

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E
ELETRÔNICA**

Márcio Moura Bridon Júnior

**ANALISADOR DE POTÊNCIA PARA APLICAÇÕES EM
ELETRÔNICA DE POTÊNCIA**

Florianópolis

2018

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

Júnior, Márcio Moura Bridon
Analisador de Potência para Aplicações em
Eletrônica de Potência / Márcio Moura Bridon Júnior
; orientador, Gean Jacques Maia de Sousa,
coorientador, Marcelo Lobo Heldwein, 2018.
191 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro
Tecnológico, Graduação em Engenharia Elétrica,
Florianópolis, 2018.

Inclui referências.

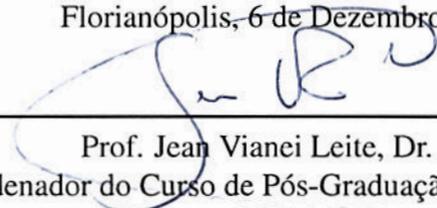
1. Engenharia Elétrica. 2. Analisador de
Potência. 3. Instrumentação Eletrônica. 4. Sistemas
Embarcados. 5. Aplicações em Eletrônica de Potência.
I. Jacques Maia de Sousa, Gean. II. Lobo Heldwein,
Marcelo. III. Universidade Federal de Santa
Catarina. Graduação em Engenharia Elétrica. IV.
Título.

Márcio Moura Bridon Júnior

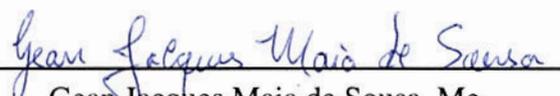
ANALISADOR DE POTÊNCIA PARA APLICAÇÕES EM ELETRÔNICA DE POTÊNCIA

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

Florianópolis, 6 de Dezembro 2018.



Prof. Jean Vianei Leite, Dr. Eng.
Coordenador do Curso de Pós-Graduação em Engenharia
Elétrica

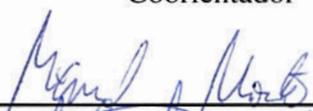


Gean Jacques Maia de Sousa, Me.
Orientador

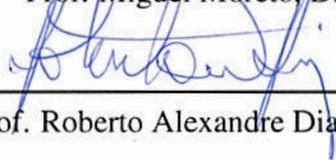
Banca Examinadora:



Prof. Marcelo Lobo Heldwein, Dr. sc. ETH
Coorientador



Prof. Miguel Moreto, Dr. Eng.



Prof. Roberto Alexandre Dias, Dr. Eng.

Aos meus avós Helena Demonti e Ovídio Sartori, minha mãe Anita e meu irmão Felipe.

AGRADECIMENTOS

Agradeço a Universidade Federal de Santa Catarina e a Universidade de Toronto, além da sociedade brasileira, que financiou diretamente os meus estudos.

Aos professores que me influenciaram positivamente ao longo do curso: Rubens Stark, Márcio Costa, Marcelo Lobo Heldwein, Telles Lazzarin, Roberto Coelho, Roberto Salgado, Antônio Simões Costa e Denizar C. Martins, da Universidade Federal de Santa Catarina, além dos professores David A. Johns, Zeb Tate, Tomasz Czajkowski, Michael Stumm e Paul Chow, da Universidade de Toronto. Um muito obrigado a cada um desses por todo o conhecimento transmitido e por serem profissionais de excelência.

Aos meus primeiros chefes, Natasha Baker e Garret Fick, da start-up SnapEDA. Ter partilhado da experiência de viver no coração do Vale do Silício foi, sem dúvidas, algo extraordinário.

Ao grande amigo Hugo Guilherme Maestri, que foi a minha maior influência para escolher o curso de Engenharia Elétrica, além de ter sido um grande irmão durante o tempo em que moramos juntos em Florianópolis.

Agradeço a minha família, sobretudo minha mãe e meu irmão, que sempre me apoiaram nas decisões que tomei durante a graduação. Também aos meus avós Helena e Ovídio (*in memoriam*), que para mim são exemplos de vida.

Agradecimentos aos grandes amigos do Ramal 7465, Lucas Munaretto, Victor "Vô" Gruner, Vinicius Fiamoncini, Victor "PT" Pirão, Murilo Koerich Lohn e Vinicius "Moco" Kremer. Estendo os agradecimentos aos agregados Luiz Gili e Roberto Buerger, vulgo Alemão, e também aos colegas e amigos do la-

boratório INEP.

Aos amigos de curso e da UFSC, Rafael Medeiros e Denise Silvestrin.

Agradecimentos ao corpo administrativo e técnico do INEP, compostos pelo Diogo Duarte Luiz, Antonio Luiz Schalata Pacheco, Luiz Marcellus Coelho e Angelo Manoel de Matos Leal.

Agradecimentos mais do que especiais para a equipe que integrei por mais de dois anos, composta pelos mestres (quase doutores) Gean Jacques Maia de Sousa e Jacson de Oliveira, além do novo integrante Bruno Maciel. Foi através de vocês que aprendi muito sobre comprometimento, excelência e resiliência. Além de várias linguagens de programação...

Ao orientador desse trabalho. Gean, é uma honra ter sido o teu primeiro orientado de TCC, e espero que tenha atendido as tuas expectativas.

E o agradecimento final fica para o professor Marcelo Lobo Heldwein. Muito obrigado pela ajuda na conquista do estágio na ABB. Creio que essa experiência será única e muito importante para o meu início de carreira.

Muito obrigado a todos vocês!

"All rational action is in the first place individual action. Only the individual thinks. Only the individual reasons. Only the individual acts."

Ludwig von Mises

Resumo

O presente trabalho visa o desenvolvimento de analisador de potência para aplicações em eletrônica de potência. O protótipo possibilita monitorar escalas de tensões de 1000 V, 400 V e 100 V, assim como escalas de correntes de 35 A, 15 A e 5 A. A partir de transdutores de corrente e tensão, realiza-se o condicionamento dos sinais de medição para conversores analógico-digitais, amostrando ambas medições de tensão e corrente a uma taxa fixa de 750 kHz. O sistema proposto é composto de uma placa de aquisição, formada pelo circuito analógico de medição e conversão analógico-digital, uma placa de desenvolvimento SoC (System-on-Chip), empregando um FPGA e um processador ARM Dual-Core, além de um computador que executa uma interface gráfica, correspondendo à interface do usuário. Dessa forma, o sistema embarcado, composto pelo SoC, processa os pacotes de dados recebidos da placa de aquisição e transfere estes para a interface do usuário, que permite a visualização das formas de onda dos sinais de corrente e tensão, além de disponibilizar os valores médios e eficazes das componentes de tensão e corrente, além dos valores de potência ativa, reativa e da frequência estimada. São apresentados a análise teórica dos circuitos analógicos, os circuitos digitais desenvolvidos para os dispositivos de descrição em hardware assim como a estrutura dos algoritmos implementados no sistema embarcado e na interface do usuário. Além disso, foi realizada uma análise comparativa de precisão de medição entre o protótipo e o analisador de potência WT210, da empresa Yokogawa.

Palavras-chave: Analisador de Potência. Sistemas de Monitoramento. Eletrônica de Potência.

Abstract

This work discusses the development of a power analyzer for power electronics applications. The prototype has capabilities of monitoring voltage scales of 1000 V, 400 V and 100 V, and current scales of 35 A, 15 A and 5 A. Through current and voltage transducers, the measured signals drive analog-to-digital converters in a fixed sampling rate of 750 kHz. The system is comprised of an acquisition board that has an analog circuitry to perform measurements and to perform the analog-to-digital conversion, a SoC (System-on-Chip) development board with an FPGA and a Dual-Core ARM processor and also a computer that executes an user interface. Hence, the SoC development board processes the data packets from the acquisition board and transfers these to the user interface, which plots the voltage and current waveforms, and also shows the user the mean and RMS values of voltage and current, as well as active power, reactive power and the estimated frequency. Theoretical analysis for the analog section is presented, as well as the digital circuits developed on the descriptive hardware devices, the algorithm structures for the SoC development board and the user interface. Also, a comparative analysis has been done between the prototype and the WT210 power analyzer from Yokogawa.

Keywords: Power Analyzer. Monitoring Systems. Power Electronics.

Lista de Figuras

Figura 2.1	Diagrama de blocos do analisador de potência WT210. Figura retirada de [15].	42
Figura 2.2	Filtro banda passante de banda estreita sintonizado em torno da frequência fundamental do sinal estimado	45
Figura 2.3	Diagrama de blocos do filtro digital	46
Figura 2.4	Diagrama de bode dos filtros banda passante projetados	47
Figura 2.5	Resposta em regime permanente do filtro projetado	48
Figura 2.6	Detecção de passagem por zero	48
Figura 3.7	Componentes do sistema de aquisição de tensão e corrente.	51
Figura 3.8	Circuito de aquisição de tensão.	53
Figura 3.9	Parcela do circuito que emprega o divisor resistivo.	54
Figura 3.10	Formas de onda do circuito de aquisição de tensão.	58
Figura 3.11	Tensão imposta nas chaves seletoras de ganho.	59
Figura 3.12	Circuito de aquisição de corrente.	61
Figura 3.13	Formas de onda do circuito de aquisição de corrente.	65
Figura 3.14	(a) Amostragem de um sinal que respeita a Taxa de Nyquist. (b) Amostragem de um sinal que não respeita a Taxa de Nyquist causando recobrimento de espectros (c) Filtro passa-baixa (FPB) atuando como filtro anti-recobrimento.	67

Figura 3.15	Circuito de aplicação do filtro passa-baixas e driver para o ADC	67
Figura 3.16	Análise do filtro passa-baixas.....	68
Figura 3.17	Diagrama da estrutura de alimentação do protótipo.....	70
Figura 3.18	Conversor ressonante CC-CC isolado para alimentação do protótipo.....	71
Figura 3.19	Tanque ressonante do conversor CC-CC isolado para alimentação do protótipo.....	72
Figura 3.20	Resposta em frequência do tanque ressonante para diferentes fatores de qualidade. A legenda representa o fator de qualidade Q	74
Figura 3.21	Formas de onda do conversor operando com carga de 4,125 W.....	75
Figura 3.22	Formas de onda do conversor operando com carga de 2,25 W.....	76
Figura 3.23	Formas de onda do conversor operando com carga de 0,75 W.....	77
Figura 4.24	Diagrama de blocos do sistema proposto, explicitando os protocolos de comunicação entre dispositivos.....	79
Figura 4.25	Hardware descritivo do sistema desenvolvido para o CPLD da placa de aquisição.....	82
Figura 4.26	Diagrama de blocos simplificado da estrutura implementada no FPGA.....	83
Figura 4.27	Diagrama de blocos do sistema de recepção e transmissão de dados.....	85
Figura 4.28	Diagrama de blocos do hardware em FPGA que realiza a escrita dos pacotes na memória SDRAM externa.....	86
Figura 4.29	Diagrama de blocos dos integradores e do sis-	

tema de estimação de frequência.....	87
Figura 4.30 Aquisição do Trigger de Nível.....	88
Figura 4.31 Aquisição do Trigger de Nível.....	89
Figura 4.32 Barramentos de interconexão entre o FPGA e o HPS. Retirado de [33].....	90
Figura 5.33 Diagrama de blocos simplificado do software desenvolvido para a placa DE-10 Nano SoC.....	96
Figura 5.34 Diagrama de blocos simplificado do software desenvolvido para o ambiente da interface de usuário.....	96
Figura 5.35 Diagrama de classes para o padrão de projeto para máquinas de estado.....	97
Figura 5.36 Diagrama da máquina de estados implementada.....	99
Figura 5.37 Diagrama de classes para a comunicação UDP.....	101
Figura 5.38 Diagrama de classes para o cálculo e transferência da DFT.....	103
Figura 5.39 Diagrama de classes de interface dos periféricos do FPGA.....	104
Figura 5.40 Diagrama de classes do cálculo de potência e do ajustador do filtro banda passante.....	105
Figura 5.41 Utilização das classes nas diferentes threads do programa.....	106
Figura 5.42 Diagrama de classes da interface de usuário.....	109
Figura 5.43 Janela de seleção dos ganhos de sensor do circuito de aquisição e a seleção dos filtros passa-baixas.....	112
Figura 5.44 Janela de configuração do trigger, de seleção do tipo de visualização, modificação do nível do trigger além do controle sobre os eixos horizontais e verticais do gráfico.....	113

Figura 5.45	Janela de visualização das medições das componentes do analisador de potência.....	113
Figura 6.46	Vista do protótipo da placa de aquisição. ...	116
Figura 6.47	Conexão do protótipo com a placa DE-10 Nano SoC desenvolvido pela empresa Terasic.....	117
Figura 6.48	Atuação do soft-start durante a inicialização da fonte. Sinal verde: tensão de saída positiva. Sinal amarelo: tensão de saída negativa.....	118
Figura 6.49	Entrada em condução da chave S_2 e a tensão V_{hb} de entrada do tanque ressonante. Sinal amarelo: tensão de entrada do tanque ressonante V_{hb} . Sinal verde: pulso de comando da chave inferior S_2	118
Figura 6.50	Entrada em condução da chave S_2 com tensão nula sobre os terminais. Sinal amarelo: tensão de entrada do tanque ressonante V_{hb} . Sinal verde: pulso de comando da chave inferior S_2	119
Figura 6.51	Diagrama de bode para diferentes escalas de tensão sem filtro passa-baixas.	120
Figura 6.52	Diagrama de bode para escala de tensão de 100 V com diferentes frequências de corte do fitro passa-baixas.	120
Figura 6.53	Diagrama de bode para diferentes escalas de corrente sem filtro passa-baixas.....	121
Figura 6.54	Diagrama de bode para escala de corrente de 5 A com diferentes frequências de corte do fitro passa-baixas.	121
Figura 6.55	Diagrama de bode para escala de corrente de 35 A com diferentes frequências de corte do fitro passa-baixas.	122
Figura 6.56	Interface gráfica usada para validação da estimação de frequência de uma forma de onda senoi-	

	dal. Sinal superior: forma de onda de entrada. Sinal inferior: forma de onda da saída do filtro passa-banda.	124
Figura 6.57	Interface gráfica usada para validação da estimação de frequência de uma forma de onda quadrada. Sinal superior: forma de onda de entrada. Sinal inferior: forma de onda da saída do filtro banda passante.	124
Figura 6.58	Formas de onda do osciloscópio Tektronix em modo trigger de nível. Sinal verde: tensão de linha. Sinal amarelo: corrente de linha.	135
Figura 6.59	Formas de onda da interface gráfica em modo trigger de nível. Sinal superior: tensão de linha. Sinal inferior: corrente de linha.	135
Figura 6.60	Formas de onda do osciloscópio Tektronix em modo trigger de nível. Sinal verde: tensão de linha. Sinal amarelo: corrente de linha.	136
Figura 6.61	Formas de onda da interface gráfica em visualização em modo trigger de nível. Sinal superior: tensão de linha. Sinal inferior: corrente de linha. ...	136
Figura 6.62	Formas de onda do osciloscópio Tektronix em visualização no modo contínuo. Sinal verde: tensão de linha. Sinal amarelo: corrente de linha. ..	137
Figura 6.63	Formas de onda da interface gráfica em visualização no modo contínuo. Sinal superior: tensão de linha. Sinal inferior: corrente de linha.	137
Figura 6.64	Formas de onda da interface gráfica em visualização da magnitude do espectro. Espectro superior: tensão de linha. Espectro inferior: corrente de linha.	138
Figura 6.65	Visualizador do analisador de potência para a análise CA.	139

Figura B.1	Esquemático de simulação do software Tina-TI para o circuito de aquisição de tensão.	164
Figura B.2	Esquemático de simulação do software Tina-TI para o circuito de aquisição de corrente.	165
Figura B.3	Esquemático de simulação do software PSIM para o conversor ressonante.	166
Figura B.4	Esquemático de simulação do software PSIM para o circuito integrado NCP1397, compreendendo as funcionalidades do oscilador controlado por tensão, soft-start e aplicação de tempo morto.	167
Figura C.1	Esquemático de ligação dos blocos que concentram os circuitos desenvolvidos para a placa de aquisição.	172
Figura C.2	Esquemático do divisor resistivo para medição de tensão.	173
Figura C.3	Esquemático do circuito de aquisição de tensão.	174
Figura C.4	Esquemático do circuito de aquisição de corrente.	175
Figura C.5	Esquemático do conversor analógico-digital.	176
Figura C.6	Esquemático dos reguladores de alimentação dos circuitos integrados da placa.	177
Figura C.7	Esquemático da alimentação do CPLD.	178
Figura C.8	Esquemático do banco de pinos de entrada e saída do CPLD.	179
Figura C.9	Esquemático dos transmissores e receptores da placa.	180
Figura C.10	Esquemático do receptor ótico.	181
Figura C.11	Esquemático do transmissor ótico.	182
Figura C.12	Layout da placa de circuito impresso.	183

Figura C.13	Esquemático do conversor série-ressonante para alimentação da placa.	184
Figura D.1	Layout da placa de circuito impresso.	187
Figura D.2	Layer L1 (top) da placa de circuito impresso.	188
Figura D.3	Layer L2 da placa de circuito impresso.	189
Figura D.4	Layer L3 da placa de circuito impresso.	190
Figura D.5	Layer L4 (bottom) da placa de circuito impresso.....	191

Lista de Tabelas

Tabela 3.1	Relação de ganho do circuito de tensão para as escalas definidas.	55
Tabela 3.2	Valores corrigidos de tensão de saída e ganhos do divisor resistivo para as diferentes escalas de tensão.	57
Tabela 3.3	Tensão de saída do estágio diferencial e ganhos do amplificador de ganho variável.	62
Tabela 3.4	Valores dos capacitores para as frequências de corte escolhidas.	69
Tabela 3.5	Valor dos componentes do tanque ressonante.	71
Tabela 4.6	Lista de registradores do <i>trigger</i> , decimação e configuração de ganhos, além do filtro de estimação de frequência acessados via barramento <i>Lightweight HPS2FPGA AXI Bridge</i>	92
Tabela 4.7	Lista de registradores das medições e calibrações dos conversores analógico-digital acessados via barramento <i>Lightweight HPS2FPGA AXI Bridge</i> .	93
Tabela 6.8	Validação dos ganhos dos circuitos analógicos.	122
Tabela 6.9	Teste do algoritmo de estimação de frequência em comparação com o multímetro.	125
Tabela 6.10	Coeficientes resultantes da regressão para um polinômio de segunda ordem.	126
Tabela 6.11	Medições utilizadas na calibração do canal de tensão.	127
Tabela 6.12	Medições utilizadas na calibração do canal de corrente.	128
Tabela 6.13	Comparativo entre o analisador de potência	

	Yokogawa WT210 e o protótipo para o canal de tensão no teste em corrente contínua.....	130
Tabela 6.14	Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para o canal de corrente no teste em corrente contínua.	131
Tabela 6.15	Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para a potência estimada no teste em corrente contínua.	132
Tabela 6.16	Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para o canal de tensão no teste em corrente alternada.	133
Tabela 6.17	Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para o canal de corrente no teste em corrente alternada.....	133
Tabela 6.18	Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para a potência estimada no teste em corrente alternada.	134

Siglas e Acrônimos

ADC	<i>analog-to-digital converter</i>
ARM	<i>advanced risc machine</i>
AXI	<i>advanced extensible interface</i>
CPLD	<i>complex programmable logic device</i>
DFT	<i>discrete fourier transform</i>
DMA	<i>direct memory access</i>
FPGA	<i>field programmable gate array</i>
FFTW	<i>fastes fourier transform in the west</i>
FFT	<i>fast fourier transform</i>
F2H	<i>fpga to hps</i>
HPS	<i>hard processor system</i>
IP	<i>intelectual property</i>
MSPS	<i>million samples per second</i>
PLL	<i>phase-locked loop</i>
SPI	<i>serial peripheral interface</i>
SDRAM	<i>synchronous dynamic random-access memory</i>
SoC	<i>system-on-chip</i>
UART	<i>universal asynchronous receiver/transmitter</i>
VI	<i>virtual instrumentation</i>

Unidades

A	<i>Ampère</i>
dB	<i>Decibel</i>
Ω	<i>Ohm</i>
F	<i>Farad</i>
H	<i>Henry</i>
Q	<i>Coulomb</i>
E	<i>Joule</i>
Hz	<i>Hertz</i>
J	<i>Joule</i>
K	<i>Kelvin</i>
rad	radianos
s	segundo
m	metro
V	<i>Volt</i>
VA	<i>Volt-ampère</i>
VAr	<i>Volt-ampère reativo</i>
W	<i>Watt</i>
°	graus

Sumário

1	INTRODUÇÃO	29
1.1	Motivação	29
1.2	Revisão Bibliográfica	30
1.3	Objetivos Gerais	31
1.4	Organização do Documento	32
2	ANALISADOR DE POTÊNCIA	35
2.1	Potência Instantânea	35
2.2	Potência em Circuitos de Corrente Contínua	36
2.3	Potência em Circuitos de Corrente Alternada com Componentes Puramente Senoidais	37
2.4	Fator de Potência	39
2.5	Potência no Domínio Discreto	39
2.6	Estimação de Frequência	41
2.6.1	Transformada Discreta de Fourier	42
2.6.1.1	Aplicação da FFTW	44
2.6.2	Filtro de Banda Definida	44
2.6.2.1	Exemplo de Projeto	47
2.6.3	Detector de Passagem por Zero	48
3	SISTEMA DE AQUISIÇÃO	51
3.1	Aquisição de Tensão	52
3.1.1	Resultados de Simulação	57
3.2	Aquisição de Corrente	60
3.2.1	Resultados de Simulação	64
3.3	Filtro de Limitação de Banda	66
3.4	Circuitos de Alimentação	69
3.4.1	Conversor CC/CC Ressonante	70
3.4.2	Formas de Onda	73
4	PLATAFORMA DE HARDWARE DIGITAL	79
4.1	Hardware da Placa de Aquisição	80
4.2	Hardware da Plataforma de Desenvolvimento DE- 10 Nano SoC	82

4.2.1	Transmissão de dados para a SDRAM	84
4.2.2	Estimação de Frequência e Integradores	87
4.2.3	Circuito de <i>Trigger</i> por Nível	88
4.2.4	Lista de Registradores	89
5	SOFTWARE	95
5.0.1	Sistema Embarcado	95
5.0.1.1	Padrão de Projeto para Máquina de Estados	97
5.0.1.2	Diagrama de UML de Classes	98
5.1	Interface de Usuário	107
5.1.1	Diagrama de Classes	108
5.1.2	Componentes da Interface	112
6	RESULTADOS	115
6.1	Circuitos Integrados da Placa de Aquisição	115
6.2	Protótipo	116
6.3	Fonte de Alimentação	117
6.4	Circuito de Aquisição	119
6.5	Estimação de Frequência	123
6.6	Calibração do Circuito de Aquisição	126
6.7	Exatidão do Protótipo	129
6.7.1	Análise com Sinais CC	129
6.7.2	Análise com Sinais CA	132
6.8	Interface Gráfica	134
7	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	141
	Referências	143
	APÊNDICE A – Parte do Software do Processador ARM da Placa de Desenvolvimento	151
	APÊNDICE B – Esquemáticos de Simulação	163
	APÊNDICE C – Esquemáticos da Placa de Aquisição	171
	APÊNDICE D – Placa de Circuito Impresso	187

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

O campo da Eletrônica de Potência engloba o desenvolvimento de sistemas eletrônicos de conversão de energia. Essa conversão se dá em sistemas CC/CA, CA/CC, além dos CC/CC e CA/CA [1]. Algumas das aplicações compreendidas nesse campo são a conversão de energia para a sua devida transmissão, tanto em corrente contínua (CC) como em corrente alternada (CA), sistemas de geração solar e eólica, fontes de alimentação, carregadores de bateria, acionadores de motores, dentre outras.

Uma característica em comum dessas aplicações é a necessidade da validação de protótipos e produtos, tanto pela funcionalidade dos circuitos elétricos que compõem o sistema sob testes, como circuitos auxiliares, por exemplo, quanto para a validação da topologia que realiza o processamento de energia. Com isso, entram os sistemas de medição de alta precisão, capazes de atender a necessidade e caracterizar os sistemas sob testes em termos de suas componentes elétricas, como a tensão, corrente e potência, tanto pela visualização gráfica quanto por meio do cálculo de valores médios e eficazes.

A partir dessa necessidade, torna-se imprescindível a utilização de equipamentos como osciloscópios e analisadores de potência durante a etapa de testes e validação de protótipos e produtos. Portanto, esse trabalho visa desenvolver um protótipo de um Analisador de Potência para Aplicações em Eletrônica de Potência, englobando qualidades de osciloscópios e analisadores de potência em um único dispositivo. Ademais, busca-se através desse projeto desenvolver um protótipo modular, em que os sistemas de aquisição possam ser integrados aos protótipos

sob testes. Dessa forma, separam-se as partes de medição e de visualização, angariando maior segurança para o usuário, tendo em vista que nos osciloscópios e analisadores de potência tradicionais as ponteiras de medição e a conexão com o protótipo ficam próximas das interfaces de visualização.

1.2 REVISÃO BIBLIOGRÁFICA

A aplicabilidade de sistemas de monitoramento é abundante e abrangente, tendo em vista que esses podem compor a aquisição de variáveis físicas como pressão e temperatura, por exemplo. O presente trabalho se restringirá à aplicação dos sistemas apenas para variáveis elétricas, tendo como objetivo a realização de aquisição de sinais elétricos de tensão e corrente.

Trabalhos como em [2] e [3] apresentam a aplicação da instrumentação virtual (VI), que se utilizam de hardwares de aquisição além de softwares que implementam interfaces do usuário e realizam o processamento de dados. Esses trabalhos mostram a capacidade de modularidade de sistemas baseados em software, além da capacidade de se utilizar múltiplos módulos de aquisição e em diferentes pontos de monitoramento. Em [4] tem-se o desenvolvimento de um analisador de potência para aplicação em sistemas de potência utilizando do conceito de instrumentação virtual. O trabalho é desenvolvido em LabVIEW, uma linguagem de programação proprietária da empresa National Instruments, possui um sistema analógico de medição e um hardware digital para interfacear o sistema com um computador de uso geral.

Acerca de osciloscópios virtuais, [5] implementa um sistema de monitoramento que realiza a transferência de dados através de um transmissor wireless. A parcela referente ao processamento e de interface com o usuário é feita em um celular com sistema operacional Android. Esse trabalho apresentou a capaci-

dade de monitorar dois canais a 200 MSPS (milhões de amostras por segundo).

Dessa forma, vê-se o desenvolvimento de sistemas de monitoramento descentralizados e com capacidades que visam diferentes nichos de aplicações. Citando alguns, tem-se sistemas de potência, microrredes e medição de sinais de eletrônica de sinal, por exemplo. Essas aplicações possuem uma série de requerimentos que permitem diferenciar umas das outras, como em requerimentos de banda, taxas de amostragem, resolução e a capacidade de realizar operações matemáticas, como para o cálculo de potência e a magnitude do espectro do sinal.

1.3 OBJETIVOS GERAIS

O presente trabalho traz como objetivos o desenvolvimento de um protótipo de analisador de potência, composto por uma placa de aquisição de tensão e corrente, uma placa de desenvolvimento contendo um FPGA e um ARM Dual-Core para realizar o processamento de dados, além de uma interface gráfica que tem por finalidade a análise das componentes elétricas, dadas pelos valores eficazes e médios de tensão e corrente, assim como a potência ativa, potência aparente e o fator de potência. Além disso, a interface possibilitará a visualização gráfica dos sinais de medição em três modos: "*Trigger* de Nível", "Contínuo" e "Espectro".

A placa de aquisição conterá circuitos de medição de tensão e corrente com escalas e filtros passa-baixas selecionáveis através de chaves analógicas, além de um conversor analógico-digital que terá por objetivo amostrar os sinais de medição. Esses serão conduzidos para um CPLD, que se comunicará com transmissores digitais para enviar os sinais amostrados para a placa de desenvolvimento.

A placa de desenvolvimento receberá os dados trans-

mitidos pela placa de aquisição, tendo por finalidade o cálculo dos valores de corrente, tensão e potência, assim como a estimação de frequência fundamental dos sinais de medição. Adicionalmente, essa placa transmitirá pacotes de informação para a interface do usuário, a fim de disponibilizar as formas de onda monitoradas e os valores de tensão, corrente, potência e frequência para o usuário. Pela interface do usuário será possível a modificação de parâmetros como o nível do *trigger*, as escalas de tempo e amplitude, assim como configurar as escalas de medição e os filtros passa-baixas dos circuitos de tensão e corrente.

1.4 ORGANIZAÇÃO DO DOCUMENTO

O Capítulo 2 trata das potências em análise de circuitos CC e CA, além das equações na forma discreta. Também, introduz a estrutura de estimação de frequência, compreendendo a Transformada Discreta de Fourier (DFT), filtro banda passante e um detector de passagem por zero.

O Capítulo 3 é referente a parcela analógica do circuito de aquisição de tensão e corrente, explicitando o cálculo e dimensionamento dos componentes, além dos resultados de simulação dos circuitos, assim como o conversor CC/CC isolado utilizado para a alimentação da placa de aquisição.

O Capítulo 4 discorre sobre os hardwares digitais implementados no CPLD, embarcado na placa de aquisição, e no FPGA, presente na placa DE-10 Nano SoC, desenvolvido pela empresa Terasic e que é utilizada como a placa de desenvolvimento.

O Capítulo 5 cobre o desenvolvimento de software para ambos processador ARM, presente na placa de desenvolvimento DE-10 Nano SoC, como o software desenvolvido para o ambiente desktop, que implementa a interface do usuário.

O Capítulo 6 discorre sobre os resultados obtidos com o sistema desenvolvido, além das características dos componentes

que compõem o protótipo da placa de aquisição, e o Capítulo 7 trata das considerações finais e traz sugestões para trabalhos futuros.

2 ANALISADOR DE POTÊNCIA

Potência, em termos físicos, é a taxa de variação de energia, sendo o watt unidade adotada pelo Sistema Internacional de Unidades (SI), equivalente a joules por segundo. Acerca do cálculo de potência, permite-se formular a potência média transferida, por exemplo, em um circuito elétrico, como a variação de energia ΔE ao longo de um intervalo de tempo ΔT , resultando na relação (2.1). Os equacionamentos presentes para os cálculos das componentes de potência para circuitos CC e CA seguem como em [6].

$$P_{med} = \frac{\Delta E}{\Delta T} \quad (2.1)$$

2.1 POTÊNCIA INSTANTÂNEA

A fim de descrever a relação de potência dada por (2.1) em termos de potência instantânea, toma-se o limite da relação (2.1) para uma variação de tempo infinitesimal, como descrito por (2.2).

$$p(t) = \lim_{\Delta t \rightarrow 0} \frac{\Delta E}{\Delta t} = \frac{dE}{dt} \quad (2.2)$$

Pode-se escrever a variação de energia em função do potencial elétrico e da variação de carga elétrica. Por análise dimensional permite-se validar as igualdades, visto que energia é dado em joule, potencial elétrico em joule por coulomb e carga elétrica é dado em coulomb. Essa relação é descrita em (2.3), que também relaciona a taxa de variação de carga, resultando na componente conhecida como corrente elétrica, dada em Ampère [7].

$$dE = VdQ = V \cdot Idt \quad (2.3)$$

Combinando as relações (2.2) e (2.3) formula-se o conceito de potência instantânea em função dos termos instantâneos de corrente e tensão, como dado pela igualdade (2.4).

$$p(t) = \frac{dE}{dt} = v(t) \cdot i(t) \quad (2.4)$$

2.2 POTÊNCIA EM CIRCUITOS DE CORRENTE CONTÍNUA

Em circuitos de corrente contínua, em que os valores de tensão e corrente são constantes, utiliza-se o conceito de potência média para mensurar a potência transferida pelo circuito elétrico. Formula-se a potência média a partir dos valores médios de tensão e corrente ao longo de um período de tempo T .

$$P = \frac{1}{T} \int_0^T p(t)dt = \frac{1}{T} \int_0^T v(t) \cdot i(t)dt \quad (2.5)$$

Ao considerar os valores de tensão e corrente como constantes e invariantes no tempo, tem-se a forma (2.6).

$$P = V_{med} \cdot I_{med} \quad (2.6)$$

Também, a potência média pode ser descrita em termos de energia, como apresentado pela relação (2.1). A derivação dessa relação parte do cálculo da energia total transferida ou consumida em um circuito elétrico ao longo de um período de tempo T , como apresentado por (2.7).

$$E = \int_0^T p(t)dt \quad (2.7)$$

Dividindo ambas as partes pela variação temporal T chega-se em (2.8). Dessa forma, torna-se possível calcular a potência

média consumida em um circuito através da energia transferida ao longo de um intervalo de tempo, ou também pelo valor médio da potência instantânea ao longo de um período de tempo.

$$P = \frac{E}{T} = \frac{1}{T} \int_0^T p(t) dt \quad (2.8)$$

2.3 POTÊNCIA EM CIRCUITOS DE CORRENTE ALTERNADA COM COMPONENTES PURAMENTE SENOIDAIS

Em circuitos de corrente alternada, utiliza-se o conceito de valor eficaz para mensurar correntes e tensões. O cálculo do valor eficaz de uma forma de onda periódica com período de T segundos é dado pela relação (2.9).

$$X_{ef} = \sqrt{\frac{1}{T} \int_0^T x^2(t) dt} \quad (2.9)$$

Para elaborar as equações de análise de potência, assume-se os sinais de tensão e corrente como puramente senoidais. Dessa forma, tem-se os sinais de tensão e corrente expressados por (2.10) e (2.11), respectivamente. O fator ϕ representa a defasagem da forma de onda da corrente em relação a da tensão, que é tida como referência para a análise.

$$v(t) = \sqrt{2} V_{ef} \sin(\omega t) \quad (2.10)$$

$$i(t) = \sqrt{2} I_{ef} \sin(\omega t - \phi) \quad (2.11)$$

A relação (2.12) traz informações sobre a potência instantânea, resultante do produto dos valores instantâneos de tensão e corrente.

$$p(t) = v(t) \cdot i(t) = V_{ef} \cdot I_{ef} \cos \phi [1 - \cos(2\omega t)] - V_{ef} \cdot I_{ef} \sin \phi \sin(2\omega t) \quad (2.12)$$

A fim de obter o valor de potência ativa transferida entre uma fonte e uma carga para um circuito monofásico, toma-se o valor médio da potência instantânea explicitada por (2.13). Percebe-se que o resultado é igual ao primeiro termo da relação (2.12) que multiplica $[1 - \cos(2\omega t)]$, e isso acontece devido ao valor médio do termo periódico ser nulo, restando o coeficiente unitário.

$$P = \frac{1}{T} \int_0^T p(t) dt = V_{ef} \cdot I_{ef} \cos \phi \quad (2.13)$$

A potência reativa é definida pelo termo que multiplica $\sin(2\omega t)$ na relação (2.12), resultando em (2.14). Define-se a potência reativa pela parcela de potência que possui valor médio nulo, sendo tratada pela literatura como a parcela de potência que não realiza trabalho [8]. Essa definição é restrita a sistemas que possuem elementos de armazenamento capacitivos e indutivos, excluindo os sistemas de eletrônica de potência, que podem gerar potência reativa e apresentar comportamento não-linear [9].

$$Q = V_{ef} \cdot I_{ef} \sin \phi \quad (2.14)$$

Portanto, a potência instantânea para sistemas monofásicos de corrente alternada com componentes puramente senoidais pode ser escrita como em (2.15). Percebe-se que a potência instantânea possui uma componente com valor constante e igual a P , como em (2.13), e componentes alternadas com frequência angular igual a duas vezes a frequência fundamental das formas de onda de tensão e corrente. Percebe-se então que o valor de potência ativa possui ondulação senoidal que varia entre 0 e $2P$ com frequência angular 2ω , resultando no valor médio P . A po-

tência reativa, no entanto, possui apenas uma componente alternada, correspondendo a troca de energia entre a fonte e a carga, e que não é convertida em trabalho devido ao valor médio ser nulo.

$$\begin{aligned} p(t) &= P[1 - \cos(2\omega t)] - Q \sin(2\omega t) \\ &= P - P \cos(2\omega t) - Q \sin(2\omega t) \end{aligned} \quad (2.15)$$

A composição das potências ativa e reativa traz como resultado a potência aparente, explicitada pela relação (2.16). Essa potência é definida pelo valor resultante da multiplicação dos valores eficazes de tensão e corrente, tendo como resultado o valor máximo de potência ativa que pode ser entregue a uma carga com o fator de potência unitário [10].

$$S = \sqrt{P^2 + Q^2} = V_{ef} \cdot I_{ef} \quad (2.16)$$

2.4 FATOR DE POTÊNCIA

Define-se o fator de potência como a razão entre a potência ativa e a potência aparente, sendo válido para sistemas operando com quaisquer sinal periódico, mesmo que não senoidal. Esse traduz o quanto da potência fornecida por uma fonte é consumida pela carga, mostrando também a defasagem entre tensão e corrente.

$$FP = \cos\theta = \frac{P}{S} \quad (2.17)$$

2.5 POTÊNCIA NO DOMÍNIO DISCRETO

A partir da amostragem dos sinais de tensão e corrente em uma taxa de amostragem f_s , a energia é calculada como em

(2.18). Como o intervalo de tempo é discreto e determinado por um número inteiro de amostras, leva-se em conta um intervalo compreendido entre N_1 e N_2 [11].

$$E = \sum_{n=N_1}^{N_2} v[n] \cdot i[n] \quad (2.18)$$

Como dado em (2.8), a potência ativa é calculada a partir da energia em relação ao intervalo de tempo, sendo essa válida para sinais contínuos e alternados, mesmo que para a análise de potência para sinais que não assumem formas puramente senoidais. No domínio discreto, tem-se que a potência ativa de um sinal discreto e periódico com período de N amostras é definido como em (2.19).

$$P = \frac{1}{N} \sum_{n=0}^{N-1} v[n] \cdot i[n] \quad (2.19)$$

O cálculo das componentes médias e eficazes de tensão e corrente seguem as relações dadas por (2.20) e (2.21).

$$V_{med} = \frac{1}{N} \sum_{n=0}^{N-1} v[n] \quad (2.20)$$

$$I_{med} = \frac{1}{N} \sum_{n=0}^{N-1} i[n]$$

$$V_{ef} = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} v[n] \cdot v[n]} \quad (2.21)$$

$$I_{ef} = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} i[n] \cdot i[n]}$$

Para o cálculo da potência aparente, utiliza-se o produto das componentes eficazes de tensão e corrente. Dessa forma,

permite-se também calcular o fator de potência, como dado em (2.16).

As relações para os cálculos das componentes médias e eficazes de tensão e corrente, além das potências ativa e aparente, são as empregadas no protótipo. Com isso, tem-se que as componentes são dependentes das amostras instantâneas de tensão e corrente, além do intervalo de integração dado por N . Como o valor de N depende da frequência fundamental das formas de onda de tensão e corrente, propõe-se um algoritmo de estimação de frequência.

2.6 ESTIMAÇÃO DE FREQUÊNCIA

Trabalhos realizados acerca da estimação de frequência empregam diferentes métodos para a detecção da frequência fundamental. Alguns desses métodos são a estimação através de filtros digitais adaptativos [12], a detecção de passagem por zero [13], malhas de travamento de fase (*Phase-Locked Loop*) [14], dentre outros. Em relação aos dispositivos analisadores de potência, como o analisador WT210 [15] da empresa Yokogawa, a estimação de frequência é dada por detectores analógicos de cruzamento por zero. O sistema de detecção do WT210 pode ser visto pelo diagrama de blocos da Figura 2.1. Nota-se que o analisador WT210 realiza a detecção de passagem por zero antes do conversor analógico-digital, sendo que o mesmo sistema de medição é empregado nos analisadores WT500 [16] e WT1800 [17], ambos da empresa Yokogawa.

O algoritmo de estimação de frequência implementado no presente trabalho é composto através da Transformada Discreta de Fourier (DFT), de um filtro passa-banda e da detecção de passagem por zero. Sucintamente, esse algoritmo realiza uma estimação inicial através da DFT. Devido ao fato da DFT não possuir resolução suficiente, faz-se o ajuste de um filtro seletivo que tem

capacidade de rejeitar as componentes harmônicas indesejadas, tendo em sua saída apenas a componente de frequência fundamental. Essa componente passa pelo bloco de detecção de passagem por zero, tendo como resultado a estimativa da frequência do sinal.

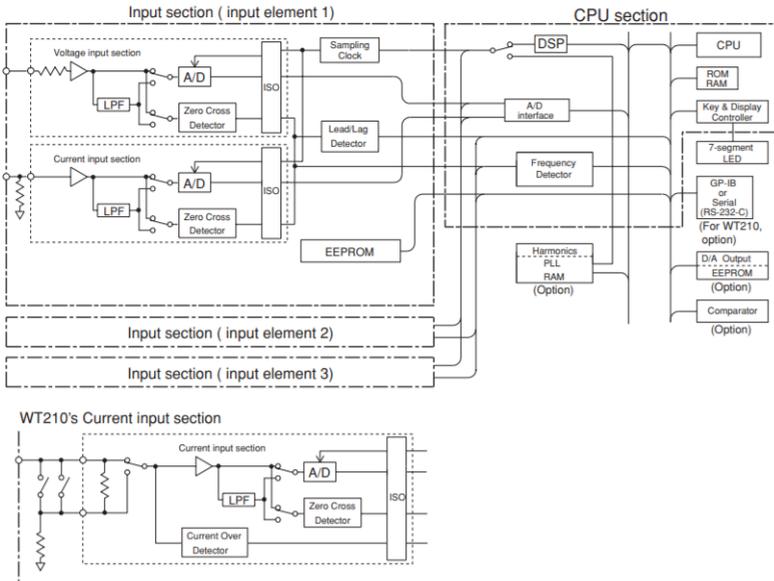


Figura 2.1 – Diagrama de blocos do analisador de potência WT210. Figura retirada de [15].

2.6.1 Transformada Discreta de Fourier

A Transformada Discreta de Fourier, ou simplesmente DFT, é uma transformada discreta que realiza a conversão de amostras igualmente espaçadas de um sinal limitado no tempo para amostras no domínio da frequência com tamanho igual ao número de amostras de entrada da transformada. Essas que resultam da transformada possuem valor complexo, sendo que o sinal de entrada pode assumir valores reais e/ou complexos. Apresenta-se

o cálculo da DFT por (2.22).

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-j\frac{2\pi}{N}kn} \quad (2.22)$$

A principal aplicação da DFT é a análise do conteúdo em frequência de um determinado sinal de interesse [18]. A fim de realizar a estimação inicial da frequência fundamental do sinal através da DFT, busca-se entender a relação entre o número de pontos usados para o cálculo da DFT e a resolução máxima do espectro em frequência. Devido o sinal ser amostrado, tem-se que as amostras estão espaçadas de $T = \frac{1}{f_s}$ segundos. Portanto, tem-se que a relação que mostra o número de pontos N_0 necessários para a computar a DFT para que se tenha a resolução máxima dada por f_0 , em Hz, segue a relação (2.23) [19].

$$N_0 = \frac{f_s}{f_0} \quad (2.23)$$

O número de pontos N_0 para se ter resolução de $f_0 = 1$ Hz, por exemplo, é igual ao valor da frequência de amostragem f_s . No presente projeto, em que a frequência f_s é igual a 750 kHz, o número de pontos necessário para atingir resolução de 1 Hz é $N_0 = 750$ mil pontos. Para uma frequência $f_0 = 5$ Hz, necessita-se de $N_0 = 150$ mil pontos. No projeto utiliza-se uma resolução f_0 de 5 Hz.

Para o caso de x_0, x_1, \dots, x_{N-1} ser um sinal real, a DFT apresenta simetria nas componentes como apresentado em (2.24). Essa simetria é denominada por Hermitiana, em função da matriz ser idêntica à sua transposta conjugada [20]. Essa relação torna-se importante devido ao sinal de entrada proveniente das medições ser puramente real, tornando mais rápida a execução do algoritmo que implementa a DFT.

$$X_{N-n} = X_n^* \quad (2.24)$$

2.6.1.1 Aplicação da FFTW

Com a finalidade de empregar o algoritmo da DFT para o primeiro passo da estimação de frequência, além de disponibilizar na interface de usuário o espectro em frequência dos sinais de entrada do protótipo, fez-se uso da biblioteca FFTW, nomeada em inglês como "*Fastest Fourier Transform in the West*", ou em português como a "Transformada mais Rápida do Oeste".

O algoritmo desenvolvido para a FFTW implementa o algoritmo de Cooley-Tukey para a realização da Transformada Rápida de Fourier (FFT) [21]. De modo a maximizar a performance de execução, a biblioteca se adapta ao hardware em que é executado [20]. Além disso, a implementação da FFTW é de ordem $O(n \log(n))$ para qualquer vetor de entrada de tamanho n , sendo que outros algoritmos são restritos apenas a larguras de dados fixas, caso contrário a ordem do algoritmo é $O(n^2)$. [20].

Como comentado anteriormente, para o caso do cálculo da FFT para vetores reais, sendo a largura do vetor um número primo, tem-se a redução do requerimento de tempo de execução e memória utilizada pela metade [20]. Através de [22] tem-se informações sobre a implementação e a aplicação da biblioteca para a execução de transformadas com uma dimensão ou multi-dimensionais, com conjunto de dados reais ou complexos.

2.6.2 Filtro de Banda Definida

A fim de mensurar a frequência da componente fundamental do sinal de interesse, propõe-se sintonizar um filtro passa-banda em torno da frequência fundamental estimada pela Transformada Discreta de Fourier (DFT). Dessa forma, através de um filtro com banda estreita obtém-se um sinal com formato senoidal e que possui a mesma frequência da componente fundamental do sinal de interesse. Refere-se a Figura 2.2 a fim de ilustrar

o objetivo do filtro de banda passante. A banda do filtro é dada pela diferença entre as frequências f_L e f_H , que são as frequências de decaimento de 3dB, sendo f_r a frequência de ressonância. A análise realizada levará em conta apenas a amplitude do filtro, tendo em vista que a fase não impacta na condição de passagem por zero.

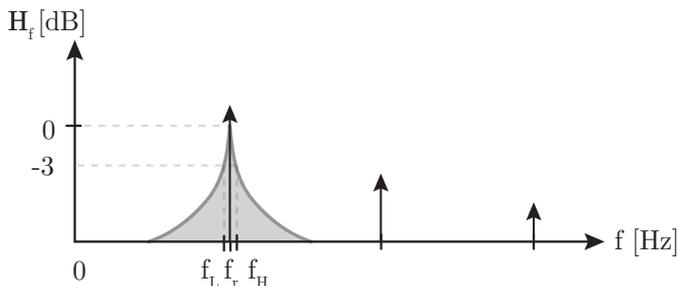


Figura 2.2 – Filtro banda passante de banda estreita sintonizado em torno da frequência fundamental do sinal estimado

A função de transferência de um filtro passa-banda no tempo contínuo é dado por (2.25). Através da aplicação da Transformada Bilinear (2.26) [18], também conhecido como Método de Tustin, na função de transferência (2.25), obteve-se um filtro discreto equivalente.

$$H_f(s) = \frac{s \frac{W_r}{Q}}{s^2 + \frac{W_r}{Q}s + W_r^2} \quad (2.25)$$

$$s \leftarrow \frac{2}{T_s} \frac{z-1}{z+1} \quad (2.26)$$

Dessa forma, o filtro equivalente resultante da aplicação de (2.26) em (2.25) é explicitado por (2.27).

$$H_{fd}(z) = \frac{b_0 z^2 + b_2}{a_0 z^2 + a_1 z + a_2} \quad (2.27)$$

Tem-se que a relação entre o fator de qualidade Q e a largura de banda do filtro B_w é da forma $B_w = \frac{\omega_r}{Q}$ [23]. Com isso, os coeficientes a_0 , a_1 , a_2 , b_0 e b_2 são explicitados em função da largura de banda B_w e da frequência de ressonância do filtro ω_r , além do período de amostragem $T_s = \frac{1}{f_r}$.

$$\begin{aligned} b_0 &= \frac{2B_w T_s}{(T_s \omega_r)^2 + 2B_w T_s + 4} \\ b_2 &= \frac{-2B_w T_s}{(T_s \omega_r)^2 + 2B_w T_s + 4} \\ a_0 &= 1 \\ a_1 &= \frac{2[(T_s \omega_r)^2 - 4]}{(T_s \omega_r)^2 + 2B_w T_s + 4} \end{aligned} \quad (2.28)$$

A partir da função transferência dada por (2.27), tem-se que a relação entre a entrada e a saída do filtro é dada pela equação a diferenças descrita por (2.29), cujos coeficientes são dados em (2.28).

$$\begin{aligned} y[n] &= b_0 \cdot x[n] + b_1 \cdot x[n-1] + b_2 \cdot x[n-2] \\ &\quad - a_1 \cdot y[n-1] - a_2 \cdot y[n-2] \end{aligned} \quad (2.29)$$

Através de (2.29) esquematiza-se o diagrama de blocos do filtro, como da Figura 2.3. Dessa forma, permite-se realizar simulações a fim de validar o conceito do filtro projetado.

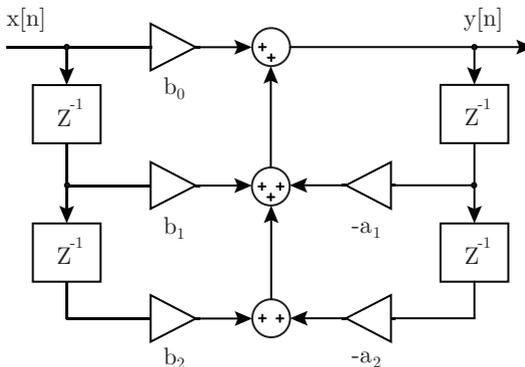


Figura 2.3 – Diagrama de blocos do filtro digital

2.6.2.1 Exemplo de Projeto

A fim de validar a proposta de projeto do filtro de banda estreita, construiu-se as funções transferências como apresentado pela relação (2.27) levando em conta as frequências de ressonância W_r de 60, 120 e 180 Hz. A frequência de amostragem f_s de 750 kHz e largura de banda B_w de 10 Hz.

Os diagramas de Bode dos filtros projetados podem ser vistos na Figura 2.4. Com isso permite-se validar a proposta do filtro, acordando que a frequência de ressonância e a banda do filtro correspondem aos requisitos propostos.

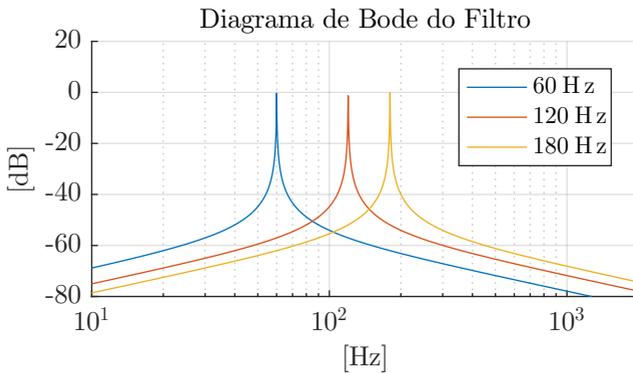


Figura 2.4 – Diagrama de bode dos filtros banda passante projetados

A Figura 2.5 apresenta a resposta em regime permanente do filtro projetado para a frequência de ressonância f_r de 60 Hz a uma entrada dada pela função $x(t) = 0,7 \cdot \text{sen}(2\pi 60t) + 0,3 \cdot \text{sen}(2\pi 180t)$. Percebe-se que o filtro recupera a componente fundamental do sinal de 60 Hz mesmo para uma forte distorção harmônica.

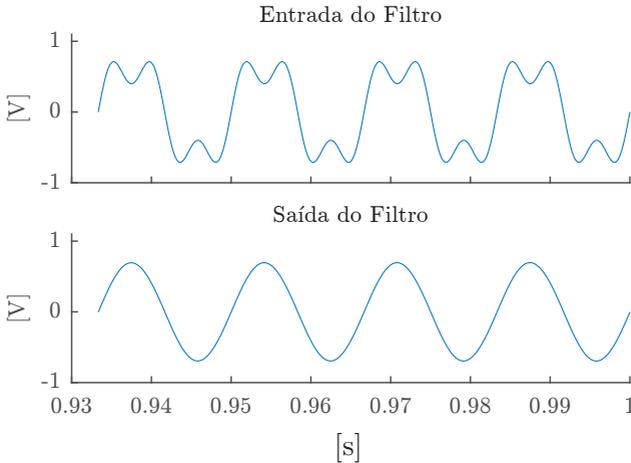


Figura 2.5 – Resposta em regime permanente do filtro projetado

2.6.3 Detector de Passagem por Zero

Devido à componente fundamental do sinal ser preservada após o processamento pelo filtro digital, realiza-se a detecção dos pontos em que o sinal assume valor nulo, cruzando o ponto a partir do semi-ciclo negativo para o positivo, indicando um período completo da forma de onda. Os instantes de interesse estão explicitados pelas setas cinza na Figura 2.6. Dessa forma, em função do filtro ser projetado a fim de recuperar apenas a componente de frequência fundamental, tem-se uma estimativa precisa da frequência do sinal.

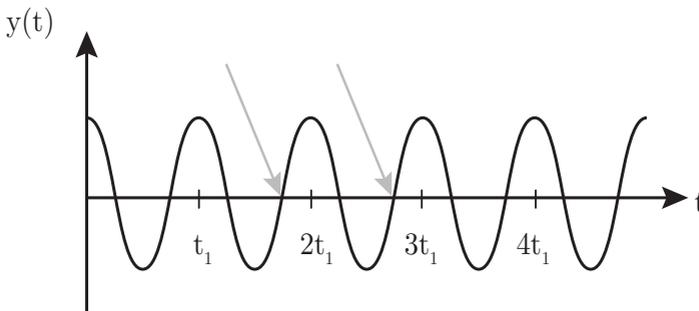


Figura 2.6 – Detecção de passagem por zero

Como o sinal é amostrado, ao cruzar o ponto de detecção, como indicado pela primeira seta cinza, um contador inicializa a fim de realizar o seu incremento para cada amostra recebida. O contador apenas reinicializa quando há a detecção da passagem do sinal do semi-ciclo negativo para o semi-ciclo positivo. Dessa forma, a frequência estimada f_{est} pode ser calculada em função do número de períodos completos de detecção N_p , do valor do contador N_{cont} e da frequência de amostragem f_s . A relação é dada por (2.30).

$$f_{est} = \frac{N_p}{N_{cont}} \cdot f_s \quad (2.30)$$

3 SISTEMA DE AQUISIÇÃO

A fim de realizar a conversão dos sinais elétricos de tensão e corrente, propõe-se a estrutura da Figura 3.7. Realiza-se o condicionamento dos sinais elétricos através dos sensores de tensão e corrente para um conversor analógico-digital, possibilitando a transmissão dos pacotes de dados amostrados para placas externas ao sistema de aquisição.

Através da Figura 3.7 nota-se a utilização de amplificadores e sensores de ganho programável. Esses permitem condicionar diferentes valores de escala dos sinais de entrada para o fundo de escala do conversor analógico-digital. Com isso, permite-se escolher as escalas de tensão e corrente mais adequadas para o tipo de análise que se deseja realizar. Ademais, empregam-se amplificadores diferenciais de modo a rejeitar sinais de modo comum, amplificadores com ganho variável para ajustar o sinal para o fundo de escala do conversor analógico-digital, filtros passa-baixas e por fim a referência de tensão e o driver do conversor analógico-digital.

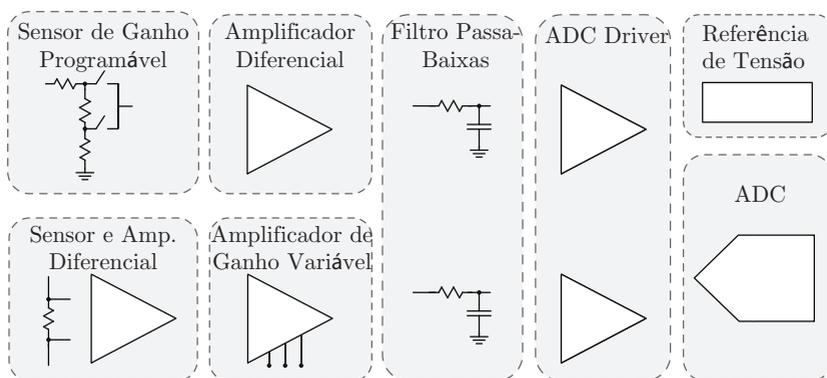


Figura 3.7 – Componentes do sistema de aquisição de tensão e corrente.

3.1 AQUISIÇÃO DE TENSÃO

O sistema de aquisição de tensão é representado na Figura 3.8. O circuito é composto por uma rede de resistores configurada como divisor de tensão com ganho controlado a partir das chaves analógicas SW_1 a SW_6 . Os ganhos de escala são selecionados a partir dos pares de chaves dados por SW_1 e SW_6 para escala de 1000 V, SW_2 e SW_5 para 400 V e por fim SW_3 e SW_4 para 100 V.

Para o cálculo dos resistores dados pelo divisor resistivo, compreendido entre os resistores R_1 até R_{18} , deve-se levar em conta a tensão imposta nas chaves analógicas utilizadas para modificar as escalas de tensão. Utilizou-se a chave analógica DG411LE, sendo que a tensão máxima empregada nos terminais dessa chave, em relação ao ponto de referência do circuito, é o valor de alimentação do dispositivo, dados por +5 V e -5 V acrescidos de um valor de 0,3 V, em função dos diodos de grampeamento interno ao circuito integrado [24]. O maior valor de tensão imposta nas chaves acontece quando o ganho do divisor resistivo é mínimo, isto é, quando SW_3 e SW_4 estão ativos. Com isso, a tensão imposta sobre a chave SW_1 é a soma das tensões sobre os resistores R_{13} , R_{14} e R_{15} . Analogamente, a tensão sobre a chave SW_6 é a soma das tensões sobre os resistores R_{16} , R_{17} e R_{18} .

Considerando que o projeto leva em conta os valores de tensão de 1000 V, 400 V e 100 V, calcula-se o ganho máximo que o divisor resistivo pode assumir em função do valor máximo da tensão sobre as chaves SW_1 e SW_6 . O presente projeto assumiu um valor máximo de tensão $V_{sw_{max}}$ como 2,5 V.

É importante mencionar que no circuito do protótipo utilizou-se de capacitores dispostos em paralelo com os resistores série, tendo a finalidade de compensar as capacitâncias intrínsecas das chaves SW_1 até SW_6 . Pode-se verificar esses capacitores através do esquemático do divisor resistivo dado pelo Apêndice C.

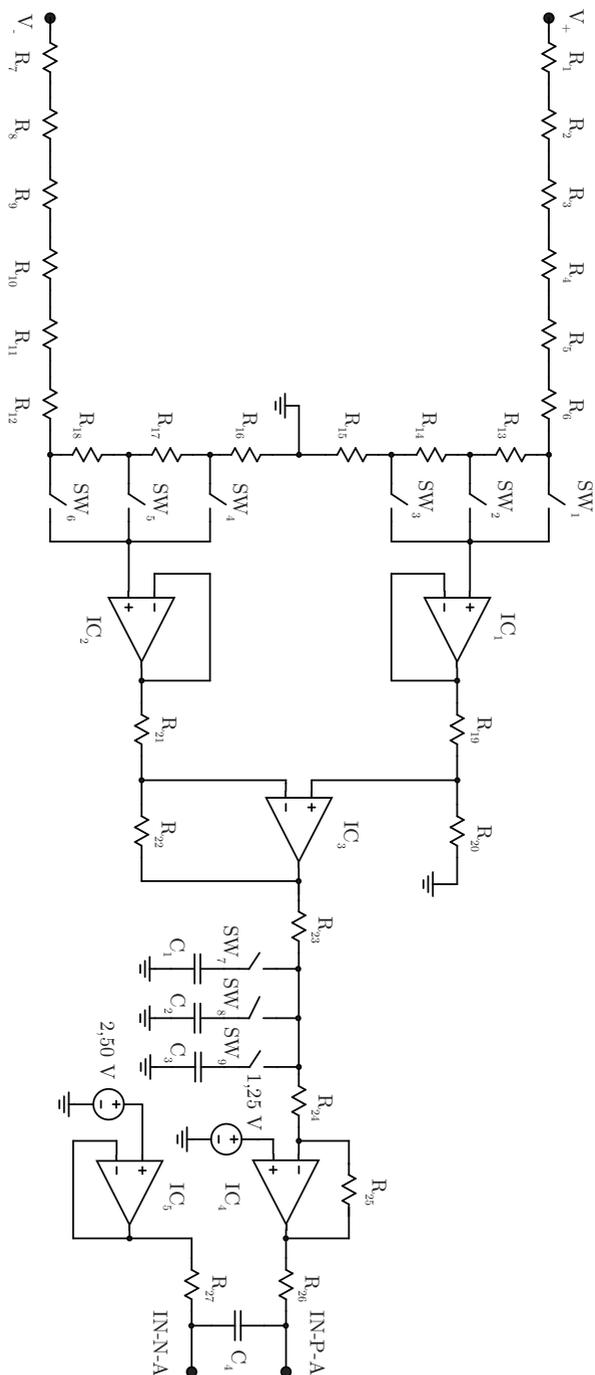


Figura 3.8 – Circuito de aquisição de tensão.

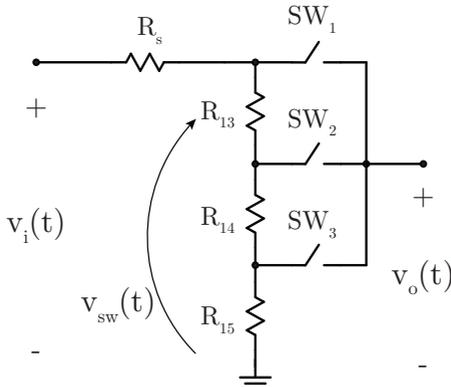


Figura 3.9 – Parcela do circuito que emprega o divisor resistivo.

Pela relação (3.1) tem-se uma simplificação da rede de resistores conectadas em série para um único resistor R_s .

$$R_s = R_1 + R_2 + R_3 + R_4 + R_5 + R_6 \quad (3.1)$$

Analisando o circuito da Figura 3.9 tem-se que a tensão $V_{sw}(t)$, que é igual a tensão na chave SW_1 , atinge o valor V_{smax} , de 2,5 V, quando $V_i(t)$ é 1000 V, e SW_3 é a chave selecionada. Com isso, tem-se que o ganho do divisor resistivo, que condiciona a entrada de 1000 V para 2,5 V é de $2,5 \times 10^{-3}$, como apresentado por (3.2). Nota-se que a equação da tensão imposta sobre a chave SW_1 é a mesma para a escala de ganho G_{100V} , resultando na primeira relação de ganho para as escalas de tensão.

$$G_{100V} = \frac{R_{13} + R_{14} + R_{15}}{R_{13} + R_{14} + R_{15} + R_s} = \frac{2,5}{1000} = 2,5 \times 10^{-3} \quad (3.2)$$

Esse valor de ganho gera na saída dos amplificadores operacionais IC_1 e IC_2 uma diferença de tensão de 250 mV para o fundo de escala. Assim, os ganhos de G_{400V} e G_{100V} serão projetados de modo a gerar esse mesmo nível de tensão de 250 mV na saída dos amplificadores. Com isso, o amplificador diferencial

do próximo estágio, que possui ganho fixo, pode amplificar esse nível de 250 mV para o fundo de escala do conversor analógico-digital, que é de $V_{ref} = 2.5 V$. Pela Tabela 3.1 tem-se o valor dos ganhos de escala G_{1000V} , G_{400V} e G_{100V} , assim como os valores de tensão na saída dos amplificadores IC_1 e IC_2 para os fundos de escala.

Tabela 3.1 – Relação de ganho do circuito de tensão para as escalas definidas.

<i>Escala</i>	<i>Chaves</i>	Fundo de Escala	<i>Ganho de Escala</i>
1000 V	SW ₃ e SW ₄	250 mV	$2,50 \times 10^{-4}$
400 V	SW ₂ e SW ₅	250 mV	$6,25 \times 10^{-4}$
100 V	SW ₁ e SW ₆	250 mV	$2,50 \times 10^{-3}$

Torna-se possível montar as relações de ganho em função dos resistores de modo a escolher o valores adequados. Para a rede de resistores em série, compreendida entre os resistores R_1 até R_6 , escolheu-se o valor de 130 k Ω . Dessa forma, a tensão máxima sobre cada resistor é de 166,67 V, considerando uma tensão imposta sobre a rede de 1000 V. Também, determinou-se R_s como sendo 780 k Ω . As relações de ganho para as três escalas de tensão são dadas por (3.3).

$$\begin{aligned}
 G_{1000V} &= \frac{R_{15}}{R_{13} + R_{14} + R_{15} + R_s} \\
 G_{400V} &= \frac{R_{14} + R_{15}}{R_{13} + R_{14} + R_{15} + R_s} \\
 G_{100V} &= \frac{R_{13} + R_{14} + R_{15}}{R_{13} + R_{14} + R_{15} + R_s}
 \end{aligned} \tag{3.3}$$

Escrevendo o sistema de equações na forma $Ax = B$ e tomando os valores de ganho da Tabela 3.1 tem-se (3.4).

$$A = \begin{pmatrix} 0,00025 & 0,00025 & 0,99975 \\ 0,00625 & 0,999375 & 0,999375 \\ 0,9975 & 0,9975 & 0,9975 \end{pmatrix} \quad (3.4)$$

$$x = \begin{pmatrix} R_{13} \\ R_{14} \\ R_{15} \end{pmatrix} B = \begin{pmatrix} 0,25 \\ 0,25 \\ 0,25 \end{pmatrix}$$

Dado que a rede de resistores é simétrica, os valores do resultado dado por (3.5) são iguais entre as parcelas superior e inferior do circuito.

$$x = \begin{pmatrix} R_{13} \\ R_{14} \\ R_{15} \end{pmatrix} = \begin{pmatrix} R_{18} \\ R_{17} \\ R_{16} \end{pmatrix} = \begin{pmatrix} 1468,0 \Omega \\ 292,3 \Omega \\ 194,6 \Omega \end{pmatrix} \quad (3.5)$$

Com o resultado proveniente de (3.5) fez-se a aproximação dos valores para os típicos comerciais. Dessa forma, fora escolhido o valor de 1,5 k Ω para R_{13} e R_{18} , 300 Ω para R_{14} e R_{17} e 200 Ω para R_{15} e R_{16} .

Pode-se notar que a variação dos valores dos resistores cria uma divergência entre o ganho do circuito, dado pela Tabela 3.1, e o ganho real, com os resistores aproximados para os valores comerciais típicos. A fim de corrigir esse problema, calcula-se a variação dos ganhos para então projetar o próximo estágio do circuito de aquisição, dado pelo amplificador diferencial.

Utilizando-se das relações dadas em (3.3) a fim de obter os ganhos $G_{v_{novo}}$ para os três casos de valores de pico de tensão junto com os valores de G_v explicitados na Tabela 3.1, tem-se o resultado da variação de ganho, explicitado em (3.6). Os valores corrigidos são dados pela Tabela 3.2.

$$G_{var} = \frac{G_{v_{novo}}}{G_v} = 1,023 \quad (3.6)$$

Tabela 3.2 – Valores corrigidos de tensão de saída e ganhos do divisor resistivo para as diferentes escalas de tensão.

<i>Escala</i>	$V_{v_{novo}}$	$G_{v_{novo}}$
1000 V	256 mV	$2,558 \times 10^{-4}$
400 V	256 mV	$6,394 \times 10^{-4}$
100 V	256 mV	$2,558 \times 10^{-3}$

Com o valor de $V_{v_{novo}}$, calcula-se o ganho necessário para o circuito de amplificação diferencial a fim de condicionar o sinal para o fundo de escala do ADC, que é de 2.5 V [25]. Portanto, esse ganho é dado por (3.7).

$$G_{vd} = \frac{V_{ref}}{V_{v_{novo}}} = 9,775 \quad (3.7)$$

A partir de (3.8) tem-se as relações que definem a tensão de saída e ganho do estágio diferencial. Vale lembrar que R_{20} e R_{22} possuem o mesmo valor, assim como R_{19} e R_{21} .

$$V_{vd} = \frac{R_{20}}{R_{19}}(V_+ - V_-) \quad (3.8)$$

$$G_{vd} = \frac{21,505 \text{ k}\Omega}{2,2 \text{ k}\Omega} = 9,775$$

3.1.1 Resultados de Simulação

A Figura 3.10 apresenta a resposta do circuito de medição para a escala de tensão de 1000 V. Com isso, tem-se também a saída do divisor resistivo, a saída do amplificador diferencial e entrada do conversor analógico-digital. A Figura 3.11 o resultado para as tensões nas chaves com a aplicação de 1000 V na entrada de medição V_+ em relação ao ponto de referência do circuito.

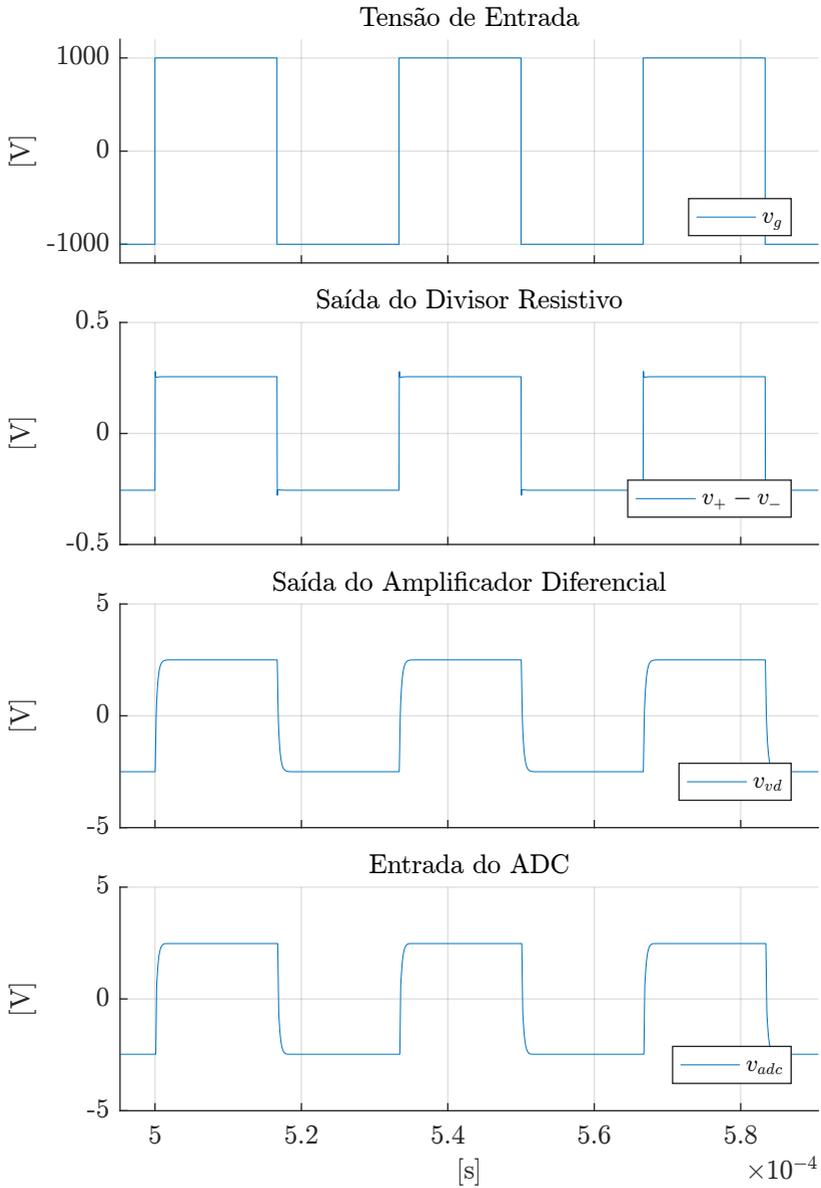


Figura 3.10 – Formas de onda do circuito de aquisição de tensão.

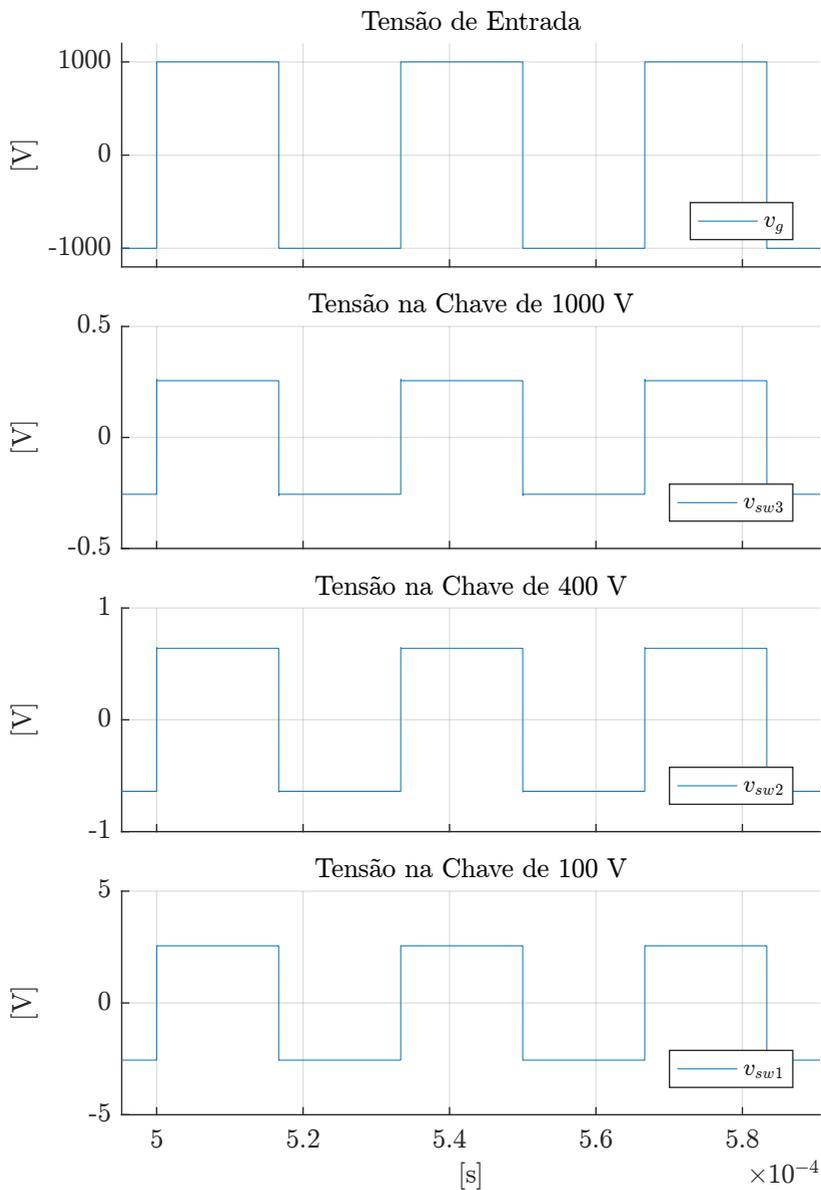


Figura 3.11 – Tensão imposta nas chaves seletoras de ganho.

3.2 AQUISIÇÃO DE CORRENTE

O circuito de aquisição de corrente, apresentado na Figura 3.12, possui capacidade de selecionar três escalas de corrente a partir das chaves analógicas SW_{10} , SW_{11} e SW_{12} . O projeto levou em conta a escolha de valores de pico de corrente a fim de condicioná-los para o fundo de escala do conversor analógico-digital, de 2.5 V. Com isso, o projeto fora desenvolvido a partir da definição dos valores de pico de 35 A, 15 A e 5 A.

Através da Figura 3.12 realiza-se a análise de ganho da estrutura diferencial dada pelo amplificador operacional IC_6 . Essa configuração permite amplificar o sinal diferencial de entrada do amplificador, proveniente da tensão imposta sobre o resistor R_{shunt} , dado por pela relação $V_+ - V_-$. Essa queda de tensão é ocasionada pela passagem de corrente do sistema sobre testes pelo resistor shunt.

Como o resistor R_{shunt} possui valor de $5\text{ m}\Omega$ e capacidade de dissipar 3 W, projetou-se o sistema de modo a suportar um valor de pico em 35 A. O cálculo de potência dissipada no resistor levará em conta uma onda puramente senoidal com valor de pico em 35 A, como mencionado anteriormente. O valor de potência é dado pela relação (3.9). A fim de garantir que o sistema tenha a capacidade de dissipar o calor devido a perda Joule, aplicou-se um dissipador no resistor shunt.

$$P_{shunt} = R_{shunt} \left(\frac{I_{pico}}{\sqrt{2}} \right)^2 = 3.062\text{ W} \quad (3.9)$$

O estágio posterior ao sensoriamento, composto pelo resistor shunt, é o amplificador diferencial, projetado com ganho dado por (3.10). A estrutura diferencial possui uma característica interessante, que é uma maior rejeição de ruídos de modo comum em relação às estruturas inversora e não-inversora.

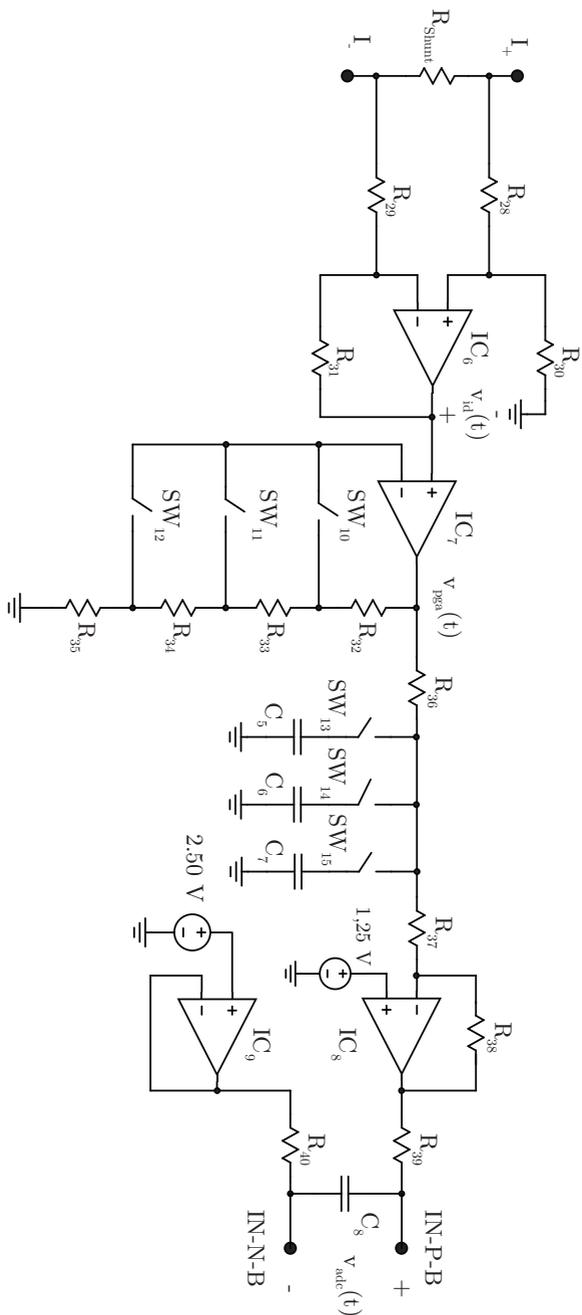


Figura 3.12 – Circuito de aquisição de corrente.

$$V_{id} = \frac{R_{30}}{R_{28}}(V_+ - V_-) \quad (3.10)$$

$$G_{id} = \frac{8 \text{ k}\Omega}{2 \text{ k}\Omega} = 4$$

A estrutura dada pelo amplificador de ganho variável deve condicionar o sinal do estágio diferencial para o nível de tensão dado pelo fundo de escala do ADC. Para isso, a escolha dos resistores R_{32} , R_{33} , R_{34} e R_{35} deve respeitar as relações de ganho conforme os níveis de corrente de pico previamente especificados para as escalas de corrente. Essas relações são dadas por (3.11) e a Tabela 3.3 apresenta os valores de tensão na saída do estágio diferencial assim como o ganho G_{PGA} necessário para cada escala.

$$V_{id} = G_{id}R_{shunt}I_{pico}$$

$$G_{PGA} = \frac{V_{ref}}{V_{id}} \quad (3.11)$$

Tabela 3.3 – Tensão de saída do estágio diferencial e ganhos do amplificador de ganho variável.

I_{pico}	$Chave$	V_{id}	G_{PGA}
35 A	SW_{10}	0,7 V	3,571
15 A	SW_{11}	0,3 V	8,333
5 A	SW_{12}	0,1 V	25

As relações que descrevem os ganhos do amplificador de ganho variável podem ser vistas em (3.12). Assim, com as especificações de ganho dadas pela Tabela 3.3 torna-se possível montar as equações em forma de um sistema linear.

$$\begin{aligned}
 G_{35A} &= 1 + \frac{R_{32}}{R_{33} + R_{34} + R_{35}} \\
 G_{15A} &= 1 + \frac{R_{32} + R_{33}}{R_{34} + R_{35}} \\
 G_{5A} &= 1 + \frac{R_{32} + R_{33} + R_{34}}{R_{35}}
 \end{aligned} \tag{3.12}$$

Como o sistema é 3x3 e possui 4 incógnitas, adota-se o valor de $9k\Omega$ para o resistor R_{32} a fim de tornar o sistema resolvível, resultando em (3.13). Escrevendo na forma matricial, $Ax = B$, tem-se (3.14).

$$\begin{aligned}
 R_{32} + R_{34} + R_{35} &= 3500,6 \\
 -R_{32} + 7,333R_{34} + 7,333R_{35} &= 9000 \\
 -R_{32} - R_{34} + 24R_{35} &= 9000
 \end{aligned} \tag{3.13}$$

$$\begin{aligned}
 x &= \begin{pmatrix} R_{33} \\ R_{34} \\ R_{35} \end{pmatrix} \\
 A &= \begin{pmatrix} 1 & 1 & 1 \\ -1 & 7,333 & 7,333 \\ -1 & -1 & 24 \end{pmatrix} \\
 B &= \begin{pmatrix} 3500,6 \\ 9000 \\ 9000 \end{pmatrix}
 \end{aligned} \tag{3.14}$$

O resultado do sistema linear é dado por (3.15).

$$x = \begin{pmatrix} R_{33} \\ R_{34} \\ R_{35} \end{pmatrix} = \begin{pmatrix} 2005 \Omega \\ 1000 \Omega \\ 500 \Omega \end{pmatrix} \tag{3.15}$$

Assumiu-se então os valores comerciais de $9 k\Omega$, $2 k\Omega$, $1 k\Omega$ e 500Ω para os resistores R_{32} , R_{33} , R_{34} , e R_{35} .

3.2.1 Resultados de Simulação

Simulações foram realizadas através do software Tina-TI, como no caso do circuito de tensão, a fim de caracterizar a resposta do circuito projetado para as escalas de corrente de 35 A, 15 A e 5 A. Dessa forma, a Figura 3.13 apresenta a resposta da saída do amplificador diferencial v_{id} , do amplificador de ganho variável v_{pga} e da entrada do ADC v_{adc} , para as escalas de corrente.

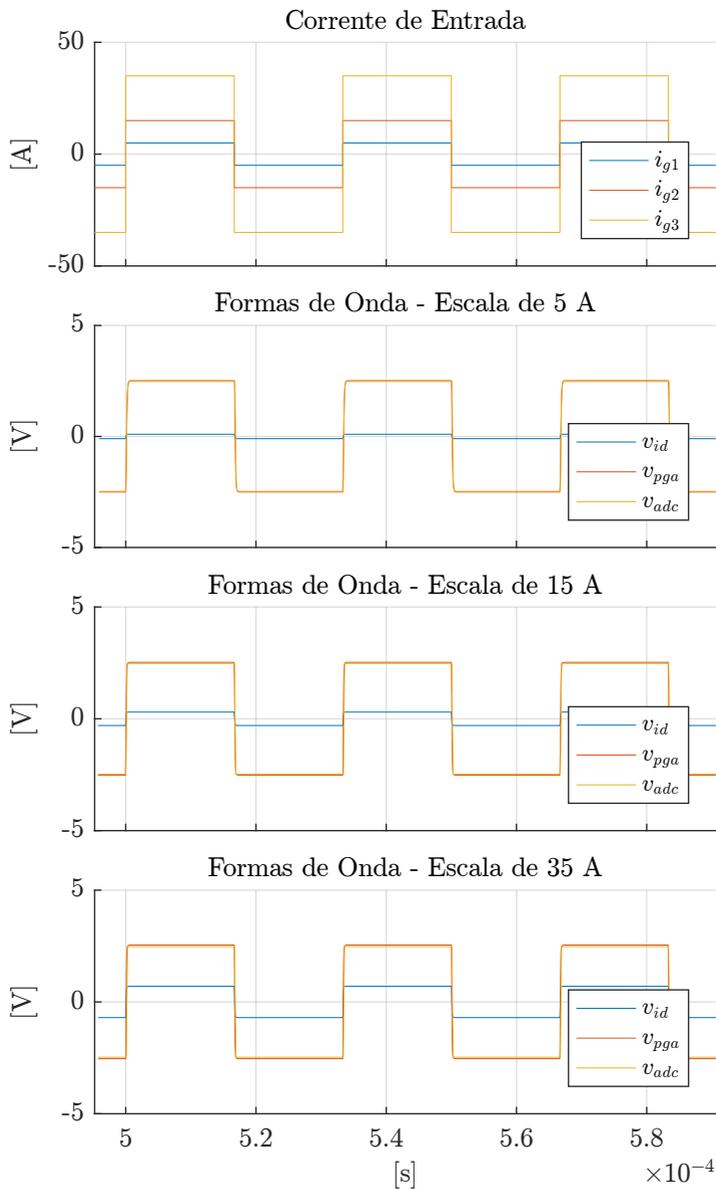


Figura 3.13 – Formas de onda do circuito de aquisição de corrente.

3.3 FILTRO DE LIMITAÇÃO DE BANDA

Em um sistema de conversão de dados analógico-digital, deseja-se converter um sinal contínuo no tempo $x(t)$ em amostras discretas $x[nT_s]$, sendo $T_s = \frac{1}{f_s}$. Essa conversão leva em conta o resultado proveniente do Teorema da Amostragem de Nyquist-Shannon [26], tratando que um sinal limitado em uma determinada banda B não deve ser superior a metade da frequência de amostragem f_s , como mostrado em (3.16). Esse resultado é ilustrado na Figura 3.14(a), em que um sinal é amostrado, replicando o espectro centrado em múltiplos inteiros da frequência de amostragem f_s .

$$f_s > 2B, \quad B < \frac{f_s}{2} \quad (3.16)$$

Contudo, a Figura 3.14(b) traz um caso em que se não respeita o Teorema de Nyquist-Shannon. Vê-se que o sinal, com banda $B > \frac{f_s}{2}$, recobre o espectro acima de $\frac{f_s}{2}$ com o restante dos espectros dispostos nos múltiplos da frequência de amostragem. A fim de evitar o recobrimento dos espectros, que introduz distorções no sinal amostrado, aplica-se um filtro passa-baixas, ou também conhecido como filtro anti-recobrimento. A aplicação do filtro é representada na Figura 3.14(c). No caso, utilizou-se de um filtro passa-baixa ideal com frequência de corte abaixo de $\frac{f_s}{2}$.

Ademais, o filtro limitador de banda empregado no projeto possui frequência de corte abaixo da frequência de Nyquist. Para a frequência de amostragem do conversor analógico-digital, que é de 750 kHz, a frequência de Nyquist é de 375 kHz, considerando o caso ideal. No sistema de aquisição desenvolvido, três frequências de corte podem ser selecionadas, conforme seleção de chaves seletoras, como no caso para a escolha de ganho de tensão e corrente. O esquemático do filtro é igual para ambos os canais, e pode ser visto pela Figura 3.15.

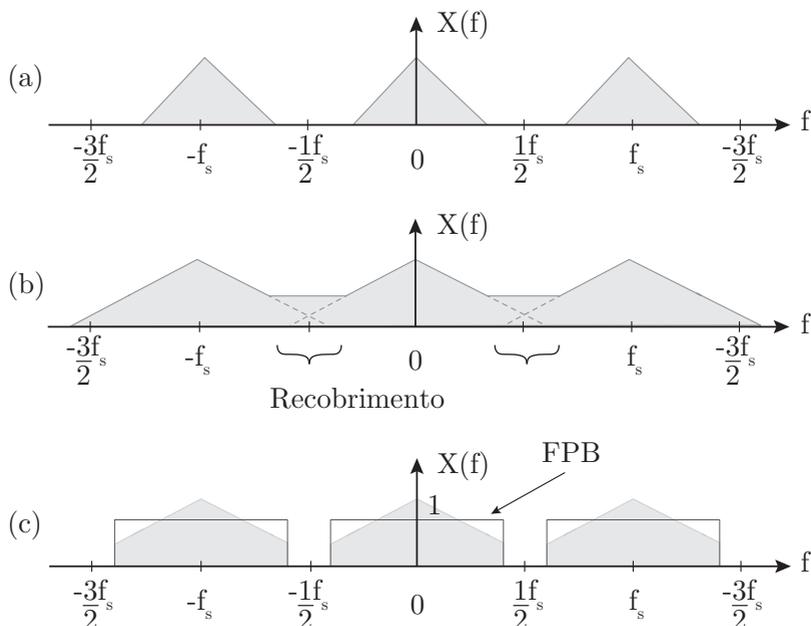


Figura 3.14 – (a) Amostragem de um sinal que respeita a Taxa de Nyquist. (b) Amostragem de um sinal que não respeita a Taxa de Nyquist causando recobrimento de espectros (c) Filtro passa-baixa (FPB) atuando como filtro anti-recobrimento.

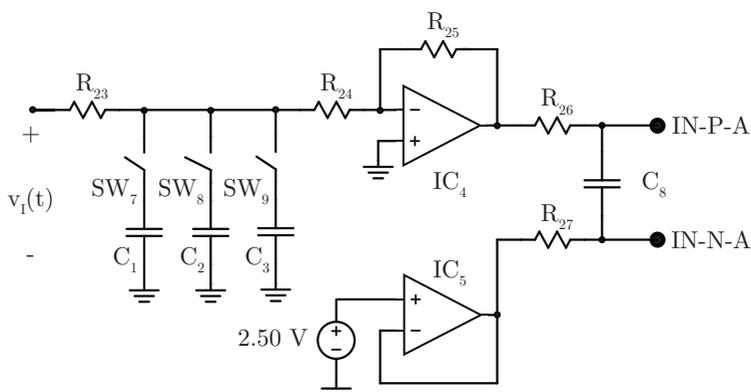


Figura 3.15 – Circuito de aplicação do filtro passa-baixas e driver para o ADC

Como comentado, considerou-se três frequências de corte para o dimensionamento do filtro passa-baixas. As frequências de corte são de 100 kHz para a chave SW_7 , 10 kHz para a chave SW_8 e 1 kHz para a chave SW_9 .

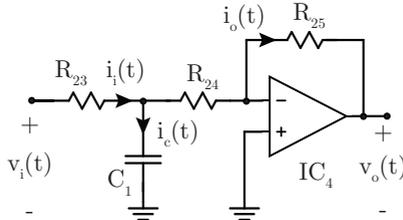


Figura 3.16 – Análise do filtro passa-baixas

A análise do circuito da Figura 3.16 é realizado através de (3.17) e (3.18).

$$\begin{aligned} v_o(s) &= -i_o(s)R_{25} \\ v_{c1}(s) &= i_o R_{24} \\ v_{c1}(s) &= -v_o(s) \frac{R_{24}}{R_{25}} \end{aligned} \quad (3.17)$$

Aplicando a Lei dos Nós no nó do capacitor C_1 , tem-se o equacionamento dado por (3.18). Para simplificar a análise, assume-se que o valor de R_{23} e R_{24} são iguais, utilizado apenas o valor de R_{24} nas relações (3.18) e (3.19)

$$\begin{aligned} i_i(s) &= i_c(s) + i_o(s) \\ \frac{v_i(s) - v_c(s)}{R_{24}} &= \frac{v_c(s)}{Z_c(s)} + \frac{v_c(s)}{R_{24}} \end{aligned} \quad (3.18)$$

De modo a obter a função transferência do circuito para retirar a informação da frequência de corte do filtro passa-baixas, combina-se as relações (3.17) e (3.18), resultando em (3.19).

$$\frac{v_o(s)}{v_i(s)} = -\frac{R_{25}}{R_{24}} \frac{1}{(2 + sC_1R_{24})} \quad (3.19)$$

A frequência de corte do filtro é dada por (3.20).

$$|2 + sC_1R_{24}| = 0$$

$$s = j\omega, \quad \omega = \frac{2}{C_1R_{24}} \quad (3.20)$$

$$f_c = \frac{1}{\pi C_1R_{24}}$$

Como o sinal que é condicionado para o estágio do filtro passa-baixas não precisa ser amplificado, configura-se o ganho da estrutura para unitário. Nesse projeto utilizou-se os valores de $1 \text{ k}\Omega$ para o resistor R_{25} e $500 \text{ }\Omega$ para os resistores R_{24} e R_{23} . Analogamente para os resistores R_{38} , R_{37} e R_{36} , do circuito de corrente.

Para os capacitores C_1 , C_2 e C_3 do circuito da Figura 3.15 tem-se os valores dados na Tabela 3.4. Os valores dos capacitores mencionados são iguais aos dos capacitores C_5 , C_6 e C_7 , respectivamente.

Tabela 3.4 – Valores dos capacitores para as frequências de corte escolhidas.

f_c	Chave	Capacitor
100 kHz	SW ₇ / SW ₁₃	6.4 nF
10 kHz	SW ₈ / SW ₁₄	64 nF
1 kHz	SW ₉ / SW ₁₅	640 nF

3.4 CIRCUITOS DE ALIMENTAÇÃO

O diagrama de blocos da Figura 3.17 apresenta o diagrama de alimentação da placa de aquisição. Projetado para entrada de 15 V, o sistema conta com um conversor CC/CC isolado com níveis de tensão de saída de $\pm 7.5 \text{ V}$. A topologia proposta para o

conversor chaveado é a LLC ressonante, empregando um retificador ponte completa com um circuito dobrador de tensão, a fim de produzir a alimentação simétrica.

Os componentes que sucedem a fonte chaveada são os reguladores lineares, utilizados para ajustar o nível de tensão que adequado para alimentar os circuitos analógicos e digitais utilizados no protótipo. Como pode ser visto no diagrama, empregam-se reguladores fixos com os níveis de tensão de ± 5 V, 3.3 V e 1.8 V. Também, separou-se as alimentações digitais e analógica dos circuitos.

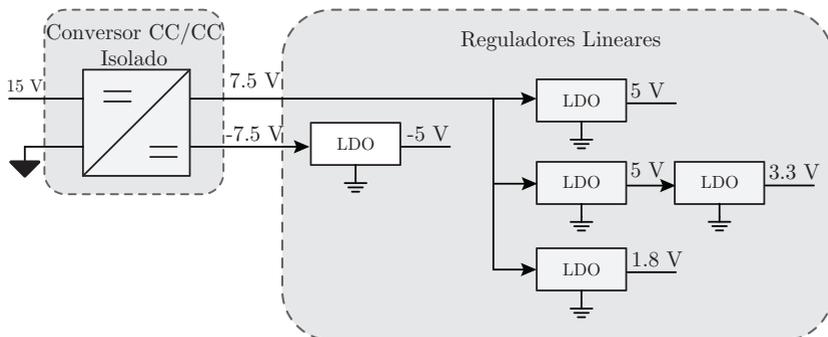


Figura 3.17 – Diagrama da estrutura de alimentação do protótipo

3.4.1 Conversor CC/CC Ressonante

Na Figura 3.18 vê-se a estrutura topológica do conversor utilizado. A escolha dessa topologia levou em conta a necessidade de isolamento entre os enrolamentos do transformador, além de possui alimentação simétrica. Também, a estrutura LLC permite atingir a comutação suave, em que o transistor entra em condução com nenhuma imposição de tensão nos terminais das chaves, diminuindo as perdas de comutação quando se comparado a estruturas que realizam comutação forçada [27].

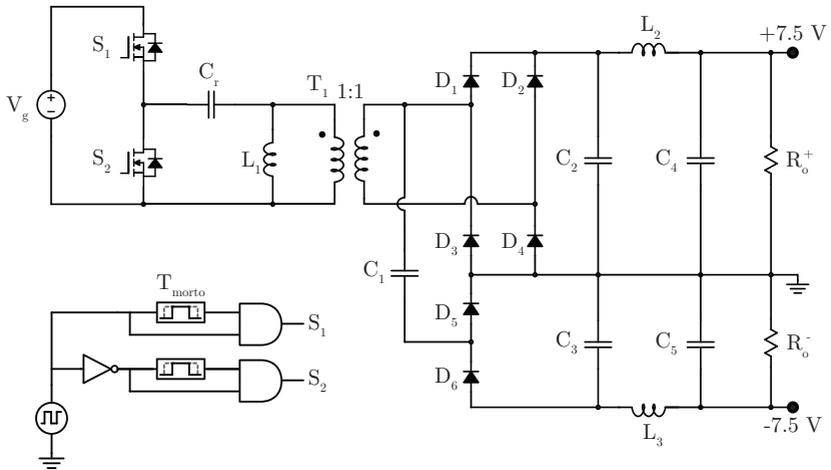


Figura 3.18 – Conversor ressonante CC-CC isolado para alimentação do protótipo.

Em relação a estrutura do conversor, tem-se na Figura 3.19 a estrutura do tanque ressonante, explicitando a capacitância de saída C_{oss} das chaves S_1 e S_2 , além do capacitor série C_r , indutância de dispersão série L_r e indutância magnetizante L_m do transformador T_1 , além de um indutor externo L_1 . Os valores dos componentes relacionados ao tanque ressonante podem ser vistos através da Tabela 3.5. $V_{hb}(t)$ é a forma de onda resultante do chaveamento das chaves S_1 e S_2 .

Tabela 3.5 – Valor dos componentes do tanque ressonante.

<i>Componente</i>	<i>Valor</i>
C_r	100 nF
L_r	2,75 μ H
L_m	137 μ H
L_1	65 μ H

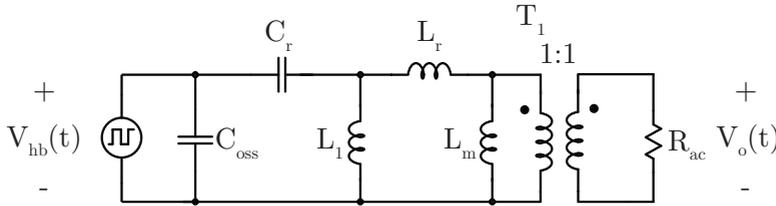


Figura 3.19 – Tanque ressonante do conversor CC-CC isolado para alimentação do protótipo.

Com esse valores pode-se analisar as frequências de ressonância do tanque ressonante a fim de caracterizar as regiões de operação que garantem a comutação suave. A primeira frequência de ressonância, dada pela combinação do capacitor C_r e das indutâncias L_1 , L_r e L_m é obtida através da relação (3.21). Já para a segunda frequência de ressonância, tem-se o valor dado pela combinação do capacitor C_r e da indutância de dispersão do transformador, L_r [28].

$$\begin{aligned}
 X_{Leq} &= (X_{Lr} + X_{Lm}) // X_{L1} \\
 X_{Cr} &= X_{Leq} \\
 f_{r1} &= \frac{1}{2\pi\sqrt{L_{eq}C_r}} \\
 f_{r1} &= 75,561 \text{ kHz}
 \end{aligned} \tag{3.21}$$

$$\begin{aligned}
 X_{Cr} &= X_{Lr} \\
 \frac{1}{j\omega_{r2}C_r} &= j\omega_{r2}L_r \\
 \omega_{r2} &= 2\pi f_{r2}, \quad f_{r2} = \frac{1}{2\pi\sqrt{L_rC_r}} \\
 f_{r2} &= 303,5 \text{ kHz}
 \end{aligned} \tag{3.22}$$

O fator de qualidade do tanque ressonante segue a relação

(3.23) [29]. R_{ac} é o valor da carga refletida para o enrolamento primário do transformador, tendo em vista a análise realizada pela componente fundamental da forma de onda quadrada, que é aproximada por uma forma senoidal, além da consideração de um retificador ponte completa na saída do conversor [30].

$$Q = \frac{\sqrt{\frac{L_r}{C_r}}}{R_{ac}} \quad (3.23)$$

$$R_{ac} = \frac{8}{\pi^2} n^2 \frac{V_{out}}{P_{out}} \quad (3.24)$$

A Figura 3.20 traz resultados da resposta em frequência do tanque ressonante apresentado na Figura 3.19. Com isso, o diagrama da Figura 3.20 relaciona a tensão de saída pela tensão aplicada na entrada do tanque ressonante $\frac{V_o(t)}{V_{hb}(t)}$, sendo que a carga do tanque segue a relação (3.24). A fim de validar os cálculos realizados para as frequências de ressonância f_{r1} e f_{r2} , refere-se à Figura 3.20. A partir desse resultado, escolheu-se operar o conversor com a frequência de comutação de 304 kHz, igual a f_{r2} , garantindo comutação suave para todo perfil de carga. Vale mencionar que o conversor projetado operará com fatores de qualidade Q menores que 0,5.

3.4.2 Formas de Onda

As Figuras 3.21, 3.22 e 3.23 apresentam as principais formas de onda do conversor operando com diferentes faixas de cargas. Nota-se que a corrente resultante no conversor meia ponte é capaz de carregar e descarregar as capacitâncias das chaves S_1 e S_2 completamente, garantindo comutação suave para qualquer condição de carga.

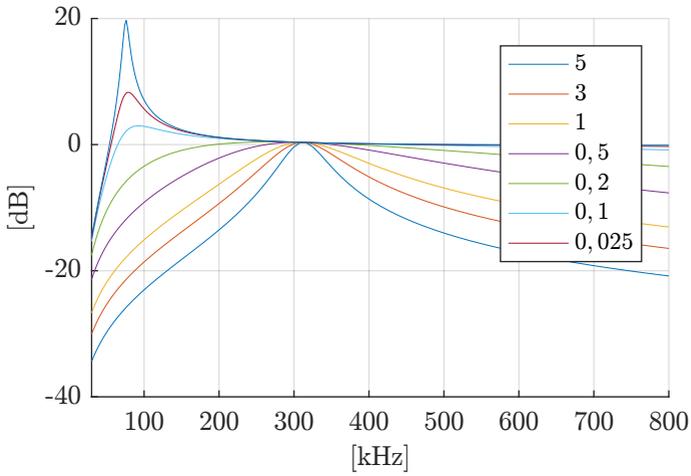


Figura 3.20 – Resposta em frequência do tanque ressonante para diferentes fatores de qualidade. A legenda representa o fator de qualidade Q .

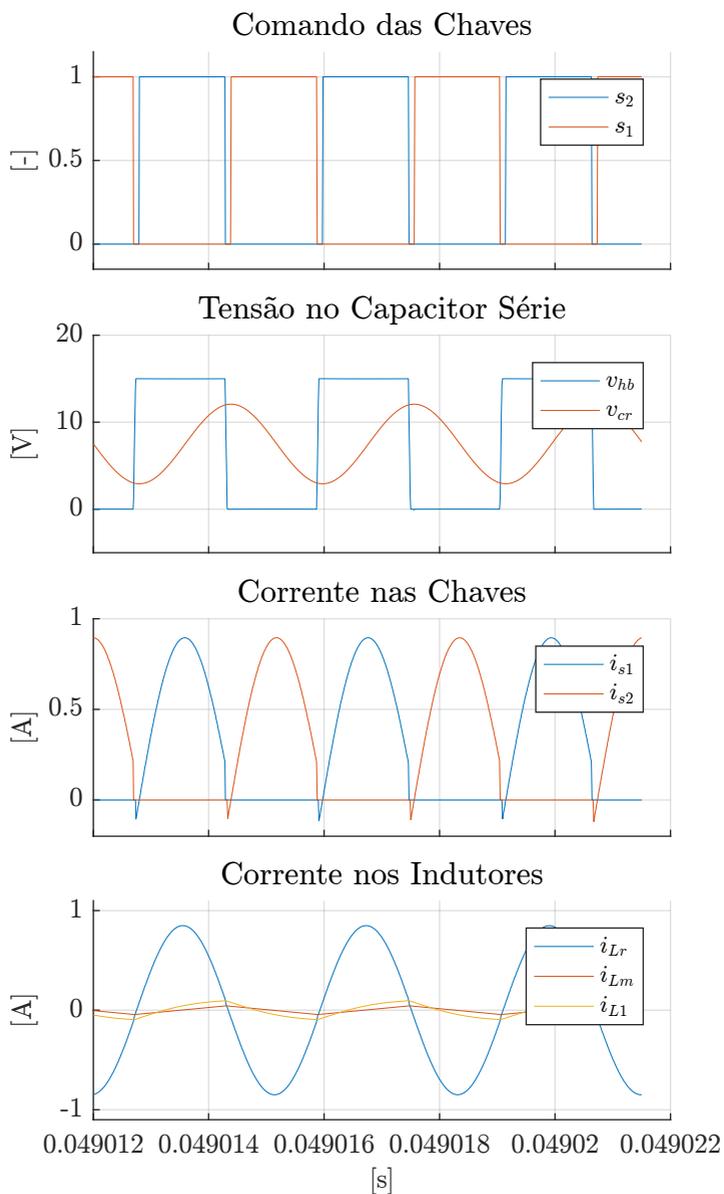


Figura 3.21 – Formas de onda do conversor operando com carga de 4,125 W.

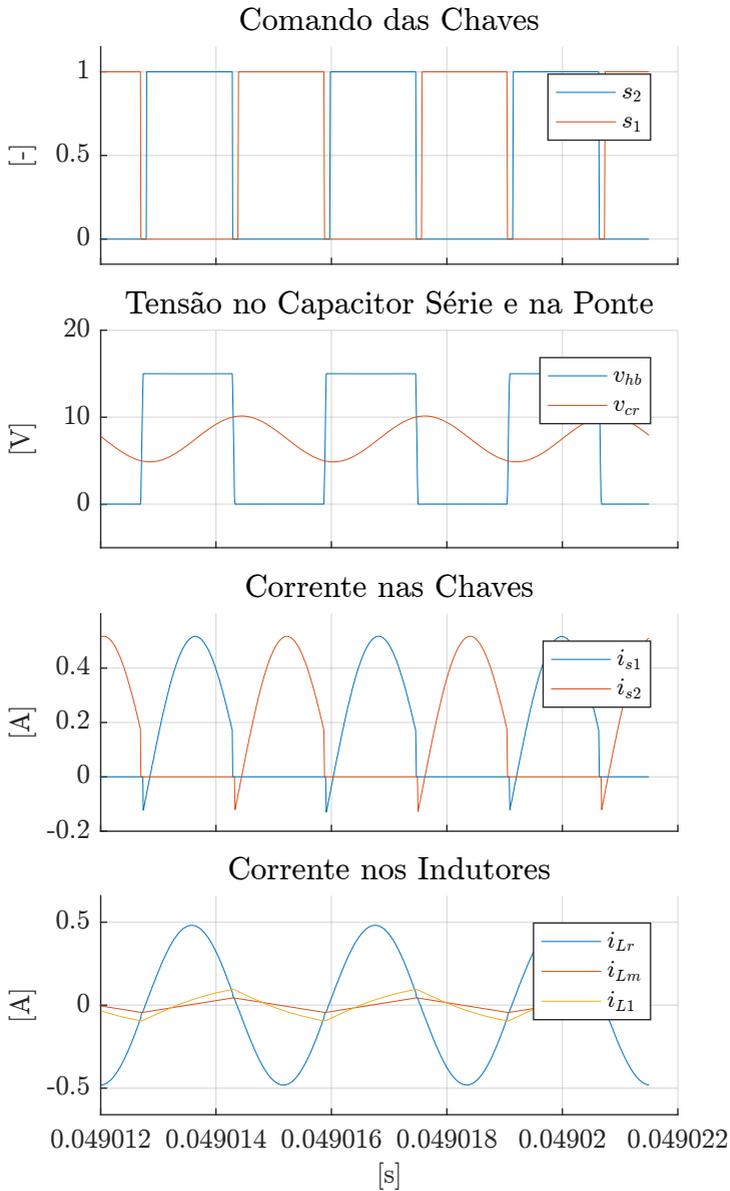


Figura 3.22 – Formas de onda do conversor operando com carga de 2,25 W.

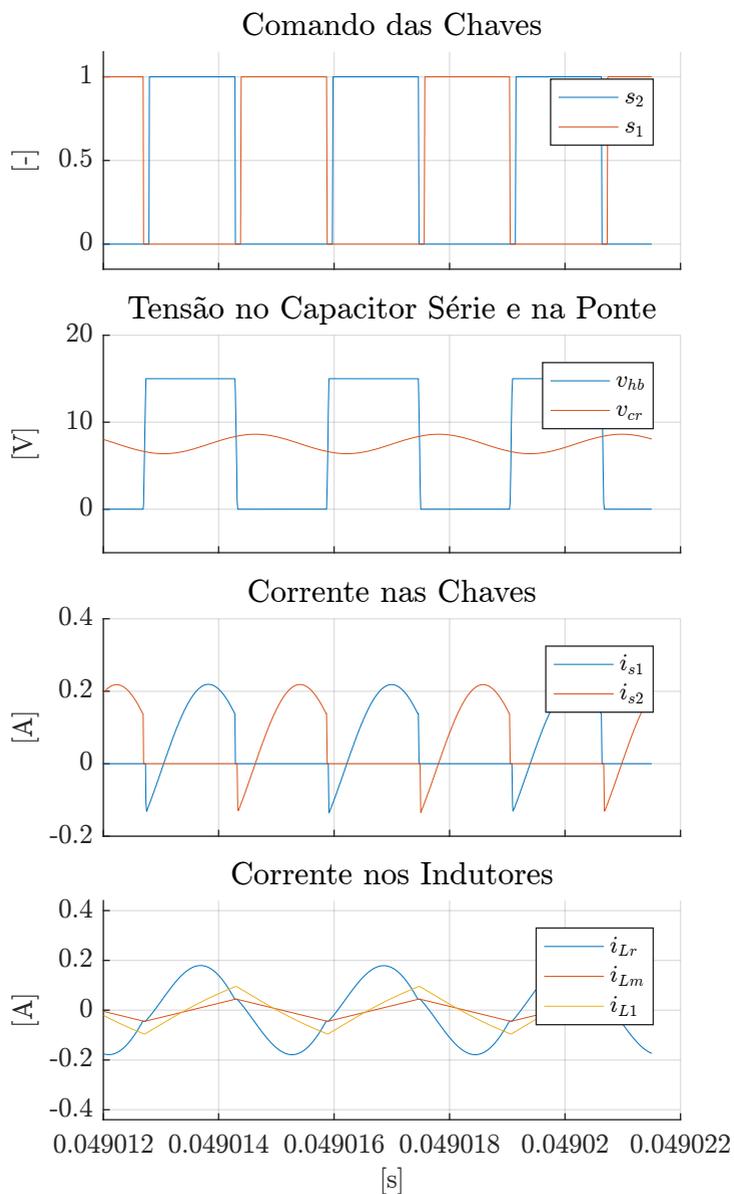


Figura 3.23 – Formas de onda do conversor operando com carga de 0,75 W.

4 PLATAFORMA DE HARDWARE DIGITAL

A solução proposta para a comunicação do protótipo de monitoramento com a placa de desenvolvimento e o computador pode ser vista pelo diagrama da Figura 4.24. Propõe-se embarcar um CPLD na placa de aquisição de tensão e corrente, possibilitando realizar a transmissão e recepção de dados através de isoladores digitais e fibras óticas. Os dados são transmitidos para uma placa de desenvolvimento DE-10 Nano SoC, da empresa Terasic. Essa placa embarca um SoC (*System-on-Chip*), que contém um FPGA Cyclone V e um processador ARM Dual Core A9 no mesmo circuito integrado. Após o processamento dos dados por parte do FPGA e do processador ARM, a transmissão dos pacotes é realizada em direção a um computador de uso geral, que executa uma interface gráfica.

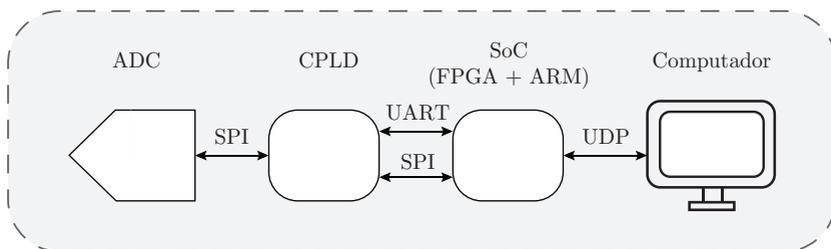


Figura 4.24 – Diagrama de blocos do sistema proposto, explicitando os protocolos de comunicação entre dispositivos.

Os protocolos de comunicação empregados na comunicação entre as unidades de processamento são explicitados na Figura 4.24. Entre o ADC e o CPLD, ambos presente na placa de aquisição, emprega-se o protocolo SPI (*Serial Protocol Interface*), que é um protocolo de comunicação sincronizado, em que o mestre, no caso o CPLD, envia o sinal de clock e o si-

nal de seleção para o respectivo escravo, que realiza a transmissão de dados sincronizada com o clock do seu mestre. Entre o CPLD e o SoC faz-se a transmissão em dois protocolos distintos, o UART (*Universal Asynchronous Receiver/Transmitter*) e uma modificação do SPI. Para essa SPI o CPLD, que nesse caso é o mestre, realiza a transmissão de dados e do sinal de clock para o FPGA, que é o escravo. Ao contrário do SPI, o protocolo UART é assíncrono. O uso da SPI é para a transmissão dos dados que requerem maiores taxas de transmissão, como o caso dos pacotes de dados convertidos pelo ADC. O protocolo UART é utilizado para transmissões que não requerem taxas de transmissões altas, como no caso para o recebimento de pacotes de configuração da placa de aquisição, selecionando os ganhos de escala e os filtros passa-baixas.

Com os pacotes de dados recebidos na placa de desenvolvimento, realiza-se a transmissão dos mesmos para regiões específicas da memória SDRAM, externa ao chip SoC, que são acessados também através do processador ARM. Dessa forma, o processador executa algoritmos que indentificam os pacotes que devem ser enviados para a interface de usuário, realizando essa transmissão através do protocolo de rede UDP (*User Datagram Protocol*).

O presente capítulo discutirá sobre o hardware desenvolvido, na linguagem Verilog, para os dispositivos de descrição de hardware, compostos pelos CPLD e FPGA.

4.1 HARDWARE DA PLACA DE AQUISIÇÃO

A proposta de desenvolvimento do hardware para o CPLD da placa de aquisição levou em conta uma série de requisitos, como os quais listados abaixo:

1. Comunicação com o ADC via protocolo SPI;

2. Comunicação via Fibras Óticas utilizando protocolo UART TX/RX;
3. Comunicação via Isolador Digital ISO7742 utilizando os protocolos SPI e UART;
4. Seletor de ganho para tensão e corrente, controlados via chaves seletoras situadas na placa de aquisição ou via pacotes recepcionados através das Fibras Óticas e/ou Isolador Digital;
5. Seletor de filtro passa-baixas para os canais de tensão e corrente, controlados via chaves seletoras situadas na placa de aquisição ou via pacotes recepcionados através das Fibras Óticas e/ou Isolador Digital;

A fim de cumprir com os requisitos propostos, a Figura 4.25 traz o diagrama de blocos do hardware projetado. Com isso, o diagrama apresenta a conexão entre os módulos de comunicação, assim como as entradas e saídas do CPLD. Pode-se ver que o sistema é composto pelo bloco *ADC SPI RX* que realiza a aquisição dos dados via comunicação SPI com o conversor analógico-digital, os blocos de transferência dos pacotes via protocolos SPI e UART, compostos pelos blocos *Isolador Digital SPI TX* e *Fibra UART TX*, além do bloco receptor de pacotes de configuração provenientes do sistema embarcado, de nome *Isolador Digital UART RX*. Os seletores das chaves analógicas ficam por conta dos blocos *Seletor de Filtro* e *Seletor de Ganho*.

O CPLD empregado na placa de aquisição da fabricante Intel é da família MAX V 5M240ZT [31] operando com clock de 48 MHz. Com isso, o bloco *ADC SPI RX* se comunica com o conversor ADC com um clock de 24 MHz a fim de amostrar os pacotes a uma taxa de 750 kHz. Após a recepção dos dados de 16 bits recepcionados pelas entradas *asc_sdo_a* e *adc_sdo_b*, o bloco *Isolador Digital SPI TX* realiza o envio do pacote formado

por ambos pacotes recebidos pelo ADC através de um protocolo a dois fios, um para o sinal de clock e outro para o sinal de dado. Dessa forma, o bloco forma um pacote de dados com um total de 35 bits, que inclui um bit no início, no fim e entre os dados, que são transmitidos junto com o clock de 48 MHz.

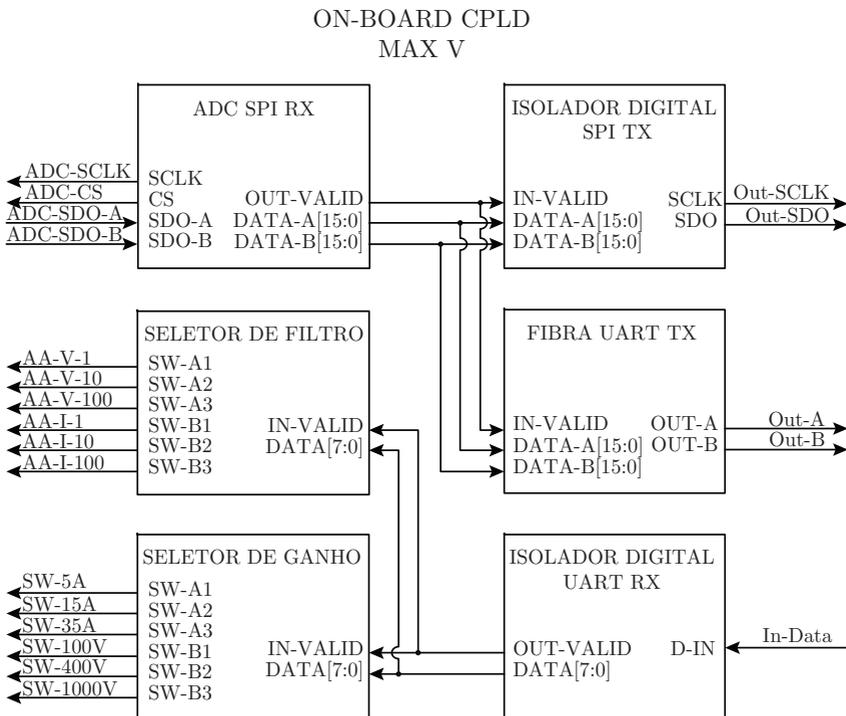


Figura 4.25 – Hardware descritivo do sistema desenvolvido para o CPLD da placa de aquisição.

4.2 HARDWARE DA PLATAFORMA DE DESENVOLVIMENTO DE-10 NANO SOC

Na Figura 4.26 é apresentado um diagrama simplificado do hardware implementado no FPGA da placa DE-10 Nano SoC. Pode-se ver os blocos de transmissão de dados para a SDRAM, de estimação de frequência, dado pelos blocos *Filtro BP* e *Det*.

Passagem Zero, que implementam o filtro banda passante e o detector de passagem por zero, além dos integradores de tensão, corrente e energia, além dos registradores utilizados para configurar e salvar variáveis, permitindo assim que o processador ARM acesse as variáveis e configure os blocos explicitados na figura. Ademais, o hardware implementa um detector de *trigger* de nível, que tem por função identificar uma determinada condição, que é o caso da passagem do sinal por um determinado nível de tensão. O uso do *trigger* auxilia na visualização de dados periódicos, tendo em vista que esse modo de visualização busca sobrepor os sinais mostrados na interface gráfica, tornando estático o sinal visualizado em relação ao ponto do *trigger*.

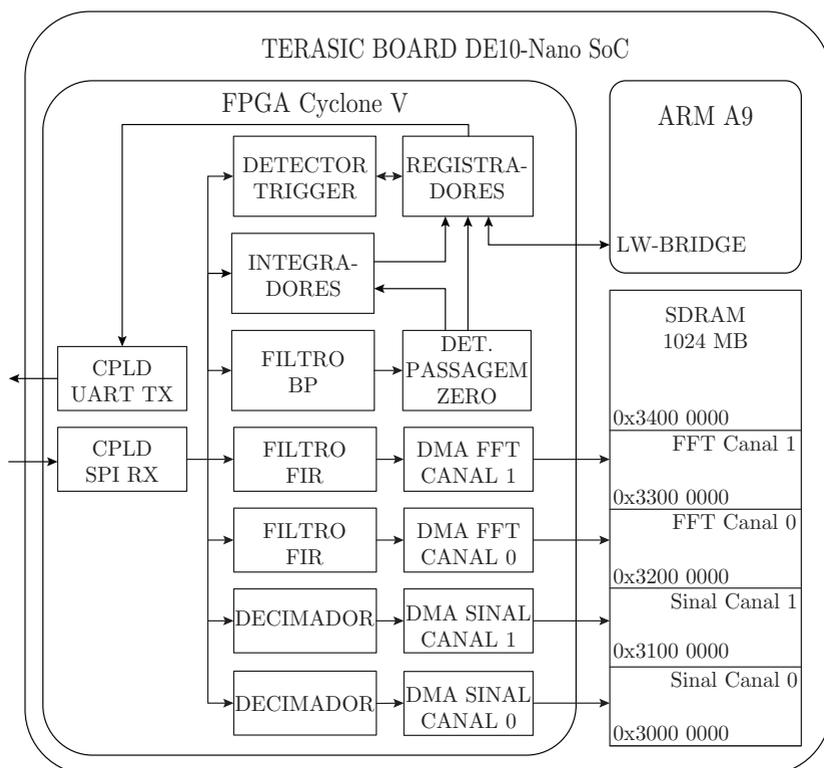


Figura 4.26 – Diagrama de blocos simplificado da estrutura implementada no FPGA.

4.2.1 Transmissão de dados para a SDRAM

O hardware inicia com a recepcionando os pacotes provenientes do CPLD através do bloco *CPLD SPI RX*. Após, realiza-se uma filtragem dos pacotes recebidos através de filtros média-móvel, dado pelos blocos *Filtro FIR*, e da subamostragem dos dados através de decimadores, dado pelos blocos *Decimador*. O ajuste do filtro média-móvel e do decimador é realizado através dos índices de decimação N_{dec} e de pacotes utilizados do filtro média móvel N_{mm} , que são configurados pelo processador ARM. Adicionalmente, ajustando os índices para o valor zero faz com que o dado seja transmitido sem nenhuma modificação.

Após, os pacotes são escritos em regiões específicas da memória SDRAM, externa ao chip SoC, utilizando diferentes DMAs (*Direct Memory Access*). Cada DMA tem por finalidade transmitir os pacotes recebidos em sua entrada para determinadas regiões de memória. No presente projeto, arbitrou-se o Canal 0 para o canal de tensão, e o Canal 1 para o canal de corrente. Ainda, fez-se a separação dos pacotes utilizados no cálculo da DFT e dos pacotes com o sinal que serão transmitidos para a interface de usuário.

O modo de funcionamento da DMA é como em um buffer circular, em que a cada transmissão de um pacote de dados para a memória o ponteiro para a memória é incrementado, varrendo toda a região da SDRAM reservada para a DMA. Ao chegar no endereço limite da região de memória especificada para aquela DMA, o bloco condiciona o ponteiro para o limite inferior da região, realizando o processo de varredura da memória novamente. No projeto, cada DMA é configurada de modo a mapear 16 MB na SDRAM externa ao chip. Nas taxas de amostragem e com o tamanho de dado do ADC, cada região mapeada é capaz de registrar o equivalente a 1,4 segundos de dados provenientes do sistema de aquisição.

Pela Figura 4.27 pode-se ver a implementação do sistema

de transmissão dos pacotes provenientes da placa de aquisição. O bloco *SPI para Altera ST* é responsável por concatenar os pacotes de 16 bits e 32 bits para formarem pacotes de 64-bits, que são usados pela DMA para assim serem transmitidos para SDRAM. Os pacotes de 16-bits são os dados provenientes dos decimadores, sendo que os de 32 bits são os provenientes dos filtros média-móvel. Nessa Figura não se encontra representado a DMA, que é conectada na saída do bloco que transfere os pacotes de 64 bits para a memória SDRAM.

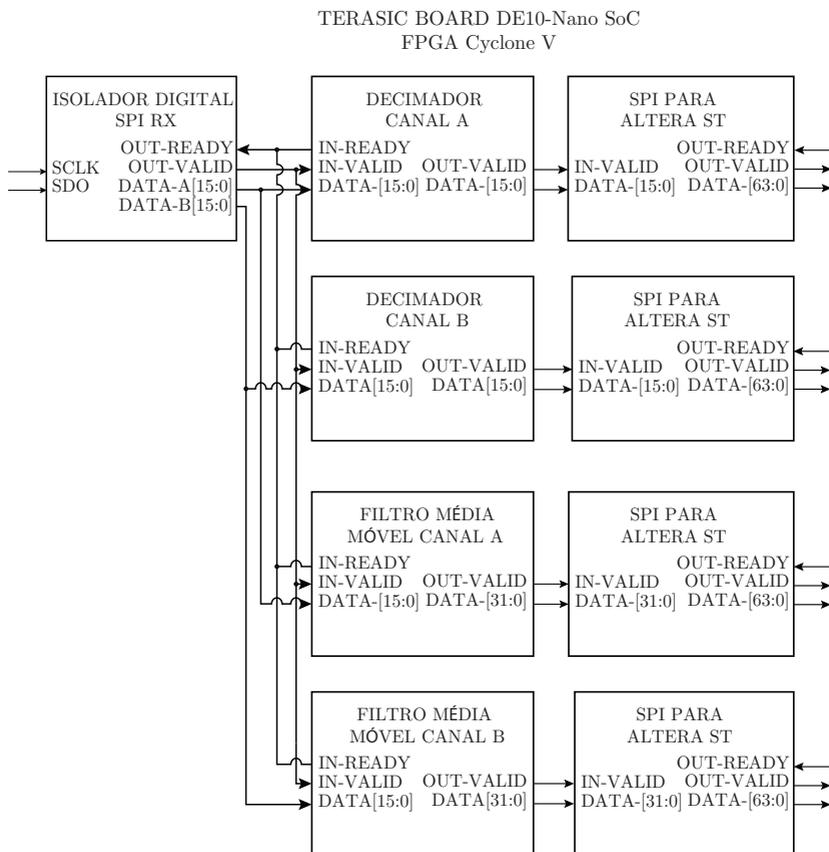


Figura 4.27 – Diagrama de blocos do sistema de recepção e transmissão de dados.

A implementação da DMA foi feita a partir das IPs disponibilizadas pela Intel FPGAs a fim de realizar a interface do hardware com a SDRAM de 1 GB da placa DE-10. Os blocos utilizados são *Write Master* e *Modular Scatter-Gather DMA Core* [32], que podem ser vistos na Figura 4.28.

O bloco *Modular Scatter-Gather DMA Core* é configurado através do processador ARM, que registra um descritor para o módulo, ajustando os limites inferior e superior para o bloco *Write Master* realizar a escrita na SDRAM, e também a configuração que reinicializa o valor do ponteiro para a SDRAM quando o mesmo atinge o valor máximo, implementando assim um buffer circular. Acerca do descritor e dos registradores do bloco *Modular Scatter-Gather DMA Core* refere-se a [32].

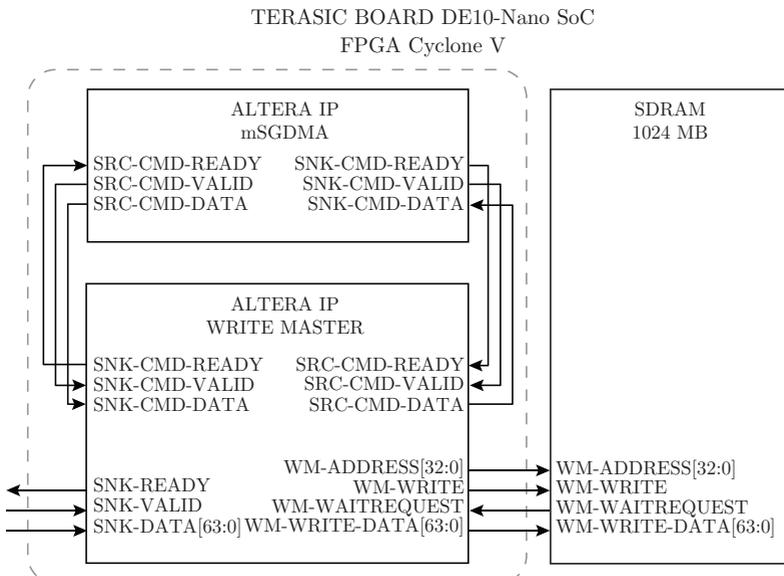


Figura 4.28 – Diagrama de blocos do hardware em FPGA que realiza a escrita dos pacotes na memória SDRAM externa.

4.2.2 Estimação de Frequência e Integradores

O algoritmo para a estimação de frequência, apresentado no Capítulo 2, é implementado no FPGA como dado pela Figura 4.29. Adicionalmente, a figura apresenta os blocos de integração de tensão, corrente e potência, tanto para valores médios quanto eficazes.

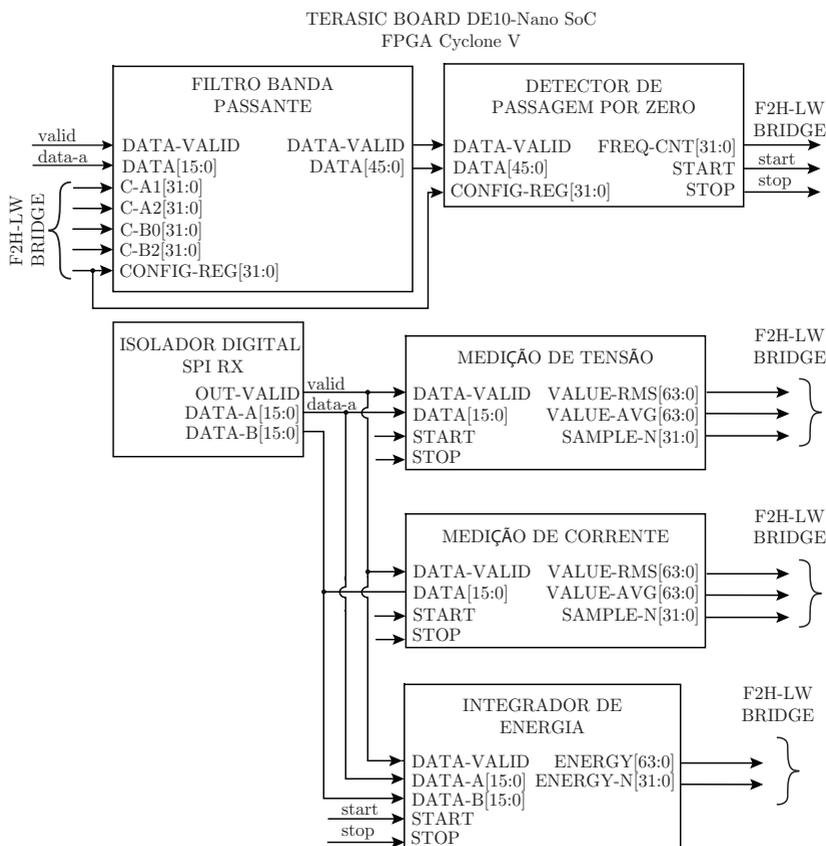


Figura 4.29 – Diagrama de blocos dos integradores e do sistema de estimação de frequência.

O resultado desses integradores é então enviado a registradores de 64 bits conectados ao barramento *F2H Lightweight Bridge*. Dessa forma o processador tem acesso a esses valores, podendo finalizar os cálculos em ponto flutuante a partir do valor médio ou eficaz e do número de amostras integradas, dado pelo sinal *SAMPLE-N*.

A atualização dos coeficientes do filtro banda passante é realizado através do processador, que atualiza os valores das entradas do bloco *Filtro Banda Passante* dado por *C-A1*, *C-A2*, *C-B0* e *C-B2*. A entrada denominada de *CONFIG-REG* permite que o processador habilite a atualização de coeficientes, a inicialização e a parada do filtro.

4.2.3 Circuito de *Trigger* por Nível

O diagrama de blocos do sistema de *trigger* proposto é dado pela Figura 4.30. Nota-se que o sistema permite a configuração do nível do *trigger* através de *TRIGGER_LEVEL* assim como o canal de detecção, dados pela corrente ou tensão e configurados pelo sinal *TRIGGER_CH_SEL*.

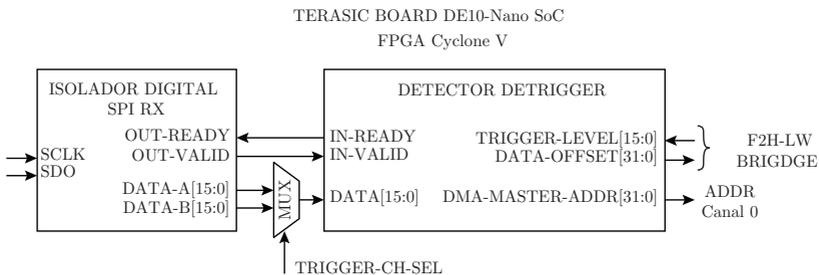


Figura 4.30 – Aquisição do *Trigger* de Nível.

A detecção de *trigger* pelo nível de um determinado sinal é representada pela Figura 4.31. Ao acontecer uma condição

de *trigger*, o hardware salva o endereço que a DMA usa para acessar a SDRAM, dado pelo sinal *WM-ADDRESS* presente na Figura 4.28. Quando o sinal cruza pelo nível de detecção com derivada negativa o sistema permite a busca de uma nova condição de *trigger*. Os endereços de memória correspondentes as posições de *trigger* são salvos pelo processador ARM, que realiza o processamento necessário para transferir os pacotes de dados para a interface gráfica. Adicionalmente, os endereços de memória salvos são sempre referentes ao Canal 0, de tensão, independentemente se o *trigger* é aplicado para o sinal de tensão - Canal 0 - ou corrente - Canal 1.

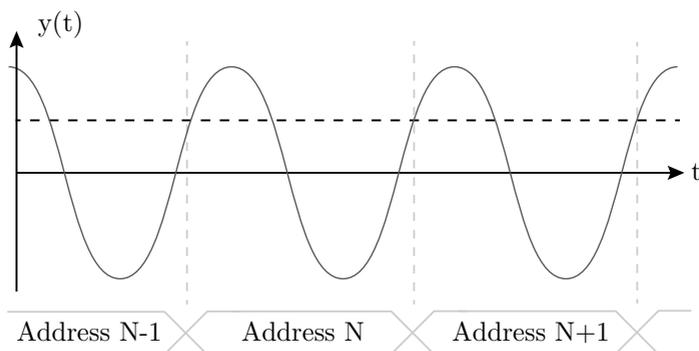


Figura 4.31 – Aquisição do Trigger de Nível.

4.2.4 Lista de Registradores

A interconexão entre o processador ARM A9 e o FPGA Cyclone V do circuito integrado SoC é feita através de barramentos que acessam periféricos mapeados na memória. Três principais interfaces de conexão permitem que o FPGA e o processador se comuniquem. Essas interfaces são *HPS-to-FPGA bridge*, *FPGA-to-HPS bridge* e *Lightweight HPS-to-FPGA bridge*. As primeiras duas interfaces de comunicação mencionadas são barramentos de alta performance, com largura de dados variáveis de 32, 64 ou 128 bits. Dessa forma o barramento possui visualização completa do mapa de endereços do FPGA/HPS. Já o bar-

ramento *Lightweight* possui largura de dados de 32 bits, menor performance que os outros barramentos e permite que o processador acesse registradores e periféricos mapeados na memória e que são implementados no FPGA. Pela Figura 4.32 pode-se ver a forma como as interfaces de comunicação realizam a conexão entre o HPS e o FPGA. Dessa forma, torna-se fácil instanciar registradores que configuram os periféricos no FPGA além de tornar dados de interesse disponíveis para o HPS.

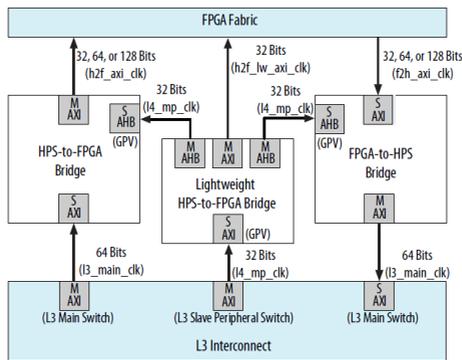


Figura 4.32 – Barramentos de interconexão entre o FPGA e o HPS. Retirado de [33].

Os registradores implementados em hardware estão listados nas Tabelas 4.6 e 4.7. Os registradores 1 até 4 são responsáveis por configurar os descritores das DMAs que são parte dos IPs da fabricante Intel FPGA. O registrador *trigger_offset_reg* salva o endereço da posição de memória em que se encontra a última ocorrência de *trigger* de nível, sendo que o registrador *trigger_memptr_reg* recebe o valor de cada posição escrita pela DMA. Como forma de selecionar os fatores de decimação N_{dec} pode-se fazê-lo através do registrador *data_decimation_reg*, já para o fator do filtro média-móvel N_{mm} faz-se através do registrador *fft_moving_avg_reg*. Ademais, *fft_memptr_reg* é responsável por atualizar a última posição de memória escrita na memó-

ria SDRAM. O registrador *sw_config_reg* atualiza os ganhos e os filtros do circuito da placa de aquisição, e os registradores de número 13 a 18 são responsáveis pela configuração do filtro digital de banda passante.

Na Tabela 4.7 tem-se os registradores que realizam a integração das variáveis elétricas de tensão, corrente e energia, além de registradores que calibram a resposta proveniente do conversor analógico-digital pela aproximação por uma função polinomial de segunda ordem. O algoritmo de calibração é realizado junto ao bloco de recepção dos dados pelo CPLD, sendo que o objetivo é melhorar a linearidade e corrigir erros de ganho e *offset* dos canais de tensão e corrente do ADC. A Seção 6.6 trata sobre a função utilizada além de disponibilizar os coeficientes resultantes da calibração.

Tabela 4.6 – Lista de registradores do *trigger*, decimação e configuração de ganhos, além do filtro de estimação de frequência acessados via barramento *Lightweight HPS2FPGA AXI Bridge*.

N	Nome	Tamanho	Endereço Inicial	Endereço Final
1	<code>dma_descriptor_a</code>	32	0x00040060	0x0004006f
2	<code>dma_descriptor_b</code>	32	0x00040070	0x0004007f
3	<code>dma_desc_fft_a</code>	32	0x00050060	0x0005006f
4	<code>dma_desc_fft_b</code>	32	0x00050070	0x0005007f
5	<code>trigger_config_reg</code>	32	0x000400c0	0x000400cf
6	<code>trigger_offset_reg</code>	32	0x00040090	0x0004009f
7	<code>trigger_memptr_reg</code>	32	0x000400a0	0x000400af
8	<code>trigger_response_reg</code>	16	0x00040080	0x0004008f
9	<code>data_decimation_reg</code>	16	0x000400b0	0x000400bf
10	<code>fft_moving_avg_reg</code>	16	0x000400d0	0x000400df
11	<code>fft_memptr_reg</code>	16	0x000400e0	0x000400ef
12	<code>sw_config_reg</code>	16	0x000400f0	0x000400ff
13	<code>coefficient_b0</code>	32	0x00050000	0x0005000f
14	<code>coefficient_b2</code>	32	0x00051000	0x0005100f
15	<code>coefficient_a1</code>	32	0x00052000	0x0005200f
16	<code>coefficient_a1</code>	32	0x00053000	0x0005300f
17	<code>fir_config_reg</code>	32	0x00054000	0x0005400f
18	<code>fir_response_reg</code>	32	0x00055000	0x0005500f

Tabela 4.7 – Lista de registradores das medições e calibrações dos conversores analógico-digital acessados via barramento *Lightweight HPS2FPGA AXI Bridge*.

N	Nome	Tamanho	Endereço Inicial	Endereço Final
19	num_samples_N	32	0x00060000	0x0006000f
20	measure_voltage_0	32	0x00061000	0x0006100f
21	measure_voltage_1	32	0x00062000	0x0006200f
22	measure_current_0	32	0x00063000	0x0006300f
23	measure_current_1	32	0x00064000	0x0006400f
24	measure_energy_0	32	0x00065000	0x0006500f
25	measure_energy_1	32	0x00066000	0x0006600f
26	measure_config_reg	32	0x00067000	0x0006700f
27	voltage_adc_cal_a0	32	0x00070000	0x0007000f
28	voltage_adc_cal_a1	32	0x00071000	0x0007100f
29	voltage_adc_cal_a2	32	0x00072000	0x0007200f
30	current_adc_cal_a0	32	0x00073000	0x0007300f
31	current_adc_cal_a1	32	0x00074000	0x0007400f
32	current_adc_cal_a2	32	0x00075000	0x0007500f

5 SOFTWARE

O presente capítulo tratará do software desenvolvido para o sistema embarcado, composto pela placa de desenvolvimento DE-10 Nano SoC, e para a interface de usuário, que é executada em um computador de uso geral. Dessa forma, será apresentado o diagrama de classes de cada parcela de software, assim como uma discussão sobre classes, métodos e processos que são executados concorrentemente. As linguagens de programação utilizadas para o desenvolvimento foram C++ para o sistema embarcado e Python para a interface do usuário.

Nas Figura 5.33 e 5.34 são apresentados diagramas de blocos que explicitam os principais processos executados na placa de desenvolvimento DE-10 Nano SoC e a interface do usuário.

Na Figura 5.33 vê-se a conexão do FPGA com o HPS a partir dos barramentos LW e F2H. Como comentado na Subseção 4.2.4, esses são barramentos de interconexão entre os dispositivos. Além disso, tem-se a presença dos blocos referentes ao cálculo da DFT, transmissão e recepção de dados via UDP, atualização dos registradores, cálculo das componentes de medição para o analisador de potência, configuração do sistema, dado pela configuração das DMAs, dos filtros média-móvel, decimadores, e do modo de plotagem, englobando os modos "Trigger de Nível", "Contínuo" e "Espectro".

5.0.1 Sistema Embarcado

A presente subseção tratará do software desenvolvido para o sistema embarcado DE-10 Nano SoC, empregando o sistema operacional *Embedded Linux* em um processador ARM Dual-Core A9. A distribuição do sistema operacional é o *Ångström*.

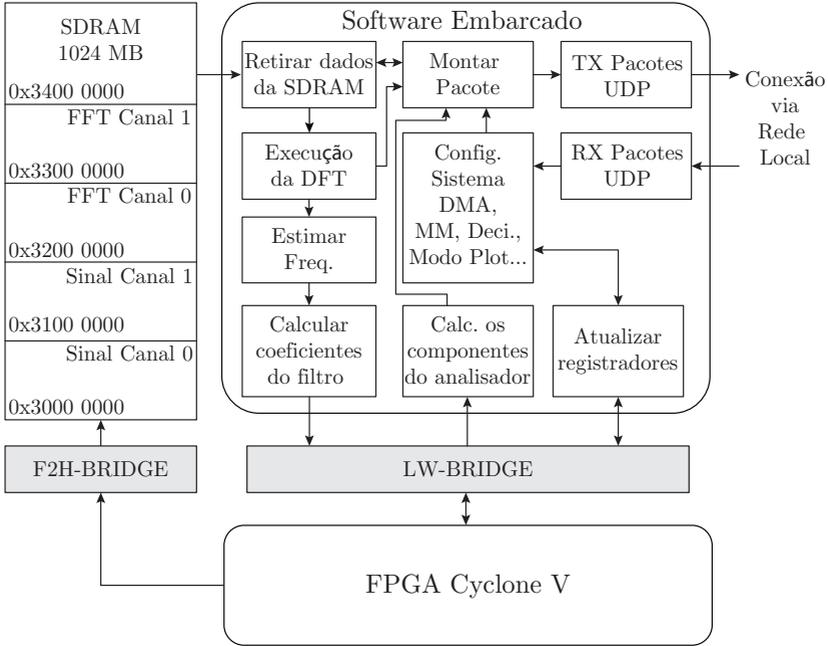


Figura 5.33 – Diagrama de blocos simplificado do software desenvolvido para a placa DE-10 Nano SoC.

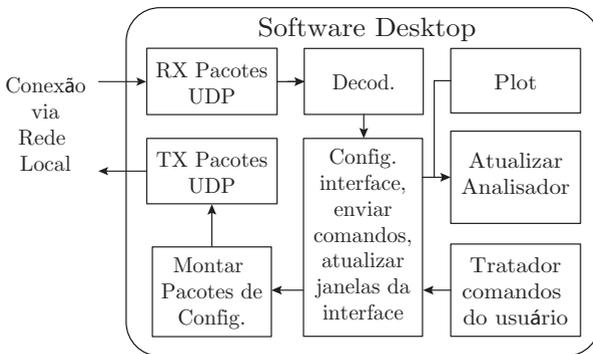


Figura 5.34 – Diagrama de blocos simplificado do software desenvolvido para o ambiente da interface de usuário.

Na Figura 5.33 pode-se ver o fluxo de processos que ocorrem no *software* da placa DE-10 Nano SoC. Alguns dos objetivos desse programa é a configuração de todos os periféricos presentes no FPGA, tratados no Capítulo 4. Esses são as DMAs, os filtros média-móvel, decimadores e os coeficientes do filtro banda passante para o algoritmo de estimação de frequência. Além disso, o programa tem por finalidade realizar o envio de pacotes de dados para a interface de usuário e atuar conforme pacotes de configuração provenientes da interface de usuário.

5.0.1.1 Padrão de Projeto para Máquina de Estados

Devido a necessidade do *software* atuar em diferentes estados, como a interface desconectada do sistema embarcado, a interface plotando dados no modo trigger de nível ou no modo contínuo, por exemplo, buscou-se uma solução que atendesse essa mudança de comportamento, que é inteiramente dependente da ação do usuário sobre a interface gráfica. Com isso, chegou-se no padrão de projeto de máquinas de estado em que o comportamento de um determinado objeto muda conforme a mudança no estado interno ao objeto [34].

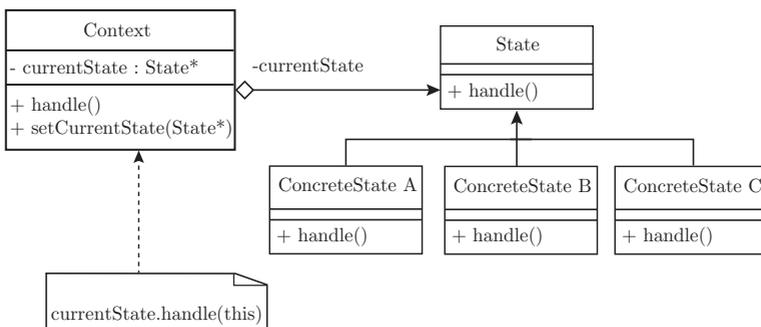


Figura 5.35 – Diagrama de classes para o padrão de projeto para máquinas de estado.

O diagrama de classes é dado pela Figura 5.35, exemplificando a mudança de comportamento do objeto do tipo *State*, que é base para as classes derivadas *ConcreteState A*, *ConcreteState B* e *ConcreteState C*. Com isso, tendo em vista que a classe *State* define todos os seus métodos como funções virtuais, ou seja, funções que esperam ser redefinidas nas classes derivadas, tem-se que ao apontar para um determinado estado concreto dado por *ConcreteState A*, *ConcreteState B* e *ConcreteState C*, as funções de interesse são redefinidas conforme o estado em que se aponta, executando assim implementações diferentes para o método de mesmo nome que é dependente do estado atual em que o sistema se encontra.

Na Figura 5.35 tem-se que a classe *Context* não implementa nenhum comportamento específico relacionado aos estados da máquina de estados, mas tem por função referenciar os métodos que serão executados e definir o estado atual da máquina de estados.

5.0.1.2 Diagrama de UML de Classes

A partir da explanação realizada acerca do padrão de projeto para máquina de estados, far-se-á uma análise acerca da implementação realizada para o projeto em questão. Pela Figura 5.36 tem-se o diagrama UML de classes que exemplifica a implementação da relação entre as classes do padrão de máquina de estados. Pode-se ver que a classe *State* implementa todos seus métodos como funções virtuais, e as suas implementações são redefinidas nos estados *ClientDisconnected*, *ClientConnected*, *ContinuousAcquisition* e *TriggeredAcquisition*.

O comportamento da máquina de estados ao inicializar o software é de se manter no estado *ClientDisconnected* até o cliente se conectar ao sistema através do método *bind_to_client()*. Após a conexão, a máquina de estados muda para o estado *Trig-*

geredAcquisition, que redefine os métodos *retrieve_ram_data()* e *set_acquisition_state()*. Caso o cliente escolha o modo de visualização contínuo através da interface, então o software da interface comunica com o sistema embarcado, que modifica o estado atual para *ContinuousAcquisition*, redefinindo esses últimos dois métodos. Interessantemente, esses métodos compartilham de uma mesma classe base, denominada *ClientConnected*, que redefine os métodos *bind_to_client()* e *send_data_over_network()*.

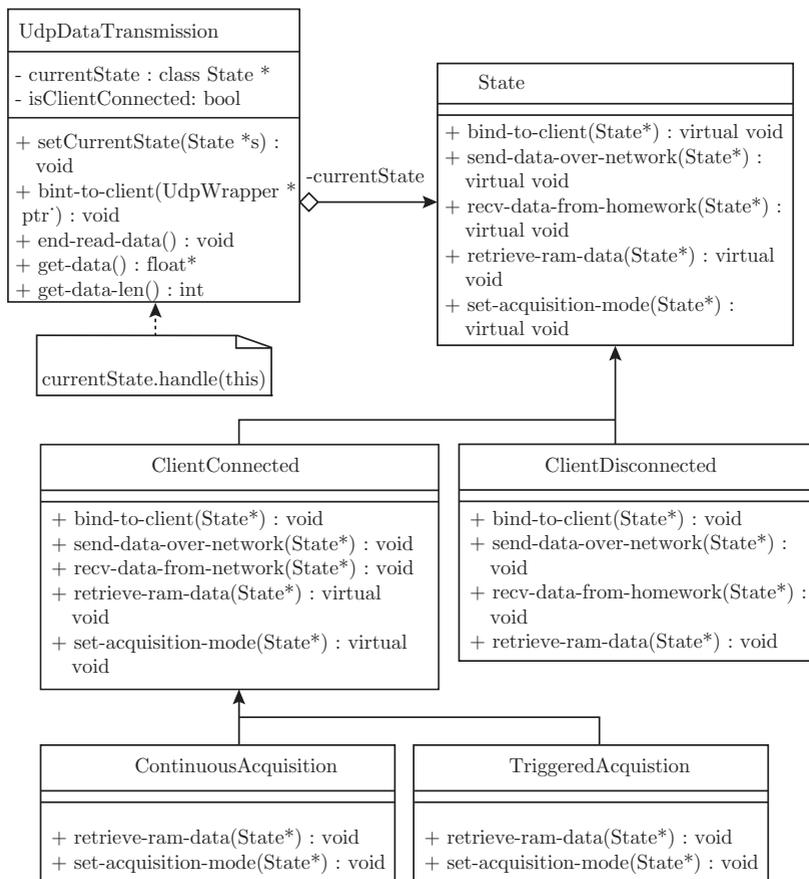


Figura 5.36 – Diagrama da máquina de estados implementada.

A fim de compreender o desenvolvimento dessa proposta em termos de código, são apresentados a declaração e implementação da classe *UdpDataTransmission*, assim como a declaração da classe *State* nos Apêndices A.1, A.2 e A.3. Através dos Apêndices pode-se ver que a classe *UdpDataTransmission* mantém a referência para as classes derivadas da classe base *State*, sendo possível entender como é feita a chamada dos métodos das classes concretas que redefinem a implementação dos métodos dados pela classe *State*.

Na Figura 5.37 pode-se ver o diagrama das classe que implementam os métodos de comunicação via protocolo UDP (*User Datagram Protocol*). A classe *UdpWrapper* realiza o envio dos pacotes para a interface do usuário, atualizando os valores do analisador de potência e também selecionando os dados para o caso do modo de visualização, se "Contínuo", "Trigger de Nível" ou "Espectro". Com a finalidade de explicitar a transferência dos pacotes em termos de código, tem-se no Apêndice A.4 a implementação do método *SendSop*, que inicia a transferência dos pacotes para a interface gráfica. Com isso vê-se que o pacote envia informações da análise de potência e o dado que será enviado, que pode ser tanto dados referente ao sinal monitorado quanto dados do espectro calculado através da DFT.

O código que define o envio dos dados referente aos sinais no domínio do tempo é dado pelo código do Apêndice A.5, que define o método *SendSignalPacket* da classe *UdpWrapper*. As variáveis passadas para o método são o ponteiro para memória SDRAM, dado por *const void* memory_ptr*, o offset do canal a ser enviado dado por *unsigned int channel_offset_space* além dos limites inferior e superior do pacote a ser enviado (como forma de endereço), dados por *unsigned int higher_boundary* e *unsigned int lower_boundary*.

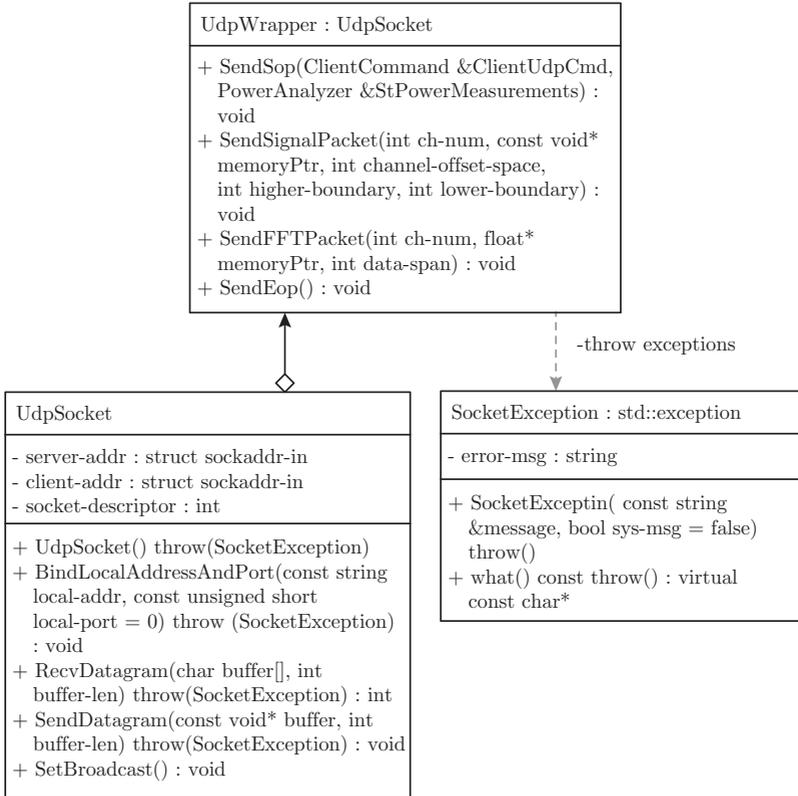


Figura 5.37 – Diagrama de classes para a comunicação UDP.

A parcela do código que realiza o envio do dados referentes ao espectro do sinal é dado abaixo pelo método *SendFFTPacket*, apresentado no Apêndice A.6.

Adicionalmente, tem-se no Apêndice A.7 os códigos que definem os métodos *SendDatagram*, *RecvDatagram* e *SendEop*. Dessa forma, pode-se ver como o software do sistema embarcado envia informações e como os pacotes são organizados quando recebidos pela interface gráfica.

A Figura 5.38 traz a relação entre as classes *fftHandler* e *FftDataStorage*, que tem por função executar o algoritmo da DFT, além de realizar a estimação da frequência, configurar a

biblioteca da DFT e realizar a transmissão dos pacotes referentes ao espectro para o cliente.

A classe *fftHandler* tem por função configurar o plano de execução da DFT através das APIs da biblioteca FFTW, além de extrair informações sobre a frequência fundamental do espectro resultante da DFT. Ademais, a classe *FftDataStorage* é derivada da classe *RdWrSync*, que sincroniza escritas e leituras entre duas threads. No caso, uma thread é bloqueada de ler uma variável enquanto outra thread executa alguma modificação nessa variável. A permissão de acesso das duas threads na mesma variável se dá somente quando há apenas leituras de variáveis. Com isso, o objeto do tipo *FftDataStorage*, que é compartilhado entre duas threads, acessa as informações de forma sincronizada. Ademais, o carregamento e a retirada dos dados resultantes do espectro calculado pela FFTW é feito pelos métodos *load_data()* e *get_data()*.

Alguns métodos foram explicitados no Apêndice A.8. Estes são importantes para exemplificar a utilização da biblioteca FFTW. A partir de *create_plan()* cria-se um plano para a execução da DFT, levando em conta a utilização do plano em que os dados de entrada são reais. O método *set_sample_size()* ajusta o tamanho das variáveis dinamicamente alocadas dados por *in* e *out*, e o método *execute_plan()* realiza o cálculo da DFT para um vetor de números reais. Através do manual da FFTW [22] tem-se acesso as funções da biblioteca, permitindo a execução de DFTs com conjunto de dados de entrada complexos ou reais.

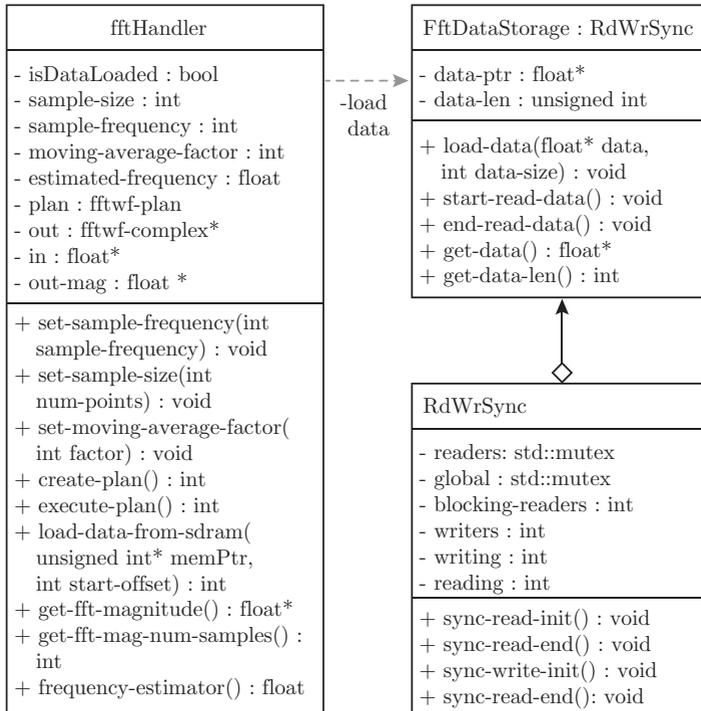


Figura 5.38 – Diagrama de classes para o cálculo e transferência da DFT.

As Figuras 5.39 e 5.40 apresentam as funções utilizadas para configurar os registradores dispostos nas Tabelas 4.6 e 4.7. Através da classe *mSGDMA_dispatcher* configuram-se as DMAs que enviam os dados dos canais de tensão e corrente, assim como os canais do dado para o cálculo da FFT para o canal de tensão e também para o de corrente. A classe *analogSwitches* registra os ganhos do circuito analógico de tensão e corrente, assim como os filtros analógico da placa de aquisição para ambos canais de aquisição.

Em relação a configuração dos registradores referente aos coeficientes do filtro banda passante, a classe *bandpassFIR* tanto atualiza os registradores quanto realiza os cálculos dos coefi-

entes a partir da frequência de amostragem, da banda passante e da frequência de ressonância do filtro. Também, essa classe tem acesso ao resultado proveniente da detecção de passagem por zero, que é obtido através do método *get_fft_estimation()*.

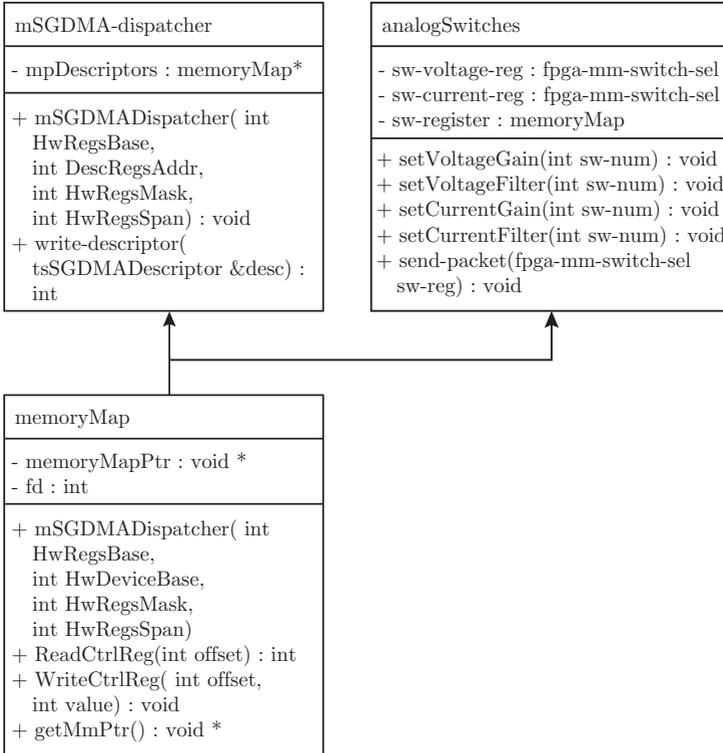


Figura 5.39 – Diagrama de classes de interface dos periféricos do FPGA.

Através da classe *electricalMeasurements*, tem-se o acesso aos valores das medidas dos integradores de corrente, tensão e energia implementados no FPGA. Caso o usuário deseje analisar valores médios ou valores eficazes, o sistema selecionará o integrador correto para disponibilizar os valores. Devido a representação do FPGA ser em ponto fixo, realiza-se a raiz quadrada

dos cálculos e a divisão pelo número de amostras no processador, que possui representação em ponto flutuante.

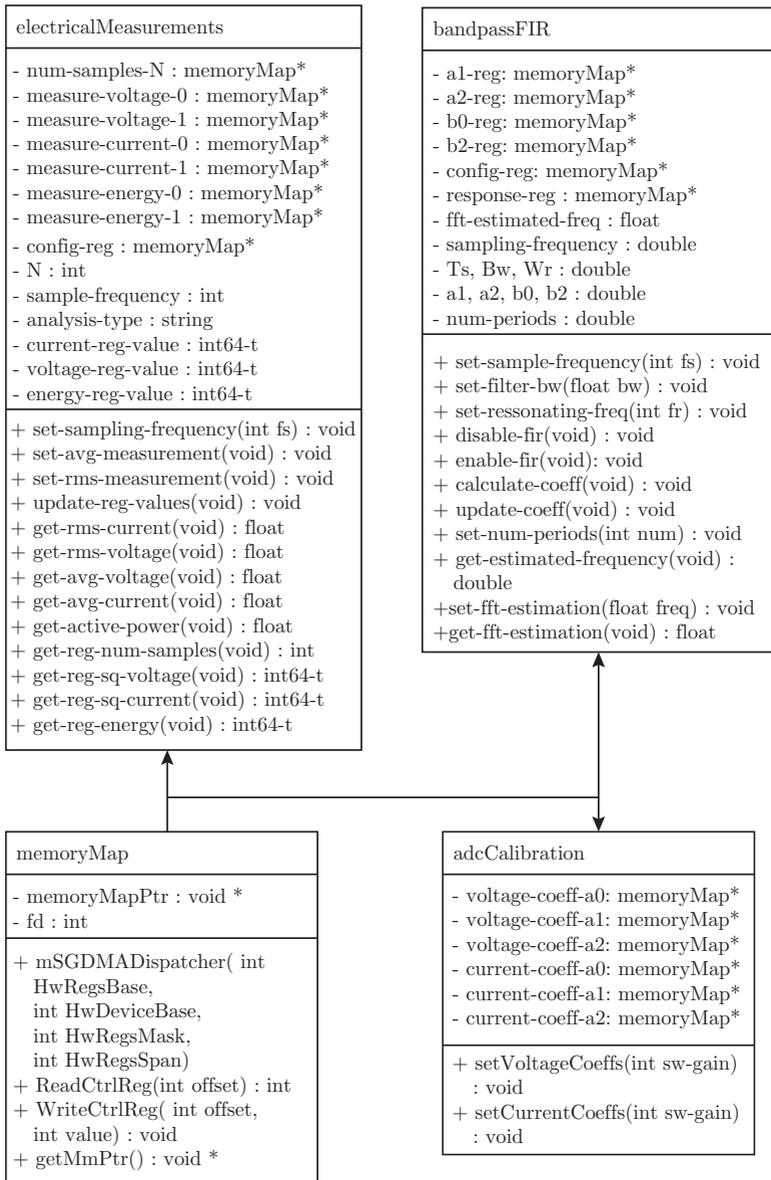


Figura 5.40 – Diagrama de classes do cálculo de potência e do ajustador do filtro banda passante.

A Figura 5.41 tem por finalidade ilustrar a utilização das classes ao longo das threads executadas no algoritmo. A thread principal, chamada de *Main*, lança as outras duas threads, nomeadas de *Network* e *Power Analysis*. A thread *Network* visa realizar o envio dos pacotes de dados de monitoramento armazenados na SDRAM e também provenientes do objeto do tipo *FftDataStorage* através do objeto do tipo *UdpWrapper*. Também, esse mesmo objeto recebe dados de configuração de ganho e de seleção dos filtros do circuito analógico, sendo então comandados pelo objeto do tipo *analogSwitch*. *UdpWrapper* envia informações referente a análise de potência, provenientes de estruturas compartilhadas entre as threads, que é atualizada pelo objeto da classe *electricalMeasurements*.

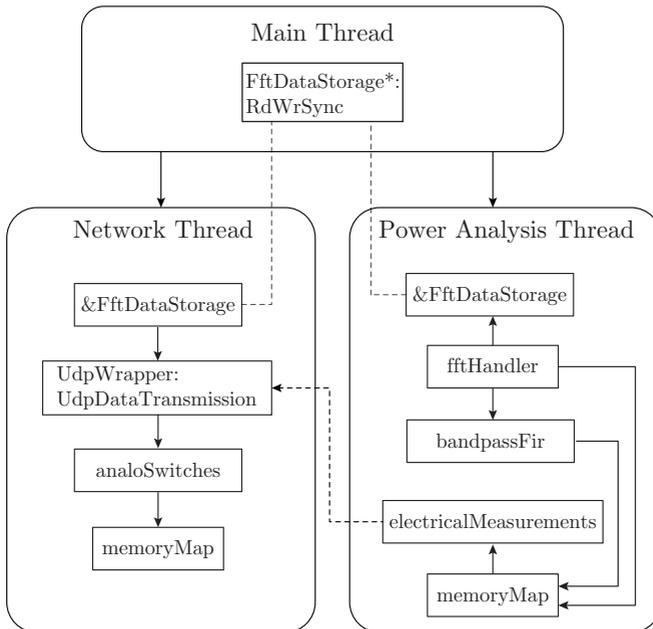


Figura 5.41 – Utilização das classes nas diferentes threads do programa.

Em relação a thread *Power Analysis*, ela é responsável por executar a DFT e, por consequência, estimar a frequência fundamental do sinal, atualizar os coeficientes do filtro com a nova frequência estimada, salvar os dados da magnitude do espectro em frequência resultantes da DFT no objeto da classe *FftDataStorage* além de ler e finalizar os cálculos das componentes elétricas de corrente, tensão, potência e frequência provenientes dos integradores presentes no FPGA.

5.1 INTERFACE DE USUÁRIO

Para a implementação da interface de usuário escolheu-se trabalhar com a biblioteca PyQt, que é uma aplicação da *framework* Qt na linguagem Python. Essa biblioteca é, em linhas gerais, uma *framework* multiplataforma para o desenvolvimento de interfaces gráficas. Além, a ferramenta Qt Designer possibilitou construir a interface gráfica através do posicionamento dos objetos gráficos e elementos provenientes da biblioteca PyQt. Ademais, utilizou-se a biblioteca *open-source* PyQtGraph, que permite a implementação de visualização gráfica de sistemas de monitoramento em tempo-real.

A Figura 5.34 apresentou um diagrama simplificado do software que implementa a interface gráfica, além do processamento que ocorre por trás dela. Nessa parte do projeto pretende-se processar os pacotes recepcionados a partir de uma conexão de rede local através do protocolo UDP, decodificar esses pacotes e realizar a plotagem dos dados adquiridos, disponibilizar para o usuário os valores referente a análise de potência, permitir que usuário controle ações como o modo visualização, se em "Trigger de Nível" ou "Contínuo", dentre outras que serão comentadas ao longo do presente Capítulo.

5.1.1 Diagrama de Classes

Através da Figura 5.42 tem-se o diagrama de classes do sistema desenvolvido. Acerca dessas, a classe *Fifo* visa implementar uma FIFO (*First In, First Out*) para os pacotes recebidos via rede local. O conteúdo do pacote é formatado em palavras hexadecimais de 16 bits, sendo que uma palavra se encontra concatenada as outras sem espaçamento ou algum limitador de palavra, como vírgulas, por exemplo. Essas palavras, recebidas via *socket* UDP, são os dados amostrados de tensão e corrente, sendo que cada pacote recebido possui dados apenas referente ao canal de tensão ou de corrente. Dessa forma, a composição dos dados do pacote são os próprios sinais plotados na interface, que podem ser dados provenientes do modo "Contínuo", "Trigger de Nível" ou "Espectro". Os dados do pacote são separados pelo objeto do tipo *EncodeToDecimal*, que realiza a conversão dos dados de 16 bits que estão concatenados para listas em que cada elemento dessa é um dado do tipo inteiro.

A classe *UdpClientWrapper* é responsável pela comunicação com o sistema embarcado composto pelo SoC, no qual a classe base *UdpClient* implementa um *socket* UDP, que é usado para o envio e recebimento de pacotes de dados. A classe *UdpClient* possui métodos que instanciam, deletam, recebem e enviam dados através do *socket* UDP. *UdpClientWrapper* possui um método de construção de uma mensagem formatada que tem como intuito atualizar as variáveis internas do sistema embarcado, composto pela placa DE-10 Nano SoC, e a placa de aquisição. Essas variáveis, por exemplo, atualizam os ganhos dos circuitos analógicos, o modo de visualização da interface e o valor de decimação dos dados de entrada.

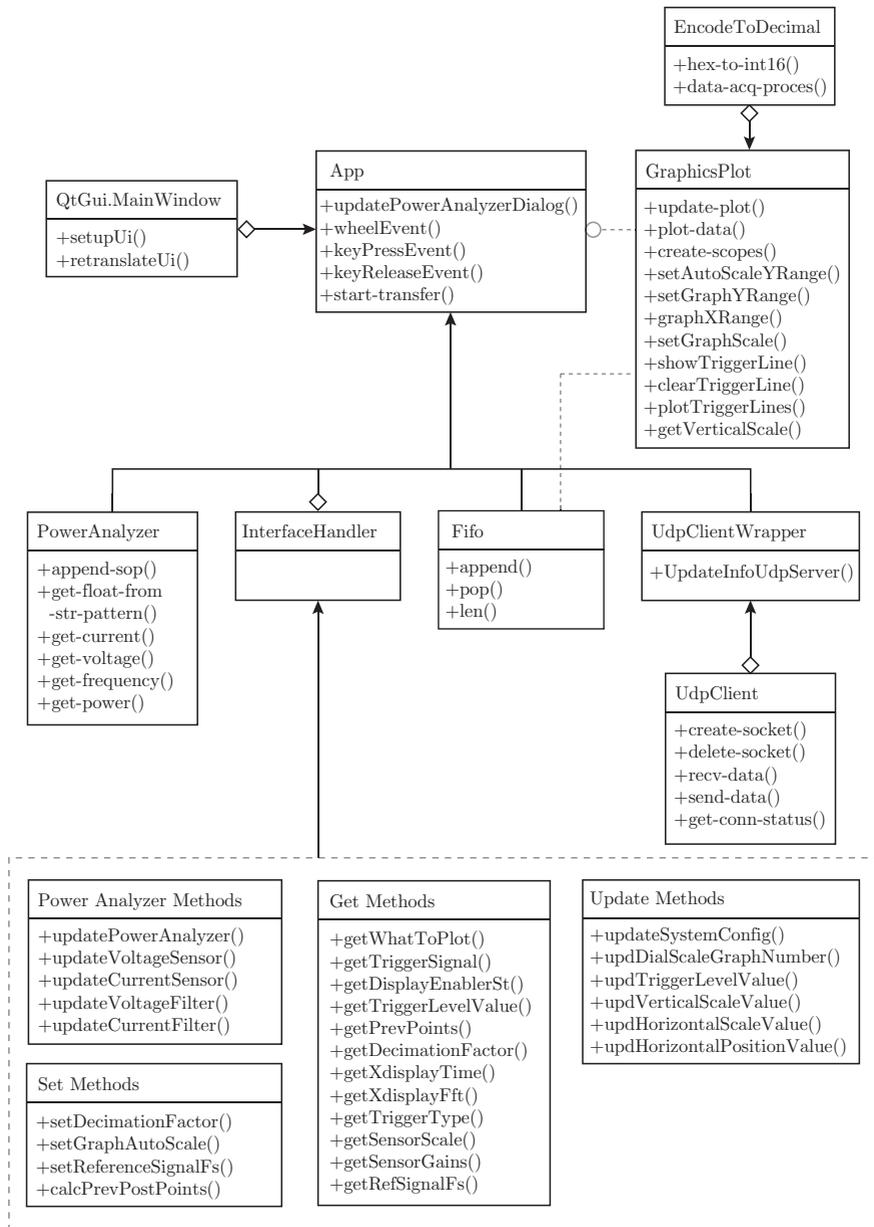


Figura 5.42 – Diagrama de classes da interface de usuário.

O objeto do tipo *PowerAnalyzer* realiza a atualização dos valores de potência mediante recebimento de pacotes via rede local, atribuindo aos membros privados da classe os valores atualizados de potência, tensão, corrente e frequência. Caso o usuário selecione análise "CA" os valores dados são eficazes. Caso "CC", os valores são as componentes médias.

A classe *InterfaceHandler* possui métodos referentes a atualização dos valores do analisador de potência. Métodos que modificam configurações como o fator de decimação dos dados de entrada, atualizado pelo método *setDecimationFactor()*. Diversos outros métodos, que podem ser vistos pela Figura 5.42, atualizam a interface mediante comando do usuário, como a modificação do nível do trigger, a variação do eixo temporal, a mudança de escala de tensão e corrente, e assim por diante.

A responsabilidade de plotar o sinal fica por conta do objeto do tipo *GraphicsPlot*, que se utiliza da biblioteca PyQt-Graph a fim de implementar métodos que atualizam os containers delimitados para a disponibilização dos gráficos para ambos canais do sistema. Como dado pela Figura 5.42, os métodos subdividem-se entre os que atualizam a interface, como os métodos *update_plot()* e *plot_data()*, além do método *create_scopes()*, que permite a criação de múltiplos *scopes* de medição para cada gráfico. No caso de um sistema trifásico, esse método permite plotar três componentes no mesmo gráfico, portanto permitindo a modularidade do sistema e a escalabilidade perante modificações pequenas de código. Outros métodos atualizam as escalas, os eixos X e Y, além de disponibilizar, ou não, a linha referente ao trigger de nível.

O código que implementa o ambiente gráfico fica por conta de *QtGui.MainWindow*. Essa classe é gerada pelo programa Qt Designer, utilizado para o desenvolvimento do ambiente gráfico. Portanto, os membros dessa classe são conectados e utilizados pelas outras classes do software a fim de tratar ações do usuário sobre a interface. Por exemplo, em um determinado botão que

exista na interface, um método conecta a ação de clicar no botão a um outro método registrado pelo objeto do tipo *InterfaceHandler*, que executa uma outra parcela de código a fim de tratar a ação do usuário. Para exemplificar como isso ocorre em termos de código, abaixo encontra-se uma parcela de código escrito em Python, dado pelo Algoritmo 5.1, que tem por função conectar os seletores de ganho de tensão de 1000 V, 400 V e 100 V no método *updateVoltageSensor()* da classe *InterfaceHandler*. O método *connect()* da biblioteca PyQt permite usar uma expressão *lambda*, passando a variável *who*, que tem valor determinado conforme o botão selecionado, para o método *updateVoltageSensor()*.

```

1 # Voltage Selector – DockWidget Connections
2 self .connect( self .sensor_voltage_gain_1000V,
3               QtCore.SIGNAL("clicked("),
4               lambda who="0":
5                   self.updateVoltageSensor(who))
6 self .connect( self .sensor_voltage_gain_400V,
7               QtCore.SIGNAL("clicked("),
8               lambda who="1":
9                   self.updateVoltageSensor(who))
10 self .connect( self .sensor_voltage_gain_100V,
11               QtCore.SIGNAL("clicked("),
12               lambda who="2":
13                   self.updateVoltageSensor(who))

```

Algoritmo 5.1 – Conectar ações do usuário com métodos e funções.

Como pode ser visto através do diagrama de classes da Figura 5.42, a classe *App* é uma classe derivada, que herda os métodos e atributos das classes base indicadas pelas flechas com um losango. Ademais, o objeto do tipo *App* emite um objeto do tipo *GraphicsPlot* como uma thread, permitindo que o objeto execute seus métodos de maneira concorrente com outros pro-

cessos. O objeto do tipo *GraphicsPlot* recebe como parte de seu construtor a referência do objeto do tipo *Fifo*, que é membro do objeto da classe *App*.

5.1.2 Componentes da Interface

Com a finalidade de apresentar algumas componentes desenvolvidas para a interface, tem-se as Figuras 5.45, 5.44 e 5.43. A partir dessas janelas torna-se possível configurar os ganhos e os filtros passa-baixas do circuito analógico, visualizar o espectro de magnitude ou o sinal no domínio do tempo, o modo de visualização "Trigger de Nível" ou "Contínuo", assim como controle sobre os eixos horizontais e verticais do gráfico.

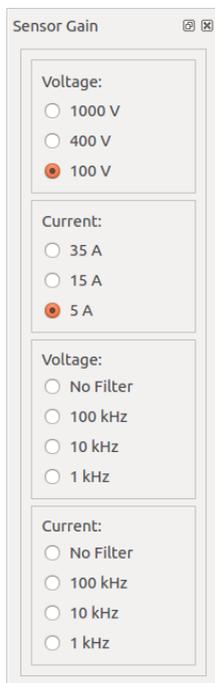


Figura 5.43 – Janela de seleção dos ganhos de sensor do circuito de aquisição e a seleção dos filtros passa-baixas.

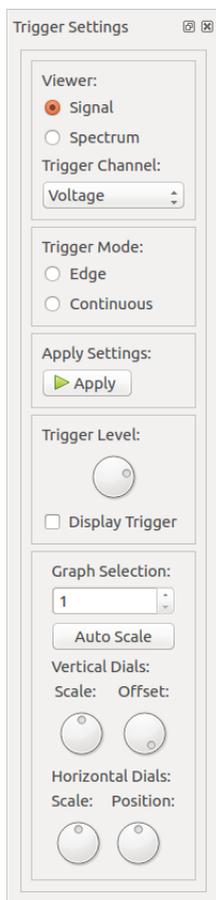


Figura 5.44 – Janela de configuração do trigger, de seleção do tipo de visualização, modificação do nível do trigger além do controle sobre os eixos horizontais e verticais do gráfico.

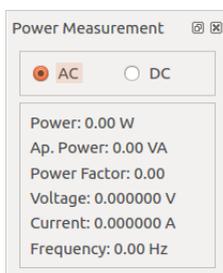


Figura 5.45 – Janela de visualização das medições das componentes do analisador de potência.

6 RESULTADOS

O presente capítulo tratará do protótipo desenvolvido, dos resultados obtidos com o sistema de aquisição, fonte de alimentação, interface de usuário e também dos resultados de medição do analisador de potência, tanto para o algoritmo de estimação de frequência quanto para os valores de corrente, tensão e potência.

6.1 CIRCUITOS INTEGRADOS DA PLACA DE AQUISIÇÃO

Os circuitos integrados escolhidos para compor a placa de aquisição foram o ADS8350 [25], que é um conversor analógico-digital de 16 bits e 750 KSPS (mil amostras por segundo), o OPA2350 [35], utilizado como driver para o ADC, sendo esse um amplificador *single-supply* com saída *rail-to-rail*. O OPA2350 é utilizado nos amplificadores IC_4 , IC_5 , IC_8 e IC_9 dos esquemáticos das Figuras 3.8 e 3.12.

As chaves analógicas, utilizadas para chavear os ganhos dos circuitos além dos filtros passa-baixas dos circuitos de aquisição, são aplicadas através do circuito integrado DG412LEDQ [24], possuindo quatro chaves analógicas SPST em um único encapsulamento.

Para o circuito de aquisição de corrente utilizou-se o resistor *shunt* Y14880R00500F9R [36], com precisão de 1,0 %, capacidade de potência de 3 W, coeficiente de variação da resistência em função da temperatura de 15 ppm (partes por milhão) e encapsulamento SMD. Ademais, escolheu-se o amplificador operacional OPA192 [37] para os amplificadores diferencial e de ganho programável, representados pelos IC_6 e IC_7 da Figura 3.12. O OPA192 é um amplificador de alta precisão, *dual-supply* e *rail-to-rail*.

Acerca do circuito de aquisição de tensão, utilizou-se o OPA4192, que é a versão do OPA192 em um encapsulamento com quatro amplificadores operacionais. Os resistores utilizados para aplicar o divisor de tensão possuem precisão de 0.1 %, tensão máxima sobre o dispositivo de 200 V e coeficiente de variação da resistência em função da temperatura de 15 ppm.

O CPLD aplicado é o MAX V 5M240ZT144C5N [31], que possui 240 elementos lógicos. Ademais, o isolador digital é o ISO7742 [38], que possui quatro canais de comunicação, sendo dois desses para a transmissão e dois para a recepção de dados. As fibras óticas são a HFBR-2528Z para a recepção de dados e a HFBR-1528Z para a transmissão de dados [39]. Outros componentes como reguladores de tensão, circuitos de *driver* para as fibras óticas podem ser vistos através dos esquemáticos do Apêndice C.

6.2 PROTÓTIPO

O protótipo construído no projeto pode ser visto através das Figuras 6.46 e 6.47. Pela segunda figura tem-se a conexão da placa



Figura 6.46 – Vista do protótipo da placa de aquisição.

No Apêndice C pode-se ver o esquemático dos circuitos que compõem a placa de aquisição, e o *layout* desenvolvido no Apêndice D.

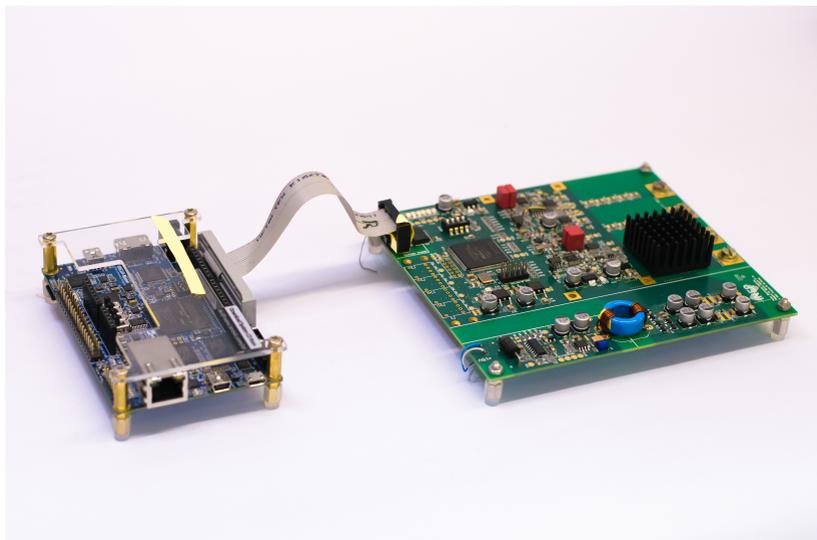


Figura 6.47 – Conexão do protótipo com a placa DE-10 Nano SoC desenvolvido pela empresa Terasic.

6.3 FONTE DE ALIMENTAÇÃO

Na Figura 6.48 vê-se a inicialização da fonte de alimentação. O tempo de subida com a atuação do *soft-start* fica próximo de 10 ms, não apresentando sobretensão. O valor médio das tensões positiva e negativa ficou em torno de 7,3 V. A alimentação externa utilizada como entrada da fonte projetada entrega 0,075 A para o protótipo com a tensão de 15 V. Dessa forma, o consumo total da placa ficou em torno de 1,11 W.

A fim de verificar a ocorrência de comutação suave mediu-se a tensão aplicada no tanque ressonante, dado por V_{hb} , como explicitado na Figura 3.19. Também é apresentada a tensão de gate de um dos *Mosfets*.

Na Figura 6.50 é possível visualizar que o pulso de acionamento do interruptor S_2 da meia ponte torna-se positivo após a tensão aplicada sobre o interruptor anular-se, configurando assim a ocorrência de comutação suave.



Figura 6.48 – Atuação do soft-start durante a inicialização da fonte. Sinal verde: tensão de saída positiva. Sinal amarelo: tensão de saída negativa.



Figura 6.49 – Entrada em condução da chave S_2 e a tensão V_{hb} de entrada do tanque ressonante. Sinal amarelo: tensão de entrada do tanque ressonante V_{hb} . Sinal verde: pulso de comando da chave inferior S_2 .



Figura 6.50 – Entrada em condução da chave S_2 com tensão nula sobre os terminais. Sinal amarelo: tensão de entrada do tanque ressonante V_{hb} . Sinal verde: pulso de comando da chave inferior S_2 .

6.4 CIRCUITO DE AQUISIÇÃO

Realizou-se a análise de resposta em frequência dos circuitos analógicos a partir do osciloscópio DSO-X 4054A, da empresa Keysight Technologies [40]. Esse osciloscópio possui uma função denominada de FRA (*Frequency Response Analysis*), em que o mesmo realiza a excursão em frequência de uma forma de onda senoidal proveniente de um gerador de funções interno ao osciloscópio. A partir das medições das tensões de entrada e saída dos circuitos de aquisição, o osciloscópio realiza o traçado da resposta em frequência.

Na Figura 6.51 pode-se observar a resposta em frequência do sistema analógico de aquisição de tensão. A frequência de corte do sistema não varia em função do ganho devido a realização dessa através de divisores resistivos. Com isso, a frequência de corte é a mesma para todas as escalas e possui o valor de 403,90 kHz. Na Figura 6.52 pode-se ver a validação dos filtros projetados, que correspondem as frequências de corte determinadas no projeto.

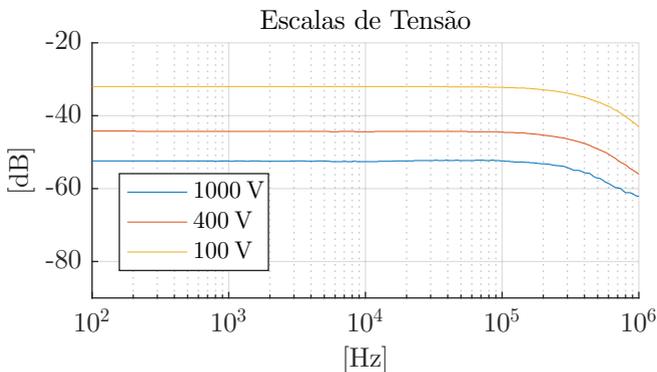


Figura 6.51 – Diagrama de bode para diferentes escalas de tensão sem filtro passa-baixas.

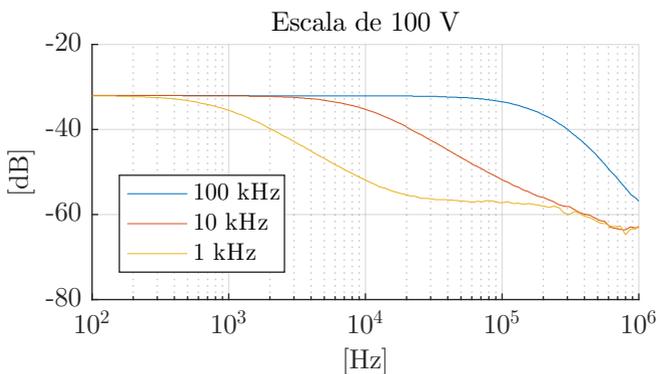


Figura 6.52 – Diagrama de bode para escala de tensão de 100 V com diferentes frequências de corte do fitro passa-baixas.

Para a escala de corrente, pela Figura 6.53, pode-se ver a variação de banda conforme o ganho do circuito. Tem-se que para a configuração de 35 A, correspondendo ao menor ganho do circuito analógico, resulta em uma maior banda, próximo de 2 MHz. Já para o ganho de 15 A a banda fica em 1,502 MHz.

Para 5 A a banda é de 380,4 kHz. Pelas Figuras 6.54 e 6.55 pode-se ver os diagramas de Bode das escala de corrente de 5 A e 35 A, respectivamente, com as variações dos os filtros passa-baixas.

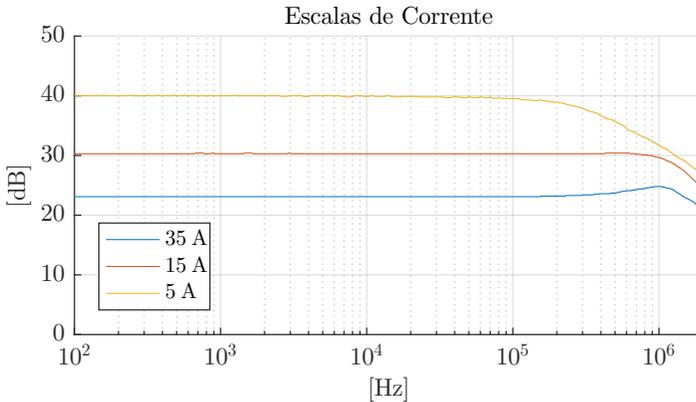


Figura 6.53 – Diagrama de bode para diferentes escalas de corrente sem filtro passa-baixas.

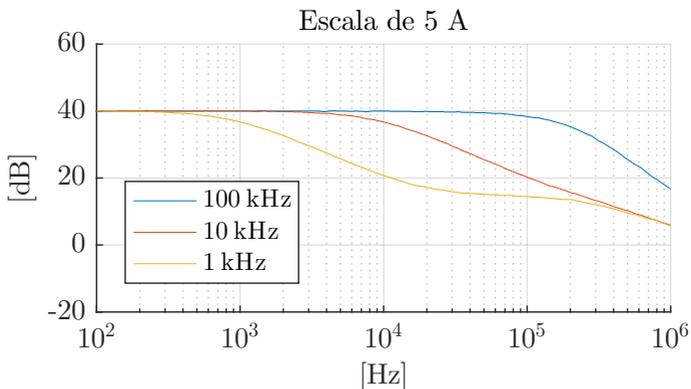


Figura 6.54 – Diagrama de bode para escala de corrente de 5 A com diferentes frequências de corte do fitro passa-baixas.

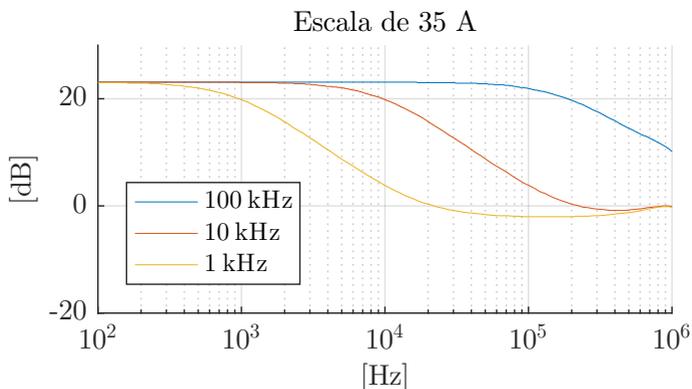


Figura 6.55 – Diagrama de bode para escala de corrente de 35 A com diferentes frequências de corte do fitro passa-baixas.

A Tabela 6.8 apresenta a relação de ganho dos circuitos analógicos para as escalas dos canais de corrente e tensão. Os ganhos foram obtidos a partir dos diagramas de resposta em frequência. Deve-se atentar ao fato de que o sinal proveniente do gerador de sinais do osciloscópio tem valor máximo de $10 V_{pp}$, ou seja, o sinal de medição na entrada do ADC assume valores próximos a 10 mV para a escala de 1000 V. Com isso, percebe-se que o ruído de fundo das medições interfere nos sinais de alta frequência.

Tabela 6.8 – Validação dos ganhos dos circuitos analógicos.

Escala	Ganho	
	Teórico (dB)	Medido (dB)
1000 V	-52,041	-52,4
400 V	-44,082	-44,3
100 V	-32,041	-32
35 A	23,098	23,1
15 A	30,457	30,3
5 A	40,000	40

6.5 ESTIMAÇÃO DE FREQUÊNCIA

Devido à representação numérica do FPGA Cyclone V ser em ponto fixo, utilizou-se da representação numérica Q , que é uma estratégia utilizada em hardwares que não possuem suporte para ponto flutuante. A representação é discutida em [41], sendo que a sua definição é dada pela relação $Q_{[I].[F]}$, em que $[I]$ é a parcela de bits com representação dos inteiros e $[F]$ a parcela fracionária. O filtro implementado usa a representação $Q_{1.15}$ para o sinal de entrada e $Q_{2.44}$ para o sinal de saída, sendo que os coeficientes do filtro são representados em $Q_{1.31}$.

As Figuras 6.56 e 6.57 mostram o resultado obtido do sistema proposto para a estimação de frequência. O teste fora realizado a partir da injeção de sinais no circuito de aquisição de tensão do protótipo. A Figura 6.56 apresenta a injeção de uma forma de onda senoidal, explicitada pelo gráfico superior da interface. A saída do filtro, recuperando a frequência fundamental do sinal, é apresentado pelo gráfico inferior.

A Figura 6.57 apresenta a estimação para uma onda quadrada, sendo possível visualizar a recuperação do sinal de componente fundamental, de 60 Hz. A medição pode ser vista no canto superior direito da interface, que mostra a frequência mensurada de 60 Hz.

Com a validação da funcionalidade do filtro, realizou-se testes que verificaram a estimação de frequência para os valores dispostos na Tabela 6.9. O multímetro utilizado para comparação foi o Tektronix DMM4040.

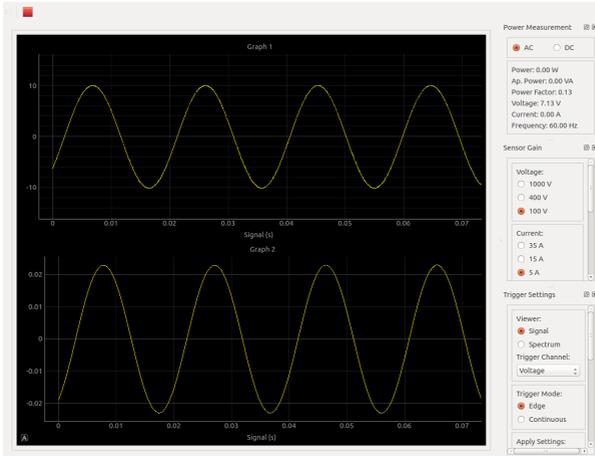


Figura 6.56 – Interface gráfica usada para validação da estimação de frequência de uma forma de onda senoidal. Sinal superior: forma de onda de entrada. Sinal inferior: forma de onda da saída do filtro passa-banda.



Figura 6.57 – Interface gráfica usada para validação da estimação de frequência de uma forma de onda quadrada. Sinal superior: forma de onda de entrada. Sinal inferior: forma de onda da saída do filtro banda passante.

Tabela 6.9 – Teste do algoritmo de estimação de frequência em comparação com o multímetro.

Gerador de Função [Hz]	Multímetro [Hz]	Estimador de Frequência [Hz]
20	20,0005	20
40	40,001	40
60	60,001	60,01
100	100,003	100,01
160	160,004	160,01
200	200,005	200,01
1000	1000,03	1000,05
1300	1300,04	1300,14
1500	1500,04	1500,12
1800	1800,05	1800,2
2000	2000,05	2000,16
2500	2500,07	2500,2
3000	3000,08	3000,3
4000	4000,11	4000,45
5000	5000,13	5001,1
6000	6000,16	6002
7000	7000,18	7001,4
8000	8000,2	8003,2
10000	10000,3	10002
12000	12000,3	12004
15000	15000,4	15009
20000	20000,6	20020
30000	30000,8	30100
40000	40001	40020

6.6 CALIBRAÇÃO DO CIRCUITO DE AQUISIÇÃO

A fim de ajustar as não-linearidades do processo de conversão analógico-digital, além de compensar os erros propagados proveniente da tolerância dos componentes realizou-se a calibração dos circuitos de aquisição, tanto de tensão quanto para o de corrente. Mediu-se a tensão e corrente através da interface desenvolvida, comparando os valores com medições realizadas através dos multímetros Agilent 34401A para medição de tensão e o multímetro Tektronix DMM4040 para a medição de corrente

A partir dos valores tabelados para a escala de tensão e corrente dados pelas Tabelas 6.11 e 6.12 realizou-se a regressão para uma função de segunda ordem, como em (6.1), sendo x o sinal de entrada, proveniente da medição do ADC, e y é a componente de medição calibrada.

$$\hat{y} = \alpha + \beta x + \gamma x^2 \quad (6.1)$$

Tabela 6.10 – Coeficientes resultantes da regressão para um polinômio de segunda ordem.

Escala	α	β	γ
1000 V	0,0006	0,7187	0,0066
400 V	0,0022	0,6958	0,0480
100 V	0,0015	0,6972	0,0446
35 A	0	1,0059	0,0015
15 A	0	1,0058	0,0007
5 A	0	1,0058	0,0005

O bloco implementado no FPGA permite que o processador atualize os coeficientes dados por α , β e γ para as escalas de tensão e corrente. Os registradores dos coeficientes estão listados a partir da Tabela 4.7, sendo que o coeficiente α mapeia o sufixo a_0 , β em a_1 e γ em a_2 .

Tabela 6.11 – Medições utilizadas na calibração do canal de tensão.

Multímetro	Tensão [V]		
	Escala 100V	Escala 400V	Escala 1000V
0,00	-0,036622	-0,158696	-0,396741
0,09	0,057985	-0,109867	-0,427259
10,02	13,806574	13,599048	13,306070
20,01	27,622000	27,417829	27,130955
30,01	41,654700	41,273232	40,894903
40,01	55,308695	55,128636	54,719687
50,08	69,014557	69,154942	68,636128
60,02	81,688000	82,851650	82,338939
70,11	94,354060	96,755882	96,469008
85,05	-	117,777032	116,916410
100,01	-	139,005707	137,791070
120,93	-	167,700000	1676,966765
141,45	-	195,465900	195,928000
150,00	-	206,268502	207,037568
170,18	-	234,601886	235,389264
185,03	-	255,745109	256,904813
200,01	-	274,861904	275,948363
220,20	-	299,740593	304,239021
240,27	-	326,462000	332,590716
260,23	-	350,010681	306,667745
280,10	-	376,036866	388,348033
300,06	-	-	416,608173
330,14	-	-	457,045198
360,08	-	-	495,742668
400,12	-	-	552,903836
420,57	-	-	581,377606
440,34	-	-	608,691671

Tabela 6.12 – Medições utilizadas na calibração do canal de corrente.

Multímetro	Corrente [A]		
	Escala 5 A	Escala 15 A	Escala 35 A
0,002562	0,002594	0,002747	0,004273
0,286245	0,284280	0,284738	0,286240
0,571900	0,567949	0,568560	0,569323
0,857070	0,852077	0,852382	0,853450
1,141700	1,136052	1,136204	1,137577
1,430500	1,421552	1,421857	1,422773
1,714300	1,704154	1,704306	1,705832
2,002540	1,991485	1,991333	1,992096
2,430090	2,415387	2,415693	2,416150
2,857150	2,840358	2,840510	2,841273
3,455000	3,431501	3,428000	3,428800
4,061800	4,036531	4,037141	4,038667
4,278880	4,253517	4,253212	4,254430
5,056830	-	5,038759	5,039521
5,201240	-	5,169225	5,170904
5,402400	-	5,369274	5,369579
5,601030	-	5,566576	5,568255
6,002600	-	5,965758	5,966674
6,310360	-	6,271554	6,272164
6,608590	-	6,566820	6,568041
7,005240	-	6,961882	6,963526
7,302650	-	7,257607	7,258065
7,603500	-	7,556078	7,557146
8,002480	-	7,953429	7,953429
8,500800	-	8,448286	8,449049
9,001810	-	8,947274	8,947874
9,500000	-	9,441206	9,442427

6.7 EXATIDÃO DO PROTÓTIPO

De modo a analisar a exatidão de medição dos circuitos de aquisição de tensão e corrente do protótipo realizou-se testes em bancada para sinais de corrente contínua (CC) e corrente alternada (CA) através de uma fonte linear e de um variac. Dessa forma, mediu-se tensão do multímetro Agilent 34401A, com precisão de medição de tensão contínua de 0,02 %, e corrente com o multímetro Tektronix DMM4040, com precisão de medição de corrente contínua de 0,15 %, além de realizar a conexão com o analisador de potência Yokogawa WT210. Com as medições de corrente, tensão e potência dos três dispositivos, calculou-se o erro relativo das medições do analisador de potência WT210 e do protótipo tomando como valor real a medição dos multímetros. Considerou-se o valor real de potência o produto dos valores de corrente e tensão dos multímetros para medição CC. Torna-se importante mencionar que não foram considerados nesse teste elevações de temperatura nem o impacto dessa variação na exatidão dos circuitos de medição.

6.7.1 Análise com Sinais CC

Nas Tabelas 6.13, 6.14 e 6.15 são apresentados as comparações entre os valores medidos do protótipo com o analisador de potência WT210 realizado com um sinal CC. Nota-se que a calibração do ADC diminui drasticamente o erro proveniente da medição de tensão. No entanto, o erro continua alto em relação a um instrumento de precisão, em que se espera erros menores do que 0,1%. Já pela tabela referente a medição de corrente tem-se que o erro relativo é tão baixo quanto o do analisador WT210, que oferece precisão de 0,1%. Para a relação de potência tem-se que a medição de tensão propaga o sinal de erro, gerando um resultado insatisfatório se comparado ao analisador WT210.

Tabela 6.13 – Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para o canal de tensão no teste em corrente contínua.

Multímetro	Tensão [V]		Erro Relativo [%]	
	Yokogawa	Protótipo	Yokogawa	Protótipo
10,008	10,007	9,903	0,01%	1,05%
20,014	20,006	19,797	0,04%	1,09%
40,016	40,019	40,116	-0,01%	-0,25%
60,062	60,04	59,779	0,04%	0,47%
80,383	80,34	79,421	0,05%	1,21%
100,102	100,06	99,746	0,04%	0,36%
120,06	120,01	120,328	0,04%	-0,22%
140,4	140,39	140,250	0,01%	0,11%
160,53	160,52	160,710	0,01%	-0,11%
180,03	179,99	180,291	0,02%	-0,14%
200,02	199,98	200,738	0,02%	-0,36%
250	249,97	250,227	0,01%	-0,09%
300,25	300,31	300,912	-0,02%	-0,22%
350,08	350,06	349,950	0,01%	0,04%
380,42	380,35	379,250	0,02%	0,31%

Tabela 6.14 – Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para o canal de corrente no teste em corrente contínua.

Multímetro	Corrente [A]		Erro Relativo [%]	
	Yokogawa	Protótipo	Yokogawa	Protótipo
0,8562	0,8559	0,8563	0,04%	-0,02%
1,7116	1,7120	1,7121	-0,02%	-0,03%
3,4251	3,4253	3,4263	-0,01%	-0,04%
5,1394	5,1393	5,1410	0,00%	-0,03%
6,8815	6,8820	6,8818	-0,01%	0,00%
8,5712	8,5730	8,5733	-0,02%	-0,02%
10,2816	10,2840	10,2853	-0,02%	-0,04%
1,3343	1,3347	1,3344	-0,03%	-0,01%
1,5253	1,5263	1,5258	-0,06%	-0,03%
1,7111	1,7115	1,7110	-0,02%	0,01%
1,9014	1,9018	1,9012	-0,02%	0,01%
2,3768	2,3777	2,3769	-0,04%	-0,01%
2,8554	2,8570	2,8559	-0,06%	-0,02%
3,3287	3,3302	3,3294	-0,04%	-0,02%
3,6174	3,6180	3,6183	-0,02%	-0,02%

Tabela 6.15 – Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para a potência estimada no teste em corrente contínua.

Potência [W]			Erro Relativo [%]	
Multímetro	Yokogawa	Protótipo	Yokogawa	Protótipo
8,569	8,565	8,480	0,05%	1,05%
34,257	34,251	33,900	0,02%	1,05%
137,057	137,080	137,470	-0,02%	-0,30%
308,684	308,540	307,200	0,05%	0,48%
553,152	553,000	546,610	0,03%	1,20%
857,996	857,800	855,320	0,02%	0,31%
1234,408	1234,500	1237,730	-0,01%	-0,27%
187,337	187,390	187,200	-0,03%	0,07%
244,861	245,000	245,260	-0,06%	-0,16%
308,053	308,080	308,530	-0,01%	-0,15%
380,312	380,360	381,720	-0,01%	-0,37%
594,195	594,400	594,790	-0,03%	-0,10%
857,334	858,200	859,600	-0,10%	-0,26%
1165,315	1165,800	1165,350	-0,04%	0,00%
1376,131	1376,300	1372,660	-0,01%	0,25%

6.7.2 Análise com Sinais CA

A partir de um Variac realizou-se os testes comparativos para sinais CA. As Tabelas 6.16, 6.17 e 6.18 apresentam os resultados dos testes para a tensão eficaz, corrente eficaz e potência ativa. Vê-se que o erro de tensão é responsável pelo maior desvio do valor real, sendo que o de corrente se manteve próximo ao resultado obtido com o WT210. Para o comparativo de potência tomou-se o valor do analisador WT210 como o valor real de potência, permitindo o cálculo do erro relativo do protótipo em relação ao analisador.

Tabela 6.16 – Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para o canal de tensão no teste em corrente alternada.

Tensão Eficaz [V]			Erro Relativo	
Multímetro	Yokogawa	Protótipo	Yokogawa	Protótipo
20,961	20,895	20,630	0,32%	1,60%
40,656	40,594	40,108	0,15%	1,37%
80,450	80,680	79,960	-0,29%	0,61%
100,582	100,580	99,770	0,00%	0,81%
120,700	120,740	119,930	-0,03%	0,64%
160,360	160,190	159,560	0,11%	0,50%
200,140	200,180	199,460	-0,02%	0,34%
220,160	220,140	219,270	0,01%	0,41%

Tabela 6.17 – Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para o canal de corrente no teste em corrente alternada.

Corrente Eficaz [A]			Erro Relativo	
Multímetro	Yokogawa	Protótipo	Yokogawa	Protótipo
0,1995	0,1988	0,1988	0,35%	0,31%
0,3864	0,3863	0,3854	0,04%	0,25%
0,7662	0,7670	0,7647	-0,11%	0,19%
0,9560	0,9582	0,9544	-0,23%	0,17%
1,1460	1,1480	1,1352	-0,17%	0,95%
1,5241	1,5250	1,5214	-0,06%	0,18%
1,9034	1,9063	1,9010	-0,15%	0,13%
2,0931	2,0956	2,0905	-0,12%	0,12%

Tabela 6.18 – Comparativo entre o analisador de potência Yokogawa WT210 e o protótipo para a potência estimada no teste em corrente alternada.

Potência Ativa [W]		Erro Relativo
Yokogawa	Protótipo	Protótipo
4,15	4,09	1,45%
15,68	15,45	1,47%
61,70	61,23	0,76%
96,55	96,11	0,46%
138,46	137,30	0,84%
244,40	242,64	0,72%
382,44	379,03	0,89%
460,50	458,48	0,44%

6.8 INTERFACE GRÁFICA

Com a finalidade de validar o visualizador de formas de onda da interface gráfica, realizou-se a comparação dessa com o osciloscópio Tektronix MSO4034. As formas de onda são de um inversor de frequência acionando um motor de indução trifásico, sendo que as variáveis medidas são a tensão e corrente de linha aplicada nos enrolamentos do motor. Pelas Figuras 6.58 e 6.59 tem-se o modo "Trigger de Nível". Nota-se pela interface que o trigger fora ajustado para o canal de corrente, na posição indicada pelo traçado em branco. Ademais, a Figura 6.64 apresenta o resultado para a visualização do espectro.

Pelas Figuras 6.60 e 6.61 pode-se ver as formas de onda em detalhes. A visualização do eixo referente ao tempo é ajustado para 12 ms, permitindo assim visualizar o sinal modulado de tensão assim como a componente de alta frequência na corrente de linha do motor.



Figura 6.58 – Formas de onda do osciloscópio Tektronix em modo trigger de nível. Sinal verde: tensão de linha. Sinal amarelo: corrente de linha.

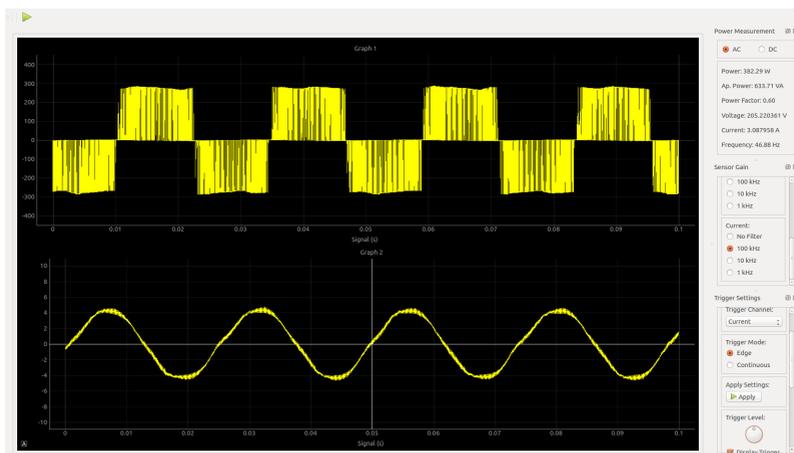


Figura 6.59 – Formas de onda da interface gráfica em modo trigger de nível. Sinal superior: tensão de linha. Sinal inferior: corrente de linha.

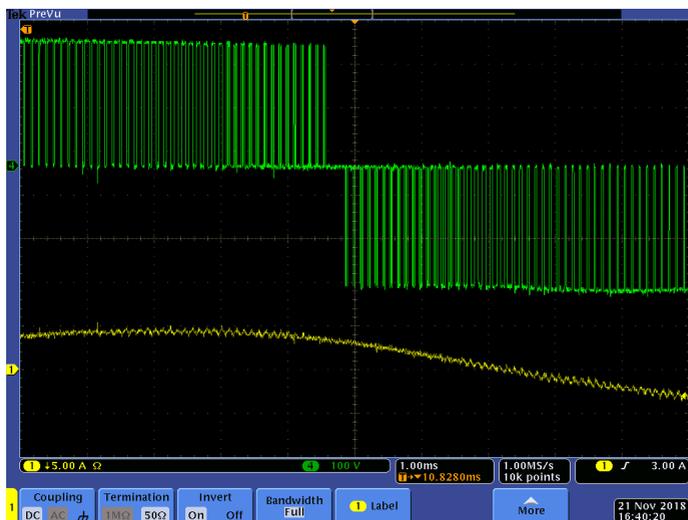


Figura 6.60 – Formas de onda do osciloscópio Tektronix em modo trigger de nível. Sinal verde: tensão de linha. Sinal amarelo: corrente de linha.

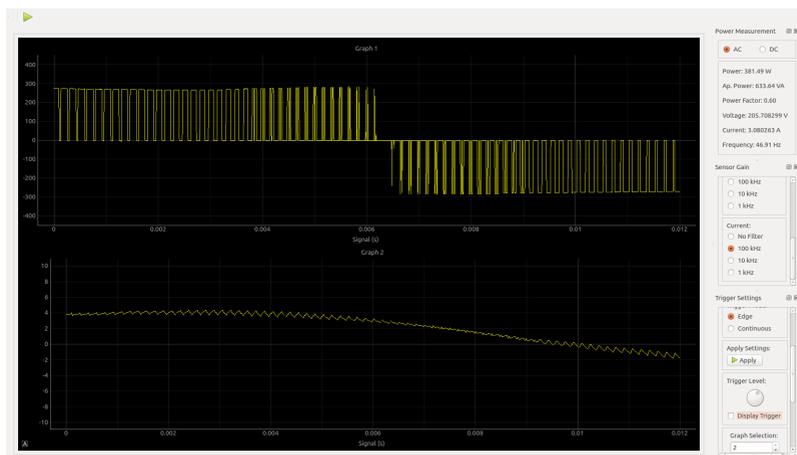


Figura 6.61 – Formas de onda da interface gráfica em visualização em modo trigger de nível. Sinal superior: tensão de linha. Sinal inferior: corrente de linha.

Para o "Modo Contínuo" tem-se o resultado dado pelas Figuras 6.62 e 6.63. Percebe-se a variação das componentes fundamentais de corrente e tensão resultante da aceleração do motor. Dessa forma, pode-se medir respostas transitórias de sistemas e que não são perceptíveis ou difíceis de caracterizar dentro do modo "Trigger de Nível".



Figura 6.62 – Formas de onda do osciloscópio Tektronix em visualização no modo contínuo. Sinal verde: tensão de linha. Sinal amarelo: corrente de linha.

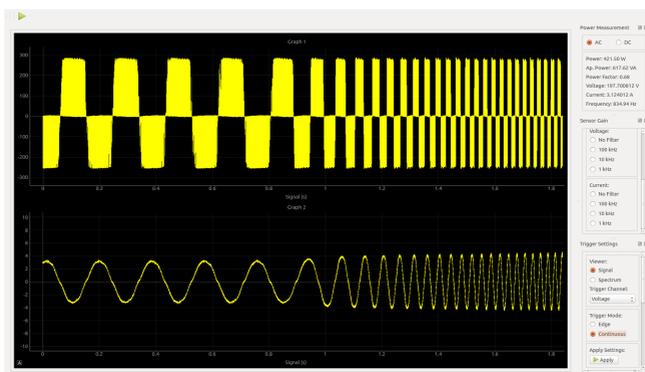


Figura 6.63 – Formas de onda da interface gráfica em visualização no modo contínuo. Sinal superior: tensão de linha. Sinal inferior: corrente de linha.

Percebe-se através da Figura 6.64 a fundamental da tensão e corrente, próximos de 61,2 Hz, como dado pela frequência estimada. Também, nota-se o impacto da modulação por largura de pulso da componente de tensão de linha aplicado no motor.

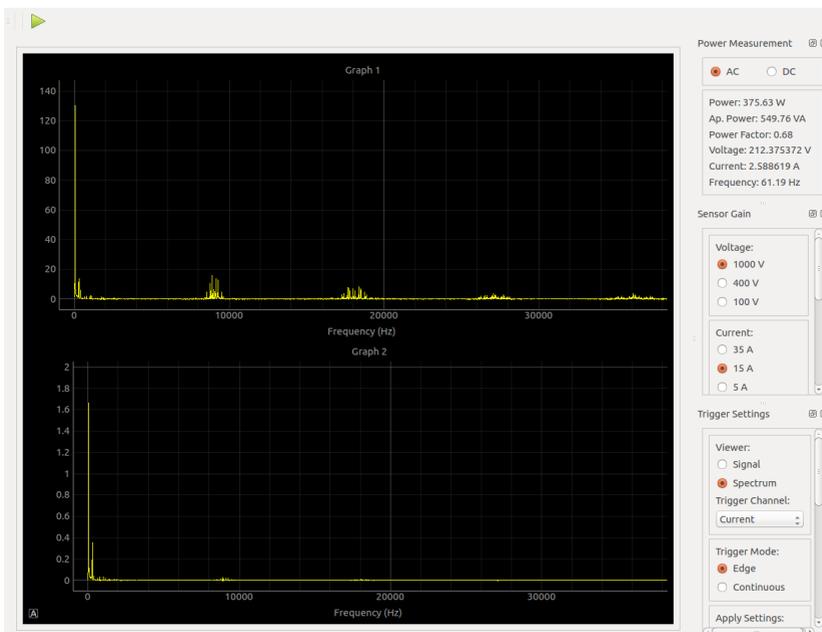


Figura 6.64 – Formas de onda da interface gráfica em visualização da magnitude do espectro. Espectro superior: tensão de linha. Espectro inferior: corrente de linha.

Através da Figura 6.65 vê-se o visualizador do analisador de potência, que disponibiliza os valores de potência ativa, aparente, fator de potência, tensão, corrente e a frequência estimada. Esse resultado foi retirado da Figura 6.58, que representa a operação do inversor acionando um motor de indução trifásico. Como a análise realizada é a das componentes alternadas CA, os valores de corrente e tensão disponibilizados são os eficazes. Caso o usuário selecione a análise das componentes contínuas,

dados pela análise CC, os valores mensurados de tensão e corrente são os médios.

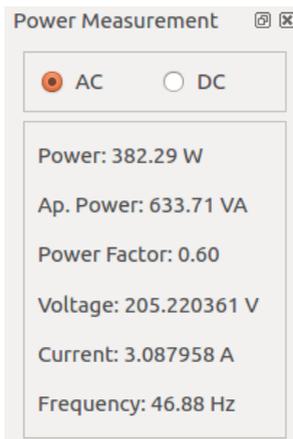


Figura 6.65 – Visualizador do analisador de potência para a análise CA.

7 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

A interface cumpriu com os requisitos de variar as escalas de ganho dos circuitos analógicos além de permitir a escolha dos filtros passa-baixas situados na placa de aquisição. O visualizador de formas de onda apresenta um ótimo desempenho para os três modos de visualização "*Trigger* de Nível", "Contínuo" e "Espectro". Obteve-se uma alta taxa de atualização das formas de onda na interface e o programa demonstrou boa responsividade mediante ações do usuário. Por exemplo, modificações no nível do *trigger* atualiza instantaneamente a visualização do gráfico. Acerca do modo "*Trigger* de Nível", sinais que possuem componentes de alta frequência podem acusar falsamente o sistema de detecção de *trigger*. Com isso, o sinal visualizado na interface pode não parecer estático, dificultando a análise dos sinais monitorados. A fim de resolver esse problema, pode-se filtrar o sinal de interesse a fim de buscar uma condição de *trigger* no sinal com a ausência de componentes de alta frequência, diminuindo as chances de ocorrer uma falsa detecção.

Relacionado a precisão, o erro de conversão do ADC para o canal de tensão impactou negativamente na precisão das medições de tensão e potência. Contudo, solucionando o problema do ADC espera-se atingir níveis de precisão comparáveis ao do canal de corrente. Esse, por sua vez, apresentou precisão comparável ao do WT210, da Yokogawa. Acredita-se que a falha é proveniente do ADC, visto que o circuito analógico de medição de tensão foi validado, confirmando os resultados de simulação. Testes futuros visam resolver esse problema através da troca do componente ADS8350.

O circuito de estimação de frequência possui bom desempenho para uma determinada faixa de frequência, que varia entre 30 Hz e 20 kHz. No entanto o número de pontos utilizado para

o cálculo da DFT impacta na frequência mínima de detecção. Nos testes em bancada, conseguiu-se estimar frequências maiores que 30 Hz, aproximadamente. Abaixo de 30 Hz a DFT acusa frequências que desviam do valor real, condicionando erroneamente o filtro passa-banda. Trabalhos futuros podem investigar outras formas para a implementação do algoritmo de estimação de frequência, como através de um PLL ou de detectores de passagem por zero situados no circuito analógico.

Para a placa de circuito impresso, permite-se analisar a capacidade de rejeição de ruídos de modo comum assim como o teste de isolamento do circuito de tensão, que fora projetado para atender um valor máximo de 1000 V. Adicionalmente, pode-se realizar estudos a fim de identificar se a resistência das chaves seletoras tem influência sobre os circuitos de ganhos programáveis. Sobre os capacitores dispostos em paralelo com os divisores resistivos do circuito divisor de tensão, como mostrado pela Figura C.2 do Apêndice C, referente ao circuito da Figura 3.8, pode-se analisar o ganho de banda através da compensação dada pelos capacitores, tendo em vista que as chaves seletoras, os amplificadores operacionais e a própria placa de circuito impresso possuem capacitâncias intrínsecas que podem influenciar negativamente na banda da resposta em frequência do circuito em questão. Resultados que permitem enriquecer o estudo são a inclusão da fase nos diagramas de Bode apresentados na Seção 6.4, permitindo verificar se a fase dos circuitos analógicos não induz erros na análise realizada acerca da energia e potência instantânea no sistema de processamento digital devido a uma defasagem entre os sinais de tensão e corrente ocasionada pelo circuito de aquisição analógico.

Por último, mudanças simples na placa de circuito impresso foram tomadas a fim de corrigir problemas de esquemático que ocorreram no amplificador operacional IC_{12B} , do circuito de aquisição de tensão. Vale lembrar que o esquemático disponibilizado no Apêndice C é a versão corrigida.

Referências

- [1] R. W. Erickson, *Fundamental of Power Electronics*. University of Colorado. Boulder, Colorado: Kluwer Academic, 2 ed., 2004.
- [2] D. Kaminsky, J. Zidek, and P. Bilik, “Virtual instrumentation based power quality analyzer,” in *Proceedings of the 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems*, vol. 1, pp. 184–187, Sept. 2011.
- [3] L. Wang, Z. Liu, G. Zhang, and L. Zhao, “A multi-channel distributed monitoring platform based on ethernet and virtual instrument technology,” in *2009 9th International Conference on Electronic Measurement Instruments*, pp. 4–579–4–584, Aug. 2009.
- [4] J.-H. Teng, S.-Y. Chan, J.-C. Lee, and R. Lee, “A LabVIEW based virtual instrument for power analyzers,” in *PowerCon 2000. 2000 International Conference on Power System Technology. Proceedings (Cat. No.00EX409)*, vol. 1, pp. 179–184 vol.1, Dec. 2000.
- [5] H. Wuhuang, P. Zhixiang, G. Jian, C. Haozhe, Y. Peng, and Q. Duyu, “Design and implementation of 250msps wireless smart virtual oscilloscope,” in *2017 13th IEEE International Conference on Electronic Measurement Instruments (ICEMI)*, pp. 373–378, Oct. 2017.
- [6] C. K. Alexander and M. N. O. Sadiku, *Fundamentals of Electric Circuits*. Raghathan Srinivasan, 5 ed.

- [7] D. Halliday, R. Resnick, and J. Walker, *Fundamentals of Physics*. 111 River Street, Hoboken: John Wiley & Sons, Inc., 9 ed., 2012.
- [8] R. M. Kerchner and G. F. Corcoran, *Alternating-Current Circuits*. Wiley, 1938.
- [9] H. Akagi, E. H. Watanabe, and M. Aredes, *Instantaneous Power Theory and Applications to Power Conditioning*. IEEE Press, 1 ed.
- [10] A. E. Emanuel, *Apparent and Reactive Powers in Three-Phase Systems: In Search of a Physical Meaning and a Better Resolution*. ETEP—Eur. Trans. Elect. Power Eng., 1 ed., 1993.
- [11] J. Fessler, *Discrete-time signals and systems*, 5 2014. Student Version.
- [12] M. Mojiri, M. Karimi-Ghartemani, and A. Bakhshai, “Estimation of Power System Frequency Using an Adaptive Notch Filter,” *IEEE Transactions on Instrumentation and Measurement*, vol. 56, pp. 2470–2477, Dec. 2007.
- [13] D. W. P. Thomas and M. S. Woolfson, “Evaluation of frequency tracking methods,” *IEEE Transactions on Power Delivery*, vol. 16, pp. 367–371, July 2001.
- [14] H. Karimi, M. Karimi-Ghartemani, and M. R. Iravani, “Estimation of frequency and its rate of change for applications in power systems,” in *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491)*, vol. 2, pp. 1111–, July 2003.
- [15] Yokogawa, *WT210/WT230. Digital Power Meter. User’s Manual*, 5 2013. 5th Edition.

- [16] Yokogawa, *WT500 Power Analyzer. User's Manual*, 9 2017. 8th Edition.
- [17] Yokogawa, *WT1800. Precision Power Analyzer. User's Manual*, 9 2017. 4th Edition.
- [18] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*. Prentice Hall, Upper Saddle River, New Jersey 07458: Prentice-Hall, Inc., 2 ed., 1999.
- [19] B. P. Lathi, *Signal Processing And Linear Systems*. Berkeley-Cambridge Press, 1 ed.
- [20] M. Frigo and S. G. Johnson, "The Design and Implementation of FFTW3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005.
- [21] M. Frigo and S. G. Johnson, "FFTW: an adaptive software architecture for the FFT," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*, vol. 3, pp. 1381–1384 vol.3, May 1998.
- [22] Frigo, M., *FFTW Manual*, 5 2018. Version 3.3.8.
- [23] A. Agarwal and J. H. Lang, *Foundations of Analog and Digital Electronic Circuits*. 500 Sansome Street, Suite 400, San Francisco, CA 94111: Morgan Kaufmann Publishers, 1 ed., 2005.
- [24] Vishay Siliconix, *DG412LE. 16 Ω , Low Parasitic Capacitance and Leakage, +12 V / +5 V / +3 V / \pm 5 V Quad SPST Switches.*, 3 2016. Rev. A.
- [25] Texas Instruments, *ADSxx50 Dual, 750-kSPS, 16-, 14-, and 12-Bit, Simultaneous-Sampling, Analog-to-Digital Converters*, 3 2018. Revised.

- [26] C. E. Shannon, "Communication in the Presence of Noise," *Proceedings of the IRE*, vol. 37, pp. 10–21, Jan. 1949.
- [27] Bo Yang, F. Lee, A. Zhang, and Guisong Huang, "LLC resonant converter for front end DC/DC conversion," in *APEC. Seventeenth Annual IEEE Applied Power Electronics Conference and Exposition (Cat. No.02CH37335)*, vol. 2, (Dallas, TX, USA), pp. 1108–1112, IEEE, 2002.
- [28] J. F. Lazar and R. Martinelli, "Steady-state analysis of the LLC series resonant converter," in *APEC 2001. Sixteenth Annual IEEE Applied Power Electronics Conference and Exposition (Cat. No.01CH37181)*, vol. 2, pp. 728–735 vol.2, Mar. 2001.
- [29] B. Lu, W. Liu, Y. Liang, F. C. Lee, and J. D. v. Wyk, "Optimal design methodology for LLC resonant converter," in *Twenty-First Annual IEEE Applied Power Electronics Conference and Exposition, 2006. APEC '06.*, pp. 6 pp.–, Mar. 2006.
- [30] G. Ivensky, S. Bronshtein, and A. Abramovitz, "Approximate Analysis of Resonant LLC DC-DC Converter," *IEEE Transactions on Power Electronics*, vol. 26, pp. 3274–3284, Nov. 2011.
- [31] Intel, *MAX V Device Handbook*.
- [32] Intel, *Embedded Peripherals IP User Guide*, 9 2018. UG-01085.
- [33] Intel, *Cyclone V Hard Processor System. Technical Reference Manual*, 7 2018.
- [34] E. Gamma, R. Helm, R. Johnon, and J. Vlissides, *Design Patterns. Elements of Reusable Object-Oriented Software*.

201 W. 103rd Street, Indianapolis, IN 46290: Addison-Wesley, 37 ed., 2009.

- [35] Texas Instruments, *OPAx350 High-Speed, Single-Supply, Rail-to-Rail Operational Amplifiers MicroAmplifier*, 12 2015. Revised.
- [36] Vishay Foil Resistors, *Bulk Metal Technology High Precision, Current Sensing, Power Surface Mount, Metal Strip Resistor with Resistance Value from 1 mOhm, Rated Power up to 3 W and TCR to 0 ± 15 ppm/ $^{\circ}$ C Maximum*, 8 2015. Revised.
- [37] Texas Instruments, *OPAx192 36-V, Precision, Rail-to-Rail Input/Output, Low Offset Voltage, Low Input Bias Current Op Amp with e-trim*, 11 2015. Revised.
- [38] Texas Instruments, *ISO774x High-Speed, Robust-EMC Reinforced Quad-Channel Digital Isolators*, 3 2016. Revised.
- [39] Texas Instruments, *HFBR-0508Z Series, HFBR-1528Z Transmitter, HFBR-2528Z Receiver 10 Megabaud Versatile Link Fiber Optic Transmitter and Receiver for 1 mm POF and 200 μ m HCS[®]*, 11 2011. Revised.
- [40] Keysight Technologies, *InfiniiVision 4000 X-Series Oscilloscopes*.
- [41] E. L. Oberstar, *Fixed-Point Representation & Fractional Math*. Oberstar Consulting. Revision 1.2.

**APÊNDICE A – Parte do Software do Processador ARM da
Placa de Desenvolvimento**


```

1  class UdpDataTransmission
2  {
3      private:
4          class State * currentState ;
5          bool isClientConnected ;
6
7      public:
8          UdpDataTransmission();
9          ~UdpDataTransmission();
10
11         // Set Current State
12         void setCurrentState (State *s){ currentState = s; };
13
14         // State Methods
15         void bind_to_client (UdpWrapper *ptr_network_handler);
16
17         void send_data_over_network(UdpWrapper
18             *ptr_network_handler, MemoryOffsetHandler mem_offset,
19             unsigned int *memoryPtr, ClientCommand
20             &ClientUdpCmd, PowerAnalyzer &StPowerMeasurements,
21             FftDataStorage fft_data[]);
22
23         void recv_data_from_network(UdpWrapper
24             *ptr_network_handler, ClientCommand &ClientUdpCmd,
25             memoryMap *decimator_m_reg, analogSwitches
26             &analog_sw);
27
28         void retrieve_ram_data (ClientCommand &ClientUdpCmd,
29             MemoryOffsetHandler &mem_offset, memoryMap
30             *trigger_offset_reg, memoryMap *memptr_reg);
31
32         void set_acquisition_mode (int i);
33     };

```

Algoritmo A.1 – Declaração da classe UdpDataTransmission.

```

1  #include "UdpDataTransmission.h"
2
3  UdpDataTransmission::UdpDataTransmission()
4  {
5      isClientConnected = false ;
6      currentState = new ClientDisconnected () ;
7  }
8
9  void UdpDataTransmission:: bind_to_client (UdpWrapper
      *ptr_network_handler)
10 {
11     currentState ->bind_to_client(this , ptr_network_handler) ;
12 }
13
14 void UdpDataTransmission::send_data_over_network(UdpWrapper
      *ptr_network_handler, MemoryOffsetHandler mem_offset, unsigned
int *memoryPtr, ClientCommand &ClientUdpCmd, PowerAnalyzer
      &StPowerMeasurements, FftDataStorage fft_data[])
15 {
16     currentState ->send_data_over_network(this,
      ptr_network_handler , mem_offset, memoryPtr,
      ClientUdpCmd, StPowerMeasurements, fft_data);
17 }
18
19 void UdpDataTransmission::recv_data_from_network(UdpWrapper
      *ptr_network_handler, ClientCommand &ClientUdpCmd,
      memoryMap *decimator_m_reg, analogSwitches &analog_sw)
20 {
21     currentState ->recv_data_from_network(this,
      ptr_network_handler , ClientUdpCmd, decimator_m_reg,
      analog_sw);
22 }
23
24 void UdpDataTransmission:: retrieve_ram_data (ClientCommand
      &ClientUdpCmd, MemoryOffsetHandler &mem_offset, memoryMap
      *trigger_offset_reg, memoryMap *memptr_reg)
25 {
26     currentState ->retrieve_ram_data(this , ClientUdpCmd,
      mem_offset, trigger_offset_reg , memptr_reg);
27 }

```

```
28  
29 void UdpDataTransmission::set_acquisition_mode(int i)  
30 {  
31     currentState ->set_acquisition_mode(this, i);  
32 }
```

Algoritmo A.2 – Implementação dos métodos da classe UdpDataTransmission.

```
1 // Forward declaration of UdpDataTransmission Class
2 class UdpDataTransmission;
3
4 class State
5 {
6 public:
7     State () {};
8     ~State () {};
9
10    virtual void bind_to_client (UdpDataTransmission *s,
11        UdpWrapper *ptr_network_handler) { return; };
12
13    virtual void send_data_over_network(UdpDataTransmission *s,
14        UdpWrapper *ptr_network_handler, MemoryOffsetHandler
15        mem_offset, unsigned int *memoryPtr, ClientCommand
16        &ClientUdpCmd, PowerAnalyzer &StPowerMeasurements,
17        FftDataStorage fft_data[]) { return; };
18
19    virtual void recv_data_from_network(UdpDataTransmission *s,
20        UdpWrapper *ptr_network_handler, ClientCommand
21        &ClientUdpCmd, memoryMap *decimator_m_reg,
22        analogSwitches &analog_sw) { return; };
23
24    virtual void retrieve_ram_data (UdpDataTransmission *s,
25        ClientCommand &ClientUdpCmd, MemoryOffsetHandler
26        &mem_offset, memoryMap *trigger_offset_reg, memoryMap
27        *memptr_reg) { return; };
28
29    virtual void set_acquisition_mode (UdpDataTransmission *s,
30        int i) { return; };
31 };
```

Algoritmo A.3 – Declaração da classe State.

```
1 void UdpWrapper::SendSop(ClientCommand &ClientUdpCmd,  
    PowerAnalyzer &StPowerMeasurements)  
2 {  
3     string sop_str = "SOP" + "POWER=[" +  
        to_string(StPowerMeasurements.power_rms) + "]" +  
4     "VOLTAGE=[" + to_string(StPowerMeasurements.voltage_rms) +  
        "]" +  
5     "CURRENT=[" + to_string(StPowerMeasurements.current_rms) +  
        "]" +  
6     "FREQUENCY=[" +  
        to_string(StPowerMeasurements.estimated_frequency) + "];  
7  
8     if (ClientUdpCmd.plot_signal)  
9         sop_str = sop_str + "SEND=[SIGNAL]";  
10  
11    if (ClientUdpCmd.plot_fft)  
12        sop_str = sop_str + "SEND=[FFT]";  
13  
14    const char* sop = sop_str.c_str();  
15    SendDatagram(sop, strlen(sop));  
16 }
```

Algoritmo A.4 – Método da Classe UdpWrapper que monta o pacote de informações para a interface do usuário.

```

1  void UdpWrapper::SendSignalPacket(int ch_num, const void*
    memory_ptr, unsigned int channel_offset_space , unsigned int
    higher_boundary, unsigned int lower_boundary)
2  {
3      // Send start packet
4      string start_message = "SIGNAL=[" + to_string(ch_num) + "];
5      const char* signal_sop = start_message . c_str ();
6      SendDatagram(signal_sop, strlen ( signal_sop ));
7
8      // Send packet data
9      unsigned int write_span = higher_boundary – lower_boundary;
10
11     if (write_span > MAX_SPAN_VALUE)
12     {
13         unsigned int it = 0;
14         while(1)
15         {
16             if (write_span >= MAX_SPAN_VALUE)
17             {
18                 write_span = write_span –
19                     MAX_SPAN_VALUE;
20                 SendDatagram(memory_ptr +
21                     ch_num*channel_offset_space +
22                     it*MAX_SPAN_VALUE +
23                     lower_boundary,
24                     MAX_SPAN_VALUE);
25                 it += 1;
26             }
27             else if (write_span > 0)
28             {
29                 SendDatagram(memory_ptr +
30                     ch_num*channel_offset_space +
31                     it*MAX_SPAN_VALUE +
32                     lower_boundary, write_span);
33                 write_span = 0;
34             }
35             else
36                 break;
37         }
38     }
39 }

```

```
31         else
32         {
33             SendDatagram(memory_ptr +
                           ch_num*channel_offset_space + lower_boundary,
                           write_span);
34         }
35     }
```

Algoritmo A.5 – Método da Classe UdpWrapper que realiza a transferência de dados dos sinais para a interface do usuário.

```

1 void UdpWrapper::SendFFTPacket(int ch_num, float *memory_ptr,
  unsigned int data_span)
2 {
3     // Send start packet
4     string start_message = "FFT=[" + to_string (ch_num) + "]";
5     const char* signal_sop = start_message . c_str ();
6     SendDatagram(signal_sop, strlen ( signal_sop ));
7
8     // Send packet data
9     unsigned int write_span = data_span;
10
11    if (write_span > MAX_SPAN_VALUE){
12        unsigned int it = 0;
13        while(1){
14            if (write_span >= MAX_SPAN_VALUE){
15                write_span = write_span -
16                    MAX_SPAN_VALUE;
17                SendDatagram(memory_ptr +
18                    it*MAX_SPAN_VALUE,
19                    MAX_SPAN_VALUE);
20                it += 1;
21            }
22            else if (write_span > 0) {
23                SendDatagram(memory_ptr +
24                    it*MAX_SPAN_VALUE,
25                    write_span);
26                write_span = 0;
27            }
28            else break;
29        }
30    }
31    else {
32        SendDatagram(memory_ptr, write_span);
33    }
34 }

```

Algoritmo A.6 – Método da Classe UdpWrapper que realiza a transferência de dados do espectro em frequência para a interface do usuário.

```

1  int UdpSocket::RecvDatagram(char buffer [], int buffer_len )
   throw(SocketException)
2  {
3      int n_bytes;
4      socklen_t len = sizeof( _client_addr );
5
6      n_bytes = recvfrom( socket_descriptor , buffer , buffer_len ,
7                          0,( struct sockaddr *) &_client_addr, &len);
8      if(n_bytes < 0)
9      {
10         throw SocketException("Recv_from_client_failed_(
11             recvfrom())", true);
12     }
13
14     return n_bytes;
15 }
16
17 void UdpSocket::SendDatagram(const void* buffer , int buffer_len )
   throw(SocketException)
18 {
19     socklen_t len = sizeof( _client_addr );
20
21     if(sendto( socket_descriptor , buffer , buffer_len , 0, ( struct
22         sockaddr *) &_client_addr, len) < 0)
23     {
24         throw SocketException("Sending_buffer_to_client_(
25             failed_(sendto())", true);
26     }
27 }
28
29 void UdpWrapper::SendEop(void)
30 {
31     const char eop[4] = "EOP";
32     SendDatagram(eop,strlen(eop));
33     return;
34 }

```

Algoritmo A.7 – Método utilizado para o recebimento e envio de dados através do socket UDP.

```

1  int fftHandler :: set_sample_size (unsigned int num_points)
2  {
3      if (sample_size != 0)
4      {
5          delete [] in;
6          fftwf_free (out);
7      }
8
9      sample_size = num_points;
10
11     in = new float [sample_size];
12     out = (fftwf_complex*) fftwf_malloc (sizeof (fftw_complex) *
13         (sample_size/2));
14 }
15 int fftHandler :: create_plan (void)
16 {
17     if (sample_size == 0)
18         return -1;
19     else
20         plan = fftwf_plan_dft_r2c_1d (sample_size, in, out,
21             FFTW_ESTIMATE);
22     return 0;
23 }
24 int fftHandler :: execute_plan (void)
25 {
26     if (isDataLoaded)
27     {
28         fftwf_execute (plan);
29         isDataLoaded = false;
30         return 0;
31     }
32     else
33         return -1;
34 }

```

Algoritmo A.8 – Métodos da classe `fftHandler` que usam da FFTW para executar o cálculo da DFT.

APÊNDICE B – Esquemáticos de Simulação

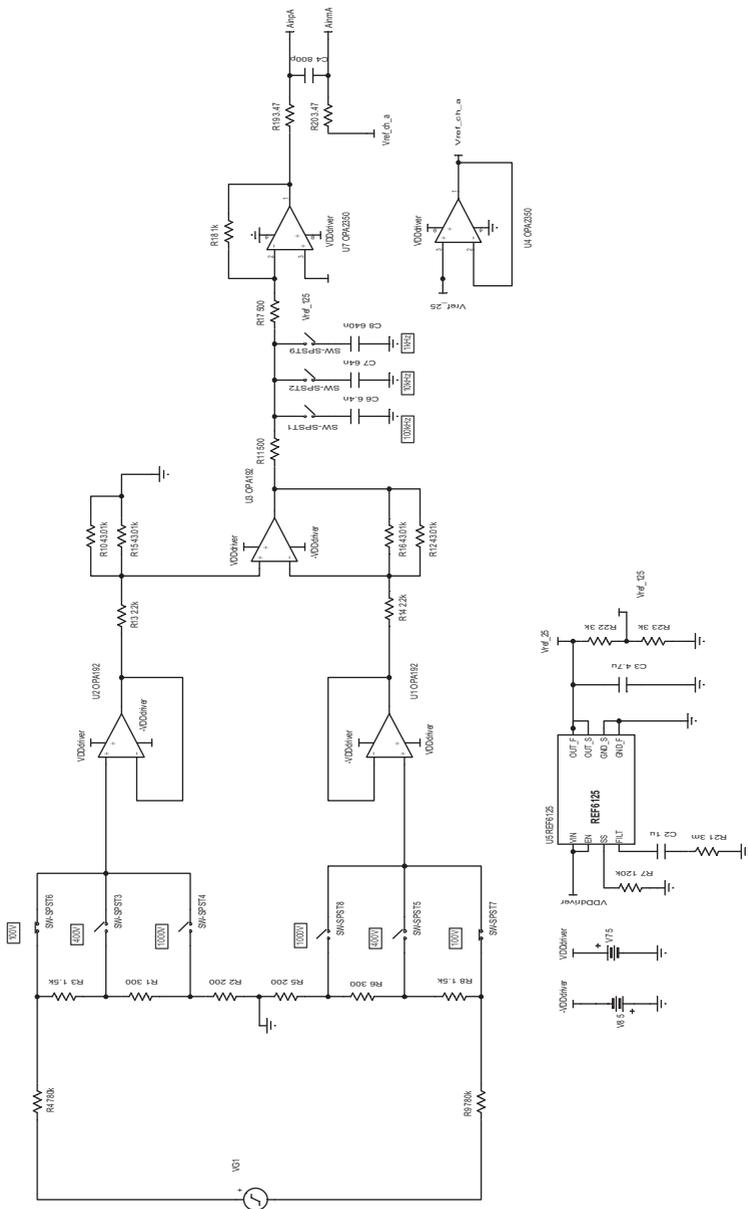


Figura B.1 – Esquemático de simulação do software Tina-TI para o circuito de aquisição de tensão.

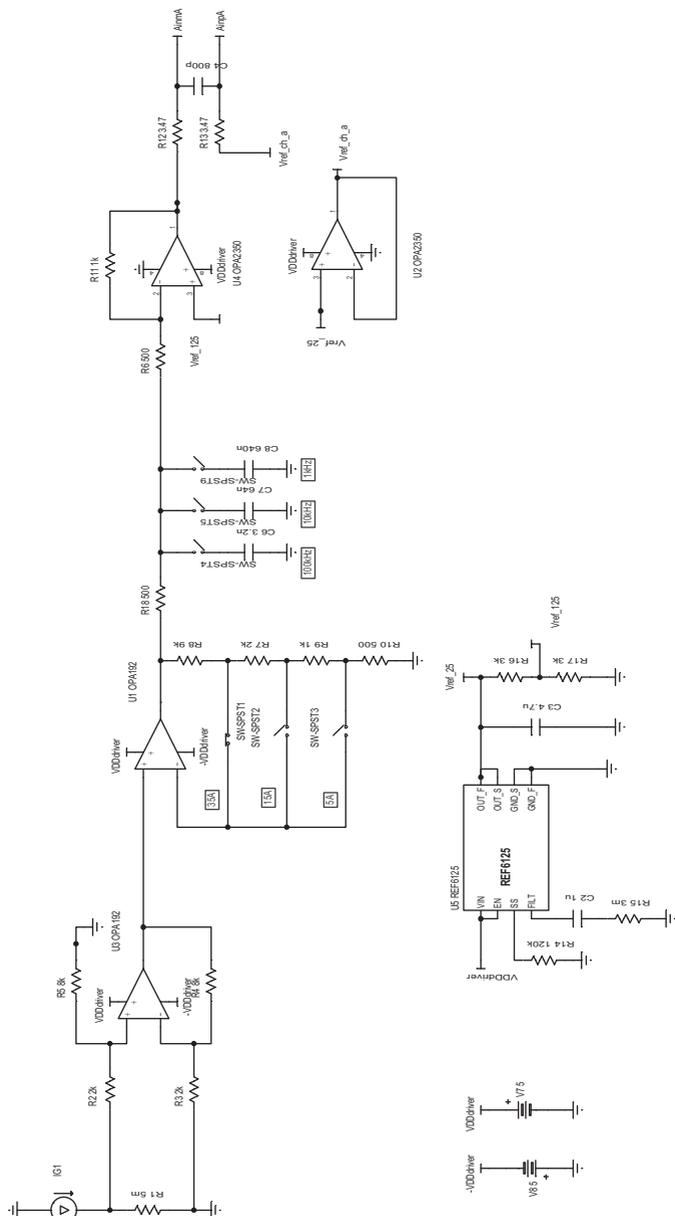


Figura B.2 – Esquemático de simulação do software Tina-TI para o circuito de aquisição de corrente.

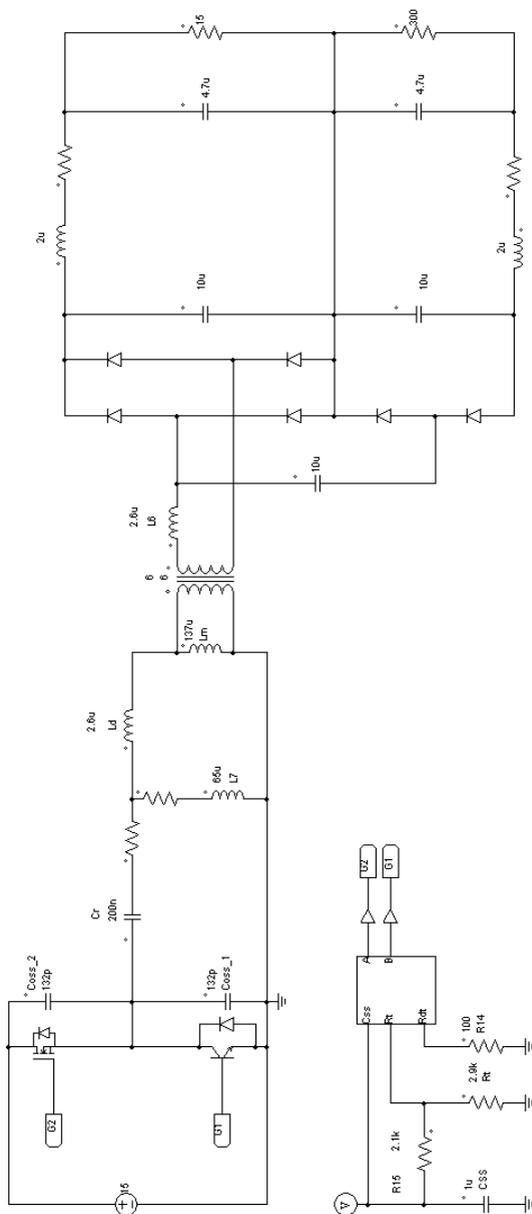


Figura B.3 – Esquemático de simulação do software PSIM para o conversor ressonante.

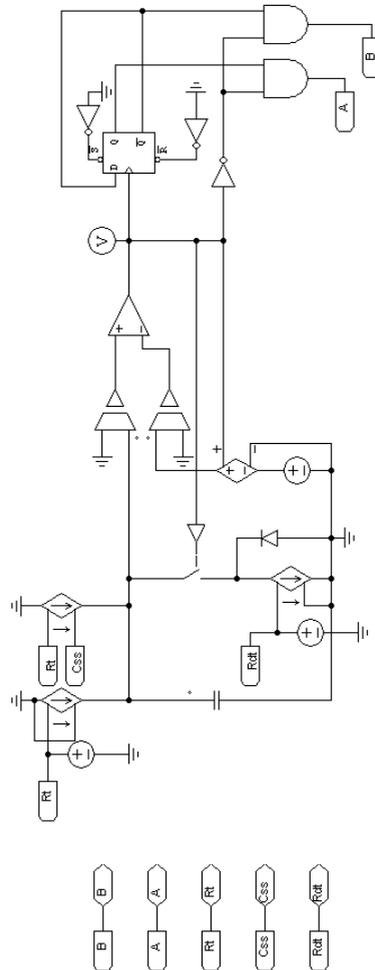


Figura B.4 – Esquemático de simulação do software PSIM para o circuito integrado NCP1397, compreendendo as funcionalidades do oscilador controlado por tensão, soft-start e aplicação de tempo morto.

APÊNDICE C – Esquemáticos da Placa de Aquisição

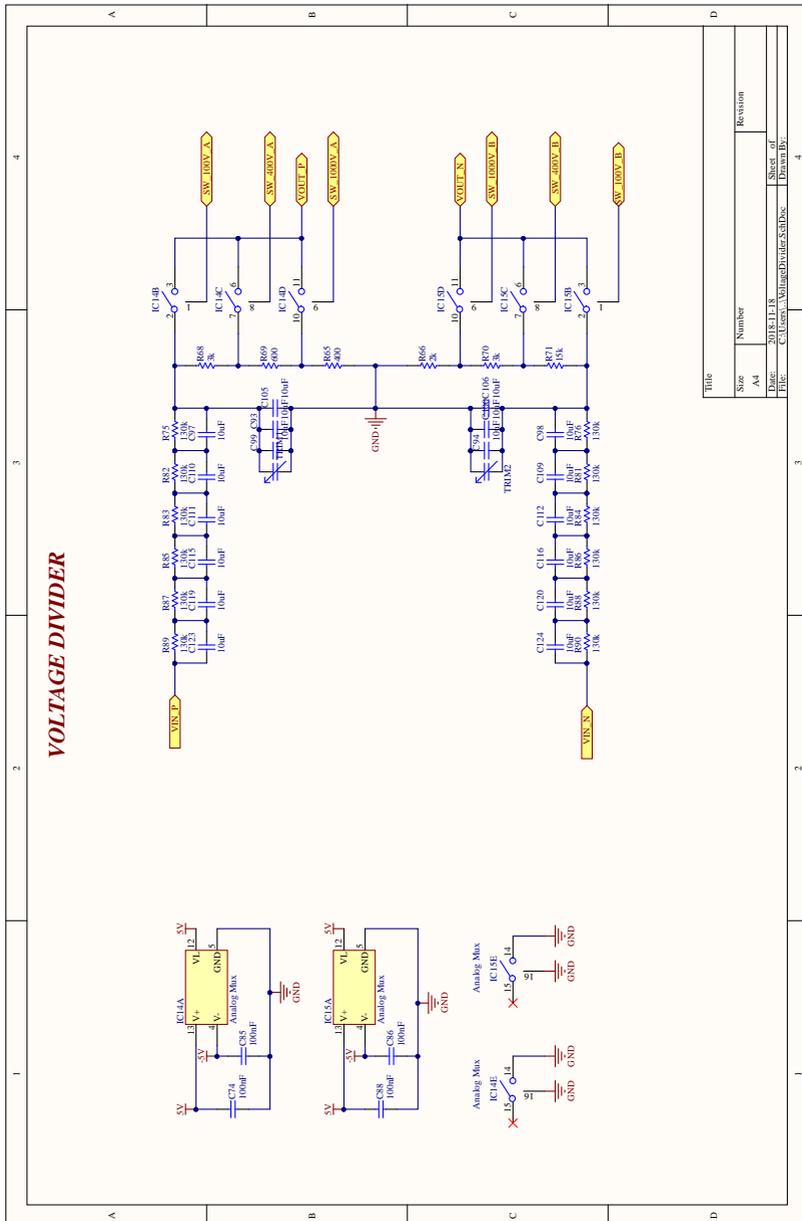


Figura C.2 – Esquemático do divisor resistivo para medição de tensão.

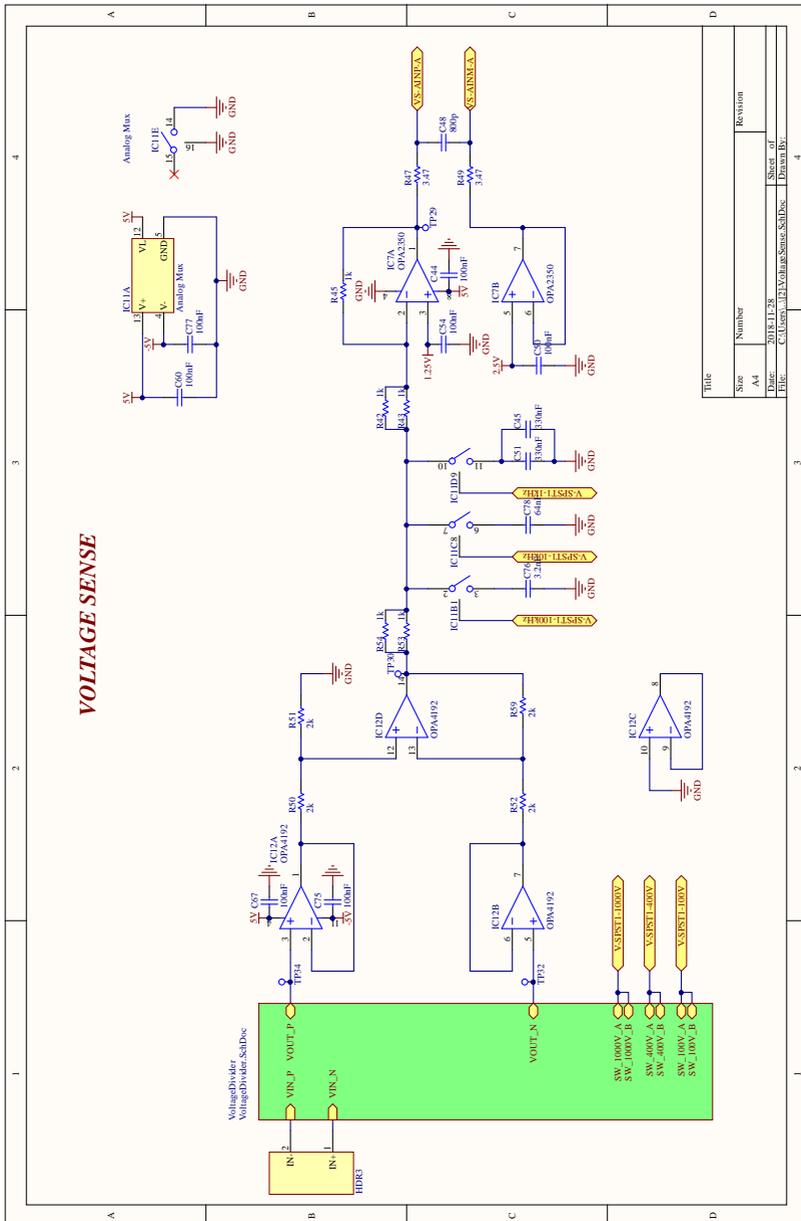


Figura C.3 – Esquemático do circuito de aquisição de tensão.

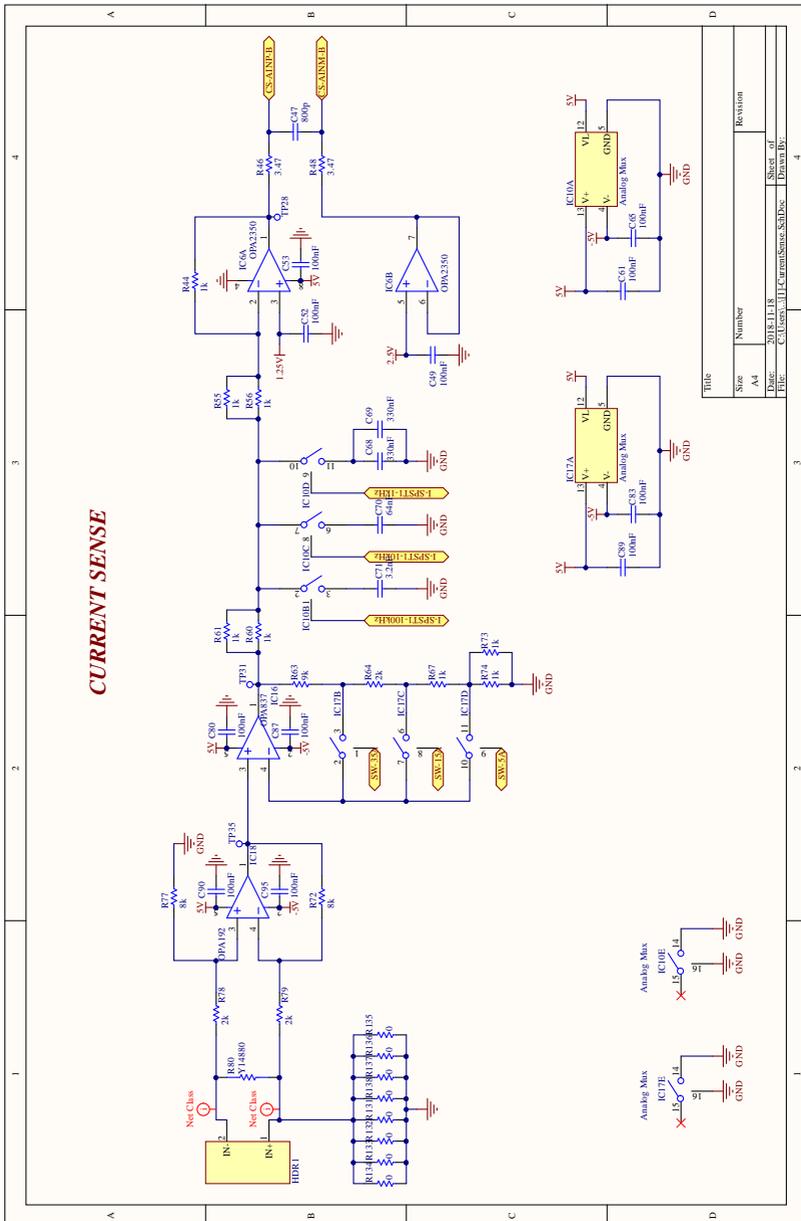


Figura C.4 – Esquemático do circuito de aquisição de corrente.

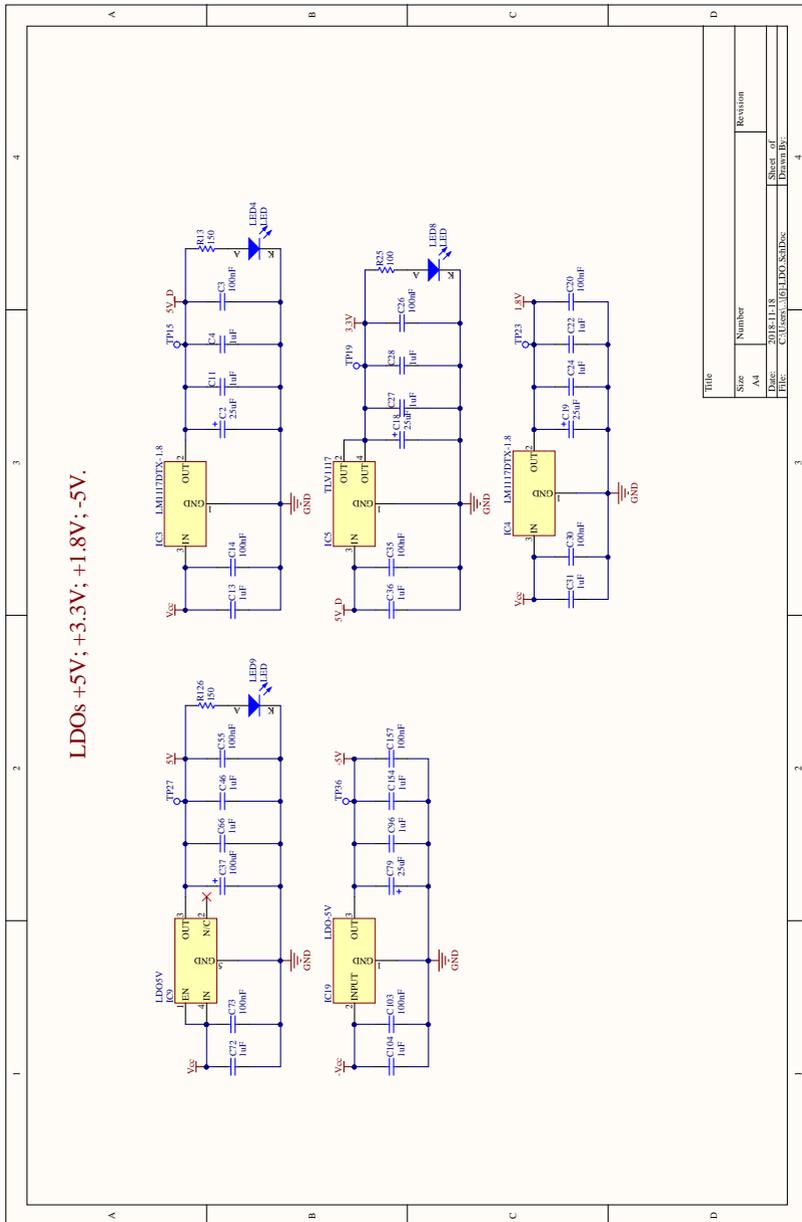


Figura C.6 – Esquemático dos reguladores de alimentação dos circuitos integrados da placa.

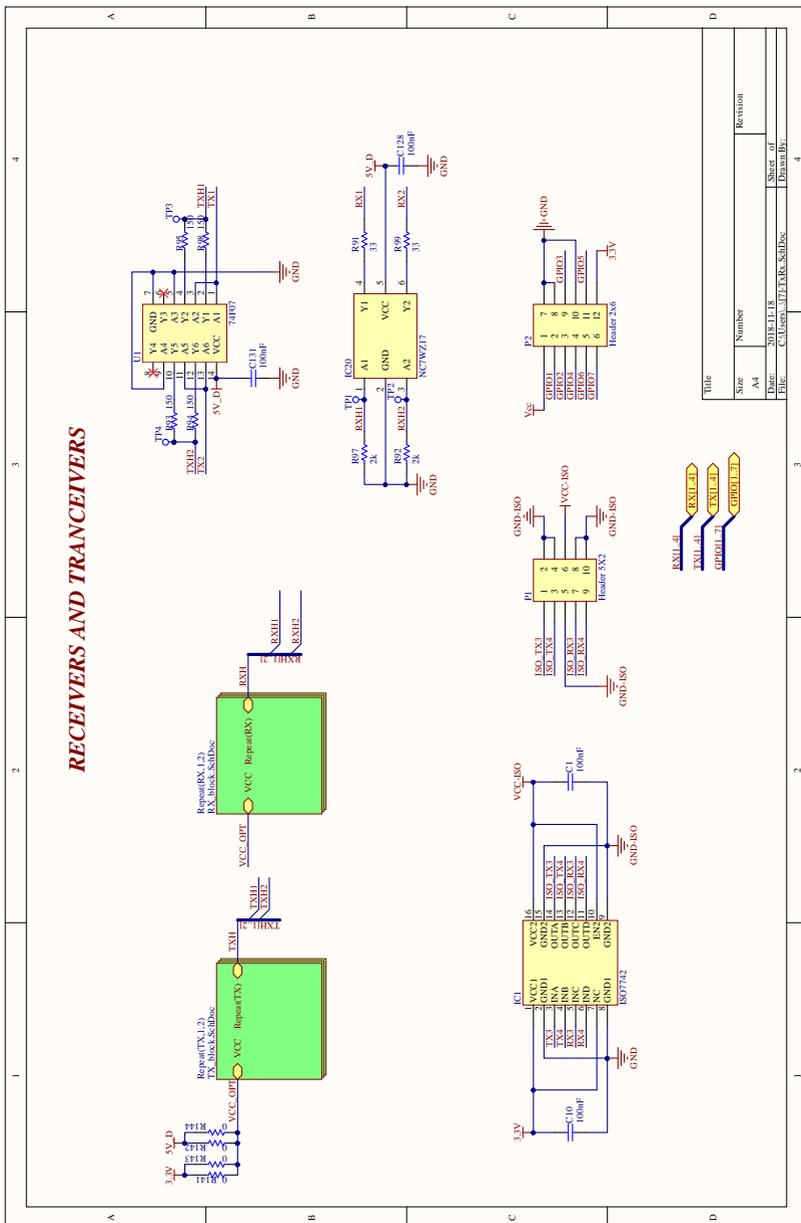


Figura C.9 – Esquemático dos transmissores e receptores da placa.

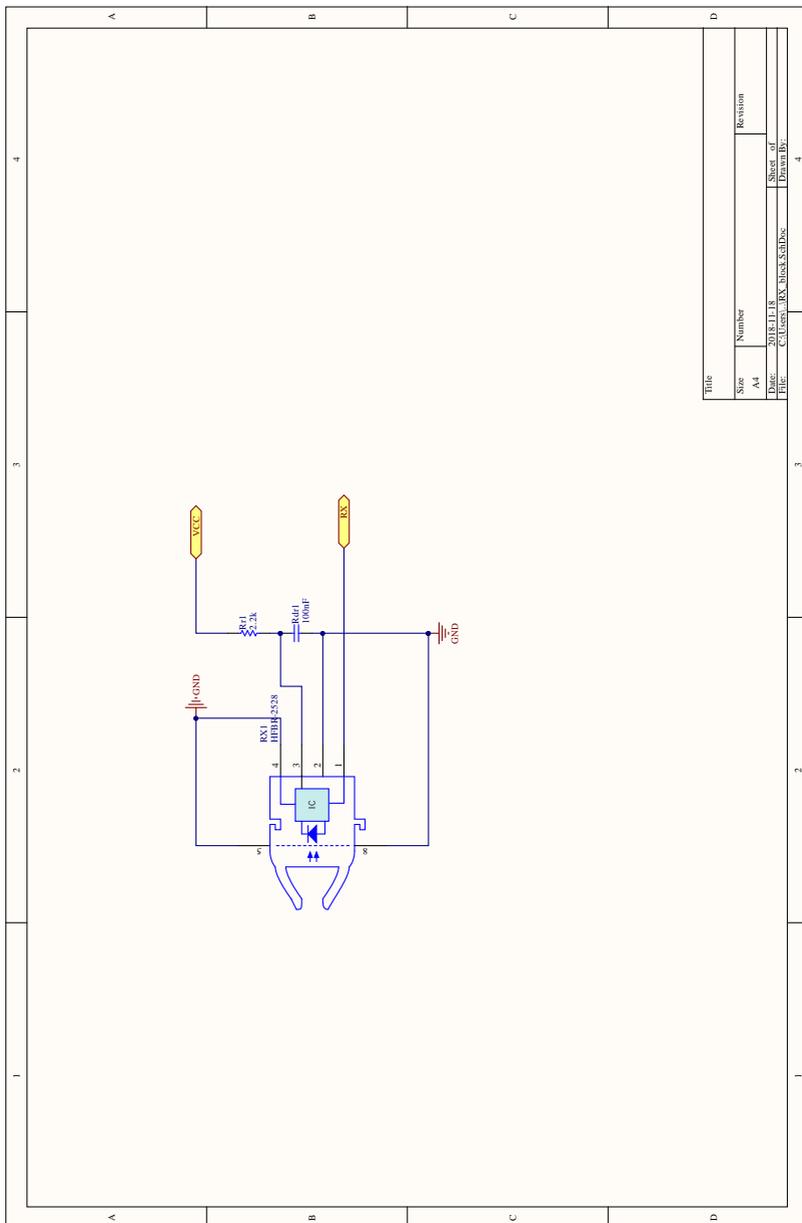


Figura C.10 – Esquemático do receptor óptico.

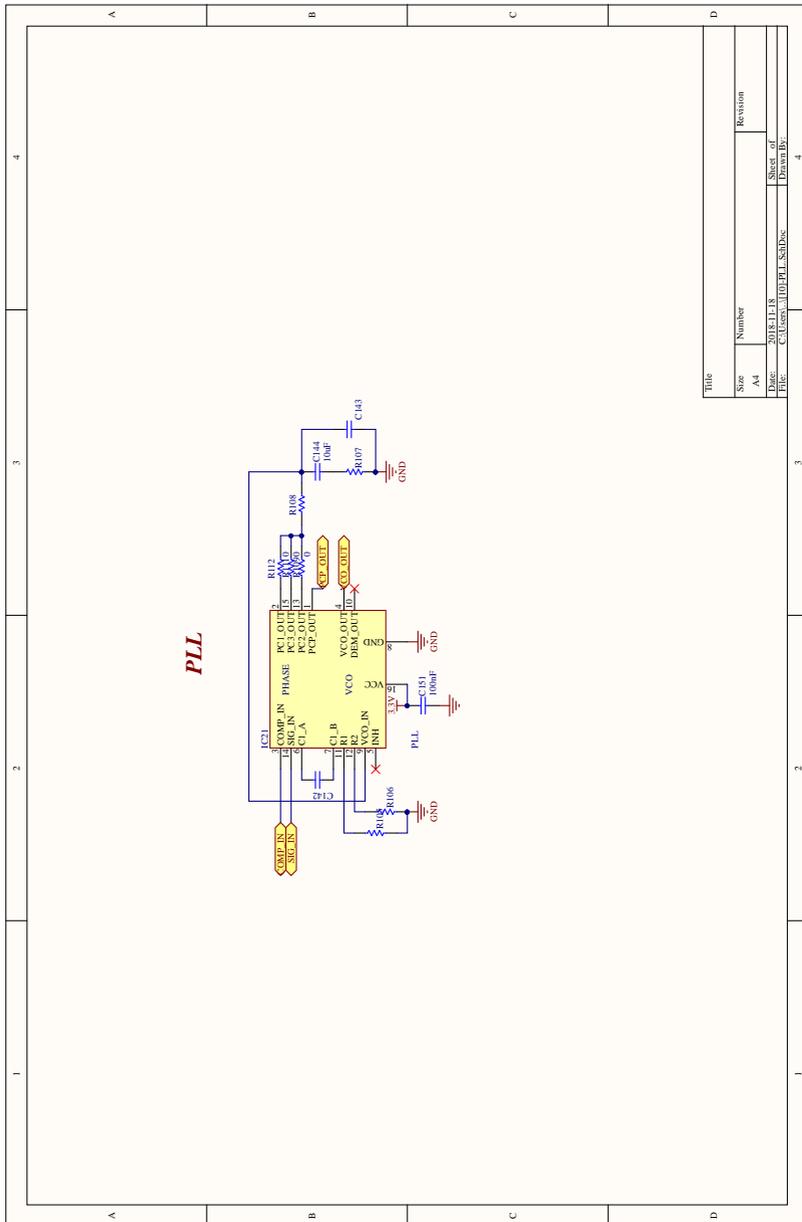


Figura C.12 – Layout da placa de circuito impresso.

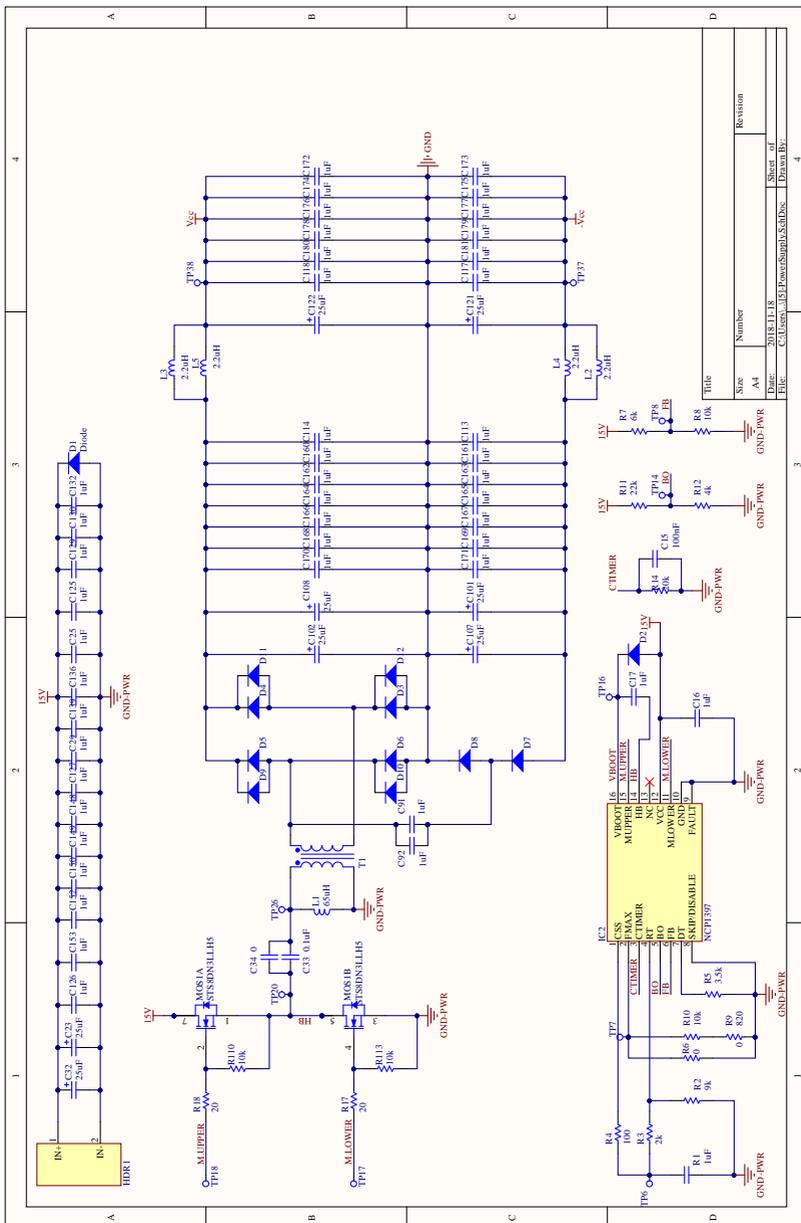


Figura C.13 – Esquemático do conversor série-ressonante para alimentação da placa.

APÊNDICE D – Placa de Circuito Impresso

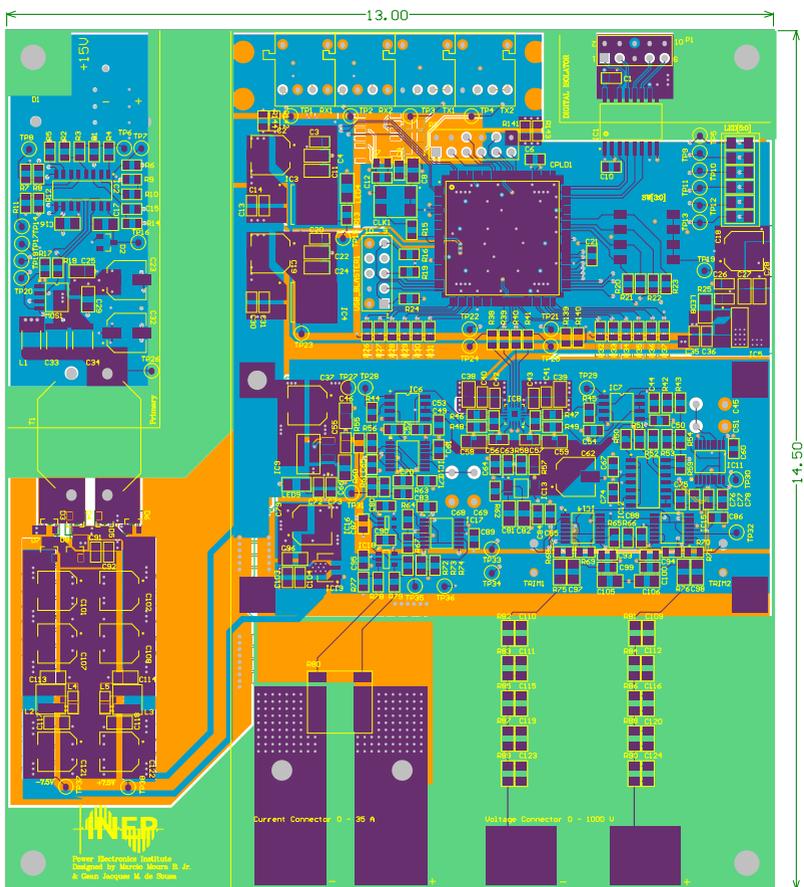


Figura D.1 – Layout da placa de circuito impresso.

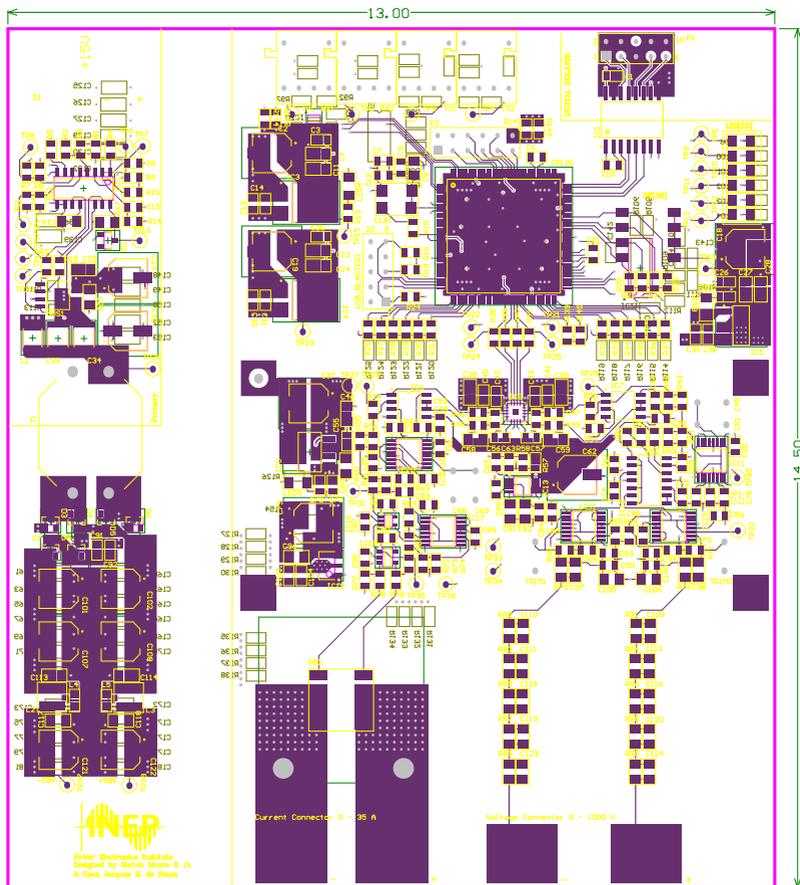


Figura D.2 – Layer L1 (top) da placa de circuito impresso.

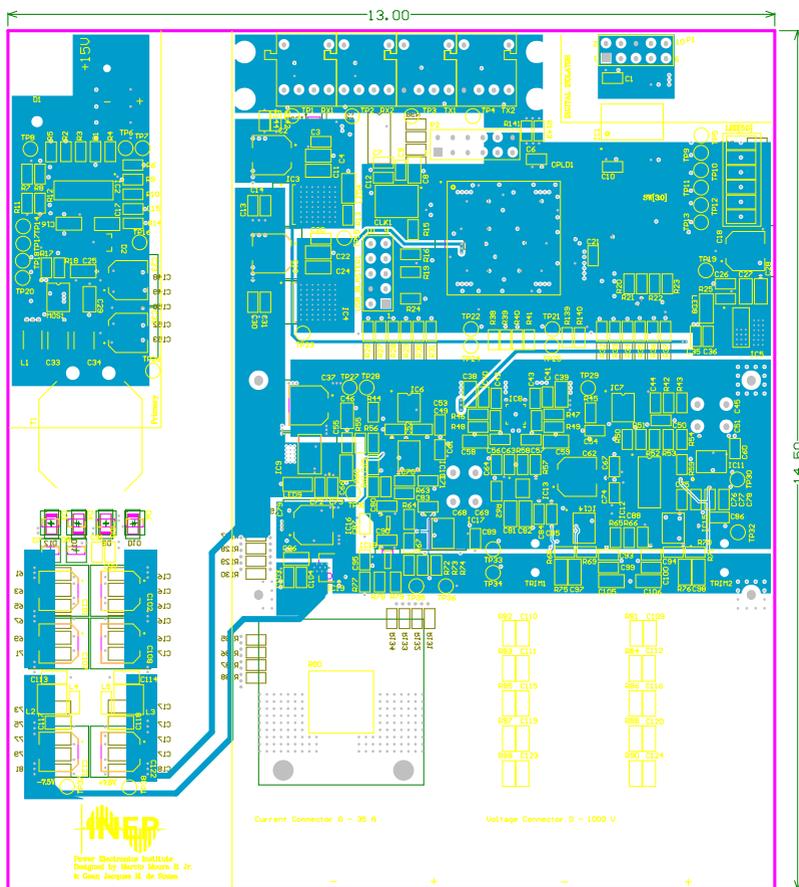


Figura D.4 – Layer L3 da placa de circuito impresso.

