunda é arquitetura (*architecture*), onde os dados que chegaram da entidade serão alterados conforme o código existente. A seguir tem-se um exemplo de código na linguagem VHDL com os dois elementos básicos e fundamentais:

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity AOI is
port (
    A, B, C, D: in STD_LOGIC;
    F : out STD_LOGIC
);
end AOI;

architecture V1 of AOI is
begin
    F <= not ((A and B) or (C and D));
end V1;

```

2.2 Fase de síntese para projeto utilizando FPGA

A geração de código para ser executado em FPGA possui várias fases, que serão apresentadas de forma resumida. Essas fases são códigos que definirão o comportamento da FPGA, a geração do *Netlist*, o processo de roteamento, a validação do mapeamento, geração do arquivo binário e configuração da FPGA.

O comportamento da FPGA é definido através da criação de um projeto, que pode ser gráfico (esquemático) ou código em HDL.

Para se conseguir o *Netlist* a partir do projeto em HDL, normalmente, usa-se uma ferramenta EDA (*Electronic Design Automation*). Esta fase descreve a conectividade entre os circuitos, pinos, portas e componentes.

O processo de roteamento é o ajuste do *Netlist* para uma determinada arquitetura FPGA.

O mapeamento é a validação do *Netlist* roteado, através de uma análise temporal e simulação.

A partir do mapeamento é gerado o arquivo de *bitstream* e, finalmente, será transferido para a FPGA para sua configuração. Normalmente, esta transferência é realizada através do protocolo JTAG [13].

2.3 Sobre a placa de desenvolvimento utilizada no projeto

A placa utilizada para o desenvolvimento deste projeto foi a Spartan3an da Xilinx. Este kit contém várias interfaces de comunicação – chaves, botões, várias opções de boot e display [14].

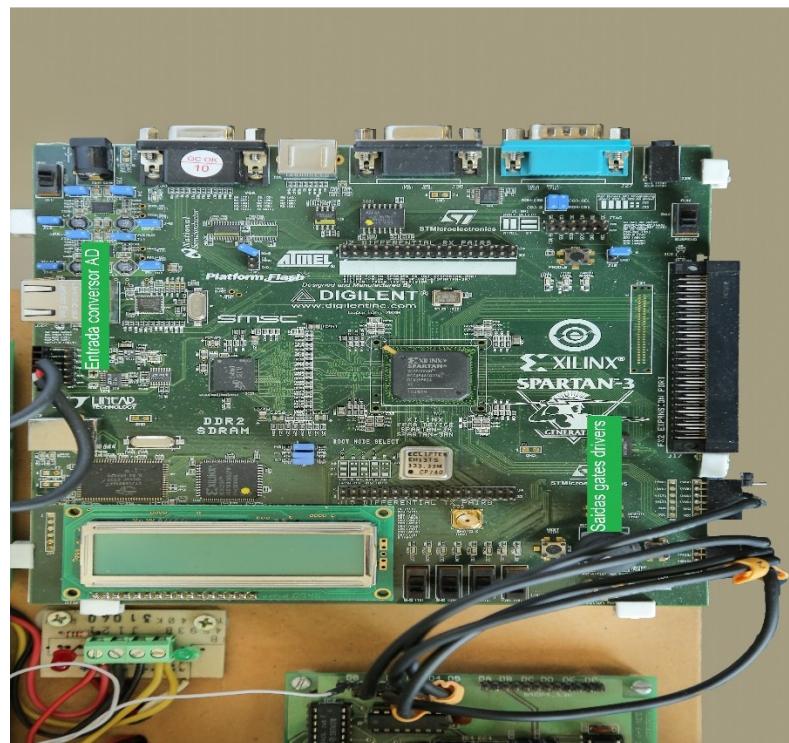


Figura 3 – Placa FPGA utilizada

Algumas características deste kit são mostradas abaixo:

FPGA

- * Spartan-3AN (XC3S700AN-FG484)

Clocks

- * Cristal oscilador de 50 MHz na placa;
- * Slot para inserção de cristal de clock opcional.

Memórias

- * 4 Mbit *Platform Flash PROM*;
- * 32M x 16bit DDR2 SDRAM;
- * 32 Mbit Flash paralela;
- * 2 dispositivos Flash SPI de 16 Mbit.

Dispositivos de interface analógica

- * Conversor D/A de quatro canais;
- * Conversor A/D de dois canais;
- * Amplificador de sinal.

Conectores e Interfaces

- * *Ethernet 10/100 PHY*;
- * Porta *JTAG USB download*;
- * Duas portas seriais RS-232 de 9 pinos;
- * Porta OS/2-style para mouse e teclado;
- * Conector VGA de 15 pinos, capaz de produzir 4096 cores;
- * Um conector de expansão FX2 de 100 pinos e dois conectores de 6 pinos;
- * 20 Entradas/saídas disponíveis em conectores padrão;
- * Stereo mini-jack para PWM áudio;
- * Chave com função rotativa e *push-button*;
- * Oito saídas LED individuais;
- * Quatro chaves *slider*, quatro chaves *push-button*.

Display

- * LCD de 16 caracteres, 2 linhas.

Para a realização do projeto foi importante o conversor AD que faz parte do kit. O conversor AD é o circuito LTC1407A-1. Este conversor possui dois canais de 14bits com saída em complemento de 2, o que proporciona uma faixa de representação que vai de -8192 a +8191, condicionado por uma faixa de tensão de entrada que abrange de 0.4V até 2.9V com o zero igual 1.65V (tensão de *offset*). A resolução para esta faixa de tensão é de aproximadamente 152×10^{-6} V. Para o projeto, um dos canais foi usado para realizar a conversão da tensão de saída analógica da planta para digital.

3 PRINCÍPIOS DOS INVERSORES DE TENSÃO

Os dispositivos que realizam a conversão de tensão contínua para tensão alternada são conhecidos como inversores de tensão. Possuem a capacidade de converter a tensão contínua para alternada, podendo gerar uma tensão de saída alternada com frequência e amplitude variáveis para diversas aplicações industriais [15] como: acionamento de máquinas elétricas de corrente alternada em velocidade variável; sistema de alimentação ininterrupta, em tensão alternada, a partir de bateria; aquecimento indutivo; UPS (*Uninterruptible Power Supplies*). Tendo como tensão de entrada fontes CC que podem ser: célula combustível; bateria; células fotovoltaicas; ou outra fonte CC.

O inversor de tensão utilizado na pesquisa é um conversor monofásico em ponte completa. O princípio de funcionamento é baseado no comando de quatro chaves (S1, S2, S3, S4), que são transistores bipolares com porta de entrada isolada (IGBT) acionadas, normalmente, por uma estratégia de controle PWM (Figura 4). A modulação utilizada será apresentada na seção 3.1 Técnica de modulação.

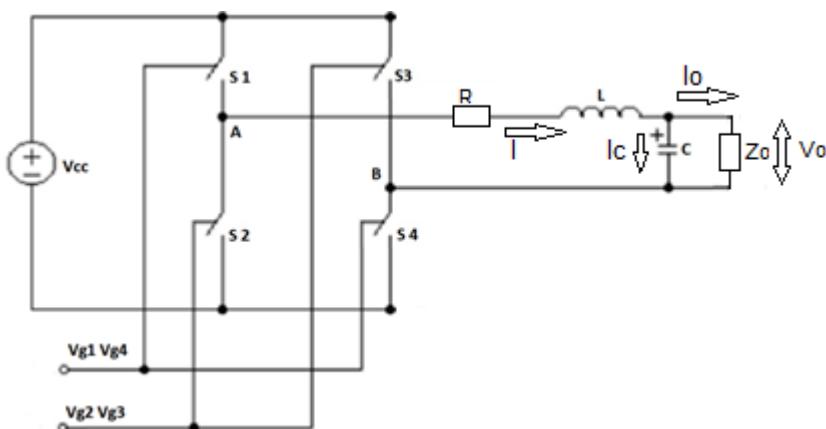


Figura 4 – Topologia ponte completa

As equações 1, 2 e 3 descrevem o comportamento do sistema do inversor, sendo que neste trabalho será considerado que a estratégia PWM apresenta um ganho unitário puro.

$$V_{ab} = RI(s) + LsI(s) + Vo(s) \quad (1)$$

$$I(s) - Io(s) = Cs \cdot V_0(s) \quad (2)$$

$$Vo(s) = Z_o(s) \cdot Io(s) \quad (3)$$

3.1 Técnica de modulação

A modulação dos sinais que controlam os IGBT (Insulated Gate Bipolar Transistor) tem como objetivo controlar a tensão de saída dos inversores para alguma aplicação. A técnica de modulação mais aplicada e eficiente é o PWM [16]. A técnica de PWM possui vários modos: modulação por largura de pulso único; modulação por largura de pulsos múltiplos; modulação por largura de pulso senoidal e modulação por largura de pulso senoidal modificada. Na pesquisa em questão foi realizada a técnica de modulação SPWM (Sinusoidal Pulse Width Modulation).

A modulação SPWM é sintetizada a partir de duas ondas, sendo uma senoidal e uma triangular. A onda triangular, a qual recebe o nome de portadora, determina a frequência de chaveamento dos IGBTs. A onda senoidal, que recebe o nome de moduladora, determina a frequência de saída e, através do índice de modulação, determina a amplitude da tensão de saída. O SPWM funciona comparando a forma de onda senoidal moduladora com a forma de onda triangular, gerando as larguras de pulsos que serão aplicados nas chaves IGBTs. As três formas de ondas necessárias ao controle dos IGBTs podem ser vistas na Figura 5.

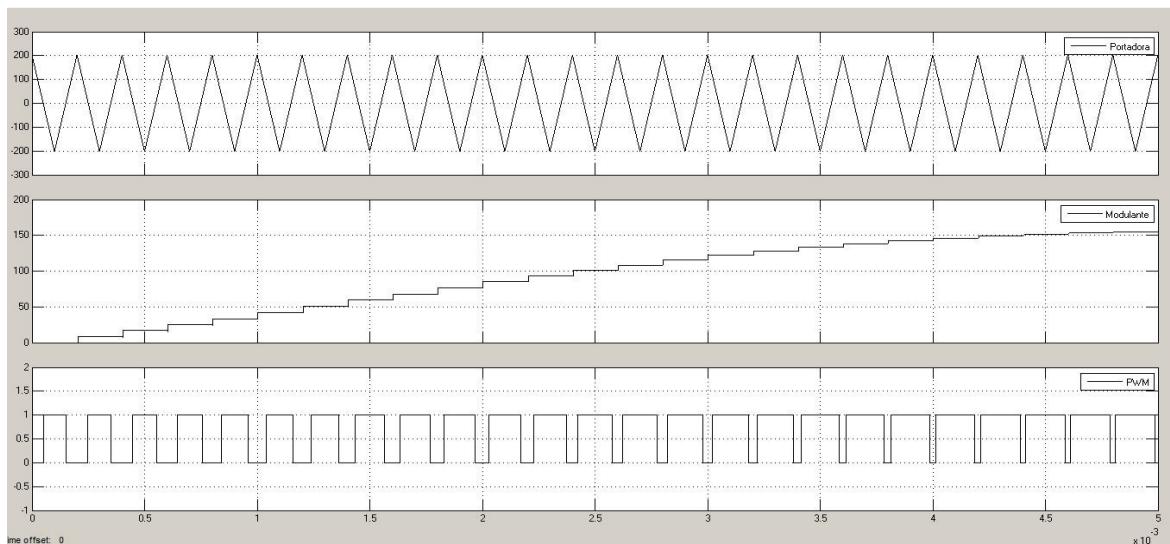


Figura 5 – Formas de ondas da modulação senoidal PWM

A estratégia de modulação PWM para inversores monofásicos pode ser Bipolar e Unipolar. Na Bipolar as chaves S1 e S3, operam de forma complementar às chaves S2 e S4. Assim, para o seu funcionamento são necessários apenas dois sinais complementares que acionam ora a chave S1, ora a chave S3. Estes sinais vêm da comparação de uma onda moduladora senoidal com uma onda triangular portadora. A tensão que será submetida ao filtro de saída será $+V_{CC}$ quando S1 estiver fechada e $-V_{CC}$ quando S3 estiver fechada. Portanto, este esquema de ligação é bipolar e caracteriza-se como estratégia Bipolar, cujos sinais são apresentados na Figura 6.

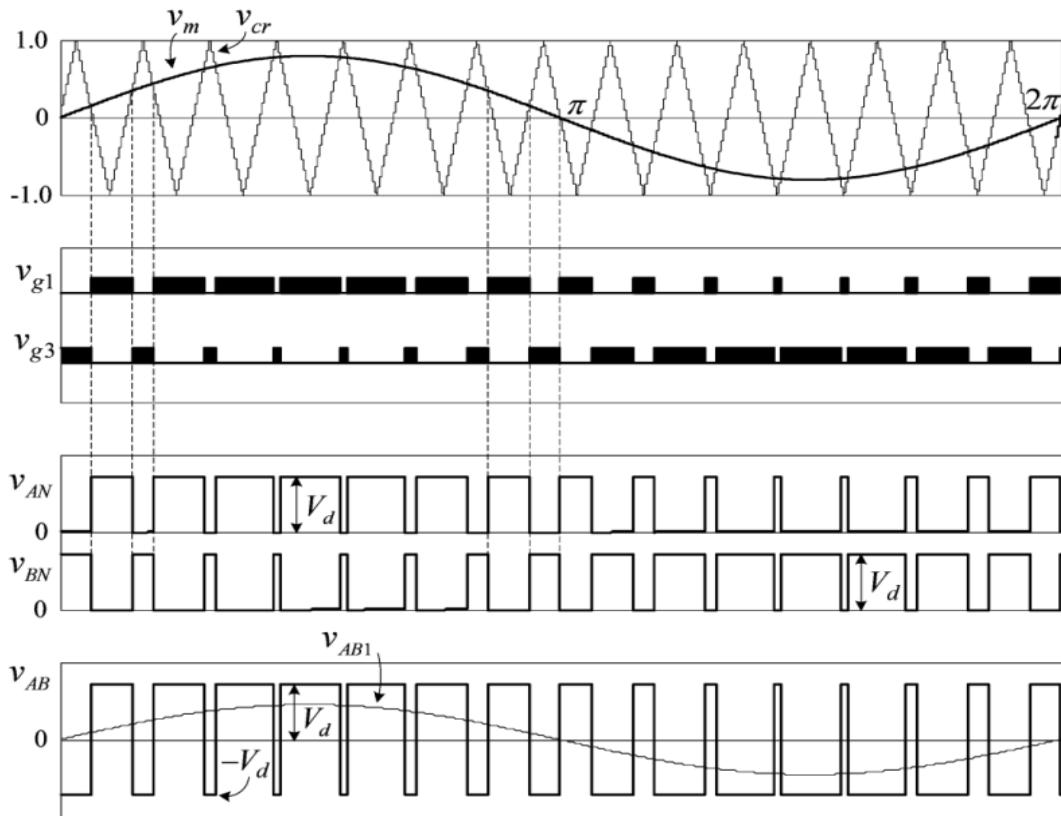


Figura 6 – Formas de ondas de um SPWM Bipolar

Fonte: Referência [17]

A estratégia Unipolar diferencia-se da Bipolar pelo fato de necessitar de duas ondas moduladoras senoidais V_m e $-V_m$ com defasagem de 180 graus para realizar o acionamento das chaves, de mesma frequência e amplitude. As duas formas de ondas são comparadas com a portadora triangular V_{cr} , gerando os sinais de controle das chaves S1 e S3 da Figura 4. Pode-se observar pela Figura 7, que

os sinais V_{g1} e V_{g3} que controlam as chaves não se alternam simultaneamente. Esta é a diferença entre o PWM Unipolar e o Bipolar, onde todas as chaves são acionadas no mesmo instante. Neste caso, a tensão de saída V_{ab} (Figura 7) alterna entre zero e $+V_{cc}$ para o semiciclo positivo e de zero e $-V_{cc}$ para o semiciclo negativo da frequência fundamental ou moduladora. Por isso, esta topologia recebe o nome de Unipolar.

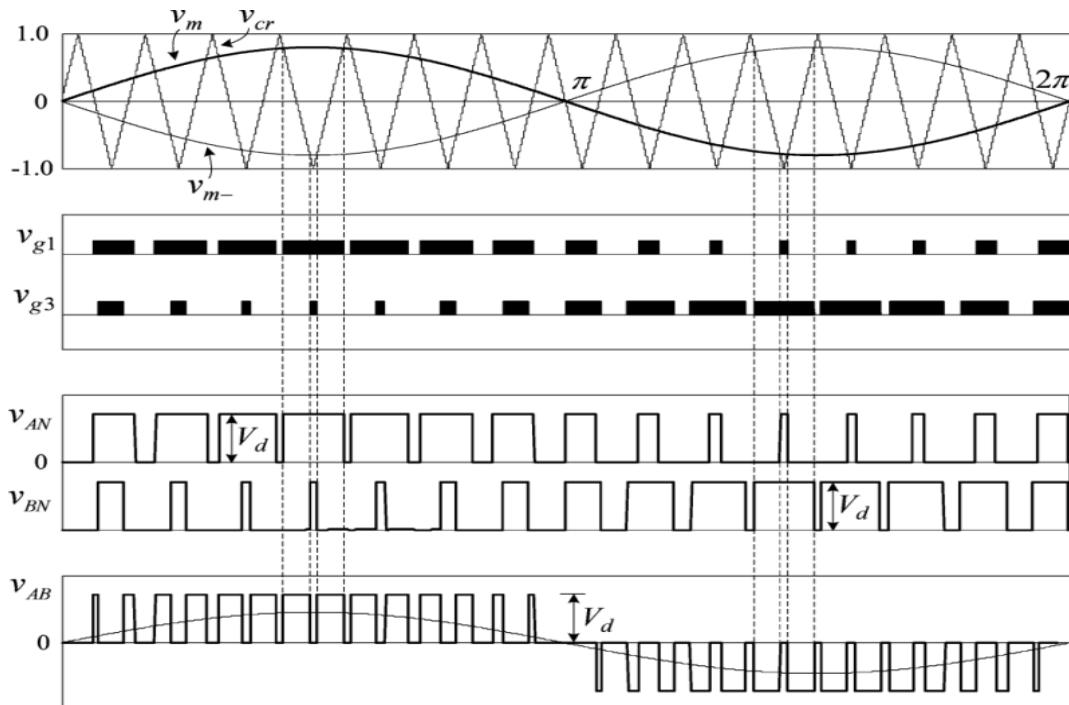


Figura 7 - Formas de ondas de um SPWM Unipolar

Fonte: Referência [17]

O SPWM Unipolar oferece vantagens como perdas de comutação reduzidas, gera menos Interferências Eletromagnéticas (EMI), redução no número de acionamento, o que diminui a geração de harmônicos maiores [17; 18]. Devido às vantagens da modulação unipolar, este tipo foi adotado no desenvolvimento da pesquisa.

3.2 Técnicas de controle

As técnicas de controle são as leis que determinam como e quando as chaves conectadas ao barramento CC irão abrir ou fechar para determinar a tensão de saída do filtro LC, especificamente a tensão no capacitor, em malha fechada.

Existem diversas técnicas para controlar as chaves, e aqui serão apresentadas algumas delas: PID, Repetitivo e Feedforward.

3.2.1 Controle PID

O controlador de ação PID é composto por três ações de controle, sendo uma proporcional, outra integral e uma última derivativa (Figura 8). A função de transferência de um controlador PID é representada pela seguinte equação:

$$H(s) = kp + \frac{Ki}{s} + kds \quad (4)$$

Onde os ganhos proporcional, integral e derivativo são representados por K_p , K_i e K_d , respectivamente. Na prática, a ação derivativa é pouco utilizada, visto que os ruídos presentes na tensão de realimentação são amplificados, dificultando a operação em regime permanente [19, p. 22].

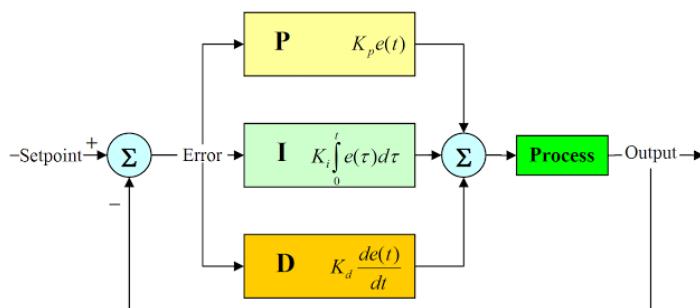


Figura 8 - Topologia do controle PID

Fonte: <https://stm32f4-discovery.net/2014/11/project-03-stm32f4xx-pid-controller/pid-controller-diagram/> em 26/01/2018

O controlador PID funciona bem para sistema com referência constante, no entanto não é capaz de produzir erro de regime nulo para referência senoidal. Mas existem autores que propõem modificações para melhorar o seu desempenho para regimes de referências periódicos, com erro nulo em regime permanente, características transitórias satisfatórias e rapidez de resposta [20; 21; 22; 23] de motores CC [24]. Contudo, para sistema de UPS, referências periódicas, controle CA, o controlador PID apresenta resultados insuficientes.

3.2.2 O controlador de Ação Repetitiva

Em razão da necessidade de gerar energia com qualidade para equipamentos que trabalham com cargas críticas (não linear), várias técnicas de controle foram propostas para atender às normas de qualidade de energia e, principalmente, para diminuir a THD (Total Harmônica de Distorção) na tensão de saída em regime permanente. Uma aplicação comum para os inversores de tensão com malha de realimentação da tensão de saída encontra-se nos sistemas UPS (*Uninterruptible Power Supply*). Estes sistemas sofrem influência da qualidade da energia fornecida pela concessionária, ficando sujeitos a todo tipo de distúrbios (variações de tipos de cargas, variações de tensão, sub-tensão e sobre-tensão, falta de rede, variação de frequência, surto de tensão ruídos, distorção harmônica). Em 1988, uma proposta para melhorar a qualidade da energia gerada pelas UPS foi apresentada pelos autores Haneyoshi, Kawamura e Hoft: O controle de Ação Repetitiva [8, p. 53].

O fundamento básico do funcionamento do controle de Ação Repetitiva é a utilização de informações dos ciclos passados para conseguir melhorar a saída atual. Este controle foi concebido a partir da necessidade de eliminação de distúrbio de carga e do rastreamento de referências periódicas. Assim, qualquer sistema que apresente distúrbios periódicos pode se beneficiar deste controle.

O controle de Ação Repetitiva tem como base o princípio do Modelo Interno [25], ou seja, “para que um sistema possua erro nulo em regime permanente na presença de qualquer referência ou distúrbio de carga, os modelos desses sinais devem estar presentes na malha fechada estável do sistema de controle” [26, p. 33]. No controle de ação repetitiva, o princípio do Modelo Interno é realizado através da realimentação positiva de um período com um atraso referente a um período T , como se vê na Figura 9.

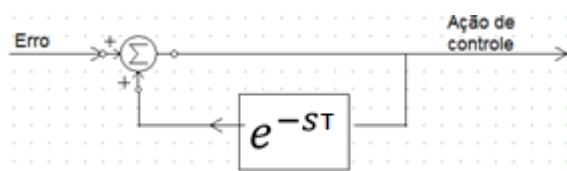


Figura 9 - Princípio do controle Repetitivo

Existe a versão do modelo externo de controle com a ação repetitiva. Para esta situação, a ação do controle Repetitivo é posicionada fora da malha de controle. Entretanto, nesta pesquisa foi realizada a implementação do modelo interno.

A versão do controle de Ação Repetitiva com o princípio do modelo interno apresenta algumas vantagens: o comportamento do algoritmo é linear e a sua taxa de convergência é mais rápida que a dos demais. Já o controle de Ação Repetitiva externo possui como vantagem o fato de que o ganho em malha fechada da planta não é significativamente modificado [8].

Ambos os modelos apresentam também algumas desvantagens. No caso do modelo externo de Ação de controle Repetitiva, sua aplicação requer um modelo paramétrico (ou matemático) da planta e possui algoritmos mais complexos que os outros controladores. No modelo da Ação de controle Repetitiva, a resposta em frequência do sistema é modificada e “a robustez ao ruído e dinâmica não modelada é normalmente reduzida” [8].

O controle Repetitivo discreto foi proposto por Tomizuka [27] e motivou o interesse de diversos pesquisadores, o que resultou em uma grande quantidade de publicações [27; 28; 29; 30; 31; 32; 33; 34; 35]. A partir de diversas publicações, este controle também ganhou espaço na área de eletrônica, passando a ser aplicado nos controles de inversores PWM.

A partir da evolução histórica do controle Repetitivo, surgiram várias versões tendo como base o controle Repetitivo que ficou conhecido como tradicional. Apresenta-se, a seguir, a descrição do controle Repetitivo tradicional [36], mostrado na Figura 10.

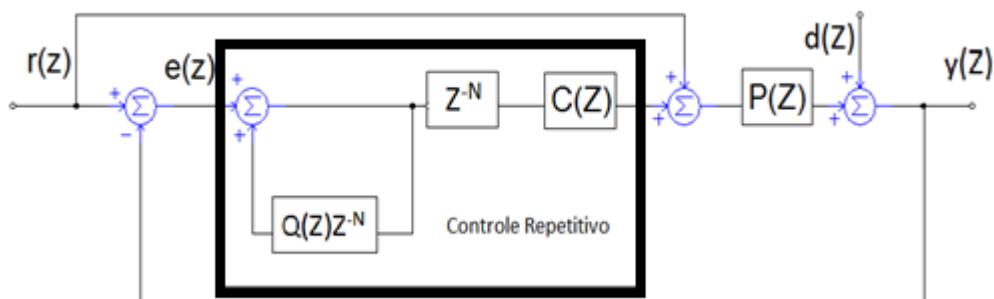


Figura 10 - Controle Repetitivo Tradicional

A saída da planta $y(z)$, sob influência do distúrbio $d(z)$, é realimentada e comparada com a referência $r(z)$, gerando o erro $e(z)$; o erro vai ser somado ao integrador $Q(z)z^{-N}$ com um ciclo de atraso, multiplicado pelo Z^{-N} com um ciclo de atraso. Para Z^{-N} o valor de $-N$ leva em conta o atraso da planta. Chegando no compensador $C(z)$ que, somado com $r(z)$, atua na planta $P(z)$.

O compensador $C(z)$ depende da planta $P(z)$ e a sua equação fica definida completamente da seguinte forma:

$$C(z) = C_r Z^{-k} S(z) \quad (5)$$

Os termos empregados na equação de $C(z)$ representam os atrasos da planta e do filtro LC e o ganho. O ganho é definido pelo C_r ; Z^k é o compensador total de fase do atraso resultante da planta $P(z)$ e de $S(z)$. E por último $S(z)$, que corresponde ao filtro de segunda ordem passa baixo, utilizado nos inversores PWM, que modifica a resposta de frequência do sistema.

3.2.3 Controle *Feedforward*

O controle *Feedforward* tem como característica principal a sua capacidade de antecipar a ação de resposta a algum distúrbio. Por isto o seu nome de *Feedforward*, alimentação adiantada, antecipatório ou método de alimentação direta. O controle *Feedforward* realiza esta antecipação de atuação na planta como apenas o sinal de referência e conhecendo o distúrbio. Para outras formas de controle ação de controle realizada passa por uma malha de realimentação, causando um atraso na resposta. A Figura 11 mostra um sistema realimentado.

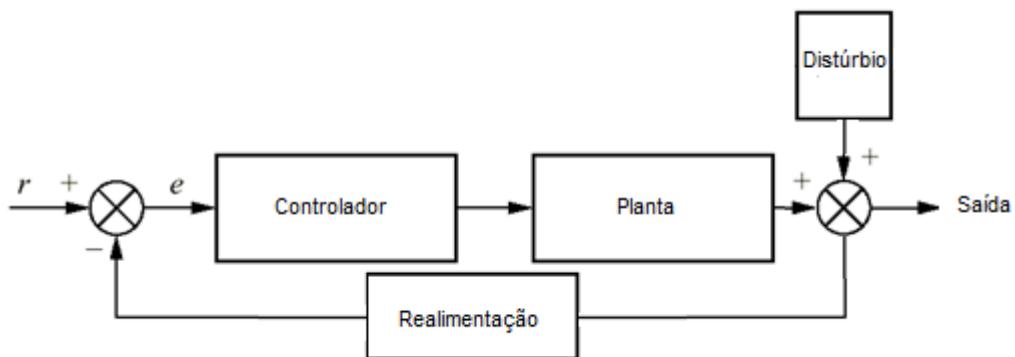


Figura 11 - Sistema realimentado

Para um sistema com o controle *Feedforward*, o sinal de referência modificado pela função de transferência do *Feedforward* é diretamente na planta, para realizar a compensação do distúrbio (Figura 12).

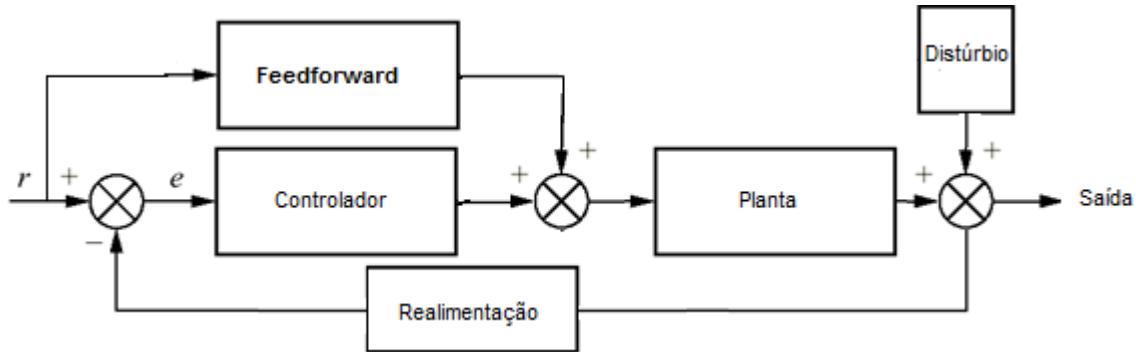


Figura 12 - Controlador Feedforward

O controle *Feedforward* não é implementado independente; para a sua atuação é necessária a existência de outro controlador para rastrear o sinal de referência. O controle que vai atuar junto com o *Feedforward* depende da planta e opções de projeto [37]. Está evidente nas Figuras 11 e 12 a existência de um controlador além do *Feedforward*.

O controle *Feedforward* implementado foi o mesmo usado em [8]: $G_p(z^1) = k_1 z^{-1} + k_1 z^{-2}$, $G_{ff}(z^{-1}) = 1$, $G_h(z^{-1}) = 1$, para $K_1=-0,175$ e $k_2=-0,011$. Este controlador Feedforward tem como coadjuvante o controle proporcional derivativo, PD-feedforward. O sistema de controle é formado pelas ações de controle Repetitiva, Proporcional Derivativa e *Feedforward*. Conforme apresentado no controle PID, tem-se três ações de controle que, por analogia, estão representadas na implementação. Os controles de ação Proporcional Derivativa e *Feedforward* representam analogicamente a ação proporcional e derivativa semelhante ao PID e a ação integral é representada pela ação repetitiva.

3.3 Técnicas de controle aplicadas ao Sistema Inversor de tensão

O controle da tensão de saída do inversor é realizado por dois controladores digitais, o primeiro de ação instantânea, e o segundo de ação repetitiva auxiliar, como mostrado na Figura 13. O objetivo do controle é fazer com que a tensão de

saída V_o acompanha a referência V_{ref} , rejeitando a ação do distúrbio D_s existente na saída da planta. É importante frisar que o controlador de ação instantânea é o elemento principal do controle, o qual tem a função de rejeitar quaisquer distúrbios provocados pela carga ou elo CC do inversor, de forma a seguir a referência. O controlador de ação repetitiva é um coadjuvante, que auxilia o controlador principal seguir a referência rejeitando distúrbios repetitivos como, por exemplo, aqueles gerados por um retificar com filtro capacitivo utilizado como carga.

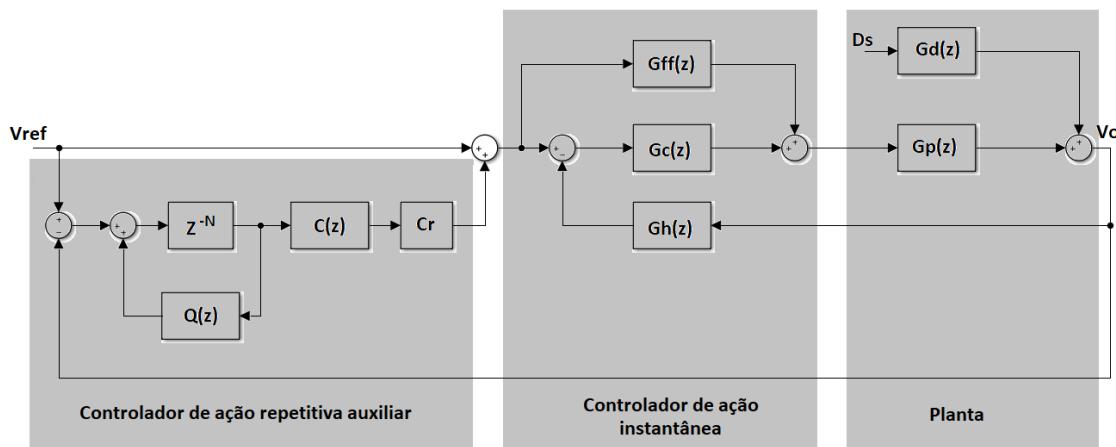


Figura 13 - Diagrama de controle do inversor com ação de controle instantânea e repetitiva [8]

Como o objetivo principal do presente trabalho foi a implementação do controle embarcado em FPGA, não foi considerado o desenvolvimento ou mesmo o aperfeiçoamento do controlador visando a uma melhor performance do sistema. Assim, as funções e parâmetros dos controladores utilizados foram os mesmos apresentados em [8]: $G_p(z^1) = k_1 z^{-1} + k_2 z^{-2}$, $G_{ff}(z^{-1}) = 1$, $G_h(z^{-1}) = 1$, para $k_1=-0,175$ e $k_2=-0,011$, buffer 100 e $C_r = 0,2$.

3.3.1 Controlador de Ação Instantânea

Existem vários tipos de funções que podem ser aplicadas aos controladores de ação instantânea. Como observado na Figura 13, o controlador de ação instantânea apresenta uma função de realimentação do erro $G_h(z)$, uma função de compensação $G_c(z)$ e uma ação de controle *feedforward* representada pela função

$G_{ff}(z)$. Conforme dito anteriormente, foram adotadas as funções apresentadas em [8] para o controlador de ação instantânea:

$$G_c(z^{-1}) = K_1 z^{-1} + K_2 z^{-2} \quad (6)$$

$$G_{ff}(z^{-1}) = 1 \quad (7)$$

$$G_h(z^{-1}) = 1 \quad (8)$$

O ganho do sistema de condicionamento de sinais e do sensor de tensão aplicados na aquisição do sinal da tensão de saída do inversor foi compensado por um ganho interno à FPGA, de modo que a função de realimentação $G_h(z)$ apresentasse ganho unitário. O controlador *Feedforward* também foi configurado com um ganho unitário puro.

O diagrama de Bode da função $G_c(z)$, considerando $K_1=-0,175$ e $K_2=-0,011$ [8] e $fs=5000\text{Hz}$, pode ser visto na Figura 14, onde observa-se que a função apresenta um avanço de fase para componentes de baixa frequência. Desta forma, o controlador de ação instantânea é dito proporcional-derivativo “*Feedforward*” (PD-feedforward) [8].

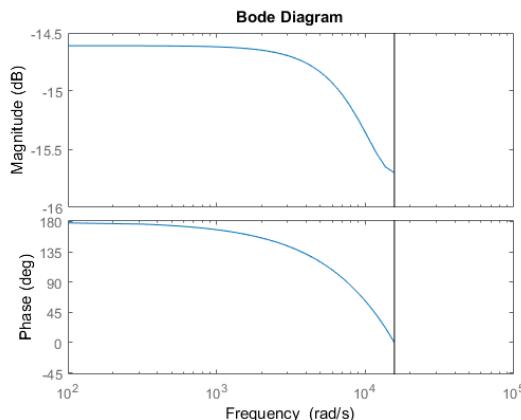


Figura 14 - Diagrama de Bode da função $G_c(z)$ para $K_1=-0,175$ e $K_2=-0,011$ e $fs=5000\text{Hz}$

O controlador PD-feedforward é responsável por rejeitar os distúrbios de carga, mantendo a tensão de saída do inversor atracada com a referência. Para avaliar o desempenho do controlador PD-feedforward, foram realizadas algumas simulações, cujos parâmetros estão na Tabela 1.

Tabela 1 - Parâmetros da simulação Feedforward

Parâmetros da Planta	
Barramento CC	VCC=200V
Tensão de referência	Vref =110 Vrms, f = 50 Hz
Indutor do filtro de saída	L=1mH
Capacitor do filtro de saída	C=30µF
Frequência de amostragem	5000 Hz
Carga linear	
Carga linear 1	40Ω
Carga linear 2	20Ω
Carga não-linear	
Retificador não controlado	Ponte de diodos RL=94Ω CL=470µF
Parâmetros do controle	
fs	5000hz
K1	-0,175
K2	-0.011

A Figura 15 apresenta a tensão de referência, a tensão de saída e a corrente de saída do inversor para um transitório de carga, onde a carga linear 1 está conectada ao inversor e em $t=205$ segundos a carga linear 2 é adicionada. Nota-se que a elevação de carga provoca uma queda significativa da tensão de saída no instante de conexão, no entanto o controlador PD-feedforward realiza a regulação da tensão de forma significativa, resultando em um pequeno desvio de amplitude e fase em relação à referência.

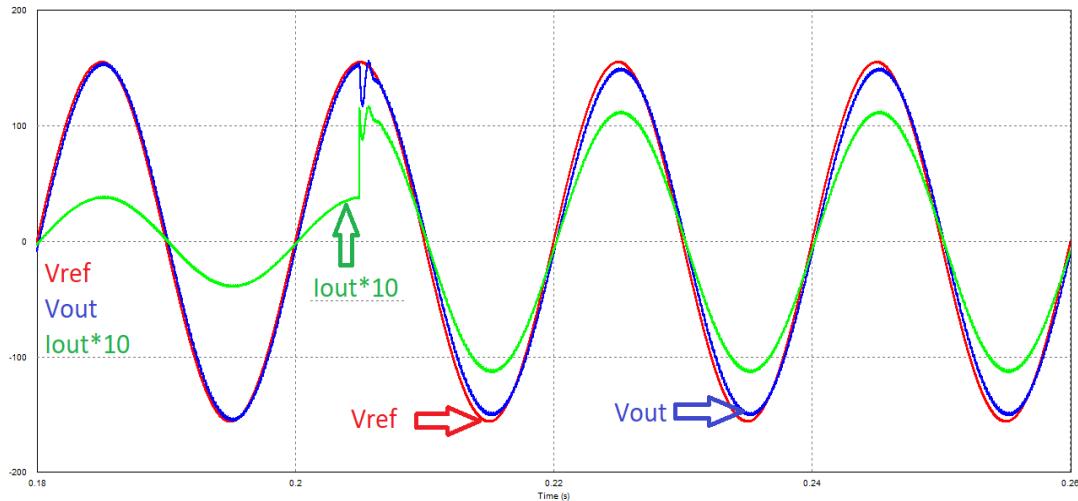


Figura 15 - Transitório de elevação de carga linear

Um transitório de redução de carga é mostrado na Figura 16, onde a carga linear 2 é desconectada do inversor. A retirada a carga linear 2 em $t=0.285$ segundos provoca uma elevação súbita da tensão de saída do inversor, a qual é corrigida prontamente pelo controlador PD-feedforward.

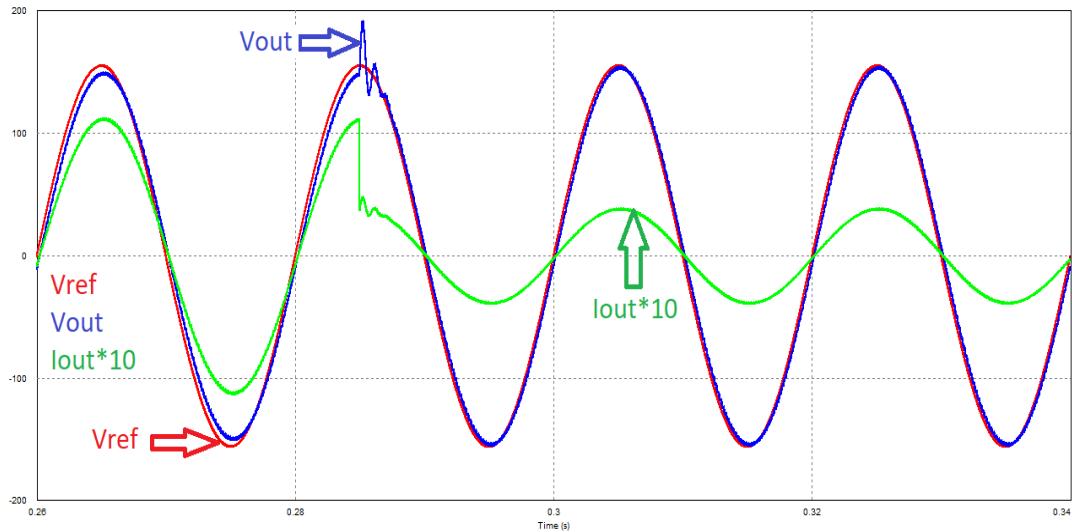


Figura 16 - Transitório de redução de carga linear

Apesar dos resultados satisfatórios para os transitórios mostrados nas Figuras 15 e 16, o controlador PD-feedforward não é capaz de seguir a tensão de referência para determinados tipos de carga, principalmente para as cargas não lineares. A Figura 17 mostra o desempenho do controlador PD-feedforward para a carga não linear apresentada na Tabela 1. A corrente pulsada requisitada pelo

retificador monofásico não controlado representa uma situação transitória que se repete a cada ciclo de operação da tensão. Nesta situação, a tensão resultante na saída do inversor apresenta uma THD de 7% para uma corrente de carga com 74% de distorção harmônica total. Desta forma, o inversor não atende à norma IEC 62040-3, onde, por exemplo, tem-se os limites de 5%, 6%, 5% e 3,5% para os harmônicos de ordem 3, 5, 7 e 11, respectivamente. É justamente para resolver este problema que recorre-se ao controlador de ação repetitiva, o qual trabalhará em auxílio ao controlador PD-feedforward rejeitando estes distúrbios da carga, os quais são repetitivos. O controlador de ação repetitiva será apresentado na próxima seção.

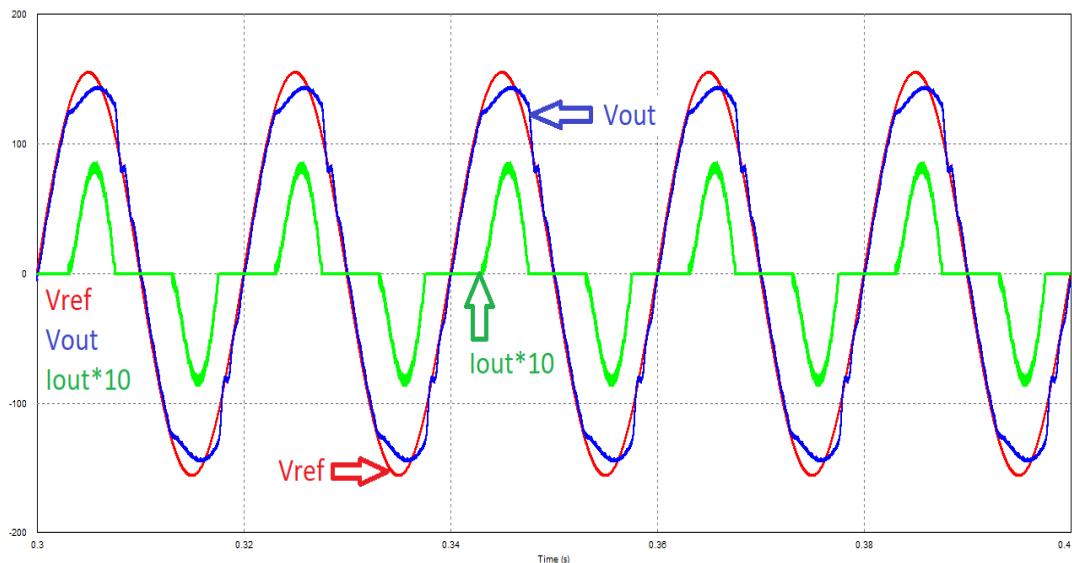


Figura 17 - Desempenho do controlador PD-feedforward para uma carga não linear

3.3.2 Controlador de Ação Repetitiva

O controlador de Ação Repetitiva apresentado na seção 2.2.2, que foi incluído na Figura 13 do diagrama de controle, está na forma tradicional. Trata-se de um compensador formado por conjunto de integradores, descritos como $(1/[1 - Q(z^{-1})z^{-N}])$ [8]. A integração corresponde à amostragem de um período da frequência de referência do sinal de erro. Os integradores operam em uma sequência formada por um atraso (z^{-N}), um ganho c_r e o filtro $C(z^{-1})$, conforme pode ser visto nas Figuras 9 e 10. A função de transferência para o controlador Repetitivo para a versão Filtro Q é:

$$G_{rp}(z^{-1}) = \frac{1}{1 - Q(z^{-1})z^{-N}} z^{-N} c_r C(z^{-1}) \quad (9)$$

O valor de N na função de transferência refere-se à quantidade de amostras por um período do sinal de referência, sendo definida por:

$$N = \frac{T_1}{T_s}, N \in N^+ \quad (10)$$

Sendo T_1 o período da referência.

Em relação ao compensador $C(z^{-1})$, deve-se destacar que, conforme [8, p. 73],

é empregado para compensar os atrasos de fase entre os geradores de sinais periódicos e a saída da planta, inserindo, desta forma, a ação de compensação aos distúrbios na planta com fase adequada. Como esta compensação de atraso de fase da planta só pode ser feita por um filtro com ação não-causal, é incluída a estrutura z^{-N} para tornar $C(z^{-1})$ causal. Desta forma, se atrasa a atuação da ação de controle em um período do sinal de referência. O ganho c_r pode ser considerado como sendo uma parte do filtro $C(z^{-1})$, que é adequadamente ajustado para se garantir a estabilidade da lei de controle.

Os parâmetros de configuração para esta função de transferência implementada estão na Tabela 2.

Tabela 2 - Parâmetros dos controles Repetitivos

Parâmetros dos controles Repetitivos	
z^{-Nd}	98
z^{-N}	100
N	100
Q	0.99
C_r	0.2

A Figura 18 mostra o desempenho do sistema para a mesma carga não linear mostrada na Figura 16, mas agora com a inclusão do controlador de ação repetitiva. Observa-se uma melhora significativa na forma de onda da tensão de

saída, a qual apresenta uma THD de 1.2% para uma THD da corrente de saída de 90%.

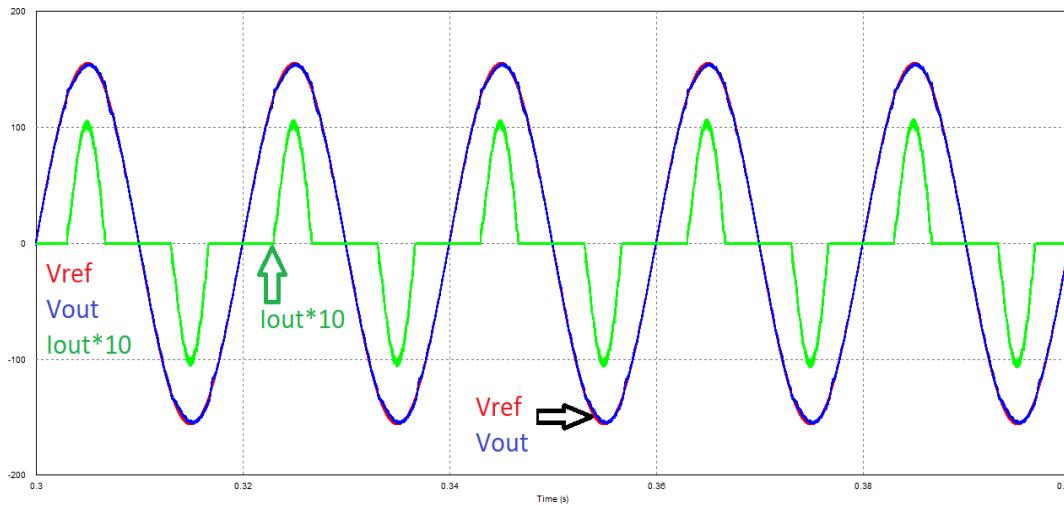


Figura 18 - Desempenho do controlador PD-feedforward + Repetitivo para uma carga não linear

4 METODOLOGIA

Durante o desenvolvimento da pesquisa houve uma série de correções de rota no que diz respeito à metodologia para a implementação do controle embarcado em FPGA. Em um primeiro momento, pensou-se em usar a geração do código a partir da linguagem VHDL (*VHSI Hardware Description Language*) e fez-se um estudo da mesma. A partir desse estudo, ficou evidente o potencial que esta linguagem oferece para implementar circuitos eletrônicos, mas também foi revelada a dificuldade de gerenciar circuitos maiores como o pretendido. Problemas como ter que controlar todos os sinais, gerar o controle de tempo para a sincronização dos sinais e a incalculável quantidade de entidades que deveriam ser implementadas levaram à conclusão de que, para este projeto, seria muito trabalho e o tempo de projeto seria aumentado drasticamente, sem contar o gerenciamento de problemas intrínsecos da FPGA. Por isso, para o desenvolvimento de todo o controle embarcado em FPGA, não seria viável usar o VHDL para sua implementação. A dificuldade de desenvolver todo o projeto diretamente em VHDL é destacado em [38], onde o autor relata não ter conseguido implementar por exemplo um controlador PWM.

4.1 System Generator

Devido à constatação da dificuldade de uso da linguagem VHDL, foi pesquisada outra solução de “programação” com maior nível de abstração [38]. Entrou em cena o *System Generator*. O *System Generator* apresentou de imediato vários benefícios: linguagem gráfica, o gerenciamento de sinais e tempos, facilidade de trabalhar com ponto fixo, interação com o ambiente Simulink do Matlab. A partir das vantagens apresentadas, definiu-se a opção pelo *System Generator*.

A primeira vantagem foi com relação à geração do código bitstream. O *System Generator* gera o código a partir do projeto em modo gráfico até alcançar o bitstream em apenas um comando. Com isso, ficam ocultas todas as fases de geração de código para a FPGA. Isso significa um ganho de tempo muito importante quando se tem que realizar este procedimento diversas vezes [39].

O *System Generator* destaca-se pela sua interface de interação com o usuário, realizada junto com o Simulink do Matlab. Nesta interface, o *System Generator* é uma *toolbox*, instalada no Simulink. Dessa forma, todo o ambiente do Simulink fica disponível para ser usado junto com o *System Generator*.

O *System Generator* oferece dois componentes que permitem fazer a interação entre estes dois ambientes: o *gateway out* e o *gateway in*. O *gateway out* devolve para o ambiente do Simulink os sinais da FPGA no formato de precisão dupla (*Double-Precision Floating Point*), ou seja, ponto flutuante com 64bits. Por outro lado, tem-se o *gateway in*, que realiza a operação inversa, ou seja, converte um sinal de saída do Simulink em ponto flutuante de precisão dupla para sinal de entrada da FPGA no formato ponto fixo, de acordo com a configuração preestabelecida. Este componente pode ser configurado para gerar a resolução de ponto fixo desejada. Com isso, a interação do *System Generator* com o Simulink fica muito “forte”, permitindo interagir com todas as bibliotecas do Simulink. A Figura 19 apresenta os blocos *Gateway in* e *out* de forma didática.

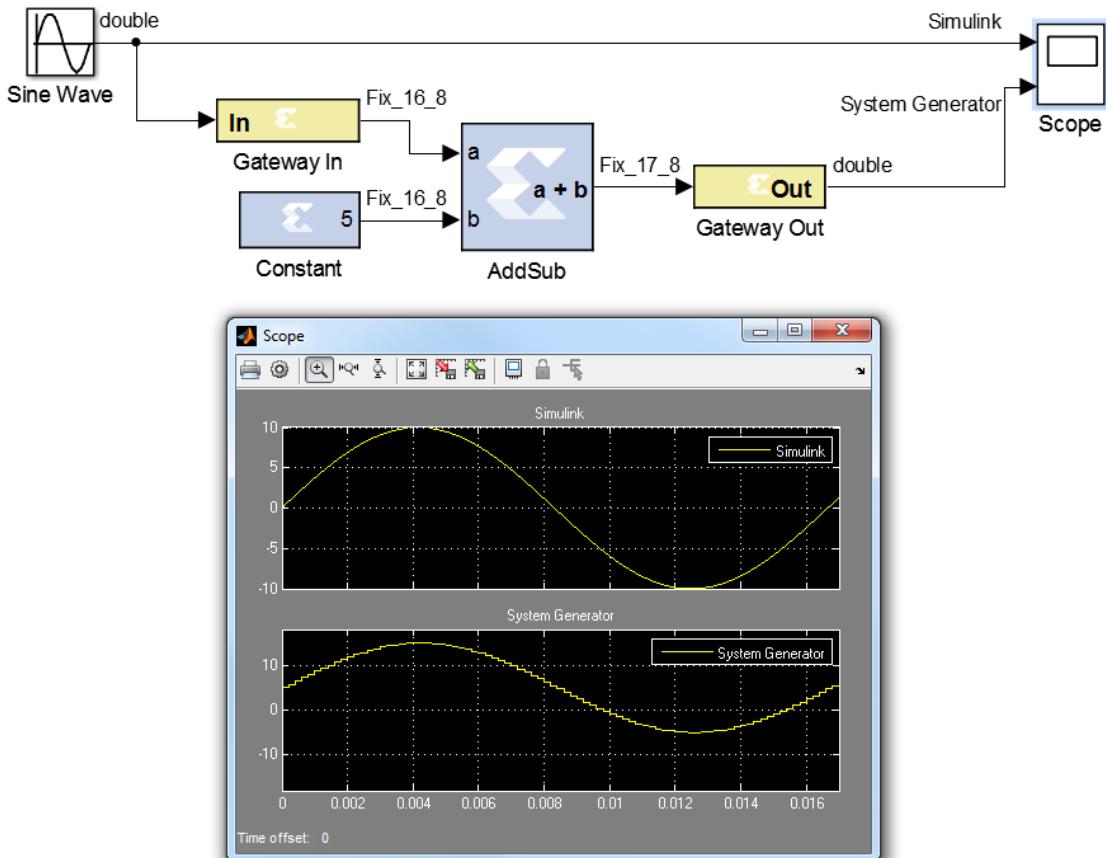


Figura 19 - Gateways

O bloco *Gateway* revela uma das facilidades de converter a representação dos números de ponto flutuante para ponto fixo (o número, para ser representado em ponto fixo, é dividido em duas partes: uma para representar a parte inteira e outra para a parte fracionária, e cada qual pode ter um determinado número de bits, de acordo com a precisão desejada). Este processo é transparente para o usuário. É suficiente colocar o formato desejado (número de bits para a parte inteira e fracionária). Na Figura 20 tem-se a interface de configuração do *Gateway In*, com a representação de S.1.14 (16 bits total, divididos da seguinte forma: 1 bit de sinal, 1 para parte inteira e 14 para a parte fracionária). Além dessa representação, também temos booleana e ponto flutuante.

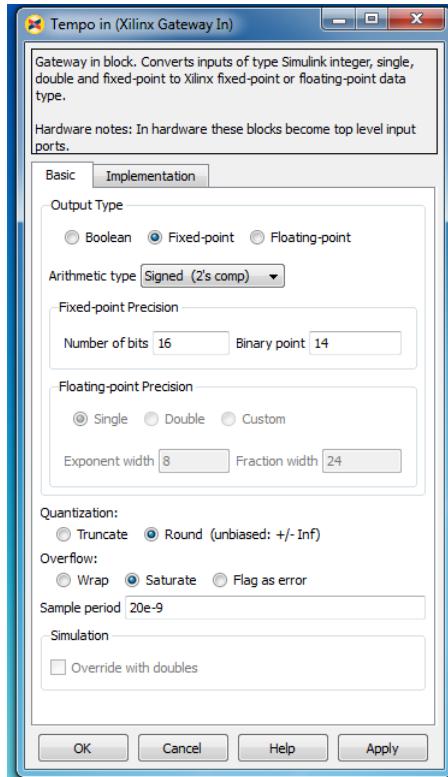


Figura 20 - Gateway In

Outro aspecto importante, visto que a aplicação envolve um controlador no domínio do tempo discreto, é que a forma de amostragem pode ser facilmente determinada através da divisão do clock base disponível na placa da FPGA. Este fato torna possível a aquisição de qualquer sinal do Simulink para FPGA. Nesse sentido, pode-se trabalhar com qualquer sistema analógico no Simulink cujo sinal vai ser diretamente amostrado para FPGA.

As simulações envolvendo o *System Generator* são *bit-True and Cycle-True Modeling*. *Bit-True* significa que os dados na fronteira, *Gateway in* e *Gateway out*, entre o *System Generator* e demais blocos do Simulink são correspondentes bit a bit. *Cycle-true* que dizer que os dados são apresentados na fronteira na taxa de amostragem definida pelo passo de simulação [39]. Em suma, isto quer dizer que o comportamento das variáveis é exatamente aquele realizado em *hardware* pela FPGA. Os ciclos de tempo das interfaces de *Gateway in* e *Gateway out* serão os mesmos da simulação e do hardware, fato que foi constatado em vários testes(comparação de períodos e resultados de operações) preliminares realizados para uso no desenvolvimento inicial dos controladores estudados neste trabalho.

No trabalho realizado utilizando o *System Generator* tem-se um ganho de produtividade em razão da existência de um grande e variado conjunto funções primitivas. Tem-se funções matemáticas com as operações de soma, subtração, multiplicação, valor absoluto, constante, divisão, logaritmo natural, negativo, acumulador, raiz quadrada; funções diversas, como atraso e registro. E ainda funções especiais, como: memórias, contadores, reinterpreta, concatena, FIFO, convolução. A Figura 21 apresenta algumas funções. No APÊNDICE A temos mais funções primitivas.

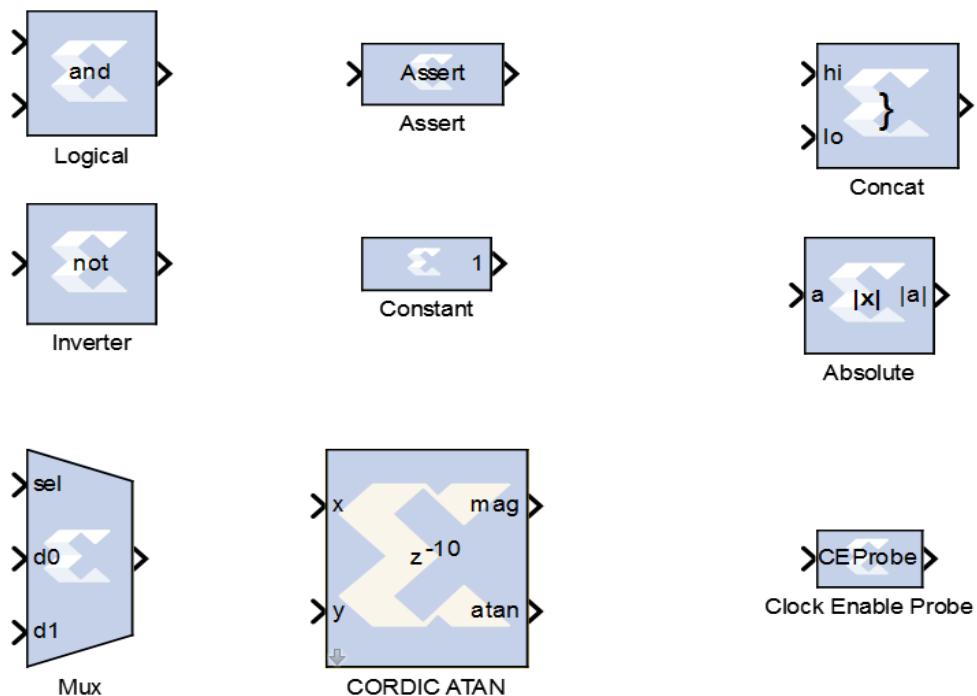


Figura 21 - Exemplos de primitivas do System Generator

Além disso, a ferramenta permite a simulação em tempo real ou a co-simulação (*hardware-in-loop*). Todos os blocos disponíveis no ambiente do Simulink podem interagir com as partes implementadas na FPGA, através de blocos de interfaces. Isto é, o código é gerado e implementado na FPGA e executado na mesma. É possível visualizar os resultados da FPGA através de bibliotecas criadas para o projeto dentro do ambiente do Simulink. Pode-se usar o *Scope*, o *To file*, *to workspace*, *Display*, entre outros, para acompanhar a execução do código. Esta

opção foi muito relevante para o desenvolvimento do presente trabalho, pois através dela pode-se fazer a depuração do código embarcado na FPGA, de forma a confirmar a exatidão dos cálculos e a correta operação dos controladores do inversor de tensão. A interface com o Simulink permite a visualização gráfica dos sinais e não uma mera listagem de bits, fato que amplia a capacidade de depuração de algoritmos de controle pelo projetista.

Apesar da grande variedade e bibliotecas primitivas, pode ser difícil encontrar um conjunto de funções primitivas para a implementação de uma função específica. Neste caso o *System Generator* oferece um bloco chamado *Black Box*, o qual permite a determinação de sua função por meio de um código VHDL. Este bloco é adicionado ao projeto diretamente e se transforma em uma biblioteca para ser utilizado dentro do projeto de maneira semelhante aos demais blocos primitivos. Pode ser copiado quantas vezes for necessário. O principal cuidado na utilização do *Black Box* é manter a compatibilidade dos tipos de dados de entradas e saída. Os tipos de dados são definidos na geração do código, sua alteração exige a reedição do código e atualização do *Black Box* na biblioteca.

Dentro do mesmo raciocínio tem-se o bloco *Mcode*, que funciona de modo semelhante ao bloco *Black Box*. O *Mcode* permite importar um código construído no Matlab usando os seus códigos de programação, mas necessita ser semelhante a uma função. Neste caso, a programação inclui os tipos de dados de entrada e saída. Diferentemente do *Black Box*, que “automaticamente” configura a sua interface, neste o programador tem a responsabilidade de definir os detalhes desta configuração.

Pode-se concluir que as duas soluções comentadas revelam uma grande flexibilidade no uso desta ferramenta de programação, permitindo ao programador fazer seu código em VHDL ou Matlab e importar para o ambiente gráfico e beneficiar-se de todos os recursos do Simulink. Apresenta-se na Figura 22 o bloco MCode, no caso, implementado para realizar uma máquina de estado. O código que foi implementado para a máquina de estado encontra-se no Anexo B.

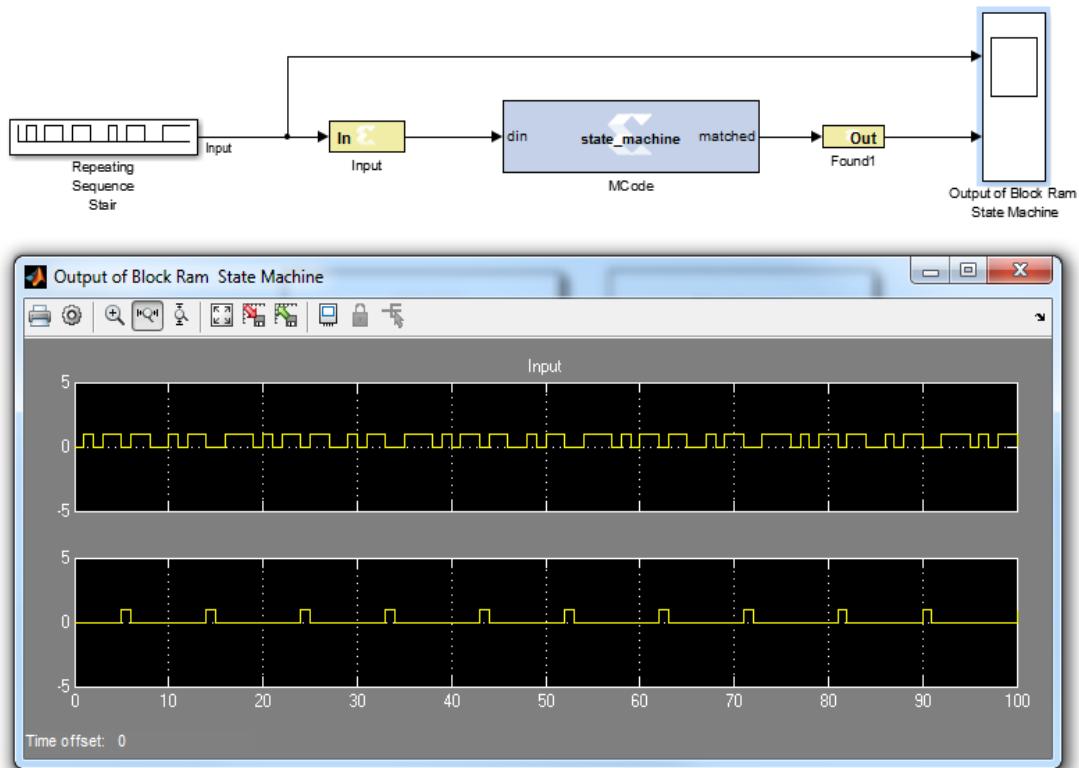


Figura 22 - Exemplo implementado com o bloco Mcode

O *Black Box* foi utilizado no projeto diretamente para realizar várias funções, entre elas a função semelhante à *Sample-and-hold*, que é mostrada na Figura 23. E o código que foi implementado está no apêndice b.

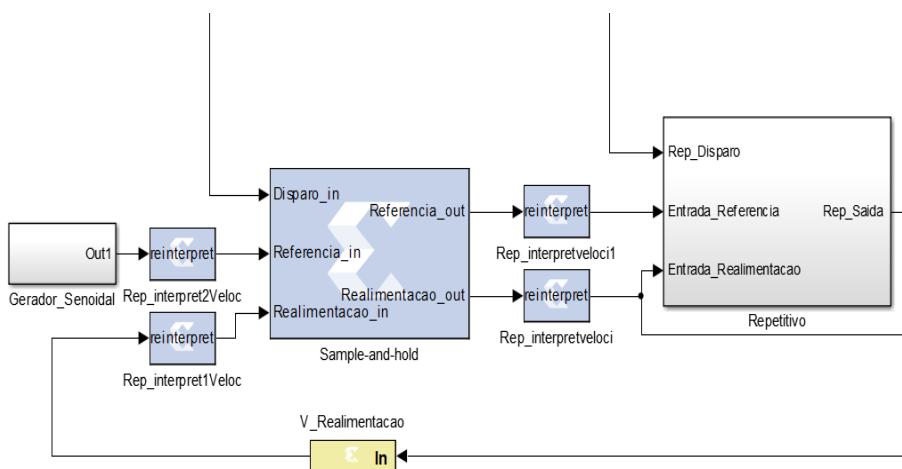


Figura 23 - Exemplo do uso do Black box para implementar a função "sample-and-hold"

4.2 VHDL

Apesar de grande parte da implementação dos controladores ter sido feita com blocos primitivos do *System Generator*, o conhecimento da linguagem VHDL foi importante, visto que a assimilação de alguns conceitos desta facilitaram a compreensão da operação de alguns blocos do *System Generator*, permitindo sua utilização com maior propriedade. Dois conceitos devem ser destacados, o de sinal e o de processo. É preciso entender que o sinal só será atualizado ao final do processo. Isso significa que não se pode tomar nenhuma decisão na manipulação de um sinal antes do término de um processo, caso contrário, o sinal não corresponderá ao resultado do processo. Já o processo integra um conjunto de variáveis às quais ele é sensível e também um algoritmo que determina a relação entre estas variáveis. Este conceito é semelhante ao da sub-rotina de atendimento à interrupção nos DSPs. Lembrando que um processo pode ter n sinais. Outro conceito relacionado ao processo é o de que ele é sequencial. Para este caso, todo o código dentro do processo segue uma sequência conforme um programa de linguagem procedural.

O *System Generator* disponibiliza uma grande quantidade de bibliotecas primitivas e avançadas. Mas pode ocorrer a necessidade de implementação de algo que as bibliotecas não oferecem, neste caso é necessário importar um código VHDL para um bloco Black Box dentro do *System Generator*. Dessa forma, o conhecimento da linguagem VHDL foi importante na implementação destes componentes, como por exemplo, a programação dos amplificadores dos conversores AD, presentes na placa de desenvolvimento, e no componente chamado velocidade, que faz a amostragem do sinal de realimentação.

4.3 Construção dos controladores

A construção dos controladores Repetitivo e *PD-feedforward* foi realizada em quatro etapas. A primeira consistiu em uma leitura sobre o controlador, para conhecer os detalhes de seu funcionamento, sua função de transferência e suas limitações. Na segunda etapa, realizou-se a simulação para se ter um maior conhecimento da dinâmica do controlador. Na terceira, a implementação no *System*

Generator com simulação e Co-Simulação. E por último, a geração do arquivo de *bitstream* para ser gravado na FPGA.

No segundo momento, as simulações foram realizadas no software PSIM. Na simulação eram contemplados a planta com todas as partes, o PWM, o filtro de saída, o barramento CC, carga não linear, tensão de referência, enfim do projeto. Na Figura 24 tem-se um exemplo do controle Repetitivo e PD-feedforward que foi simulado.

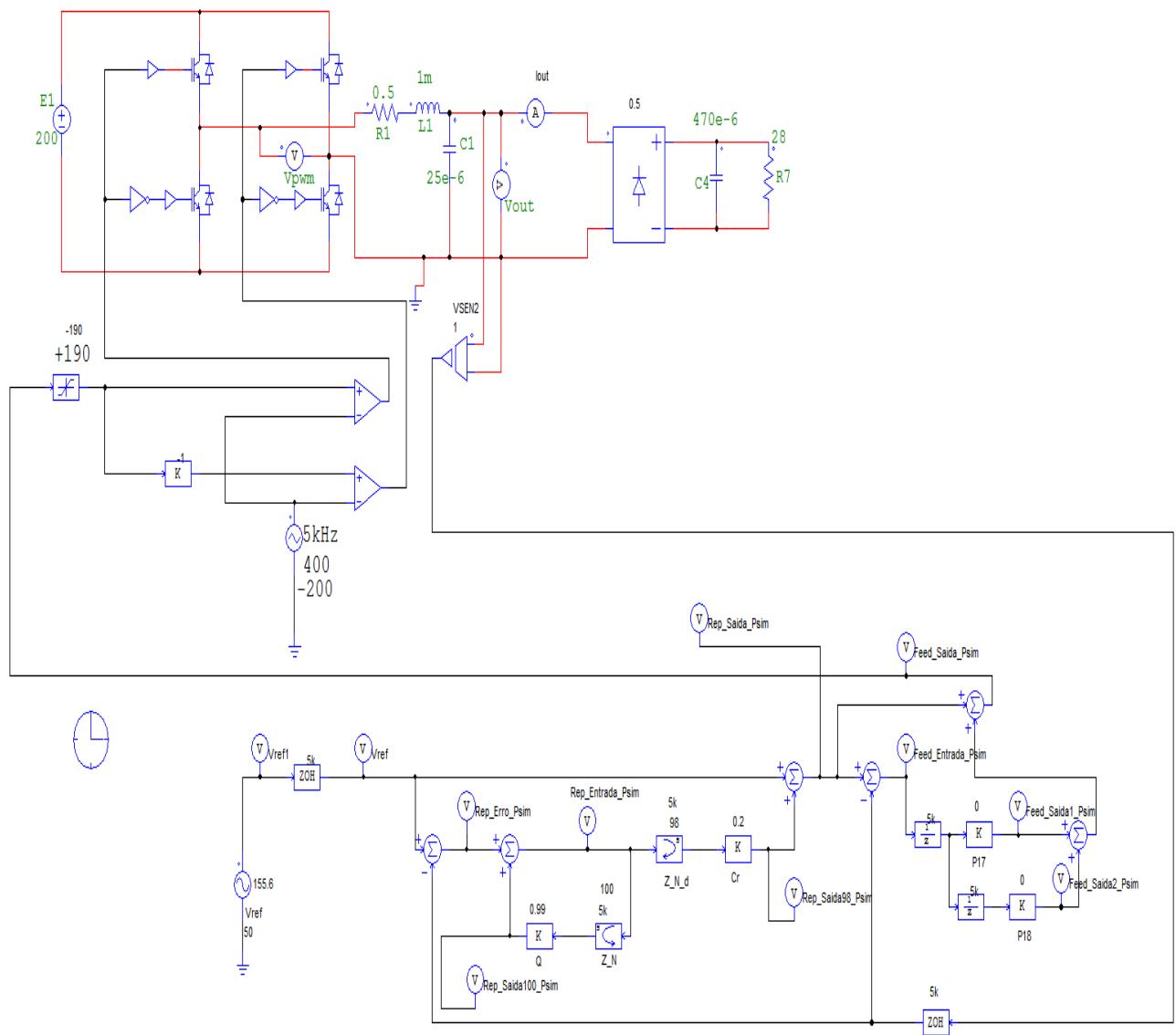


Figura 24 - Simulação PSIM Controle Repetitivo

A simulação possibilitou, além de conhecer melhor a dinâmica do controle, o conhecimento dos sinais relativos às variáveis internas do controlador, de modo que partes deste pudessem ser analisadas de forma independente nas depurações do código da FPGA. Pode-se ver as formas de onda da saída do Repetitivo, da saída do PD-feedforward, da tensão de saída e da corrente de saída. Com isso, foi sendo construído um conhecimento sobre cada ação de controle. A Figura 25 mostra um exemplo de forma de onda da saída desta fase. Este sinal amostrado na saída do controlador PD-feedforward serve para conferir se o controle realizado na simulação no System generator esta coerente com a simulação do Psim.

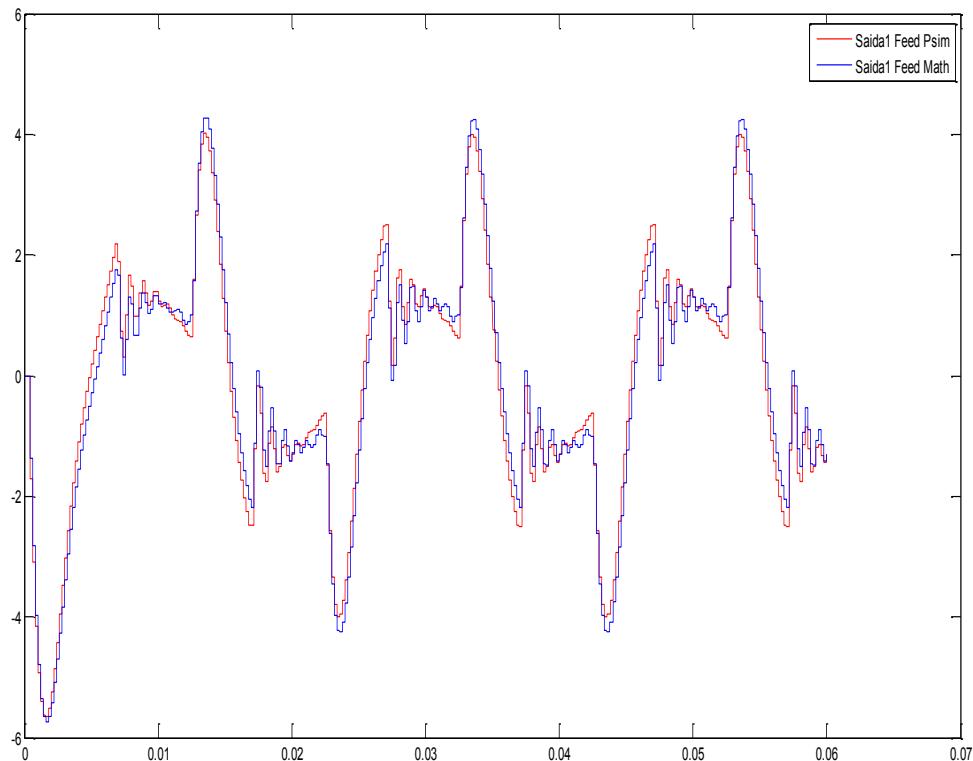


Figura 25 - Sinal amostrado na saída do controlador *PD-feedforward* no System generator e Psim.

Na sequência foi realizada a simulação dos controladores no *System Generator* dentro do ambiente do Simulink. A partir dos circuitos realizados na simulação do PSIM, foi realizada a migração destes para o ambiente do Simulink. As equações de diferenças foram escritas de forma semelhante às do PSIM e novamente foi possível ver as formas de onda nos vários pontos do circuito. Mas as

principais foram a tensão de saída e o sinal de referência, de forma a avaliar o funcionamento das ações de controle. A Figura 26 mostra o esquema de simulação para o controle Repetitivo.

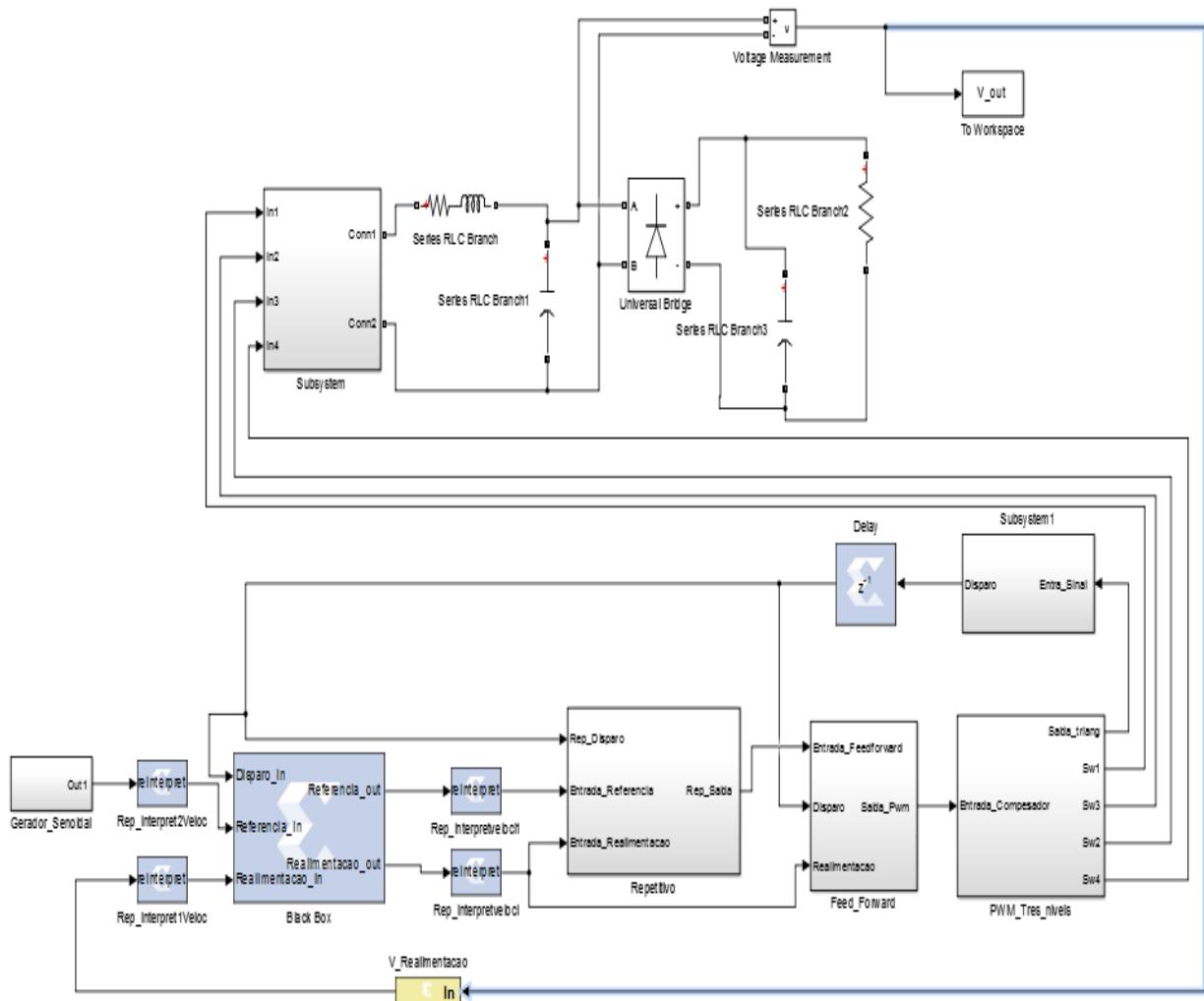


Figura 26 - Projeto do controlador Repetitivo e *PD-ffeforward* Simulado

Por fim, foi realizada a co-simulação. A co-simulação consiste em executar o código referente à FPGA diretamente na mesma e o restante do circuito no ambiente do Simulink. Na co-simulação pode-se verificar se todo o circuito que será executado pela FPGA está funcionando corretamente. Faz-se o uso das ferramentas do Simulink, para ver os resultados e fazer a respectiva depuração.

A partir da programação da FPGA utilizada nos testes de co-simulação, foi gerado o arquivo de bitstream utilizado para embarcar o código do controlador na FPGA.

4.4 Detalhes da metodologia

Para a operação adequada da planta, além dos controladores, outros subsistemas são necessários. Estes também foram embarcados na FPGA, a saber: o circuito que gera a forma de onda de referência, o circuito que gera a forma de onda triangular para o PWM, o circuito que sincroniza a aquisição do sinal de referência juntamente com o de realimentação.

O circuito que gera a onda de referência é uma memória que possui nos seus registros um conjunto de pontos de uma senóide. E para gerar a forma de onda senoidal, faz-se o endereçamento sequencial destas posições de memória através de um contador cujo período de contagem é um submúltiplo do período da senóide. Para o projeto foi utilizado um vetor de 1000 posições. Então, tem-se uma senóide de 1000 pontos para um período de 0.02 segundos, frequência usada em [8]. A Figura 27 mostra o circuito gerado do sinal de referência.

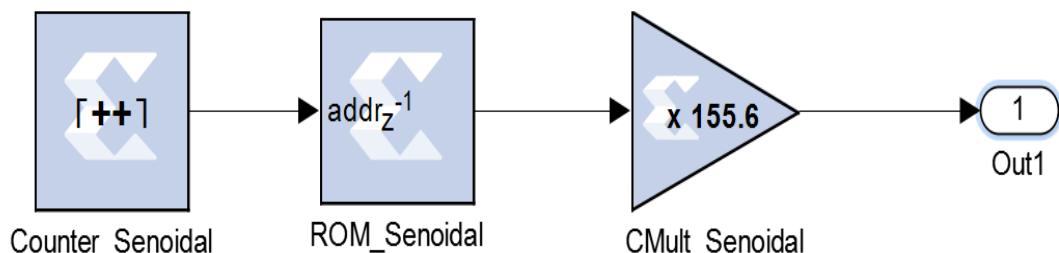


Figura 27 - Circuito gerador do sinal de referência

O circuito da onda triangular é formado por dois contadores: um para definir a contagem da saída e outro para alterar o sentido da contagem. O contador que gera a saída conta de 0 a 500, mas começa em 250, e o outro contador muda o sentido para cada 500 passos de contagem. Dessa forma obtém-se uma onda triangular. Os tempos dos contadores foram ajustados para gerar uma forma de onda triangular de 5 kHz. O circuito está exposto na Figura 28.

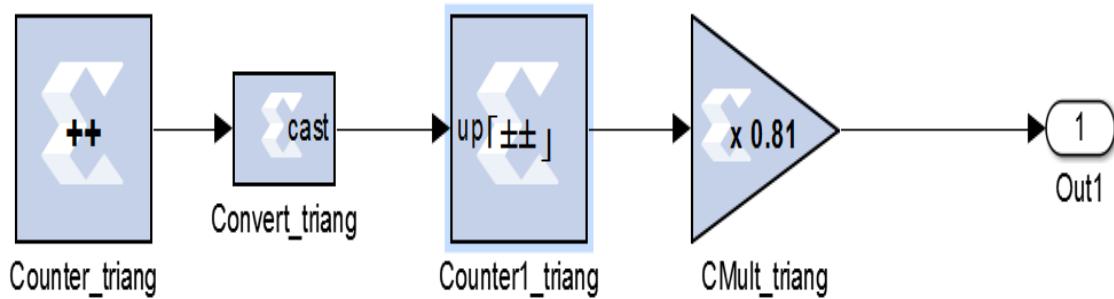


Figura 28 - Circuito gerado da forma de onda triangular

O outro circuito utilizado foi o amostrador semelhante ao *Sample-and-hold*. Este circuito foi construído a partir do código VHDL e importado para o circuito através do bloco *Black Box*. O circuito amostra os sinais e mantém o resultado na sua saída até o próximo pulso de sincronismo.

4.4.1 Funções de transferências

As funções de transferências foram implementadas na forma de equação de diferenças, a partir das bibliotecas primitivas disponibilizadas pelo *System Generator*. Apenas um componente foi realizado utilizando-se o bloco *Black Box* para importar o código em VHDL, de forma a realizar os atrasos necessários para cada função de transferência. Nas Figuras 29 e 30 temos a equação da função de transferência do Repetitivo e do PD-feedforward, respectivamente.

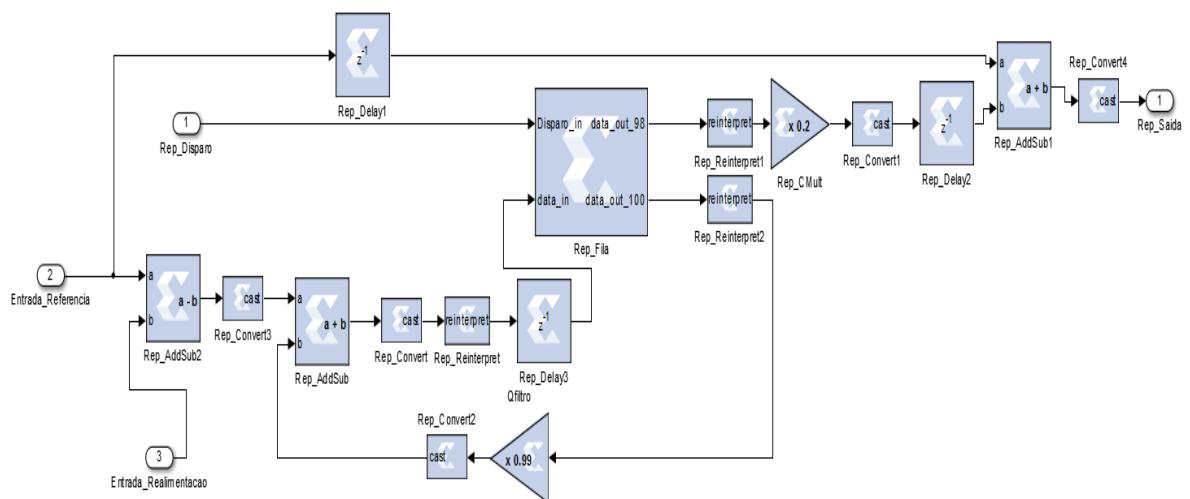
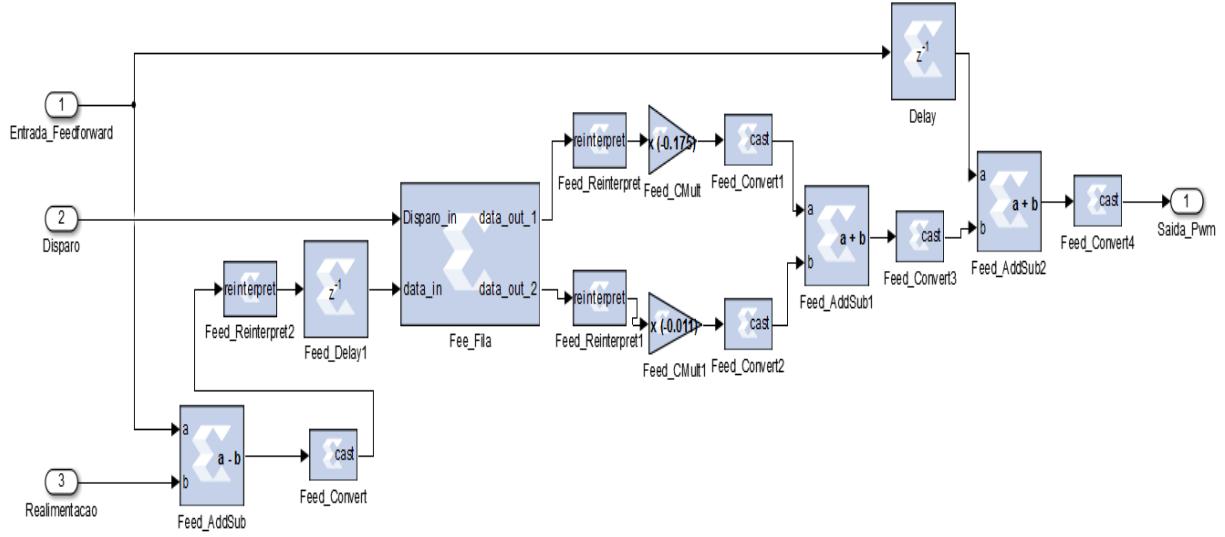


Figura 29 - Função de transferência do Repetitivo

Figura 30 - Equação de diferenças do *PD-feedforward*

Outro circuito auxiliar é o PWM, que foi realizado apenas com as bibliotecas primitivas do *System Generator*, comparadores, inversor e relacional. A forma de onda triangular já foi apresentada como circuito auxiliar. Este circuito foi o mesmo utilizado na implementação da modulação por largura de pulso em configuração unipolar (Figura 31).

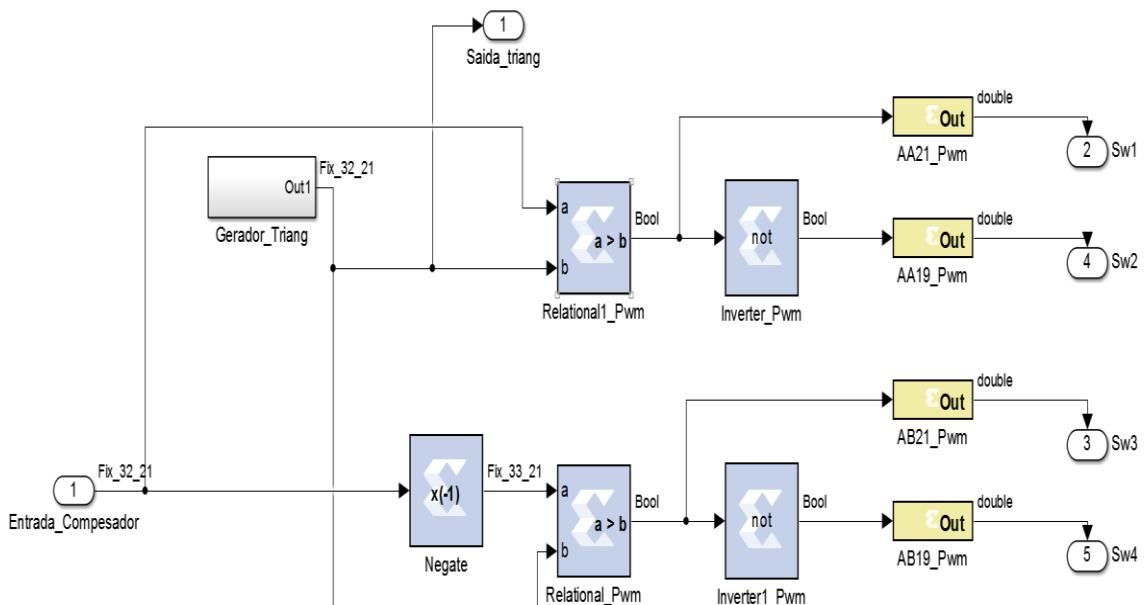


Figura 31 - Circuito PWM

4.4.2 O circuito de ajuste do tempo de aquisição

Este circuito é semelhante a um atraso programado para que ocorra a aquisição da tensão de saída da planta para as ações de controle. Para a ação Repetitiva foi necessário adquirir a amostra com exatidão. A primeira tentativa foi realizar uma amostra durante o período de 5 KHz, em qualquer ponto, respeitando apenas o limite da janela da tensão da portadora maior que a tensão da moduladora. Mas isto conduziu o sistema para um erro entre a amostra atual e a amostra que estava no Buffer de memória. E, para agravar, este erro aumentava com o tempo de simulação. Portanto, o sistema tornava-se instável.

A solução foi alterar o ponto de aquisição para que ocorresse ao final da descida do pulso de 5KHz. O algoritmo foi o seguinte: calculou-se a quantidade de pulsos necessários para ocorrer uma amostra e a entrega do resultado, neste caso 70 pulsos, sendo 35 pulsos para gerar a 1^a aquisição e depois mais 35 pulsos para ler o resultado referente à primeira aquisição. Cada pulso do conversor AD foi programado com duração de 40ns, resultando em 2,8 microssegundos o tempo de uma amostra. O pulso de 5KHz não é simétrico e resultava em uma janela de 6,2 microssegundos. Então, para se obter uma amostra mais perto da borda de descida, onde foi programada a leitura, o processo de aquisição deve começar em 3,4 micro segundos antes da descida da borda do pulso de 5KHz.

O circuito foi implementado com o objetivo de realizar esta aquisição, conforme descrito acima. Assim, o processo ficou programado para iniciar a contagem do tempo na borda de subida do pulso de 5KHz e a leitura do resultado na borda de descida. Implementou-se um atraso de 3,3 microssegundos para que o sistema funcionasse corretamente. E a partir desta solução, todas as amostras foram realizadas exatamente no mesmo momento, eliminando-se os erros entre a leitura das amostras. A Figura 32 mostra o diagrama de ajuste de tempo de aquisição implementado.

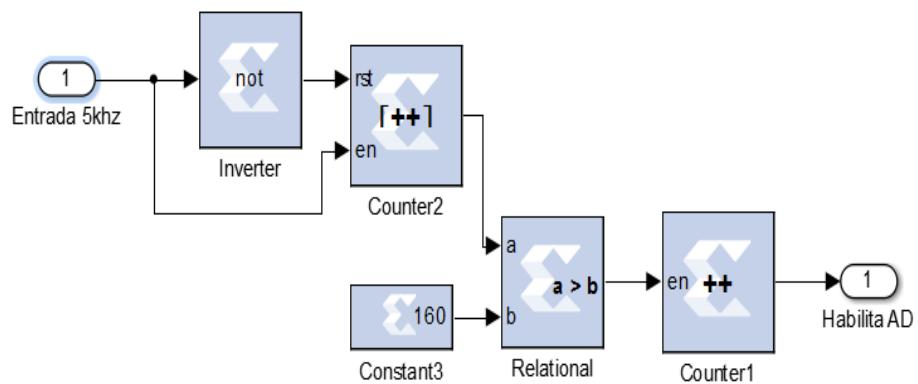


Figura 32 - Circuito de atraso para aquisição

O funcionamento deste circuito ocorre quando o sinal de 5kHz está em nível baixo, o contador 2 é zerado, não tem contagem, e consequentemente não temos pulsos na saída. Caso contrário, o sinal de 5kHz está em nível alto, temos o sinal *rst* igual a zero e o contador habilitado. A partir deste momento inicia-se a contagem (atraso) até o valor tornar-se maior que 160. Depois do valor ultrapassar 160, o contador 1 começa a gerar os 70 pulsos para o conversor AD funcionar. Depois o pulso de 5kHz volta para nível baixo, o contador 2 é zerado e a contagem para e este ciclo repete-se enquanto o sistema estiver funcionando.

5 DESCRIÇÃO DA PLANTA

Nesta seção serão apresentadas todas as partes que integram o sistema inversor, uma visão geral da planta é mostrada na Figura 33 e o diagrama na Figura 34.

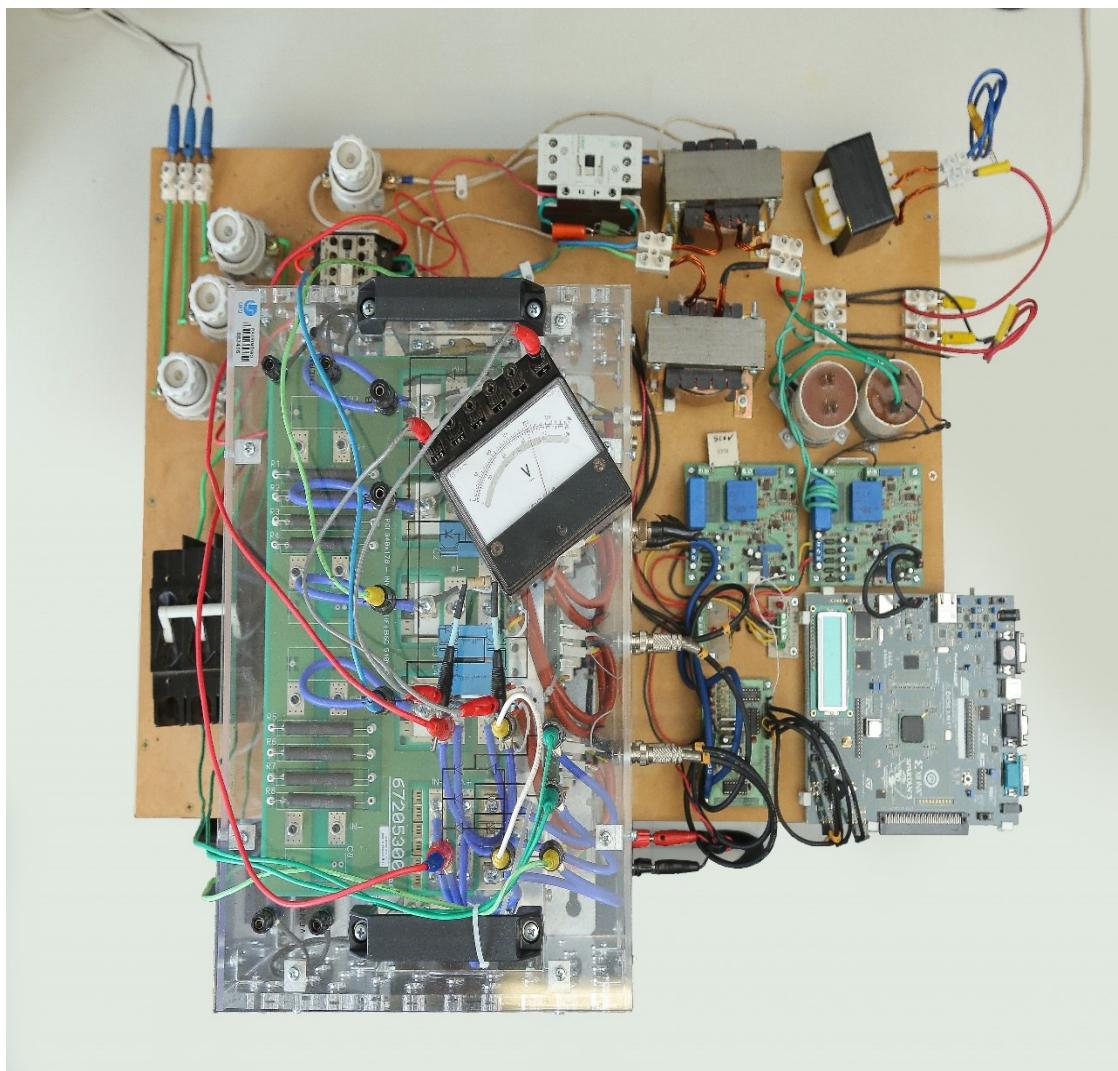


Figura 33 - Imagem da planta

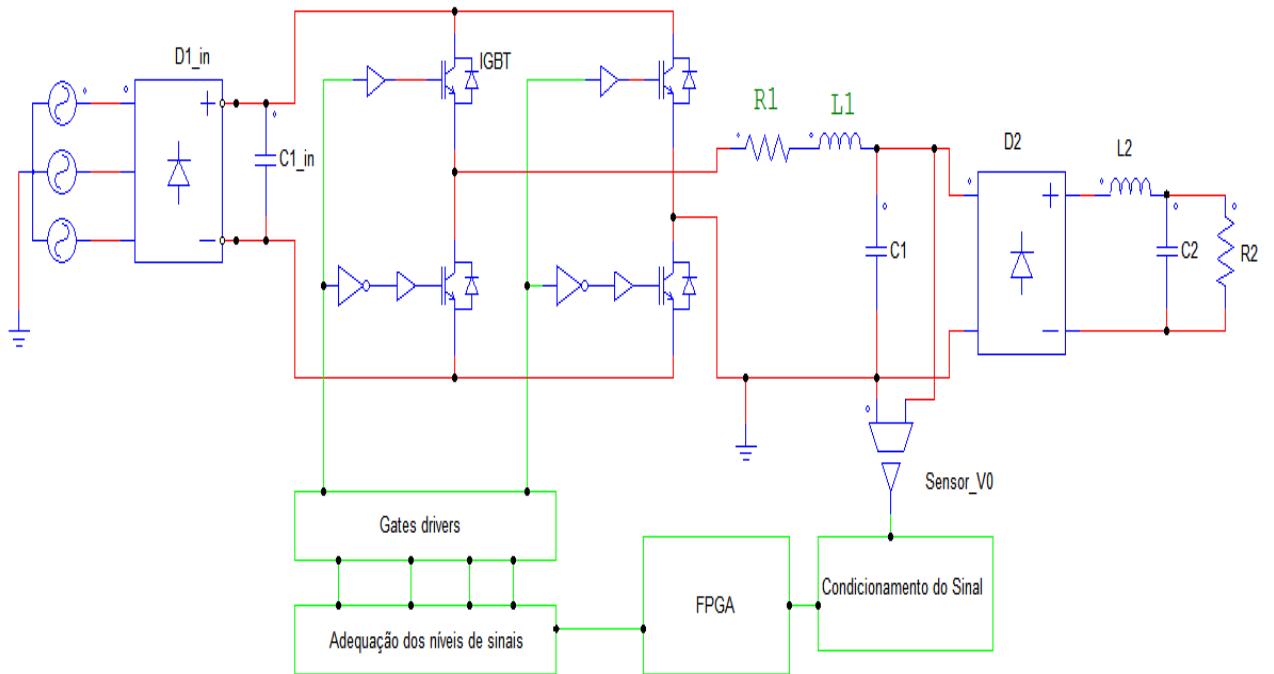


Figura 34 - Diagrama de ligação da planta

5.1 Placa de aquisição

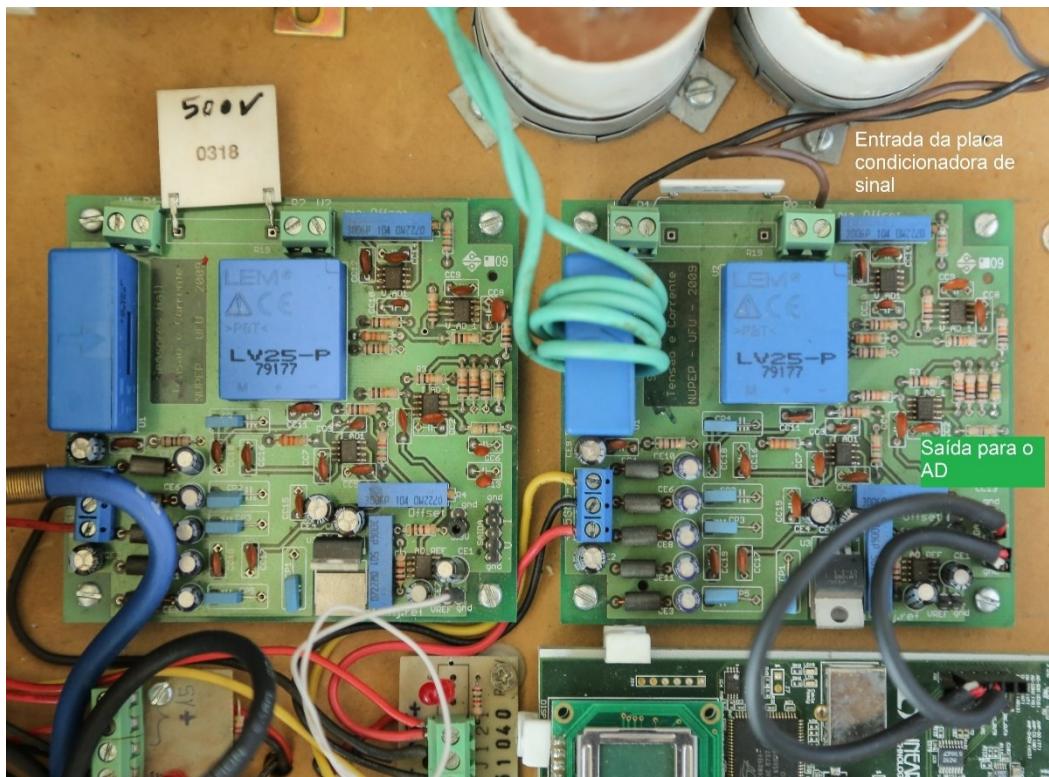


Figura 35 - Placa de aquisição

A placa de aquisição e condicionamento de sinal utilizada foi produzida no NUPEP (Figura 35). É formada por dois sensores de efeito Hall, um sensor de tensão e outro de corrente e vários outros componentes necessários ao condicionamento do sinal. O módulo de aquisição pode ser dividido em três partes: estágio de filtro e regulação da tensão de alimentação, estágio de tratamento do sinal e o estágio do filtro anti-aliasing.

O circuito do primeiro estágio pode ser visto na Figura 36.

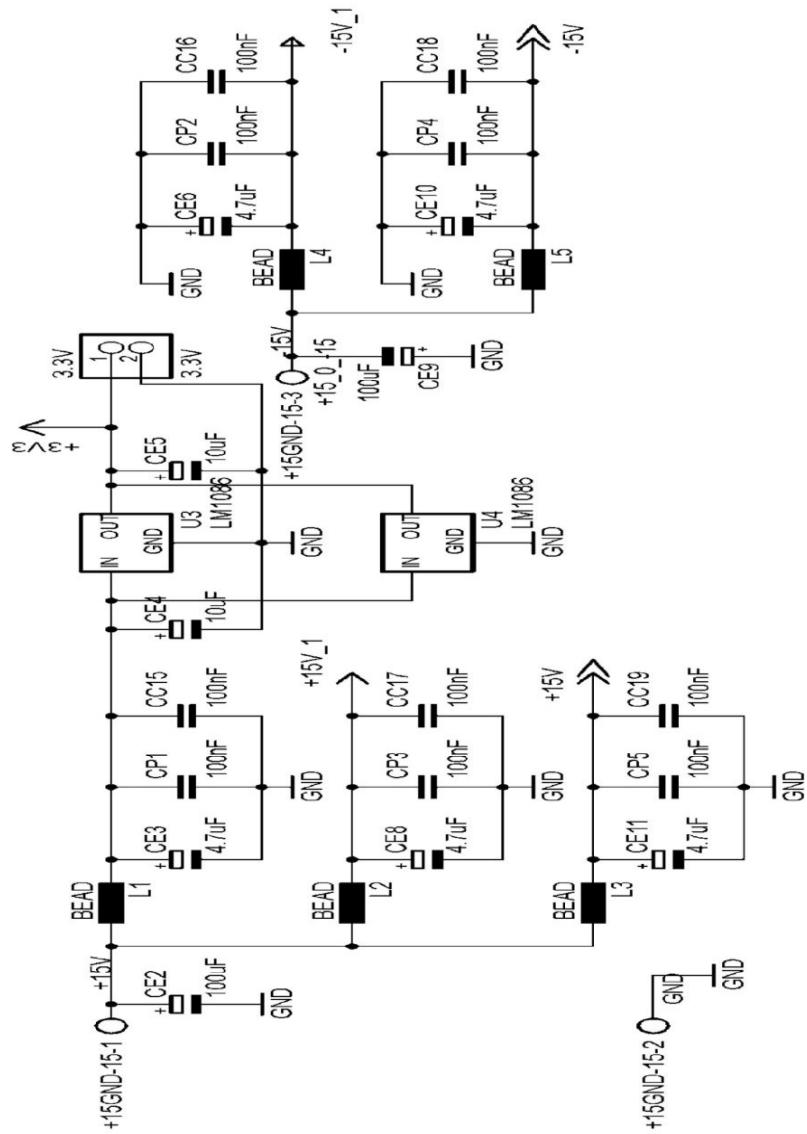


Figura 36 - Estágio de regulação e filtragem

É importante destacar que nesse estágio ocorre a separação entre as alimentações dos AO (Amplificador Operacional) do estágio de filtro de tensão e o de corrente, possibilitando o desacoplamento e, assim, impedir a interferência entre os sinais. Outro circuito relacionado com a parte de alimentação é o de ajuste de tensão de *offset*, apresentado na Figura 37.

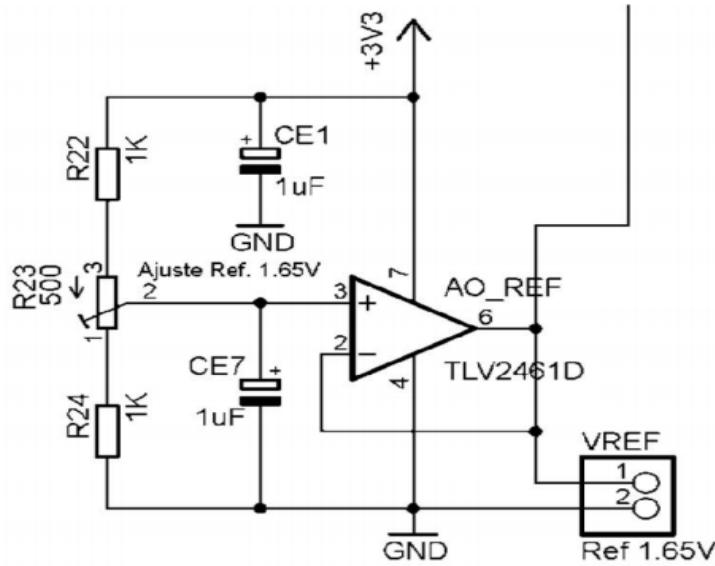


Figura 37 - Estágio de regulagem: Divisor de tensão gerador do sinal de referência para compensação de *offset*

O segundo circuito é um filtro anti-aliasing para os dois canais, um de corrente e outro de tensão, responsável por limitar a frequência do sinal amostrado. Circuito exibido na Figura 38.

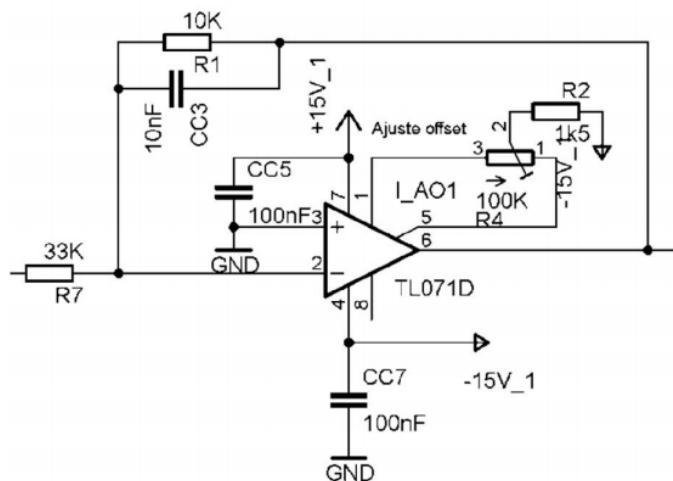


Figura 38 - Filtro anti-aliasing

A terceira parte é um circuito que soma o sinal vindo dos estágios antecedentes com um nível de *offset* de aproximadamente 1,5V. Dessa forma possibilitando a simetria de sinal para FPGA. Onde o valor de 1,5v corresponde a 0 (Figura 39).

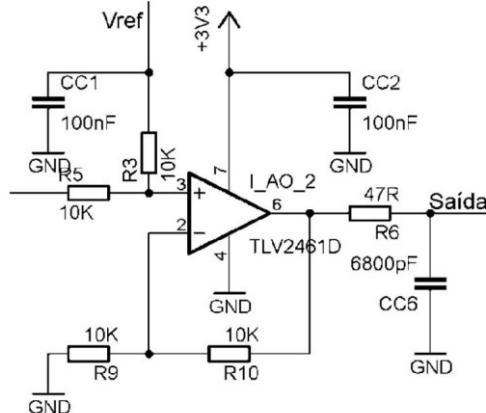


Figura 39 - Estágio somador e gerado de offset

Maiores detalhes da placa de sensores e condicionamento de sinais podem ser obtidos no Anexo A, que apresenta o diagrama esquemático completo da placa.

5.2 Placa de adequação dos sinais da FPGA para os Gate Drivers dos IGBTs

Devido à diferença de tensão de funcionamento da FPGA e dos Gate Drivers, tem-se a necessidade de fazer uma adequação nos níveis de tensão para que possam trabalhar juntos. A FPGA utiliza tensão de alimentação de 3.3V e trabalha no padrão de tensão de LVTTL, portanto nível alto em 3.3V. Por outro lado, os Gate driver trabalham com o padrão CMOS para os níveis de tensão de entrada. A placa utilizada é mostrada na Figura 40.

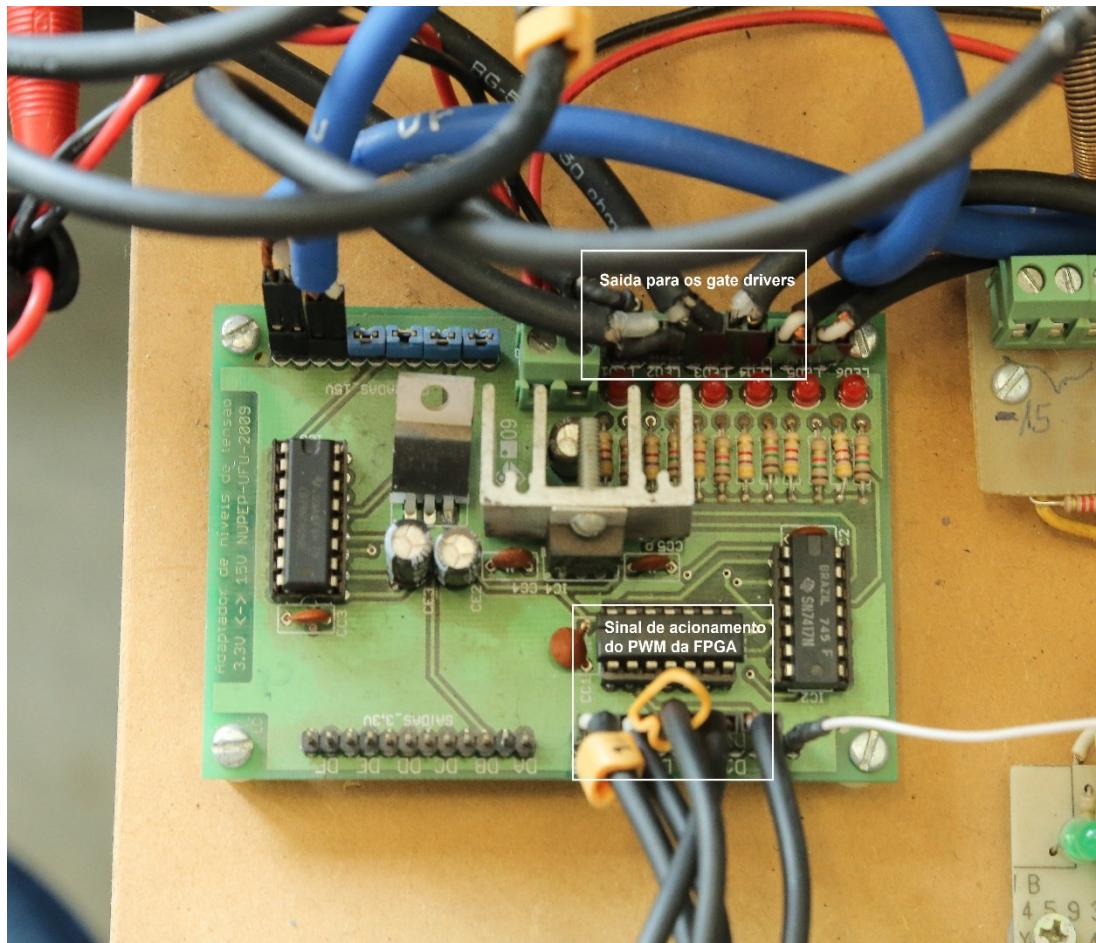


Figura 40 - Placa de adequação de tensão entre FPGA e Gate Driver, conectada.

5.3 A carga

Visando demonstrar o elevado desempenho do controle de tensão de saída do inversor, assim como na referência original do controlador utilizado [8], foi utilizada uma carga não linear. Trata-se de um retificador não controlado com filtro capacitivo, um resistor conectado em paralelo, conforme o diagrama apresentado na Figura 41.

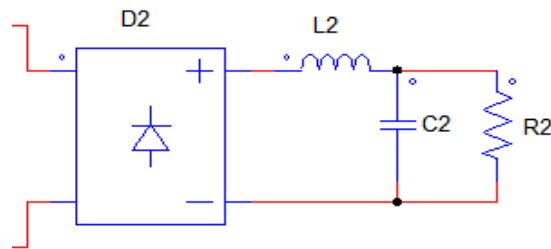


Figura 41 - Esquema da ligação da carga

Na Figura 42, foto da placa com o retificador e capacitor de filtro utilizado na planta.



Figura 42 - Retificador não controlado e capacitor de filtro

A carga utilizada como resistor está apresentada na Figura 43.

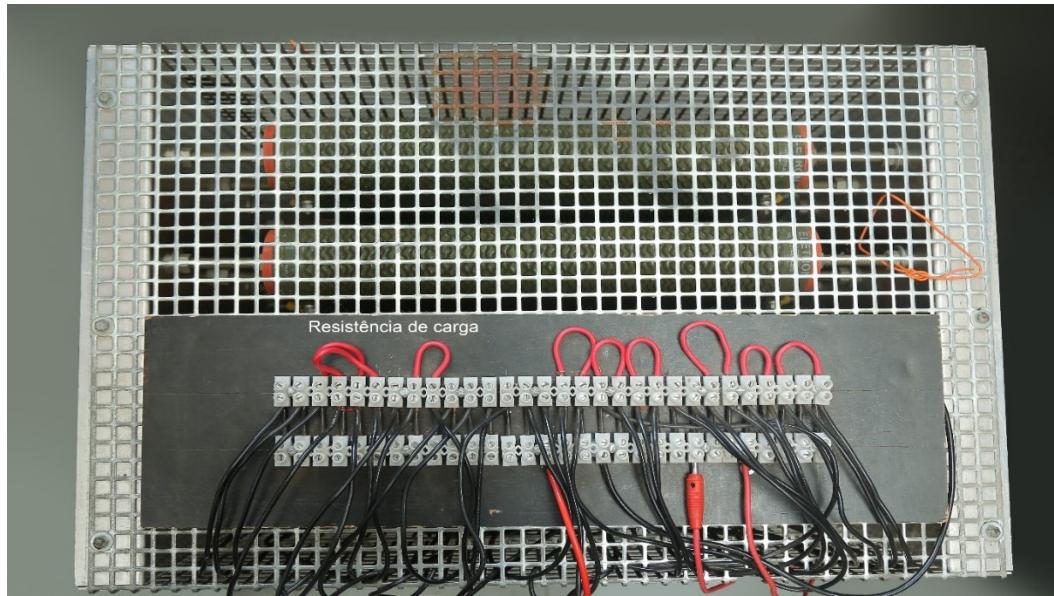


Figura 43 - Resistência de Carga

Tabela 3 - Parâmetros da carga

Descrição	Parâmetros
D1	6A10 DC
R2	98 Ohms
C2	470 μ F 400V
L2	1mH

5.4 Filtro LC

O filtro LC realizado no projeto tem como parâmetro o indutor de 1mH e o capacitor de 30 μ F. No caso, o indutor foi dividido em duas partes de 0.5 mH, para diminuir os ruídos e evitar que a tensão pulsante do PWM interferisse no sensor de efeito *hall*, sem, contudo, alterar as características dinâmicas do filtro [40]. O filtro de saída realiza a demodulação do sinal, que antes foi modulado pelo circuito PWM, promovendo a atenuação das componentes espectrais superiores, deixando passar apenas as componentes de baixa frequência. O filtro LC constitui então um filtro passa baixa de segunda ordem, com frequência de corte maior que a fundamental

[8, p. 47]. Não faz parte do escopo do presente trabalho discutir as metodologias para projetos de filtros de saída de inversores de tensão. No caso em questão, foram consideradas as referências [41; 42] para o desenvolvimento do filtro adotado neste trabalho. O filtro implementado é mostrado na Figura 44.

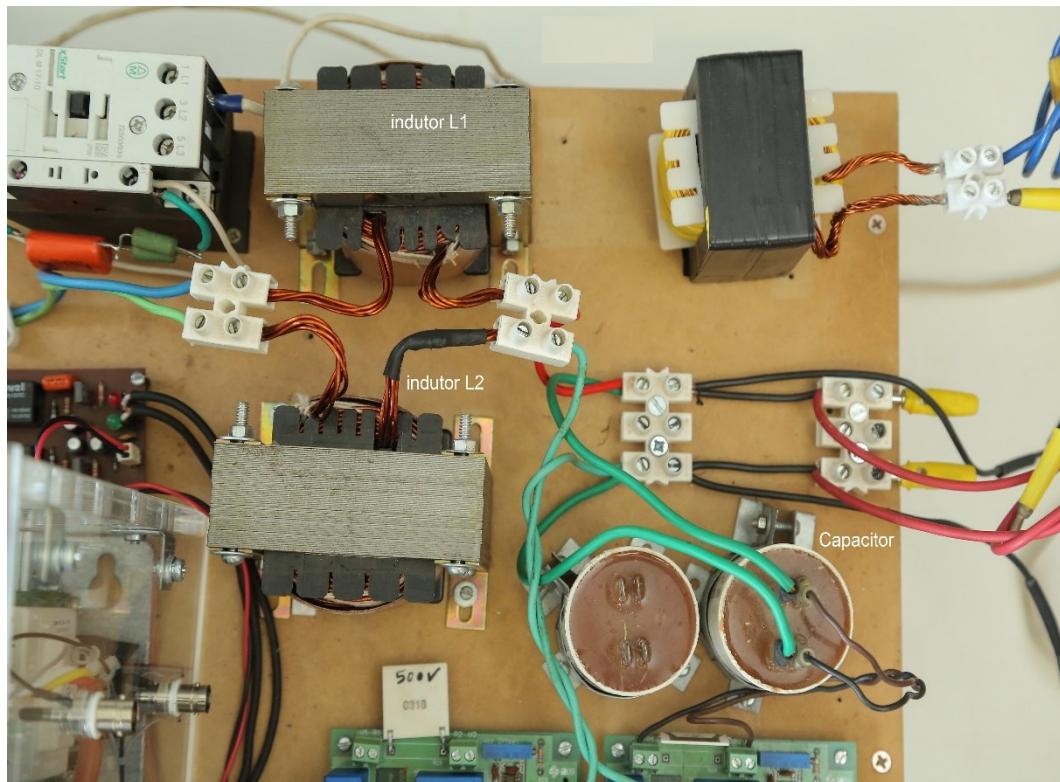


Figura 44 - Filtro LC

5.5 Módulo da Semikron

No anexo F a ponte de IGBT e respectivos diodos de roda livre são integrados em um módulo SemiKron utilizado no projeto, o qual pode ser visto na Figura 45. Além do circuito de potência o módulo integra os *gate drivers* para comandar os IGBTs, os quais são conectados à FPGA via placa conversor de sinais apresentada na Figura 40.

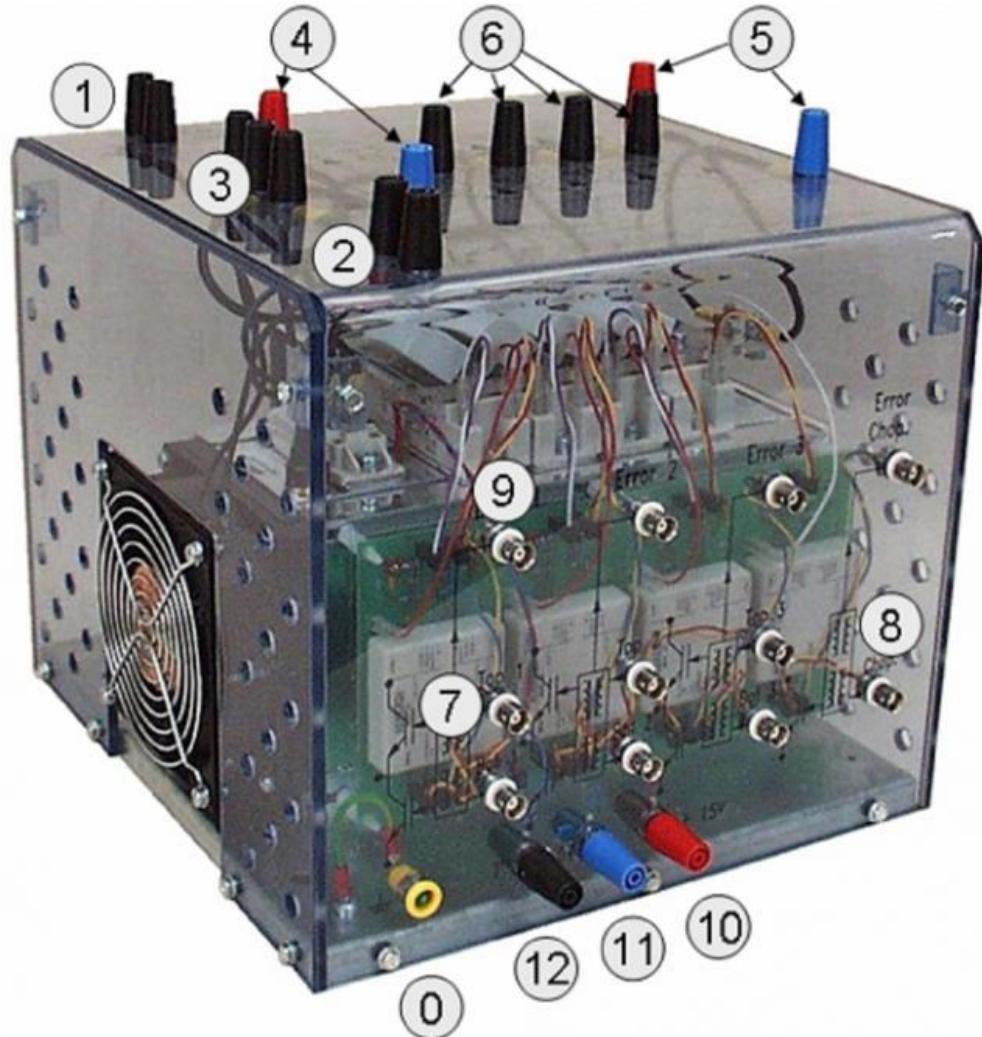


Figura 45 - Módulo SemiKron

5.6 FPGA

O módulo FPGA utilizado já foi apresentado em capítulo anterior, neste momento apenas a figura da FPGA será apresentada com as duas ligações principais destacadas, a entrada do conversor AD conectado à placa condicionadora de sinal e a saída dos pulsos para os *gate drivers* (Figura 46).

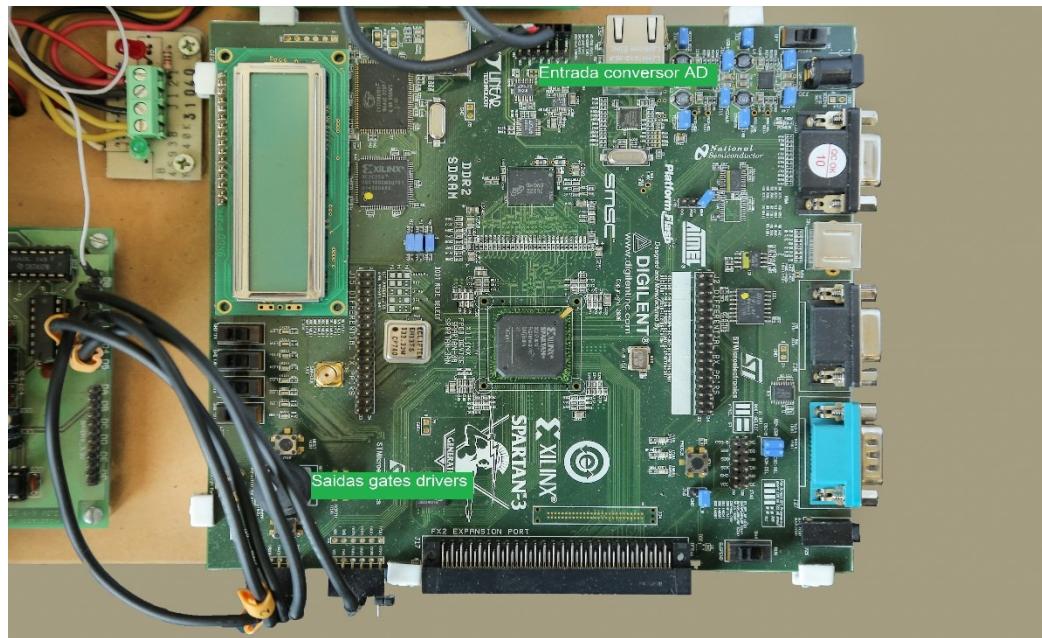


Figura 46 - FPGA e conexão com AD e saída Gate drivers

Para melhor compreensão da planta, apresenta-se na Tabela 4 os seus parâmetros gerais.

Tabela 4 - Parâmetros da Planta

Parâmetros da Planta	
Barramento CC	VCC=200V
Tensão de referência	Vref =110 Vrms, f = 50 Hz
Indutor do filtro de saída	L=1mH
Capacitor do filtro de saída	C=30 μ F
Frequência de amostragem	5000 Hz
Carga não-linear –retificador não controlado	Ponte de diodos RL=94 Ω CL=470 μ F
Parâmetros da FPGA	
Slices	57
Flip flop	37
LUTs	105
IOBs	9
Resolução do ponto fixo	s.10.21 (Sinal, Inteiro, fração)

6 RESULTADOS

Os resultados descritos a seguir foram obtidos ao longo de toda a pesquisa e serão apresentados na ordem em que foram gerados.

6.1 Resposta da Função de Transferência do Repetitivo e PD-feedforward no Sistema

A comparação entre resultados gerados pelo PSIM e o *System Generator* foi muito importante para o fornecimento de informações para avaliar se o trabalho estava no caminho certo, gerando confiança na utilização da FPGA.

As figuras 47 e 48 apresentam os sinais de entrada e saída para a função de transferência do PD-*feedforward* gerado pelo PSIM e System Generator.

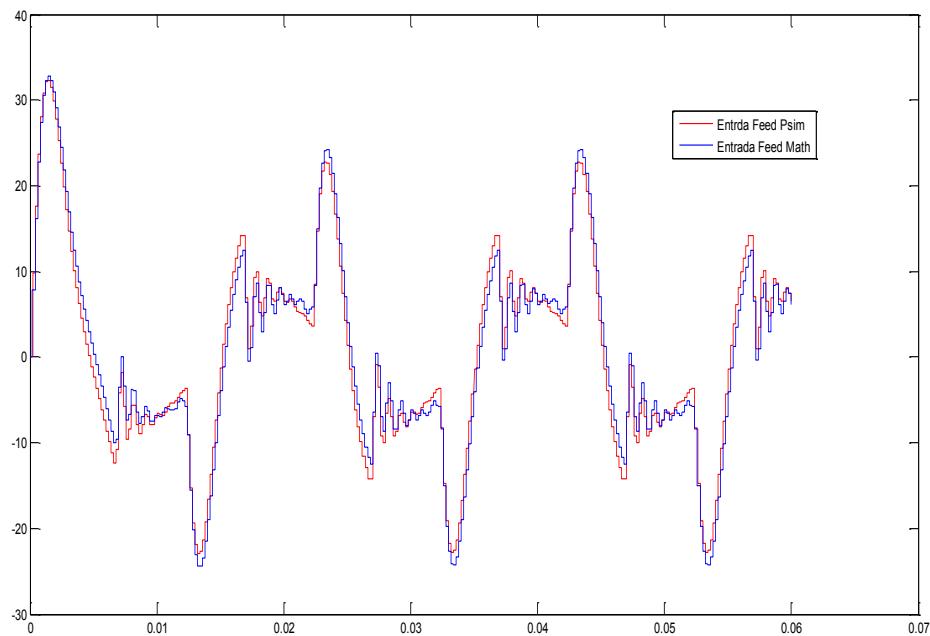


Figura 47 - Sinais de entrada na função Feed Forward no PSIM e System Generator

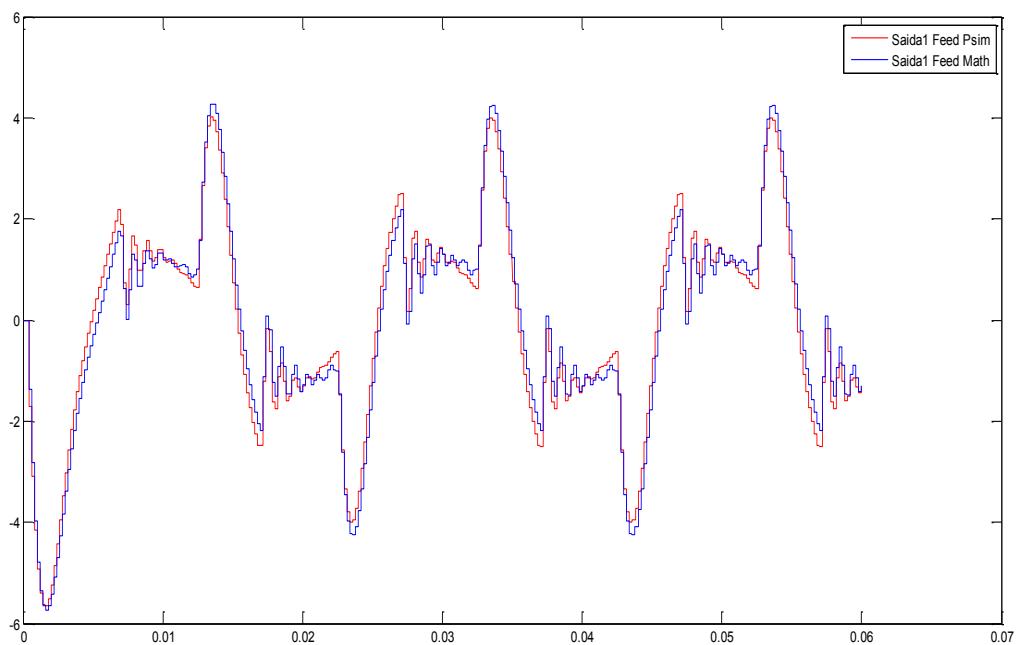


Figura 48 - Sinais de saída na função Feed Forward no PSIM e System Generator

Este resultado foi conseguido simulando 70 ms, neste caso em mais de três ciclos. Os resultados confirmam que a FPGA está trabalhando corretamente, em fase e amplitude, e que as operações estão sendo realizadas corretamente com a simulação realizada no Psim. Onde foi realizada a primeira simulação da planta sendo a referência para a simulação no System generator.

6.2 Resposta ao Degrau da Função de Transferência do Repetitivo e PD-feedforward

O primeiro resultado foi o teste das funções de transferência. Para verificar o comportamento das funções de transferências, após montá-las as mesmas eram submetidas ao teste da resposta ao degrau. Foram realizados dois testes de cada função de transferência, sendo o primeiro no PSIM e depois no *System Generator* dentro do ambiente do Simulink. As Figuras 49 e 50 mostram a forma de onda do *PD-feedforward* no PSIM e no *System Generator*, respectivamente.

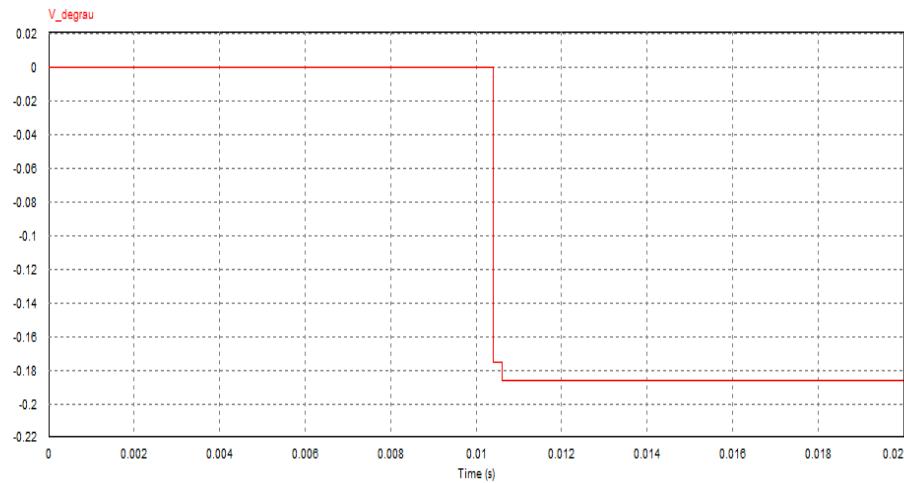


Figura 49 - Resposta ao Degrau da função de transferência Feed Forward no PSIM

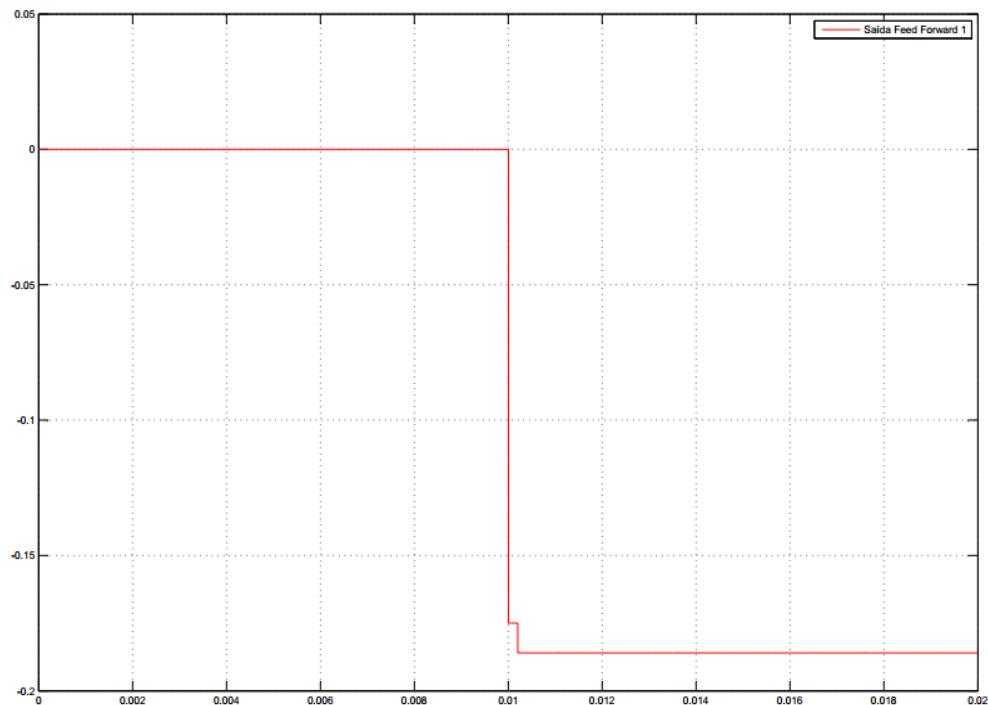


Figura 50 - Resposta ao Degrau da função de transferência Feed Forward no System Generator

A função de transferência do controle Repetitivo tem como resultado apenas um atraso de 20ms quando submetido à função degrau, conforme mostrado nas Figuras 51 e 52.



Figura 51 - Resposta ao degrau no PSIM para a função de transferência Repetitiva

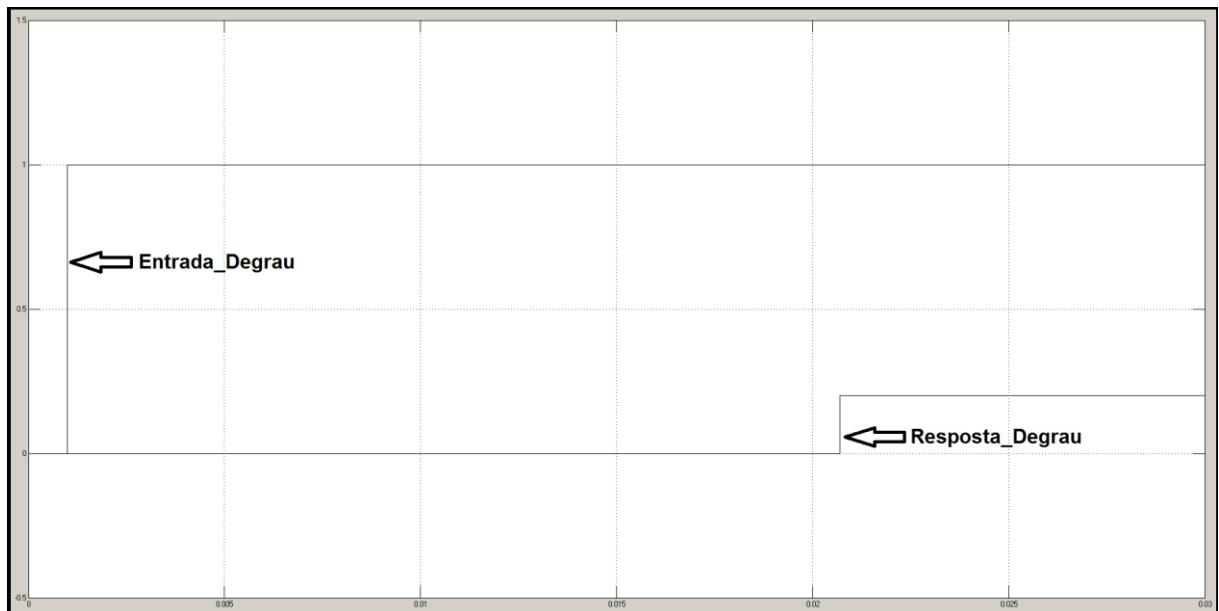


Figura 52 - Resposta ao degrau no System Generator para a função de transferência Repetitiva

6.3 Resultado da Simulação no PSIM

O resultado apresentado na Figura 53 é parte de uma simulação de 1s, apresentando os últimos ciclos da simulação no PSIM. Os sinais que estão sendo comparados são o sinal de tensão de referência e o sinal de tensão da saída para o controle Repetitivo e PD-feedforward.

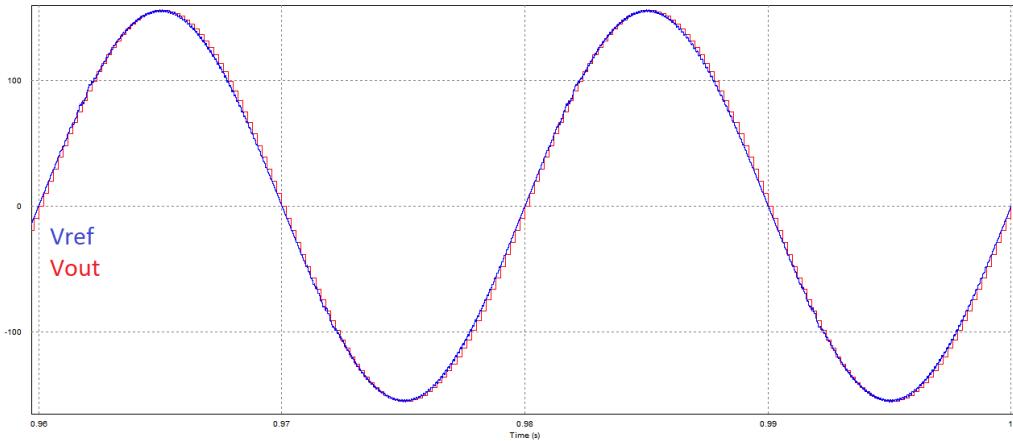


Figura 53 - Resultado da simulação do controle Repetitivo e PD-feedforward no PSIM com tensão de referência e de saída

A Figura 54 apresenta os resultados de simulação para a tensão de referência, tensão de saída e corrente de saída do inversor, com detalhe do transitório de carga em zoom. Onde uma grande distorção da corrente resulta em pouquíssima distorção da forma de onda da tensão de saída. A carga não linear requisita uma corrente com THD de 123,6%, no entanto, o controle foi capaz de impor uma tensão de saída do inversor com THD de 0,64%. Pode-se, então, afirmar que o controlador PD-feedforward com ação de controle repetitiva apresenta um bom desempenho na rejeição de distúrbios repetitivos, como é o caso da carga não linear utilizada.

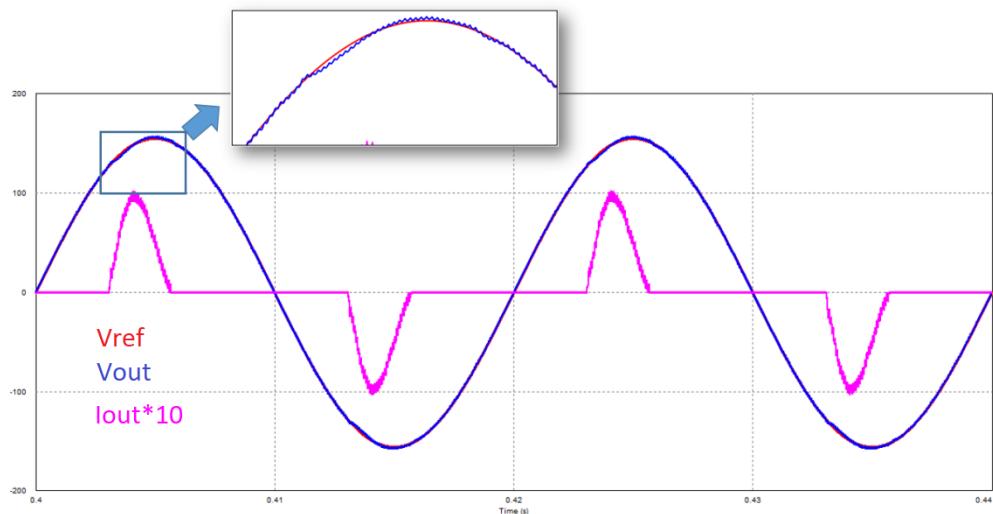


Figura 54 - Resultado da simulação no PSIM do controle Repetitivo e PD-feedforward com o sinal de tensão e corrente de saída

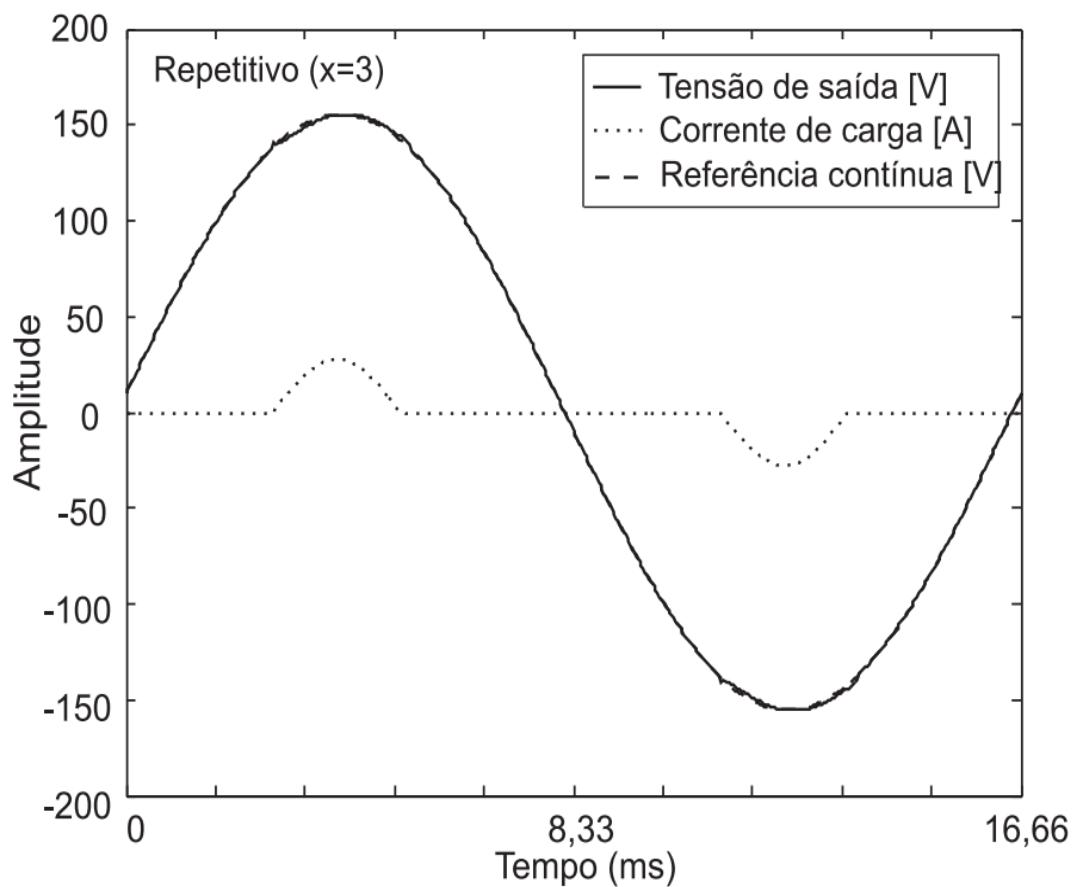


Figura 55 - Resultado da referência [8] para comparação

Visando uma comparação de resultados, a Figura 55 mostra os resultados apresentados em [8] para tensão de saída do inversor e a corrente de carga, considerando uma carga não linear referente a um retificador não controlado com resistor de 28Ω e capacitor de $4700\mu F$. É importante frisar que em função da frequência base de 50 MHz disponível na placa Kit Spartan3an, a fim de gerar um submúltiplo exato para a composição do "buffer" de 100 posições do controle repetitivo, optou-se por utilizar uma referência senoidal em 50Hz, a qual difere da frequência de 60 Hz da tensão de saída da Figura 55. Além disso, a carga não linear utilizada em [8] é diferente daquela usada no resultado apresentado na Figura 54, mas ambas resultam num considerável nível de distorção para a corrente de saída. Apesar de tais diferenças, comparando então as Figuras 54 e 55, pode-se observar que o desempenho do controle PD-feedforward com ação repetitiva implementado nesta pesquisa é tão satisfatório quanto aquele apresentado em [8].

6.4 Resultado da Simulação no System Generator

A Figura 56 mostra os resultados com os sinais da tensão de saída e tensão de referência, agora no System Generator, para o tempo 0.5 s de simulação. Como pode ser observado o resultado gerado pelo System Generator apresenta uma correspondência significativa com o resultado do PSIM mostrado na Figura 53.

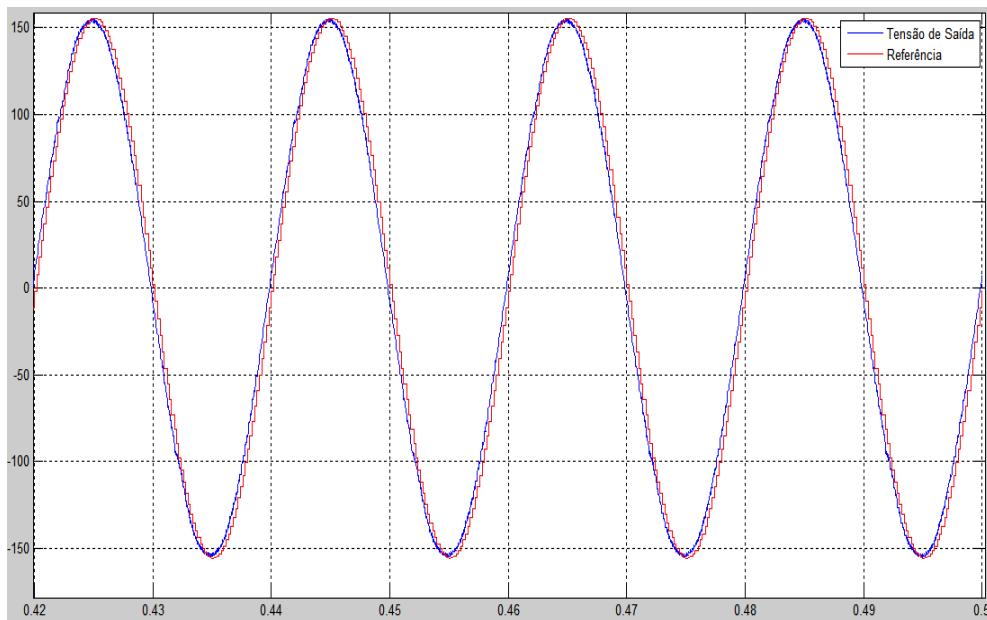


Figura 56 - Resultado da simulação do controle Repetitivo no System Generator com o sinal de referência e tensão de saída.

6.5 Resultado da co-simulação no System Generator

Uma das facilidades disponibilizadas pelo System Generator/Matlab/Simulink é a operação chamada de *hardware-in-loop* (HIL), onde tem-se a planta simulada em ambiente Simulink sendo comandada pelo controle real embarcado na FPGA. Blocos de interface específicos disponibilizados pelo software de desenvolvimento permitem a transição de sinais entre a planta simulada e o controle embarcado na FPGA. A Figura 56 apresenta o resultado da tensão de saída e tensão de referência do inversor realizada através da co-simulação, ou seja, HIL, para o sistema operando com as mesmas configurações do resultado da Figura 53, mas com o controle embarcado na FPGA. Comparando as Figuras 56 e 57, pode-se dizer que o controle embarcado na FPGA reproduz adequadamente o controle simulado no

PSIM. Desta forma, pode-se avaliar se o controle real funciona adequadamente sem comprometer a planta real, em caso de falha ou projeto equivocado.

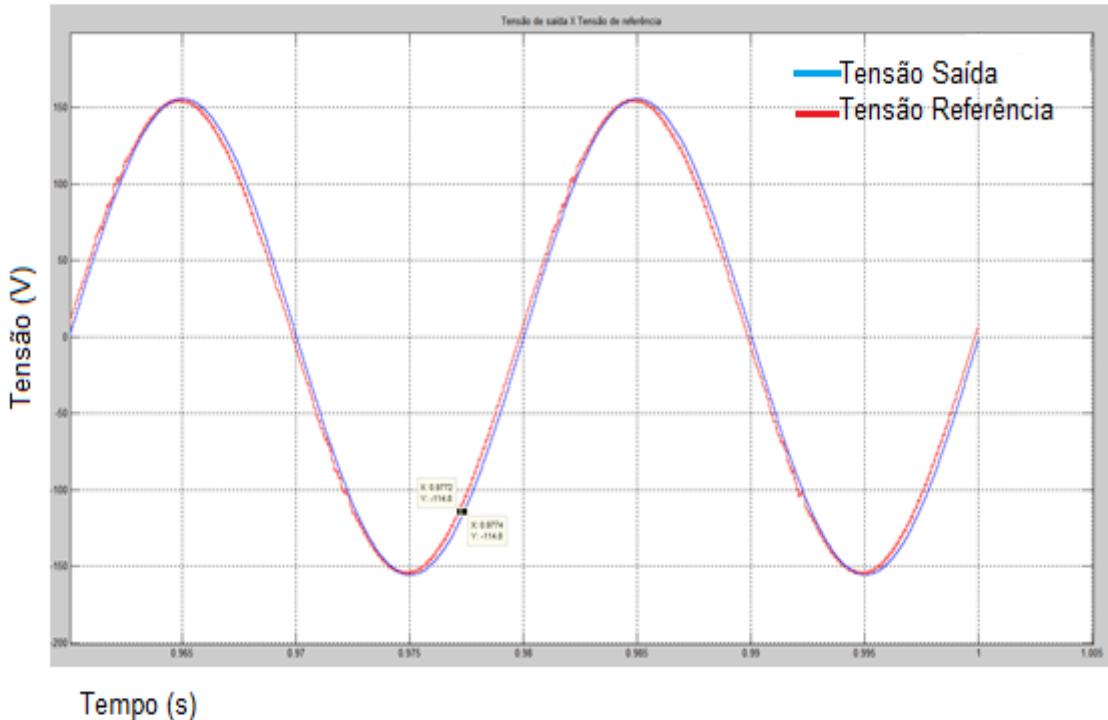


Figura 57 - Resultado da co-simulação do controle Repetitivo com os sinais de tensão de saída e tensão de referência

6.6 Resultados experimentais do controle Repetitivo implementado

Em razão da disponibilidade de recursos de laboratório, alguns parâmetros da aplicação foram configurados de forma distinta daquela encontrada em [8]. Os parâmetros utilizados na obtenção dos resultados são apresentados na Tabela 4.

A Figura 58 apresenta a tensão de saída do inversor para a operação em malha aberta. Observa-se que a modulação PWM funciona a contento, sendo a frequência de saída condizente com o sinal modulante, mas a presença da carga não linear provoca fortes distorções na tensão de saída do inversor, como era de se esperar. Este ensaio serviu também para a verificação das polaridades do sistema de aquisição, de forma a configurar adequadamente a realimentação negativa nos ensaios em malha fechada

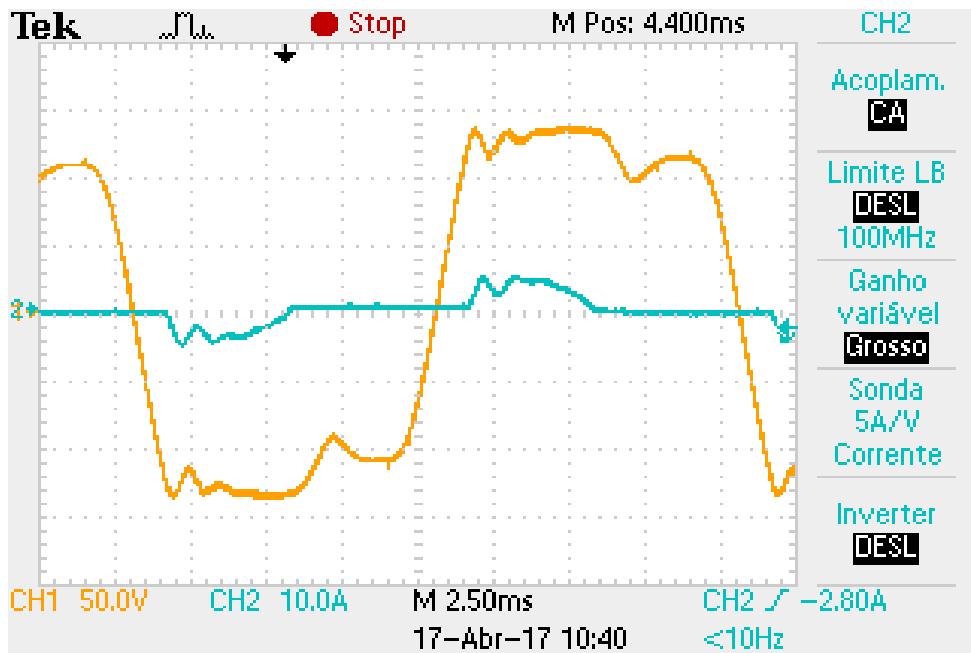


Figura 58 - Resultado da planta em malha aberta para tensão e corrente de saída

O resultado para a operação em malha fechada do conversor é apresentado na Figura 59. Como a amostragem do osciloscópio não é sincronizada com o chaveamento do inversor, nota-se a presença de alguns “spikes”, principalmente na forma de onda de corrente, os quais não estão presentes nos sinais reais, sendo apenas ruídos da medição. Neste caso, a tensão de saída do inversor apresenta uma THD de 1.69% para uma THD da corrente de saída de 128.74%, assim, em termos de especificação de THD de tensão, a unidade inversora atende às normas IEEE 519 (máximo de 5 %) e IEC 62040-3 (máximo de 8%).

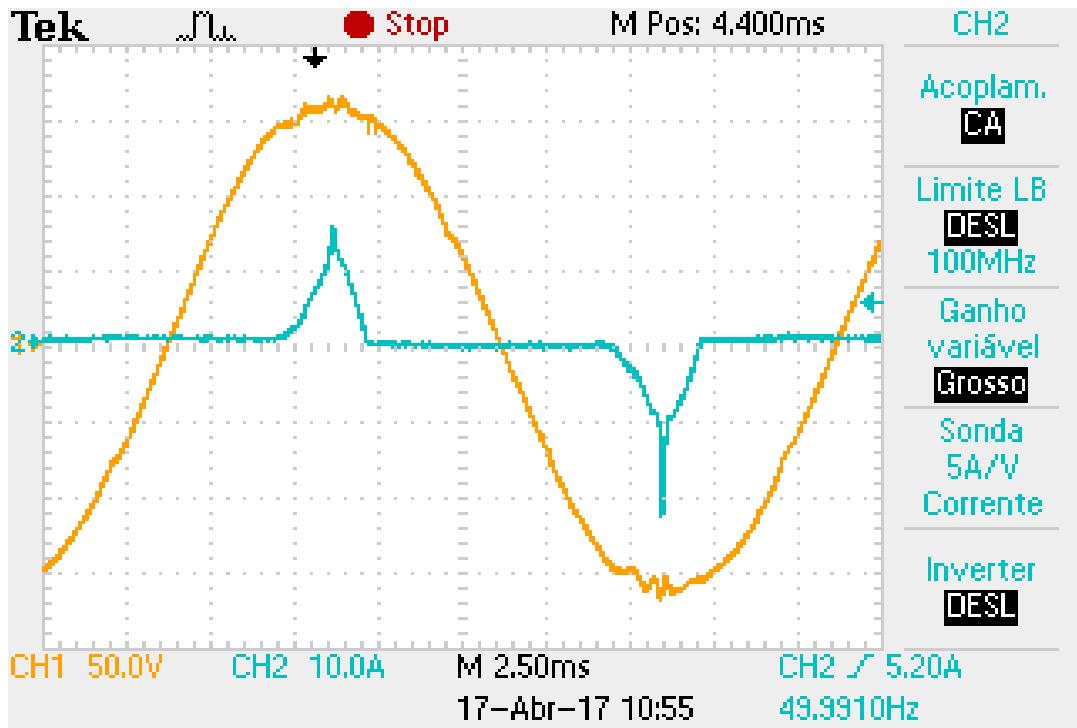


Figura 59 - Resultado em malha fechada, controle Repetitivo, tensão e corrente de saída

Os espectros harmônicos da tensão de saída e da corrente de saída do inversor são apresentados nas Figuras 60 e 61, respectivamente. Como pode ser observado, o espectro harmônico da tensão também atende aos limites específicos por harmônicos da norma IEC 62040-3, onde, por exemplo, tem-se os limites de 5%, 6%, 5% e 3,5% para os harmônicos de ordem 3, 5, 7 e 11, respectivamente.

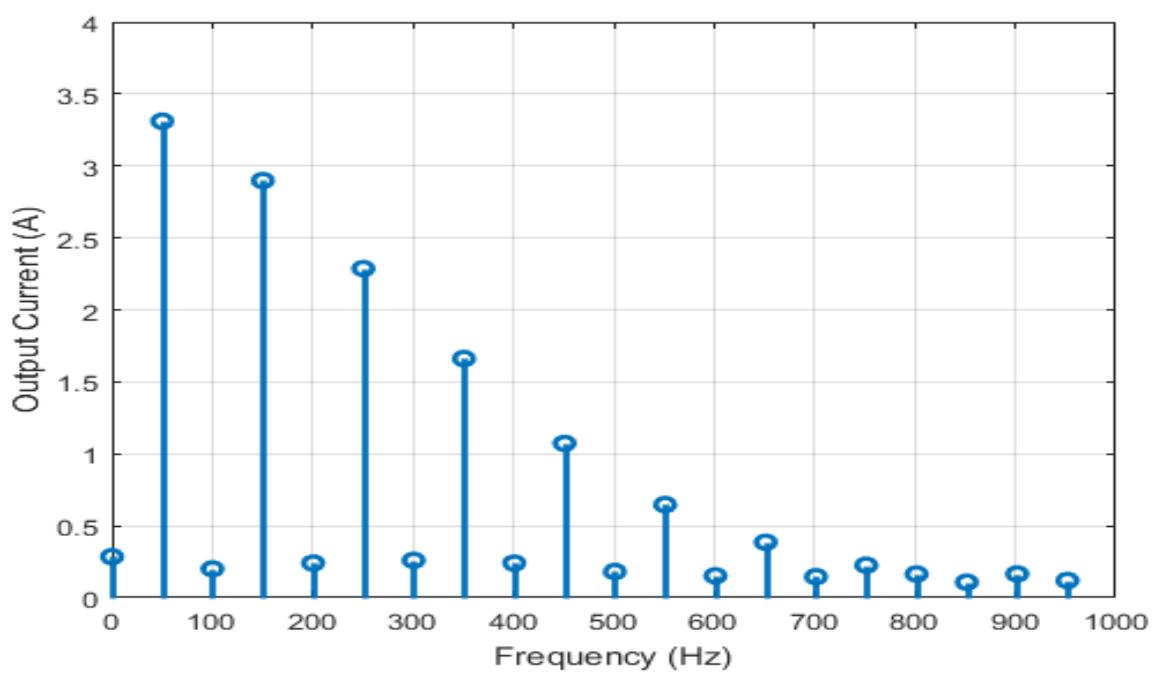


Figura 60 - Espectro harmônico da corrente

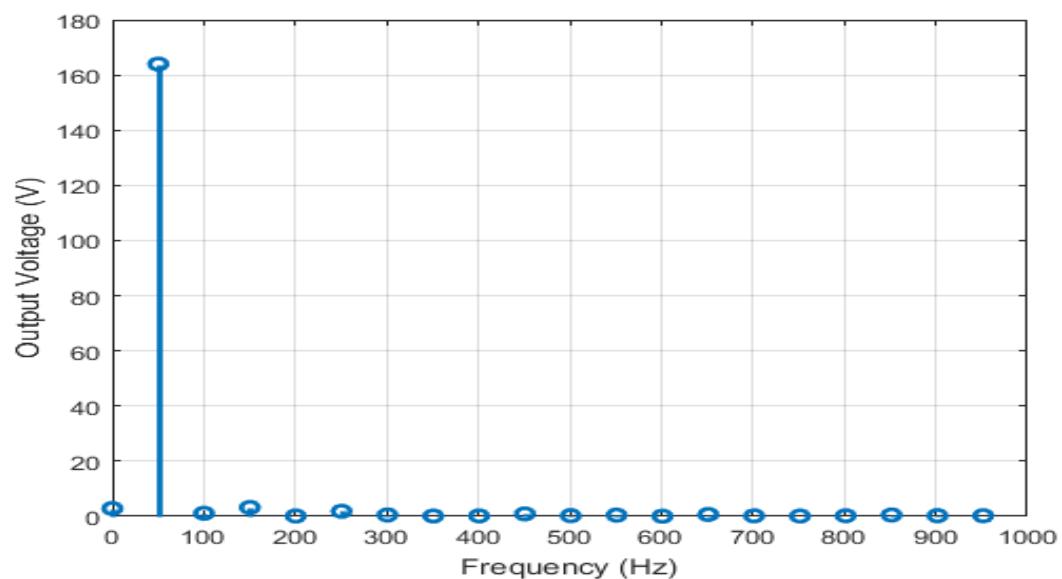


Figura 61 - Espectro harmônico da tensão de saída

7 CONCLUSÃO E PROPOSTA PARA TRABALHOS FUTUROS

O presente trabalho apresentou um controlador digital completamente embarcado em FPGA, com ações de controle PD-Feedforward e repetitiva, visando ao controle da tensão de saída de um inversor PWM senoidal. Foi apresentada uma breve descrição da estrutura de uma FPGA e das ferramentas usuais para sua programação, assim como a descrição da planta, da modulação PWM unipolar e dos respectivos controladores PD-Feedforward e repetitivo.

O desenvolvimento de um controlador a ser embarcado em FPGA não é tão simples quanto a elaboração de código a ser embarcado em DSP. Via de regra, em aplicações de Eletrônica de Potência, os controladores embarcados em DSP's são implementados por meio de serviços de interrupção gerados na taxa de amostragem do processo, os quais contemplam cálculos sequenciais realizados numa única unidade aritmética. As FPGA's permitem a elaboração de cálculos em paralelo, sendo que, por exemplo, todos os termos de um produto de uma função digital podem ser obtidos simultaneamente. Este fato traz uma preocupação adicional com o sincronismo, de forma a fazer com que todas as informações geradas em ramos paralelos cheguem no tempo correto ao nó do processo onde serão utilizadas. Este trabalho contribui na discussão deste aspecto para a implementação dos controladores PD-Feedforward e repetitivo. É importante frisar, nas questões de sincronismo, que também devem ser considerados os atrasos inerentes à planta. No caso do projeto em questão, existem atrasos inerentes à modulação PWM, à conversão analógica-digital e à transmissão de dados do conversor analógico-digital para a FPGA, a qual é feita de forma serial.

O trabalho mostrou que o uso de ferramentas com alto nível de abstração facilita a implementação do controlador e acelera o desenvolvimento da aplicação. No entanto, nem sempre é possível realizar todas as funções necessárias ao projeto por meio de blocos dentro do ambiente de alto nível de abstração. Para a implementação do controle do inversor, foi necessário recorrer a ferramentas de baixo nível de abstração, no caso a linguagem VHDL, para a realização de funções específicas. Felizmente, nas situações em que foi necessária a utilização do código

VHDL, as respectivas funções a serem implementadas eram de pequena ordem, com relativa facilidade de implementação.

Visando validar os primeiros códigos embarcados em FPGA, foram realizados ensaios de simulação e de co-simulação, sendo que nestes últimos é executado o controlador real a ser usado nos experimentos. Isto é importante, pois garante ao projetista que o controlador embarcado atende às especificações de projeto, eliminando possíveis problemas que possam existir no controlador antes da sua aplicação na planta real.

Os resultados experimentais apresentados demonstram que o controlador embarcado em FPGA opera adequadamente, sendo capaz de regular a tensão de saída para uma carga não linear, mantendo-a dentro dos limites de conformidade com as normas vigentes.

A partir dos resultados deste trabalho, espera-se que as FPGA's sejam incorporadas como alternativa de implementação de controladores digitais nas pesquisas do Laboratório de Eletrônica de Potência da UFU, onde hoje os DSP's são predominantemente utilizado, e as FPGA's se destinam à mera substituição de circuitos lógicos discretos, como portas lógicas, flip-flops, etc. De fato, este é um trabalho pioneiro no Laboratório de Eletrônica de Potência, onde as FPGA's foram utilizadas para a elaboração de cálculos matemáticos em ponto-fixo.

Como continuidade da presente pesquisa, os seguinte tópicos podem ser elencados:

-Elevação da taxa de amostragem e aumento do tamanho do "buffer" do controlador repetitivo, visando à melhora da resposta dinâmica do inversor;

-Implementação de outros controladores para a mesma aplicação, como o proporcional-resonante e controladores baseados no modelo interno com um e dois graus de liberdade, e respectiva comparação de desempenho;

-Implementação do controlador para um inversor trifásico;

-Implementação, em FPGA, dos controladores internos, primário e secundário, para uma microrrede.

REFERÊNCIAS

- [1] BRASIL. Ministério de Minas e Energia (MME). PROINFA 2014. Disponível em: <<http://www.mme.gov.br/programas/proinfa>>. Acesso em: 26 jul. 2014.
- [2] Energias não renováveis. Disponível em: < <http://www.ageneal.pt/content01.asp?BTreelD=00/01&treelD=00/01&newsID=7>>. Acesso em: 8 dez. 2017.
- [3] Vantagens e desvantagens da energia eólica. Disponível em: <<https://www.portal-energia.com/vantagens-desvantagens-da-energia-eolica/>>. Acesso em: 8 dez. 2017.
- [4] Energia solar: vantagens e desvantagens. Disponível em: <<http://portaldaenergia.com/energia-solar-vantagens-e-desvantagens>>. Acesso em: 8 dez. 2017.
- [5] BIMAL, K. B. (Ed.). *Power Electronics and Variable Frequency Drives: Technology and Applications*. Wiley-IEEE Press, 1996. ISBN: 978-0-7803-1084-1.
- [6] CÂMARA, R. A. *Análise comparativa de desempenho de conversores CA-CC monofásico utilizando FPGA para aplicação em NO-BREAKS*. 2012. 205f. Tese (Doutorado em Engenharia Elétrica) – Centro de Tecnologia, Universidade Federal do Ceará, Fortaleza, 2012.
- [7] KAHL, T.; DIECKERHOFF, S. Comparison of FPGA - and microcontroller - based Control of a High-Dynamic Power Electronic Converter. In: 2017 IEEE 18th WORKSHOP ON CONTROL AND MODELING FOR POWER ELECTRONICS (COMPEL). Disponível em: <<http://ieeexplore.ieee.org/document/8013288/>>. Acesso em: 2016.
- [8] MICHELS, L. *Metodologia de projeto de fontes Ininterruptas de energia monofásicas empregando controladores de ação repetitiva auxiliar no estágio de saída*. 2006. 235f. Tese (Doutorado em Engenharia Elétrica) - Centro de Tecnologia, Universidade Federal de Santa Maria, Santa Maria, 2006.
- [9] PROGRAMMABLE logic device. Disponível em: <https://en.wikipedia.org/wiki/programmable_logic_device>. Acesso em: 26 abr. 2017.

- [10] ORDONEZ, E. D. M.; PEREIRA, F. D.; PENTEADO, C. G.; PERICINI, R. A. *Projeto, desempenho e aplicações de sistemas digitais em circuitos programáveis (FPGAs)*. São Paulo: Bless, 2003. 300 p.
- [11] COVER Story. Three articles discuss the history of Xilinx and the programmable logic industry. *The Quarterly Journal for Programmable Logic Users*, X CELL Issue 32, San Jose, CA, Second Quarter, 1999.
- [12] TRIMBERGER, S. M. Three Ages of FPGAs: A Retrospective on the First Thirty Years of FPGA Technology. *Proceedings of the IEEE*, [s.l.], v. 103, n. 3, p. 318-331, mar. 2015. Disponível em: <<http://dx.doi.org/10.1109/jproc.2015.2392104>>. Acesso em: 12 mar. 2015. <https://doi.org/10.1109/JPROC.2015.2392104>
- [13] CHU, P. P. *FPGA Prototyping by VHDL Examples*: Xilinx Spartan-3 Version. Cleveland State University. John Wiley & Sons, 2008. <https://doi.org/10.1002/9780470231630>
- [14] Spartan-3AN Starter Kit. Disponível em:<<https://www.xilinx.com/products/boards-and-kits/hw-spar3an-sk-uni-g.html>>. Acesso em: 20 abr. 2017.
- [15] POMILIO, J. A.; PAREDES, H. K. M.; DECKMANN, S. M. *Eletrônica de potência para geração, transmissão e distribuição de energia elétrica*. Julho/2017. Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação. Departamento de Sistemas e Energia. Disponível em: <<http://www.unicamp.br/dse/antenor/it744>>. Acesso em: 24 fev. 2016.
- [16] RASHID, M. H. *Eletrônica de potência: circuitos, dispositivos e aplicações*. Tradução de Carlos Alberto Favato. São Paulo: Makron Books, 1999.
- [17] NAMBOODIRI, A. (UG Student); WANI, H. S. (Assistant Professor). Unipolar and Bipolar PWM Inverter. *IJIRST - International Journal for Innovative Research in Science & Technology*, v., n. 7, Dec. 2014.
- [18] YU, Z.; MOHAMMED, A.; PANAHY, I. A Review of Three PWM Techniques. Proceedings of the *American Control Conference*, Albuquerque, New Mexico, v. 1, p. 257-261, 1997. <https://doi.org/10.1109/ACC.1997.611797>
- [19] SILVA, A. F. B. O.; SILVA, S. M.; SANTOS, C. H. G.; CARDOSO FILHO, B. J. Aplicação do controle repetitivo a inversor PWM monofásico com filtro LC de saída utilizado em fonte programável CA. *Eletrônica de Potência*, v. 18, n. 4, p. 1161-1169, nov. 2013. Disponível em: <http://www.sobraep.org.br/sobraep_opmain.php>. Acesso em: 19 ago. 2016.

- [20] RYAN, M. J.; BRUMSICKLE, W. E; LORENZ, R. D. Control topology options for single-phase ups inverters. *IEEE Transactions on Industry Applications*, v. 33, n. 2, p. 493-501, mar./abr. 1997. <https://doi.org/10.1109/28.568015>
- [21] YEPES, A. G. *Digital resonant current controllers for voltage source converters*. 2011. 215f. Tese (Doutorado em Filosofia) - Department of Electronics Technology, University of Vigo, Vigo, Spain, 2011. Disponível em:<<https://pdfs.semanticscholar.org/140c/1aa55e9ef8698aa56a24e278dd69520095d9.pdf>>. Acesso em: 19 set. 2016.
- [22] SCHAUDER, C. D.; CADDY, R. Current control of voltage-source inverters for fast four quadrant drive performance. *IEEE Transactions on Industry Applications*, v. IA-18, n. 2, p. 163-171, mar. 1982. <https://doi.org/10.1109/TIA.1982.4504051>
- [23] SCHAUDER, C. D.; MORAN, S. A. Multiple reference frame controller for active filters and power line conditioners. *U.S. Patent*, n. 5, p. 309-353, May 1994.
- [24] VUKOSAVIC, S. N. *Digital Control of Electrical Drives*. New York: Springer, 2007. 353 p.
- [25] FRANCIS, B. A.; WONHAM, W. M. The internal model principle for linear multivariable regulators. *Applied Mathematics and Optimization*. Springer-Verlag, New York, v. 2, n. 1, p. 170-194, jan. 1975.
- [26] SILVA, A. F. B. O. *Aplicação de controle repetitivo em inversor PWM monofásico com filtro LC de saída utilizado em fonte programável CA*. 2012. 70f. Dissertação (Mestrado em Engenharia Elétrica) – CEFET-MG/Universidade Federal de São João Del-Rei, Belo Horizonte, 2012.
- [27] TOMIZUKA, M. Zero phase error tracking algorithm for digital control. *Journal of Dynamic Systems, Measurement and Control*, American Society of Mechanical Engineers, New York, v. 109, p. 65-68, mar. 1987.
- [28] KEMPF, C.; MESSNER, W.; TOMIZUKA, M. Comparison of four discrete-time repetitive control algorithm. *IEEE Control Systems Magazine*, The Institute of Electrical and Electronic Engineers, Piscataway, USA, v. 13, n. 6, p. 48-54, dez. 1993.

- [29] BROBERG, H. L.; MOLYET, R. G. A new approach to phase cancellation in repetitive control. In: IEEE IAS, 1994. *Conference Record of the 1994 IEEE Industry Applications Society Annual Meeting*, Piscataway, 1 USA, 994. p. 1766-1770. <https://doi.org/10.1109/IAS.1994.377667>
- [30] HILLERSTROM, G.; STERNBY, J. Repetitive control using low order models. In: IFAC ACC, 1994. *Proceedings of the American Control Conference*, Dayton, USA, 1994. p. 1873-1878. <https://doi.org/10.1109/ACC.1994.752398>
- [31] AMANUMA, K.; FUWA, M.; SAKAKI, Y. High accurate ripple reducing method based on the repetitive control. In: IEEE PESC, 1994. *Conference Record of Power Electronics Specialists Conference*, Piscataway, USA, 1994. p. 571-576. <https://doi.org/10.1109/PESC.1994.349679>
- [32] YAMADA, M.; FUNAHASHI, S. I. Y.; MATSUSHITA, M. Extended discrete-time prototype repetitive controllers and its application. In: IEEE CDC, 1996. *Proceedings of the 35th IEEE Conference on Decision and Control*, Piscataway, USA, 1996. p. 3606-3611. <https://doi.org/10.1109/CDC.1996.576956>
- [33] SMITH, C.; TOMIZUKA, M. Shock rejection for repetitive control using a disturbance observer. In: IEEE CDC, 1996. *Proceedings of the 35th IEEE Conference on Decision and Control*, Piscataway, USA, 1996. p. 2503-2504. <https://doi.org/10.1109/CDC.1996.573469>
- [34] LI, J.; TSAO, T. C. A two parameters robust repetitive control design using structured singular values. In: IEEE CDC, 1998. *Proceedings of the 37th IEEE Conference on Decision and Control*, Piscataway, USA, 1998. p. 1230-1235.
- [35] KIM, B. S.; LI, J.; TSAO, T. C. Two-parameter robust repetitive control with application to a novel dual-stage actuator for noncircular machining. *IEEE/ASME Transactions on Mechatronics*, The Institute of Electrical and Electronic Engineers, Piscataway, USA, v. 9, n. 4, p. 644-652, dez. 2004.
- [36] LI, M.; LI, W.; LIU, C.; WEI, J.; CHEN, T. In: THE 6TH INTERNATIONAL FORUM ON STRATEGIC TECHNOLOGY, 2011. Comparison of Two Repetitive Control Strategies of UPS Inverter on Saber. School of Electrical and Electronic Engineering. Harbin University of Science & Technology, Harbin, China, 2011.
- [37] BROLILOW, C., JOSEPH, B. *Techniques of Model-Based Control*. New York: Prentice Hall, 2002.

[38] CARNEIRO, L. S. *Um circuito modulador de largura de pulso (DPWM) utilizando FPGA*. 2015. 81f. Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal de Itajubá, Itajubá, 2015.

[39] XILINX. *System Generator for DSP User Guide*. UG640 (v11.4) December 2, 2009. Disponível em:<https://www.xilinx.com/support/documentation/sw_manuals/xilinx11/sysgen_user.pdf>. Acesso em: 25 fev. 2015.

[40] VAN DER BROECK, H.; MILLER, M. Harmonics in dc to ac converters of single phase uninterruptible power supplies. In: IEEE INTELEC, 1995. *Proceedings of the 17th Annual International Telecommunications Energy Conference*, Piscataway, USA, 1995. p. 653-658. <https://doi.org/10.1109/INTLEC.1995.499027>

[41] BOOST, M. A.; ZIOGAS, P. D. State-of-the-art carrier pwm techniques: a critical evaluation. *IEEE Transactions on Industry Applications*, The Institute of Electrical and Electronic Engineers, Piscataway, v. 24, n. 2, p. 271-280, mar./abr. 1988.

[42] BOTTERÓN, F. et al. Digital voltage and current controllers for three-phase pwm inverter for ups applications. In: IEEE IAS, 2001. *Conference Record of the 37th IAS Annual Meeting Industry Applications Conference*, Piscataway, USA, 2001. p. 2667-2674. <https://doi.org/10.1109/IAS.2001.955995>

BIBLIOGRAFIA CONSULTADA

BAEZA, J. R. *Controle não linear aplicado a malhas de controle com válvula de alto atrito*. 2013. 91f. Dissertação (Mestrado em Engenharia) - Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Telecomunicações e Controle, Universidade de São Paulo, São Paulo, 2013.

BRITO, M. A. G. *Pré-regulador retificador Boost com controle digital por valores médios, para sistema de iluminação fluorescente multi-lâmpadas, utilizando dispositivo FPGA e VHDL*. 2008. 200f. Dissertação (Mestrado em Engenharia Elétrica) - Faculdade de Engenharia de Ilha Solteira - FEIS/UNESP, Ilha Solteira, 2008.

CHAVES, E. N. *Otimização meta heurística e controle baseado no modelo interno aplicados em sistema de geração fotovoltaica à rede elétrica monofásica*. 2016. 222f. Tese (Doutorado em Engenharia Elétrica) - Programa de Pós-graduação em Engenharia Elétrica, Universidade Federal de Uberlândia, Uberlândia, 2016.

GNOATTO, C. L. Estratégias de controle repetitivo para aplicação em sistemas de alimentação ininterrupta de energia. 2011. 85f. Dissertação (Mestrado em Engenharia Elétrica) - Programa de Pós-graduação em Engenharia Elétrica, Universidade Tecnológica Federal do Paraná, Pato Branco, 2011.

MARQUES, F. N. *Inversor flyback a quarto transistor controlado por um dispositivo FPGA para obter MPPT*. 1980. 128f. Dissertação (Mestrado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia, Uberlândia, 1980.

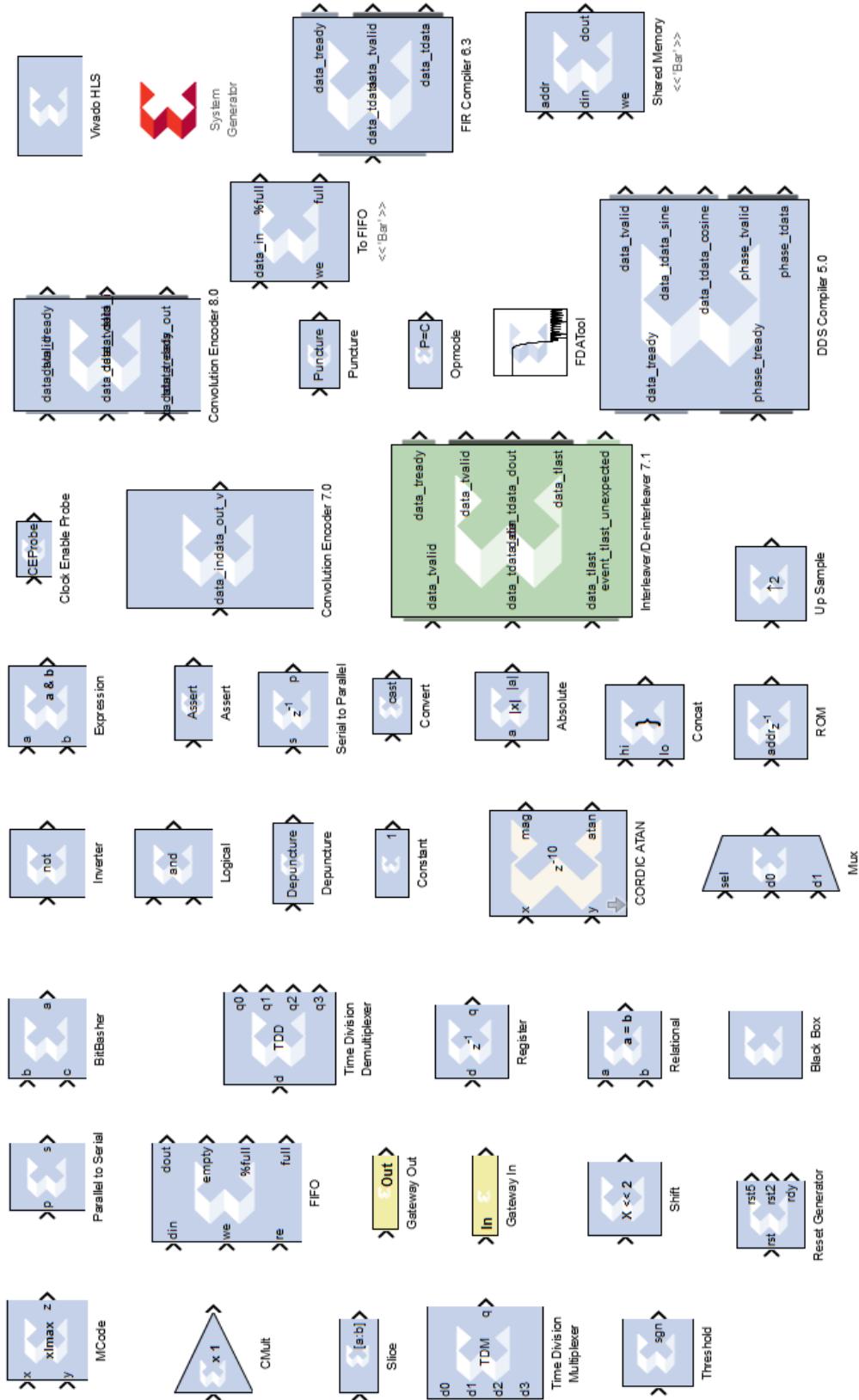
PANKAJ LAHARI, M. V.; VEDULA, S. V.; UMAMAHESWARA RAO, V. Real Time Models for FPGA based Control of Power Electronic Converters: A Graphical Programming Approach. In: 2015 IEEE IAS JOINT INDUSTRIAL AND COMMERCIAL POWER SYSTEMS / PETROLEUM AND CHEMICAL INDUSTRY CONFERENCE (ICPSPCIC). Disponível em: <<http://ieeexplore.ieee.org/document/7974054/>>. Acesso em: 20 abr. 2017.

PORTAL DA RADIO ANTIGUIDADE. A verdadeira história do transistor. 2014. Disponível em: <<http://www.bn.com.br/radiosantigos/semicond.htm>>. Acesso em: 2 jul. 2014.

TEODORESCU, R.; LISERRE, M.; Rodríguez, P. *Grid converters for photovoltaic and wind power systems*. John Wiley & Sons: 2011. 398 p. ISBN: 978-0-470-05751-3. <https://doi.org/10.1002/9780470667057>

APÊNDICE

APÊNDICE A - Exemplo de funções primitivas do System Generator



APÊNDICE B - Código implementado no bloco Black Box

-- Engineer: Nei Souza
-- Create Date: 17:32:05 08/12/2016
-- Module Name: "Sample-and_hold"
-- Project Name: Controle Repetitivo
-- Target Devices: Sapartan 3AN700AN
Comments:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use ieee.std_logic_unsigned.all;
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
-- Uncomment the following library declaration if instantiating
-- any Xilinx primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;
-- Autor Eng. Nei Souza
-- Modulo com o objetivo de trabalhar como memoria temporaria
```

```
entity velocidade is
  Port (
    Disparo_in:      in std_logic;
    Referencia_in:   in std_logic_vector(63 downto 0);
    Referencia_out:  out std_logic_vector(63 downto 0);
    Realimentacao_in: in std_logic_vector(63 downto 0);
    Realimentacao_out: out std_logic_vector(63 downto 0));
end velocidade;
```

architecture Behavioral of velocidade is

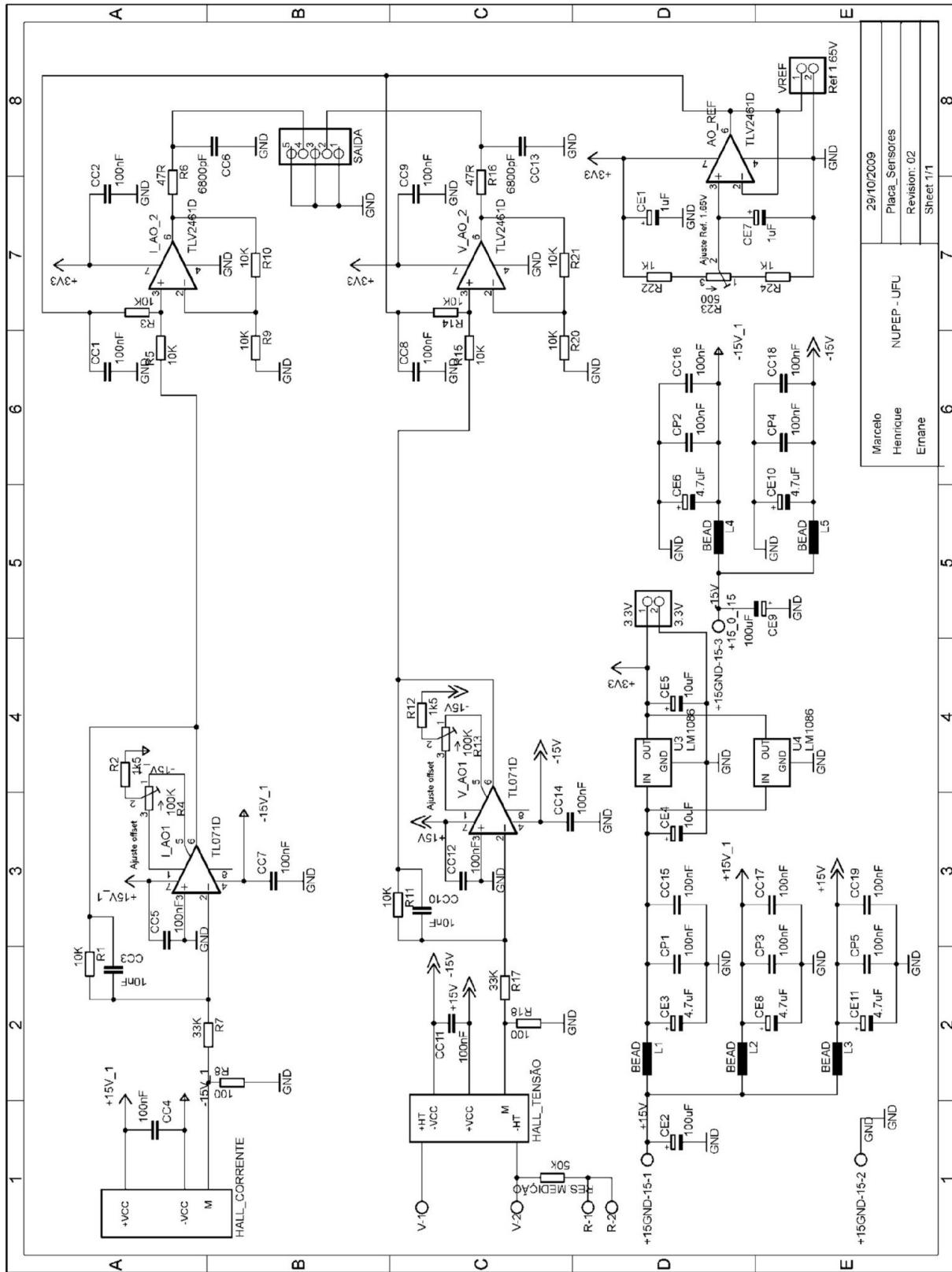
```
end if;

if rising_edge(Disparo_in) and (conta>0)
then
    Reali_in:=Realimentacao_in;
    Ref_in:=Referencia_in;
    Referencia_out  <= Ref_in;
    Realimentacao_out <=  Reali_in;
end if;

conta:=2;
end process;
end Behavioral;
```

ANEXOS

ANEXO A - Circuito da placa de aquisição e condicionamento de sinais



ANEXO B - Código implementado no bloco Mcode

```
function matched = state_machine(din)

persistent state, state = xl_state(0,{xlUnsigned, 3, 0});

switch state
    case 0
        if din == 1
            state = 1;
        else
            state = 0;
        end
        matched = 0;
    case 1
        if din == 0
            state = 2;
        else
            state = 0;
        end
        matched = 0;
    case 2
        if din == 1
            state = 3;
        else
            state = 0;
        end
        matched = 0;
    case 3
        if din == 1
            state = 4;
        else
            state = 2;
        end
        matched = 0;
    case 4
        if din == 0;
            state = 0;
        else
            state = 1;
        end
        matched = 1;
    otherwise
        state = 0;
        matched = 0;
end
```