



UNIVERSIDADE ESTADUAL PAULISTA
“Júlio de Mesquita Filho”

Faculdade de Engenharia
Campus de Ilha Solteira

Frank Alberto Ibarra Hernández

Simulação em Tempo Real de Sistemas de Distribuição de
Energia Elétrica Utilizando-se Estruturas com Descrição de
Hardware em Software

Ilha Solteira
2015

Frank Alberto Ibarra Hernández

**Simulação em Tempo Real de Sistemas de Distribuição de
Energia Elétrica Utilizando-se Estruturas com Descrição de
Hardware em Software**

Tese apresentada à Faculdade de
Engenharia de Ilha Solteira – FEIS/UNESP
– como parte dos requisitos para a obtenção
do título de Doutor em Engenharia Elétrica.

Área de concentração: Automação.

Prof. Dr. Carlos Alberto Canesin

Orientador

**Ilha Solteira
2015**

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

- I12s Ibarra Hernández, Frank Alberto.
Simulação em tempo real de sistemas de distribuição de energia elétrica utilizando-se estruturas com descrição de hardware em software / Frank Alberto Ibarra Hernández. -- Ilha Solteira: [s.n.], 2015
277 f. : il.
- Tese (doutorado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2015
- Orientador: Carlos Alberto Canesin
Inclui bibliografia
1. Alimentadores de distribuição. 2. Dispositivos FPGA. 3. Redes elétricas do futuro. 4. Simulação em tempo real. 5. VHDL. 6. VHDL-AMS.



UNIVERSIDADE ESTADUAL PAULISTA
CAMPUS DE ILHA SOLTEIRA
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA

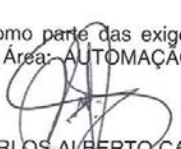
CERTIFICADO DE APROVAÇÃO

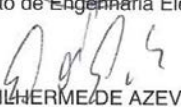
TÍTULO: Simulação em tempo real de sistemas de distribuição de energia elétrica utilizando-se estruturas com descrição de hardware em software

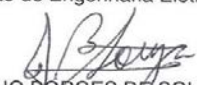
AUTOR: FRANK ALBERTO IBARRA HERNANDEZ

ORIENTADOR: Prof. Dr. CARLOS ALBERTO CANESIN

Aprovado como parte das exigências para obtenção do Título de DOUTOR EM ENGENHARIA ELÉTRICA, Área: AUTOMAÇÃO, pela Comissão Examinadora:


Prof. Dr. CARLOS ALBERTO CANESIN
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira


Prof. Dr. GUILHERME DE AZEVEDO E MELO
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira


Prof. Dr. JÚLIO BORGES DE SOUZA
Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira


Prof. Dr. RUBEN BARROS GODOY
Departamento de Engenharia Elétrica / Universidade Federal de Mato Grosso do Sul


Prof. Dr. LUIGI GALOTTO JUNIOR
Departamento de Engenharia Elétrica / Universidade Federal de Mato Grosso do Sul

Data da realização: 16 de junho de 2015.

Ao meu pai **Francisco Alberto Ibarra Scharberg** e minha mãe **Ant3nia Hern3ndez de Ibarra**, que me educaram com amor e deram a minha pr3pria exist4ncia, exemplos de vida fundamentais para o meu crescimento pessoal e profissional.

Agradecimentos

Aquilo que é acreditado é criado, porque nossa mente é criadora. Neste contexto, aquilo que é decretado é executado, porque a palavra tem poder e também é criadora. Logo, quando criamos nossos projetos em nossa vida, nós percebemos que a importância das coisas que realizamos em nossa existência é medida pelo grau de dificuldade que é gerado, como reação contrária. Neste âmbito, nós podemos valorizar todas as nossas conquistas. Por isto, como consequência do desenvolvimento deste trabalho, desejo agradecer:

Primeiramente ao meu orientador Prof. Dr. Carlos Alberto Canesin pela disposição, profissionalismo e por não medir esforços para disponibilizar e adquirir itens fundamentais para o êxito da pesquisa. Ainda pela motivação constante e pela produção de artigos científicos de qualidade.

Aos professores doutores Júlio Borges de Souza, Guilherme de Azevedo e Melo, Ruben Barros Godoy e Luigi Galotto Junior pela participação na banca examinadora da tese de doutorado, juntamente com suas contribuições e sugestões para o aprimoramento deste trabalho.

Aos professores Júlio Borges de Souza, Luis Carlos Origa de Oliveira, Laurence Duarte Colvara e Dionízio Paschoareli Júnior pelos ensinamentos adquiridos em sala de aula, pela amizade e disponibilidade que sempre demonstraram.

Aos meus colegas (e ex-colegas) dos Laboratórios de Eletrônica de Potência – LEP e de Qualidade da Energia Elétrica - LQEE, José Carlos, Moacyr, Luciano, Leonardo, Danusia, Marcos, Luis, Tarcísio, Cindy, João Carlos, Marcus e Rodrigo, pela amizade e colaboração em várias ocasiões.

Ao programa de pós-graduação em Engenharia Elétrica da UNESP, Campus de Ilha Solteira, pela oportunidade de estudar e concluir o curso de doutorado. À CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, ao CNPQ - Conselho Nacional de Desenvolvimento Científico e Tecnológico, à FEPISA - Fundação de Ensino, Pesquisa e Extensão de Ilha Solteira e à FAPESP - Fundação de Amparo à Pesquisa do Estado de São Paulo, pelo apoio financeiro concedido, possibilitando o desenvolvimento deste trabalho e permitindo a participação em congressos.

À toda a equipe do laboratório de investigação, pesquisa e demonstração de redes elétricas inteligentes (*SGDRIL*) da *Washington State University*, pela amizade e colaboração no trabalho com um *RTDS*TM – *Real Time Digital Simulator*, durante o ano de 2013.

Ao meu pai Francisco Alberto Ibarra Scharberg pelo incentivo incessante para a culminação do meu doutorado, pela sua contribuição com suas valiosas ideias cada vez que eu precisei, pela sua visão abrangente da vida, por todo seu grande amor e pelo seu incomensurável suporte em todos os aspectos que eu necessitei.

À minha mãe Antónia Hernández de Ibarra pelos ensinamentos de vida, o apoio e força constante, o amor recebido, embora longe, mas sempre perto do coração. Às minhas irmãs Hilda Patrícia e Maria Cristina, pelo apoio desde sempre e pelo carinho durante toda a nossa vida em conjunto. Pensar em vocês sempre me motivou a continuar com meu trabalho.

Ao meu cunhado Gustavo António Gómez Palácio pela valiosa companhia aos meus pais durante a minha longa ausência, o grande amor a minha irmã e o seu importantíssimo suporte cada vez que eu necessitei.

A todos e a cada uma das pessoas que fizeram uma contribuição para que este projeto fosse realizado com sucesso. Se estiver esquecendo alguns, peço mil desculpas.

A Deus pela vida, pela saúde e por colocar em meu caminho todas as condições necessárias para o meu desenvolvimento pessoal e profissional.

¡Muchas Gracias!

“Se qualquer estudante vier para mim e disser que quer ser útil à humanidade e entrar na carreira de pesquisador para aliviar o sofrimento humano, eu lhe aconselho que vá à caridade no seu lugar. A pesquisa requer egoístas reais que procuram o seu próprio prazer e satisfação, mas acham a solução dos enigmas da natureza”

Albert Szent-Györgyi

RESUMO

Esta tese de doutorado se baseia na necessidade atual e tendência mundial da busca por tornar mais inteligentes os sistemas de distribuição de energia elétrica, por isso, o objetivo geral deste trabalho é desenvolver uma Arquitetura de Simulação em Tempo Real e Controle (ASTR&C) para alimentadores elétricos de distribuição, com o intuito de analisar a qualidade da energia e melhorar as ações de controle nos sistemas de distribuição, procurando assim aumentar a confiabilidade e sustentabilidade do sistema de potência. A ASTR&C utiliza uma plataforma VHDL-AMS como interface gráfica do usuário (*Graphical user interface - GUI*) para desenvolver a simulação do sistema elétrico e a linguagem VHDL (*Very High Speed Integrated Circuit Description Language*) para o desenvolvimento do sistema de gerenciamento e controle da rede de distribuição (*Distribution Management System and Control - DMS&C*), através de um dispositivo FPGA (*Field Programmable Gate Array*). Ambas as linguagens de descrição de hardware VHDL e VHDL-AMS (*VHDL analog and mixed-signal*), juntamente com as informações do sistema elétrico de distribuição, tornam possível a simulação em tempo real e controle de alimentadores de distribuição de energia elétrica. A GUI na plataforma VHDL-AMS, além de executar a simulação do sistema elétrico de distribuição, envolve dois processos: 1) Importação de todos os parâmetros do sistema de distribuição real, a partir de um arquivo de texto, possibilitando a alteração de quaisquer dados deste alimentador de distribuição em ambiente VHDL-AMS e 2) Envio dos dados de controle necessários para o dispositivo FPGA. O DMS&C desenvolvido está focado no gerenciamento do perfil de tensão do alimentador admitido como estudo de caso, realizado através de um dispositivo FPGA, o qual dispõe como prioridade o controle do regulador de tensão do sistema de distribuição, com base na comutação de TAP do mesmo. Neste contexto, o DMS&C foi desenvolvido para propor ao controlador real do sistema de distribuição ações de regulação do nível de tensão do alimentador, considerando-se as características de um alimentador real de distribuição, com cargas concentradas e com redução da rede, para constituir o estudo de caso desta tese. Destaca-se como contribuição principal deste trabalho de tese, a apresentação de uma nova arquitetura de simulação em “tempo real” e controle para sistemas elétricos de distribuição, como uma nova ferramenta voltada à avaliação da qualidade da energia elétrica e ao melhoramento das ações de controle nos alimentadores de distribuição, utilizando-se as linguagens de descrição de hardware VHDL e VHDL-AMS. Finalmente, observa-se que, ainda, a ASTR&C não se encontra totalmente desenvolvida e

integrada no dispositivo FPGA, ficando tal desenvolvimento como proposta de continuidade para o trabalho.

Palavras-chave: Alimentadores de distribuição. Dispositivos FPGA. Qualidade da energia elétrica. Redes elétricas do futuro. Simulação em tempo real. VHDL. VHDL-AMS.

ABSTRACT

This doctoral thesis is based on current need and global trend in the search for making smarter electric power distribution systems. For this reason, the objective of this work is to develop a Real-Time Simulation and Control (RTSC) architecture of electrical distribution feeders, in order to analyze power quality and improve the control actions in distribution systems, to increase power system reliability, and sustainability. The RTSC architecture uses VHDL-AMS platform as graphical user interface (GUI) to develop the simulation of the electrical system and VHDL (*Very High Speed Integrated Circuit Description Language*) language for developing the Distribution Management System and Control (DMS&C) through a FPGA device. Both VHDL and VHDL-AMS (*VHDL analog and mixed-signal*) hardware description languages along with electric distribution system information make possible the real-time simulation and control for electrical distribution feeders. The GUI in VHDL-AMS platform, which, besides running the simulation of the electrical distribution system, involves two processes: 1) Import of all parameters of the distribution system from a text file, making it possible to change any data of this distribution feeder into a VHDL-AMS environment, and 2) Sending necessary control data to the FPGA device. The developed DMS&C is focused on voltage profile management of admitted feeder as a case study, performed through a FPGA device, which provides as priority control of the distribution system voltage regulator, based on the voltage regulator TAP switching. In this context, DMS&C was developed to propose feeder voltage level regulation actions to the distribution system real controller, considering the real feeder characteristics, with concentrated loads and network reduction, for constituting the case study of this thesis. It stands out as the main contribution of this thesis work, the presentation of a novel real-time simulation and control architecture for electrical distribution systems, as a new tool focused on evaluating the power quality and improvement of control actions in distribution feeders, using both VHDL and VHDL-AMS hardware description languages. Finally, it stands out, however, that the ASTR&C is not yet fully developed and integrated into the FPGA device, getting such development as proposed continuing work.

Key words: Distribution feeders. Future Electrical Grids. FPGA devices. Real-Time Simulation, VHDL. VHDL-AMS.

Lista de Figuras

Figura 1-	Fluxograma geral para o desenvolvimento da pesquisa proposta.	26
Figura 2-	Esquema geral da arquitetura de simulação em “tempo real” e controle para alimentadores de distribuição de energia elétrica.	27
Figura 3-	Simulação digital em tempo real de um sistema elétrico utilizando um RTDS comercial (RTDS®).	29
Figura 4-	Sistema de teste IEEE de 13 nós modelado em VHDL-AMS.	40
Figura 5-	Tensões trifásicas na saída do regulador de tensão (tensões de fase).	45
Figura 6-	Correntes de linha que fluem através do regulador de tensão.	46
Figura 7-	Tensões trifásicas no nó 75 (tensões de fase).	46
Figura 8-	Tensões trifásicas no nó 71 (tensões de fase).	46
Figura 9-	Utilização do SystemVision™ para a implementação da linguagem VHDL-AMS.	48
Figura 10-	Sistema de múltiplas microrredes baseado no conceito CERTS.	51
Figura 11-	Alimentador de uma rede de distribuição primária de 14,4 kV (caso de Estudo).	53
Figura 12-	Alimentador de uma rede de distribuição primária de 14,4 kV (com as impedâncias das linhas somadas).	54
Figura 13-	Diagrama unifilar de um alimentador de uma rede de distribuição primária de 14,4 kV.	55
Figura 14-	Projeto para a modelação e simulação em ambiente VHDL-AMS do alimentador da figura 2.10.	56
Figura 15-	Interface para a importação de dados na rede (mudança de parâmetros em ambiente VHDL-AMS).	58
Figura 16-	Potência ativa consumida/gerada pela microrrede 1 no sistema de múltiplas microrredes baseado no conceito CERTS.	61
Figura 17-	Potência ativa consumida/gerada pela microrrede 2 no sistema de múltiplas microrredes baseado no conceito CERTS.	61
Figura 18-	Potência ativa consumida/gerada pela microrrede 3 no sistema de múltiplas microrredes baseado no conceito CERTS.	62
Figura 19-	Porcentagem de regulação para as cinco cargas mais críticas do alimentador (com maior regulação de tensão).	65
Figura 20-	Perfil de tensão do alimentador baseados nos testes realizados em VHDL-AMS.	66
Figura 21-	Regulação do alimentador baseados nos testes realizados em VHDL-AMS.	66
Figura 22-	Análise dos transitórios das tensões e correntes da subestação (de 0 até 15 milissegundos).	67
Figura 23-	Simulação em 5 segundos das tensões e correntes da subestação.	68
Figura 24-	Simulação de 500 até 525 milissegundos das tensões e correntes da subestação.	68
Figura 25-	Simulação de 500 até 600 milissegundos das tensões e correntes da subestação.	69
Figura 26-	Alimentador de 14,4 kV com marcação das linhas e cargas a simular para a análise comparativa.	71
Figura 27-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-	72

	X539727_1605338.	
Figura 28-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X539727_1605338.	72
Figura 29-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X28676063_4782056.	73
Figura 30-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X28676063_4782056.	73
Figura 31-	Simulação em ambiente Orcad/Pspice do regulador de tensão.	74
Figura 32-	Simulação em ambiente VHDL-AMS do regulador de tensão.	74
Figura 33-	Estrutura do sistema de gerenciamento da distribuição e controle (DMS&C), para ser implementado através de um dispositivo FPGA.	87
Figura 34-	Algoritmo para o sistema de gerenciamento da distribuição e controle (DMS&C) feito pelo FPGA.	90
Figura 35-	Placa de desenvolvimento ZedBoard Zynq™-7000.	93
Figura 36-	Ambiente de projeto em VHDL.	95
Figura 37-	Ambiente de projeto do <i>Vivado Design Suite</i> .	97
Figura 38-	Configurações iniciais do projeto no <i>Vivado Design Suite</i> .	98
Figura 39-	Seleção da placa ZedBoard no ambiente <i>Vivado Design Suite</i> .	100
Figura 40-	Fluxograma geral do desenvolvimento do DMS&C.	100
Figura 41-	Fluxo de projeto no dispositivo FPGA (utilizando a ferramenta <i>Vivado Design Suite</i>).	101
Figura 42-	Sequência da mensagem trafegada do PC para o SoC Zynq 7000.	106
Figura 43-	Diagrama de blocos do projeto do DMS&C.	108
Figura 44-	Diagrama demonstrativo do bloco FPGA.	111
Figura 45-	Hardware resultante do código VHDL (Esquema Simplificado).	112
Figura 46-	Resultado de simulação da descrição VHDL do componente “Regulacao” (algoritmo de controle da figura 3.2).	114
Figura 47-	Ilustração conceitual da ASTR&C, para um alimentador elétrico de distribuição, utilizando as linguagens VHDL e VHDL-AMS.	118
Figura 48-	Esquema informático da Arquitetura de Simulação em “Tempo Real” e Controle Desenvolvida	119
Figura 49-	Estágios do projeto com a utilização da interface gráfica do usuário (GUI) em plataforma SystemVision™ (empregando a ferramenta “Experiment Manager”).	120
Figura 50-	Principais configurações iniciais para a simulação do projeto do alimentador utilizando a ferramenta SystemVision™.	121
Figura 51-	Resultados das simulações em VHDL-AMS (tensões na subestação e correntes na carga não-linear trifásica desequilibrada conectada em triângulo), fazendo uma única simulação do sistema.	122
Figura 52-	Utilização da ferramenta “Experiment Manager” na GUI em plataforma SystemVision™, utilizando a Linguagem VHDL-AMS.	122
Figura 53-	Especificação da folha eletrônica de cálculo derivada da utilização da ferramenta “Experiment Manager”.	123
Figura 54-	Folha eletrônica derivada da utilização da ferramenta “Experiment Manager”.	124
Figura 55-	Especificação dos parâmetros (grandezas que desejam-se mudar em ambiente VHDL-AMS).	125
Figura 56-	Especificação das medições (resultados das simulações automáticas do	125

	alimentador em VHDL-AMS que desejam-se armazenar na folha eletrônica de cálculo).	
Figura 57-	Andamento das simulações automáticas do alimentador em VHDL-AMS (depois de estabelecer os valores numéricos dos parâmetros especificados).	126
Figura 58-	Resultados parciais das simulações automáticas em plataforma SystemVision™ (Especificamente na simulação número 4 de 17).	126
Figura 59-	Resultados finais das simulações automáticas em VHDL-AMS (Especificamente na simulação número 17 de 17).	127
Figura 60-	Apresentação das principais estatísticas das simulações em VHDL-AMS, utilizando a GUI em plataforma SystemVision™.	127
Figura 61-	Arquivo de texto que constitui os dados de entrada no dispositivo FPGA.	128
Figura 62-	Escolha do arquivo de texto com os resultados das simulações em VHDL-AMS.	129
Figura 63-	Montagem e funcionamento do sistema de gerenciamento da distribuição e controle (DMS&C), utilizando o dispositivo FPGA Artix-7 e o processador ARM Cortex-A9.	129
Figura 64-	Placa de desenvolvimento ZedBoard Zynq™-7000 em funcionamento.	130
Figura 65-	Execução do DMS&C para o caso no qual dá-se prioridade à diminuição do TAP na fase A, no regulador de tensão (RT), para aumentar a tensão a jusante do RT.	133
Figura 66-	Possibilidades que podem ser geradas pelo sistema de gerenciamento da distribuição e controle (DMS&C).	134
Figura 67-	Simulações em ambiente VHDL-AMS para a carga CLTPEQE-XCLTPEQE539727 no terceiro cenário de simulação com operação normal do alimentador.	136
Figura 68-	Simulações em ambiente VHDL-AMS para a carga CLTPEQE-XCLTPEQE539727 no terceiro cenário de simulação com uma condição anômala do alimentador.	137
Figura 69-	Simulações em ambiente VHDL-AMS para a carga CLTPEQE-XCLTPEQE539727 para o terceiro cenário de simulação.	137
Figura 70-	Execução do DMS&C para o primeiro cenário de simulação.	138
Figura 71-	Execução do DMS&C para o segundo cenário de simulação.	139
Figura 72-	Execução do DMS&C para o terceiro cenário de simulação.	139
Figura 73-	Execução do DMS&C para o quarto cenário de simulação.	140
Figura 74-	Execução do DMS&C para o quinto cenário de simulação.	140
Figura 75-	Execução do DMS&C para o sexto cenário de simulação.	141
Figura 76-	Execução do DMS&C para o sétimo cenário de simulação.	141
Figura 77-	Execução do DMS&C para o oitavo cenário de simulação.	142
Figura 78-	Execução do DMS&C para o nono cenário de simulação.	142
Figura 79-	Execução do DMS&C para o décimo cenário de simulação	143
Figura 80-	Execução do DMS&C para o décimo primeiro cenário de simulação.	143
Figura 81-	Execução do DMS&C para o décimo segundo cenário de simulação.	144
Figura 82-	Execução do DMS&C para o décimo terceiro cenário de simulação.	144
Figura 83-	Execução do DMS&C quando a placa ZedBoard Zynq™-7000 está desconectada ou desligada.	145
Figura 84-	Execução do DMS&C para o décimo quarto cenário de simulação.	145
Figura 85-	Melhor condição de regulação de tensão para a comutação simultânea nas	146

	três fases do TAP no RT.	
Figura 86-	Pior condição de regulação de tensão para a comutação simultânea nas três fases do TAP no RT.	147
Figura 87-	Laboratório de investigação, pesquisa e demonstração de redes elétricas inteligentes - SGDRIL.	168
Figura 88-	Simulação em tempo real de um sistema fotovoltaico numa micro-rede ilhada.	174
Figura 89-	Potência de cada micro fonte (azul) [kW] e da carga sensível (preta).	174
Figura 90-	Descarregamento da bateria de cada micro fonte PV e Tensão da Bateria.	175
Figura 91-	Corrente de descarregamento da Bateria.	175
Figura 92-	Diagrama sniffer da modelação e implementação do sistema multi-agentes.	177
Figura 93-	Conceito de operações para o Cenário Modelado para o Gerenciamento da Potência Ativa em Várias Micro-redes.	179
Figura 94-	Número de teses, dissertações, artigos e patentes analisadas para a primeira visão global do estado da arte segundo os anos de publicação.	181
Figura 95-	Perguntas-respostas classificadas segundo os alvos da revisão global do estado da arte.	182
Figura 96-	Perguntas-respostas classificadas segundo as palavras-chave utilizadas.	182
Figura 97-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X2919791_1329777.	188
Figura 98-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X2919791_1329777.	188
Figura 99-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X1525339_117180.	189
Figura 100-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X1525339_117180.	189
Figura 101-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X4781628_28676045.	190
Figura 102-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X4781628_28676045.	190
Figura 103-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X2047531_28675987.	191
Figura 104-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X2047531_28675987.	191
Figura 105-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X28675721_2019128.	188
Figura 106-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X28675721_2019128.	192
Figura 107-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X28676197_588470.	193
Figura 108-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X28676197_588470.	193
Figura 109-	Simulação em ambiente Orcad/Pspice da LINHA_PRI-X1527172_1525345.	194
Figura 110-	Simulação em ambiente VHDL-AMS da LINHA_PRI-X1527172_1525345.	194
Figura 111-	Simulação em ambiente Orcad/Pspice da carga CCTPEQT-XCCTPEQT2220655.	195
Figura 112-	Simulação em ambiente VHDL-AMS da carga CCTPEQT-	195

	XCCTPEQT2220655.					
Figura 113-	Simulação em ambiente XCLTPEQE512943.	Orcad/Pspice	da	carga	CLTPEQE-	196
Figura 114-	Simulação em ambiente XCLTPEQE512943.	VHDL-AMS	da	carga	CLTPEQE-	196
Figura 115-	Simulação em ambiente XCLTPEQE539727.	VHDL-AMS	da	carga	CLTPEQE-	197
Figura 116-	Simulação em ambiente XCLTPEQE588470.	Orcad/Pspice	da	carga	CLTPEQE-	198
Figura 117-	Simulação em ambiente XCLTPEQE588470.	VHDL-AMS	da	carga	CLTPEQE-	198
Figura 118-	Simulação em ambiente XCLTPEQE1525339.	VHDL-AMS	da	carga	CLTPEQE-	199
Figura 119-	Simulação em ambiente XCLTPEQE1525345.	Orcad/Pspice	da	carga	CLTPEQE-	200
Figura 120-	Simulação em ambiente XCLTPEQE1525345.	VHDL-AMS	da	carga	CLTPEQE-	200
Figura 121-	Simulação em ambiente XCLTPEQE1624998.	Orcad/Pspice	da	carga	CLTPEQE-	201
Figura 122-	Simulação em ambiente XCLTPEQE1624998.	VHDL-AMS	da	carga	CLTPEQE-	201
Figura 123-	Simulação em ambiente XCLTPEQE1994739.	Orcad/Pspice	da	carga	CLTPEQE-	202
Figura 124-	Simulação em ambiente XCLTPEQE1994739.	VHDL-AMS	da	carga	CLTPEQE-	202
Figura 125-	Simulação em ambiente XCLTPEQE2019128.	Orcad/Pspice	da	carga	CLTPEQE-	203
Figura 126-	Simulação em ambiente XCLTPEQE2019128.	VHDL-AMS	da	carga	CLTPEQE-	203
Figura 127-	Simulação em ambiente XCLTPEQE2047531.	VHDL-AMS	da	carga	CLTPEQE-	204
Figura 128-	Simulação em ambiente XCLTPEQE2919791.	Orcad/Pspice	da	carga	CLTPEQE-	205
Figura 129-	Simulação em ambiente XCLTPEQE2919791.	VHDL-AMS	da	carga	CLTPEQE-	205
Figura 130-	Simulação em ambiente XCLTPEQE4781628.	Orcad/Pspice	da	carga	CLTPEQE-	206
Figura 131-	Simulação em ambiente XCLTPEQE4781628.	VHDL-AMS	da	carga	CLTPEQE-	206
Figura 132-	Simulação em ambiente XCNTDQT4782056.	Orcad/Pspice	da	carga	CNTDQT-	207
Figura 133-	Simulação em ambiente XCNTDQT4782056.	VHDL-AMS	da	carga	CNTDQT-	207

Lista de Tabelas

Tabela 1- Modelo da fonte de tensão monofásica.	41
Tabela 2- Modelo do transformador monofásico.	42
Tabela 3- Modelos de componentes elétricos usados várias vezes na modelação.	43
Tabela 4- Convenções para cada microrrede.	49
Tabela 5- Convenções para o sistema elétrico global.	50
Tabela 6- Características do módulo fotovoltaico de cada microrrede.	50
Tabela 7- Características do módulo fotovoltaico de cada microrrede calculados analiticamente com base nas linhas de tendência.	51
Tabela 8- Cargas totais nas microrredes na figura 10.	60
Tabela 9- Cargas totais no alimentador C na figura 10.	60
Tabela 10- Convenções para cada microrrede vs. Valores simulação 3.	63
Tabela 11- Convenções para o sistema elétrico global vs. Valores simulação 3.	63
Tabela 12- Estatísticas derivadas de uma simulação entre 500 e 600 milissegundos.	64
Tabela 13- Testes realizados em VHDL-AMS com mudanças de TAP no regulador de tensão para a carga com a pior regulação de tensão.	65
Tabela 14- Escolha do tempo de simulação para as mudanças de TAP'S.	67
Tabela 15- Análise quantitativa dos resultados das simulações em Orcad/Pspice versus VHDL-AMS.	75
Tabela 16- Comparação entre os resultados dos dois métodos de simulação feitos junto a VHDL-AMS.	78
Tabela 17- Faixa adequada de variação da tensão na carga (TL) em relação à tensão na subestação (TR).	89
Tabela 18- Características mais relevantes do ZYNQ SOC XC7Z020-CLG484.	92
Tabela 19- Sequência de bits resultante do algoritmo.	110
Tabela 20- Dados estatísticos estimados com o processo de implementação do DMS&C.	113
Tabela 21- Cenários de simulação em ambiente VHDL-AMS.	135
Tabela 22- Resultados do desenvolvimento da nova arquitetura de simulação em “tempo real” e controle para um alimentador de distribuição de energia elétrica.	148
Tabela 23- Resultados das simulações para o gerenciamento da potência ativa em várias micro-redes usando um sistema multi-agentes.	177
Tabela 24- Bibliotecas modelos da linguagem VHDL-AMS.	184
Tabela 25- Biblioteca dos sistemas de controle.	184
Tabela 26- Biblioteca dos sistemas digitais.	186
Tabela 27- Biblioteca dos sistemas elétricos.	187

Lista de Abreviaturas e Siglas

AMBA	Arquitetura Avançada de Barramento para Microcontrolador - “Advanced Microcontroller Bus Architecture”
ANEEL	Agência Nacional de Energia Elétrica.
ASIC	Aplicação Específica de Circuitos Integrados - “Application-Specific Integrated Circuit”
AC	Corrente Alternada – “Alternate Current”
DDR	Taxa dupla de transferência – “Double-Data-Rate”
DC	Corrente Contínua – “Direct Current” .
DER	Recursos de Energia Distribuída – “Distributed Energy Resources”
DERMS	Sistema de gerenciamento de recursos de energia distribuída – “Distributed Energy Resources Management System”
DSP	Processador Digital de Sinal – “Digital Signal Processor”
DMS	Sistema de Gerenciamento da Distribuição – “Distribution Management System”
CERTS	Consortium for Electric Reliability Technology Solutions
ELS	Rejeição de Carga de Emergência – “Emergency Load Shedding”
EMTP	Programa de Transitórios Eletromagnéticos – “Electro Magnetic Transients Program”
EPRI	Instituto de Pesquisa da Energia Elétrica – “Electric Power Research Institute”
FACTS	Sistema de Transmissão Flexível de Corrente Alternada - “Flexible Alternating Current Transmission System” .
FLISR	Ilhamento para Localização de Falhas e Restauração do Serviço – “Fault Location Isolation and Service Restoration”
FPA	Matriz Analógica Programável em Campo - “Field Programmable Analog Array” .
FPGA	Matriz de Portas Programáveis em Campo – “Field Programmable Gate Array” .
GDEE	Geração Distribuída de Energia Elétrica.
GUI	Interface Gráfica do Usuário – “Graphical User Interface” .
HDL	Linguagem de Descrição de Hardware – “Hardware Description Language” .
HIL	Teste de equipamento em malha fechada ou Interface entre o modelo da planta e o sistema integrado em teste – “Hardware-In-the-Loop” .
HMI	Interface Homem-Máquina – “Human-Machine Interface” .
IDE	Ambiente de projeto integrado – “Integrated Design Environment”
JADE	Java Agent DEvelopment Framework
LEP	Laboratório de Eletrônica de Potência do Departamento de Engenharia Elétrica - DEE, da Faculdade de Engenharia de Ilha Solteira – FE/IS.
MPPT	Rastreamento do ponto de máxima potência.
NEMA	Associação Nacional de Fabricantes Elétricos (EUA) – “National Electrical Manufacturers Association”
PLC	Controlador Lógico Programável – “Programmable logic controller” .
PV	Painel Fotovoltaico.
QEE	Qualidade da Energia Elétrica.
RMS	Valor Quadrático Médio – “Root Mean Square” .
RTL	Nível de transferência do registrador – “RTL- Register Transfer Level”.

RTDS	Simulador Digital em Tempo Real – “ <i>Real Time Digital Simulator</i> ”.
RTOS	Sistema Operacional de Tempo Real – “ <i>Real Time Operating System</i> ”
S-E-D (Sistema- Elétrico-de- Distribuição)	Rede ou alimentador de distribuição de energia elétrica primário (em média tensão) ou secundário (em baixa tensão), que pode vir de um alimentador real (baseado nos dados das características técnicas de um sistema elétrico existente) ou, somente é simulado (com dados fictícios), o qual pretende-se modelar e simular na linguagem VHDL-AMS.
SoC	Sistema em um Chip - “ <i>System on a chip</i> ”
VHDL	VHSIC mais HDL. Linguagem utilizada para descrever circuitos digitais.
VHDL-AMS	Extensão da linguagem de descrição de hardware digital VHDL, utilizada para descrever sistemas digitais, analógicos e de sinal misto.
VHSIC	Circuito Integrado de Alta Velocidade – “ <i>Very High Speed Integrated Circuit</i> ”.
VVO	Otimização Volt – Var

Sumário

Capítulo 1	INTRODUÇÃO GERAL	20
1.1	FORMULAÇÃO DO PROBLEMA DE PESQUISA	22
1.2	EXPERIMENTAÇÃO CIENTÍFICA RELACIONADA COM A MODELAÇÃO E SIMULAÇÃO EM TEMPO REAL, DERIVADAS DE ESTUDOS COM UM RTDS™ COMERCIAL	27
1.3	ORGANIZAÇÃO DO TRABALHO	34
Capítulo 2	MODELAÇÃO DE SISTEMAS DE DISTRIBUIÇÃO EM AMBIENTE VHDL-AMS	36
2.1	A LINGUAGEM VHDL-AMS	36
2.2	JUSTIFICATIVA PARA O USO DA LINGUAGEM VHDL-AMS	38
2.3	MODELAÇÃO E SIMULAÇÃO DE UM SISTEMA DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA UTILIZANDO VHDL-AMS: CASO DE ESTUDO BÁSICO	39
2.4	PROJETO DA ARQUITETURA DE SIMULAÇÃO EM “TEMPO REAL” E CONTROLE PARA ALIMENTADORES DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA, UTILIZANDO VHDL-AMS	47
2.5	CENÁRIOS DE SIMULAÇÃO PARA UM SISTEMA ELÉTRICO DE DISTRIBUIÇÃO ESTUDO DE CASO	48
2.6	INTERFACE PARA A IMPORTAÇÃO DE DADOS NO SISTEMA ELÉTRICO DE DISTRIBUIÇÃO EM AMBIENTE VHDL-AMS	57
2.7	RESULTADOS OBTIDOS NOS CENÁRIOS DE SIMULAÇÃO COM O USO DA INTERFACE PARA A IMPORTAÇÃO DE DADOS NO SISTEMA ELÉTRICO DE DISTRIBUIÇÃO EM AMBIENTE VHDL-AMS	59
2.8	CONCLUSÕES	78
Capítulo 3	SISTEMA DE GERENCIAMENTO E CONTROLE ATRAVÉS DE UM DISPOSITIVO FPGA	80
3.1	INTRODUÇÃO	80
3.2	SISTEMA DE GERENCIAMENTO E CONTROLE DA REDE DE DISTRIBUIÇÃO	81
3.3	ALGORITMO PARA O CONTROLE DA REGULAÇÃO DE TENSÃO FEITO PELO FPGA	87
3.4	MATRIZ DE PORTAS PROGRAMÁVEIS EM CAMPO – FPGA	91
3.5	VISÃO GERAL DO AMBIENTE DE PROJETO EM VHDL	95
3.6	CONTROLADOR DIGITAL PROGRAMÁVEL PARA O DMS&C	96
3.7	CONCLUSÕES	115
Capítulo 4	ARQUITETURA PROPOSTA PARA SIMULAÇÃO EM “TEMPO REAL” E CONTROLE PARA UM ALIMENTADOR DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA, UTILIZANDO VHDL-AMS e VHDL	117
4.1	INTRODUÇÃO	117
4.2	INTERFACE GRÁFICA DO USUÁRIO EM PLATAFORMA SYSTEMVISION™	119
4.3	CONCEITO INTEGRADO DA ARQUITETURA DE SIMULAÇÃO EM “TEMPO REAL” E CONTROLE PARA UM ALIMENTADOR DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA (RESULTADOS EXPERIMENTAIS DO DESENVOLVIMENTO)	149
4.4	CONCLUSÕES	

Capítulo 5 CONCLUSÕES GERAIS E TRABALHOS FUTUROS	150
5.1 CONCLUSÕES GERAIS	150
5.2 TRABALHOS FUTUROS	153
Referências	155
Apêndice A PRODUÇÃO DE ARTIGOS CIENTÍFICOS	166
A.1 PUBLICAÇÕES EM CONGRESSOS	166
A.2 ARTIGOS ACEITOS PARA PUBLICAÇÃO:	166
A.3 ARTIGO PRELIMINARMENTE ACEITO EM REVISTAS (JOURNALS):	166
Apêndice B DESENVOLVIMENTO COM RTDS EM PARCERIA COM A WSU – ESTADOS UNIDOS DE AMÉRICA DO NORTE	167
B.1. ESPECIFICAÇÕES TÉCNICAS RELEVANTES DO RTDS® UTILIZADO	168
B2. RESULTADOS EXPERIMENTAIS DA SIMULAÇÃO E MODELAÇÃO EM TEMPO REAL COM UM SIMULADOR DIGITAL EM TEMPO REAL COMERCIAL (RTDS®)	172
Apêndice C DADOS QUANTITATIVOS REFERENTES À REVISÃO GLOBAL DO ESTADO DA ARTE DA PESQUISA	180
C.1 METODOLOGIA	180
C.2 ESQUEMA DE REFERÊNCIA DA VISÃO GLOBAL DA PESQUISA PROPOSTA	180
C.3 ESTRATÉGIA DE ANÁLISE DO MATERIAL ENCONTRADO NA REVISÃO DO ESTADO DA ARTE	181
Apêndice D BIBLIOTECAS DA LINGUAGEM VHDL-AMS, SIMULAÇÕES DAS CARGAS E LINHAS NÃO CRÍTICAS DO ALIMENTADOR ESCOLHIDO, CÓDIGOS DE PROGRAMAÇÃO E, HARDWARE RESULTANTE NO DISPOSITIVO FPGA	183
D.1- BIBLIOTECAS MODELOS DA LINGUAGEM VHDL-AMS (ENCONTRADAS NO SOFTWARE SYSTEMVISION™)	184
D.2- SIMULAÇÕES DE 7 LINHAS E DAS 13 CARGAS TOTAIS DO SISTEMA DE DISTRIBUIÇÃO PRIMÁRIO A 14,4 KV	188
D.3- CÓDIGO VHDL-AMS DO ALIMENTADOR DE BT DE 26 NÓS (SISTEMA DE MÚLTIPLAS MICRO-REDES BASEADO NO CONCEITO CERTS)	208
D.4- CÓDIGO VHDL-AMS DO ALIMENTADOR DE MT DE 925 NÓS (ALIMENTADOR DE UMA REDE DE DISTRIBUIÇÃO PRIMARIA DE 14,4 KV)	214
D.5- CÓDIGO VHDL: BLOCO “Regulacao_Transf_v1_0.vhd”	232
D.6- CÓDIGO VHDL: BLOCO “Regulacao_Transf_v1_0_S00_AXI_TRASNf_parameterized0.vhd”	236
D.7- CÓDIGO VHDL: BLOCO “Regulacao.vhd”	248
D.8- CÓDIGO EM LINGUAGEM DE PROGRAMAÇÃO C: “main.c”	252
D.9- CÓDIGO EM MICROSOFT VISUAL BASIC: “mainModuleChk.bas”	257
D.10- HARDWARE RESULTANTE NO DISPOSITIVO FPGA (ESQUEMAS COMPLETOS)	265
Apêndice E CÓDIGO EM ORCAD/PSPICE ORIGINAL, USADO PARA O CASO DE ESTUDO NAS SIMULAÇÕES EM VHDL-AMS PARA UM ALIMENTADOR EM MÉDIA TENSÃO A 14,4 KV	268

INTRODUÇÃO GERAL

Na atualidade, nos segmentos residencial, comercial, industrial e rural tem-se observado um aumento na utilização de cargas elétricas não lineares, as quais geralmente são formadas por equipamentos eletroeletrônicos. Estes dispositivos eletroeletrônicos, em especial aqueles do domínio da eletrônica de potência, processadores e controladores eletrônicos de energia elétrica, estão cada vez mais presentes em todos os setores, especialmente no segmento industrial. No entanto, apesar de suas enormes vantagens técnicas operacionais pelo lado da carga, podem, como desvantagens, comprometer os indicadores de qualidade de energia pelo lado dos sistemas de distribuição de energia elétrica (OLIVEIRA et al., 2011).

Como resultado deste cenário no Brasil, tem-se gerado um aumento significativo do número de reclamações e pedidos de compensações financeiras em virtude de eventuais danos, tanto de produção, como físicos em máquinas e equipamentos, ocasionados por distúrbios no fornecimento de energia elétrica, principalmente nas áreas de concessão de empresas de distribuição das regiões Sul e Sudeste do país.

O fornecimento de energia elétrica é responsabilidade das empresas concessionárias de distribuição, porém estas empresas são responsáveis pelos possíveis efeitos que são derivados do panorama atual de crescimento das cargas não lineares. Este panorama atual nas redes de distribuição impõe uma elevada circulação de correntes harmônicas nestas redes, provocando distorções harmônicas nas tensões e perdas técnicas adicionais nos diferentes elementos do sistema elétrico, além de importantes implicações que podem levar equipamentos à operação ineficaz, errônea e/ou a uma falha no seu funcionamento, incluindo-se dispositivos de proteção e controle, de regulação e compensação (OLIVEIRA et al., 2011). Neste sentido, as empresas concessionárias precisam conhecer com exatidão as razões das possíveis perturbações que as cargas não lineares podem e/ou poderão produzir no sistema elétrico.

Adicionalmente, todos os agentes dos sistemas de energia elétrica têm-se preocupado pelo cumprimento dos índices de Qualidade da Energia Elétrica (QEE), abrangendo os segmentos de geração, transmissão e distribuição (MCGRANAGHAN, 2007). No segmento da distribuição, no qual estão conectadas uma enorme diversidade de cargas não lineares, as

empresas concessionárias têm a obrigação de investir na modelação e simulação destas cargas, o que abrange metodologias de desenvolvimento tecnológico e científico, medições elétricas em campo, e, finalmente, o desenvolvimento e implementação de ferramentas de simulações computacionais que permitam a análise da qualidade da energia elétrica nas redes de distribuição, considerando, por exemplo, a análise da propagação de harmônicos e os efeitos deste fenômeno no sistema elétrico.

Para estudar a propagação das distorções harmônicas, as ferramentas computacionais constituem-se hoje em dia em instrumentos indispensáveis para as concessionárias de distribuição de energia elétrica (OLIVEIRA et al., 2011). O emprego destas ferramentas computacionais permite a avaliação do desempenho de um sistema típico de distribuição e, por conseguinte, a qualificação e quantificação dos efeitos das harmônicas nos equipamentos conectados. Além disso, o uso de ferramentas computacionais fornece um suporte importante na análise técnica de pedidos de novas ligações, instalação de equipamentos de mitigação e verificação de atendimento aos limites estabelecidos pela legislação vigente.

Assim, a empresa concessionária, além de solicitar ao cliente uma análise interna de sua própria carga, precisa de ferramentas modernas de engenharia e conhecimento científico, para garantir uma acertada tomada de decisão, com o intuito de dar solução aos problemas apresentados nos sistemas de distribuição (MCGRANAGHAN, 2007). No país, segundo a ANEEL-Agência Nacional de Energia Elétrica (ANEEL, 2011), a qualidade da energia elétrica tem duas abordagens: qualidade do produto e qualidade do serviço prestado. A qualidade do produto refere-se à conformidade das tensões em regime permanente e às perturbações nas formas de onda das tensões; por outro lado, a qualidade do serviço faz referência à apuração dos indicadores de continuidade e dos tempos de atendimento a ocorrências emergenciais, assim como, aos problemas das redes elétricas que possam afetar as cargas e instalações dos consumidores conectados.

Pelo exposto, pode-se concluir que a implementação de ferramentas computacionais de simulações das redes elétricas de distribuição, tornou-se uma metodologia alternativa necessária para a atual (e futura) operação dos sistemas de distribuição de energia elétrica, também no país, para a análise da qualidade do produto. Entretanto, considerando-se os cenários futuros das redes elétricas inteligentes (**Smart Grids**), ou também, redes elétricas do futuro, para se conseguir obter tempos de respostas adequados e uma maior precisão sobre os estudos de qualidade de energia nos sistemas de energia elétrica, de forma técnica e econômica, a simulação em tempo real é uma opção viável (GREEN, R.C.; LINGFENG WANG ; ALAM, M., 2013).

Portanto, considerando a natureza atual (e futura) da operação dos sistemas de distribuição de energia elétrica, os simuladores em tempo real (simulação em fração de tempo de atuação do sistema: denominado tempo real) tornam-se uma alternativa tecnológica necessária para atender aos requisitos da operação. Nestes simuladores, a modelação dos dispositivos convencionais de um sistema de distribuição pode ser realizada, assim como a de dispositivos de regulação, compensação convencional passiva e mitigação harmônica estática (circuitos de eletrônica de potência), permitindo a simulação em tempo real e a interação com os sistemas de controle e gerenciamento externos (periféricos), admitindo-se um ambiente completo de simulação do tipo HIL (**Hardware in the Loop**).

Neste contexto, considerando a possível complexidade e tamanho dos modelos dos sistemas de distribuição, principalmente nos cenários futuros das microrredes com geradores distribuídos, torna-se possível a aplicação de simuladores que descrevem tais estruturas através de linguagens de programação específicas (VHDL-AMS, por exemplo), com o propósito de garantir precisão e simulação em tempo real.

Portanto, considerando-se questões estratégicas e de domínio das tecnologias dos dispositivos e aplicativos associados aos simuladores em tempo real para sistemas de distribuição de energia elétrica, esta tese se insere no estado da arte atual desta importante área de pesquisa associada aos sistemas elétricos de distribuição.

1.1 FORMULAÇÃO DO PROBLEMA DE PESQUISA

Dada a complexidade dos sistemas de energia elétrica atuais e principalmente, nos cenários futuros, os softwares simuladores de redes convencionais utilizados em estudos de planejamento, nada podem ou poderão contribuir para a operação segura destes sistemas elétricos e atendimento aos índices normatizados para a QEE. Desta forma, novas ferramentas têm surgido para análises nestes novos cenários, tais como os simuladores comerciais em tempo real de sistemas elétricos de potência, como o eMEGAsim, da OPAL-RT, ou o RTDSTM da empresa RTDS Technologies¹. Com tais aplicativos, com custos da ordem de centenas de milhares de dólares, é possível simular sistemas elétricos convencionais, incluindo os conversores estáticos de potência, em tempo real, dependendo apenas da capacidade de processamento e dispositivos microcontroladores inseridos, e, obviamente, da complexidade do sistema modelado.

¹ No Apêndice B, apresentam-se atividades realizadas com um RTDS comercial, desenvolvidas em parceria com o laboratório de investigação, pesquisa e demonstração de redes elétricas inteligentes (SGDRIL), pertencente à Washington State University (WSU).

Neste contexto, a simulação em tempo real² dos sistemas elétricos de distribuição representa um desafio atual, estratégico e de assimilação tecnológica. A presença de descontinuidades abruptas dadas pelos eventos de comutação impõe à simulação escalas de tempo reduzidas e com precisão nas respostas (LOK-FU et al., 2006). Além disso, a simulação digital em tempo real dos sistemas elétricos de potência requer consideráveis recursos computacionais devido a configurações de sistemas cada vez mais complexos, algoritmos de controle dos sistemas e maior frequência de comutação. Por isso, é prudente explorar vários recursos computacionais para aperfeiçoar o design de simuladores ou controladores para tais sistemas (GOYAL, S.; LEDWICH, G.; GHOSH, A., 2010).

Por outro lado, os transitórios eletromagnéticos apresentados frequentemente nos sistemas elétricos (YUAN; DINAHAHI, 2009) e nas máquinas elétricas rotativas (MATAR; IRAVANI, 2011; BACHIR et al., 2010; AUNG; DINAHAHI, 2011; REN et al., 2011; DUFOUR et al., 2007), as quais são os principais equipamentos de potência nas indústrias em geral, também podem ser simulados em tempo real, baseados em tecnologia FPGA (**Field Programmable Gate Array**).

No contexto dos simuladores digitais em tempo real, a tecnologia FPGA é particularmente adequada para o desenvolvimento dos controladores dos sistemas (**CHIL-Controller Hardware-In-the-Loop**) (REN et al., 2011). Os sistemas CHIL caracterizam-se pela atuação de um equipamento em malha fechada, nomeado controlador, que atua como uma interface entre o modelo da planta (sistema elétrico simulado) e o sistema integrado em teste (sistema elétrico real).

Na aplicação e utilização dos dispositivos FPGA, a linguagem de descrição de hardware mais comumente utilizada para programar um FPGA é a VHDL (VHSIC **Hardware Description Language**) (DE LOS ANGELES GOMEZ LOPEZ et al., 2013).

O estilo de design mais comumente utilizado para sintetizar modelos VHDL é o que pode ser chamado estilo de fluxo de dados (NAGASAMY et al., 1992). Um maior número de declarações simultâneas em VHDL e pequenos processos conectados através de sinais são usados para programar a funcionalidade desejada. A leitura e compreensão do fluxo de dados do código VHDL é uma tarefa difícil, pois desde as declarações concorrentes até os processos, não são executados na ordem em que são escritos. Utilizando o LabVIEW como uma linguagem de programação gráfica, o FPGA, por exemplo, é um bloco de uma função,

² É a ação de reproduzir o funcionamento de um sistema (neste caso: um sistema elétrico) através de um dispositivo, designado simulador, o qual interage com o mundo real através de sinais de entrada e saída, em um tempo que coincide com o tempo de atuação do sistema, denominado tempo real.

ou fluxo de dados, na qual a linguagem de programação gráfica é compilada no dispositivo FPGA (JOSIFOVSKA, 2003). As implementações FPGA são frequentemente modeladas como um diagrama de estados ou fluxograma para fins de visualização. Essa abordagem permite que especialistas ou engenheiros de teste possam configurar e visualizar sistemas complexos em hardware sem qualquer conhecimento de VHDL ou C, mantendo o poder oferecido pelo dispositivo FPGA (WASHINGTON; DOLMAN, 2010).

Tendo isso em consideração e sabendo que a linguagem de descrição VHDL não tem uma forma direta para modelar elementos de circuitos passivos e ativos das redes de distribuição de energia elétrica através de suas bibliotecas padrões, fundamentalmente porque a linguagem vem projetada para descrever circuitos digitais, temos a necessidade de utilizar uma linguagem que descreva sistemas digitais, analógicos e de sinal misto. Desta forma, VHDL-AMS é uma linguagem adequada para descrever sistemas digitais, analógicos e de sinal misto (CHRISTEN; BAKALAR, 1999). VHDL-AMS é uma extensão da linguagem de descrição de hardware digital VHDL, que surgiu do programa de circuitos integrados de altíssima velocidade (VHSIC) do governo dos Estados Unidos da América do Norte.

VHDL-AMS é um superconjunto exato de VHDL. Ele estende as capacidades da linguagem VHDL (apenas digital) pela adição de extensões analógicas e de sinal misto (AMS) (CHRISTEN; BAKALAR, 1999).

Desta forma, por exemplo, é possível projetar (da forma “**top-down**”) e modelar em VHDL-AMS um controlador digital eletrônico, no qual, o seu controle é baseado em um dispositivo FPGA (VHDL), assim como os dispositivos de um sistema de potência convencional (MCDERMOTT et al., 2006; KAZMIERSKI; HAMID, 2002).

Finalmente, desde o ponto de vista do gerenciamento e controle nos sistemas elétricos de distribuição, considerando-se a necessidade do atendimento dos níveis mínimos para os valores eficazes das tensões de alimentação, principalmente nos pontos críticos de acoplamento de cargas, tem-se a necessidade do uso de reguladores de tensão (ROYTELMAN et al., 1995). Entretanto, estes equipamentos têm sua operação fortemente influenciada pelos níveis de distorções harmônicas dos alimentadores, características de carregamento, e também, as características físicas destes alimentadores, sendo necessária uma ampla análise para sua correta instalação/operação.

Baseados no anterior panorama da simulação em tempo real para os sistemas elétricos, assim como a implementação das linguagens de descrição de hardware em software (VHDL e VHDL-AMS) e considerando-se os novos desafios das microrredes e redes inteligentes

(GUNGOR et al., 2011), a formulação do problema está caracterizada por meio da seguinte pergunta-problema:

- Baseados no conceito CHIL (*controller hardware-in-the-loop*), quais são as melhores estratégias de simulação em tempo real para as redes elétricas em média tensão, para uma operação eficiente e segura dos alimentadores de distribuição, em atendimento aos índices normatizados de QEE, utilizando a linguagem VHDL-AMS para simular o sistema elétrico e a linguagem VHDL para o gerenciamento e controle do sistema elétrico simulado, através de algoritmos digitais implementados em um dispositivo FPGA?

Portanto, para resolver a anterior pergunta-problema e desta forma atingir os objetivos desta tese, o desenvolvimento da pesquisa envolveu, desde o ponto de vista metodológico, as seguintes seis fases principais:

- Fase I: Realizou-se uma revisão global do estado da arte da pesquisa (teses, dissertações, artigos e patentes).
- Fase II: Desenvolveu-se a modelação comportamental e simulação em VHDL-AMS (cenários) de um sistema elétrico escolhido como caso de estudo.
- Fase III: Desenvolveu-se uma interface para a importação de dados da rede, permitindo-se a mudança de parâmetros em ambiente VHDL-AMS.
- Fase IV: Avaliou-se as melhores opções custo-benefício para a implementação de um dispositivo FPGA.
- Fase V: Projetou-se o sistema de gerenciamento e controle da rede no FPGA especificado.
- Fase VI: Obtiverem-se os resultados de simulação e experimentais para o estudo de caso, permitindo as conclusões finais para a tese e a formulação de propostas de continuidade para os trabalhos desta tese.

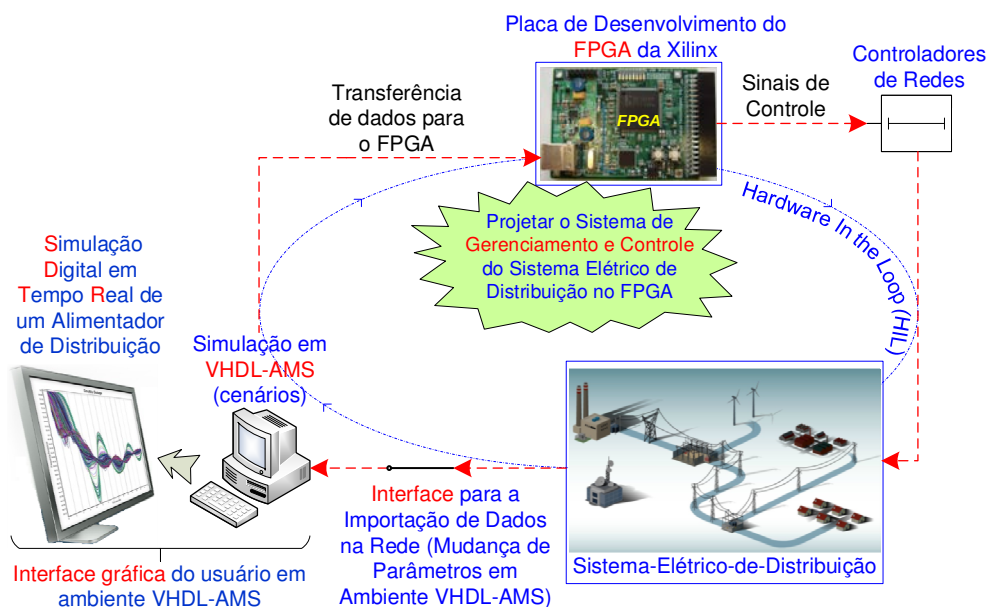
Segundo a figura 1, com o intuito de definir o planejamento e programação das atividades que resultaram nesta tese, estabeleceram-se as seguintes metodologias de desenvolvimento:

- ✓ Revisão global do estado da arte para a pesquisa. Esta ampla revisão abrangeu desde 1994 até o ano de 2015 e tem em conta 6 teses de doutorado, 8 dissertações de mestrado, 27 artigos científicos e 25 patentes. As dissertações e teses estão referenciadas (OUIOMET, M, 2008; YUKISHIMA, C, 2006; VENKATA R, 2000; MACHACA, 1994; PELLINI, 2010; FAVORETTO, C, 2006; LIMA, A. de., 2002; FAZANARO, F., 2007; OLIVEIRA, M. de., 2003; VANEGAS, M., 2010; WALTER,

Segundo a figura 1, após a definição do estado de arte atual no qual encontra-se inserida a tese, foi feita uma revisão global das referências no estado da arte da pesquisa. Além de utilizar o escopo onde encontra-se inserida a tese para a realização desta ampla revisão bibliográfica, foram utilizados como base os seguintes objetivos:

- ✓ A tese propõe uma nova arquitetura de simulação em “tempo real” e controle para alimentadores de distribuição de energia elétrica, de acordo com a Figura 2.
- ✓ A simulação em “tempo real” para sistemas de distribuição de energia elétrica deverá incorporar o conceito CHIL - **controller hardware-in-the-loop**.
- ✓ O sistema de gerenciamento e controle do sistema elétrico simulado deverá ser implementado em um dispositivo FPGA.

Figura 2- Esquema geral da arquitetura proposta para simulação em “tempo real” e controle de alimentadores de distribuição de energia elétrica.



Fonte: Próprio autor.

1.2 EXPERIMENTAÇÃO CIENTÍFICA RELACIONADA COM A MODELAÇÃO E SIMULAÇÃO EM TEMPO REAL, DERIVADAS DE ESTUDOS COM UM RTDS™ COMERCIAL

As observações apresentadas neste item são derivadas das atividades desenvolvidas com um RTDS™ comercial em parceria com a **Washington State University – WSU**, apresentado no Apêndice B. Este RTDS comercial encontra-se na WSU no **Smart Grid**

Demonstration and Research Investigation Lab (SGDRIL) e foi adquirido da empresa canadense RTDS Technologies.

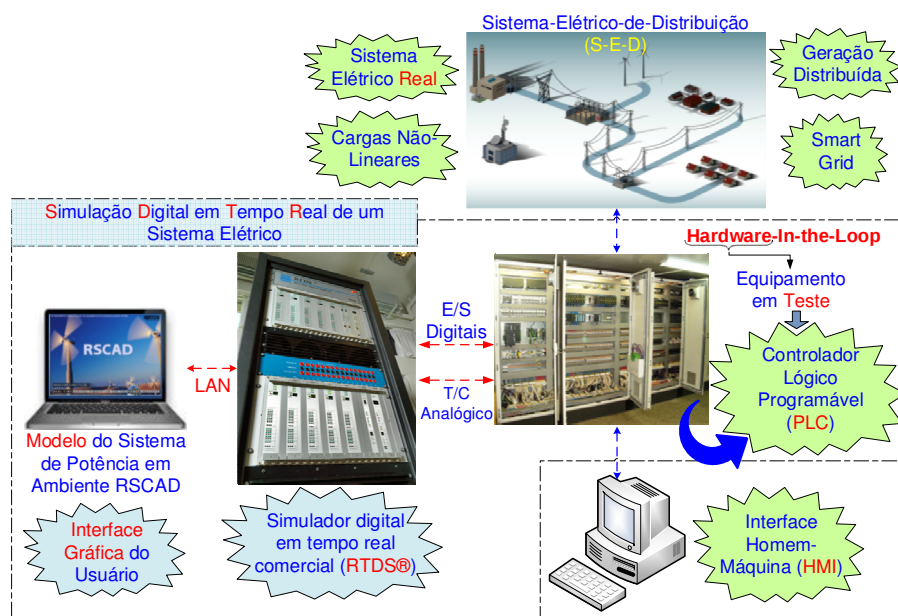
A modelação e simulação em tempo real de um sistema elétrico, além de reproduzir com exatidão o sistema real, implica em reproduzir fielmente o funcionamento deste sistema através de um dispositivo, denominado simulador (RTDS); o qual interage com o mundo real através de um controlador lógico programável, com sinais de entrada e saída (analógicas e/ou digitais), em um tempo que coincide com o tempo de atuação do sistema, chamado tempo real (JIN-HONG et al., 2010). O anterior quer dizer que as respostas no simulador ficam com o mesmo tempo que as respostas no sistema real ante uma perturbação determinada. Neste contexto, a simulação digital em tempo real é aquela simulação em tempo real que faz a recepção, processamento e transmissão da informação desde o mundo real, representada por meio de quantidades físicas (sinais), que só podem ter valores discretos (digitais).

Segundo (NEMA, 2005) um Controlador Lógico Programável (PLC) é um aparelho eletrônico digital que utiliza uma memória programável para armazenar internamente instruções e para implementar funções específicas, tais como, lógica, sequenciamento, temporização, contagem e aritmética, controlando por meio de módulos de entradas e saídas, vários tipos de máquinas ou processos.

Segundo o conceito de equipamento em teste em malha fechada, em um sistema elétrico convencional, um PLC é responsável por: 1) A recepção, processamento e transmissão da informação desde o mundo real, através dos dispositivos de controle e proteção, até dispositivos periféricos (por exemplo, interface homem-máquina - HMI ou RTDS); 2) O gerenciamento e controle do sistema (sequências de atuação do sistema autômata); 3) A HMI para as manobras no sistema (as quais podem ser feitas em modos local ou remoto); 4) A interface entre o sistema elétrico real e o simulador digital em tempo real comercial (RTDS®), para o teste do PLC (**Hardware-In-the-Loop**) ou para fazer a simulação do sistema elétrico real (OPAL - RT TECHNOLOGIES, 2015).

A figura 3 apresenta a ilustração da simulação digital em tempo real de um sistema elétrico utilizando um RTDS comercial (RTDS®).

Figura 3- Simulação digital em tempo real de um sistema elétrico utilizando um RTDS comercial (RTDS®).



Fonte: Próprio autor.

Com base no anterior e considerando as atividades desenvolvidas com um simulador digital em tempo real comercial (RTDS®), podemos estabelecer as seguintes conclusões que sem dúvida serão úteis para o desenvolvimento e avaliação dos resultados desta tese de doutoramento:

- Desde o ponto de vista das condições mínimas essenciais para fazer a simulação digital em tempo real de um sistema elétrico real, são necessárias três características:
 - ✓ Um controlador digital programável para fazer o gerenciamento e controle de alto nível do sistema.
 - ✓ Um simulador em tempo real para reproduzir e simular com exatidão o sistema real.
 - ✓ Uma interface gráfica do usuário para modelar o sistema elétrico de potência que se quer simular em um ambiente configurável e amigável.
- A simulação digital em tempo real de um sistema elétrico convencional pode ser arranjada com dois objetivos, os quais podem ser feitos ao mesmo tempo ou separadamente:
 - ✓ Testar o controlador lógico programável (PLC). Neste caso o PLC é desligado do sistema elétrico real e conectado ao RTDS, com o intuito de “enganar” o PLC e dessa maneira, são feitos diversos testes no controlador. A simulação em tempo

real atua como um teste de equipamento (PLC) em malha fechada (Hardware-In-the-Loop).

- ✓ Fazer a simulação do sistema elétrico real, incorporando ou não algumas modificações no modelo do sistema de potência (na interface gráfica do usuário).
- Para fazer a modelação e simulação em tempo real para microrredes inteligentes, que envolvem sistemas elétricos de distribuição com geração distribuída, são necessários dois tipos de sistemas de controle, os quais devem operar ao mesmo tempo (IBARRA HERNANDEZ et al., 2013):
 - ✓ Um Sistema de Gerenciamento da Distribuição (Distribution Management System – DSM), o qual procura as melhores condições de fornecimento e consumo da energia elétrica dentro da microrrede. Na atualidade é muito comum usar um sistema multi-agentes para fazer, no contexto da simulação em tempo real, a representação do DSM. Em resumo, podemos dizer que um DMS é um sistema de suporte à decisão para ajudar o centro de operação e controle das redes elétricas e ao pessoal da operação de campo, com o monitoramento e controle do sistema de distribuição de energia elétrica, da melhor forma possível e de acordo com critérios hierárquicos estabelecidos, melhorando a segurança das redes e a proteção dos ativos da concessionária.
 - ✓ Um controle local desenvolvido para o acionamento individual dos Recursos de Energia Distribuída (Distributed Energy Resources – DER). Segundo o Instituto de Pesquisa da Energia Elétrica (ELECTRIC POWER RESEARCH INSTITUTE - EPRI, 2012), os números de DER que são instalados no sistema de distribuição têm aumentado consideravelmente nos últimos anos. Os DER variam em tamanho que vão desde unidades geradoras de escala da concessionária (1 megawatt ou superior), até pequenas unidades de propriedade do consumidor, tais como: energia solar fotovoltaica nos tetos das edificações, com uma capacidade de apenas alguns quilowatts. O recente crescimento dos DER é devido em parte, ao interesse na redução das emissões de gases, que provocam o efeito de estufa, e são provenientes dos combustíveis fósseis, derivados dos recursos centralizados de geração. Outra razão para o aumento da implantação dos DER é o crescimento do interesse dos consumidores na redução dos custos da energia elétrica e o melhoramento da confiabilidade do serviço. Muitas unidades geradoras de propriedade dos clientes geram mais energia do que é demandada pelo cliente em determinados momentos do dia, as quantidades excessivas podem então ser

armazenadas em baterias locais ou vendidas à distribuidora de energia elétrica, como um incentivo financeiro acrescentado pela instalação de tais recursos.

- Um RTDS comercial pode-se conectar diretamente a um dispositivo de controle e/ou a relés de proteção reais para realizar testes destes equipamentos em malha fechada (***Hardware in the Loop - HIL***).
- Um RTDS comercial está composto na sua forma básica por:
 - ✓ Um hardware, formado por um ou mais armários (racks), os quais, pela escalabilidade do RTDS, podem aumentar o desempenho deste com o acréscimo de hardware (racks), proporcionalmente à capacidade acrescida. Em um rack existem dois ou mais tipos de processadores de alta velocidade operando em paralelo. Por cada rack pode-se resolver um sistema elétrico de até 42 nós (no caso do RTDS utilizado).
 - ✓ Uma interface gráfica do usuário (Graphical User Interface - GUI), conhecida como RSCAD no caso do RTDS™, através da qual o usuário é capaz de construir, executar e analisar os casos de simulação. O software RSCAD contém uma biblioteca de componentes do sistema de controle e do sistema elétrico que permite desenhar o sistema simulado em um entorno CAD (Computer-Aided Design), ou seja, projeto assistido por computador. Neste contexto, o RSCAD inclui modelos precisos de componentes do sistema elétrico de potência, necessários para representar os complexos elementos que compõem os sistemas elétricos de potência físicos (ou reais).
- O RTDS® realiza simulações para o estudo e análise desde pequenas, até muito grandes e complexas redes de energia elétrica. As aplicações incluem: 1) Estudo do funcionamento de um sistema AC em geral, incluindo o comportamento de sistemas de geração, transmissão e/ou distribuição; 2) Integração e operação da geração distribuída e energias renováveis; 3) Investigação e testes de iniciativas de redes inteligentes (***smartgrids***), incluindo proteções elétricas e controle; 4) Testes em malha fechada de relés de proteção e/ou sistemas de controle; 5) Investigação da interação de equipamentos com o sistema de potência; 6) Estudo da interação entre os sistemas integrados AC/DC; 7) Desenvolvimento de dispositivos FACTS e controles associados e 8). Treinamento e formação de pessoal em sistemas elétricos de potência. No caso desta pesquisa de doutoramento, o projeto do simulador em “tempo real” só está circunscrito parcialmente às aplicações 1) e 3), envolvendo: funcionamento de um

sistema elétrico de distribuição de pequeno porte e o controle não avançado para microrredes “inteligentes”.

- Além das anteriores observações, para o RTDS comercial no qual se desenvolveram as atividades em parceria com a WSU, verificamos ainda que:
 - ✓ Na modelação e simulação em tempo real, uma hora no mundo real é igual a uma hora no simulador. Além disso, embora o RTDS TM tenha uma alta exatidão nos seus cálculos, a faixa temporal para os estudos elétricos pode ir desde uns poucos microssegundos, até dezenas de horas de operação sem interrupções.
 - ✓ O tempo de passo da simulação, com o modelo dinâmico totalmente em operação é de 50 microssegundos e para a simulação de sistemas que contém dispositivos eletrônicos de potência de alta frequência, o tempo de passo é de 1,5 microssegundos.
- A simulação digital em tempo real deve cumprir com os seguintes conceitos:
 - ✓ Previsibilidade: Os fenômenos simulados apresentados no sistema devem ser calculados com certa margem de precisão.
 - ✓ Estabilidade: O nível de falhas do simulador tem que ficar abaixo de certo limite.
 - ✓ Controlabilidade: Um sistema tem que ser controlado em todo o seu espaço de configuração usando apenas certas manipulações admissíveis.
 - ✓ Acessibilidade: A simulação deve entregar a maior quantidade de informação disponível do sistema de acordo com certos parâmetros e independentemente do sistema simulado.
- Na modelação e simulação em tempo real para microrredes inteligentes que envolvem sistemas elétricos de distribuição com geração distribuída, necessita de dois tipos de sistemas de controle:
 - ✓ Um controle distribuído para a microrrede, o qual procura a descentralização da tomada de decisões dentro da smartgrid. A tomada de decisões está relacionada com a procura das melhores condições de fornecimento e consumo da energia elétrica dentro da microrrede, isto se chama Sistema de Gerenciamento da Distribuição (Distribution Management System – DMS) em tempo real.
 - ✓ Um controle local desenvolvido para o acionamento individual dos recursos de energia distribuída (Distributed Energy Resources – DER). Por exemplo, o sistema de rastreamento do ponto de máxima potência para os painéis solares.
- Segundo (VENKATARAMANAN; MARNAY, 2008), uma microrrede (**microgrid**) é um sistema elétrico que contém recursos energéticos distribuídos, especialmente em

pequena escala (combinação de fontes de calor e de energias renováveis) e várias cargas elétricas, no qual o dito sistema, pode ser operado em paralelo com a rede elétrica pública, ou também como uma rede elétrica ilhada e além disso, atender aos requisitos exigentes de qualidade e confiabilidade, alcançados através da incorporação de controles modernos que lhes permitem operar com certo grau de autonomia em relação à rede elétrica tradicional. Com base no anterior, em condições normais de operação de uma microrrede, o comportamento ideal desta aponta para três características: 1) As transições entre os modos de operação conectada à rede elétrica tradicional e microrrede isolada devem ser “suaves” e automáticas, 2) A proteção elétrica dentro de uma microrrede não deve depender de altas correntes de falha típicas das redes de distribuição convencionais, 3) A estabilidade de frequência e tensão de uma microrrede, em condição de conectada à rede elétrica ou isolada, deve ser alcançada sem a necessidade de comunicações de alta velocidade.

- Uma microrrede deve conter fontes primárias de calor e de energias renováveis integradas e em condições de operação ideal, esta deve ser projetada para exportar grandes quantidades de energia fotovoltaica. Neste contexto, uma microrrede deve incluir sistemas de armazenamento para absorver a energia adicional, tais como: baterias ou supercapacitores.
- As microrredes (**microgrids**) exigem um robusto sistema de controle e gerenciamento de energia em tempo real para operar simultaneamente, com a rede elétrica ou ilhadas, para fornecer um serviço confiável e ajudar a alcançar os objetivos voltados para o cliente, utilizando os ativos do sistema de distribuição local.
- Conforme a quantidade e complexidade dos elementos do sistema elétrico crescem rapidamente, o controle centralizado é cada vez mais complexo e oneroso. Em uma microrrede, precisamos evitar que o controle seja feito através de um controlador central, em vez disso, temos que descentralizar a tomada de decisões dentro da microrrede. Por isto, hoje em dia, no contexto de simulações de “redes inteligentes”, são muito utilizados os sistemas multiagentes para efetuar o controle em tempo real de uma microrrede. Segundo (NIANNIAN et al., 2011), um sistema multiagente tem dois ou mais agentes (entidades) que podem se comunicar entre si para atingir a sua meta local e conseqüentemente, desta forma, cumprir a meta do sistema global. O aspecto mais importante da implementação de sistemas multiagentes é dividir um problema de controle complexo em vários problemas simples menores, de tal forma que cada problema simples será resolvido por um agente. Os agentes são proativos, em outras

palavras, eles podem tomar decisões com base nos seus objetivos e na informação que obtêm a partir do meio ambiente, sem um controlador central.

- Tipicamente os agentes que fazem parte de um sistema multiagentes para o controle e gerenciamento da energia de uma microrrede, para a operação em tempo real são: agentes das fontes de energias renováveis, agentes das cargas, agentes do sistema de armazenamento e agentes para o gerenciamento da energia, além dos agentes que por defeito trazem os softwares utilizados para projetar sistemas multiagentes, por exemplo, o agente *sniffer* ou Sistema de Gestão de Agentes (AMS).
- Os maiores desafios no gerenciamento da energia e controle (***Energy Management and Control - EM&C***) para uma microrrede ilhada são:
 - ✓ A interconexão de unidades de recursos energéticos distribuídos (Distributed Energy Resources) aumenta a complexidade do EM&C.
 - ✓ A indisponibilidade da rede de distribuição local (grid) requer de um gerenciamento específico das microrredes ilhadas, devido aos recursos renováveis, não despacháveis (por exemplo, a energia eólica ou fotovoltaica).
 - ✓ Para que exista um desempenho confiável de uma microrrede ilhada deve se manter a frequência e a tensão dentro dos limites exigidos pelas cargas.
- Do ponto de vista comercial, hoje estão disponíveis vários simuladores digitais em tempo real (RTDS), sendo os mais importantes no mundo: RTDS TM desenvolvido pela RTDS Technologies Inc., eMEGAsim desenvolvido pela empresa OPAL-RT Technologies Inc e, HYPERSIM desenvolvido pela Hydro-Québec.

1.3 ORGANIZAÇÃO DO TRABALHO

O presente trabalho de tese encontra-se organizado da seguinte maneira:

No Capítulo 1 apresenta-se a introdução geral para o trabalho, na qual, foi feita uma panorâmica da situação atual (e futura) dos sistemas elétricos de distribuição (necessidades e desafios) e a partir disto, foi possível fazer a formulação do problema de pesquisa (item 1.1). Logo, a partir dos estudos e familiarização com um RTDS comercial derivados do trabalho, desenvolvido em parceria com a ***Washington State University - WSU***, foi possível apresentar a experimentação científica relacionada com a modelação e simulação em tempo real utilizando este RTDS (item 1.2).

No Capítulo 2 apresentam-se as modelações desenvolvidas e simulações exemplos de casos para sistemas de distribuição de energia elétrica, utilizando-se a linguagem de descrição

de hardware VHDL-AMS, destacando-se as potencialidades desta técnica de descrição de hardware em software.

No Capítulo 3 apresenta-se o controlador digital proposto e implementado através de um dispositivo FPGA, para o sistema de gerenciamento e controle de um alimentador de uma rede de distribuição primária de 14,4 kV, como caso de estudo simulado.

No Capítulo 4 apresentam-se os detalhes da implementação prática e resultados experimentais da arquitetura proposta de simulação em “tempo real” e controle, desenvolvida para um alimentador de distribuição de energia elétrica, utilizando-se as linguagens de descrição de hardware VHDL-AMS e VHDL.

Finalizando, o Capítulo 5 apresenta as conclusões gerais e as propostas de continuidade para o trabalho.

Adicionalmente, no apêndice A apresentam-se os artigos científicos produzidos, decorrentes da tese de doutoramento, até o momento atual.

No apêndice B se apresenta o desenvolvimento das atividades com um RTDS em parceria com a **Washington State University** – WSU, nos Estados Unidos da América do Norte.

No apêndice C apresentam-se os dados quantitativos referentes à revisão geral do estado da arte da pesquisa.

No apêndice D apresentam-se as bibliotecas da linguagem VHDL-AMS, as simulações das cargas e linhas não críticas do alimentador elétrico escolhido como estudo de caso e, os códigos de programação e hardware resultante no dispositivo FPGA.

No apêndice E apresenta-se o código em Orcad/Pspice, usado para as comparações com o estudo de caso das simulações em VHDL-MAS, para um alimentador modelado pelo lado da média tensão.

MODELAÇÃO DE SISTEMAS DE DISTRIBUIÇÃO EM AMBIENTE VHDL-AMS

Neste capítulo são apresentados a modelação desenvolvida em linguagem de descrição de hardware VHDL-MAS e os resultados das simulações no ambiente VHDL-AMS, com o propósito da implementação de uma arquitetura de simulação em “tempo real” e controle para alimentadores de distribuição de energia elétrica e, desta forma, contribuindo para a consecução dos objetivos deste trabalho.

2.1 A LINGUAGEM VHDL-AMS

Segundo (KAZMIERSKI, 1998), VHDL-AMS é uma linguagem para descrever sistemas digitais, analógicos e de sinal misto. É uma extensão da linguagem de descrição de hardware digital VHDL, que surgiu do programa de circuitos integrados de altíssima velocidade (VHSIC) do governo dos Estados Unidos da América do Norte. Neste programa, ficou claro que havia a necessidade de uma linguagem padrão para descrever as estruturas e funções dos circuitos integrados (CIs). Daí a linguagem de descrição de hardware VHSIC (VHDL) foi desenvolvida. Posteriormente, a linguagem foi ainda padronizada pelo Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE), constituindo-se o padrão IEEE 1076, manual de referência da linguagem VHDL, em 1987. Esta primeira versão padrão da linguagem é muitas vezes referida como VHDL-87.

Como todos os padrões IEEE, o padrão VHDL está sujeito a uma revisão a cada cinco anos. Comentários e sugestões dos usuários do padrão de 1987, foram analisados pelo grupo de trabalho responsável pela IEEE VHDL e, em 1992, uma versão revisada do padrão foi proposta. Esta acabou por ser adotada em 1993 através da VHDL-93. Uma segunda rodada de revisão desta norma foi iniciada em 1998. Esse processo foi concluído em 2001, resultando na versão atual da linguagem digital VHDL-2001.

No início de 1990, a necessidade de uma linguagem de descrição de hardware de suporte de modelação analógica e de sinal misto tornou-se necessária, desta forma, um grupo de trabalho IEEE foi estabelecido para sua implementação. Este grupo desenvolveu um

conjunto de objetivos de projeto para uma extensão para VHDL e completou um projeto de Manual de Referência da Linguagem em 1997. O projeto foi refinado e posteriormente, aprovado em 1999, tornando-se o padrão IEEE1.076,1: Definições de Extensões de sinais analógicos e mistos para o padrão IEEE VHDL, resultando na linguagem denominada VHDL-AMS.

VHDL-AMS é concebida para preencher uma série de necessidades no processo de projeto, pois primeiro ela permite a descrição da estrutura de um sistema, isto é, como este sistema é decomposto em subsistemas e como estes subsistemas estão interligados, além de permitir a especificação da função de um sistema usando uma linguagem de programação familiar e formas de equação. Consequentemente, permite que o projeto de um sistema seja simulado antes de ser fabricado, de modo que os projetistas possam, rapidamente, comparar alternativas e testar a correção, sem a demora e o custo da criação de protótipos de hardware, permitindo que a estrutura detalhada de um projeto possa ser sintetizada a partir de uma especificação mais abstrata, permitindo que os projetistas se concentrem em decisões de projeto mais estratégicas, reduzindo o tempo de entrega ao mercado.

Segundo (CHRISTEN; BAKALAR, 1999), VHDL-AMS é um superconjunto da linguagem VHDL, pois ela estende as capacidades da linguagem VHDL (apenas digital) pela adição de extensões analógicas e de sinal misto (AMS). Entre seus muitos recursos o VHDL-AMS:

- Representa modelos complexos diretamente. Estes modelos podem ser algébricos em sua natureza ou representados por equações diferenciais ordinárias não lineares.
- Representa valores contínuos (analógicos) e discretos (digitais).
- Também pode modelar fenômenos não elétricos (isto é, mecânicos, fluídos e térmicos).
- Tem um conceito de tempo (ao contrário de linguagens de programação convencionais tais como C / C ++, Java, dentre outras).
- Tem um ciclo de simulação bem definido que permite o comportamento do modelo de forma consistente.
- Pode resolver equações usando as leis de conservação de energia (por exemplo, Lei de Tensão de Kirchhoff, Leis de Newton).
- Trabalha com condições iniciais e o comportamento pode ser definido por partes.

Para uma ampla informação sobre a linguagem VHDL-AMS, no apêndice D encontram-se as bibliotecas modelos que estão disponíveis para serem utilizadas nas

modelações e simulações nas quais é empregada a linguagem VHDL-AMS, disponíveis no software SystemVision™.

2.2 JUSTIFICATIVA PARA O USO DA LINGUAGEM VHDL-AMS

No trabalho desenvolvido por (BOUSSETTA et al., 2010) é feita uma comparação de um conjunto de soluções atuais já existentes, que incluem as linguagens e ferramentas dedicadas a fim de escolher a mais adequada opção para as necessidades comuns associadas à esta tese, quais sejam:

- A linguagem SPICE pode ser utilizada para simular os sistemas mistos, sem limite teórico de tamanho. No entanto, na prática, é limitado em termos de desempenho e interoperabilidade de simulações ao lidar com sistemas cada vez mais complexos que requerem uma abordagem hierárquica. Além disso, quando se trata de domínios multifísicos, não temos outra escolha a não ser utilizar circuitos equivalentes elétricos.
- Simulink™ é utilizado principalmente em aplicações de controle. Uma de suas principais propriedades é que as conexões são apenas do tipo de fluxo de sinal (um formalismo abstrato onde o comportamento é unidirecional com um fluxo de entrada e saída, explicitamente definido). Uma comparação entre a modelação em Simulink™ e em VHDL-AMS, por meio de um exemplo é detalhado em (GUESSAB e OUDINOT, 2004). O autor conclui que VHDL-AMS permite mais fácil e mais direta a descrição de sistemas de sinais mistos, através de seu tipo de conexão conservativa respeitando as leis de conservação de energia e suas amplas possibilidades de descrever equações físicas.
- Linguagens para projetar hardware analógico: As duas principais linguagens de descrição de hardware analógico são VHDL-AMS e Verilog-AMS, extensões respectivamente de VHDL e Verilog. Para uma cuidadosa comparação entre VHDL-AMS e Verilog-AMS pode-se recorrer a (PECHEUX et al., 2005). Neste estudo, os autores destacam as razões práticas que nos levaram a privilegiar o uso da norma IEEE-1.076.1, a qual é a linguagem de modelação padrão VHDL-AMS. Por um lado, o sequenciamento de equações é uma tarefa transparente em VHDL-AMS, graças a potente notação de equações fornecidas por declarações simultâneas. Por outro lado, Verilog-AMS restringe o projetista a empregar declarações sequenciais em blocos analógicos. Além disso, VHDL-AMS permite a utilização de variáveis intermediárias chamadas “quantidades livres” em contraste com as quantidades ramais que estão

relacionadas com os nós. Esta última característica é muito útil na modelação de circuitos complexos com descrições matemáticas complicadas e assim, todas as variáveis desconhecidas em Verilog-AMS estão associadas aos nós. Consequentemente, um projetista usando Verilog-AMS tem que incluir um nó fictício para cada variável intermédia, o que aumenta a complexidade do modelo. Esta característica reforça a abordagem em nível de circuito de Verilog-AMS, mas induz a uma perda em termos de "grau de liberdade" em comparação com VHDL-AMS, durante a codificação de complexas Equações Diferenciais Algébricas.

- SystemC-AMS é uma extensão do SystemC que fornece construções de linguagem adicionais para a modelação do comportamento de sistemas de sinais mistos analógicos. A versão atualmente disponível de SystemC-AMS é otimizada para aplicações dominadas por processamento de sinais. Esta linguagem permite a modelação do comportamento de redes lineares conservativas e sistemas não conservativos (Fluxo do sinal) com equações estáticas não lineares e dinâmicas lineares. Em relação aos sistemas conservativos, (YAO et al., 2014) propõe algumas experiências do SystemC-AMS para a modelação destes elementos com equações algébricas dinâmicas não lineares.

Pelas razões anteriormente apresentadas, foi selecionada a VHDL-AMS como a melhor linguagem de descrição de hardware disponível que atende aos requisitos de desenvolvimento desta tese. Muitas ferramentas comerciais existem atualmente para trabalhar com simulações VHDL-AMS. Neste trabalho de doutoramento, as simulações dos sistemas elétricos em VHDL-AMS foram realizadas utilizando-se o software SystemVision™, fornecido pela empresa Mentor Graphics.

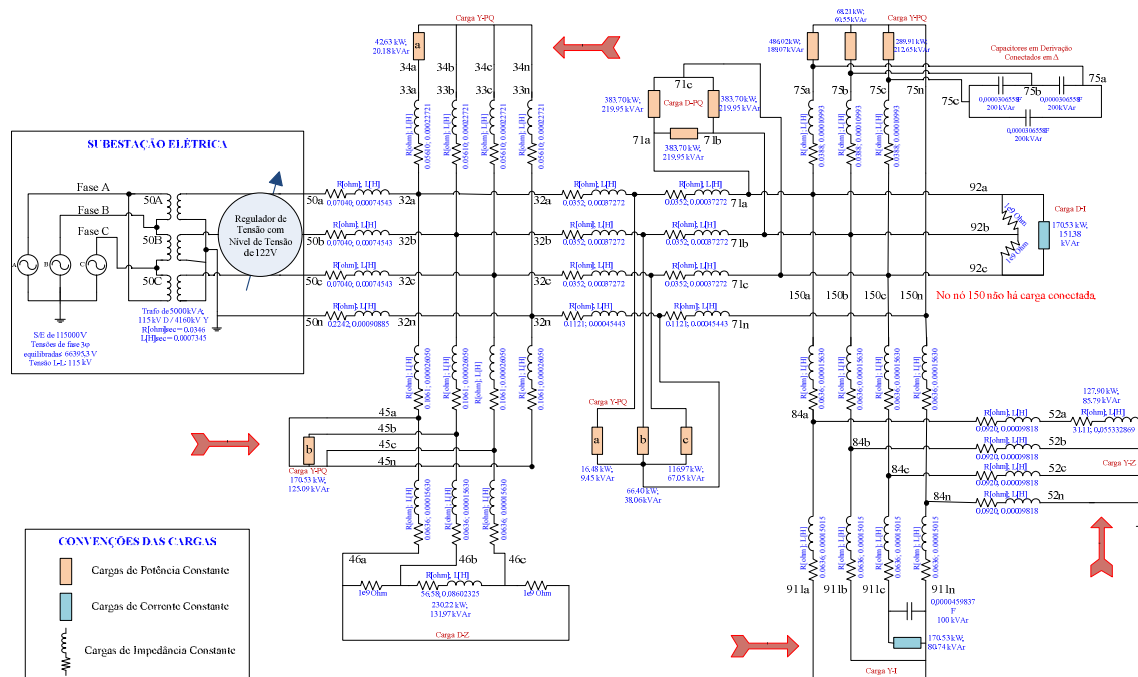
2.3 MODELAÇÃO E SIMULAÇÃO DE UM SISTEMA DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA UTILIZANDO VHDL-AMS: CASO DE ESTUDO PADRÃO IEEE ELEMENTAR

Em 1991 foi publicado um trabalho que apresentou os dados completos para o teste de alimentadores de distribuição radiais trifásicos em estrela a quatro fios (lado baixa tensão) e trifásicos em delta a três fios (lado de média tensão), expostos na forma de alimentadores radiais IEEE de 123, 37, 34 e 13 nós (KERSTING, 1991). O objetivo desta publicação foi disponibilizar um conjunto de dados que poderiam ser usados por desenvolvedores de software e usuários, a fim de verificar a exatidão das suas soluções. Neste contexto, segundo o

trabalho publicado em (IBARRA HERNANDEZ; CANESIN, 2012), foi utilizado o teste do alimentador IEEE de 13 nós para desenvolver um conjunto de modelos preliminares na linguagem VHDL-AMS. O artigo em referência, que integra os trabalhos decorrentes desta tese, apresenta a simulação em VHDL-AMS de um sistema elétrico de potência (Padrão IEEE 13 Nós), resultando no ponto de partida para o desenvolvimento da arquitetura de simulação em “tempo real” e controle, aqui proposta, para alimentadores de distribuição de energia elétrica.

De acordo com a figura 4, o alimentador IEEE de 13 nós é de pequeno porte, porém exibe algumas características muito interessantes, sendo: 1) De extensão curta e altamente carregado para um alimentador de 4,16 kV, no qual as potências zero das cargas monofásicas são curto-circuitadas, 2) O regulador de tensão da subestação é composto por três unidades monofásicas conectadas em estrela e 3) Possui cargas pontuais desequilibradas e também distribuídas.

Figura 4- Sistema de teste IEEE de 13 nós modelado em VHDL-AMS.



Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

Considerando-se o anterior sistema IEEE de 13 nós, usando o software SystemVision™ (fornecido pela empresa Mentor Graphics), foram utilizados modelos básicos de componentes elétricos em VHDL-AMS, para desenvolver a simulação deste alimentador,

servindo como base para construir o simulador proposto em “tempo real”. Estes modelos são apresentados como segue:

2.3.1 Modelo da fonte de tensão monofásica: A subestação de distribuição foi modelada usando três fontes monofásicas de tensão conectadas em estrela aterrada e três transformadores monofásicos 115 kV delta - 4,16 kV em estrela aterrada, além de um regulador de tensão com um nível de tensão de 122 V. O modelo da fonte monofásica de tensão em VHDL-AMS utilizado é dado por:

TABELA 1- MODELO DA FONTE DE TENSÃO MONOFÁSICA.

Modelo da Fonte de Tensão Monofásica
<pre> library IEEE; use IEEE.MATH_REAL.all; -- Use IEEE natures and packages use IEEE.ELECTRICAL_SYSTEMS.all; entity v_sine is generic (freq : real:= 60.0; -- frequency [Hertz] amplitude : voltage:= 115000.0; -- amplitude [Volts] phase : real:=0.0; -- initial phase [Degrees] offset : voltage := 0.0; -- DC value [Volts] df : real := 0.0; -- damping factor [1/second] ac_mag : voltage := 0.0; -- AC magnitude [Volts] ac_phase : real := 0.0; -- AC phase [Degrees]) port (terminal pos, neg : electrical); end entity v_sine; architecture of v_sine is function calc_limit(freq : real) return real is variable lim : real; begin if freq = 0.0 then lim := 1.0e12; -- A large value elsif freq < 0.0 then lim := 1.0 / (-20.0 * freq); else lim := 1.0 / (20.0 * freq); end if; return lim; end function; constant v_limit : real := calc_limit(freq); quantity v across i through pos to neg; limit v : voltage with v_limit; quantity phase_rad : real; quantity ac_spec : real spectrum ac_mag, math_2_pi*ac_phase/360.0; begin phase_rad == math_2_pi *(freq * NOW + phase / 360.0); if domain = quiescent_domain or domain = time_domain use </pre>

```

v == offset + amplitude * sin(phase_rad) * exp(-now * df);
else
v == ac_spec;
end use;
end architecture;

```

Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

2.3.2 Modelo do transformador monofásico: Para o desenvolvimento deste modelo foi usada uma formulação semelhante a (MCDERMOTT et al., 2006), sendo que no lado secundário temos o comportamento de um transformador ideal $n:1$, e no lado primário incluímos uma impedância $r + j\ell$. A tensão primária em cada enrolamento, por exemplo, $va1$, é igual à tensão do enrolamento secundário referido ao primário, além de uma queda de tensão de $r.i + \ell.(di / dt)$. A corrente secundária é igual à corrente do enrolamento primário, que se refere ao lado secundário, porque este modelo ignora a impedância magnetizante. O modelo do transformador monofásico em VHDL-AMS é dado por:

TABELA 2- MODELO DO TRANSFORMADOR MONOFÁSICO.

Modelo do Transformador Monofásico

```

library IEEE;
use IEEE.electrical_systems.all;
use IEEE.fundamental_constants.all;
entity transformer is
generic (
kv1: Kilo voltage := 115.0;
kv2: Kilo voltage := 2.401;
mva: real := 1.667;
freq: real := 60.0;
x_pct : real := 8.0;
r_pct : real := 1.0);
port (terminal a1, b1, a2, b2);
end entity transformer_phy;
architecture transformer is
constant omega : real := freq*math_2_pi;
constant n : real := kv1 / kv2;
constant zbase : real := kv1 * kv1 / mva;
constant r : resistance := 0.01 * r_pct * zbase;
constant l : inductance := 0.01 * x_pct * zbase / omega;
quantity mmf across flux through mag1 to mag2;
quantity va1 across ia1 through a1 to electrical_ref;
quantity vb1 across ib1 through b1 to electrical_ref;
quantity va2 across ia2 through a2 to electrical_ref;
quantity vb2 across ib2 through b2 to electrical_ref;
begin
va1 == n * va2 + r * ia1 + l * ia1'dot;

```

```

vb1 == n * vb2 + r * ib1 + l * ib1'dot;
ia2 == -n * ia1;
ib2 == -n * ib1;
end architecture;

```

Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

2.3.3 Modelos de componentes elétricos usados na modelação: Os seguintes modelos básicos gerais de fonte de corrente monofásica, resistor, indutor e capacitor desenvolvidos em VHDL-MAS, foram usados de forma recorrente na modelação do circuito mostrado na figura 4.

TABELA 3- MODELOS DE COMPONENTES ELÉTRICOS USADOS NA MODELAÇÃO.

Modelo da Fonte de Corrente Monofásica
<pre> library IEEE; use ieee.math_real.all; use IEEE.electrical_systems. all; entity i_sine is generic (freq : real; -- frequency [Hertz] amplitude : current; -- amplitude [Amps] phase : real := 0.0; -- initial phase [Degrees] offset : current := 0.0; -- DC value [Amps] df : real := 0.0; -- damping factor [1/second] ac_mag : current := 0.0; -- AC magnitude [Amps] ac_phase : real := 0.0; -- AC phase [Degrees]) port (terminal pos, neg : electrical); end entity i_sine; architecture of i_sine is function calc_limit(freq : real) return real is variable lim : real; begin if freq = 0.0 then lim := 1.0e12; -- A large value elsif freq < 0.0 then lim := 1.0 / (-20.0 * freq); else lim := 1.0 / (20.0 * freq); end if; return lim; end function; constant i_limit : real := calc_limit(freq); quantity v across i through pos to neg; limit i : current with i_limit; quantity phase_rad : real; quantity ac_spec : real spectrum ac_mag, math_2_pi*ac_phase/360.0; begin phase_rad == math_2_pi *(freq * NOW + phase / 360.0); </pre>

```

use
  i == offset + amplitude * sin(phase_rad) * exp(-NOW*df);
else
  i == ac_spec;
end use;
end architecture;

```

Modelo do Resistor

```

library IEEE;
use IEEE.electrical_systems.all;
entity resistor is
  generic (
    res : resistance; -- resistance (no initial value)
  )
  port (terminal p1, p2 : electrical);
end entity resistor;
architecture of resistor is
  quantity v across i through p1 to p2;
begin
  v == i*res; -- Fundamental equation
end architecture;

```

Modelo do Indutor

```

library IEEE;
use IEEE.electrical_systems.all;
entity inductor is
  generic (
    ind : inductance; -- Nominal inductance
    i_ic : real := real'low; -- Initial current
  )
  port (terminal p1, p2 : electrical);
end entity inductor;
architecture of inductor is
  quantity v across i through p1 to p2;
begin
if domain = quiescent_domain and i_ic /= real'low use
  i == i_ic;
else
  v == ind * i'dot -- Fundamental equation
end use;
end architecture;

```

Modelo do Capacitor

```

library IEEE;
use IEEE.electrical_systems.all;
entity capacitor is
  generic (
    cap : capacitance; -- Capacitance [F]
    v_ic : real := real'low; -- Initial voltage
  )
  port (terminal p1, p2 : electrical);
end entity capacitor;
architecture of capacitor is
  quantity v across i through p1 to p2;
begin
if domain = quiescent_domain and v_ic /= real'low use
  v == v_ic;
else

```

```

i == cap * v'dot; -- Fundamental equation
end use;
end architecture.

```

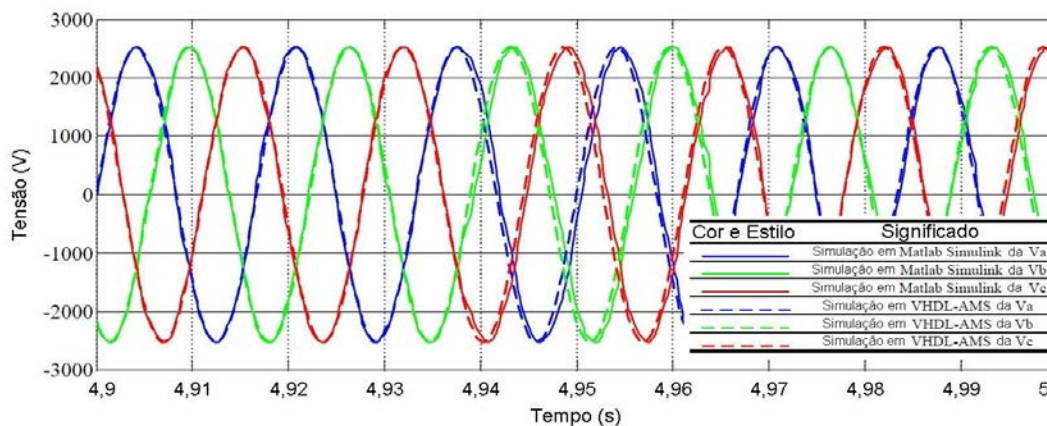
Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

Segundo (RAZZAGHI et al., 2013), uma distinção primária entre os diferentes tipos de simuladores digitais é a velocidade a que os cálculos são efetuados para gerar os pontos de dados discretos correspondentes das formas de onda. Se o tempo requerido para o cálculo de cada ponto de dados é menor do que o passo de tempo fixo, o cálculo é dito ser em tempo real, se o tempo de computação é maior do que o passo de tempo da simulação, então as formas de onda não são geradas em tempo real.

Com base no exposto anteriormente e, considerando-se as especificações do alimentador selecionado, foram feitos todos os cálculos de engenharia (resistências, indutâncias, capacitâncias e tipos de cargas) com um grau de exatidão adequado, em uma planilha eletrônica de Microsoft Excel para depois implementar a modelação em VHDL-AMS. Estas simulações em VHDL-AMS foram confrontadas com simulações no padrão altamente personalizável MATLAB/SIMULINK (biblioteca **SimPowerSystems**).

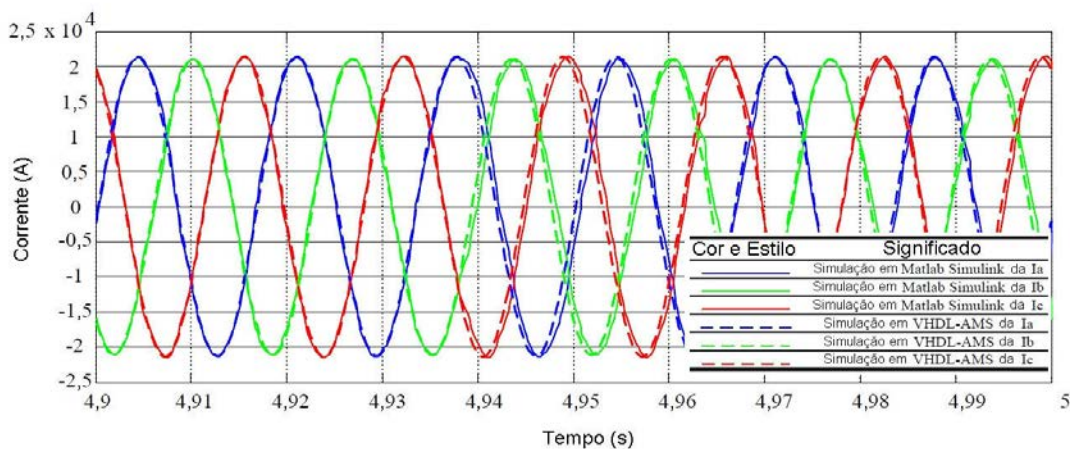
2.3.4 Simulações em VHDL-AMS: As figuras 5 até 8 contêm os resultados das simulações de um alimentador de distribuição de energia elétrica (padrão IEEE de 13 nós), utilizando a linguagem VHDL-AMS e confrontadas com MATLAB/SIMULINK. Estes resultados foram publicados em (IBARRA HERNANDEZ; CANESIN, 2012).

Figura 5- Tensões trifásicas na saída do regulador de tensão (tensões de fase).



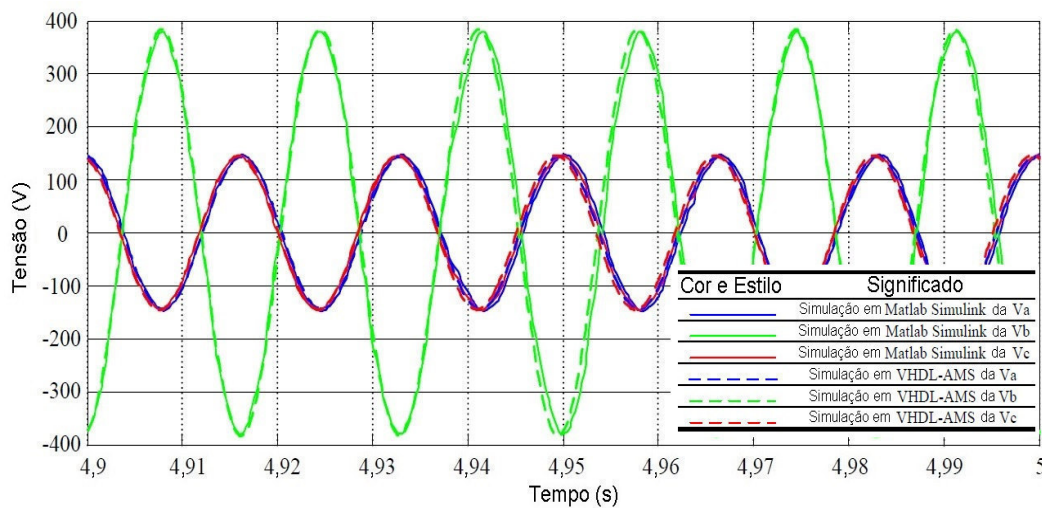
Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

Figura 6- Correntes de linha que fluem através do regulador de tensão.



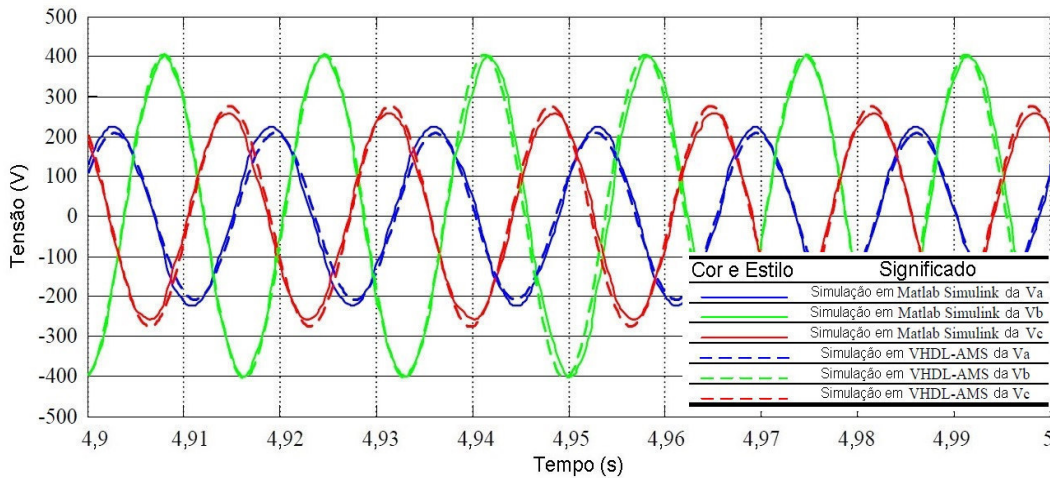
Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

Figura 7- Tensões trifásicas no nó 75 (tensões de fase).



Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

Figura 8- Tensões trifásicas no nó 71 (tensões de fase).



Fonte: (IBARRA HERNANDEZ; CANESIN, 2012).

A partir dos resultados, considerando as comparações da linguagem VHDL-AMS com uma plataforma de simulação bem conhecida (MatLab/Simulink), pode-se concluir que: 1) As tensões de saída do transformador são trifásicas e equilibradas de 1,00 p.u., 2) O alimentador trifásico é desequilibrado e inclui três tipos de cargas: De potência constante (kW e kVAr), de impedância constante (Z) e de corrente constante (I), e 3) De acordo com o valor RMS em estado estacionário medido em 5 segundos, o maior erro entre os resultados dos dois métodos de simulação foi observado na fase C no nó 71, quando seu valor foi igual a 1,929%.

Portanto, a partir destas conclusões preliminares, pode-se com segurança admitir o uso de VHDL-AMS para a modelação do estudo de caso principal admitido para esta tese.

2.4 PROJETO DA ARQUITETURA DE SIMULAÇÃO EM “TEMPO REAL” E CONTROLE PARA ALIMENTADORES DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA, UTILIZANDO VHDL-AMS

Considerando o caso de estudo elementar (Padrão IEEE 13 nós) exposto no item anterior, constituindo-se como trabalho primário para o projeto do simulador em “tempo real” usando o software SystemVision™, proposto para a tese, esboçou-se as seguintes etapas para a realização do projeto de arquitetura do simulador proposto, empregando linguagem VHDL-AMS:

- 2.4.1 Desenvolver em plataforma SystemVision™ (VHDL-AMS), os diferentes cenários de simulação de um Sistema Elétrico de Distribuição (S-E-D)¹.
- 2.4.2 Desenvolver interfaces entre o Sistema Elétrico de Distribuição e o dispositivo FPGA, que será responsável pelo gerenciamento e controle deste sistema elétrico, conforme figura 9, para o qual se deve:
 - 2.4.2.1 Desenvolver uma interface para a importação de dados do Sistema Elétrico de Distribuição em ambiente VHDL-AMS. Isto significa que, depois que o Sistema Elétrico de Distribuição seja reproduzido integralmente na linguagem VHDL-AMS, deve-se desenvolver um mecanismo informático para que, em forma automatizada, sejam permitidas mudanças em determinados parâmetros de

¹ Neste capítulo 2, o termo “Sistema Elétrico de Distribuição” faz referência a uma rede ou alimentador primário de distribuição de energia elétrica (em média tensão) ou secundário (em baixa tensão), que pode vir de um alimentador real (baseado nos dados das características técnicas de um sistema elétrico existente) ou somente é simulado (com dados fictícios), que pretende-se modelar e simular utilizando a linguagem VHDL-AMS.

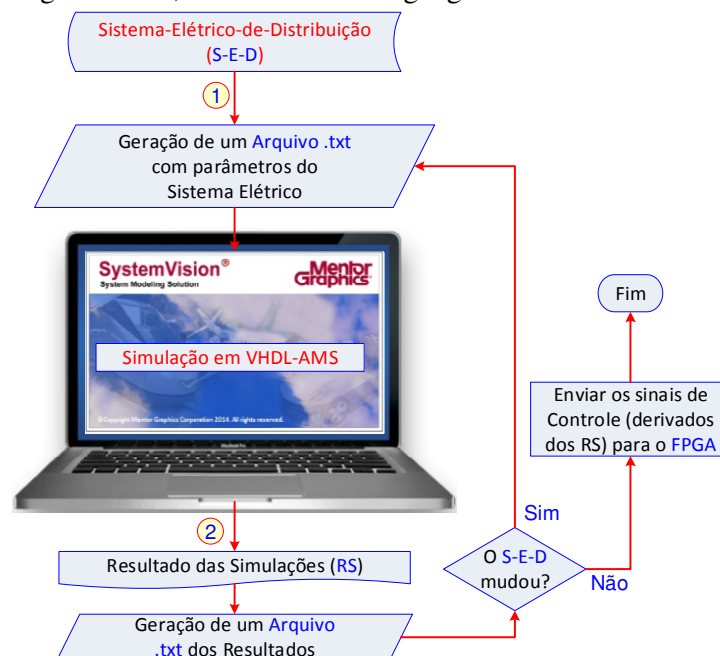
interesse do Sistema Elétrico de Distribuição, considerando-se o modelo desenvolvido em ambiente VHDL-AMS e, conseqüentemente, sejam realizadas múltiplas simulações no ambiente de SystemVision™ (VHDL-AMS), para posteriormente se criar um arquivo de texto com os resultados mais importantes escolhidos, derivados das simulações.

2.4.2.2 Desenvolver uma interface para o gerenciamento e controle do Sistema Elétrico de Distribuição usando um dispositivo FPGA. Desta forma, no ambiente SystemVision™ (VHDL-AMS) deve-se exportar todos os dados (resultados das simulações) provenientes do modelo do Sistema Elétrico de Distribuição para, desta maneira, permitir o controle e gerenciamento do sistema através de algoritmos digitais, implementados em um FPGA.

2.4.3 Implementar uma interface gráfica do usuário (**Graphical User Interface** - GUI), (VHDL-AMS) para o desenvolvimento de uma arquitetura de simulação em “tempo real” e controle para alimentadores de distribuição de energia elétrica.

Por outro lado, a interface para o gerenciamento e controle do Sistema Elétrico de Distribuição em ambiente FPGA será abordada no capítulo 3 e a interface gráfica do usuário em ambiente SystemVision™ (VHDL-AMS), será tratada no capítulo 4.

Figura 9- Arquitetura Base para a implementação da modelação de alimentadores de distribuição de energia elétrica, utilizando-se a linguagem VHDL-AMS.



Fonte: Próprio autor.

Na figura 9, observam-se os números (1) e (2), os quais indicam que as interfaces para a importação de dados do sistema elétrico de distribuição e para os dados simulados em VHDL-AMS serão melhor apresentadas nos capítulos seguintes desta tese, considerando-se a especificação do ambiente de simulação SystemVision™ (VHDL-AMS).

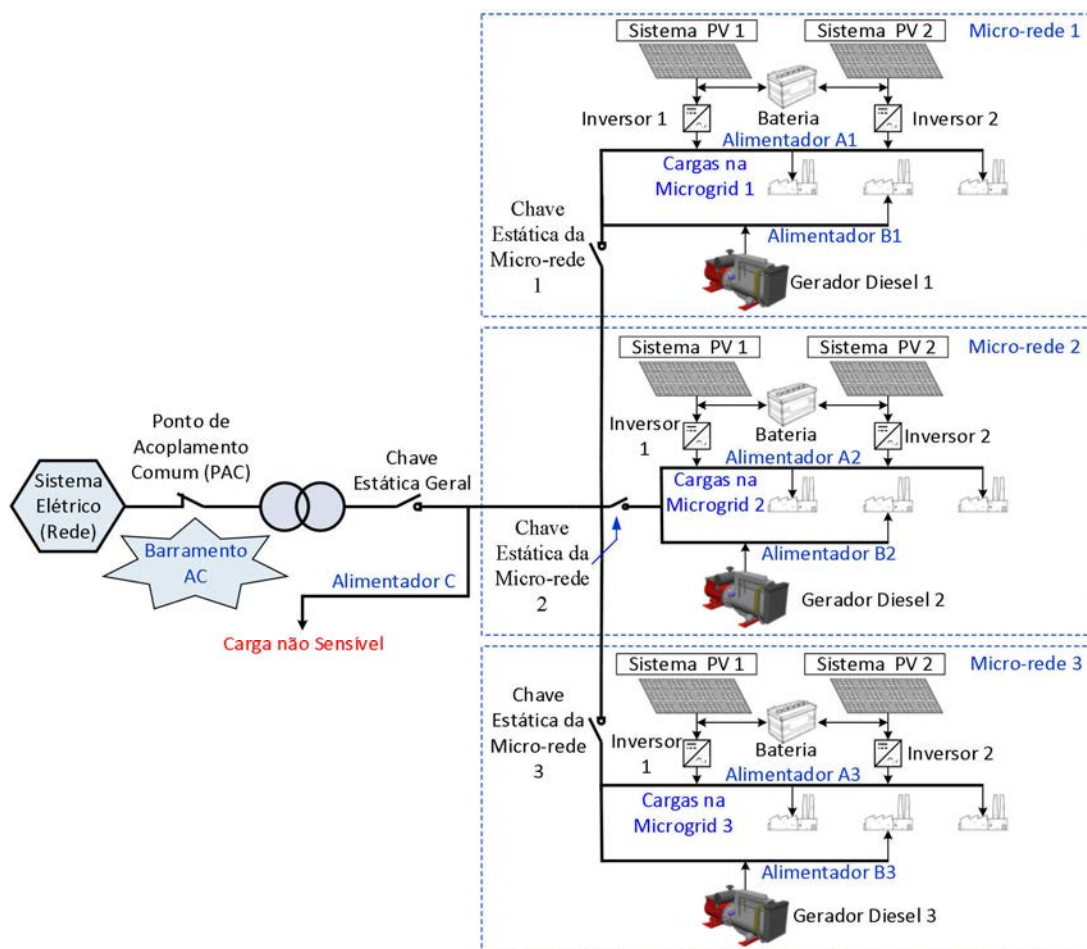
2.5 CENÁRIOS DE SIMULAÇÃO PARA DIFERENTES SISTEMAS ELÉTRICOS DE DISTRIBUIÇÃO, MODELADOS EM VHDL-AMS: ESTUDOS DE CASOS

Considerando-se as diversidades para os Sistemas Elétricos de Distribuição e de suas redes elétricas que o alimentam, diretamente ou por intermédio de seus ramais, os primários dos transformadores de distribuição do concessionário e/ou de consumidores e com o intuito de abranger todo o conceito de alimentador de distribuição de energia elétrica, os cenários de modelação e simulação de um Sistema Elétrico de Distribuição em VHDL-AMS serão desenvolvidos com duas abordagens: 1) Um enfoque de um alimentador secundário, ou seja, alimentação dos consumidores em baixa tensão (valor nominal RMS entre fases igual a 480 V), e 2) Um enfoque de um alimentador primário, ou seja, alimentação dos primários dos transformadores de distribuição do concessionário em média tensão (valor nominal RMS entre fases igual a 14,4 kV). Por outro lado, estes cenários de simulação serão desenvolvidos considerando os modelos básicos dos componentes elétricos em VHDL-AMS expostos no item 2.5.2.

2.5.1 Enfoque de simulação em baixa tensão (conceito estudo de caso CERTS)

Foi utilizado o sistema de múltiplas microrredes baseado no conceito CERTS (LASSETER et al., 2011). Este caso de estudo pode-se considerar como um caso construído a partir de um sistema real, testado experimentalmente. Desta forma, segundo (LASSETER et al., 2011) o conceito da microrrede CERTS, o qual foi proposto em 1998, integra um grupo de microgeradores e elementos de armazenamento com a capacidade de separar-se e isolar-se da rede de distribuição com pouca, ou nenhuma interrupção para as cargas. As características técnicas deste caso de estudo CERTS, representado na figura 10, estão descritas nas tabelas 4, 5 e 6.

Figura 10- Sistema de múltiplas microrredes baseado no conceito CERTS.



Fonte: (IBARRA HERNANDEZ FRANK et al., 2014).

TABELA 4- CONVENÇÕES PARA CADA MICRORREDE.

Variável	Significado da Variável	Valores Limites
P_{PV}	Potência Fotovoltaica	$-120 \text{ kW} \leq P_{PV} \leq 0 \text{ kW}$
P_{BAT}	Potência da Bateria	$-400 \text{ kW} \leq P_{BAT} \leq 400 \text{ kW}$
P_{DIE}	Potência do Gerador Diesel	$-200 \text{ kW} \leq P_{DIE} \leq 0 \text{ kW}$
P_{LOAD}	Potência Ativa da Carga	$0 \text{ kW} \leq P_{LOAD} \leq 270 \text{ kW}$
SOC	Estado de carga da bateria	$-1 \leq SOC \leq 1$
SSP	Posição da Chave Estática da Microrrede	ON ou OFF

Fonte: (IBARRA HERNANDEZ FRANK et al., 2014).

Segundo a figura 10, tem-se um sistema de baixa tensão de três microrredes com alimentadores de cargas sensíveis (alimentadores A e B) e um alimentador com cargas não sensíveis (alimentador C). O sistema fundamentado no conceito CERTS tem duas unidades de geração de energia solar como fontes de energias renováveis, as quais são instaladas nos

alimentadores A (A1, A2 e A3) e uma unidade de geração térmica (um gerador a diesel), instalado nos alimentadores B (B1, B2 e B3). Além do anterior, pode se observar que cada microrrede pode ficar ilhada do resto do sistema elétrico por meio das chaves estáticas de microrrede, assim como também todo o sistema das três microrredes pode permanecer isolado do sistema elétrico de distribuição do concessionário através da chave estática geral ou do ponto de acoplamento comum. Portanto, para o sistema elétrico global adotou-se a seguinte convenção, segundo a tabela 5.

TABELA 5- CONVENÇÕES PARA O SISTEMA ELÉTRICO CERTS GLOBAL.

Variável	Significado da Variável	Valores Limites
V_{L-L}	Tensão AC de Linha Nominal do Sistema	480 V
PLC	Potência Ativa Total no Alimentador C	$0 \text{ kW} \leq \text{PLC} \leq 360 \text{ kW}$
GSSP	Posição da Chave Estática Geral	OFF
PCC	Posição do Ponto de Acoplamento Comum	ON ou OFF
PT	Exportação de Potência desde o Sistema das Microrredes (Microrredes 1, 2 e 3)	$-300 \text{ kW} \leq \text{PT} \leq 2160 \text{ kW}$

Fonte: (IBARRA HERNANDEZ FRANK et al., 2014).

Com base na informação apresentada no artigo (IBARRA HERNANDEZ et al., 2013), utilizando o SystemVision™ (VHDL-AMS), decorrente desta tese, é modelado e simulado um módulo fotovoltaico o qual é implementado no caso de estudo da figura 10. As características do módulo fotovoltaico de cada microrrede estão descritas na tabela 6, sendo o módulo fotovoltaico modelado e simulado usando VHDL-MAS, considerando-se os mesmos estão sempre no ponto de máxima potência (hipótese simplificadora para o estudo), conforme dados da tabela 7.

TABELA 6- CARACTERÍSTICAS DO MÓDULO FOTOVOLTAICO DE CADA MICRO-REDE.

Variável	Significado da Variável	Valores
V_{pvOC}	Tensão DC de Circuito Aberto	44,97 V
V_{pvPmax}	Tensão DC de Máxima Potência	21,35 V
I_{pvPmax}	Corrente DC de Máxima Potência	2,62 A
I_{pvSC}	Corrente DC de curto circuito	4,90 A
P_{max}	Máxima Potência	56,00 W
R_{sPmax}	Resistência para Máxima Potência	8,14 Ω
N_{pv}	Número de Módulos Fotovoltaicos em Cada Microrrede (conectados em série e em paralelo)	2136

Fonte: (IBARRA HERNANDEZ FRANK et al., 2014).

TABELA 7- CARACTERÍSTICAS DO MÓDULO FOTOVOLTAICO DE CADA MICRORREDE.

Variável	Significado da Variável	Valores	% de Erro
VpvOC	Tensão DC de Circuito Aberto	44,99 V	0.039%
VpvPmax	Tensão DC de Máxima Potência	22,50 V	5.363%
IpvPmax	Corrente DC de Máxima Potência	2,50 A	4.439%
IpvSC	Corrente DC de curto circuito	4,93 A	0.665%
Pmax	Máxima Potência	56,16 W	0.278%

Fonte: (IBARRA HERNANDEZ FRANK et al., 2014).

2.5.2 Enfoque de simulação em média tensão (Estudo de caso baseado em rede Real)

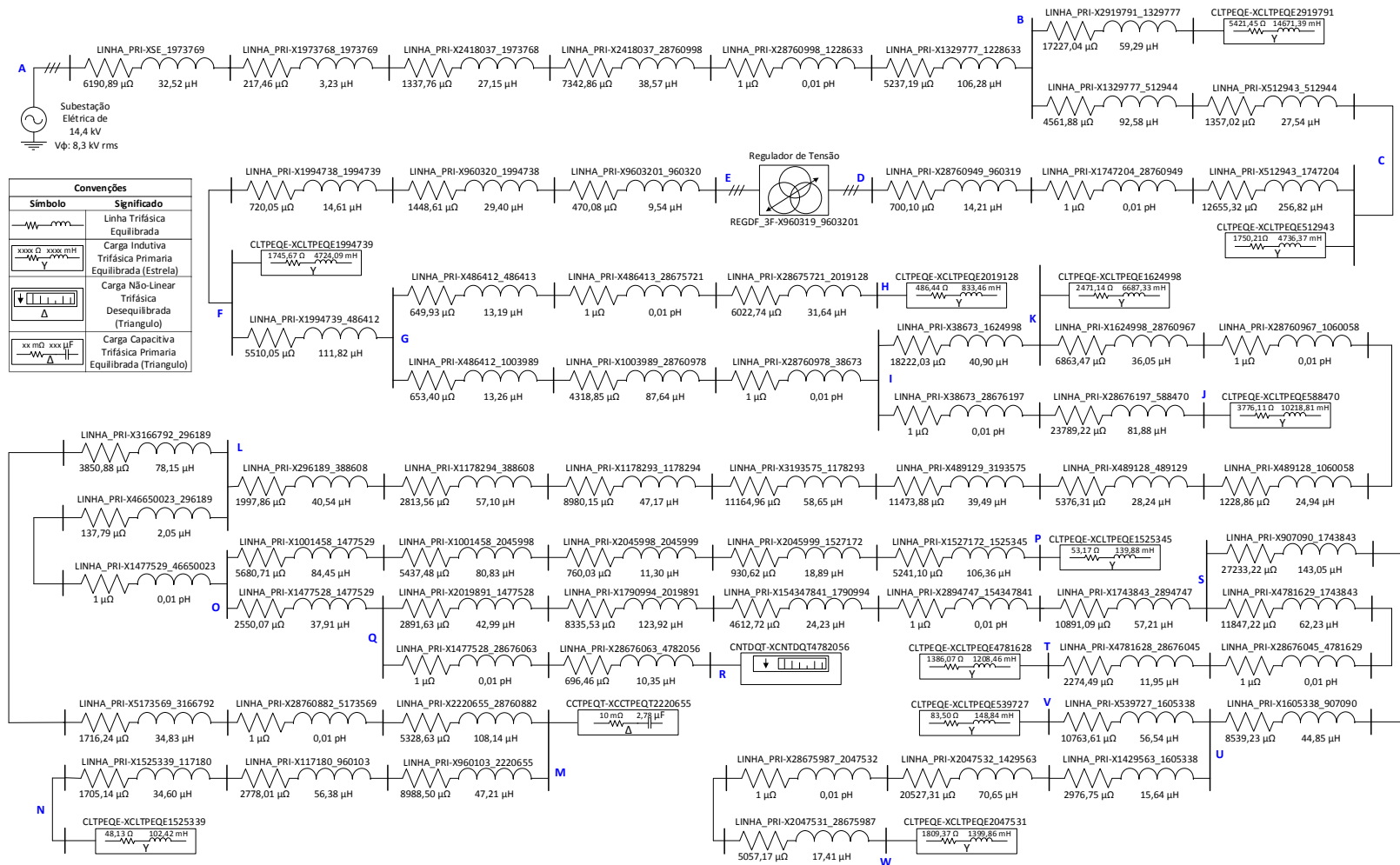
Para as simulações feitas em VHDL-AMS em média tensão, foi selecionado um alimentador com 14,4 kV de tensão de linha nominal, considerando-se um cenário típico real de uma distribuidora nacional de energia elétrica, no qual, segundo este caso particular analisado, aplicou-se uma ferramenta de redução da rede e concentração de cargas, resultando em um código escrito em formato texto, compatível com o Orcad/Pspice (conforme Apêndice E). A partir deste conjunto de dados da rede em formato texto, esboça-se o circuito esquemático da figura 11.

O sistema elétrico da figura 11 apresenta as seguintes características técnicas:

- Uma fonte trifásica não equilibrada e com distorções harmônicas.
- Um regulador de tensão trifásico em configuração Delta-fechado com um comutador de TAP's (8 posições aditivas e 8 subtrativas), havendo uma variação total de tensão de 19,5 % nas 16 posições.
- Sessenta e seis (66) segmentos de linhas trifásicas com parâmetros obtidos de suas constituições físicas reais (linhas compactas ou convencionais).
- Onze (11) cargas indutivas trifásicas referidas para o primário (média tensão) equilibradas e em estrela, resultantes das parcelas lineares do carregamento da rede.
- Uma carga não linear trifásica desequilibrada, refletida para o primário (média tensão).
- Um Banco de Capacitores em Delta (triângulo), trifásico e equilibrado, constituindo-se como elemento adicional de regulação de tensão para a rede.

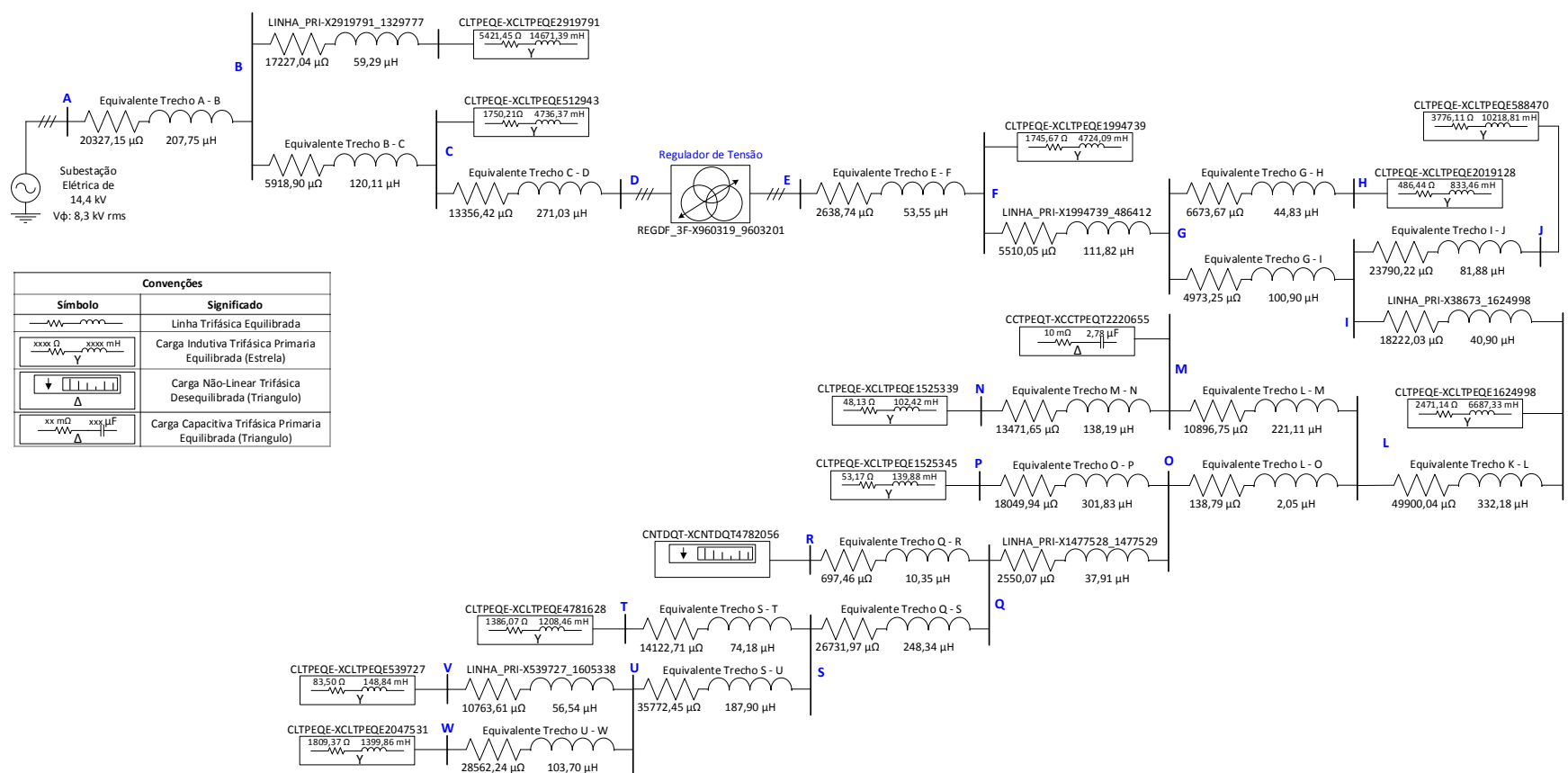
Na figura 12 é desenhado o alimentador da figura 11 com as impedâncias das linhas somadas (sistema elétrico reduzido), para os segmentos sem carregamento. Na figura 13 é apresentado o diagrama unifilar do mesmo alimentador da figura 11, tendo-se considerado que o desenho das linhas elétricas é proporcional ao comprimento real destas.

Figura 11- Alimentador de uma rede de distribuição primária de 14,4 kV (Caso de Estudo baseado em rede real de distribuição).



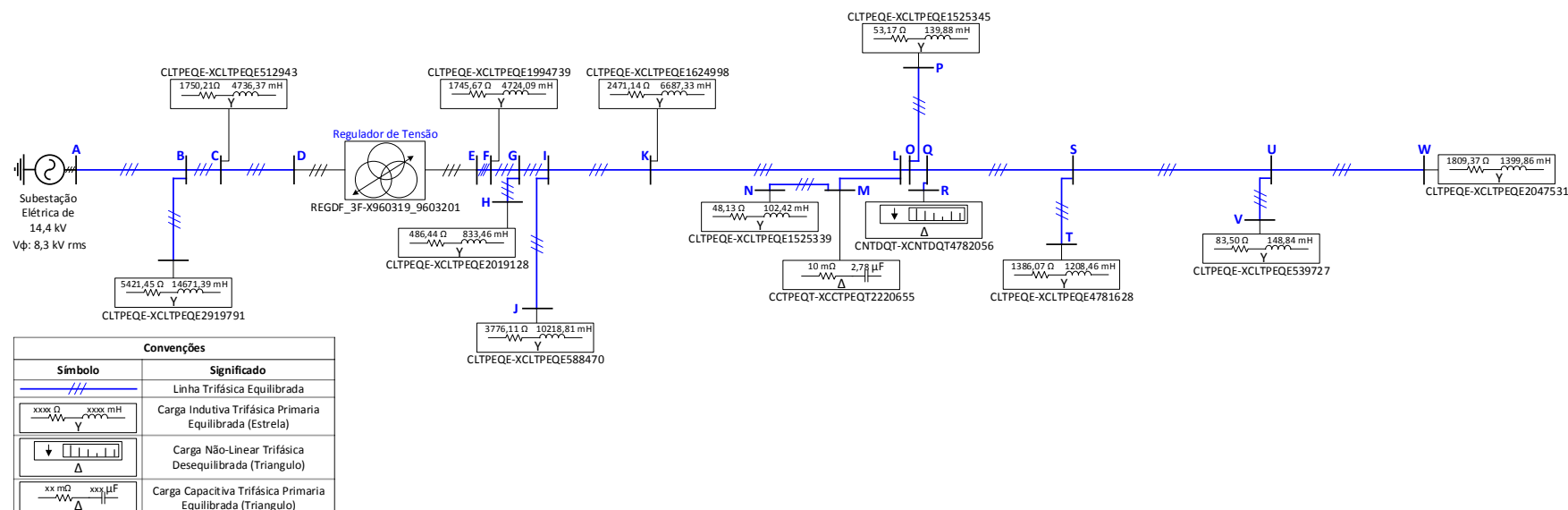
Fonte: Próprio autor.

Figura 12- Alimentador de uma rede de distribuição primária de 14,4 kV (com as impedâncias das linhas somadas, para os segmentos de barras sem carregamento).



Fonte: Próprio autor.

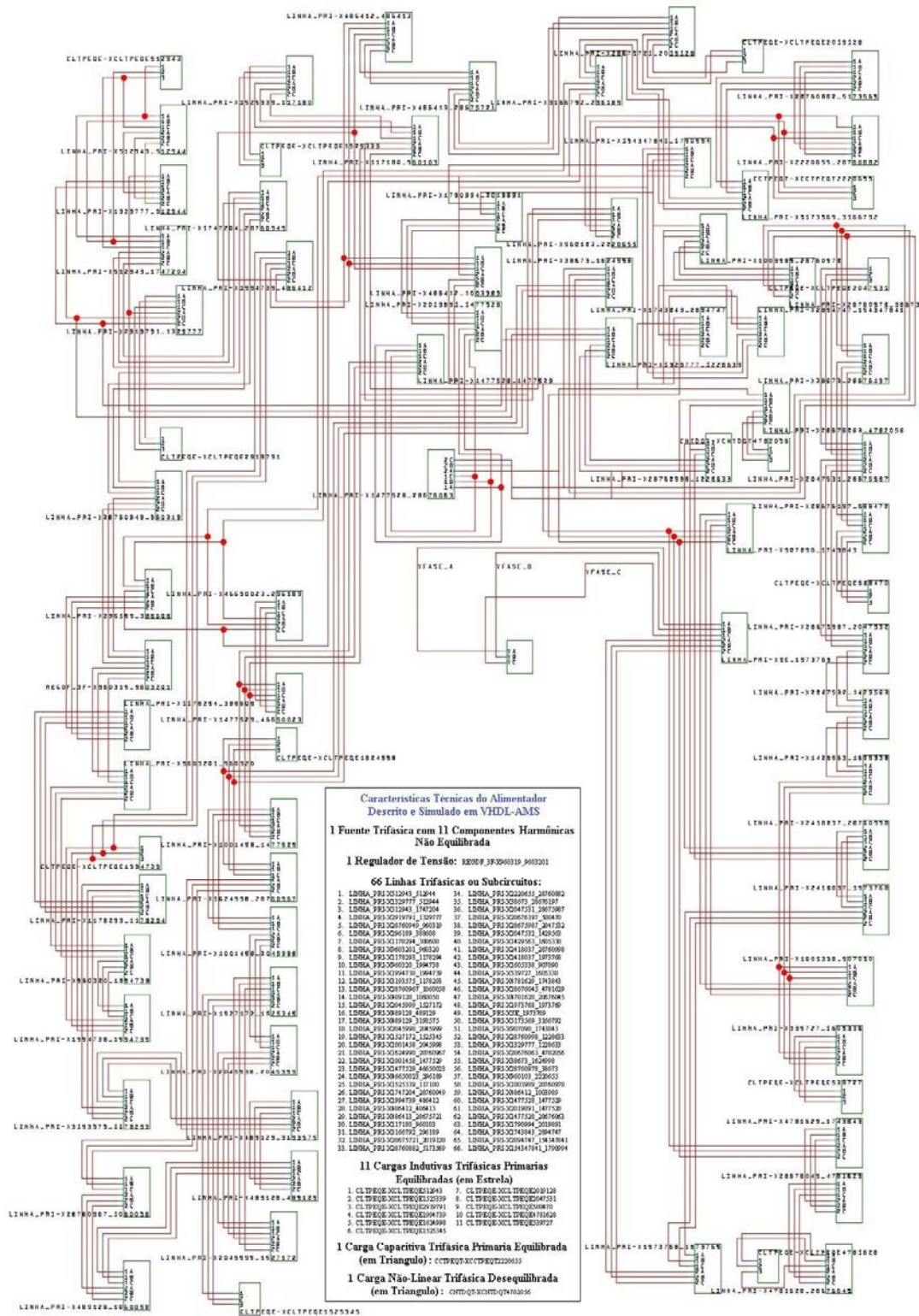
Figura 13- Diagrama unifilar de um alimentador de uma rede de distribuição primária de 14,4 kV: Estudo de Caso reduzido.



Fonte: Próprio autor.

Dada a complexidade da modelação e simulação em VHDL-AMS do sistema elétrico de distribuição da figura 11, na figura 14 apresenta-se o projeto para a modelação e simulação em ambiente VHDL-AMS deste alimentador, desenvolvido na plataforma do software SystemVision™, considerando-se as reduções propostas na figura 12.

Figura 14- Projeto para a modelação e simulação em ambiente VHDL-AMS do alimentador estudo de caso da figura 11.



Fonte: Próprio autor usando a plataforma do Software SystemVision.

2.6 INTERFACE PARA A IMPORTAÇÃO DE DADOS NO SISTEMA ELÉTRICO DE DISTRIBUIÇÃO EM AMBIENTE VHDL-AMS

A plataforma computacional empregada para a utilização da linguagem VHDL-AMS é o software licenciado SystemVision™ (fornecido pela empresa Mentor Graphics®). SystemVision™ é um sistema informático que fornece uma solução de modelação com a capacidade para projetar e simular sistemas de sinal misto ou tecnologia mista, especificamente no sistema operacional Microsoft Windows. SystemVision™ integra os seguintes recursos: 1). Uma captura esquemática baseada em uma interface gráfica do usuário (**Graphical User Interface – GUI**); 2) Um ambiente de simulação de modelos e técnicas de modelação de VHDL-AMS, VHDL, SPICE e C; e 3) Análise de formas de onda em uma abrangente simulação.

Considerando a utilização da linguagem VHDL-AMS no projeto da arquitetura de simulação em “tempo real” e controle para alimentadores de distribuição de energia elétrica, a interface para a importação de dados no Sistema Elétrico de Distribuição em Ambiente VHDL-AMS, conforme figura 15, tem os seguintes objetivos:

- Executar, de forma automática, a inserção dos dados derivados das mudanças do Sistema Elétrico de Distribuição em um sistema elétrico idêntico, desenvolvido em ambiente VHDL-AMS. Embora estes dados normalmente correspondam aos valores de todas as cargas pertencentes ao sistema elétrico, ou seja, os valores de resistência e indutância, podem-se inserir (ou modificar) qualquer outra grandeza do sistema de distribuição desenvolvido em VHDL-AMS.
- Após a inserção das mudanças do Sistema Elétrico de Distribuição na plataforma VHDL-AMS, a interface deve executar, de forma automática, as diferentes simulações derivadas destas variações (cenários).
- Depois de concluídas as simulações, a interface deve executar, também de forma automática, a produção e armazenamento das grandezas de interesse resultantes das simulações que servirão como entradas para o algoritmo de controle digital da rede a ser implementado no dispositivo FPGA, com a finalidade de tornar a microrrede mais “inteligente”.

Desde o ponto de vista da licença do SystemVision™, existem diferentes níveis de capacidades licenciadas que correspondem aos seguintes pacotes de capacidades operacionais:

- **Entrada (Entry)** - Destinado ao conceito de exploração, em que o usuário pode fazer uma exploração do design básico ou inicial. Este pacote inclui as capacidades

fundamentais para sinais mistos ou simulação multidisciplinar e análise. Inclui: Schematic entry, Analog simulation, Unlimited capacity, Digital VHDL simulation, Mixed-signal simulation, SPICE model support, Datasheet curve modeler, VHDL-AMS language support and Waveform analysis and measurement.

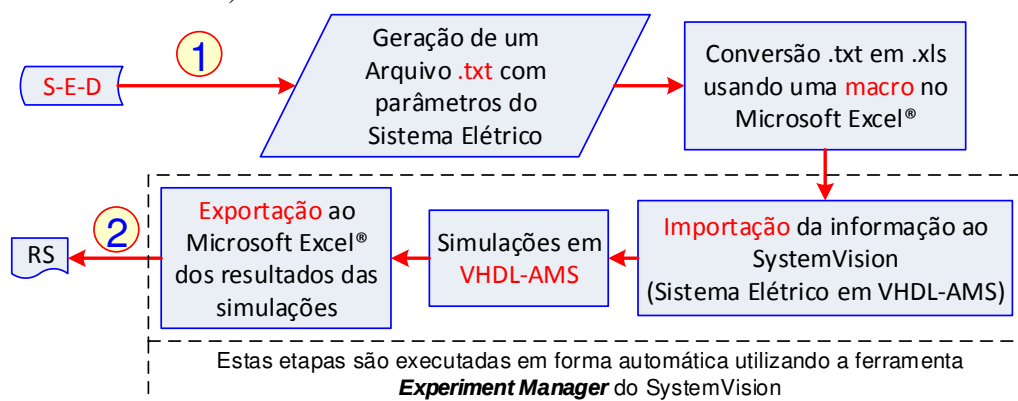
- **Paramétrico (*Parametric*)** - Destinado a ajustar um projeto para a variabilidade do mundo real em componentes, ambiente e/ou condições de operação. Este pacote contempla o pacote de entrada, adicionando capacidades de análise paramétrica poderosas para ajudar a identificar os ajustes de parâmetros. Também está incluso no pacote de entrada: Scripting automation support, Advanced waveform analysis and measurement, SPICE parameter sweeps, VHDL-AMS parameter sweeps, Parametric analysis, ***Experiment Manager*** e Multi-core parallel processing.
- **Integração (*Integration*)** - Destinado à integração de sistemas no qual o usuário precisa analisar um sistema multidisciplinar em um ambiente multiferramenta. Este pacote contempla o pacote paramétrico adicionando bibliotecas de fornecedores ASIC/FPGA e inclui as capacidades de conexão básicas de SystemVision™, que conecta SystemVision™ para C /C++, Simulink, o LabVIEW e outras plataformas. Além disso, também temos no pacote paramétrico: Digital Verilog simulation, ModelSim Digital Library support, SystemVision conneXion Application Sequencer, Any three conneXion clients (per session), Embedded software support, MATLAB Simulink integration and National Instruments LabVIEW integration.

Segundo a figura 15, é possível notar que para uma importação de dados do Excel, ou arquivo txt para um modelo feito em VHDL-AMS precisa-se da ferramenta “***Experiment Manager***”. Para o desenvolvimento dos anteriores objetivos, a interface para a importação de dados no sistema de distribuição de energia elétrica, desenvolvido em ambiente VHDL-AMS, deve ter os seguintes requerimentos computacionais:

- Desde o ponto de vista da licença do SystemVision, deve-se ter como mínimo o pacote de capacidade operacional paramétrico.
- Deve-se ter um sistema operacional Windows XP Pro (SP1, SP2), Windows 7 ou 8.
- Deve-se ter o Microsoft Excel (Office 2003 ou posterior) de **32 bits** instalado no computador onde está rodando SystemVision™, que possui as ferramentas de ***visual basic*** para a manipulação de dados externos.

- Recomenda-se fortemente a utilização de um processador de 2,50 GHz (ou superior) com uma memória RAM de 8,00 GB (ou superior) e um disco rígido de 250 GB (ou superior).

Figura 15- Interface para a importação de dados da rede (mudança de parâmetros em ambiente VHDL-AMS).



Fonte: Próprio autor.

2.7 RESULTADOS OBTIDOS NOS CENÁRIOS DE SIMULAÇÃO COM O USO DA INTERFACE PARA A IMPORTAÇÃO DE DADOS DO SISTEMA ELÉTRICO DE DISTRIBUIÇÃO EM AMBIENTE VHDL-AMS

Como apresentado no item 2.5, os cenários de modelação e simulação para o Sistema Elétrico de Distribuição em VHDL-AMS foram desenvolvidos com duas abordagens: 1) Um exemplo de caso para um alimentador secundário (Baixa Tensão – BT), utilizando um exemplo do conceito CERTS e 2) Um exemplo de caso para um alimentador primário (Média Tensão - MT), baseado em um alimentador real de distribuição. Neste contexto, os resultados obtidos nos cenários de simulação com o uso da interface para a importação de dados do Sistema Elétrico de Distribuição em Ambiente VHDL-AMS serão categorizados separadamente em BT e MT, assim:

2.7.1 Resultados das simulações em baixa tensão (Caso CERTS)

Segundo o enfoque de Simulação em Baixa Tensão), foi simulada a rede elétrica da figura 10, utilizando o conceito da implementação da linguagem VHDL-AMS (figura 9) e a interface para a importação de dados à plataforma VHDL-AMS (figura 15). As características técnicas do sistema elétrico simulado estão descritas nas tabelas 4, 5 e 6.

Para as simulações, segundo a figura 9, no Sistema Elétrico de Distribuição somente mudam os dados das cargas totais nas três microrredes e no alimentador C, ou seja, os valores de resistência e indutância de todo o sistema elétrico, de acordo com as tabelas 8 e 9. Os valores de resistência e indutância de todo o sistema elétrico foram escolhidos considerando (VANDOORN et al., 2011), que discorre sobre as redes de distribuição de baixa tensão, portanto, as linhas são principalmente, resistivas nas microrredes. Os resultados mais relevantes destas simulações são representados nas figuras 16, 17 e 18.

TABELA 8- CARGAS TOTAIS NAS MICRORREDES NA FIGURA 10 (CONCEITO CERTS)

Cargas Totais no Sistema de múltiplas microrredes baseado no Conceito CERTS										
Simulações	Carga Total	Microrrede 1			Microrrede 2			Microrrede 3		
		Fase A	Fase B	Fase C	Fase A	Fase B	Fase C	Fase A	Fase B	Fase C
1	R [Ω]	0,0001	0,0001	0,0001	0,0001	0,0001	0,0001	0,0001	0,0001	0,0001
	L [H]	0,00004	0,00004	0,00004	0,00004	0,00004	0,00004	0,00004	0,00004	0,00004
2	R [Ω]	0,001	0,001	0,001	0,001	0,001	0,001	0,001	0,001	0,001
	L [H]	0,0004	0,0004	0,0004	0,0004	0,0004	0,0004	0,0004	0,0004	0,0004
3	R [Ω]	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01
	L [H]	0,004	0,004	0,004	0,004	0,004	0,004	0,004	0,004	0,004
4	R [Ω]	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1
	L [H]	0,04	0,04	0,04	0,04	0,04	0,04	0,04	0,04	0,04
5	R [Ω]	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	L [H]	0,4	0,4	0,4	0,4	0,4	0,4	0,4	0,4	0,4
6	R [Ω]	10,0	10,0	10,0	10,0	10,0	10,0	10,0	10,0	10,0
	L [H]	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0
7	R [Ω]	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0
	L [H]	40,0	40,0	40,0	40,0	40,0	40,0	40,0	40,0	40,0
8	R [Ω]	1000,0	1000,0	1000,0	1000,0	1000,0	1000,0	1000,0	1000,0	1000,0
	L [H]	400,0	400,0	400,0	400,0	400,0	400,0	400,0	400,0	400,0

Fonte: Próprio autor.

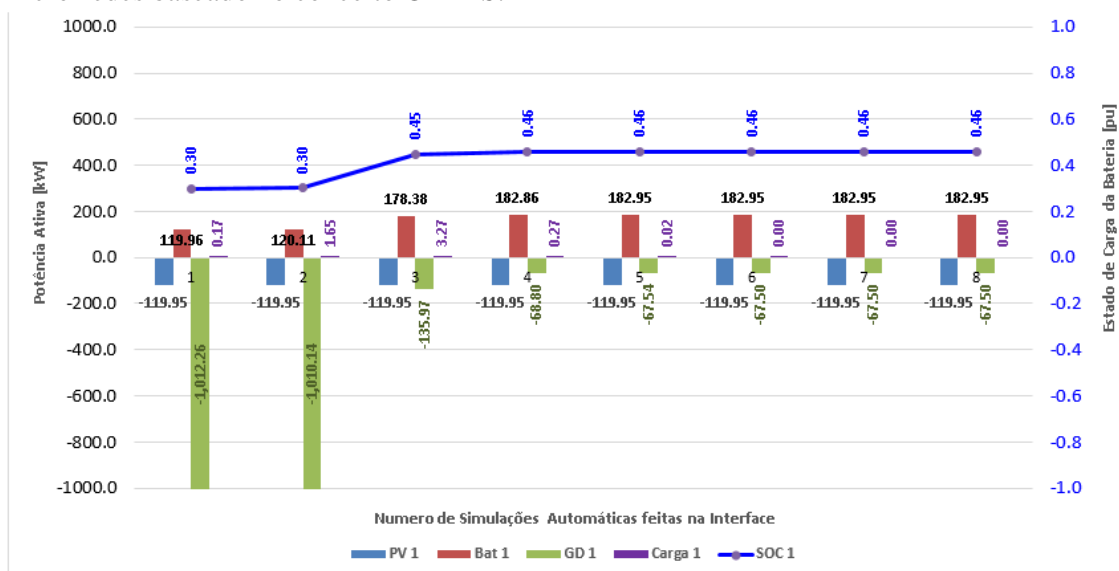
TABELA 9- CARGAS TOTAIS NO ALIMENTADOR C NA FIGURA 10 (CERTS).

Número de Simulações	Carga Total	Alimentador C		
		Fase A	Fase B	Fase C
1	R [Ω]	0,0001	0,0001	0,0001
	L [H]	0,0	0,0	0,0
2	R [Ω]	0,001	0,001	0,001
	L [H]	0,0	0,0	0,0
3	R [Ω]	0,01	0,01	0,01
	L [H]	0,004	0,004	0,004
4	R [Ω]	0,1	0,1	0,1
	L [H]	0,04	0,04	0,04
5	R [Ω]	1,0	1,0	1,0
	L [H]	0,4	0,4	0,4
6	R [Ω]	10,0	10,0	10,0
	L [H]	4,0	4,0	4,0
7	R [Ω]	100,0	100,0	100,0
	L [H]	40,0	40,0	40,0
8	R [Ω]	1000,0	1000,0	1000,0
	L [H]	400,0	400,0	400,0

Fonte: Próprio autor.

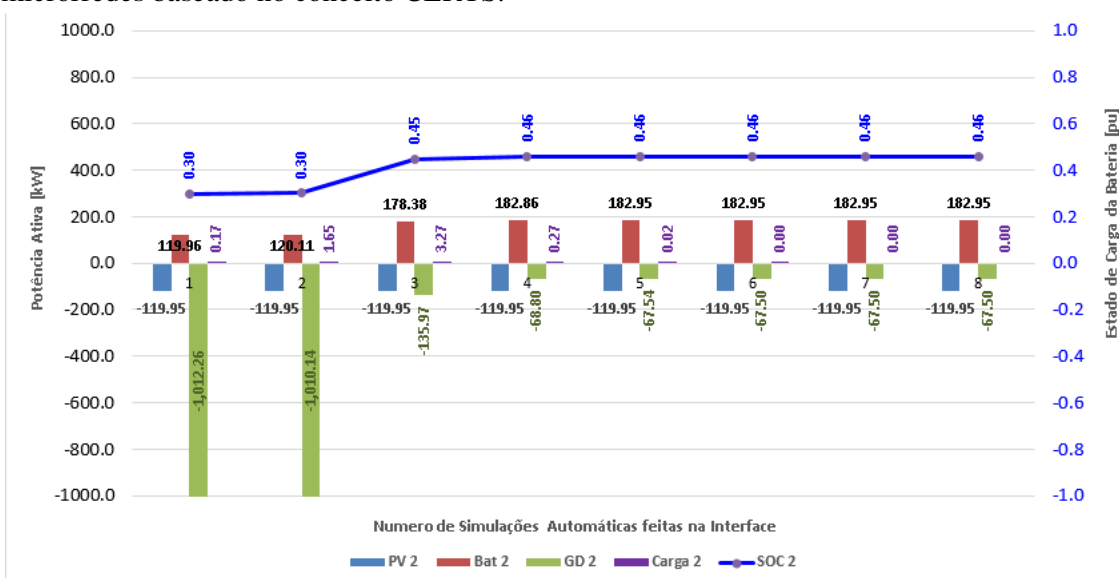
Nas figuras 16 até 18, para as simulações executadas, as potências negativas correspondem a fontes geradoras e as positivas obedecem às cargas.

Figura 16- Potência ativa consumida/gerada pela microrrede 1 no sistema de múltiplas microrredes baseado no conceito CERTS.



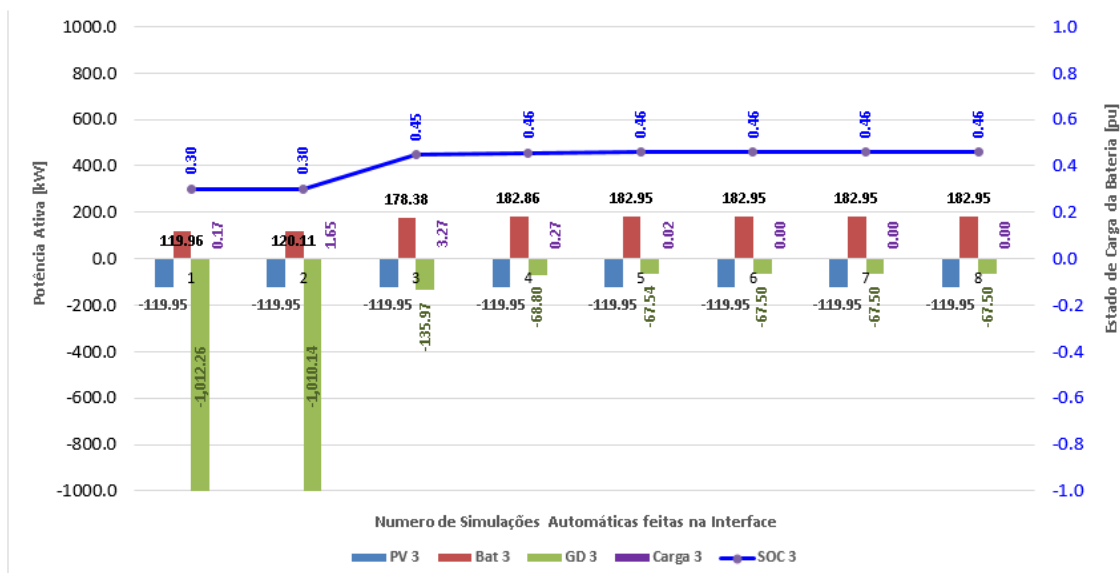
Fonte: Próprio autor.

Figura 17- Potência ativa consumida/gerada pela microrrede 2 no sistema de múltiplas microrredes baseado no conceito CERTS.



Fonte: Próprio autor.

Figura 18- Potência ativa consumida/gerada pela microrrede 3 no sistema de múltiplas microrredes baseado no conceito CERTS.



Fonte: Próprio autor.

Com a utilização da interface para a importação de dados à plataforma VHDL-AMS (figura 15) e após a execução das consequentes simulações da rede elétrica da figura 10, podemos estabelecer as seguintes conclusões para os resultados em baixa tensão:

- Considerando que em todas as simulações os módulos fotovoltaicos do sistema PV de cada microrrede se encontram todos em condição de fornecimento de máxima potência, foi utilizada a equação quadrática da potência vs. tensão para cada módulo fotovoltaico para calcular a máxima potência do sistema fotovoltaico (PV) de cada microgrid, segundo a qual a máxima potência PV disponível em cada microrrede é igual a 119,95 kW. De acordo com a tabela 6, o número de módulos fotovoltaicos em cada microrrede (conectados em série e em paralelo) deve ser igual a 2136, uma vez que a máxima potência de cada módulo fotovoltaico é igual a 56,16 W (tabela 7). Pelo exemplo anterior, a potência fotovoltaica disponível em cada microrrede e em todas as simulações é a máxima e igual a 119,95 kW (figuras 16, 17 e 18).
- Considerando que todas as simulações foram feitas com o sistema elétrico da figura 10 em modo ilhado, ou seja, com a posição da chave estática geral aberta, nas figuras 16, 17 e 18 pode-se observar que o cenário de simulação que oferece a máxima transferência de potência ativa às cargas de cada microrrede, assim como ao alimentador de carga não sensível (alimentador C), é a simulação número três (3). Os valores numéricos resultantes desta simulação 3 estão nas tabelas 10 e 11.
- Na tabela 11, pode-se observar que a simulação número três (3) apresenta as tensões RMS de fase do sistema mais próximas à tensão de fase nominal da microrrede. As porcentagens de variação em cada fase são: 2,41% (fase A), 3,32% (fase B) e 3,06% (fase C).

- No exemplo anterior, pode-se notar que a simulação 3 cumpre perfeitamente com os valores limites ou nominais do sistema elétrico simulado (ver tabelas 10 e 11). Pelos três pontos anteriores, a simulação 3 foi escolhida como uma ótima simulação para o funcionamento operacional da rede em baixa tensão da figura 10.
- Sabendo que nas simulações, as potências negativas correspondem a fontes geradoras e as positivas obedecem às cargas, segundo as figuras 16, 17 e 18, na simulação 3, cada bateria correspondente a cada microrrede se encontra em estado de carga com 45% da potência nominal desta.
- Embora existam infinitos cenários de simulação para a rede em baixa tensão da figura 10, os cenários escolhidos (expostos nas tabelas 8 e 9), têm como objetivo principal, avaliar e testar as ferramentas desenvolvidas para a simulação automática em VHDL-AMS, segundo o exposto nas figuras 9 e 15. De acordo com os resultados obtidos na rede de baixa tensão, o método exposto nas figuras 9 e 15 atinge os objetivos propostos para a tese, satisfatoriamente.

TABELA 10- CONVENÇÕES PARA CADA MICRORREDE VS. VALORES SIMULAÇÃO 3.

Variável	Significado da Variável	Valores Limites	Valores Simulação 3		
			Microrrede 1	Microrrede 2	Microrrede 3
P_{PV}	Potência Fotovoltaica	$-120 \text{ kW} \leq P_{PV} \leq 0 \text{ kW}$	-119,95 kW	-119,95 kW	-119,95 kW
P_{BAT}	Potência da Bateria	$-400 \text{ kW} \leq P_{BAT} \leq 400 \text{ kW}$	178,38 kW	178,38 kW	178,38 kW
P_{DIE}	Potência do Gerador Diesel	$-200 \text{ kW} \leq P_{DIE} \leq 0 \text{ kW}$	-135,97 kW	-135,97 kW	-135,97 kW
P_{LOAD}	Potência Ativa da Carga	$0 \text{ kW} \leq P_{LOAD} \leq 270 \text{ kW}$	3,27 kW	3,27 kW	3,27 kW
SOC	Estado de carga da bateria	$-0.1 \leq SOC \leq 1.0$	0,45	0,45	0,45
SSP	Posição da Chave Estática da Microrrede	ON ou OFF	ON	ON	ON

Fonte: Próprio autor.

TABELA 11- CONVENÇÕES PARA O SISTEMA ELÉTRICO GLOBAL VS. VALORES SIMULAÇÃO 3.

Variável	Significado da Variável	Valores Limites/Nominal	Valores Simulação 3		
			Fase A	Fase B	Fase C
V_{L-L}	Tensão RMS de Fase Nominal do Sistema	277,13 V	283,82	286,34	285,61
PLC	Potência Ativa Total no Alimentador C	$0 \text{ kW} \leq PLC \leq 360 \text{ kW}$	3,27 kW		
GSSP	Posição da Chave Estática Geral	ON ou OFF	OFF (Aberta)		
PCC	Posição do Ponto de Acoplamento Comum	ON ou OFF	ON (Fechado)		
PT	Exportação de Potência desde o Sistema das Microrredes (Microrredes 1, 2 e 3)	$-300 \text{ kW} \leq PT \leq 2160 \text{ kW}$	0,00 kW		

Fonte: Próprio autor.

2.7.2 Resultados das Simulações em Média Tensão (Rede baseada em Caso Real)

Segundo o enfoque de Simulação em Média Tensão), foi simulada a rede elétrica da figura 11, utilizando o conceito da implementação da linguagem VHDL-AMS (figura 9) e a interface para a importação de dados à plataforma VHDL-AMS (figura 15). As características técnicas do sistema elétrico simulado estão descritas no item 2.7.2 e nas figuras 11, 12, 13 e 14.

Para as simulações, segundo a figura 9, no Sistema Elétrico de Distribuição somente mudam os valores da posição dos TAPs no Regulador de Tensão no alimentador. De acordo com o anterior, para um tempo de simulação de 500 até 600 milissegundos, onde são observados os parâmetros para os cálculos em regime permanente:

- Foram derivadas as seguintes estatísticas de simulação em ambiente VHDL-AMS.

TABELA 12- ESTATÍSTICAS DERIVADAS DE UMA SIMULAÇÃO (DADOS ENTRE 500 E 600MS).

1. Número de nós: 959.	9. Número de blocos: 658.
2. Número de modelos: 7.	10. Número de tipos: 51.
3. Número de sinais: 1.	11. Número de constantes: 1347.
4. Número de quantidades: 1557.	12. Número de pinos: 444.
5. Número de passos: 13909.	13. Número de terminais: 1680.
6. Número de pacotes: 4.	14. Número de quantidades: 1557.
7. Número de instâncias: 570.	15. Tempo de CPU 2 min 5 s.
8. Número de subcircuitos: 81.	16. Passo de Tempo: 71,9 μ s

Fonte: Próprio autor.

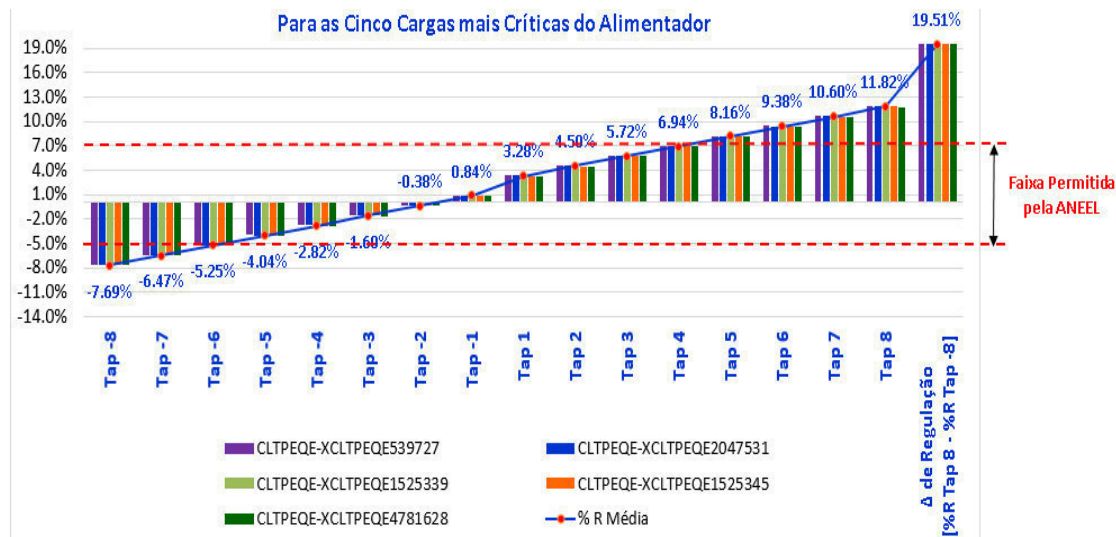
- Foram feitas 17 simulações automáticas em VHDL-AMS, segundo as variações dos TAPs no Regulador de Tensão para fazer uma análise da regulação de tensão em todas as cargas e assim, determinar as 5 cargas com a mais alta regulação de tensão (Figura 19), bem como as características da regulação de tensão para a carga com a pior regulação do alimentador (Tabela 13.)
- Foram feitas simulações adicionais para construir os perfis da tensão do alimentador (figuras 19, 20 e 21), sendo que as figuras 20 e 21 apresentam os perfis para as condições de TAPs consideradas originais (TAPs todos na posição 8), conforme dados adquiridos dos parâmetros de rede, assim como, para o cenário de simulação para a condição de melhor regulação (conforme se analisará nos capítulos seguintes, para o estudo de caso analisado e condições impostas para o controlador da rede), considerando-se os TAPs todos na posição -2.

TABELA 13- TESTES REALIZADOS EM VHDL-AMS COM MUDANÇAS DE TAP NO REGULADOR DE TENSÃO PARA A CARGA COM A PIOR REGULAÇÃO DE TENSÃO.

Posição dos TAPs no Regulador de Tensão			Tensão RMS na S/E			Carga com Pior Regulação			% Regulação de Tensão				
						CLTPEQE-XCLTPEQE539727			%R Inicial (Tap -8)		%R Final (Tap 8)		%Δ total
									-7,64%		11,86%		19,50%
TAP A	TAP B	TAP C	VA rms	VB rms	VC rms	VA rms	VB rms	VC rms	%R Fase A	%R Fase B	%R Fase C	%R Média	Δ por TAP
-8	-8	-8	8304,60	8295,10	8349,90	8939,40	8928,70	8988,90	-7,64%	-7,64%	-7,65%	-7,64%	1,22%
-7	-7	-7	8304,60	8295,10	8349,90	8838,60	8827,70	8887,00	-6,43%	-6,42%	-6,43%	-6,43%	1,22%
-6	-6	-6	8304,60	8295,10	8349,90	8737,00	8726,90	8784,90	-5,21%	-5,21%	-5,21%	-5,21%	1,22%
-5	-5	-5	8304,60	8295,10	8349,90	8636,20	8625,30	8683,50	-3,99%	-3,98%	-4,00%	-3,99%	1,22%
-4	-4	-4	8304,60	8295,10	8349,90	8535,00	8524,50	8,58E+03	-2,77%	-2,77%	-2,78%	-2,77%	1,22%
-3	-3	-3	8304,60	8295,10	8349,90	8433,80	8423,30	8480,30	-1,56%	-1,55%	-1,56%	-1,55%	1,22%
-2	-2	-2	8304,60	8295,10	8349,90	8332,50	8322,60	8378,40	-0,34%	-0,33%	-0,34%	-0,34%	1,22%
-1	-1	-1	8304,60	8295,10	8349,90	8231,70	8221,20	8276,80	0,88%	0,89%	0,88%	0,88%	1,22%
0	0	0	8304,60	8295,10	8349,90	8130,20	8120,60	8175,00	2,10%	2,10%	2,09%	2,10%	1,22%
1	1	1	8304,60	8295,10	8349,90	8029,10	8019,20	8073,20	3,32%	3,33%	3,31%	3,32%	1,22%
2	2	2	8304,60	8295,10	8349,90	7928,00	7917,80	7971,60	4,53%	4,55%	4,53%	4,54%	1,22%
3	3	3	8304,60	8295,10	8349,90	7826,90	7816,90	7869,80	5,75%	5,76%	5,75%	5,76%	1,22%
4	4	4	8304,60	8295,10	8349,90	7725,40	7715,90	7767,70	6,97%	6,98%	6,97%	6,98%	1,22%
5	5	5	8304,60	8295,10	8349,90	7624,20	7614,20	7666,10	8,19%	8,21%	8,19%	8,20%	1,22%
6	6	6	8304,60	8295,10	8349,90	7522,70	7513,50	7564,20	9,42%	9,42%	9,41%	9,42%	1,22%
7	7	7	8304,60	8295,10	8349,90	7421,50	7412,30	7462,20	10,63%	10,64%	10,63%	10,64%	1,22%
8	8	8	8304,60	8295,10	8349,90	7320,00	7311,00	7360,50	11,86%	11,86%	11,85%	11,86%	1,22%

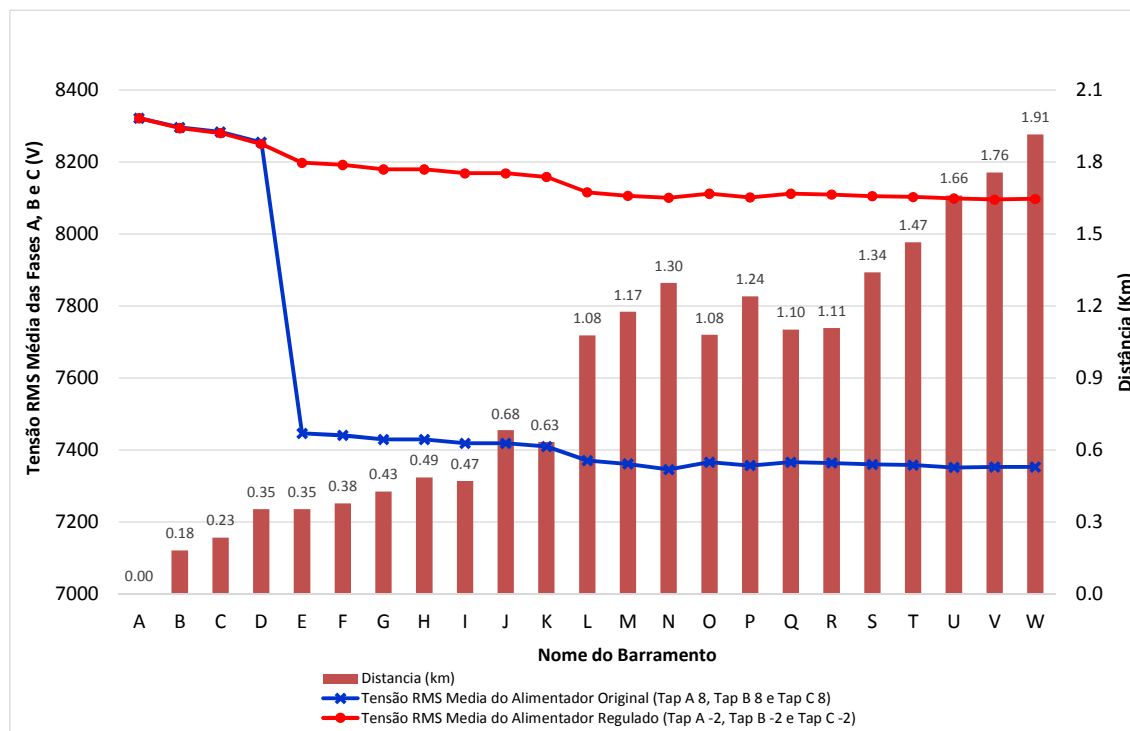
Fonte: Próprio autor.

Figura 19- Percentagem de regulação para as cinco cargas mais críticas do alimentador (com maior regulação de tensão).



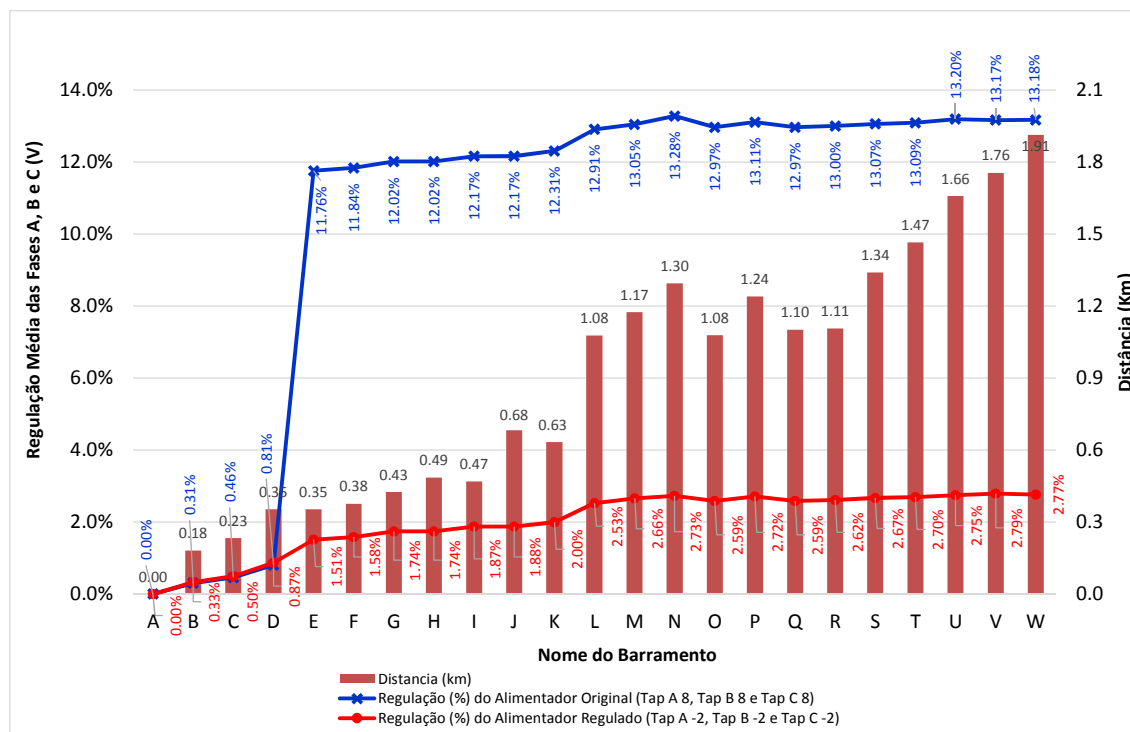
Fonte: Próprio autor.

Figura 20- Perfil de tensão do alimentador baseado nos testes realizados em VHDL-AMS.



Fonte: Próprio autor.

Figura 21- Regulação do alimentador baseados nos testes realizados em VHDL-AMS.



Fonte: Próprio autor.

2.7.3 Cenários de simulação do alimentador em VHDL-AMS para escolha do tempo de simulação para as mudanças de TAPs no regulador de tensão

De acordo com o item anterior, as simulações para os testes realizados em VHDL-AMS com as mudanças de TAPs no regulador de tensão para todas as cargas do alimentador (tabela 13 e figura 19) foram feitos considerando-se os valores eficazes de tensões calculados entre 500 e 600 milissegundos.

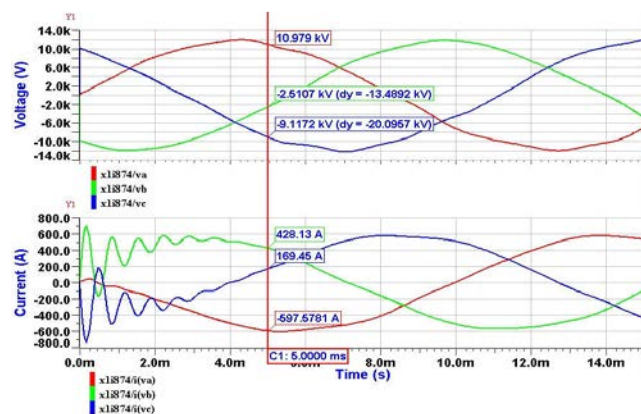
Lembrando que o caso de estudo em média tensão esquematizado na figura 11 foi obtido através de uma ferramenta de redução da rede e concentração de cargas, resultando em um código em Orcad/Pspice (Apêndice E). Este código em Orcad/Pspice para este estudo de, para fins de comparação dos resultados de simulação, tiveram seus dados calculados considerando-se o intervalo de simulação entre 500 até 525 milissegundos, baseados nas simulações representadas nas figuras 22, 23, 24 e 25, de acordo com a tabela 14, sendo que os testes realizados em VHDL-AMS com mudanças automáticas de TAPs no regulador de tensão foram realizados com dados de simulação no intervalo de 500 até 600 milissegundos, considerando as mínimas variações das tensões e correntes RMS comparadas com um período de simulação de cinco segundos (simulação base).

TABELA 14- ESCOLHA DO TEMPO DE SIMULAÇÃO PARA AS MUDANÇAS DE TAP'S.

Cenários de Simulação	Tensões e Correntes RMS da Subestação					
	Va	Vb	Vc	Ia	Ib	Ic
Simulação de 0 a 5 segundos (BASE)	8311,70	8289,60	8362,80	408,44	409,60	404,32
Simulação de 500 até 525 milissegundos	8304,70	8298,20	8345,90	408,28	409,92	404,38
% Variação respeito à BASE	-0,08%	0,10%	-0,20%	-0,04%	0,08%	0,01%
Simulação de 500 até 600 milissegundos	8311,70	8289,60	8362,80	408,40	409,64	404,40
% Variação respeito à BASE	0,00%	0,00%	0,00%	-0,01%	0,01%	0,02%

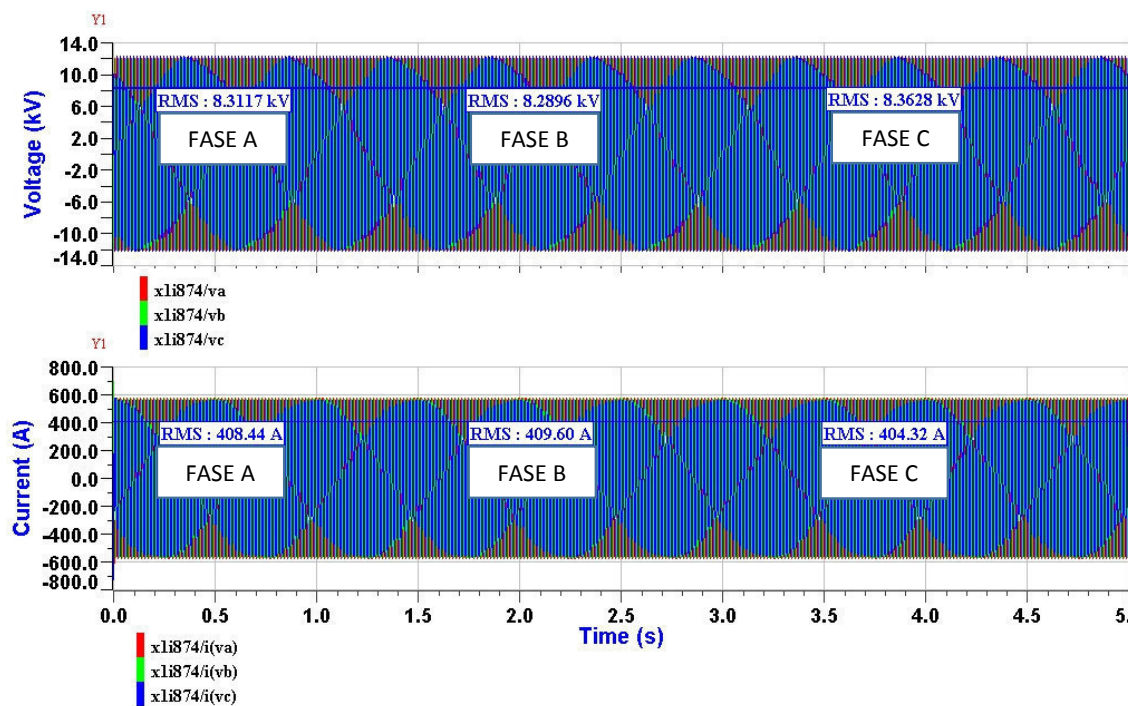
Fonte: Próprio autor.

Figura 22- Análise dos transitórios das tensões e correntes da subestação (de 0 até 15 milissegundos).



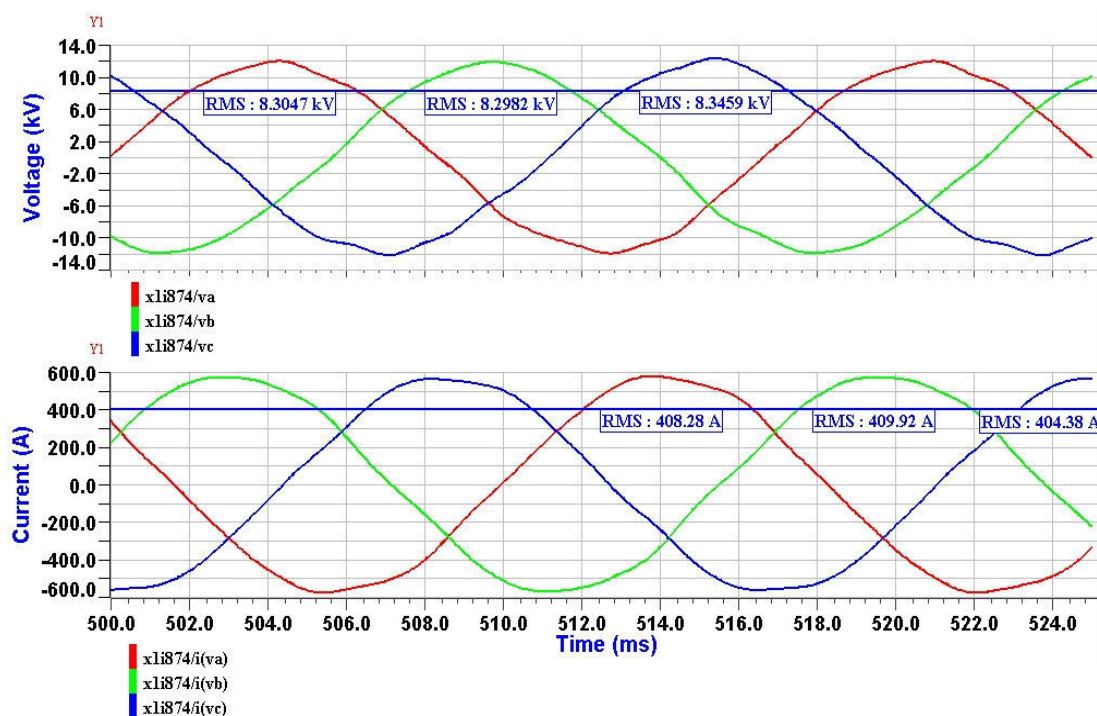
Fonte: Próprio autor.

Figura 23- Simulação em 5 segundos das tensões e correntes da subestação.



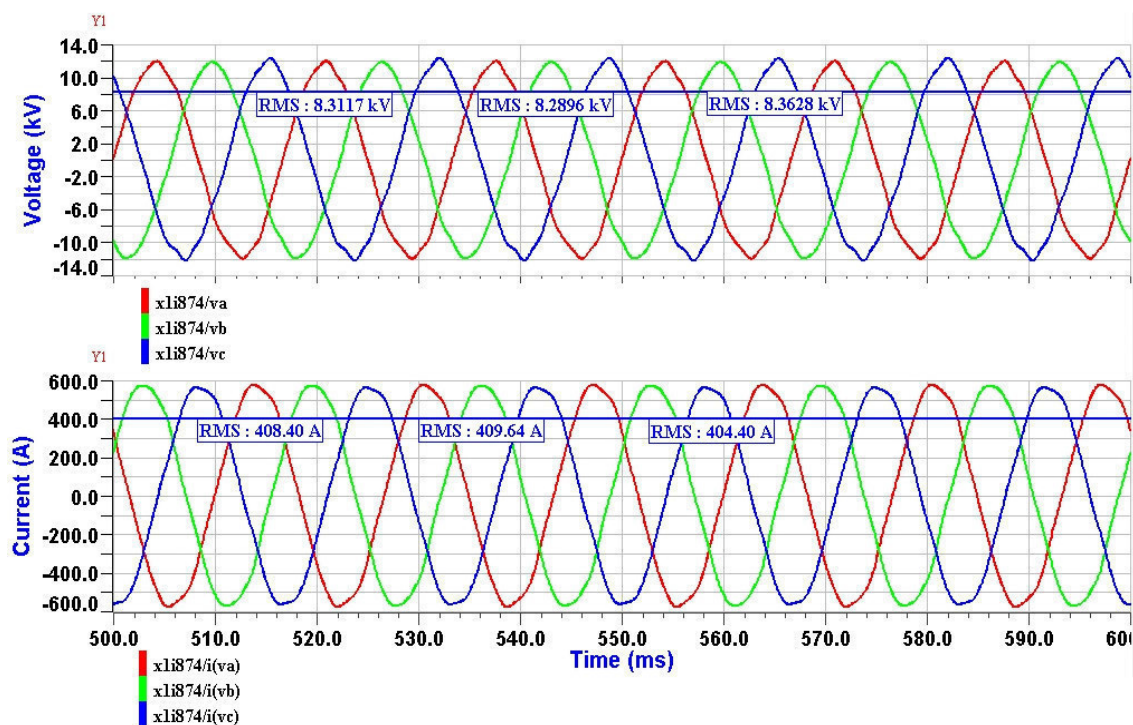
Fonte: Próprio autor.

Figura 24- Simulação de 500 até 525 milissegundos das tensões e correntes da subestação.



Fonte: Próprio autor.

Figura 25- Simulação de 500 até 600 milissegundos das tensões e correntes da subestação.



Fonte: Próprio autor.

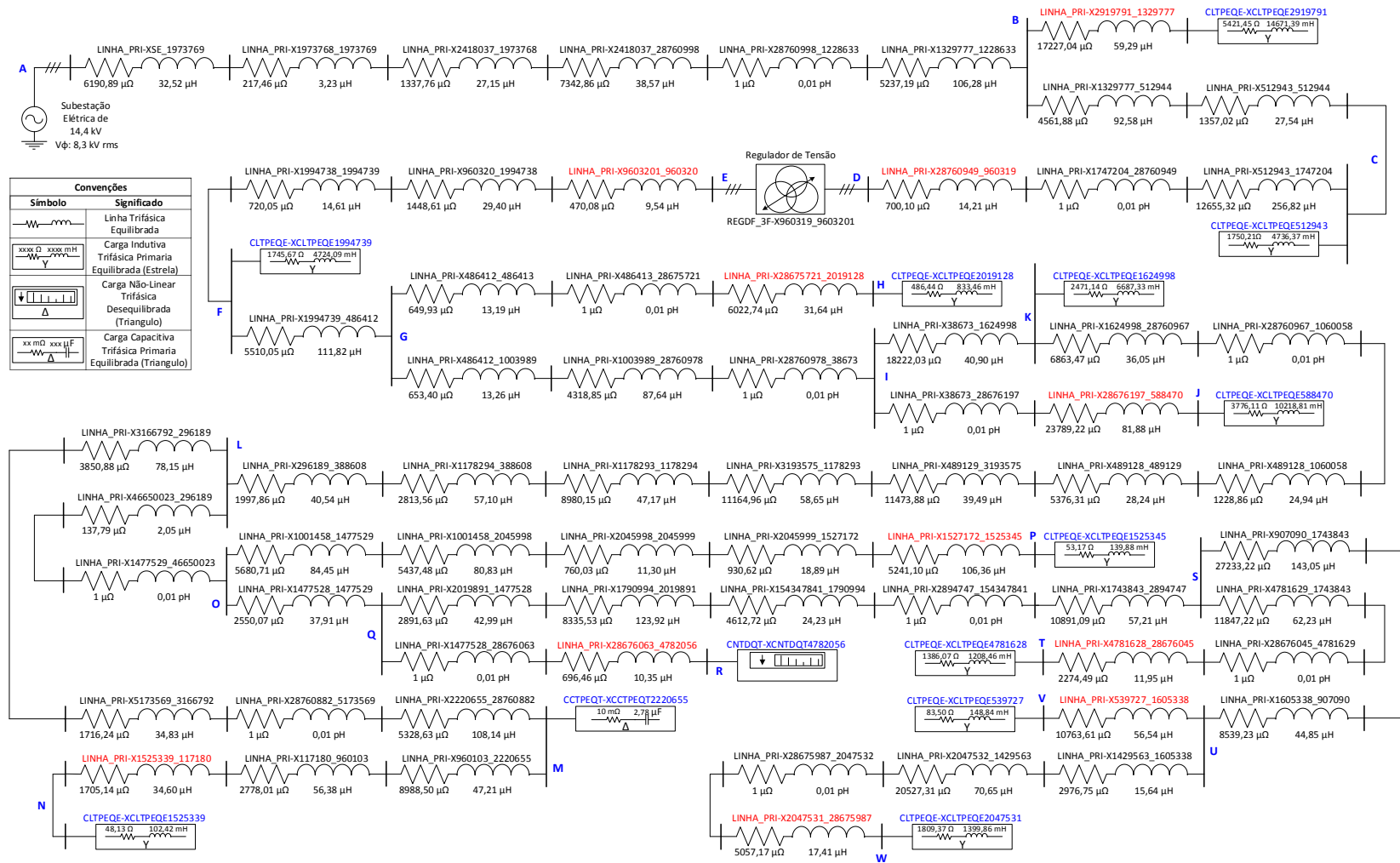
2.7.4 Análise comparativa dos resultados das simulações em VHDL-AMS de um alimentador de uma rede de distribuição primária de 14,4 kV

Com o objetivo de validar e avaliar o modelo do alimentador desenvolvido em VHDL-AMS da figura 14, dado seu projeto para a modelação e simulação em ambiente VHDL-MAS considerando-se a figura 11, foi feita uma análise comparativa entre os resultados obtidos em VHDL-AMS e as simulações obtidas em Orcad/Pspice.

Neste contexto, as seguintes observações são apresentadas:

- Manteve-se intacto o código Orcad/Pspice original sem nenhuma modificação. Foram feitas as simulações em VHDL-MAS, considerando-se os cálculos no intervalo desde 500 até 525 milissegundos e para o máximo nível do TAP do regulador de tensão (TAP A=8, TAP B=8 e TAP C=8).
- Baseados no item anterior, não foram criadas medições adicionais em Orcad/Pspice, com o objetivo de deixar o código do alimentador no seu formato original. Algumas simulações em Orcad/Pspice não aparecem nas comparações, porém, em VHDL-AMS foram projetadas as medições de tensão e corrente de 100% dos nós do circuito alimentador da figura 11.
- Considerando o esquema do alimentador, conforme figura 26, foram feitas todas as simulações das tensões e correntes em Orcad/Pspice e em VHDL-AMS das linhas que antecedem às cargas elétricas selecionadas com nomenclatura em **vermelho**. Nestas cargas elétricas, todas as tensões e correntes foram simuladas em VHDL-AMS e algumas destas em Orcad/Pspice (nomenclatura em **azul**).
- Também foram feitas as simulações das tensões (de fase) e correntes para 11 linhas (figuras 27 até 32 e, figuras 97 até 110), além das cargas concentradas para o sistema de distribuição estudo de caso (figuras 111 até 133). Com o intuito de preservar os resultados nos dois ambientes de simulação, estas figuras são apresentadas na formatação padronizada pelos Orcad/Pspice e em VHDL-AMS, respectivamente.

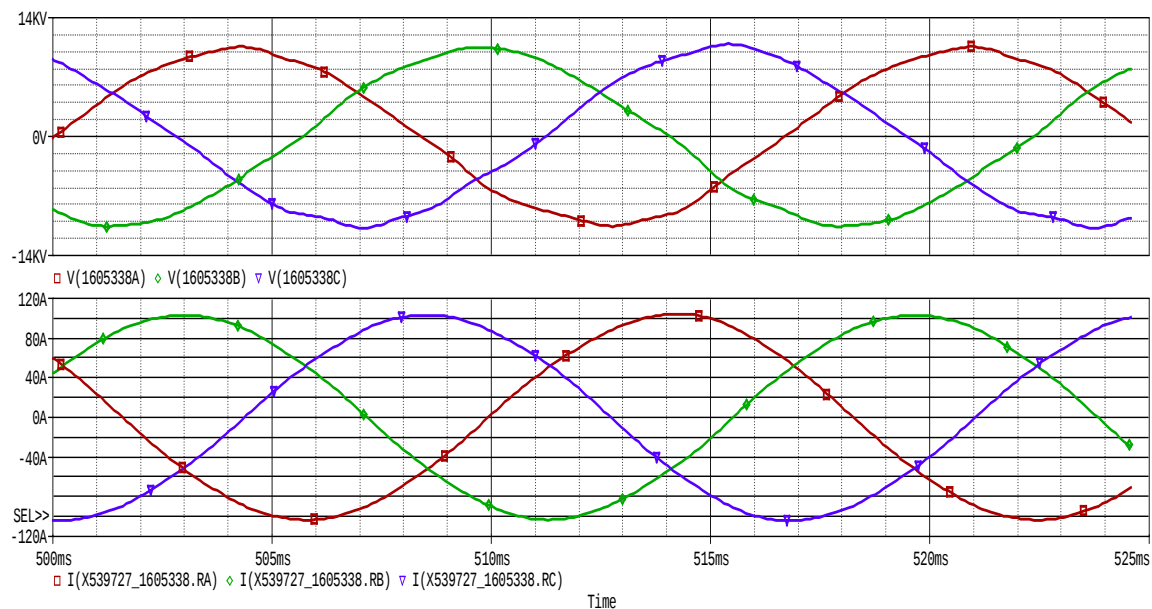
Figura 26- Alimentador de 14,4 kV com marcação das linhas e cargas a simular para a análise comparativa.



Fonte: Próprio autor.

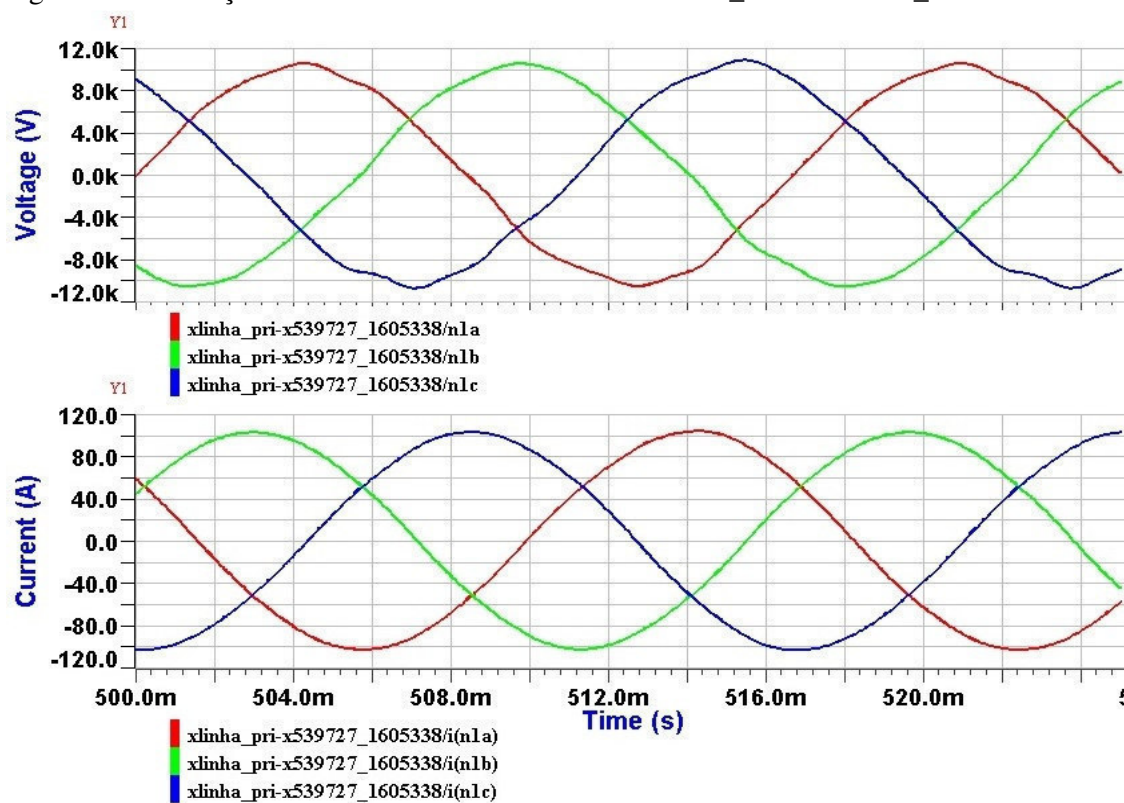
LINHA_PRI-X539727_1605338
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 27- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X539727_1605338.



Fonte: Próprio autor.

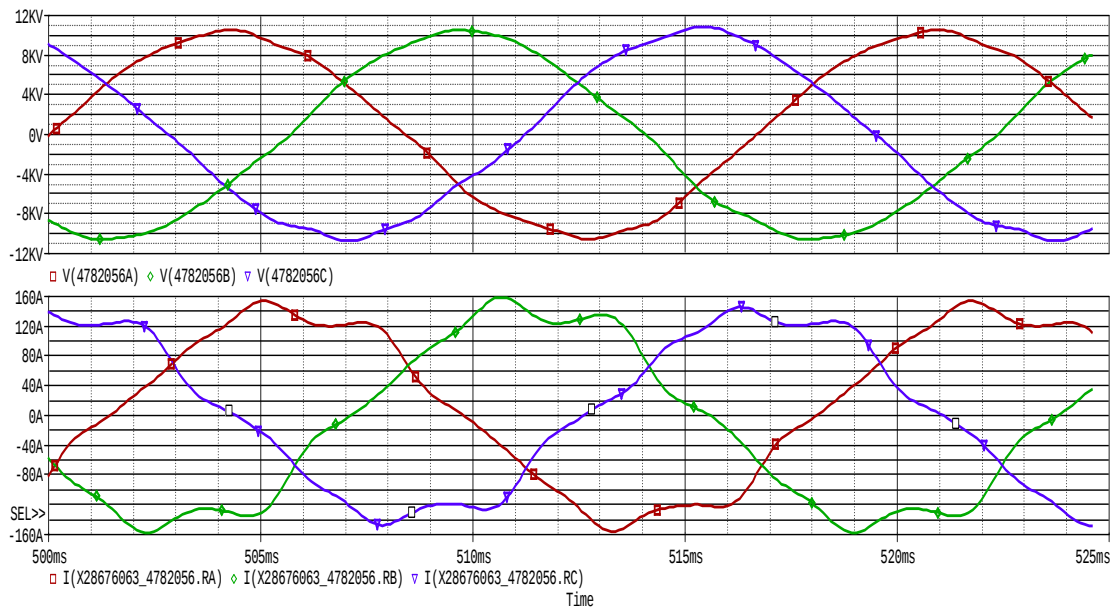
Figura 28- Simulação em ambiente VHDL-AMS da LINHA_PRI-X539727_1605338.



Fonte: Próprio autor.

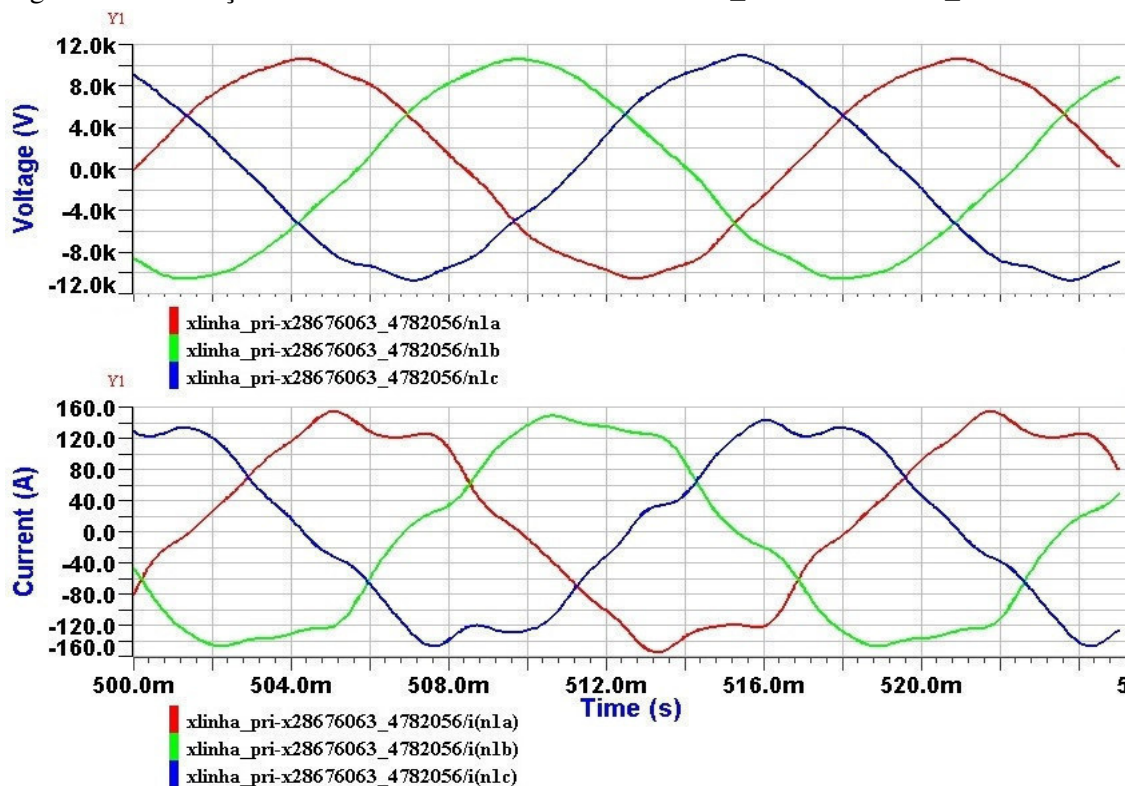
LINHA_PRI-X28676063_4782056
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 29- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X28676063_4782056.



Fonte: Próprio autor.

Figura 30- Simulação em ambiente VHDL-AMS da LINHA_PRI-X28676063_4782056.



Fonte: Próprio autor.

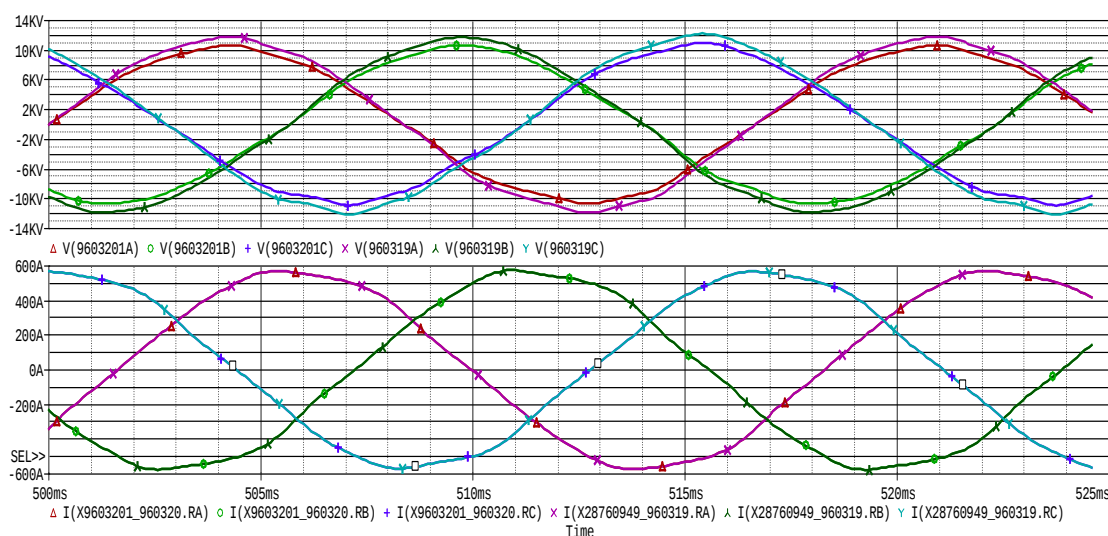
Tensões e Correntes a Entrada e Saída do Regulador de Tensão:

Entrada: Nó 'D' (960319) e Linha LINHA_PRI-X28760949_960319 respectivamente

Saída: Nó 'E' (9603201) e Linha LINHA_PRI-X9603201_960320 respectivamente

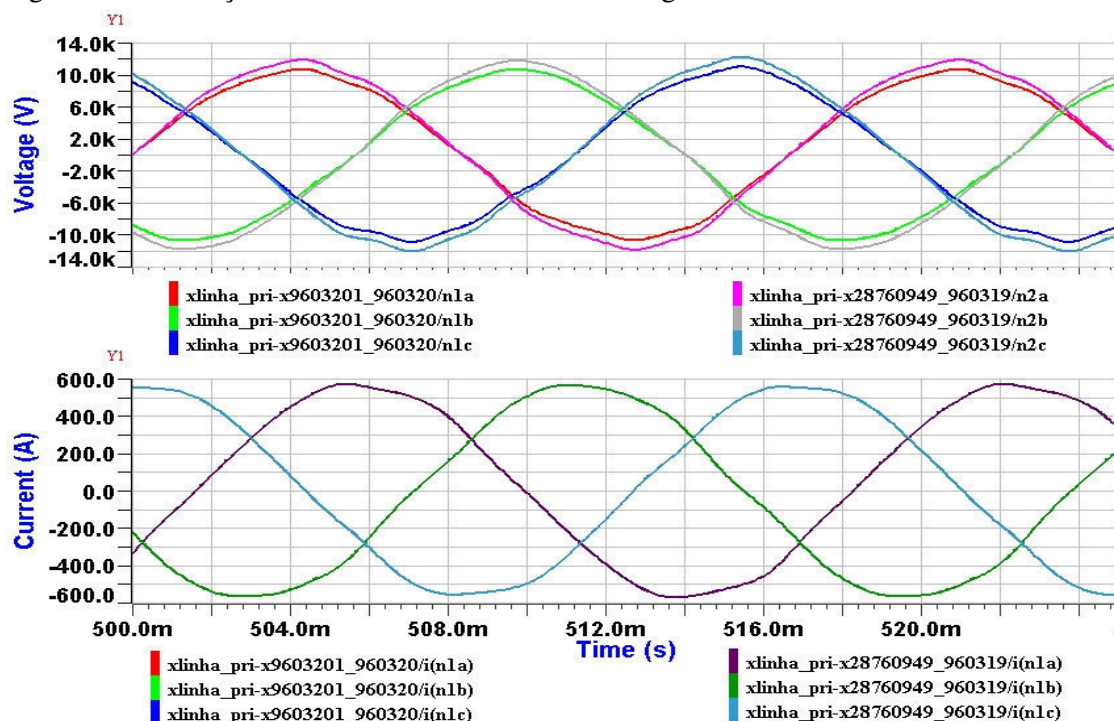
Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 31- Simulação em ambiente Orcad/Pspice do regulador de tensão.



Fonte: Próprio autor.

Figura 32- Simulação em ambiente VHDL-AMS do regulador de tensão.



Fonte: Próprio autor.

As figuras 27 até 32 apresentam os resultados das simulações mais significativas do alimentador, as quais correspondem às tensões de fase e correntes de linha para a linha que alimenta a carga com a pior regulação de tensão, a linha que alimenta a única carga não linear desequilibrada, a linha à entrada do regulador de tensão e a linha à saída do regulador de tensão, respectivamente. No apêndice D se encontram os resultados das simulações das 7 linhas restantes e das 13 cargas totais do sistema de distribuição primário a 14,4 kV.

A análise comparativa dos resultados das simulações de todas as linhas do alimentador, em Orcad/Pspice versus VHDL-AMS, do ponto de vista quantitativo, pode-se representar de acordo com a tabela 15.

TABELA 15- ANÁLISE QUANTITATIVA DOS RESULTADOS DAS SIMULAÇÕES EM ORCAD/PSPICE VERSUS VHDL-AMS.

TAP A = 8, TAP B = 8 e TAP C = 8		PSPICE				VHDL-AMS				% de Erro nos Métodos de Simulação		
Tensões das Cargas no Alimentador		VA rms	VB rms	VC rms	FD	VA rms	VB rms	VC rms	FD	Fase A	Fase B	Fase C
1	LINHA_PRI-X2919791_1329777	8279,1	8269,8	8324,0	0,002%	8279,3	8272,5	8320,3	0,001%	0,002%	0,033%	0,044%
2	LINHA_PRI-X28760949_960319	8238,6	8228,2	8283,4	0,002%	8239,9	8232,7	8280,5	0,001%	0,016%	0,055%	0,035%
3	LINHA_PRI-X9603201_960320	7414,2	7456,5	7454,6	0,001%	7414,5	7459,4	7451,0	0,001%	0,004%	0,039%	0,048%
4	LINHA_PRI-X28676063_4782056	7332,7	7374,1	7372,6	0,001%	7333,0	7377,0	7369,0	0,001%	0,004%	0,039%	0,049%
5	LINHA_PRI-X1525339_117180	7325,6	7367,4	7365,4	0,001%	7324,4	7368,6	7360,4	0,001%	0,016%	0,016%	0,068%
6	LINHA_PRI-X4781628_28676045	7326,9	7368,4	7366,8	0,001%	7326,9	7370,9	7362,9	0,001%	0,000%	0,034%	0,053%
7	LINHA_PRI-X539727_1605338	7321,8	7363,4	7361,5	0,001%	7320,2	7364,3	7356,2	0,001%	0,022%	0,012%	0,072%
8	LINHA_PRI-X2047531_28675987	7321,7	7363,2	7361,5	0,001%	7321,6	7365,7	7357,6	0,001%	0,001%	0,034%	0,053%
9	LINHA_PRI-X28675721_2019128	7397,2	7439,4	7437,3	0,001%	7397,4	7442,1	7433,7	0,001%	0,003%	0,036%	0,048%
10	LINHA_PRI-X28676197_588470	7387,3	7429,2	7427,1	0,001%	7387,4	7432,0	7423,5	0,001%	0,001%	0,038%	0,048%
11	LINHA_PRI-X1527172_1525345	7328,7	7370,4	7368,5	0,001%	7328,7	7372,9	7364,8	0,001%	0,000%	0,034%	0,050%
Correntes das Cargas no Alimentador		IA rms	IB rms	IC rms	FD	IA rms	IB rms	IC rms	FD	Fase A	Fase B	Fase C
1	LINHA_PRI-X2919791_1329777	1,075	1,065	1,072	0,003%	1,069	1,060	1,066	0,003%	0,540%	0,535%	0,597%
2	LINHA_PRI-X28760949_960319	405,29	406,98	401,49	0,006%	403,98	405,64	400,08	0,007%	0,324%	0,329%	0,351%
3	LINHA_PRI-X9603201_960320	405,29	406,98	401,49	0,006%	403,98	405,64	400,08	0,007%	0,324%	0,329%	0,351%
4	LINHA_PRI-X28676063_4782056	101,98	105,22	98,86	0,130%	101,95	105,05	98,78	0,126%	0,033%	0,163%	0,085%
5	LINHA_PRI-X1525339_117180	119,50	118,83	119,17	0,001%	118,86	118,20	118,46	0,001%	0,535%	0,526%	0,593%
6	LINHA_PRI-X4781628_28676045	5,055	5,028	5,039	0,001%	5,039	5,011	5,020	0,001%	0,324%	0,340%	0,379%
7	LINHA_PRI-X539727_1605338	73,25	72,84	73,04	0,001%	72,88	72,47	72,63	0,001%	0,508%	0,507%	0,563%
8	LINHA_PRI-X2047531_28675987	3,910	3,890	3,898	0,001%	3,899	3,878	3,884	0,001%	0,292%	0,311%	0,349%
9	LINHA_PRI-X28675721_2019128	12,86	12,79	12,82	0,001%	12,79	12,73	12,75	0,001%	0,490%	0,493%	0,554%
10	LINHA_PRI-X28676197_588470	1,378	1,371	1,375	0,001%	1,371	1,363	1,367	0,001%	0,544%	0,533%	0,596%
11	LINHA_PRI-X1527172_1525345	98,48	97,91	98,21	0,001%	97,94	97,39	97,62	0,001%	0,549%	0,535%	0,601%

Fonte: Próprio autor.

Para a tabela 15 foram utilizadas as seguintes expressões matemáticas (1), (2) e (3):

Fator de Desequilíbrio de Fases (FD):

Sendo:

$$FD\% = 100 \sqrt{\frac{1-\sqrt{3-6\beta}}{1+\sqrt{3-6\beta}}} \quad (1)$$

$$\beta = \frac{V_{ab}^4 + V_{bc}^4 + V_{ca}^4}{(V_{ab}^2 + V_{bc}^2 + V_{ca}^2)^2} \quad (2)$$

O FD foi Extraído do Módulo 8 – Qualidade da Energia Elétrica, Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional – PRODIST da Agência Nacional de Energia Elétrica – ANEEL.

% de Erro nos Métodos de Simulação:

$$\% \text{ de Erro} = \frac{|(\text{Valor em VHDL-AMS}) - (\text{Valor em PSPICE})|}{(\text{Valor em PSPICE})} * 100 \quad (3)$$

Na tabela 16 apresentam-se os resultados consolidados para as duas redes simuladas em VHDL-AMS, estudos de casos (padrão IEEE e rede real reduzida), em comparação com o Orcad/Pspice e Matlab/Simulink, considerando-se apenas os maiores valores de erros apresentados entre as simulações.

TABELA 16- COMPARAÇÃO ENTRE OS RESULTADOS DE DOIS MÉTODOS DE SIMULAÇÃO (PSPICE E SIMULINK) COM O PROPOSTO EM VHDL-AMS, PARA DUAS REDES ESTUDOS DE CASOS

Sistema Elétrico de Distribuição	Número de Nós	Simulações Confrontadas com	Maior % Erro entre os Resultados dos dois Métodos de Simulação
Alimentador de teste IEEE	13 nós	MATLAB/SIMULINK	1,929%
Alimentador Real Reduzido	959 nós	ORCAD/PSPICE	0,601%

Fonte: Próprio autor.

Com a utilização da interface para a importação de dados à plataforma VHDL-AMS (figura 15) e após da execução das consequentes simulações da rede elétrica da figura 11, podemos estabelecer as seguintes observações para os resultados em média tensão:

- Considerando o item 15 da Tabela 12, podemos concluir que o tempo total empregado pela CPU para a realização das 17 simulações automáticas do sistema elétrico de distribuição feita pela interface para a importação de dados à plataforma VHDL-AMS (figura 15), devido às mudanças do TAP no regulador de tensão e representadas na Tabela 13, é de aproximadamente 36 minutos.
- As simulações automáticas nomeadas também como testes realizados em VHDL-AMS com mudanças de TAP no regulador de tensão (para as três fases simultaneamente), foram feitas para calcular a regulação de tensão das 13 cargas pertencentes ao sistema elétrico de distribuição estudo de caso, sendo possível conhecer:
 - ✓ As características da regulação para as cinco cargas com maior regulação de tensão (Figura 19). Estas cargas de pior a melhor regulação, são: 1) CLTPEQE-XCLTPEQE539727, 2) CLTPEQE-XCLTPEQE2047531, 3) CLTPEQE-XCLTPEQE1525339, 4) CLTPEQE-XCLTPEQE1525345 e 5) CLTPEQE-XCLTPEQE4781628. A nomenclatura CLTPEQE-XCLTPEQE significa: Carga indutiva trifásica equilibrada em estrela conectada em Média Tensão (MT).

- ✓ As características da regulação de tensão para a carga com a pior regulação, segundo a variação das posições do TAP do regulador (Tabela 13),
- ✓ Os perfis de tensão e regulação do alimentador (Figura 20 e 21 respectivamente). Neste contexto, é importante mencionar que os nós destas duas figuras (eixo “X”) estão representados com letra azul no alimentador, esquematizados nas figuras 11, 12, 13 e 26.
- ✓ As variações de regulação para cada movimento da posição do TAP no regulador de tensão correspondem a 1,22% para cada TAP. Pode-se ter como exemplo, a tabela 13, que aplica para a carga com a pior regulação de tensão.
- Baseados na figura 19, podemos observar que, segundo (ANEEL, 2011), as posições dos TAP que cumprem com a faixa de variação da Tensão de Leitura (TL) em relação à Tensão de Referência (TR), que deve ser $0,93TR \leq TL \leq 1,05TR$, são: TAP -5, TAP -4, TAP -3, TAP -2, TAP -1, TAP 1, TAP 2, TAP 3 e TAP 4 para as variações dos TAP's, simultaneamente, para as três fases. Na figura 19 também se pode observar que o valor absoluto da percentagem de regulação média (entre as 5 cargas críticas) mais baixo, isto é a melhor regulação de tensão média para as 5 cargas críticas, corresponde às posições do TAP -2, igual a 3,8%. Salienta-se que o cálculo da percentagem de regulação média para cada carga foi feito com a regulação média, considerando as três fases, nessa carga.
- Com base no sistema elétrico de distribuição original proveniente de um arquivo em Orcad/Pspice (Apêndice E), as posições originais dos TAP's no regulador de tensão para as três fases, encontram-se no TAP 8. Segundo os testes realizados em VHDL-AMS, com mudanças de TAP no regulador de tensão, esta posição oferece as piores condições de regulação de tensão para todo o sistema elétrico de distribuição. De acordo com a figura 19 no TAP 8 (para as três fases), as cinco cargas com maior regulação de tensão apresentam uma regulação média de 11,82%.
- Conforme a percentagem de regulação para as cinco cargas mais críticas do alimentador (figura 19), a posição do TAP no regulador de tensão que oferece as melhores condições de regulação de tensão para todo o sistema elétrico de distribuição é o TAP -2 (para as três fases). Segundo o qual, no TAP -2, as cinco cargas com maior regulação de tensão apresentam uma regulação média de -0,38%.
- Segundo a figura 19 também pode-se inferir que a variação da regulação média total entre as cinco cargas mais críticas do alimentador corresponde a 19,51%. Isso significa que nas 5 cargas com maior regulação de tensão no sistema de distribuição a

variação da regulação média desde o TAP “-8” até o TAP “8” é quase 20%. Segundo a tabela 13, para a carga com a pior regulação de tensão, esta variação tem um valor de 19,50%.

- De acordo com a figura 20 (Perfil de Tensão do Alimentador Baseados nos Testes realizados em VHDL-AMS), pode-se observar que:
 - ✓ No sistema elétrico de distribuição, a distância desde a subestação (nó A) até a carga mais distante (nó W) é de quase 2 quilômetros (rede reduzida).
 - ✓ Para o nó W, a diferença de tensão entre o cenário de regulação mais favorável (TAP -2 para as três fases) e o mais desfavorável (TAP 8 para as três fases) é aproximadamente 1 kV.
- Segundo a figura 21 (Regulação do Alimentador Baseados nos Testes realizados em VHDL-AMS), pode-se observar que para o nó mais longe (nó W), a diferença de regulação de tensão entre o cenário de regulação mais favorável (TAP -2 para as três fases) e o mais desfavorável (TAP 8 para as três fases) é aproximadamente 13,8%.
- Considerando a análise quantitativa dos resultados das simulações em Orcad/Pspice versus VHDL-AMS (tabela 15), pode-se concluir:
 - ✓ As maiores percentagens de erro nos métodos de simulação Orcad/Pspice vs. VHDL-AMS foram: 0,072% para as tensões e 0,601% para as correntes. Estes dois resultados foram obtidos na fase C (tabela 15, células em amarelo).
 - ✓ Os maiores Fatores de Desequilíbrio de fase foram obtidos para as correntes da linha LINHA_PRI-X28676063_4782056, sendo: 0,130% (Orcad/Pspice) e 0,126% (VHDL-AMS). Este resultado era o esperado considerando que esta linha alimenta a carga CNTDQT-XCNTDQT4782056, única desequilibrada do sistema elétrico de distribuição, porém, com reduzido desbalanceamento.

2.8 CONCLUSÕES

Com o desenvolvimento do capítulo 2, podemos apresentar as seguintes conclusões:

- Em condições normais, antes de simular um circuito descrito com VHDL-AMS, este arquivo de projeto deve ser compilado previamente. Para o resultado da compilação, além de realizar-se a correspondente comprovação da sintaxe, este dado compilado é o que se relaciona com a biblioteca VHDL-AMS única do projeto (código resultante em VHDL-AMS). Esta biblioteca contém todas as descrições de todos os elementos que compõem o circuito.

- Neste capítulo, foram modelados em VHDL-AMS dois sistemas de distribuição de energia elétrica que foram comparados com outros softwares de simulação, de acordo com a tabela 16. Destes resultados, podemos afirmar que o Orcad/Pspice apresentou maiores similaridades com os resultados das simulações feitas em VHDL-MAS, em comparação com aqueles obtidos com o Matlab/Simulink.
- Neste capítulo, o enfoque de simulação em baixa tensão foi utilizado, principalmente, para avaliar e examinar as ferramentas desenvolvidas para a simulação automática em VHDL-AMS, especificamente testar a interface para a importação de dados no Sistema Elétrico de Distribuição em ambiente VHDL-AMS (item 2.6 e figura 15), considerando que o sistema de gerenciamento e controle do Sistema Elétrico de Distribuição usando um dispositivo FPGA estará baseado no sistema elétrico em média tensão.
- O enfoque de simulação em média tensão abordado neste capítulo, concernente a um alimentador de uma rede de distribuição primária de 14,4 kV (figura 11) será o caso de estudo para o qual se desenvolverá o controle digital da rede estudo de caso, conforme se apresentará nos capítulos 3 e 4.
- Baseados nas conclusões dos resultados das simulações apresentadas para o estudo de caso em média tensão, serão consideradas somente as características da regulação para as cinco cargas com maior regulação de tensão (Figura 19), ou seja, aquelas com os piores níveis de regulação de tensão, para os desenvolvimentos do controlador a ser apresentado nos capítulos 3 e 4.

SISTEMA DE GERENCIAMENTO E CONTROLE ATRAVÉS DE UM DISPOSITIVO FPGA

3.1 INTRODUÇÃO

A utilização dos dispositivos lógicos programáveis, na implementação de sistemas de controle, permite a utilização de operações que fazem uso da execução concorrente (simultânea) de várias tarefas computacionais interativas. Desta forma, em um dispositivo FPGA (***Field Programmable Gate Array***), por exemplo, podem ser implementadas tarefas simultâneas como programas separados ou como um conjunto de fios (***threads***) criados por um único programa, permitindo que todos os processos de controle possam ser executados continua e simultaneamente, atingindo, em alguns casos, o processamento de algoritmos em velocidades superiores à um processador digital de sinais - DSP (***Digital Signal Processor***). Este tipo de implementação aceita a utilização de novas metodologias de projeto digital, na qual, a funcionalidade do controlador pode ser descrita através de um modelo comportamental, o qual utiliza linguagens de descrição de hardware como VHDL (YING-YU; HAU-JEAN, 1997; CASTRO, A. de. et al., 2003).

O dispositivo FPGA é um circuito integrado projetado para ser configurado pelo cliente ou projetista após a fabricação, daí o nome "***Field Programmable***", ou seja, programável pelo usuário. Portanto, o FPGA é um hardware reconfigurável com suas funcionalidades definidas exclusivamente pelos usuários e não pelos fabricantes.

Uma grande vantagem do uso dos dispositivos FPGA é que um único FPGA poderia controlar diversos processos paralelos e diferentes funcionalidades de uma rede de distribuição, tais como o gerenciamento de uma rede ilhada, controle inteligente de rede ou controle dos parâmetros elétricos, aproveitando todas as potencialidades deste FPGA e reduzindo os custos reais de sua aplicação.

Desta forma, quanto mais complexos forem os circuitos de comando para uma rede elétrica de distribuição e se a potência processada pelo simulador for muito elevada, mais adequada será a aplicação dos sistemas de controle baseados em dispositivos FPGA.

Desta maneira, com o intuito de abranger um conceito mais restrito na padronização da distribuição elétrica nacional, neste capítulo, o termo Sistema-Elétrico-de-Distribuição faz referência a um alimentador de distribuição de energia elétrica primário em média tensão, segundo a definição padronizada pela ANEEL no Brasil.

O sistema de gerenciamento da rede de distribuição e controle (DMS&C) a ser implementado utilizando a linguagem VHDL através de um dispositivo FPGA, será desenvolvido para um alimentador de distribuição em média tensão, que para nosso caso de estudo será de 14,4 kV de tensão de linha, conforme estudo de caso já apresentado no capítulo 2.

3.2 SISTEMA DE GERENCIAMENTO E CONTROLE DA REDE DE DISTRIBUIÇÃO (DMS&C)

Lembrando o exposto no capítulo 1, no qual se informou que para a modelação e simulação em tempo real de redes e/ou microrredes inteligentes, que envolvem inclusive sistemas elétricos de distribuição com geração distribuída, são necessários dois tipos de sistemas de controle, os quais devem operar ao mesmo tempo:

- Um sistema de gerenciamento da distribuição (DMS), o qual procura as melhores condições de fornecimento e consumo da energia elétrica dentro de um sistema elétrico de distribuição (dadas determinadas regras pré-estabelecidas).
- Um controle local desenvolvido para o acionamento individual dos recursos de energia distribuída (DER), pertencente ao sistema elétrico de distribuição.

Nesse contexto, podemos afirmar dois fatos (nomeados A e B):

A) No caso de estudo da figura 11 não existem controles locais desenvolvidos para o acionamento individual dos recursos de energia distribuída (DER), considerando que foi desprezado o que acontece a montante do gerador elétrico, ou seja, o acionamento das tecnologias de geração utilizadas para a produção da energia elétrica.

B) Em condições normais, os sistemas elétricos de distribuição típicos, no contexto nacional, ainda não apresentam sistemas de gerenciamento de recursos de energia distribuída (***Distributed Energy Resources Management System – DERMS***).

Além disso, de acordo com o Instituto de Pesquisa da Energia Elétrica (EPRI, 2012), as funções típicas do sistema de gerenciamento da distribuição (***Distribution Management System – DMS***), no contexto dos sistemas elétricos de distribuição, são:

- **Gerenciamento do Perfil da Tensão:** Um dos objetivos fundamentais da gestão do sistema de distribuição é a conservação do perfil de tensão desejado ao longo do

alimentador. As concessionárias de distribuição têm usado bancos de capacitores e reguladores de tensão (incluindo comutadores de TAPs do transformador na subestação – LTCs) para manter a tensão aceitável para todos os clientes em todas as condições de carga.

- **Otimização Volt – Var (VVO):** O objetivo da Otimização Volt – Var na aplicação do sistema de gerenciamento de distribuição (DMS) é aumentar a eficiência global do sistema de distribuição, reduzir a demanda elétrica, promover a conservação da energia e melhorar a qualidade da energia elétrica, mantendo um perfil de tensão aceitável, baseado no controle da potência reativa nos DER.
- **Ilhamento para Localização de Falhas e Restauração do Serviço (FLISR):** Quando uma falta permanente ocorre no sistema de distribuição, inúmeros clientes que estão ligados às seções “saudáveis” (sem faltas) do alimentador podem experimentar uma interrupção prolongada do serviço, até que os reparos sejam feitos ou a comutação realizada. O objetivo da função do DMS para o ilhamento para localização de faltas e restauração do serviço é restaurar a energia para os clientes atendidos pelas seções saudáveis em menos de um minuto, considerando-se possíveis fontes distribuídas.
- **Balanceamento de Carga (Reconfiguração Ótima da Rede):** Normalmente, os sistemas de distribuição são projetados para serem mais eficientes na demanda da carga de pico. Muitas vezes, a rede pode ser mais eficiente e/ou mais confiável, através da reconfiguração desta de acordo com a variação na demanda da carga. A aplicação do balanceamento de carga do circuito permite à concessionária de distribuição determinar a melhor configuração dos alimentadores interconectados para alcançar uma ou mais “funções objetivas”, sem infringir qualquer das restrições operacionais estabelecidas no alimentador de distribuição.
- **Balanceamento de Fase:** As concessionárias se esforçam para manter um sistema de distribuição trifásico equilibrado. Os níveis desiguais de tensão e/ou ângulos de fase afetarão as cargas trifásicas conectadas, tais como os motores de indução, resultando em elevados níveis de corrente, os quais produzem um aquecimento anormal dentro das bobinas do motor. Enquanto as concessionárias tentam distribuir as cargas monofásicas igualmente entre as fases, como as cargas residenciais e comerciais leves, o desequilíbrio ocorre considerando que estas cargas são conectadas e desconectadas de forma aleatória. No futuro, as capacidades de balanceamento de fase podem ser melhoradas através do gerenciamento da quantidade de potência de saída do gerador

distribuído em cada fase, em uma base de tempo continua quase real, para alcançar o melhor equilíbrio das fases.

- **Análise de Contingência:** A análise de contingências avalia, fornece e prioriza os impactos dos efeitos das contingências prováveis, como sobrecargas de equipamentos ou clientes não atendidos. Isso permite aos operadores estar melhor preparados para reagir diante de faltas (interrupções), usando cenários de recuperação previamente planejados.
- **Localização Preditiva de Faltas:** A aplicação da localização preditiva de faltas em um sistema de gerenciamento de distribuição (DMS) emprega um programa de análise de curto-circuito (SCA), juntamente com a magnitude da falta, os alvos dos relés de proteção e outras informações a partir dos dispositivos eletrônicos inteligentes (IEDs) da subestação para ajudar a localizar a falta. A ferramenta do SCA identifica as possíveis variações da localização da falta, dado o modelo do sistema de distribuição, as entradas dos IEDs e as premissas especificadas da concessionária sobre a impedância de falta.
- **Rejeição de Carga de Emergência (ELS):** A rejeição de carga de emergência reduzirá rapidamente a quantidade de carga no sistema elétrico de distribuição, em resposta a uma emergência no nível do sistema de potência, tais como a perda de uma instalação chave de geração ou de uma linha de transmissão. A ELS identifica os grandes clientes específicos com cargas susceptíveis de interrupção, os circuitos inteiros de distribuição e os circuitos inteiros para ser acionados, procurando cumprir com a exigência de limitação da carga, considerando o nível de carregamento atual e a quantidade desejada de disparos.
- **Gerenciamento de Ordens de Comutação:** A função de gerenciamento de ordens de comutação auxilia o despachante na preparação e execução de procedimentos de comutação para vários elementos do sistema de potência, incluindo os dispositivos da subestação e os dispositivos de campo (fora do muro da subestação).
- **Previsão da Carga a Curto Prazo:** O sistema de gerenciamento de distribuição (DMS) inclui uma função de previsão da carga a curto prazo que é capaz de prever a carga sobre uma porção específica do sistema de distribuição em um determinado momento (7 a 10 dias no futuro). Esta informação pode ser usada para planejar um trabalho sobre o sistema de distribuição, procurando determinar se essas tarefas podem ser realizadas durante o horário normal de trabalho, ou durante o horário de pico.
- **Gerenciamento das Oscilações de Tensão:** Uma das principais responsabilidades do operador do sistema de distribuição (assistida pelo DMS), é manter a tensão aceitável

para todos os clientes em todas as condições de carga. As funções convencionais de gerenciamento da tensão não são adequadas para o gerenciamento de flutuações de tensão mais rápidas, associadas com uma saída variável de unidades geradoras de energia solar fotovoltaica e unidades de energia eólica, por exemplo. Os inversores inteligentes relacionados com esses recursos de energia distribuída (DER) podem fornecer mecanismos para lidar com essas flutuações (por exemplo o apoio ao rápido crescimento da potência reativa), sejam de tensões ou frequências.

- **Gerenciamento do Carregamento de Veículos Elétricos:** Espera-se que o DMS desempenhe um papel no gerenciamento do carregamento de veículos elétricos com o foco no gerenciamento da demanda da energia elétrica. Ou seja, quando o DMS precisa reduzir a demanda em uma determinada subestação, alimentador, ou seção do alimentador, o DMS pode limitar a quantidade de carregamento rápido que é permitido nessa subestação, alimentador, ou seção do alimentador.
- **Gerenciamento de microrredes:** As microrredes utilizam recursos de energia distribuída (geração distribuída e armazenamento de energia) para alimentar uma parte do sistema de distribuição que se desconectou da rede elétrica principal (operação ilhada, constituindo uma microrrede). A função do DMS é monitorar a carga, tensão e frequência na microrrede para assegurar que essas magnitudes estejam dentro dos limites aceitáveis. O DMS também funcionará com os DER para assegurar que existe um equilíbrio entre a carga e a capacidade de geração na microrrede.
- **Monitoramento da Qualidade da Energia:** O DMS monitora a distorção harmônica total e outras medidas do conteúdo harmônico no alimentador elétrico e determina que quantidade de potência do transformador está baseada no conteúdo harmônico. Inversores inteligentes podem fornecer um mecanismo para mitigação de harmônicas, operando como filtros ativos localizados, permitindo a redução dos valores eficazes de correntes pelas redes e equipamentos, redução das perdas e ampliando, conseqüentemente, a vida útil da instalação.
- **Gerenciamento do Fluxo de Potência no Alimentador:** Com a geração centralizada nos alimentadores de distribuição radiais, há apenas um caminho possível em que a potência pode fluir através da rede de distribuição, ou seja, da subestação para as cargas dos clientes. Desta forma, o único mecanismo disponível para controlar o fluxo de potência está associado ao controle da demanda. Com os DER presentes nos alimentadores de distribuição, o operador de distribuição é capaz de controlar o fluxo de potências de uma forma plena para as redes.

- **Balanceamento de Potência e Despacho de Geração:** Como os recursos de potência tornam-se cada vez mais distribuídos, caracterizados como os sistemas mais favoráveis para responder à demanda e à geração mais intermitente, a operação de um sistema robusto e eficiente dependerá criticamente da capacidade dos novos métodos de despacho, procurando fornecer uma previsão melhor e visão holística das condições do sistema e padrões de geração. Um desafio particular relacionado com o monitoramento e controle do equilíbrio entre a oferta e a demanda, quando se trata de recursos renováveis, é que as suas respectivas produções muitas vezes não seguem a correlação tradicional entre geração e carga, mas estes recursos têm fortes dependências das condições meteorológicas.
- **Resposta à Demanda / Redução da Demanda de Pico:** A resposta à demanda ou a redução da demanda de pico refere-se a mecanismos utilizados para incentivar os consumidores a contê-la, o que reduziria o pico no uso da eletricidade. Os programas de resposta à demanda e controle direto da carga têm sido utilizados para essa redução e busca evitar a construção de novos recursos de geração, com o intuito de acomodar as cargas de pico que existem, apenas por um curto período de tempo. O DMS pode usar a energia armazenada e outros DER para permitir atividades adicionais de resposta à demanda ou redução da demanda de pico.

Considerando os dois fatores anteriores (**A** e **B**) e as dezessete funções padronizadas pelo **EPRI** do sistema de gerenciamento da distribuição (DMS), o DMS considerado neste trabalho de doutorado terá o seguinte objetivo: Fazer um gerenciamento do perfil de tensão de um alimentador estudo de caso, ou seja, buscar as condições ótimas de tensão nas cargas elétricas, cumprindo com os critérios de regulação de tensão exigidos pela ANEEL no Brasil. Isto, com o intuito de procurar as melhores condições de fornecimento e consumo da energia elétrica, para este estudo de caso simplificado com apenas um regulador de tensão e um banco de capacitores.

Observando que a linguagem de descrição de hardware mais comumente utilizada para programar um FPGA é VHDL (DE LOS ANGELES GOMEZ LOPEZ et al., 2013), com base nos requisitos e restrições do sistema elétrico (definido pelo cumprimento das normas brasileiras), o sistema de gerenciamento da distribuição e controle (DMS&C), mostrado na figura 33, utiliza um algoritmo desenvolvido em linguagem VHDL, o qual é implementado em uma placa de programação FPGA.

De acordo com a figura 33, o DMS&C a ser implementado em um dispositivo FPGA dará prioridade ao controle do regulador de tensão do alimentador, com base na mudança de comutação do TAP no regulador. A nova arquitetura do DSMS&C é desenvolvida para regular o nível de tensão de cada carga (para cada fase). Neste contexto, com base nos resultados de simulação em plataforma VHDL-AMS, o algoritmo digital VHDL (implementado em uma placa de desenvolvimento ZedBoard Zynq™-7000), comparará os valores de regulação de tensão das cinco cargas com a pior regulação de tensão (para cada fase), com as normas nacionais, de acordo com a Agência Nacional de Energia Elétrica –ANEEL, gerando mensagens de comando para o DSMS&C tomar as ações de controle. Para nosso caso de estudo da figura 11, as correções da regulação da tensão serão feitas somente com a movimentação do comutador de TAPs no regulador de tensão, segundo o enfoque de simulação em média tensão. Isto quer dizer que o DMS&C de nosso alimentador gerenciará o controle da regulação de tensão unicamente através das mudanças do TAP no regulador de tensão (para cada fase).

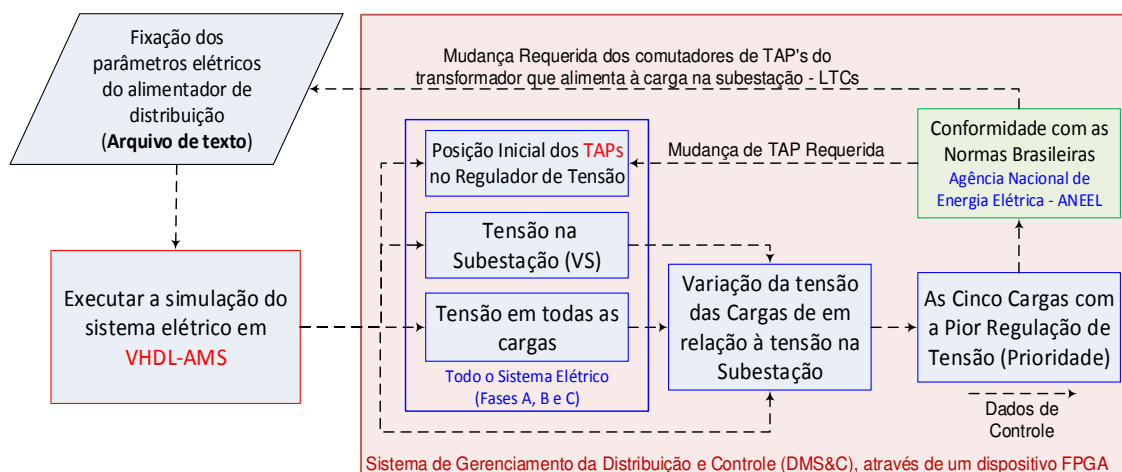
É importante destacar que, apesar das elevadas dinâmicas de atuação esperadas para o DSMS&C, ou seja, para suas ordens de regulação, as mudanças reais que ocorreriam no alimentador real estariam associadas às dinâmicas do mesmo, aos seus mecanismos de controle e proteção, devendo, portanto, existirem outras ações hierárquicas de controle para as mudanças reais nas comutações de TAPs do regulador de tensão.

Portanto, esta tese trata apenas, neste momento, das ordens de comando do controlador digital inteligente associado às ferramentas de simulação em tempo real da rede.

Os resultados da execução do algoritmo proposto e esquematizado na figura 34, dentro do dispositivo FPGA, mostrarão todas as ações de controle (através de mensagens de aviso) em relação à necessidade de mudança nos TAPs do regulador de tensão (que posição do TAP por fase), se é necessária uma mudança nos comutadores de TAP's do transformador que alimenta a carga na subestação - LTCs, a montante do regulador de tensão, e se o alimentador atenderá ou não às normas brasileiras na condição de controle proposta.

Este sistema de gerenciamento da distribuição e controle (DMS&C) foi implementado em um dispositivo FPGA Artix-7 (inserido em uma placa de desenvolvimento ZedBoard Zynq™-7000), utilizando um controle digital, empregando o algoritmo para o controle da regulação de tensão indicado na figura 34.

Figura 33- Estrutura desenvolvida para o sistema de gerenciamento da distribuição e controle (DMS&C), implementado através de um dispositivo FPGA.



Fonte: Próprio autor.

3.3 ALGORITMO PARA O CONTROLE DA REGULAÇÃO DE TENSÃO

Do ponto de vista do funcionamento interno de um regulador de tensão para sistemas de distribuição de energia elétrica, pode-se observar que quando as posições do TAP no regulador diminuem, a tensão a jusante do regulador de tensão aumenta; ou também, quando as posições do TAP no regulador aumentam, a tensão a jusante do regulador de tensão diminui (ARAMIZU; VIEIRA, 2013). Então, quando, utilizando um regulador de tensão, requer-se diminuir a tensão na carga, devemos aumentar a posição do TAP no regulador. Ou, quando requer-se aumentar a tensão na carga a jusante do regulador de tensão, devemos diminuir a posição do TAP no regulador.

Para o projeto do algoritmo do sistema de gerenciamento da distribuição e controle do alimentador da figura 11, proposto na figura 34, foram considerados os seguintes aspectos:

- O sistema de gerenciamento da distribuição e controle (DMS&C) do alimentador somente controlará a regulação de tensão através das mudanças do TAP no regulador de tensão (segundo a figura 33). Isto porque o alimentador de distribuição estudo de caso possui apenas um regulador de tensão.
- Para o controle da regulação de tensão no alimentador da figura 11, será utilizado o critério de controle para as cinco (5) cargas mais críticas, ou seja, aquelas com os maiores valores de regulação. Isso, porque se estas cargas cumprirem com os padrões nacionais de regulação, então as demais também cumprirão.

- Segundo as simulações automáticas apresentadas no capítulo 2, em VHDL-AMS, das treze cargas totais da rede de média tensão proposta como estudo de caso, as cinco cargas que têm a pior regulação de tensão (mais elevada), para todas as posições dos TAP no regulador de tensão, de pior a melhor são: **CLTPEQE-XCLTPEQE539727**, **CLTPEQE-XCLTPEQE2047531**, **CLTPEQE-XCLTPEQE1525339**, **CLTPEQE-XCLTPEQE1525345** e **CLTPEQE-XCLTPEQE4781628**.
- O regulador de tensão tem um comutador de TAP's com 8 posições aditivas e 8 subtrativas, havendo uma variação total de tensão de 19,5 % em 16 posições.
- O algoritmo para o controle digital da regulação de tensão leva em consideração a não simultaneidade das posições dos TAP's do regulador de tensão, ou seja, as posições iniciais e finais dos TAP's do regulador não necessariamente devem ser idênticas para as três fases A, B e C. Apesar deste fato, deve-se considerar as condições de melhor equilíbrio para as tensões, nas operações de comutação de TAPs.
- Segundo o algoritmo, o processo de controle digital finaliza quando acontece alguma das seguintes condições:
 - ✓ O alimentador atende as normas Brasileiras. Neste caso o controle (através do FPGA) deve fornecer a regulação das cinco cargas mais críticas para as três fases e as posições finais do TAP A, TAP B e TAP C no regulador de tensão.
 - ✓ O alimentador NÃO atende as normas Brasileiras. Neste caso, além de fornecer a regulação das cinco cargas mais críticas para as três fases e as posições finais do TAP A, TAP B e TAP C no regulador de tensão, o controle recomenda manobrar o transformador a montante para melhorar a regulação no alimentador.

Segundo o Módulo 8 do Procedimento de Distribuição de Energia Elétrica no Sistema Elétrico Nacional – PRODIST da Agência Nacional de Energia Elétrica – ANEEL vigente no Brasil, a faixa adequada de variação da Tensão de Leitura (TL) em relação à Tensão de Referência (TR) é: $0,93TR \leq TL \leq 1,05TR$. A TL é estimada como a tensão na carga e a TR é considerada como a tensão na subestação. Assim, a faixa adequada da variação da tensão na carga (TL) em relação à tensão na subestação (TR) pode ser expressada, em função da magnitude da regulação de tensão, segundo as duas equações (5) e (6), conforme tabela 17.

Por outro lado, pode-se expressar a magnitude da regulação de tensão na carga, em função das tensões na subestação e na carga, para um alimentador de distribuição, segundo a expressão (4).

$$MR\% = \frac{TR-TL}{TR} * 100\% \quad (4)$$

TABELA 17- FAIXA ADEQUADA DE VARIAÇÃO DA TENSÃO NA CARGA (TL) EM RELAÇÃO À TENSÃO NA SUBESTAÇÃO (TR).

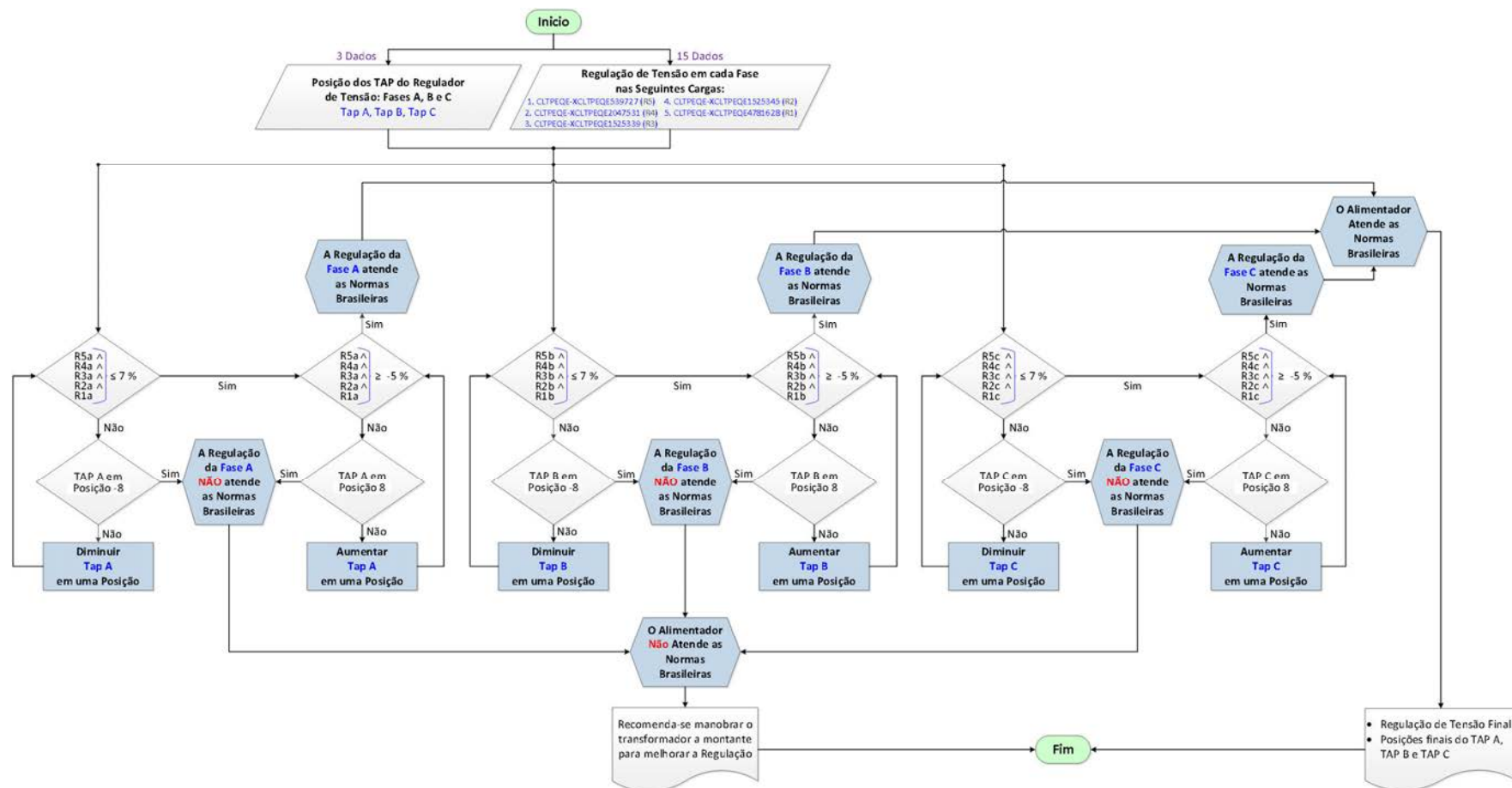
1) $TL \geq 0,93TR$	2) $TL \leq 1,05TR$
$TL \geq 0,93TR * (-1) \rightarrow - TL \leq - 0,93TR$	$TL \leq 1,05TR * (-1) \rightarrow - TL \geq - 1,05TR$
$\frac{TR-TL}{TR} * 100\% \leq \frac{TR-0,93TR}{TR} * 100\%$	$\frac{TR-TL}{TR} * 100\% \geq \frac{TR-1,05TR}{TR} * 100\%$
$MR\% \leq \frac{TR-0,93TR}{TR} * 100\%$	$MR\% \geq \frac{TR - 1,05TR}{TR} * 100\%$
MR % ≤ 7 % (5)	MR % $\geq - 5$ % (6)

Fonte: Próprio autor.

Segundo a tabela 17, especificamente as equações (5) e (6), pode-se deduzir que para cumprir com a faixa adequada de variação da Tensão de Leitura (tensão na carga - TL) em relação à Tensão de Referência (tensão na subestação - TR), deve cumprir-se que os valores numéricos das magnitudes da regulação de tensão (as quais podem ser valores positivos ou negativos), para as cargas, sejam assim: $- 5 \% \leq MR\% \leq 7 \%$. Na qual, MR significa Magnitude da Regulação de tensão na carga.

Baseado no exposto, foi esquematizado o algoritmo para o sistema de gerenciamento da distribuição e controle (DMS&C) implementado no FPGA, conforme figura 34.

Figura 34- Algoritmo para o sistema de gerenciamento da distribuição e controle (DMS&C) feito pelo FPGA.



Fonte: Próprio autor.

3.4 MATRIZ DE PORTAS PROGRAMÁVEIS EM CAMPO – FPGA

Os dispositivos lógicos programáveis FPGA, aceleram o trabalho de concepção e facilitam a atualização rápida de um projeto digital. Neste contexto, estes dispositivos são componentes vitais, principalmente, na implementação e testes de sistemas digitais (FREEMAN, 1988). Os dispositivos FPGA tornaram-se elementos eficazes na implementação de sistemas digitais, de maneira rápida, com elevada flexibilidade, elevado desempenho e apresentando um custo reduzido (HUANG et al., 2012).

Atualmente, o aumento nas capacidades dos dispositivos FPGA tem resultado no nascimento de soluções avançadas de sistema-em-um-chip (**SoC**) (TRIMBERGER; MOORE, 2014). Neste contexto, a montagem do sistema de gerenciamento da distribuição e controle (DMS&C) do alimentador da figura 11, baseada na figura 33 e segundo o algoritmo da figura 34, será realizada utilizando uma arquitetura SoC de primeira geração, pertence à família Xilinx Zynq-7000, a qual encapsula um dispositivo FPGA Artix-7 e um processador ARM Cortex-A9 de núcleo duplo em um único chip. Este chip está inserido em uma placa de desenvolvimento ZedBoard Zynq™-7000.

3.4.1 Dispositivos da família Xilinx Zynq®-7000

A família Xilinx Zynq®-7000 está baseada na arquitetura SoC programável. Segundo (YOUNG-SU; CHONG-MIN, 2005) os SOC projetados, geralmente, consomem menos energia e têm um custo mais baixo e maior confiabilidade do que os sistemas multichips que eles substituem. Além disso, os SOC apresentam menos pacotes no sistema e os custos de montagem são bem reduzidos.

A família de dispositivos Xilinx Zynq®-7000 possui um dos mais avançados sistemas embarcados desenvolvidos para FPGAs disponíveis atualmente, além de algumas das mais avançadas tecnologias de arquitetura SoC programável. Dentre elas, podemos destacar a integração de um sistema de processamento (PS), com um dispositivo lógico programável (PL), em um único circuito integrado (chip).

O sistema de processamento (PS) inclui um processador de um núcleo duplo ARM® Cortex™-A9, um controlador de memória DDR dedicada, e uma grande variedade de periféricos de entrada/saída, permitindo que dois dados sejam transferidos ao mesmo tempo.

O dispositivo lógico programável (PL) envolve um dispositivo de matriz de portas programáveis pelo usuário, FPGA Artix-7, de 28 nanômetros.

Os dois sistemas do dispositivo Xilinx Zynq®-7000, o PS e o PL, são ligados por uma série de interfaces que aderem ao padrão de interconexão AXI4 (***Advanced eXtensible Interface***). As interfaces AXI estão em conformidade com as especificações da versão 4 da AMBA® AXI, da ARM®, incluindo o subconjunto da interface de registro de controle AXI4-Lite (uma variante de funcionalidade reduzida da especificação AXI4).

Neste trabalho de tese será utilizado o dispositivo ZYNQ SOC XC7Z020-CLG484, inserido na placa ZedBoard. As características mais relevantes deste dispositivo podem-se encontrar na seguinte tabela 18 (XILINX, 2014).

TABELA 18- CARACTERÍSTICAS MAIS RELEVANTES DO ZYNQ SoC XC7Z020-CLG484.

Dispositivo Lógico Programável Xilinx:	Artix-7 FPGA
Núcleo do Processador:	Dual ARM® Cortex™-A9 MPCore™ com CoreSight™
Extensões do Processador:	NEON™ & único/duplo a precisão de ponto flutuante para cada processador
Frequência Máxima de Operação:	866 MHz
Sistema de Processamento para as Portas de Interface de Lógica Programável:	2x AXI 32b Mestre 2x AXI 32-bit Escravo; 4x AXI 64-bit/32-bit de Memória e AXI 64-bit ACP
Células Lógicas Programáveis:	Células lógicas 85K
Look-Up Tables (LUTs):	53.200
Flip-Flops:	106.400
Bloco RAM Extensible:	560 KB
DSP Slices Programáveis (Processamento Digital de Sinal):	220

Fonte: (XILINX, 2014).

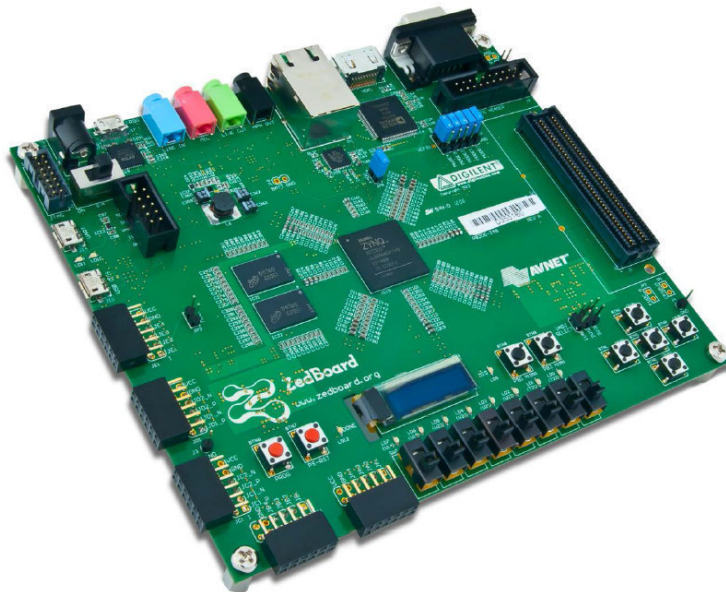
3.4.2 Placa de desenvolvimento ZedBoard Zynq™-7000

Revisando os dispositivos FPGAs de última geração, os quais oferecem todas as vantagens do suporte tecnológico atualizado do fabricante, no trabalho desenvolvido por (SCHWIEGELSHOHN; HUBNER, 2014) foi utilizada uma placa Xilinx ZedBoard Zynq™-7000 para o desenvolvimento de um sistema de detecção de atenção para os motoristas de automóveis. Neste trabalho, a placa ZedBoard tinha uma conexão com um simulador e foi utilizada para gerar mensagens de aviso, controlando assim as falhas de atenção dos motoristas. Nesse contexto, foram

investigadas as características técnico-econômicas desta placa de desenvolvimento no sentido da adequação à montagem do sistema de gerenciamento da distribuição e controle (DMS&C) do alimentador da figura 11, segundo as figuras 33 e 34 (Algoritmo para o DMS&C).

Como a família de dispositivos FPGA Artix é de baixo custo, a placa ZedBoard Zynq™ 7000, conforme figura 35, é uma placa de desenvolvimento também de baixo custo, para todos os SoC programáveis da série Xilinx Zynq™-7000 (em torno de 500 dólares americanos). Esta placa contém os recursos necessários para se criar um projeto baseado em Windows, Linux, Android, ou outro sistema operacional de tempo real. Além disso, vários conectores de expansão expõem o sistema de processamento e as portas lógicas programáveis de Entrada/Saída (E/S), para um fácil acesso do usuário. A placa ZedBoard é a primeira placa de seu tipo no mercado e tem um enorme potencial como um sistema totalmente integrado (DIGILENT, 2015).

Figura 35- Placa de desenvolvimento ZedBoard Zynq™-7000.



Fonte: (DIGILENT, 2015).

O desenvolvimento do controle digital para o sistema de gerenciamento do alimentador de distribuição será implementado nesta placa de desenvolvimento ZedBoard de última geração, com ótima relação custo-benefício. A placa apresenta uma magnífica flexibilidade para implementações e modificações futuras (pela presença do processador). Por outro lado, ela oferece a capacidade de interfaceamento entre o PC ou FPGA e o mundo real através de conversores analógicos – digitais (A/D) e entradas – saídas (I/Os) digitais, importantes para a

ampliação deste trabalho de pesquisa com a necessária expansão adequada do hardware, por exemplo, para inserir medições de grandezas de um sistema elétrico real no dispositivo FPGA da placa. O dispositivo FPGA Artix-7 apresenta uma grande capacidade de processamento, própria desta família.

3.5 VISÃO GERAL DO AMBIENTE DE PROJETO EM VHDL

Segundo (NAGASAMY et al., 1992) os principais componentes do ambiente de projeto em VHDL são: o simulador de nível misto VHDL, o compilador, a ferramenta de exploração de arquitetura, o motor de síntese arquitetônica, a ferramenta de síntese do nível de transferência do registrador - RTL¹, os geradores de módulos e o otimizador da lógica. Segundo o ambiente de projeto em VHDL (figura 36), o compilador, ao analisar o arquivo VHDL de entrada, determina quais técnicas de síntese são necessárias aplicar a cada parte do projeto. Em seguida, ele invoca ferramentas de síntese apropriadas para cada parte do projeto.

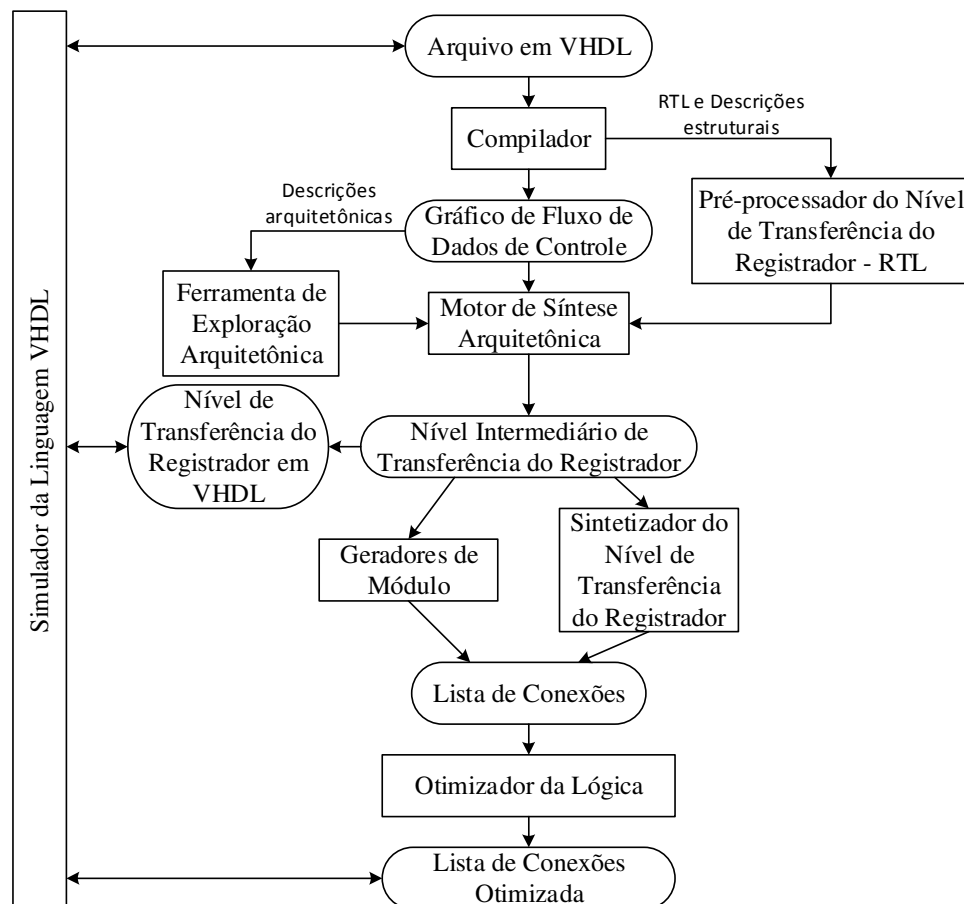
O compilador passa todas as partes do projeto descritos no estilo arquitetônico para a ferramenta de exploração arquitetônica e para o motor de síntese arquitetônica. As partes descritas no estilo RTL ou no estilo estrutural passarão pelo pré-processador do Nível de Transferência do Registrador – RTL. O usuário deve indicar para o compilador se um processo de projeto é uma descrição arquitetônica ou uma descrição RTL por meio de um atributo em VHDL, associado com a etiqueta do processo, porque, em alguns casos, os estilos de descrição são semelhantes.

O pré-processador do RTL reconhece qualquer descrição de máquina de estado e cria elementos de armazenamento (chaves e registros) para variáveis condicionalmente atribuídas. Os fluxos de dados que saem do motor de síntese arquitetônica e do pré-processador do RTL, recombina-se nesta fase. Em seguida, o usuário pode gerar um arquivo VHDL que descreve o projeto inteiro predominantemente no RTL (feito pela ferramenta de síntese). Somente as partes do projeto que estavam originalmente no nível estrutural estarão nesse nível, neste arquivo VHDL, que por sua vez, tem dois usos principais. Primeiro, o usuário pode simular o design no RTL para verificar a exatidão da ferramenta de síntese arquitetônica. Em segundo lugar, o usuário pode escolher uma ferramenta de síntese RTL para obter a lista de conexões (*netlist*).

¹ Quando os circuitos digitais são descritos no nível de transferência do registrador (RTL- *Register Transfer Level*), os detalhes das portas lógicas e suas interconexões para implementar o circuito, são gerados automaticamente pela ferramenta de síntese lógica da descrição no Nível de Transferência do Registrador (RTL).

Os geradores de módulos, que compõem um conjunto de ferramentas especializadas em sintetizar estruturas regulares, tais como somadores, multiplicadores, e decodificadores, também sintetizam blocos de memória utilizados no projeto, tais como as portas individuais ou RAMs multiportas. O compilador reconhece as partes da descrição em VHDL direcionadas para essas ferramentas e as passa para o gerador de módulo apropriado. O simulador VHDL ajuda o usuário a verificar o projeto em diferentes fases. Durante a especificação inicial, ele verifica se a descrição em VHDL de entrada satisfaz a funcionalidade estabelecida nos requisitos. Em seguida, ele simula o projeto RTL para verificar a exatidão das ferramentas de síntese arquitetônica. Finalmente, o simulador verifica a lista de conexões (*netlist*) após a otimização da lógica.

Figura 36- Ambiente de projeto em VHDL.



Fonte: (NAGASAMY et al., 1992).

3.6 CONTROLADOR DIGITAL PROGRAMÁVEL PARA O DMS&C

Baseados no ambiente de Projeto em VHDL, conforme figura 36, o desenvolvimento do projeto do sistema de gerenciamento da distribuição e controle (DMS&C) empregou a estrutura apresentada na figura 33, consistindo no uso de linguagem de descrição de hardware VHDL, em conjunto com as ferramentas atuais do ambiente de projeto integrado – IDE (***Integrated Design Environment***) do fabricante do dispositivo lógico programável – FPGA (Xilinx), especificadas para a implementação do projeto, em um estilo “***top-down***”. Estas ferramentas atuais do ambiente de projeto integrado – IDE estão representadas pelo software de projeto ***Vivado Design Suite***, o qual será abordado nos próximos parágrafos.

Obviamente, este trabalho de tese emprega o dispositivo FPGA Xilinx Artix-7 para, de maneira experimental, fazer o controle digital da regulação de tensão do alimentador de distribuição de energia elétrica com as cinco (5) cargas com a pior regulação, considerando-se o estudo de caso aqui admitido.

Segundo as especificações do fabricante, o dispositivo ZYNQ SoC XC7Z020-CLG484 pode ser usado de três maneiras:

- O processador ARM pode ser utilizado, juntamente, com o dispositivo lógico programável (FPGA Artix-7). O fabricante recomenda usar esta opção para conseguir projetos mais eficientes no sentido de melhorar a comunicação entre o FPGA e o PC. Isso porque o processador ARM traz instrumentos para fazer possível esta comunicação entre o PC e o FPGA.
- O processador pode ser utilizado independentemente do dispositivo lógico programável (FPGA Artix-7).
- O dispositivo lógico programável (FPGA Artix-7) pode ser utilizado independentemente do processador ARM Cortex-A9. No entanto, o processador ou a interface JTAG devem ser utilizadas para programar o FPGA.

No projeto do DMS&C o dispositivo ZYNQ SoC XC7Z020-CLG484 foi utilizado o primeiro modo de operação.

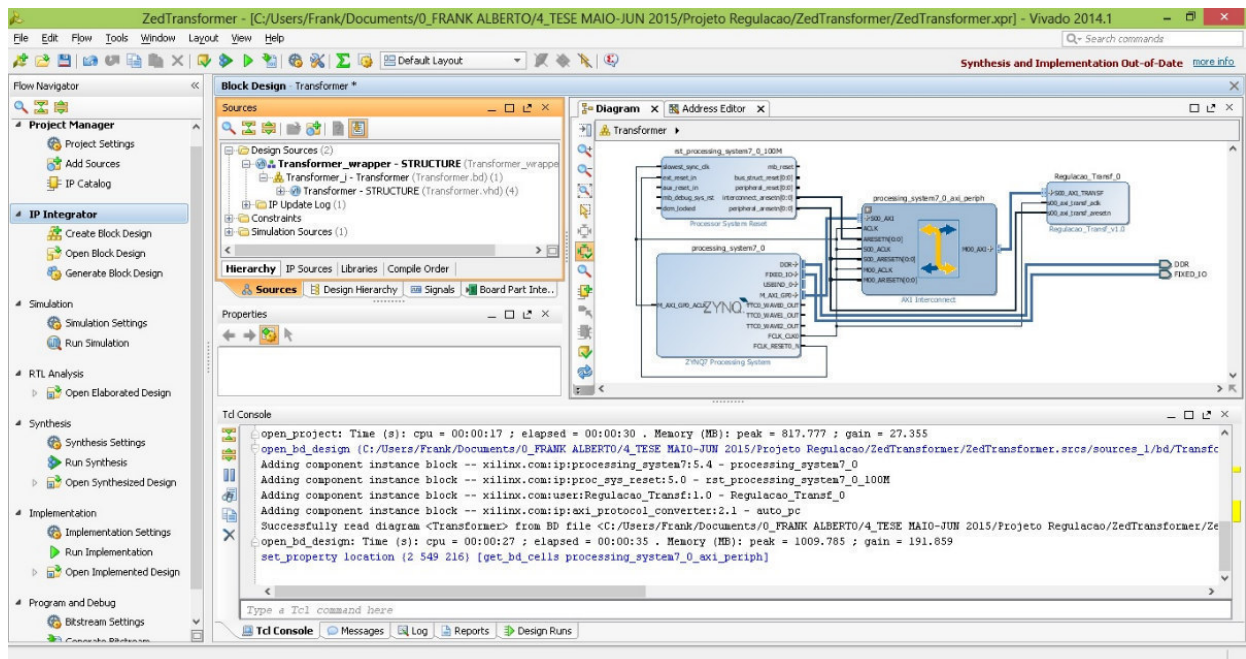
A ferramenta IDE fornecida pelo fabricante para a concepção do DMS&C, é o software de projeto ***Vivado Design Suite***, que substituiu ao antigo Xilinx ISE ® Design Suite, o qual era o ambiente de projeto padrão para os dispositivos FPGA da Xilinx (ferramentas de projeto). Embora o Xilinx ISE ® Design Suite ainda exista, este é utilizado pelas tecnologias antigas (Spartan-6,

Virtex-6 e Coolrunner). Desde outubro de 2013, o Xilinx ISE ® Design Suite mudou-se para a fase de sustentação de seu ciclo de vida do produto, e não há planejadas novas versões.

A empresa Xilinx recomenda utilizar o **Vivado Design Suite** para os novos projetos, as quais utilizam as seguintes tecnologias disponíveis no mercado, para dispositivos FPGA: Ultrascale, Virtex-7, Kintex-7, Artix-7 e Zynq-7000. Em termos gerais, os pilares de produtividade do **Vivado Design Suite** são: Integração da propriedade intelectual (IP) e implementação, Verificação e depuração e Exploração do projeto e geração da IP.

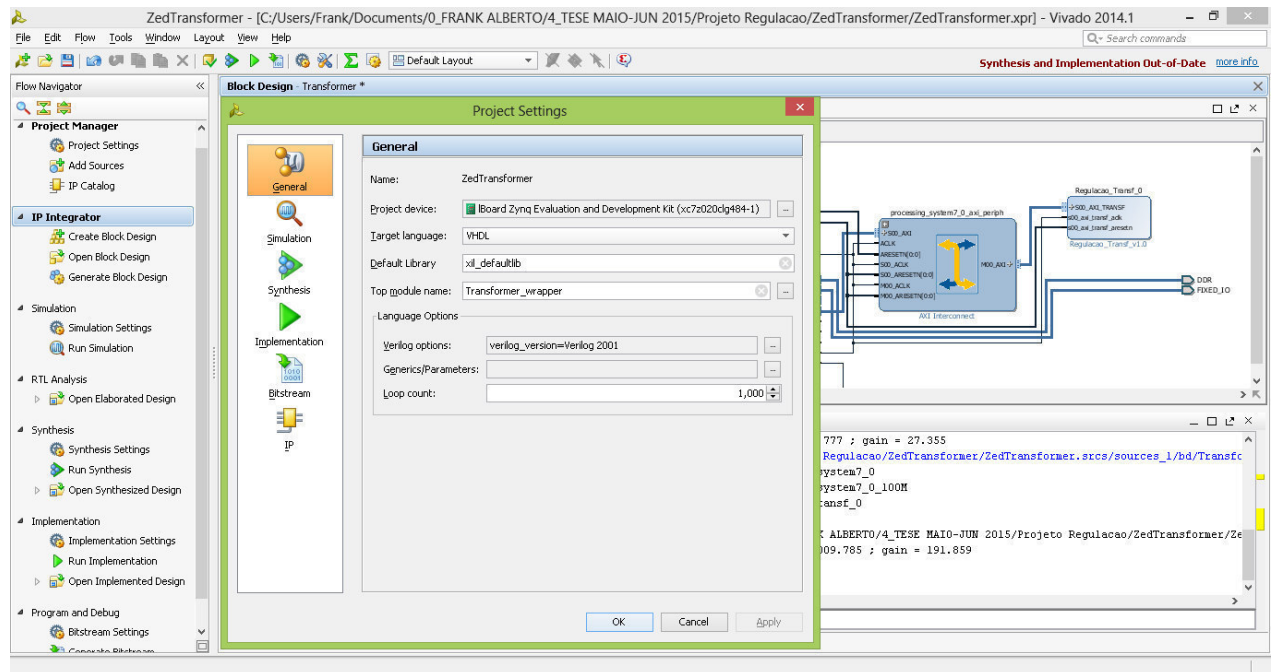
Nas figuras 37 e 38 se pode observar o ambiente de projeto do **Vivado Design Suite**, utilizado no projeto do DMS&C, e as configurações iniciais deste projeto neste ambiente, respectivamente.

Figura 37- Ambiente de projeto do **Vivado Design Suite**.



Fonte: Próprio autor utilizando o **Vivado Design Suite**.

Figura 38- Configurações iniciais do projeto no **Vivado Design Suite**.



Fonte: Próprio autor utilizando o **Vivado Design Suite**.

O **Vivado Design Suite** foi introduzido em abril de 2012, e é um ambiente de projeto integrado (IDE), com um sistema de ferramentas de nível de circuito integrado, construído em um modelo escalável compartilhado de dados e que possui um ambiente comum de depuração (em inglês: **debugging**), ou seja, em um mesmo ambiente de software é possível encontrar e reduzir os erros de programação (aqueles que previnem o programa de ser executado, ou aqueles que produzem um resultado inesperado).

Neste projeto do DMS&C foi utilizada a avaliação gratuita de 30 dias do Vivado, a qual apresenta as seguintes características: a) Integração da propriedade intelectual (IP) e implementação: Ambiente de projeto Integrado e kit de desenvolvimento de software - SDK; b) Verificação e depuração: Simulador Vivado, Analisador Lógico Vivado e Analisador serial de entrada/saída Vivado e; c) Exploração do projeto e geração da IP: Síntese de alto nível Vivado e Sistema Gerador para processamento digital de sinais. Atualmente, a licença do **Vivado Design Suite** tem um custo de 5.800 dólares americanos para ser instalada em um servidor de rede com vários usuários, ou 4.800 dólares americanos para um único computador.

O **Vivado Design Suite** inclui a interface gráfica do usuário Vivado (GUI Vivado), um integrador de propriedade intelectual da Xilinx (Integrador IP) e um kit de desenvolvimento de software - SDK (**Software Development Kit**), dentre outros.

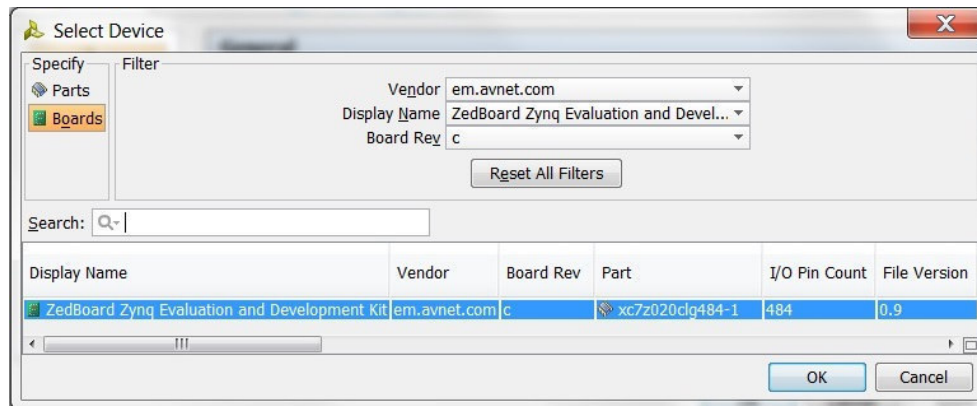
GUI Vivado inclui ferramentas de projeto ao nível do sistema eletrônico para analisar, sintetizar, implementar e verificar os modelos. Por isto, na concepção do DMS&C, a ferramenta de síntese utilizada é o Vivado síntese 2014 (Versão 2014.1). Entretanto, o simulador utilizado é o Vivado **Simulator**, componente do **Vivado Design Suite**. Este é um simulador de linguagem compilada, que suporta a linguagem mista, os scripts da linguagem de comandos de ferramentas (**Tool Command Language - TCL**), o protocolo de internet (**Internet Protocol** – IP) criptografado e uma verificação reforçada.

O integrador IP tem acesso ao catalogo IP da Xilinx (incluindo núcleos de processadores embarcados) e ao kit de desenvolvimento de software - SDK, para completar o projeto. O fluxo da implementação do projeto pode ser iniciado centralmente a partir da GUI Vivado, utilizando o estilo de projeto “**top-down**”. O integrador IP pode ser utilizado para projetar a porção de hardware do seu sistema do processador incorporado. Pode-se especificar o núcleo do processador ARM Cortex-A9, dispositivos periféricos do integrador IP e a interligação destes componentes, juntamente com a sua respectiva configuração detalhada.

O kit de desenvolvimento de software SDK (**Software Development Kit**) é um ambiente de desenvolvimento integrado, complementar ao processo de design, que é empregado para a criação e verificação da aplicação do software embarcado C ou C++. O SDK é construído sobre a estrutura de código aberto Eclipse, originado a partir do VisualAge da IBM. Na atualidade, o Eclipse é um dos principais ambientes integrados de desenvolvimento de software (IDE) para a plataforma Java.

Considerando a placa ZedBoard Zynq™-7000 utilizada no desenvolvimento do DMS&C do alimentador de distribuição estudo de caso, segundo a figura 39, pode-se visualizar que esta placa já tem um suporte padrão na ferramenta **Vivado Design Suite**, o que, além de facilitar o estabelecimento da configuração inicial da mesma, possibilita o desenvolvimento do projeto digital.

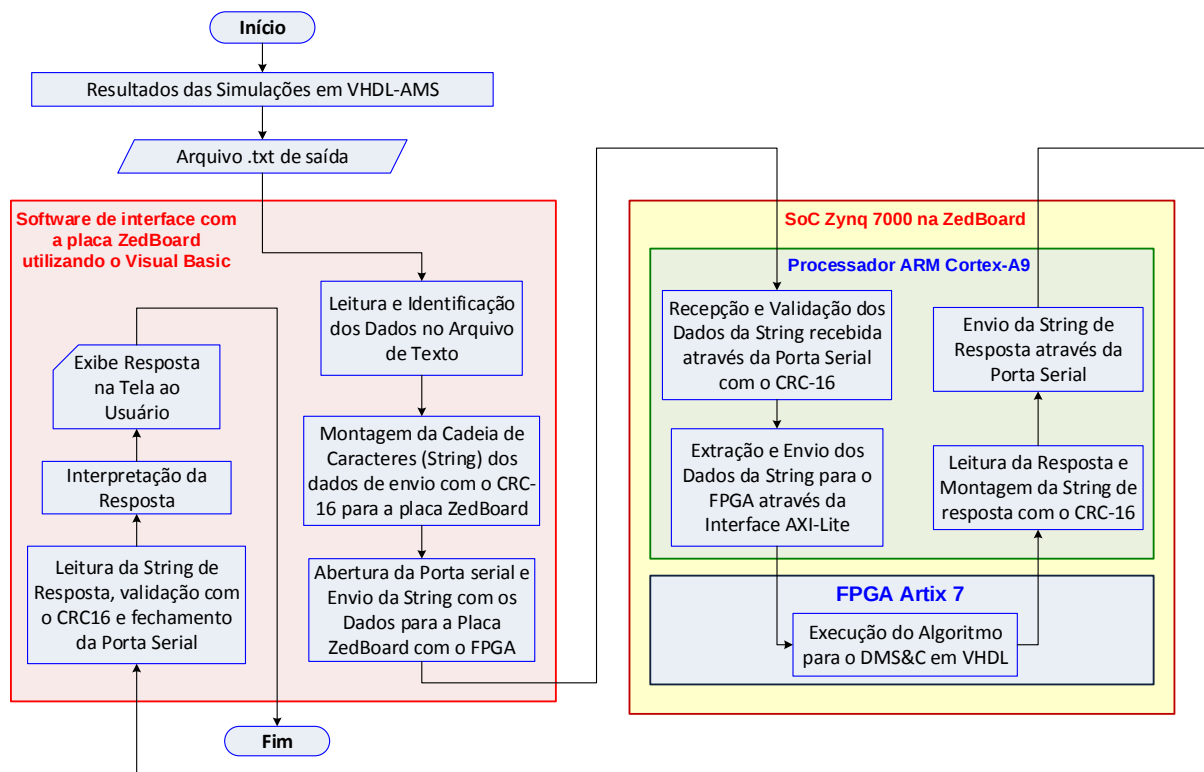
Figura 39- Seleção da placa ZedBoard no ambiente **Vivado Design Suite**.



Fonte: Próprio autor utilizando o **Vivado Design Suite**.

Considerando que toda a lógica digital do DMS&C foi desenvolvida em um dispositivo FPGA Artix 7, usando a linguagem VHDL, com o auxílio da ferramenta **Vivado Design Suite**, a recepção e envio de informação com relação ao dispositivo FPGA foi desenvolvida segundo a figura 40.

Figura 40- Fluxograma geral do desenvolvimento do DMS&C.

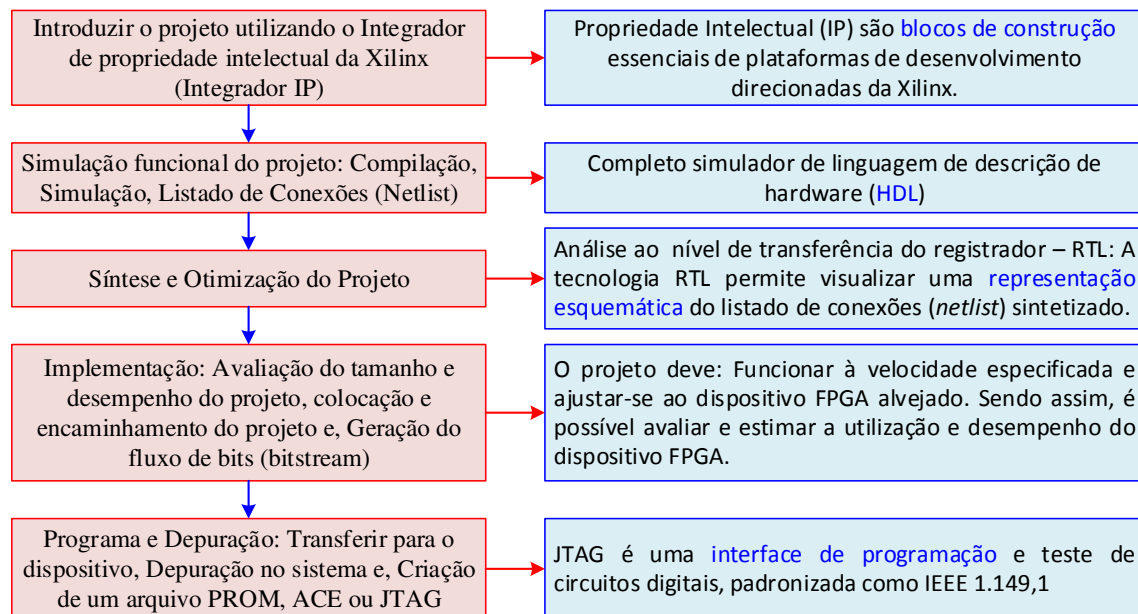


Fonte: Próprio autor.

3.6.1 Metodologia de desenvolvimento do DMS&C

O ambiente de projeto integrado (IDE) **Vivado Design Suite** possui a sequência de fluxo de projeto de acordo com a figura 41.

Figura 41- Fluxo de projeto no dispositivo FPGA (utilizando a ferramenta **Vivado Design Suite**).



Fonte: Próprio autor

Baseados no fluxo de projeto no dispositivo FPGA (figura 41), foi implementado o sistema de gerenciamento da distribuição e controle (DMS&C), de acordo com a seguinte metodologia: Utilizando o ambiente de projeto integrado (IDE) da Xilinx, **Vivado Design Suite**, cria-se um novo projeto e escolhe-se a placa ou o dispositivo que será usado nesta concepção (segundo a figura 39). Posteriormente, adicionam-se os arquivos que serão utilizados no projeto, criando a propriedade intelectual (IP), ou seja, utilizando-se o integrador IP do Vivado, que tem acesso ao catálogo IP da Xilinx, são integrados os blocos funcionais que compõem o projeto (segundo o item 3.6.3.1: Implementação dos Blocos Funcionais do Projeto do DMS&C). Destes blocos funcionais são derivados os códigos fontes que são sintetizados pelo **Vivado Design Suite** e implementados no dispositivo FPGA Artix-7 (estes códigos fontes são nomeados: “Regulacao_Transf_v1_0.vhd”, “Regulacao_Transf_v1_0_S00_AXI_TRASNf_parameterized0.vhd” e “Regulacao.vhd”, encontram-se no apêndice D, itens D4, D5 e D6, respectivamente).

Quando os blocos funcionais do projeto estão concluídos, o projeto é sintetizado e otimizado, onde, pela primeira vez, os erros de sintaxe da linguagem do código (no nosso caso a linguagem VHDL) são apresentados para correção e então, é gerado o circuito equivalente do código ao nível de transferência do registrador RTL.

Após a correta execução da sintetização, executa-se a implementação do circuito gerado anteriormente ao nível de RTL, para a placa ou dispositivo (FPGA) selecionado no projeto (no nosso caso para a placa ZedBoard Zynq), na qual será convertida a implementação para o hardware real selecionado (no nosso caso o FPGA Artix-7).

Em seguida, através do **Vivado Design Suite**, é gerado um fluxo de bits, ou “**bitstream**” (implementação do projeto), que será a sequência em que o FPGA Artix-7 deverá configurar suas portas e barramentos internos para remontar o circuito gerado pelo projeto ao nível de RTL. O “**bitstream**” pode ser considerado como uma série temporal de bits. O arquivo “**bitstream**” é gerado pelo **Vivado Design Suite** e deve ser exportado para o kit de desenvolvimento de software – SDK (**Software Development Kit**), que por sua vez, está contido no pacote licenciado do **Vivado Design Suite**. No projeto foi utilizada a versão Xilinx SDK 2014.1.

Posteriormente, o projeto é continuado criando um software, em ambiente SDK, que será executado pelo processador ARM Cortex-A9 (este código fonte é nomeado “main.c” e encontra-se no apêndice D, item D7). Este software, desenvolvido em linguagem C, tem a função de verificar o status da porta serial que é emulada no PC pela placa de desenvolvimento ZedBoard Zynq™-7000, coletando todos os dados recebidos por ela e verificando se os mesmos são dados de um pacote válido. Sendo um pacote válido, o próximo passo é extrair os dados do pacote e enviá-los para o FPGA Artix-7, através da interface AXI4-Lite. Após o processamento dos dados pelo FPGA, o resultado é lido através da interface AXI4-Lite, para depois ser montado um pacote de dados de resposta e enviado à porta serial. Após todo o código ser escrito e validado, o projeto do software é compilado pelo SDK e executado pelo processador ARM Cortex-A9.

A porta serial é emulada no PC pela placa de desenvolvimento ZedBoard, pois existe um Circuito Integrado (CI) da empresa **Cypress** que é um conversor USB↔Serial na placa. Esse CI, quando está conectado ao PC, cria uma porta serial virtual no PC, na qual, quando o PC tem o sistema operacional Windows, passa a chama-la de porta COM e a partir daí o Windows entende que o PC tem uma porta serial física. No lado da placa (dentro da placa ZedBoard) o CI é ligado diretamente na porta serial do processador ARM. Assim é feita a conversão de interface de USB

para serial e vice-versa. Quando a placa ZedBoard é utilizada pela primeira vez, e é necessário utilizar o conversor USB↔Serial, deve-se instalar no PC (com o Windows) o **driver “Cypress USB to UART”** (baixado da web site da **Cypress**).

Depois, utilizando as ferramentas computacionais do SDK, cria-se um novo projeto que será responsável pela inicialização da placa ZedBoard Zynq™-7000, sempre que ela é energizada, ou seja, o projeto que gerencia o processo de **“boot”** da placa. Esse novo projeto incluirá o arquivo de **“bitstream”** gerado pelo Vivado. Este projeto que executa a inicialização da placa, apesar de ser um novo projeto, será criado na mesma árvore de arquivos em que está o projeto do software **“main.c”**. Essa árvore pode ser considerada como um único projeto maior que engloba os outros projetos dentro dela.

Esse único projeto maior, que contém o código fonte **“main.c”**, mais o processo de **“boot”** (que, ao mesmo tempo, contém o **“bitstream”** do dispositivo FPGA) é gravado na raiz do SD Card (de 4 GB de memória) da ZedBoard Zynq™-7000. Esse arquivo, derivado deste projeto maior, é nomeado **“boot.bin”** e o processador ARM Cortex-A9 fará a leitura dele para configurar o FPGA Artix-7 e dar início à execução do projeto do DMS&C. Com o início do projeto, o dispositivo ZYNQ SoC XC7Z020-CLG484 tem estabelecido uma comunicação com uma porta serial que é emulada no PC pela placa ZedBoard. O arquivo **“boot.bin”** é um arquivo binário otimizado para a interpretação e inicialização da placa e que foi gerado pelo SDK (por isso não é possível visualizar o seu conteúdo), daí o nome da extensão.

Finalmente, através do Software de Interface entre o PC e a placa ZedBoard Zynq™-7000 (item 3.6.1), em termos gerais, são enviados os resultados da simulação em VHDL-AMS (em um arquivo de texto) para a porta serial e após da execução do algoritmo digital no FPGA Artix-7, são visualizados, no PC, os resultados da simulação digital em VHDL, feita pelo dispositivo FPGA. O código fonte deste software de interface é nomeado **“mainModuleChk.bas”** e encontra-se no apêndice D, item D8.

Assim, segundo o fluxograma geral e a metodologia de desenvolvimento do DMS&C (figura 40), são necessárias duas plataformas de comunicação e projeto digital para a execução integral do sistema de gerenciamento da distribuição e controle DMS&C: 1) Software de interface entre o PC e a placa ZedBoard (área vermelha); 2) Configuração da estrutura interna do dispositivo ZYNQ SOC XC7Z020-CLG484 (área amarela). Estas duas plataformas serão abordadas nos itens seguintes, segundo os itens 3.6.2 e 3.6.3, respetivamente.

3.6.2 Software de Interface entre o PC e a placa ZedBoard

Este software foi desenvolvido em ambiente Visual Basic (VB), linguagem de programação produzida pela empresa Microsoft, que é parte integrante do pacote Microsoft Visual Studio. No apêndice D (item D8) é apresentado o código fonte deste software, cujo nome é: “mainModuleChk.bas”.

O software tem a função de estabelecer uma interface de comunicação entre o PC e a placa ZedBoard. Neste sentido, e segundo a figura 40, o software: 1) Comunica a placa ZedBoard com uma porta serial que é emulada no PC pela mesma placa, utilizando o **Universal Serial Bus** (USB) do PC; 2) Envia os resultados da simulação em VHDL-AMS (em um arquivo de texto) para a porta serial (emulada no PC); 3) Estabelece um tempo de espera para a execução do algoritmo digital no FPGA Artix-7 e 4) Visualiza, no PC, os resultados da simulação digital em VHDL, feita pelo dispositivo FPGA. Para uma maior amplitude da funcionalidade deste software, apresentam-se detalhadamente as suas cinco funções essenciais:

- **Função 1:** O nome desta primeira função é “SerialEnvio”, o argumento é objeto mapeando o Formulário Principal e não tem nenhum valor de retorno. A descrição desta função é explicada pelas seguintes atividades: a) Abre a porta serial, b) Envia uma string para a porta serial (chamando a função “SerialEscrita”), c) Aguarda recepção da string de resposta (chamando a função “SerialLeitura”) e d) Fecha a porta serial após a recepção da string da função “SerialLeitura”.
- **Função 2:** O nome desta segunda função é “SerialEscrita”, os argumentos são: Objeto mapeando o Formulário Principal e Variável do tipo texto (prefixo) e não tem nenhum valor de retorno. A descrição desta função é: se o prefixo for “AB” envia pela porta serial a string retornada pela função “LeituraDadoFPGA”. O prefixo AB é definido segundo o Protocolo de Comunicação Serial (item 3.6.3.1).
- **Função 3:** O nome desta terceira função é “LeituraDadoFPGA”, o argumento é objeto mapeando o Formulário Principal e o valor de retorno é uma **string** de texto. A descrição desta função é explicada pelas seguintes atividades: a) Abre o arquivo de dados (texto) a ser enviado para o FPGA (com os resultados da simulação em VHDL-AMS), b) Lê linha a linha o arquivo de texto validando seu conteúdo e montando **string** de envio para o FPGA e c) Retorna a **string** formatada para envio para o FPGA.

- **Função 4:** O nome desta quarta função é “SerialLeitura”, os argumentos são: Objeto mapeando o Formulário Principal, Variável do tipo texto (prefixo) e Variável do tipo numérica (tempo). Esta função não tem nenhum valor de retorno. A descrição desta função é explicada pelas seguintes atividades: a) Durante tempo predefinido, aguarda pela porta serial a recepção de **string** de texto contendo o prefixo esperado e b) Caso receba a **string** de texto válida encaminha a **string** para a função “SerialProcessa”.
- **Função 5:** O nome desta quinta função é “SerialProcessa”, os argumentos são: Objeto mapeando o Formulário Principal e variável do tipo Texto (mstr). Esta função não tem nenhum valor de retorno. A descrição desta função é: Processa a **string** de texto recebida do FPGA via comunicação serial e exibe os resultados no Formulário Principal.

3.6.2.1 *Protocolo de Comunicação Serial*

Considerando que um protocolo é uma convenção que controla e possibilita uma conexão, comunicação e transferência de dados entre dois sistemas computacionais, foi estabelecido um protocolo de comunicação serial para a troca de informações entre o software de interface do PC, e o SoC Zynq 7000 na ZedBoard (dispositivo ZYNQ SOC XC7Z020-CLG484), através de uma porta serial que é emulada no PC pela placa ZedBoard. Uma introdução ao protocolo está descrita abaixo.

Na sua forma básica, o protocolo é a montagem de um pacote de mensagem caracterizado pela sequência a seguir: Cabeçalho, N° de bytes da mensagem, Mensagem e CRC-16 (**Cyclic Redundancy Check**, em português: verificação de redundância cíclica). Nessa sequência:

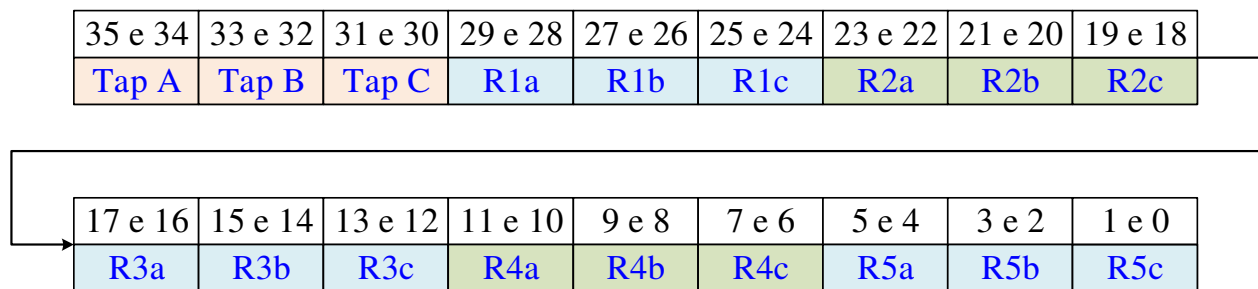
- O cabeçalho é definido de acordo com o sentido em que a mensagem trafega, ou do PC para o SoC Zynq 7000 ou do Soc Zynq 7000 para o PC. Este cabeçalho sempre terá 2 bytes de tamanho;
- N° de bytes é a quantidade de bytes da mensagem que será trafegada e sempre terá 1 byte de tamanho;
- A mensagem, a qual pode conter: a) Os dados das posições dos TAPs no regulador de tensão e a regulação para as três fases das cinco piores cargas no alimentador, ou b) A resposta da análise do alimentador (feita pelo FPGA), sempre de acordo com o número de bytes informado anteriormente;

- O valor de conferência de integridade dos dados trafegados, que sempre terá 2 bytes de tamanho e é calculado com todos os bytes contidos no pacote.

Com base nesse protocolo de comunicação foi definida a mensagem trafegada do PC para o SoC Zynq 7000, na qual:

- O cabeçalho foi definido com os caracteres “A” e “B”, de acordo com a tabela ASCII, completando os 2 bytes de tamanho;
- O número de bytes é fixo em 36, pois são trafegadas 18 informações e foram dedicados 2 bytes para cada informação, totalizando 36 bytes;
- A mensagem segue a sequência segundo a seguinte figura 42:

Figura 42- Sequência da mensagem trafegada do PC para o SoC Zynq 7000.



Fonte: Próprio autor

- Considerando que o CRC (**Cyclic Redundancy Check**) é um número verificador usado para verificar a integridade de um conjunto de dados, o CRC16 é calculado com todos os bytes contidos no pacote.

A mensagem que é trafegada do SoC Zynq 7000 para o PC tem o seguinte conteúdo:

- O cabeçalho foi definido com os caracteres “B” e “C”, de acordo com a tabela ASCII, completando os 2 bytes de tamanho;
- O número de bytes é fixo em 1, pois somente é retornado um byte de resposta que contém a resposta de cada fase e do alimentador;
- A resposta de 1 byte;
- O CRC16 é calculado com todos os bytes contidos no pacote.

3.6.3 Configuração da estrutura interna do dispositivo ZYNQ SOC XC7Z020-CLG484

Para implementar a configuração da estrutura interna do dispositivo ZYNQ SOC XC7Z020-CLG484, foram tidas em conta todas as ferramentas oferecidas pelo **Vivado Design Suite**, incluindo o kit de desenvolvimento de software – SDK (**Software Development Kit**), além das características de hardware da placa de desenvolvimento ZedBoard Zynq™-7000.

3.6.3.1 Implementação dos Blocos Funcionais do Projeto do DMS&C

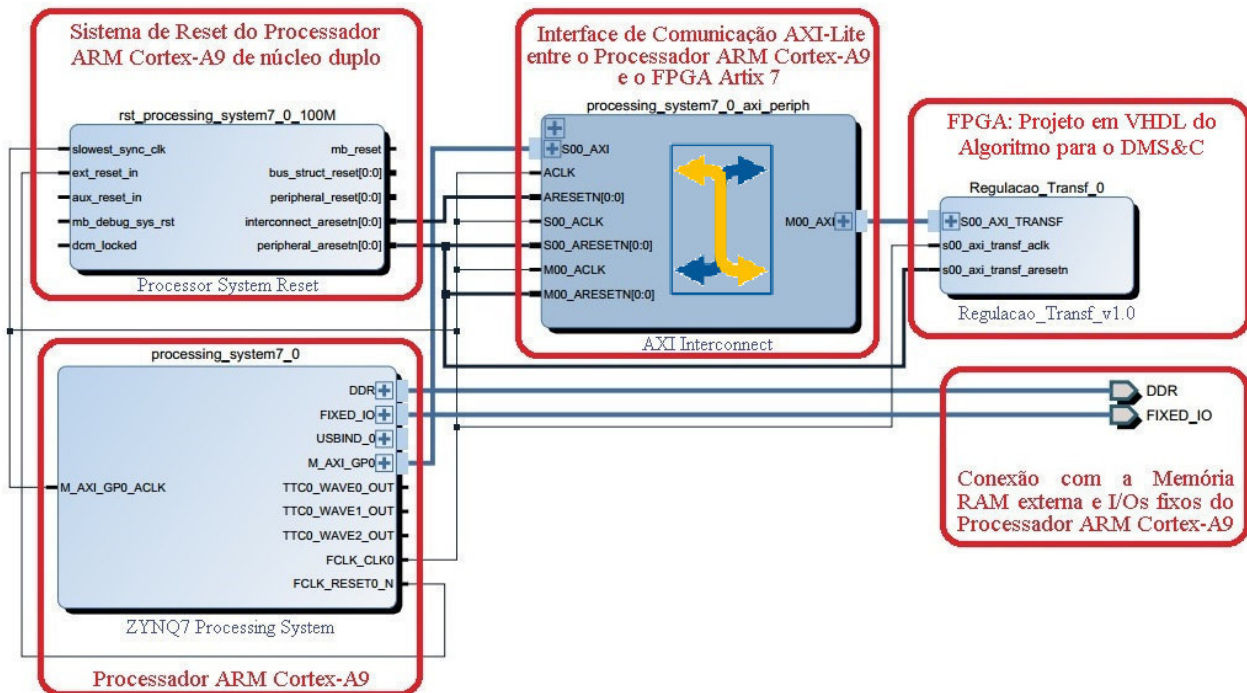
Foram utilizadas as ferramentas do **Vivado Design Suite** para a concepção do projeto, baseados especificamente no integrador de propriedade intelectual (IP) da Xilinx. Os blocos resultantes encontram-se esquematizados na figura 43.

A síntese lógica é um processo pelo qual uma forma abstrata de comportamento de um circuito desejado (descrição em VHDL, por exemplo), é transformado em uma implementação de projeto em termos de portas lógicas, normalmente por um programa de computador nomeado ferramenta de síntese. O **Vivado Design Suite** permite sintetizar o projeto através da descrição da operação de um circuito síncrono digital chamado nível de transferência do registrador RTL (do inglês **Register Transfer Level**). Em um projeto RTL, o comportamento do circuito é descrito em termos do fluxo de sinais (ou transferência de dados entre os registradores presentes no hardware e as operações lógicas conduzidas com estes sinais). No **Vivado Design Suite** a abstração RTL é usada em VHDL na representação do circuito de baixo nível do qual derivam-se as conexões do hardware. Um registrador é um circuito digital capaz de armazenar e deslocar informações binárias. Os registradores são tipicamente usados como dispositivos de armazenamento temporário. A descrição RTL é usualmente convertida para a descrição de circuitos em “**gate-level**” pela ferramenta de síntese lógica. Esta síntese resultante representa a descrição física do circuito. Na figura 45 pode-se observar o esquema simplificado do hardware resultante do código VHDL, utilizado no projeto do DMS&C. No apêndice D (item D9), podem-se observar os esquemas completos do hardware resultante no dispositivo FPGA, relacionado com o projeto.

Posteriormente, as ferramentas de simulação lógica podem utilizar a descrição RTL para verificar sua funcionalidade. Por outro lado, se existe um caminho lógico cíclico da saída de um registrador para a sua entrada (ou da saída de um conjunto de registradores para suas entradas), este circuito é chamado de máquina de estados. Se existem caminhos lógicos de um registrador

para outro, que operam sem a necessidade de ciclos de relógio, estes blocos são chamados pipelines.

Figura 43- Diagrama de blocos do projeto do DMS&C.



Fonte: Próprio autor utilizando o **Vivado Design Suite**.

O primeiro estágio está relacionado com o sistema de reset do processador ARM Cortex-A9 de núcleo duplo, responsável pela distribuição e sincronização do sinal de reset em todos os dispositivos criados e adicionados no projeto dentro do ambiente do **Vivado Design Suite**.

O segundo estágio está relacionado diretamente com processador ARM Cortex-A9, o qual arranja a comunicação entre o PC e o FPGA Artix-7. O processador ARM recebe o pacote de dados do PC, valida o pacote, extrai os dados do pacote e encaminha esses dados para o FPGA Artix-7. Depois que o FPGA processa o algoritmo para a execução do sistema de gerenciamento da distribuição e controle (DMS&C), o processador ARM faz a leitura do resultado e monta um pacote de dados para enviar ao PC novamente.

O terceiro estágio está relacionado com a interface de comunicação AXI-Lite entre o processador ARM Cortex-A9 e o dispositivo FPGA Artix-7, o qual é responsável por trafegar os dados entre o processador ARM e o FPGA.

O quarto estágio está relacionado com a conexão com a memória RAM externa e I/Os fixos do processador ARM Cortex-A9. A conexão com a memória RAM externa é necessária devido a que o ARM Cortex-A9, dentro do dispositivo ZYNQ SoC XC7Z020-CLG484 (referenciado na placa de desenvolvimento ZedBoard Zynq™-7000 como Zynq) é um processador, e não um microcontrolador. Por isto, ele necessita de acesso a uma memória RAM para a criação e manipulação de variáveis que são declaradas no arquivo, em ambiente SDK, que será executado pelo processador ARM (código fonte designado “main.c”) e em todo o projeto que envolva firmware (conjunto de instruções operacionais programadas diretamente no hardware de um equipamento eletrônico, para nosso caso o FPGA). O bloco de I/Os fixos define as portas (pinos) do Zynq que são ligados diretamente ao ARM e que não tem contato com o FPGA Artix-7. O Zynq possui vários pinos de interface em seu encapsulamento, a maioria deles são ligados ao FPGA e a minoria são ligados ao ARM, que são declarados e ligados nesse bloco ao processador ARM.

Finalmente, no quinto estágio, pode-se notar que o bloco: “FPGA: Projeto em VHDL do Algoritmo para o DMS&C”, o qual, para efeitos do ambiente do **Vivado Design Suite**, é nomeado: “Regulacao_Transf_v1.0”, é responsável pela execução do algoritmo da figura 34, pelo qual é definido o sistema de gerenciamento da distribuição e controle (DMS&C) e que é executado no dispositivo FPGA Artix-7. Na figura 44 é feito um detalhamento interno deste bloco “Regulacao_Transf_v1.0”.

De acordo com a figura 44, todo o projeto sobre o dispositivo FPGA Artix-7 pode ser referenciado ao bloco “Regulacao_Transf_v1.0”, que foi desenvolvido em VHDL com o auxílio da ferramenta **Vivado Design Suite** da Xilinx. Segundo a figura 44, pode-se verificar que o bloco “Regulacao_Transf_v1.0” é constituído por outros dois blocos, o bloco “Regulacao” e o bloco “Regulacao_Transf_v1_0_S00_AXI_TRANSF__parameterized0”. Este bloco “Regulacao_Transf_v1.0” tem a função de estabelecer a conexão dos barramentos de dados criados dos anteriores dois blocos. O código em VHDL desse bloco “Regulacao_Transf_v1.0” foi gerado por um assistente de configuração e criação da interface AXI-Lite e posteriormente modificado para as necessidades do projeto. Este código fonte é sintetizado pelo **Vivado Design Suite** e implementado no FPGA Artix-7. Pode-se ver este código no apêndice D (item D4).

O bloco “Regulacao_Transf_v1_0_S00_AXI_TRANSF__parameterized0” é responsável pelo tráfego de dados entre o processador ARM Cortex-A9 e o bloco “Regulacao”, segundo o Diagrama demonstrativo do bloco FPGA (figura 44). A interface com o processador ARM Cortex-

A9 é feita através de um barramento de dados bidirecional AXI-lite, no qual os dados são organizados individualmente em um banco de registradores. O código em VHDL desse bloco foi gerado por um assistente do **Vivado Design Suite** para a criação dessa interface e posteriormente foi adaptado para uso no projeto. Este código é sintetizado pelo **Vivado Design Suite** e implementado no FPGA Artix-7. Pode-se ver este código fonte no apêndice D (item D5).

O algoritmo do Sistema de Gerenciamento da Distribuição e Controle (DMS&C), da figura 34, foi implementado diretamente no bloco “Regulacao”. Assim, o código em VHDL desse bloco foi desenvolvido utilizando-se o algoritmo da figura 34. Pode-se ver este código fonte no apêndice D (item D6). Os dados das posições de cada TAP do Regulador de Tensão (para cada fase) e as magnitudes das regulações de tensão para cada fase (das cinco piores cargas do alimentador), são recebidos, por barramentos paralelos de dados no FPGA, do bloco “Regulacao_Transf_v1_0_S00_AXI_TRANSF__parameterized0”, através dos registradores onde estão armazenados. Os dados passam pela análise do algoritmo, o qual retorna um dado (uma sequência de 7 bits), em outro barramento paralelo de dados. A sequência de 7 bits está organizada na seguinte forma:

TABELA 19- SEQUÊNCIA DE BITS RESULTANTE DO ALGORITMO.

Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Resp_A	Resp_A	Resp_B	Resp_B	Resp_C	Resp_C	Resp_Alimentador
[1]	[0]	[1]	[0]	[1]	[0]	[1]

Fonte: Próprio autor.

Segundo a anterior tabela 19:

Resp_A [1..0] = Resposta da fase A;

Resp_B [1..0] = Resposta da fase B;

Resp_C [1..0] = Resposta da fase C;

Resp_Alimentador [1]= Resposta geral do alimentador.

As respostas das fases A, B e C são interpretadas como:

00 → aumentar TAP no Regulador de Tensão;

01 → diminuir TAP no Regulador de Tensão;

10 → atende as normas Brasileiras;

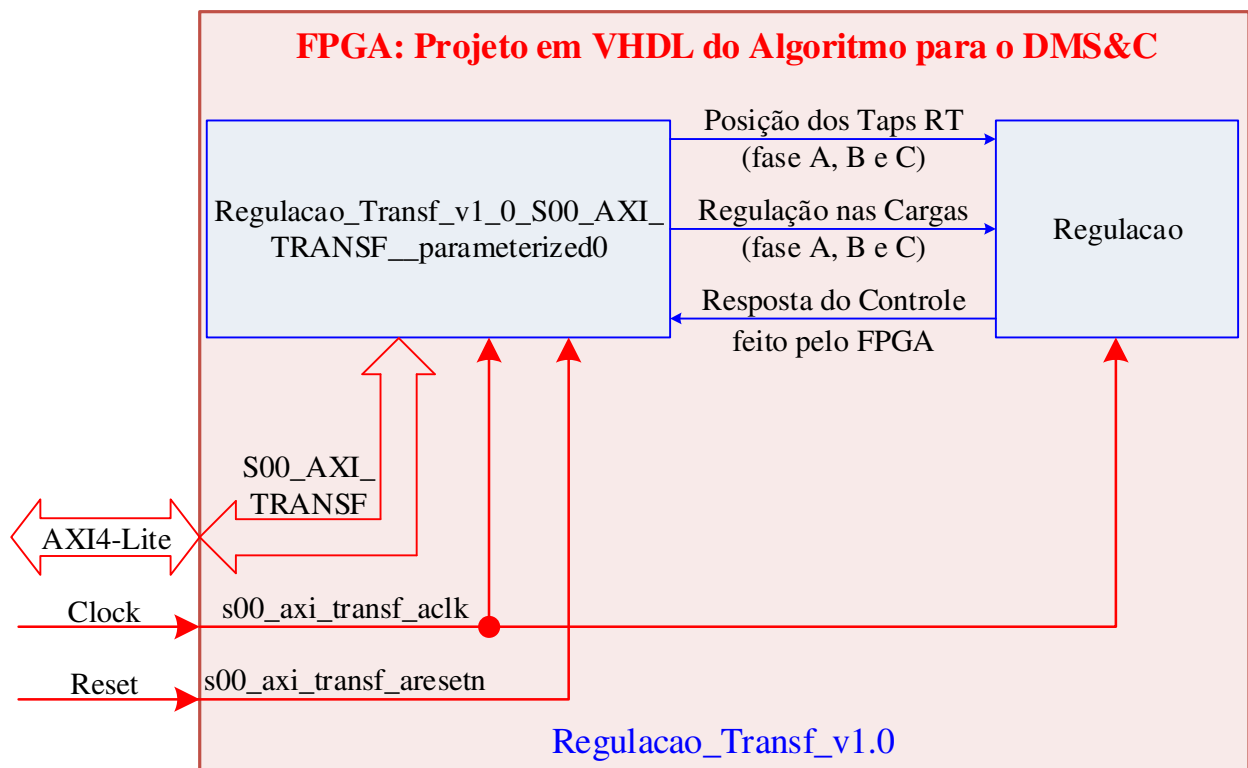
11 → não atende as normas Brasileiras.

A resposta geral do alimentador é interpretada como:

- 0 → O alimentador não atende as normas;
- 1 → O alimentador atende as normas.

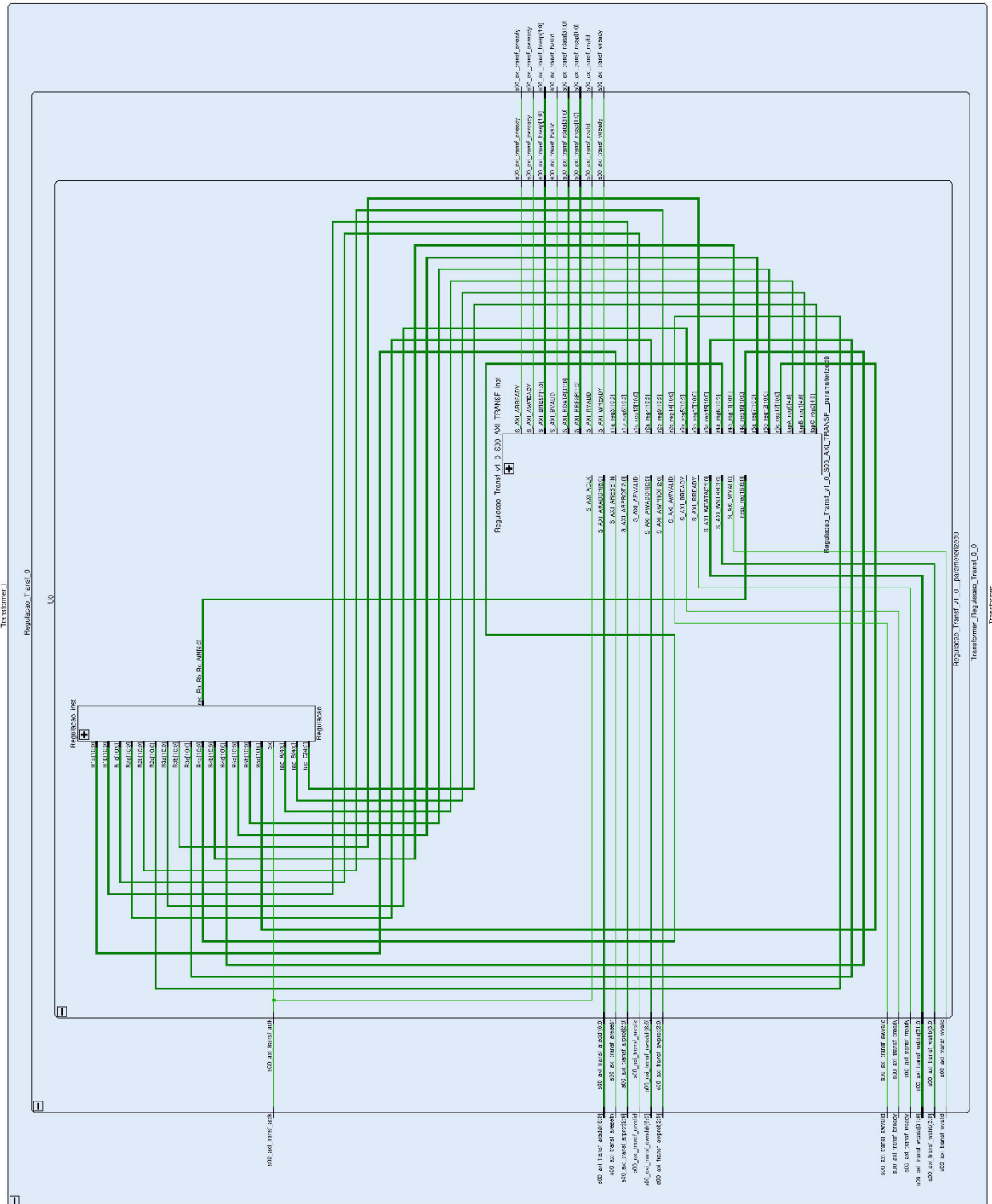
Esses 7 bits de retorno, derivados da execução do algoritmo no FPGA Artix-7, são enviados para o bloco “Regulacao_Transf_v1_0_S00_AXI_TRANSF__parameterized0”, o qual se encarrega de disponibilizar os resultados para o processador ARM Cortex-A9.

Figura 44- Diagrama demonstrativo do bloco FPGA Artix-7.



Fonte: Próprio autor.

Figura 45- Hardware resultante da síntese do código VHDL (Esquema Simplificado): No apêndice D (item D9), podem-se observar os esquemas completos.



Fonte: Próprio autor utilizando o **Vivado Design Suite**.

A tabela 20 mostra os dados estatísticos referentes aos recursos requeridos no dispositivo ZYNQ para a implementação do sistema de gerenciamento da distribuição e controle (DMS&C). Estes resultados foram obtidos da ferramenta **Vivado Design Suite**.

TABELA 20- DADOS ESTATÍSTICOS ESTIMADOS COM O PROCESSO DE IMPLEMENTAÇÃO DO DMS&C.

Dispositivo ZYNQ SOC XC7Z020-CLG484			
Recursos	Utilizado	Disponível	Porcentagem
Número de Slices	443,0	13.300,0	3,3 %
Número de Flip-flops do Slice	1.156,0	106.400,0	1,1 %
Número de LUT como Lógica	1.256,0	53.200,0	2,4 %
Número de LUT como Memória	62,0	17.400,0	0,4 %
Número de LUT Flip Flop Pairs	1.446,0	53.200,0	2,7 %
Número de Clocking BUFGCTRL	1,0	32,0	3,1 %

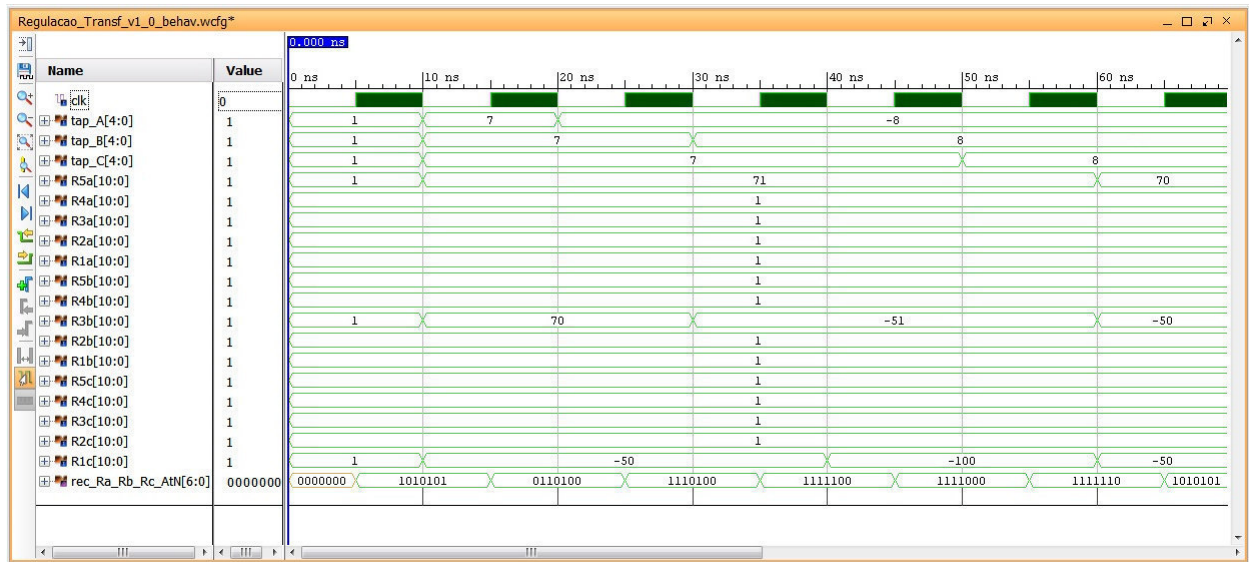
Fonte: Próprio autor utilizando o **Vivado Design Suite**.

3.6.4 Resultados das simulações digitais do DMS&C

Segundo a figura 46, utilizando a ferramenta **Vivado Design Suite**, pode-se visualizar os resultados das simulações digitais do DMS&C, especificamente do bloco “Regulacao_Transf_v1.0”, no qual é executado o algoritmo da figura 34.

No caso particular do exemplo da simulação digital da figura 46, os TAP A, B e C estão na mesma posição “1”. Cada borda de subida representa um “**clock**” de 10 nano segundos e de acordo com a figura 46, estão representados 7 “**clocks**” totais na simulação. Segundo a Implementação dos Blocos Funcionais do Projeto do DMS&C (item 3.6.3.1), no primeiro “**clock**”, o resultado pode ser interpretado assim: 10 = a fase A atende as normas, 10 = a fase B atende as normas, 10 = a fase C atende as normas, 1 = o alimentador atende as normas. No segundo “**clock**” o resultado pode ser interpretado assim: 01 = a fase A não atende as normas, 10 = a fase B atende às normas, 10 = a fase C atende as normas, 0 = o alimentador não atende as normas. Cada “**clock**” tem uma situação diferente até que foi piorando até que nenhuma fase atende as normas (1111110), e consequentemente o alimentador não atende às normas brasileiras.

Figura 46- Resultado de simulação da descrição VHDL do componente “Regulacao” (algoritmo de controle da figura 34).



Fonte: Próprio autor utilizando o **Vivado Design Suite**.

Segundo o exemplo anterior e baseados no diagrama demonstrativo do bloco FPGA (figura 44), o processador ARM Cortex-A9 coleta os 18 dados (15 regulações mais 3 TAPs) da porta serial que é emulada no PC pela placa ZedBoard (provenientes do PC, das simulações em VHDL-AMS), extrai as informações e a carrega nos registradores do FPGA, através da interface de comunicação AXI-Lite. Posteriormente, depois de que é executado o algoritmo no bloco “Regulacao_Transf_v1.0”, especificamente no bloco “Regulacao”, o FPGA faz o processamento e devolve a resposta em 7 bits para o bloco “Regulacao_Transf_v1_0_S00_AXI_TRANSF__parameterized0”, o qual se encarrega de disponibilizar os resultados para o processador ARM Cortex-A9. Este processamento é feito pelo dispositivo FPGA em um “**clock**” (10 nano segundos).

3.7 CONCLUSÕES

Neste capítulo foi desenvolvido o sistema de gerenciamento e controle (DMS&C) de um alimentador elétrico de distribuição de 14,4 kV de tensão de linha, estudo de caso, o qual gerencia o perfil da tensão deste alimentador, através de um dispositivo ZYNQ SOC XC7Z020-CLG484, que encapsula um FPGA Artix-7 e um processador ARM Cortex-A9, utilizando-se uma placa de desenvolvimento ZedBoard Zynq™-7000. Os recursos de hardware e software empregados mostraram uma alta eficiência do dispositivo ZYNQ SOC para realizar o trabalho, inclusive para ampliações futuras. A linguagem de descrição de hardware utilizada no dispositivo FPGA Artix-7 foi VHDL (***Very High speed Integrated circuit hardware Description Language***).

A utilização da linguagem VHDL, em conjunto com as ferramentas de síntese do Xilinx ***Vivado Design Suite***, garantiram qualidade e robustez ao projeto, uma vez que a especificação do hardware poderá ser submetida a experimentos de validação, através das etapas de simulação. Além disso, o VHDL permite o desenvolvimento no estilo de projeto “***top-down***”, baseado em funções com elevado nível de abstração, ao invés de projetos elaborados ao nível de portas lógicas.

Neste trabalho, evidenciou-se que os dispositivos baseados no sistema-em-um-chip (SoC), em conjunto com as ferramentas de desenvolvimento fornecidas pelo fabricante, tornaram-se elementos essenciais na implementação de sistemas digitais de forma rápida (da ordem dos nano segundos), com elevada flexibilidade (podem-se fazer modificações futuras), elevado desempenho (a placa de desenvolvimento ZedBoard apresenta alta versatilidade nas suas funcionalidades) e apresentando um reduzido custo (da ordem dos 500 dólares americanos). O DMS&C foi, portanto, desenvolvido com uma arquitetura baseada no sistema-em-um-chip (SoC) flexível e programável, na qual o processador ARM se comporta como mestre do sistema de controle (disponibiliza os dados para o FPGA) e o FPGA como escravo (fica aguardando as instruções do ARM). Esta disposição facilitou a comunicação entre o dispositivo FPGA Artix-7 e o PC, fundamentalmente porque o processador ARM faz todo o trabalho de comunicação entre o PC e o FPGA Artix-7, utilizando-se as ferramentas e interfaces de comunicação do processador ARM Cortex-A9, especialmente a interface de registro de controle AXI4-Lite (conhecida também AXI-Lite).

A implementação total do sistema de gerenciamento da distribuição e controle (DMS&C), utilizou apenas 3,3% dos recursos de “***slices***” disponíveis pelo dispositivo, sendo que a máxima frequência de operação permitida pelo sistema global é 866 MHz. Portanto, permitindo grandes

evoluções para o sistema de controle digital, envolvendo outros diversos elementos de controle da rede de distribuição, para as continuidades futuras desta pesquisa.

ARQUITETURA PROPOSTA PARA SIMULAÇÃO EM “TEMPO REAL” E CONTROLE DE ALIMENTADOR DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA, UTILIZANDO VHDL-AMS e VHDL

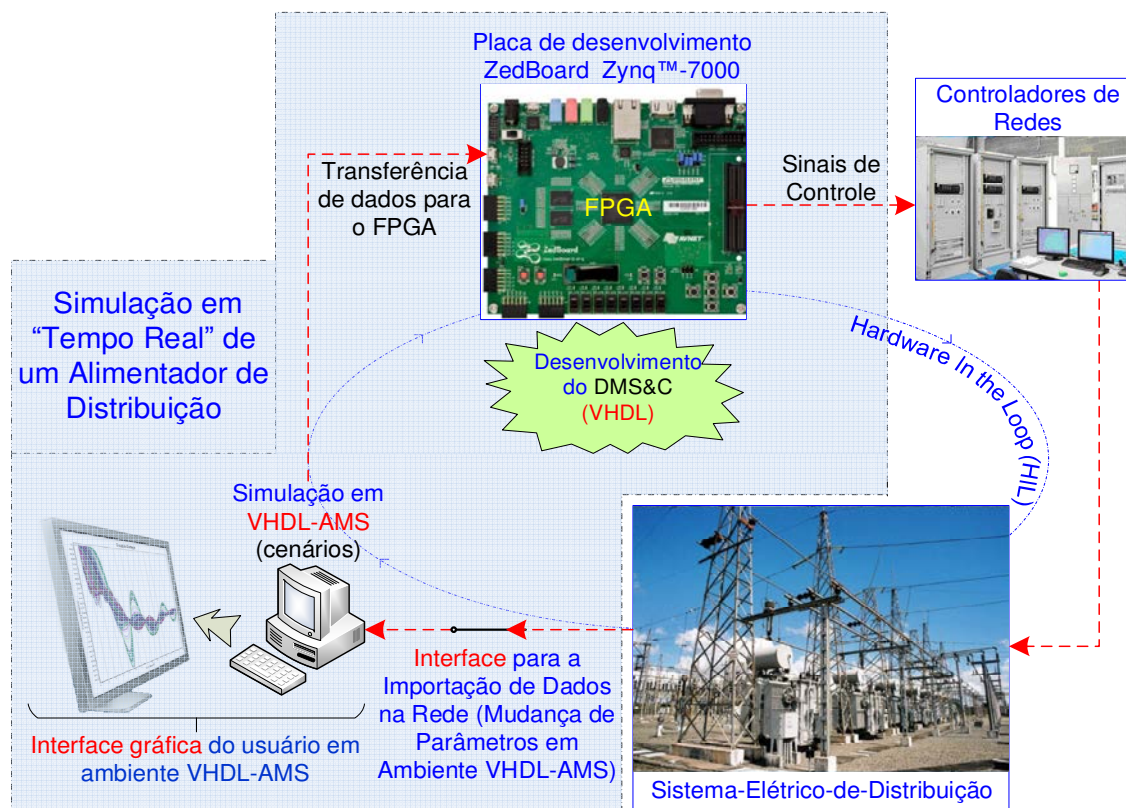
4.1 INTRODUÇÃO

A arquitetura proposta e desenvolvida para simulação em “tempo real” e controle de um alimentador de distribuição de energia elétrica está baseada em três características:

- Contempla um controlador digital programável para fazer o gerenciamento e controle do perfil de regulação de tensão do alimentador, utilizando a linguagem VHDL. Este controlador foi apresentado em 3.6 e está constituído com os seguintes dois elementos principais:
 - ✓ Uma interface para a transferência de dados entre o PC e a placa de desenvolvimento ZedBoard Zynq™-7000 (a qual contém o dispositivo FPGA, no qual é empregada a linguagem VHDL). Esta interface é desenvolvida utilizando o software Visual Basic/Microsoft.
 - ✓ Uma Placa de desenvolvimento ZedBoard Zynq™-7000 para programação ou avaliação de um dispositivo ZYNQ SOC XC7Z020-CLG484.
- Um simulador em “tempo real” para modelar e simular, com exatidão, o sistema de distribuição de energia elétrica. Este simulador foi apresentado no capítulo 2 e é constituído por três elementos:
 - ✓ Um computador (PC) de última geração com as características descritas nos itens 2.6.5 até 2.6.7;
 - ✓ A linguagem VHDL-AMS com a licença do SystemVision;
 - ✓ Uma interface para a importação de dados da rede de distribuição que se deseja modelar e simular (permitindo mudanças nos parâmetros de interesse em ambiente VHDL-AMS), conforme apresentado na figura 2.14.
- Uma interface gráfica do usuário (GUI) para permitir as operações de modelação e simulação do alimentador de distribuição de energia elétrica, em um ambiente configurável e amigável. Esta GUI foi implementada em plataforma SystemVision™, utilizando a linguagem VHDL-AMS, conforme apresentado em 4.2.

Neste contexto, a ilustração conceitual da arquitetura desenvolvida para a simulação em “tempo real” e controle (ASTR&C), utilizando as linguagens de descrição de hardware, VHDL e VHDL-AMS, de um alimentador de distribuição de energia elétrica, abrange a área azul da figura 47. Emprega-se nesta tese a denominação de simulação em “tempo real” e não literalmente em tempo real (sem aspas), fundamentalmente porque nosso simulador não interatua com o mundo real através de sinais de entrada e saída, em um tempo que coincide com o tempo de atuação do sistema, denominado tempo real. Ou seja, o desenvolvimento desta tese ainda não contempla as interfaces necessárias para que os sinais de controle possam comandar as ações dos controladores reais de redes. Este desenvolvimento está proposto como uma continuidade aos trabalhos desta tese, uma vez que envolve a necessidade de grandes investimentos para o desenvolvimento de um protótipo real de rede de distribuição inteligente em escala reduzida, com todos os seus elementos (fonte, linha, transformadores, reguladores, banco capacitores, cargas, relés de proteção, controladores PLC, dentre outros).

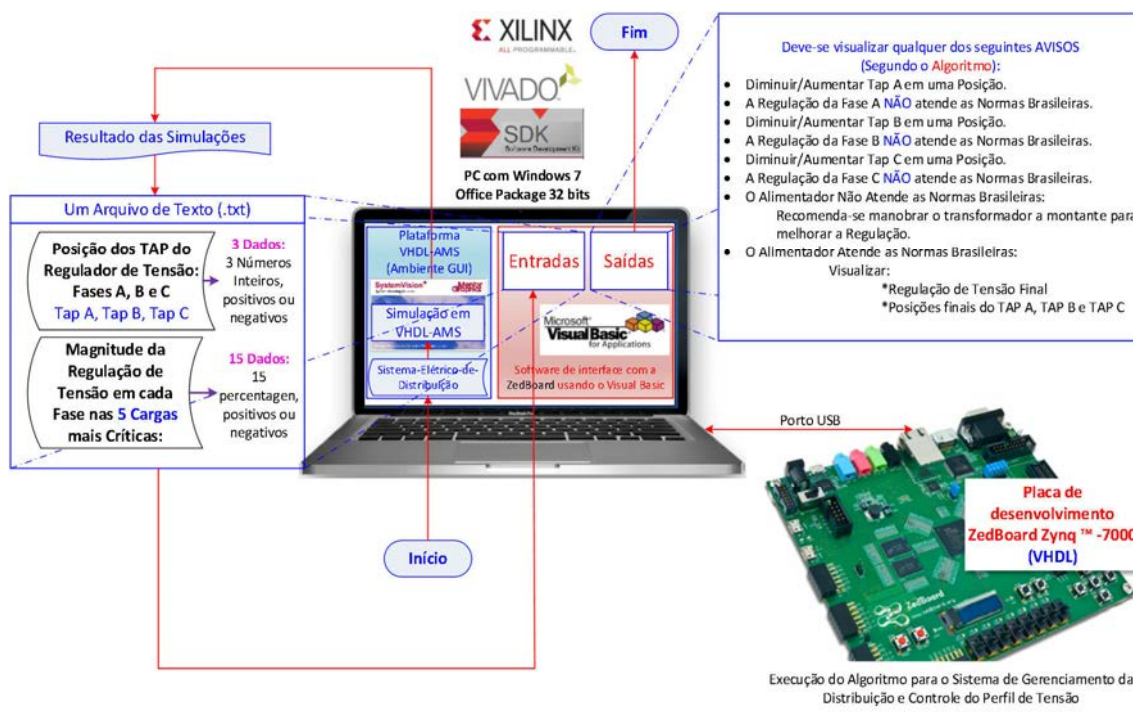
Figura 47- Ilustração conceitual da ASTR&C desenvolvida, para um alimentador elétrico de distribuição, utilizando as linguagens VHDL e VHDL-AMS.



Fonte: Próprio autor.

Baseados na figura 47 e segundo o capítulo 3, podemos estabelecer a representação esquemática do processo para o desenvolvimento desta nova arquitetura de simulação em “tempo real” e controle, de acordo com a figura 48. Este fluxograma enfoca o tratamento da informação segundo os recursos de informática utilizados no desenvolvimento do projeto.

Figura 48- Esquema informático da Arquitetura de Simulação em “Tempo Real” e Controle desenvolvida.



Fonte: Próprio autor.

Para o desenvolvimento completo da Arquitetura de Simulação em “Tempo Real” e Controle (ASTR&C) para um alimentador de distribuição de energia elétrica, conforme figura 47, é necessária a implementação da interface gráfica do usuário em plataforma SystemVision™, a qual é apresentada no item 4.2 a seguir.

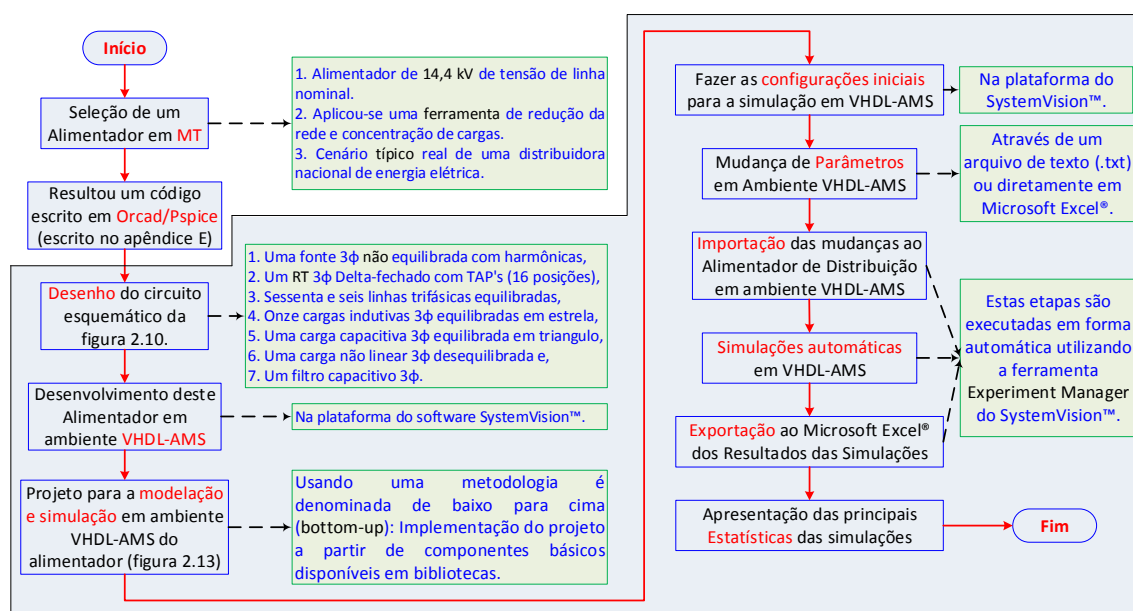
4.2 INTERFACE GRÁFICA DO USUÁRIO EM PLATAFORMA SYSTEMVISION™

Em informática, uma interface gráfica do usuário (ou utilizador), abreviada com o acrônimo GUI, do inglês **Graphical User Interface**, é um tipo de interface que permite a interação do usuário com dispositivos digitais (ou software) através de elementos gráficos como ícones e outros indicadores visuais.

Neste trabalho de tese, para a simulação em “tempo real” de um alimentador de distribuição de energia elétrica, é implementada uma interface gráfica do usuário (GUI) em plataforma SystemVision™, utilizando a linguagem VHDL-AMS. Esta GUI é realizada com o suporte e apoio de todas as ferramentas do software SystemVision™ (Versão 5.10.3), fornecido pela empresa **Mentor Graphics**.

Baseado no estudo do capítulo 2, concernente à utilização da linguagem VHDL-AMS, os estágios da implementação desta GUI encontram-se na figura 49, especificamente na área azul.

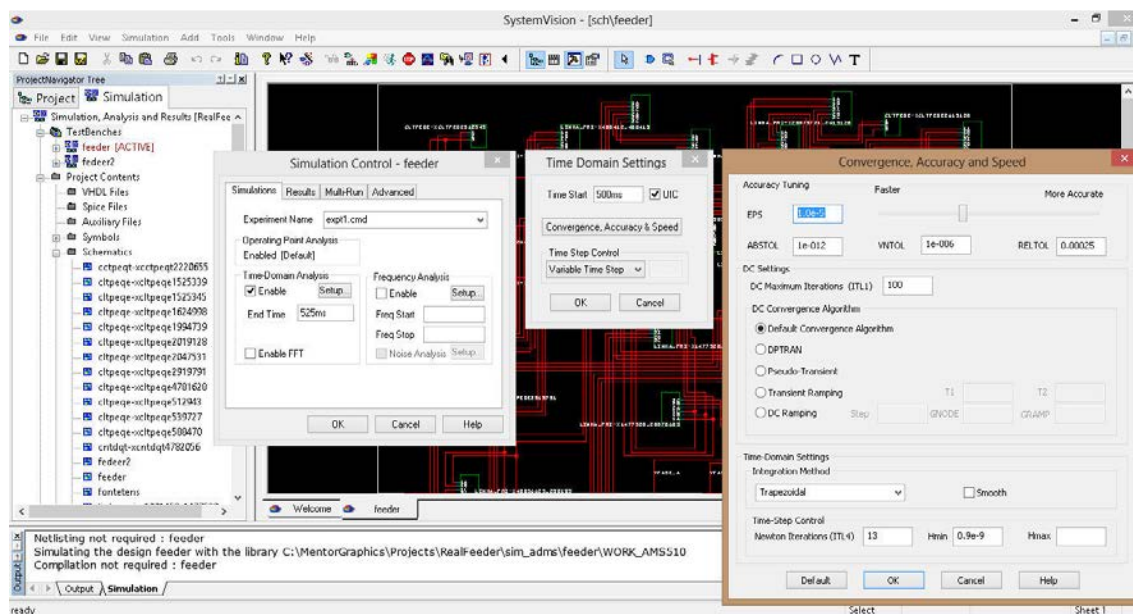
Figura 49- Estágios do projeto com a utilização da interface gráfica do usuário (GUI) em plataforma SystemVision™ (empregando a ferramenta “**Experiment Manager**”).



Fonte: Próprio autor.

Segundo o fluxograma da figura 49, após o projeto para a modelação e simulação em ambiente VHDL-AMS do alimentador escolhido para análise, devem-se fazer as configurações iniciais para a execução dos diferentes cenários de simulação. Estas configurações iniciais para as simulações em VHDL-AMS são elaboradas utilizando a ferramenta SystemVision™, segundo a figura 50.

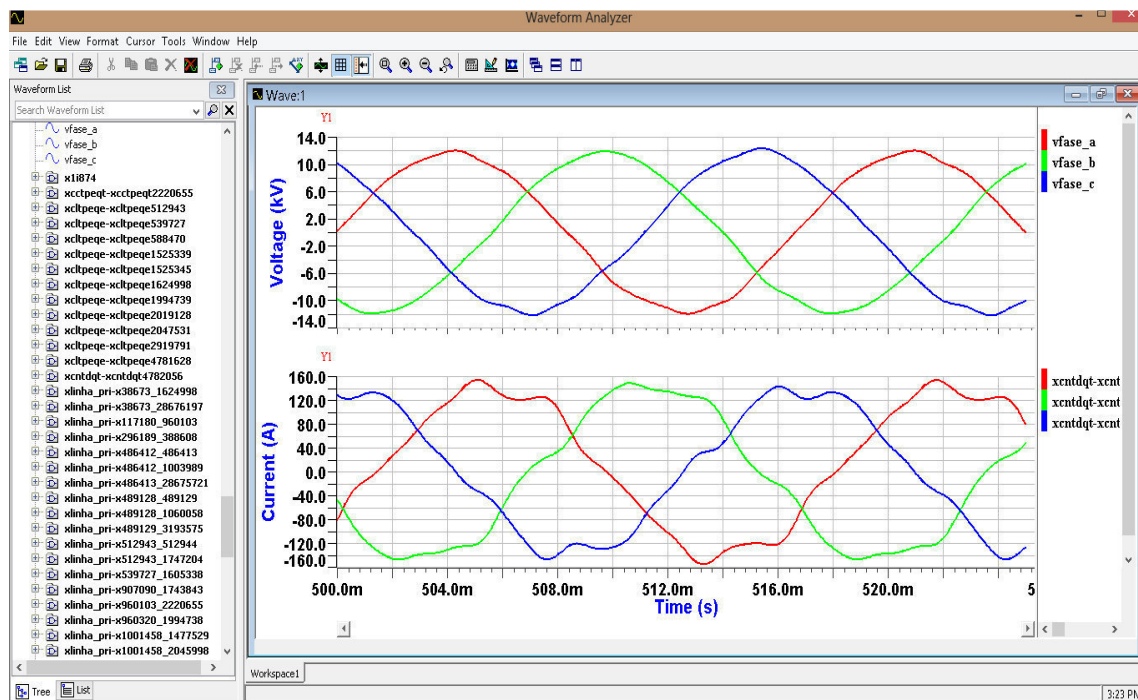
Figura 50- Principais configurações iniciais para a simulação do projeto do alimentador utilizando a ferramenta SystemVision™.



Fonte: Próprio autor utilizando a ferramenta SystemVision.

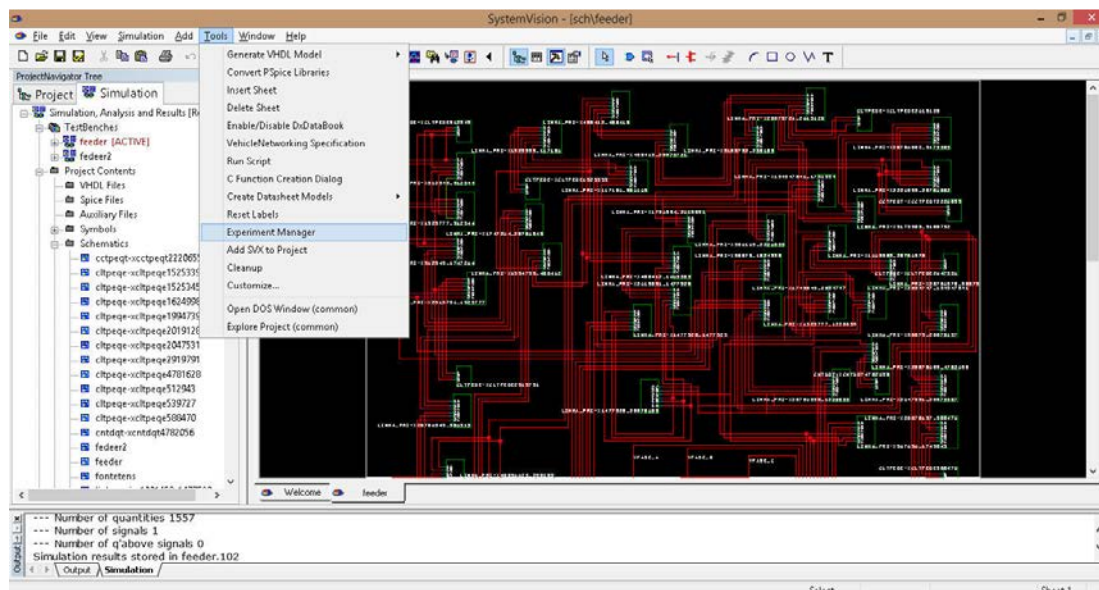
Após as configurações iniciais para os ajustes da simulação em ambiente VHDL-AMS, como usuários da GUI em plataforma SystemVision™, temos duas opções: 1) Fazer uma única simulação do sistema, configurando manualmente os parâmetros do alimentador elétrico. Esta opção reproduz um único conjunto de resultados derivados das simulações, que são mostrados (e podem ser representados graficamente) no **“Waveform Analyzer”** na plataforma SystemVision™. Estes resultados podem ser observados na figura 51; 2) Utilizar a ferramenta **“Experiment Manager”** na GUI, em plataforma SystemVision™, para fazer as últimas cinco etapas do fluxograma da figura 49 (depois de fazer as configurações iniciais). Esta última opção reproduz um conjunto de múltiplos resultados, derivados das simulações automáticas, que, além de ser mostradas (e representadas graficamente) no **“Waveform Analyzer”** na plataforma SystemVision™, podem ser armazenadas em uma folha eletrônica de cálculo do Microsoft Excel. Na figura 52 pode-se visualizar o início da utilização da ferramenta **“Experiment Manager”**.

Figura 51- Resultados das simulações em VHDL-AMS (tensões na subestação e correntes na carga não-linear trifásica desequilibrada conectada em triângulo), fazendo uma única simulação do sistema (no gráfico apresentam-se dois resultados parciais, podendo-se observar, na prática, qualquer resultado da simulação).



Fonte: Próprio autor utilizando a ferramenta SystemVision.

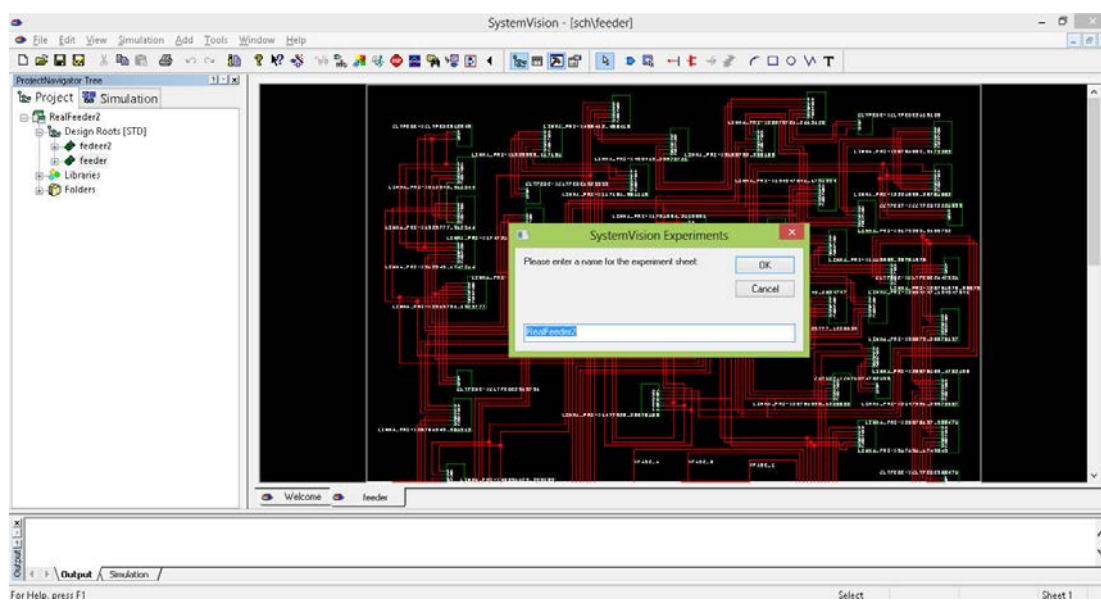
Figura 52- Utilização da ferramenta “*Experiment Manager*” na GUI em plataforma SystemVision™, utilizando a linguagem VHDL-AMS.



Fonte: Próprio autor utilizando a ferramenta SystemVision.

Após o início da utilização da ferramenta “**Experiment Manager**”, segundo a figura 53, deve-se especificar o nome da folha eletrônica de cálculo do Microsoft Excel, na qual estará armazenada a informação concernente aos múltiplos resultados das simulações automáticas. Logo, a folha eletrônica é armazenada na pasta do projeto descrito em VHDL-AMS. Estas simulações são feitas pela interface para a importação de dados da rede, de acordo com a figura 15.

Figura 53- Especificação da folha eletrônica de cálculo derivada da utilização da ferramenta “**Experiment Manager**”.



Fonte: Próprio autor utilizando a ferramenta SystemVision.

Após a especificação do nome da folha eletrônica de cálculo do Microsoft Excel, conforme figura 54, é possível se realizar três ações: 1) Estabelecer os parâmetros do alimentador elétrico (descrito em VHDL-AMS) que requerem mudanças em suas grandezas, para, baseado nisto, fazer as correspondentes simulações automáticas. Neste sentido, podemos escolher qualquer conjunto de parâmetros do alimentador elétrico, segundo os quais desejamos mudar as suas grandezas, e assim, observar os resultados destas mudanças nas simulações feitas em VHDL-AMS; 2) Estabelecer as medições do alimentador elétrico, que desejam-se armazenar na folha eletrônica de cálculo, ou seja, aqueles resultados das simulações automáticas que desejam-se extrair do ambiente VHDL-AMS, para que fiquem armazenados na folha eletrônica; 3) Iniciar as simulações automáticas, segundo o número de vezes que especificamos para as mudanças nos parâmetros do alimentador elétrico. Por exemplo, na figura

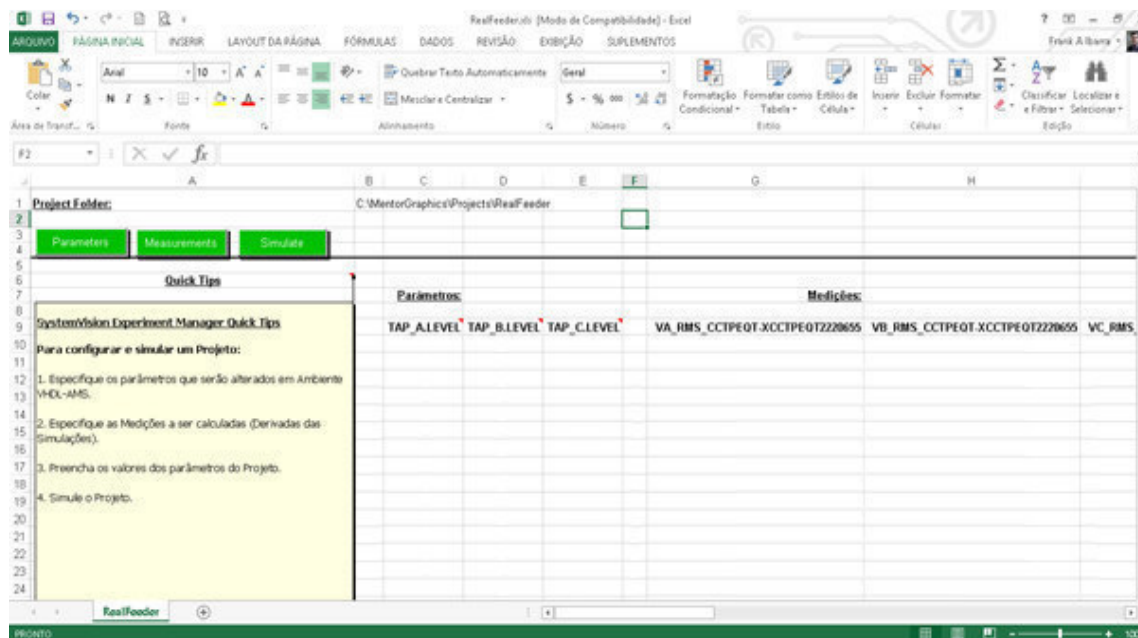
56 foram especificadas 17 vezes, relacionadas às alterações dos parâmetros das posições do TAP do regulador de tensão no alimentador elétrico (desde -8 até 8, passando pelo zero).

Nas figuras 55 e 56 encontram-se um exemplo da especificação dos parâmetros e da especificação das medições, respetivamente.

Depois de ter especificado os parâmetros e as medições do alimentador, em ambiente VHDL-AMS, podemos iniciar o processo das simulações sucessivas (automáticas), pressionando o ícone “**Simulate**”, na folha eletrônica de cálculo. Embora as especificações dos parâmetros e das medições do alimentador são feitas na folha eletrônica, as simulações automáticas são desenvolvidas no ambiente SystemVision (VHDL-AMS).

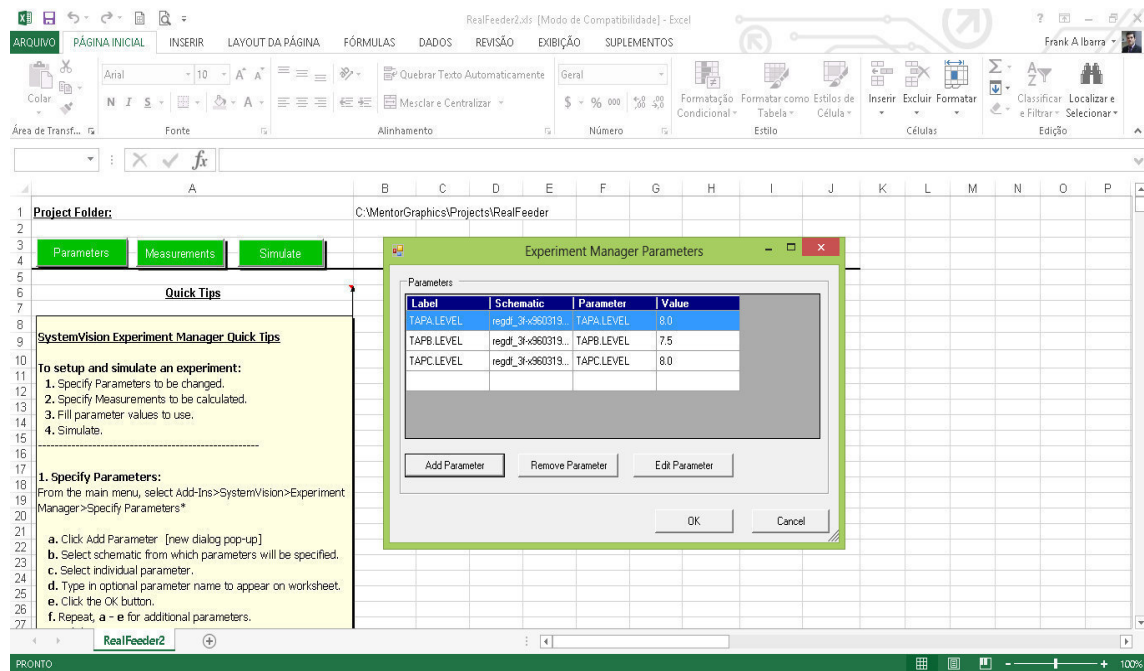
Segundo o exemplo considerado, na figura 56 foram especificadas 17 simulações automáticas e na figura 57 podemos observar o andamento destas 17 simulações. Na medida em que são realizadas as simulações sucessivas no “**Waveform Analyzer**”, na plataforma SystemVision™, conforme figura 58, são armazenados também, sucessivamente, os resultados destas simulações automáticas que desejam-se extrair do ambiente VHDL-AMS, de acordo com a figura 57 (segundo o exemplo estão sendo armazenados os resultados de três simulações).

Figura 54- Folha eletrônica derivada da utilização da ferramenta “**Experiment Manager**”.



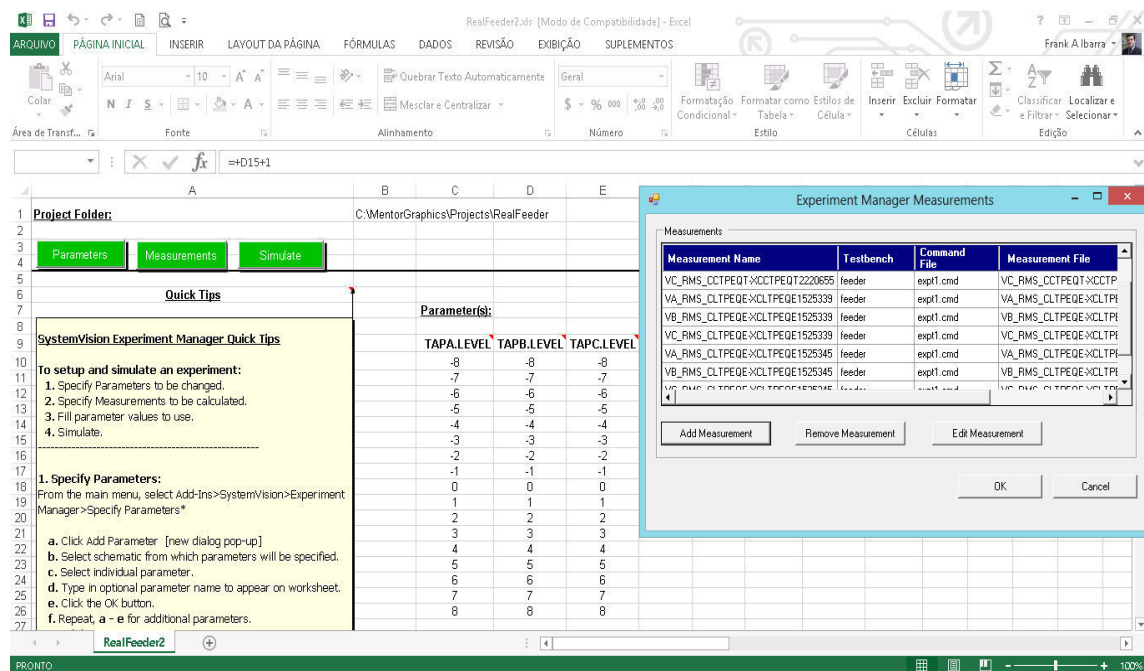
Fonte: Próprio autor utilizando a ferramenta SystemVision.

Figura 55- Especificação dos parâmetros (grandezas que desejam-se alterar em ambiente VHDL-AMS).



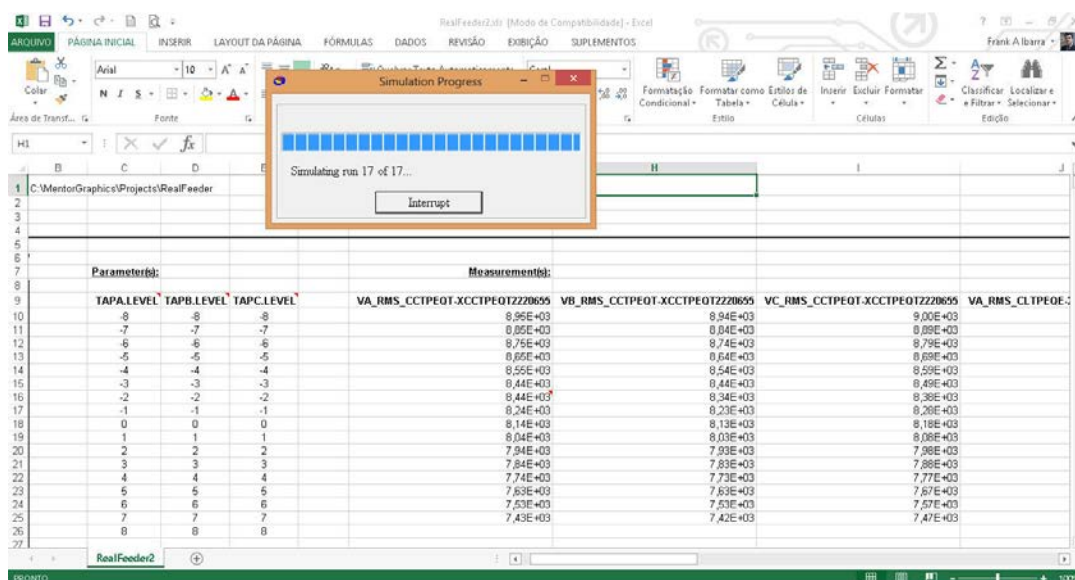
Fonte: Próprio autor utilizando a ferramenta SystemVision.

Figura 56- Especificação das medições (resultados das Simulações automáticas do alimentador em VHDL-AMS que desejam-se armazenar na folha eletrônica de cálculo).



Fonte: Próprio autor utilizando a ferramenta SystemVision.

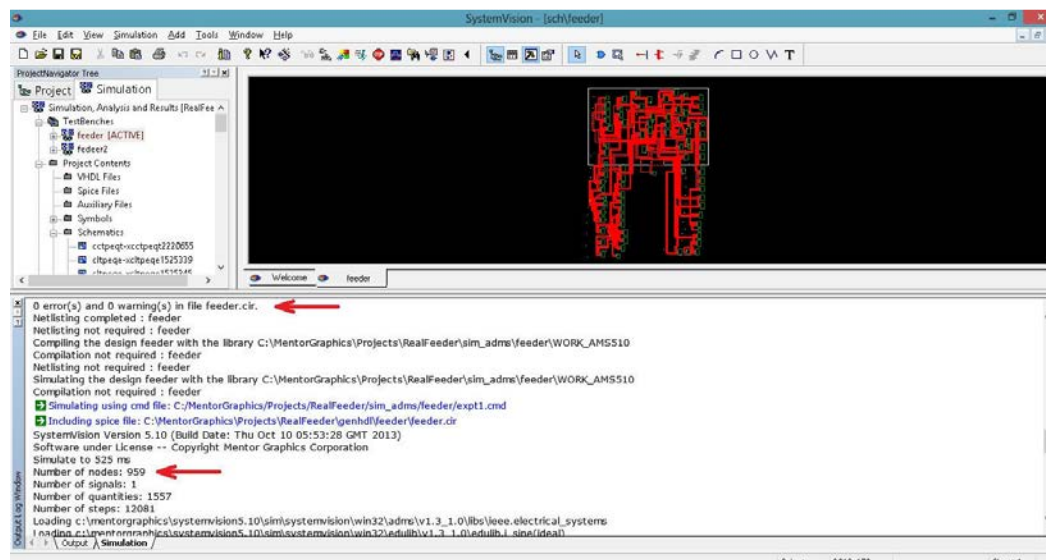
Figura 59- Resultados finais das simulações automáticas em VHDL-AMS (especificamente na simulação número 17 de 17).



Fonte: Próprio autor utilizando a ferramenta SystemVision.

Finalmente, após o armazenamento na folha eletrônica de cálculo dos resultados das simulações automáticas que desejam-se extrair do ambiente VHDL-AMS, segundo a figura 60, são apresentadas as principais estatísticas das simulações em VHDL-AMS, utilizando a GUI em plataforma SystemVision™.

Figura 60- Apresentação das principais estatísticas das simulações em VHDL-AMS, utilizando a GUI em plataforma SystemVision™.



Fonte: Próprio autor utilizando a ferramenta SystemVision.

4.3 CONCEITO INTEGRADO DA ARQUITETURA DE SIMULAÇÃO EM “TEMPO REAL” E CONTROLE DE UM ALIMENTADOR DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA (RESULTADOS EXPERIMENTAIS DO DESENVOLVIMENTO)

De acordo com a plataforma SystemVision™, após a implementação da linguagem VHDL-AMS, são obtidos todos os resultados das simulações do alimentador de distribuição. Além disso, são filtrados os resultados necessários para o desenvolvimento do sistema de gerenciamento e controle (DMS&C) do alimentador de distribuição, através da criação e alocação, no PC, de um arquivo de texto, de acordo com a utilização do SystemVision™ para a implementação da linguagem VHDL-AMS, conforme apresentado na figura 9.

Estes resultados necessários para o DMS&C, derivados das simulações em ambiente VHDL-AMS, são enviados para a placa ZedBoard Zynq™-7000, que contém o dispositivo FPGA Artix-7, pelo software de interface entre o PC e a placa ZedBoard (de acordo com o item 3.6.2).

Estes resultados, indispensáveis para o DMS&C, são obtidos e representados em formato de texto, conforme a figura 61, ponto de partida da implementação do DMS&C.

Figura 61- Arquivo de texto que constitui os dados de entrada no dispositivo FPGA.

Posição dos TAP do Regulador de Tensão (Fases A, B e C):			
Tap A	Tap B	Tap C	
-1	-7	8	

Regulação de Tensão em cada Fase nas Seguintes Cargas mais Críticas:			
CARGA CRÍTICA	FASE A	FASE B	FASE C
CLTPEQE-XCLTPEQE539727	-6,0%	5,0%	7,0%
CLTPEQE-XCLTPEQE2047531	-7,0%	25,0%	14,0%
CLTPEQE-XCLTPEQE1525339	8,0%	12,0%	2,0%
CLTPEQE-XCLTPEQE1525345	9,0%	8,0%	8,0%
CLTPEQE-XCLTPEQE4781628	15,0%	3,0%	5,0%

Fonte: Próprio autor utilizando o Notepad.

Uma vez alocado no PC, o arquivo de texto com os resultados das simulações em VHDL-AMS (que constitui os dados de entrada no dispositivo FPGA), este é utilizado pelo software de interface entre o PC e a placa ZedBoard, para ser inserido no dispositivo FPGA Artix-7 e, assim, processar o algoritmo do sistema de gerenciamento e controle (DMS&C) para o alimentador de distribuição de 14,4 kV de tensão de linha. Nas três telas da figura 62 está

representada a sequência da implementação, inicialização e funcionamento do software de interface entre o PC e a placa ZedBoard, assim: Na primeira tela deve-se escolher o endereço no qual encontra-se o arquivo de texto com os resultados das simulações em VHDL-AMS. Com o endereço selecionado, na segunda tela o DMS&C está pronto para ser utilizado, para o qual, a placa ZedBoard deve estar energizada e conectada ao PC através da porta USB. Depois de apertar o botão “Enviar”, o qual inicia o DMS&C enviando os dados para o dispositivo FPGA, os resultados da execução do sistema de gerenciamento da distribuição e controle (DMS&C) são mostrados na terceira tela. Na figura 63 pode-se verificar a montagem do DMS&C, utilizando o PC e a placa ZedBoard Zynq™-7000.

Figura 62- Escolha do arquivo de texto com os resultados das simulações em VHDL-AMS.



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

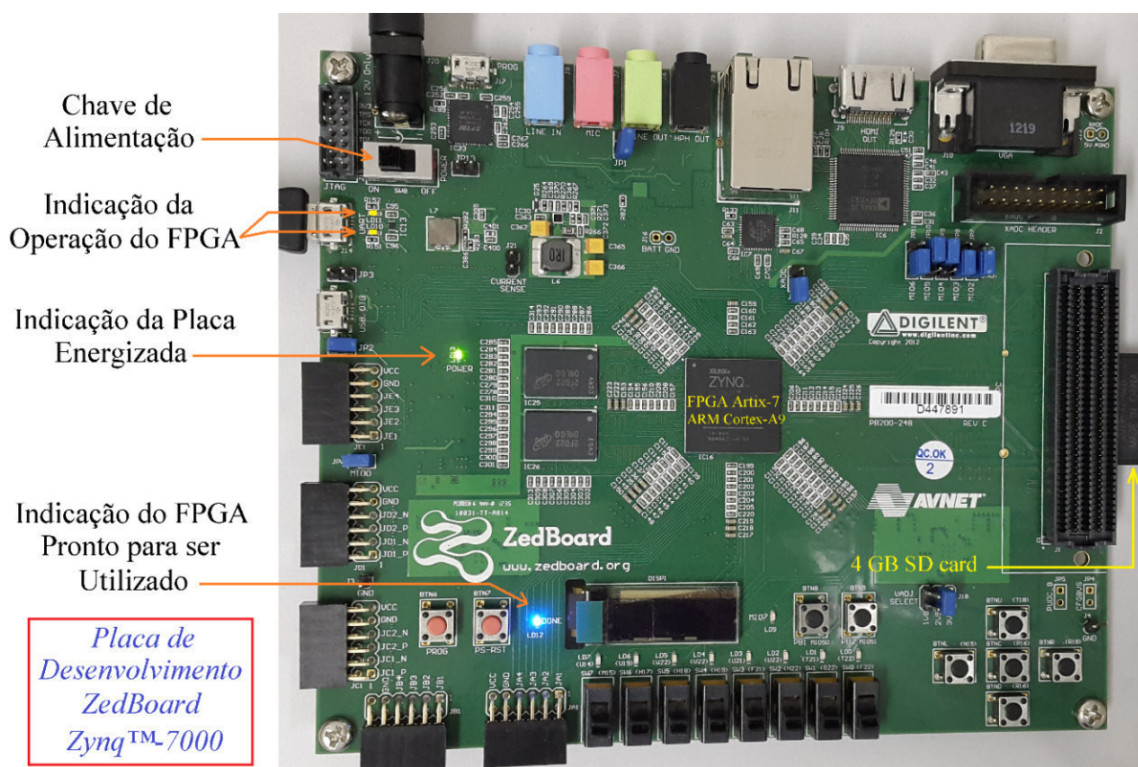
Figura 63- Montagem e funcionamento do sistema de gerenciamento da distribuição e controle (DMS&C), utilizando a placa ZedBoard Zynq™-7000 e o PC.



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Na figura 64 podem-se visualizar os indícios da implementação e funcionamento do DMS&C, considerando os diferentes indicadores de operação próprios da placa ZedBoard Zynq™-7000, caracterizados assim: 1) A chave de alimentação na posição esquerda significa que, se a placa está conectada na tomada pela porta de energização, a placa esta energizada; 2) Os dois LEDs amarelos acesos piscando, indicam o funcionamento do dispositivo ZYNQ SoC XC7Z020-CLG484, ou seja, o funcionamento do FPGA Artix-7 e/ou o processador ARM Cortex-A9 (etiquetado na placa como ZYNQ); 3) O LED verde aceso em forma fixa significa que a placa esta energizada corretamente; e 4) O LED azul aceso em forma fixa significa que o dispositivo ZYNQ SoC XC7Z020-CLG484 está pronto para ser utilizado.

Figura 64- Placa de desenvolvimento ZedBoard Zynq™-7000 em funcionamento.



Fonte: Próprio autor utilizando o funcionamento do DMS&C.

4.1.1. Resultados experimentais do desenvolvimento do sistema de gerenciamento e controle (DMS&C)

Segundo o DMS&C implementado no dispositivo FPGA Artix-7, para que o alimentador de distribuição atenda às normas brasileiras no quesito de regulação de tensão, deve-se atender que TODAS as Magnitudes das Regulações de tensão (MRs) das cinco cargas mais críticas, consideradas para o estudo de caso aqui admitido nesta tese, para TODAS as três

fases A, B e C, devem estar na faixa: $-5\% \leq MR\% \leq 7\%$. Neste caso, o DMS&C deve indicar que o alimentador atende às normas brasileiras e deve apresentar visualmente os resultados das MRs e as posições dos TAPs A, B e C indicadas para o controle do Regulador de Tensão (RT).

Assim, se, em qualquer das três fases, ALGUMA das Magnitudes das Regulações de tensão (MRs) das cinco cargas mais críticas NÃO estão na faixa padronizada, pode-se apresentar qualquer das seguintes duas situações, para as fases A, B ou C:

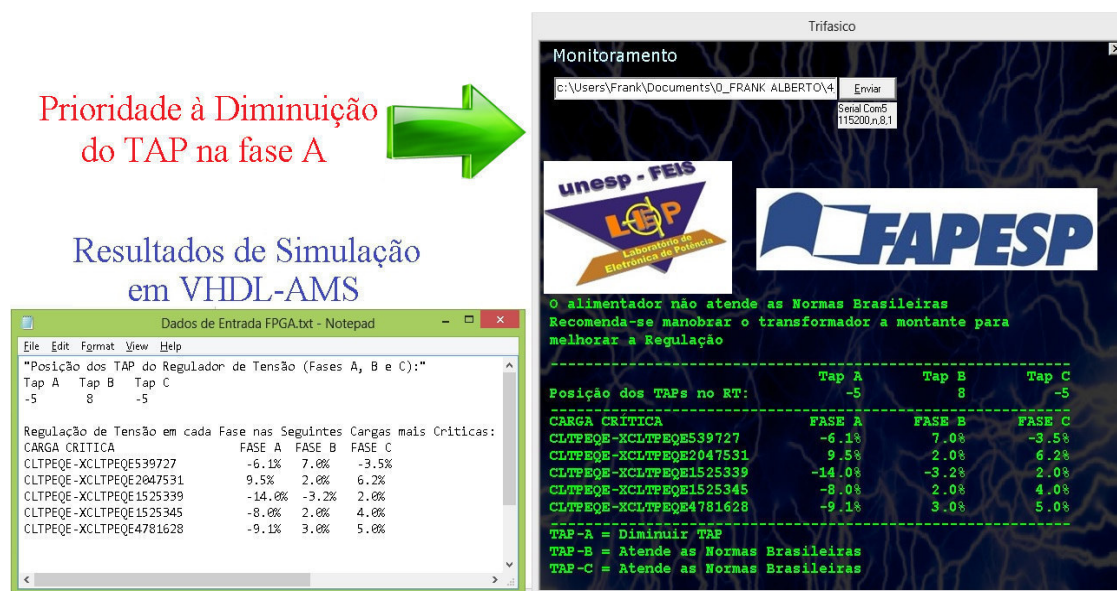
- No caso que, em qualquer das três fases, ALGUMA das Magnitudes das Regulações de tensão (MRs) das cinco cargas mais críticas SEJAM maiores a 7%, o DMS&C verificará, simultaneamente para cada fase, se o TAP no RT está na posição “- 8”. Se o TAP no RT não está na posição “- 8”, o DMS&C indicará a redução de TAP na fase (ou nas fases) correspondente (s), em uma posição, para assim, aumentar a tensão à jusante do RT e diminuir a magnitude da regulação de tensão na fase correspondente (com o intuito de que seja menor ou igual a 7%). Por outro lado, se o TAP no RT está na posição “- 8”, este não pode diminuir de posição e consequentemente, NÃO é possível diminuir a magnitude da regulação de tensão na fase correspondente. Em ambos os casos, se o TAP no RT não está na posição “- 8” ou se o TAP no RT está na posição “- 8”, a magnitude da regulação de tensão na fase (ou nas fases) correspondente (s) NÃO atende (m) às normas brasileiras e, portanto, o alimentador NÃO atende às normas brasileiras. Nestes cenários, o DMS&C, além de apresentar visualmente as MRs e as posições dos TAPs A, B e C possíveis indicadas para o Regulador de Tensão (RT), recomendará manobrar o transformador a montante para melhorar a regulação de tensão no alimentador de distribuição.
- No caso que, em qualquer das três fases, ALGUMA das Magnitudes das Regulações de tensão (MRs) das cinco cargas mais críticas SEJAM menores que - 5%, o DMS&C verificará, simultaneamente para cada fase, se o TAP no RT está na posição “8”. Se o TAP no RT não está na posição “8”, o DMS&C indicará o aumento da posição do TAP na fase (ou nas fases) correspondente (s), em uma posição, para assim, diminuir a tensão à jusante do RT e aumentar a magnitude da regulação de tensão na fase correspondente (com o intuito de que seja maior ou igual a - 5%). Por outro lado, se o TAP no RT está na posição “8”, este TAP não pode aumentar de posição e consequentemente, NÃO é possível aumentar a magnitude da regulação de tensão na fase correspondente. Em ambos os casos, se o TAP no RT está na posição “8” ou se o TAP no RT não está na posição “8”, a magnitude da regulação

de tensão na fase (ou nas fases) correspondente (s) NÃO atende (m) às normas brasileiras e, portanto, o alimentador NÃO atende às normas Brasileiras. Nestes cenários, o DMS&C, além de apresentar visualmente as MRs e as posições dos TAPs A, B e C indicadas para o Regulador de Tensão (RT), recomendará manobrar o transformador a montante para melhorar a regulação de tensão no alimentador de distribuição.

É importante mencionar que, devido às características radiais do alimentador escolhido como estudo de caso, conforme figura 11, o perfil de tensão deste alimentador, conforme figura 20 mostra que todas as cargas sofrem problema de queda de tensão em relação à tensão da subestação. No trabalho desenvolvido por (ATTIA, 2008) evidenciou-se que, em condições normais de operação, as cargas de um alimentador radial de distribuição em média tensão sempre vão apresentar uma queda de tensão maior, com respeito à tensão na subestação, conforme estas vão ficando mais distante da subestação de distribuição. Por isto, é necessária a operação conjunta de, pelo menos, um regulador de tensão nos alimentadores radiais de distribuição em média tensão (tensão eficaz entre fases superior a 1 kV e inferior a 69 kV). Com base nesta característica intrínseca dos alimentadores radiais de distribuição em média tensão, neste trabalho, se em uma mesma fase duas ou mais cargas, das 5 cargas críticas, apresentarem, ao mesmo tempo, valores de Magnitudes da Regulação de tensão (MRs) que estão nos limites opostos ao não atendimento das normas brasileiras, por exemplo, uma carga com $MR > 7\%$ e outra carga (ou outras) com $MR < -5\%$, o desenvolvimento do DMS&C dará prioridade à diminuição do TAP nessa fase e, conseqüentemente, o aumento da tensão na carga. Assim, se em uma mesma fase existisse uma carga crítica com a $MR > 7\%$ e o TAP no regulador de tensão, nessa fase, é diferente de “-8”, independentemente do valor da MR nas outras quatro cargas críticas restantes, o DMS&C solicitará diminuir o TAP na fase correspondente em uma posição e aumentar a tensão a jusante do regulador de tensão nessa fase da carga.

Embora seja muito pouco provável que, em condições operativas normais do alimentador, em uma mesma fase, duas ou mais cargas, das 5 cargas críticas, apresentem, ao mesmo tempo, valores de magnitudes da regulação de tensão que estão nos limites opostos ao não atendimento das normas brasileiras, conforme exemplo da figura 65, pode-se visualizar o funcionamento do DMS&C nesta condição, onde a segunda carga crítica da fase A apresenta um $MR > 7\%$, e as demais apresentam um $MR < -5\%$.

Figura 65- Execução do DMS&C para o caso no qual dá-se prioridade à diminuição do TAP na fase A, no regulador de tensão (RT), para aumentar a tensão a jusante do RT.

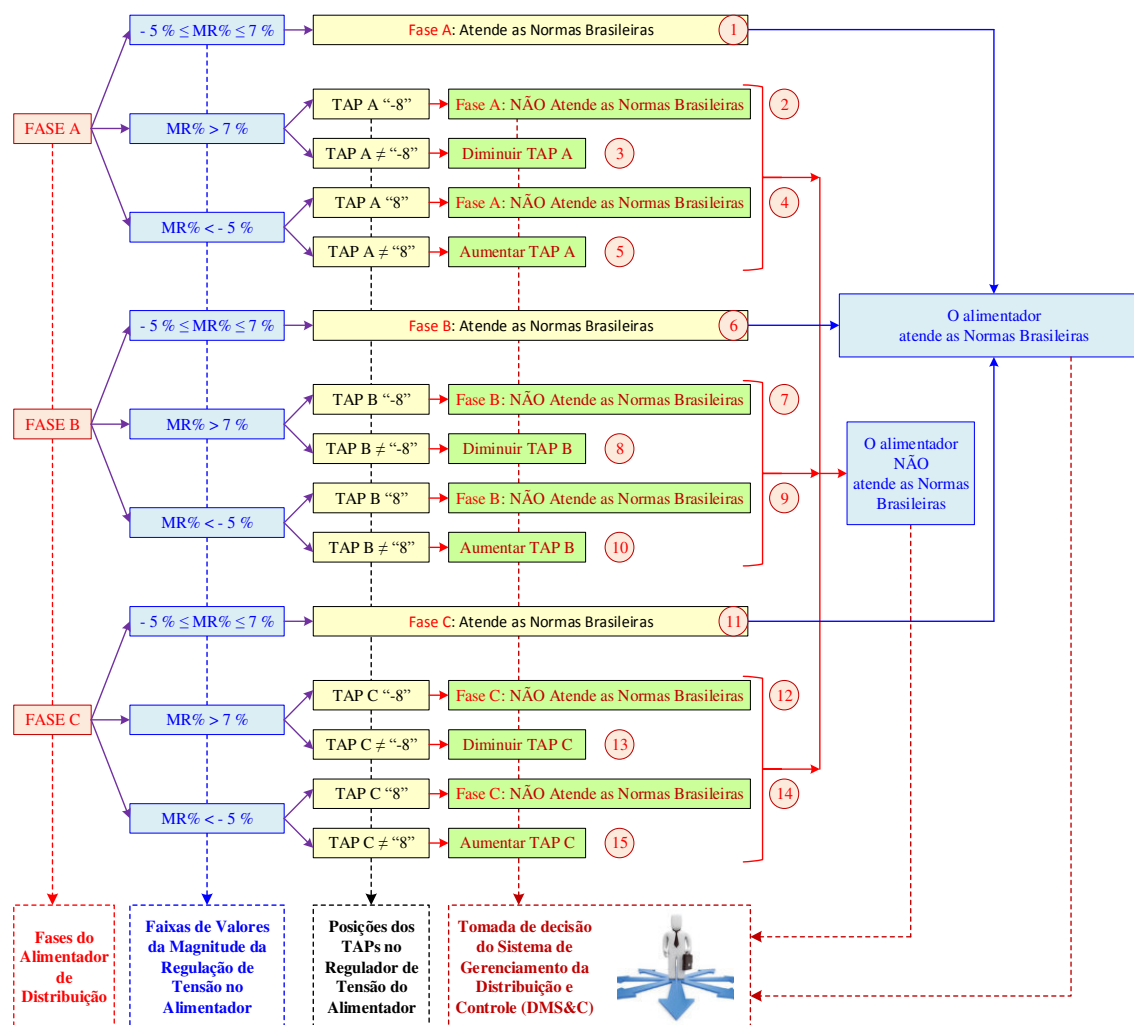


Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Por outro lado, devido ao paralelismo empregado pelo dispositivo FPGA, é possível realizar o desenvolvimento das operações lógicas, simultaneamente, para cada uma das três fases. Estas operações lógicas, estabelecidas no algoritmo da figura 3.2, foram determinadas concorrentemente (em paralelo).

Assim, é possível esquematizar as quinze (15) possibilidades exemplos de atuação que podem ser geradas pelo Sistema de Gerenciamento e Controle (DMS&C), segundo a figura 66, as quais estão indicadas por um número vermelho (de 1 a 15), contido dentro de um círculo vermelho.

Figura 66- Possibilidades que podem ser geradas pelo sistema de gerenciamento e controle (DMS&C).



Fonte: Próprio autor.

Segundo as 15 possibilidades exemplos de atuação do sistema de gerenciamento e controle (DMS&C), numeradas em vermelho, podem-se estabelecer os seguintes cenários de simulação do alimentador estudo de caso em ambiente VHDL-AMS, os quais são executados no DMS&C. Estes cenários de simulação, os quais foram concebidos com o intuito de testar na sua totalidade o DMS&C, encontram-se na tabela 21.

TABELA 21- CENÁRIOS DE SIMULAÇÃO EM AMBIENTE VHDL-AMS.

Cenário de Simulação	Possibilidade de Atuação FASE A	Possibilidade de Atuação FASE B	Possibilidade de Atuação FASE C
Primeiro	1	6	11
Segundo	2	6	11
Terceiro	3	6	11
Quarto	4	6	11
Quinto	5	6	11
Sexto	1	7	11
Sétimo	1	8	11
Oitavo	1	9	11
Nono	1	10	11
Décimo	1	6	12
Décimo Primeiro	1	6	13
Décimo Segundo	1	6	14
Décimo Terceiro	1	6	15

Fonte: Próprio autor

Como exemplo da atuação do DMS&C dentro da nova ASTR&C desenvolvida, é utilizada a carga com a pior regulação de tensão (CLTPEQE-XCLTPEQE539727), dentro do terceiro cenário de simulação, o qual apresenta a primeira manobra de diminuição de TAP, neste caso, na fase A do regulador de tensão.

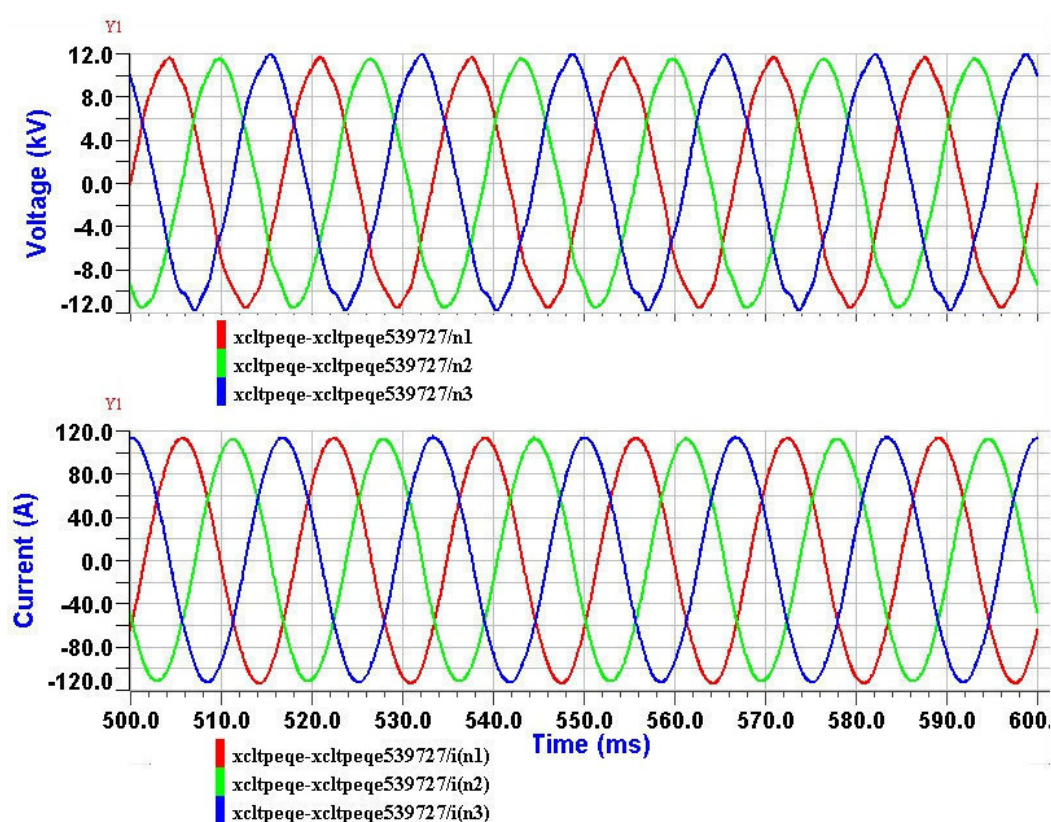
A figura 67 expõe as formas de onda de tensão e corrente da carga CLTPEQE-XCLTPEQE539727 dentro da operação normal do alimentador de distribuição, isto é, as posições dos TAP A, B e C no regulador de tensão estão situadas todas em “1”. Após a execução da simulação do alimentador em VHDL-AMS com essas posições dos TAPs, o valor da magnitude da regulação de tensão resultou igual a 3,3 % para as três fases, atendendo assim as normas Brasileiras.

Considerando as condições do terceiro cenário de simulação, a figura 68 apresenta as formas de onda de tensão e corrente da carga CLTPEQE-XCLTPEQE539727 com uma condição anômala do alimentador, isto é, as posições dos TAP A, B e C no regulador de tensão foram situadas em “5”, “1” e “1”, respectivamente. Depois de executar a simulação do alimentador em VHDL-AMS com essas posições dos TAPs, os valores das magnitudes da regulação de tensão (RT) da carga CLTPEQE-XCLTPEQE539727 resultaram iguais a 8,2 %, 3,3 % e 3,3 % para as fases A, B e C, respectivamente, NÃO atendendo assim as normas Brasileiras, por causa do valor da magnitude da RT na fase A.

Depois da atuação do DMS&C neste terceiro cenário de simulação, o sistema de gerenciamento do alimentador recomendou diminuir uma posição do TAP na fase A no RT. A

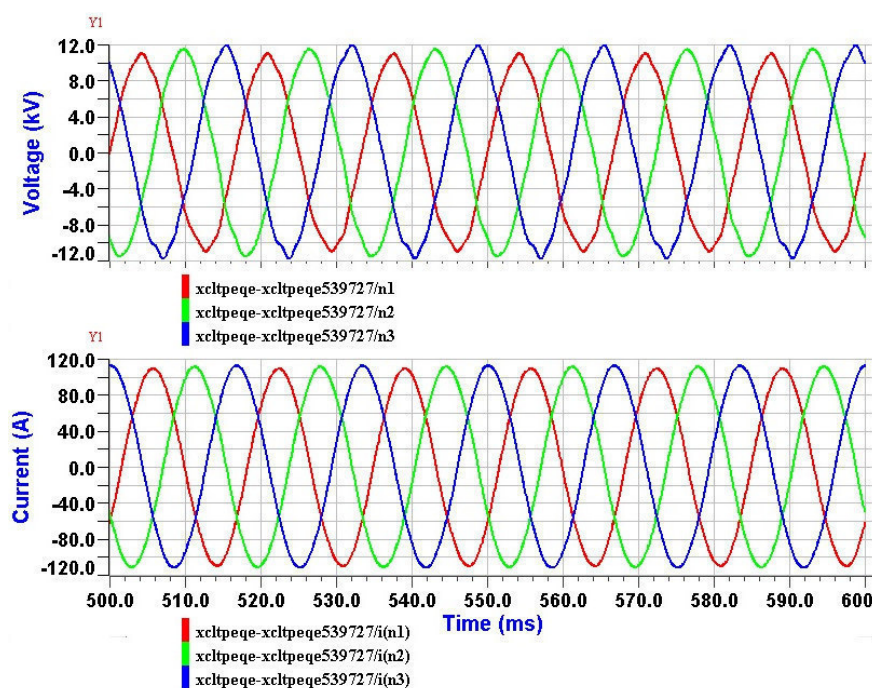
figura 69 apresenta as formas de onda de tensão e corrente da carga CLTPEQE-XCLTPEQE539727 com a recomendação do DMS&C realizada, isto é, as posições dos TAP A, B e C no regulador de tensão foram situadas em “4”, “1” e “1”, respectivamente. Depois de executar a simulação do alimentador em VHDL-AMS com essas posições dos TAPs, os valores das magnitudes da regulação de tensão (RT) da carga CLTPEQE-XCLTPEQE539727 resultaram iguais a 6,5 %, 3,3 % e 3,3 % para as fases A, B e C, respectivamente, atendendo assim as normas Brasileiras.

Figura 67- Simulações em ambiente VHDL-AMS para a carga CLTPEQE-XCLTPEQE539727 no terceiro cenário de simulação com operação normal do alimentador.



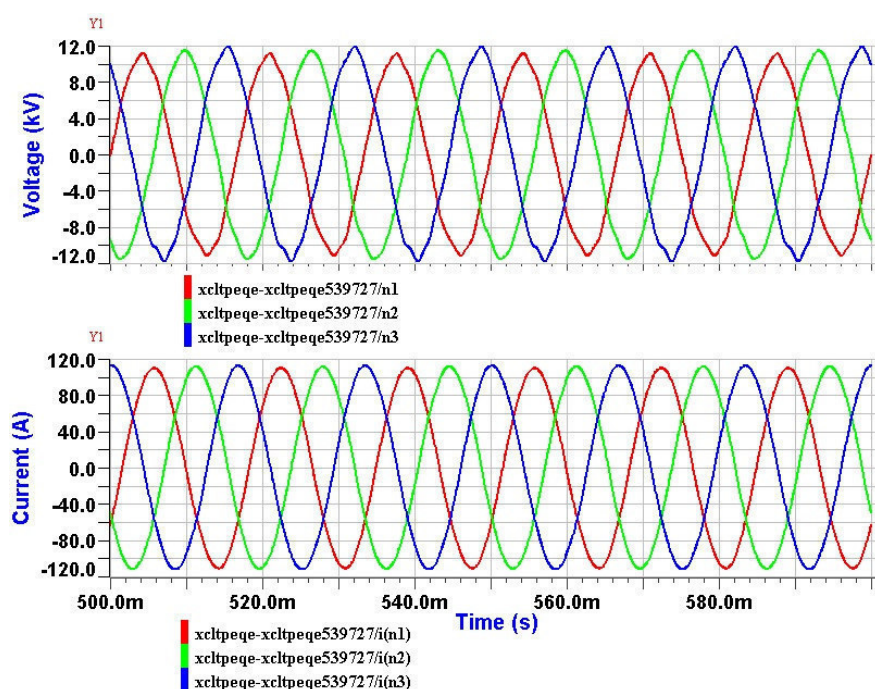
Fonte: Próprio autor.

Figura 68- Simulações em ambiente VHDL-AMS para a carga CLTPEQE-XCLTPEQE539727 no terceiro cenário de simulação com uma condição anômala do alimentador.



Fonte: Próprio autor.

Figura 69- Simulações em ambiente VHDL-AMS para a carga CLTPEQE-XCLTPEQE539727 para o terceiro cenário de simulação, depois de executar a recomendação do DMS&C.



Fonte: Próprio autor.

Empregando o método descrito na interface gráfica do usuário em plataforma SystemVision™, foram rodados os 13 cenários anteriores de simulação em VHDL-AMS, os quais apresentaram vários conjuntos de resultados de simulação, na tela nomeada “Dados de Entrada FPGA.txt”, de acordo com as figuras 70 até 82. Também, de acordo com estas figuras, pode-se visualizar a implementação e funcionamento do DMS&C, para o alimentador de distribuição estudo de caso, na tela nomeada “Trifásico”. Esta tela representa o software de interface entre o PC e a Placa ZedBoard e mostra os resultados do DMS&C, derivados dos diferentes cenários de simulação desenvolvidos em ambiente VHDL-AMS.

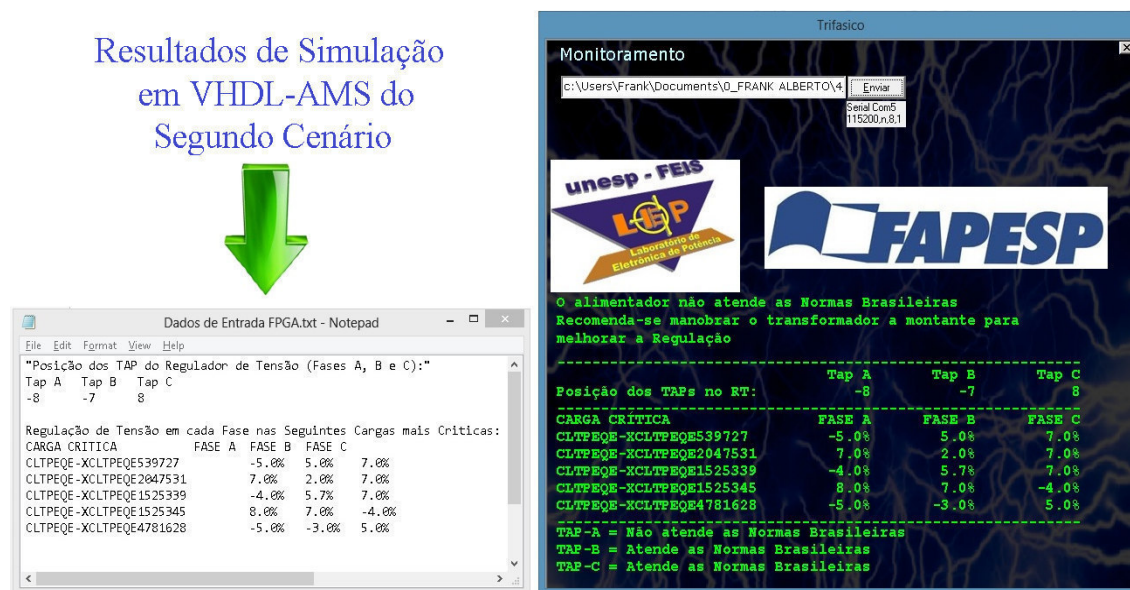
Na figura 83 mostra-se o resultado da execução do DMS&C, quando a placa está desligada da alimentação da eletricidade, ou, desligada da porta USB do PC.

Figura 70- Execução do DMS&C para o primeiro cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, nas três fases, cumprem - $5\% \leq MR\% \leq 7\%$. Portanto, todas as fases atendem às normas brasileiras e consequentemente, o alimentador atende às normas nacionais.



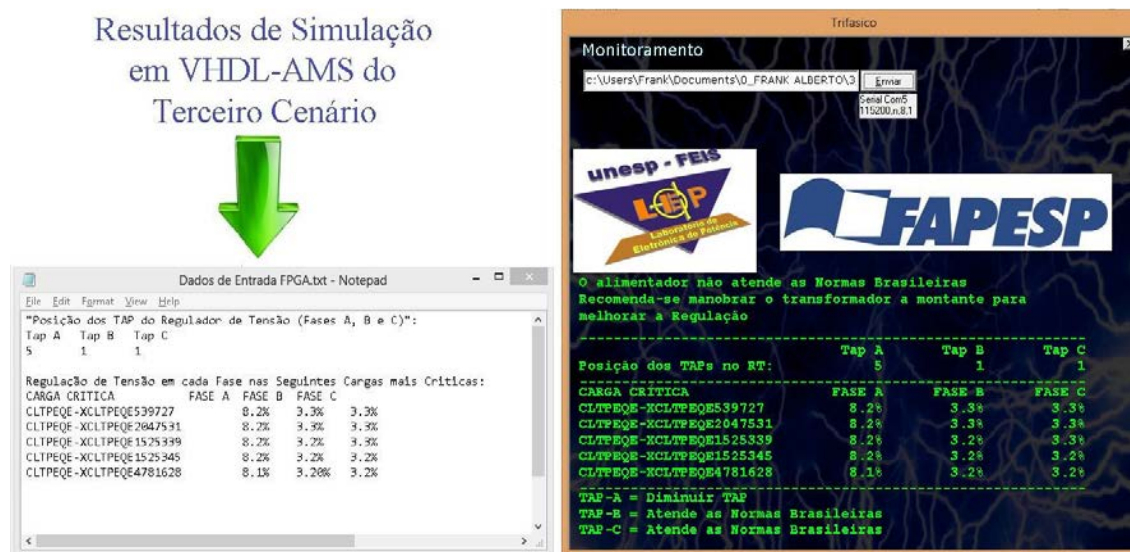
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 71- Execução do DMS&C para o segundo cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase B e C, cumprem - $5\% \leq MR\% \leq 7\%$. A fase A apresenta $MR > 7\%$ na quarta carga crítica e tem um TAP “-8”. Portanto, a fase A não atende às normas brasileiras.



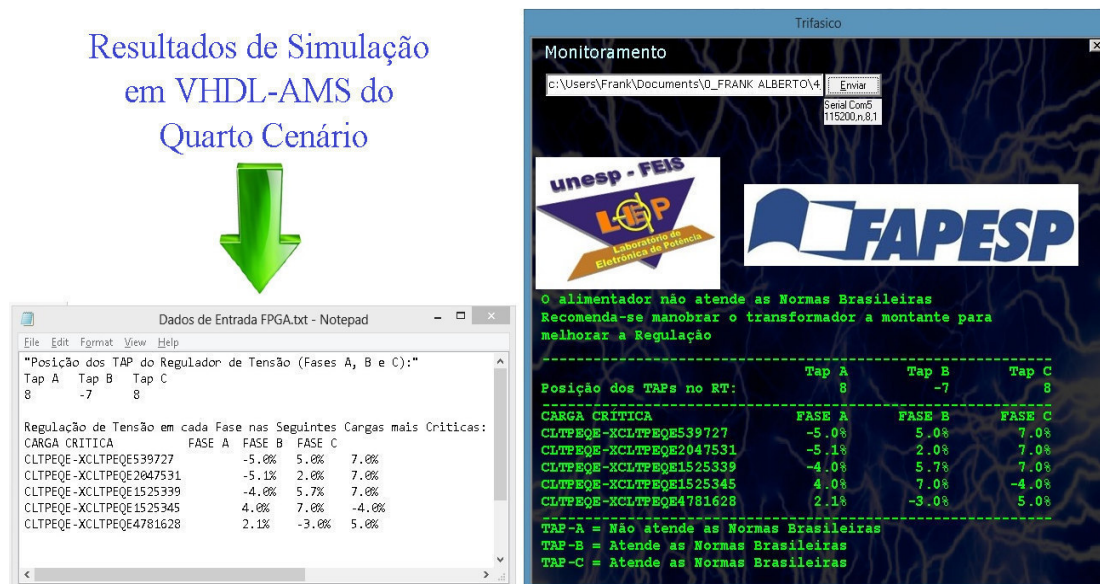
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 72- Execução do DMS&C para o terceiro cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase B e C, cumprem - $5\% \leq MR\% \leq 7\%$. A fase A apresenta $MR > 7\%$ nas cargas críticas e tem um TAP $\neq$ “-8”. Portanto, na fase A, no regulador de tensão, pede-se diminuir o TAP em uma posição.



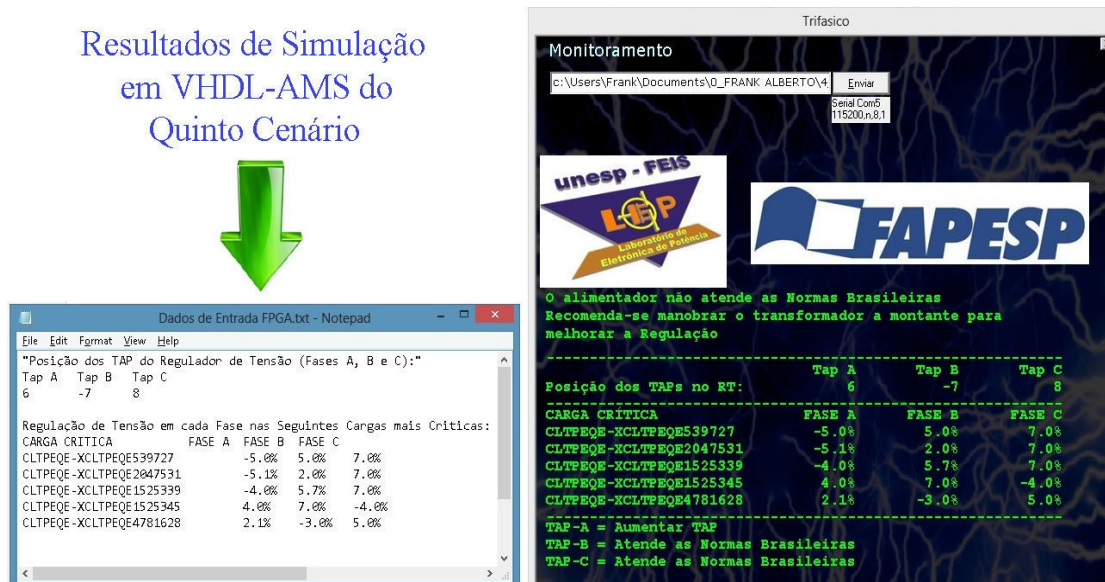
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 73- Execução do DMS&C para o quarto cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase B e C, cumprem $-5\% \leq MR\% \leq 7\%$. A fase A apresenta $MR < -5\%$ na segunda carga crítica e tem um TAP “8”. Portanto, a fase A não atende às normas brasileiras.



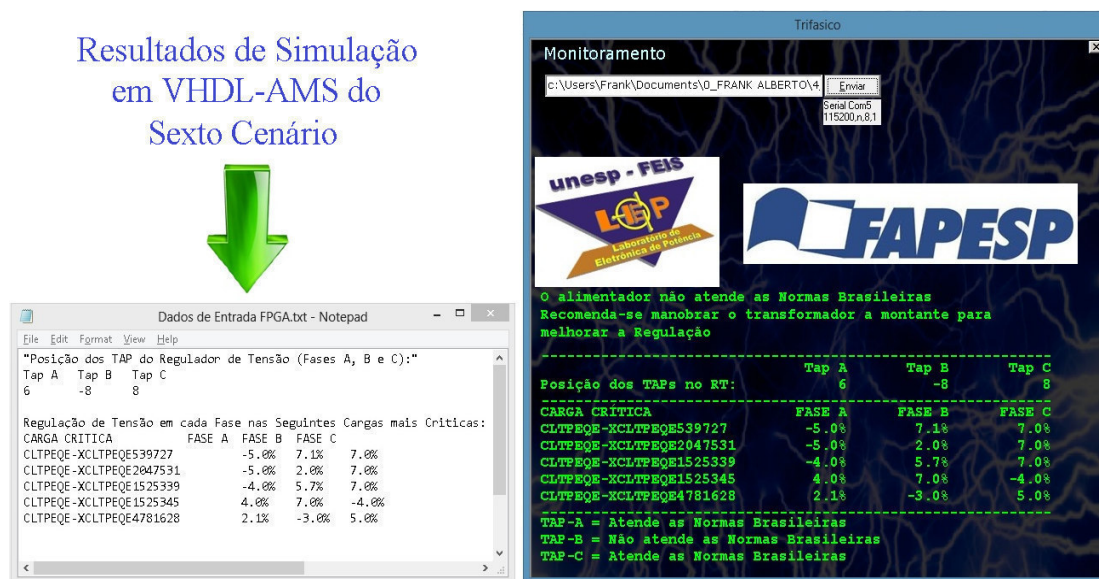
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 74- Execução do DMS&C para o quinto cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase B e C, cumprem $-5\% \leq MR\% \leq 7\%$. A fase A apresenta $MR < -5\%$ na segunda carga crítica e tem um TAP $\neq$ “8”. Portanto, na fase A, no regulador de tensão, pede-se aumentar o TAP em uma posição.



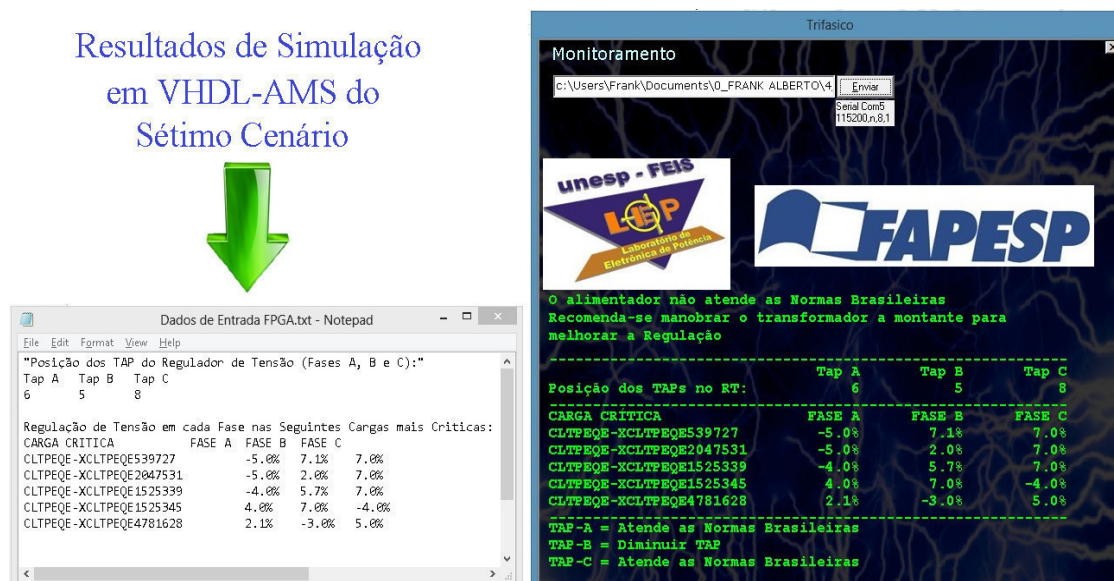
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 75- Execução do DMS&C para o sexto cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e C, cumprem - 5 % $\leq$ MR% $\leq$ 7 %. A fase B apresenta MR>7% na primeira carga crítica e tem um TAP “-8”. Portanto, a fase B não atende às normas brasileiras.



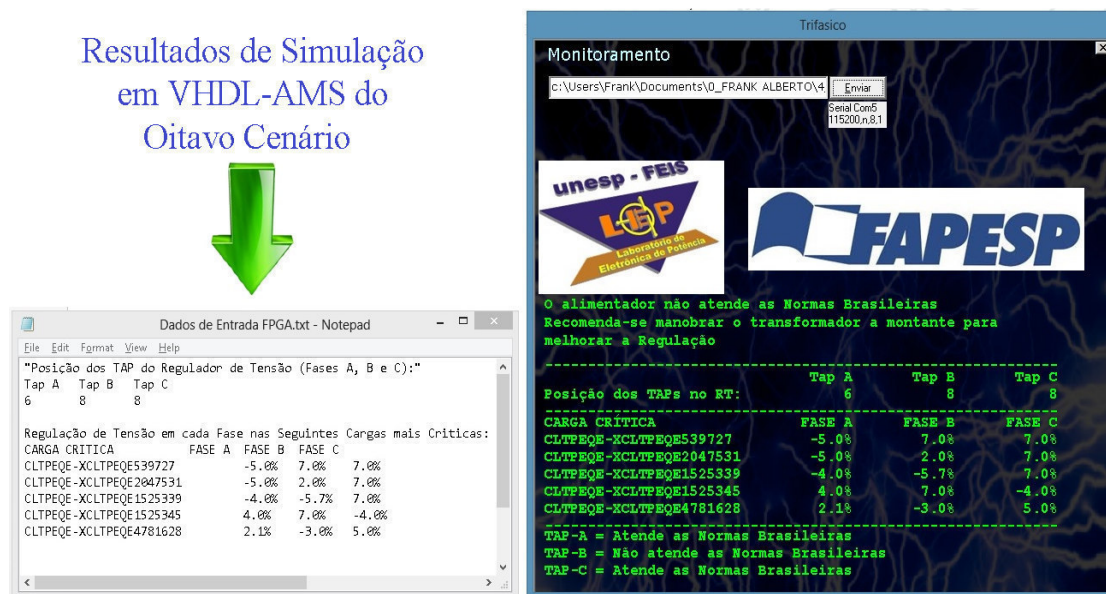
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 76- Execução do DMS&C para o sétimo cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e C, cumprem - 5 % $\leq$ MR% $\leq$ 7 %. A fase B apresenta MR>7% na primeira carga crítica e tem um TAP $\neq$ “-8”. Portanto, na fase B, no regulador de tensão, pede-se diminuir o TAP em uma posição.



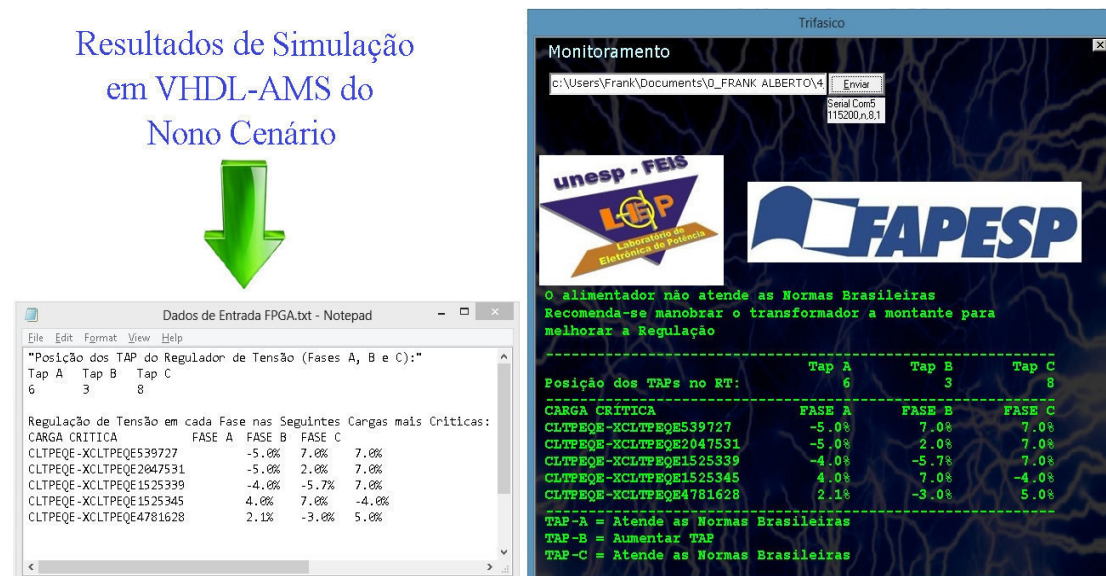
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 77- Execução do DMS&C para o oitavo cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e C, cumprem - $5\% \leq MR\% \leq 7\%$. A fase B apresenta $MR < -5\%$ na terceira carga crítica e tem um TAP “8”. Portanto, a fase B não atende às normas brasileiras.



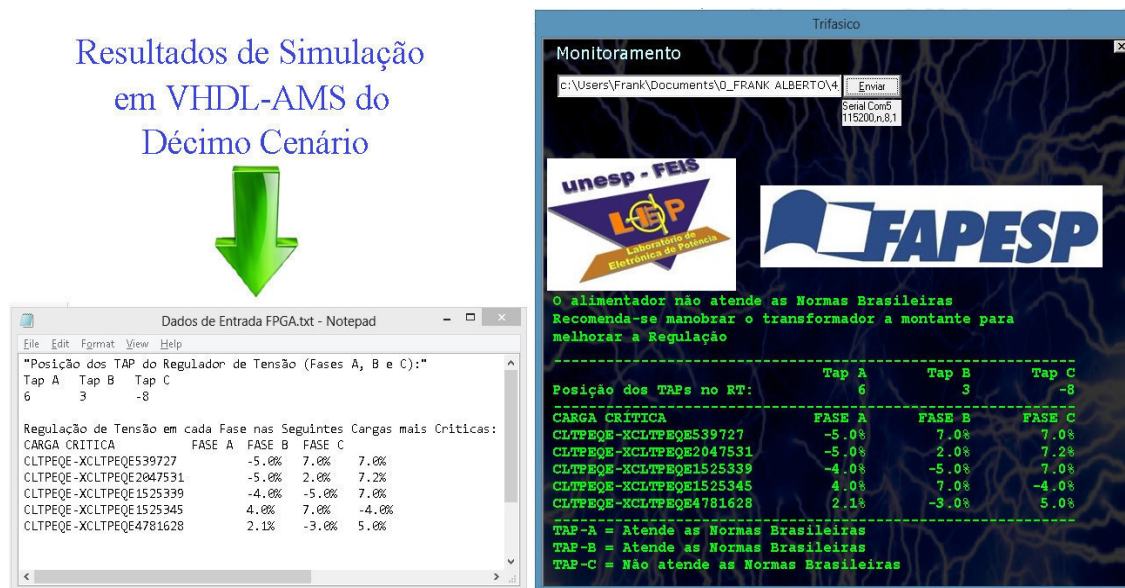
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 78- Execução do DMS&C para o nono cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e C, cumprem - $5\% \leq MR\% \leq 7\%$. A fase B apresenta $MR < -5\%$ na terceira carga crítica e tem um TAP $\neq$ “8”. Portanto, na fase B, no regulador de tensão, pede-se aumentar o TAP em uma posição.



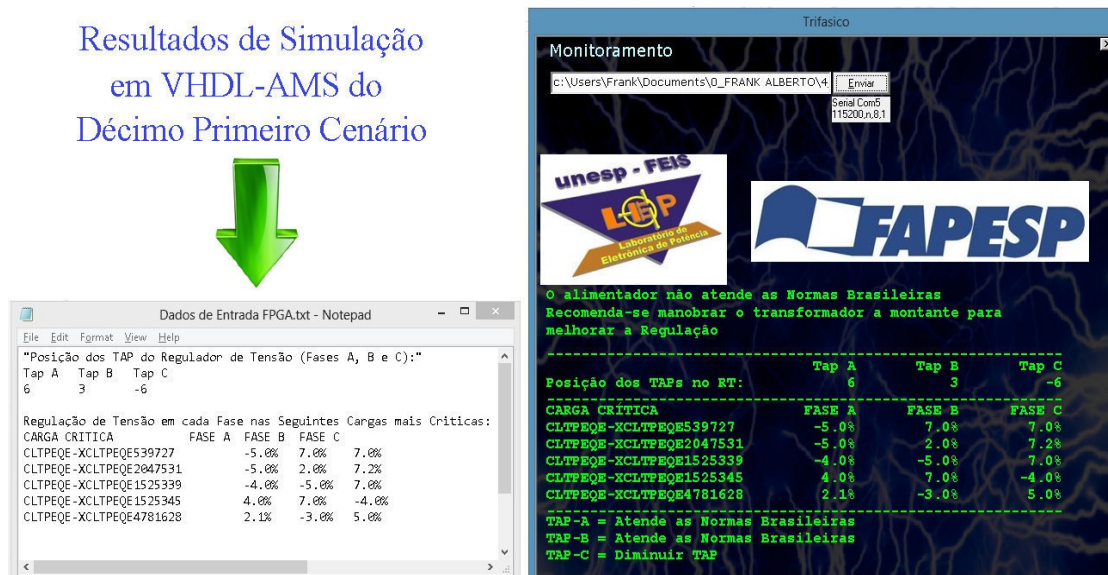
Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 79- Execução do DMS&C para o décimo cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e B, cumprem - 5 % $\leq$ MR% $\leq$ 7 %. A fase C apresenta MR>7% na segunda carga crítica e tem um TAP “-8”. Portanto, a fase C não atende às normas brasileiras.



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 80- Execução do DMS&C para o décimo primeiro cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e B, cumprem - 5 % $\leq$ MR% $\leq$ 7 %. A fase C apresenta MR>7% na segunda carga crítica e tem um TAP $\neq$ “-8”. Portanto, na fase C, no regulador de tensão, pede-se diminuir o TAP em uma posição.



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 81- Execução do DMS&C para o décimo segundo cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e B, cumprem $-5\% \leq MR\% \leq 7\%$. A fase C apresenta $MR < -5\%$ na segunda carga crítica e tem um TAP “8”. Portanto, a fase C não atende às normas brasileiras.

Resultados de Simulação em VHDL-AMS do Décimo Segundo Cenário



Dados de Entrada FPGA.txt - Notepad

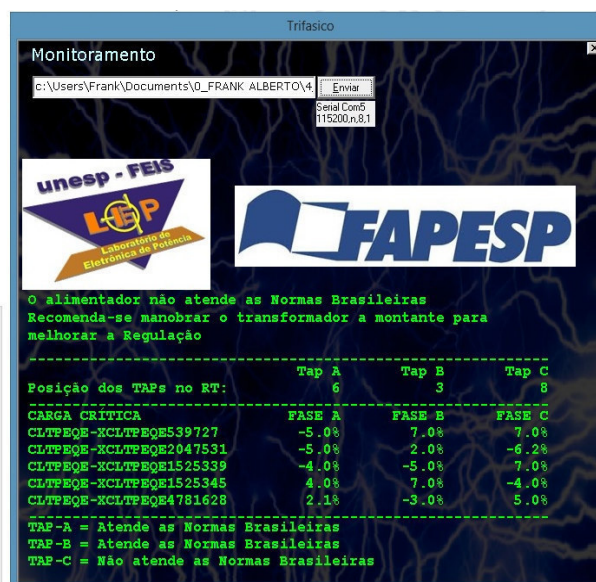
File Edit Format View Help

"Posição dos TAP do Regulador de Tensão (Fases A, B e C):"

Tap A	Tap B	Tap C
6	3	8

Regulação de Tensão em cada Fase nas Seguintes Cargas mais Críticas:

CARGA CRÍTICA	FASE A	FASE B	FASE C
CLTPEQE-XCLTPEQE539727	-5.0%	7.0%	7.0%
CLTPEQE-XCLTPEQE2047531	-5.0%	2.0%	-6.2%
CLTPEQE-XCLTPEQE1525339	-4.0%	-5.0%	7.0%
CLTPEQE-XCLTPEQE1525345	4.0%	7.0%	-4.0%
CLTPEQE-XCLTPEQE4781628	2.1%	-3.0%	5.0%



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 82- Execução do DMS&C para o décimo terceiro cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e B, cumprem $-5\% \leq MR\% \leq 7\%$. A fase C apresenta $MR < -5\%$ na segunda carga crítica e tem um TAP $\neq$ “8”. Portanto, na fase C, no regulador de tensão, pede-se aumentar o TAP em uma posição.

Resultados de Simulação em VHDL-AMS do Décimo Terceiro Cenário



Dados de Entrada FPGA.txt - Notepad

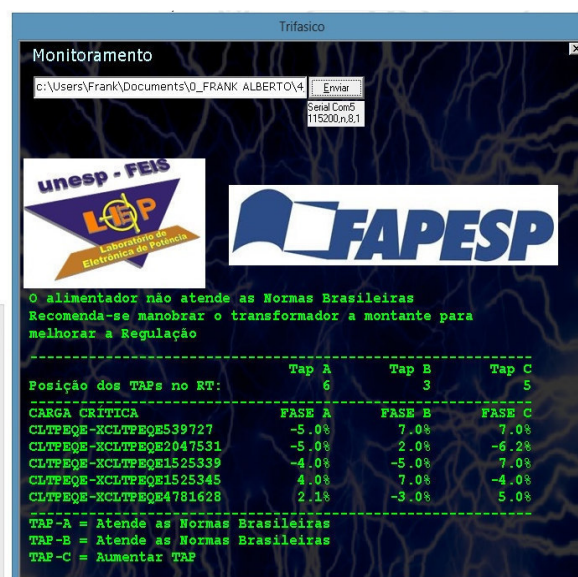
File Edit Format View Help

"Posição dos TAP do Regulador de Tensão (Fases A, B e C):"

Tap A	Tap B	Tap C
6	3	5

Regulação de Tensão em cada Fase nas Seguintes Cargas mais Críticas:

CARGA CRÍTICA	FASE A	FASE B	FASE C
CLTPEQE-XCLTPEQE539727	-5.0%	7.0%	7.0%
CLTPEQE-XCLTPEQE2047531	-5.0%	2.0%	-6.2%
CLTPEQE-XCLTPEQE1525339	-4.0%	-5.0%	7.0%
CLTPEQE-XCLTPEQE1525345	4.0%	7.0%	-4.0%
CLTPEQE-XCLTPEQE4781628	2.1%	-3.0%	5.0%



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Figura 83- Execução do DMS&C quando a placa ZedBoard Zynq™-7000 está desconectada ou desligada.



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

Segundo a Além dos anteriores treze cenários rodados utilizando a nova ASTR&C, tem um decimo quarto cenário, segundo o qual apresenta as piores condiciones de regulação de tensão para a comutação simultânea nas três fases do TAP no regulador de tensão.

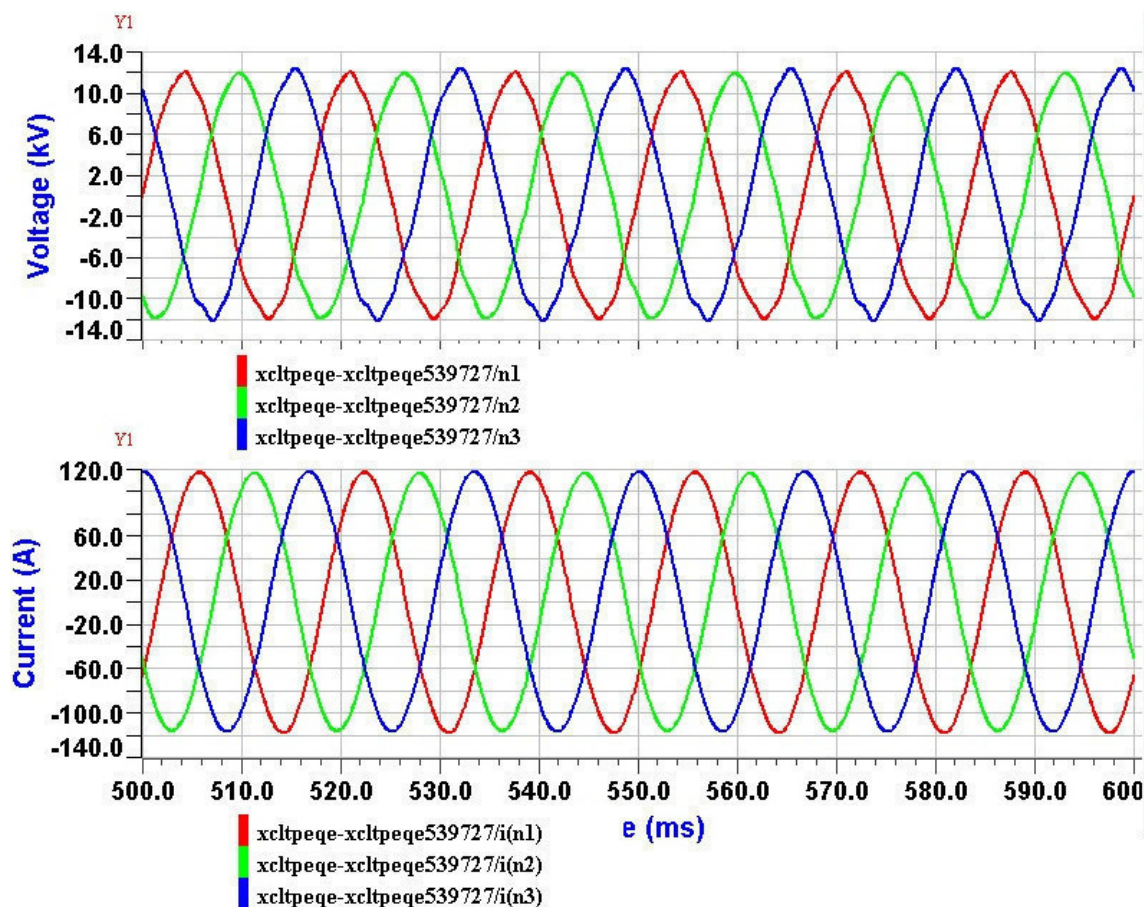
Figura 84- Execução do DMS&C para o décimo quarto cenário de simulação: as magnitudes da regulação de tensão para todas as cinco cargas críticas, na fase A e B, não cumprem - $5\% \leq MR\% \leq 7\%$. As três fases apresentam um TAP= “8” e têm $MR > 7\%$ para todas as cargas. Portanto, nas três fases, no regulador de tensão, pede-se diminuir o TAP em uma posição.



Fonte: Próprio autor utilizando o software de interface entre o PC e a Placa ZedBoard.

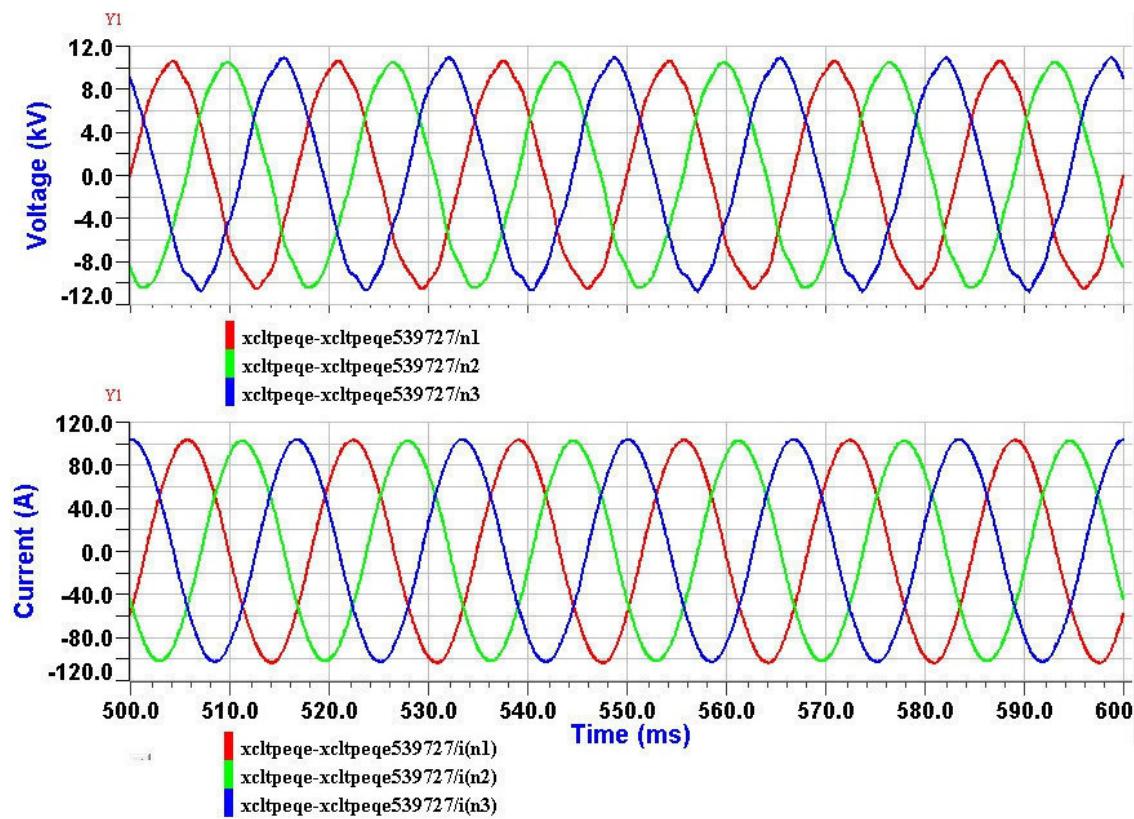
Para finalizar a atuação do DMS&C, as figuras 85 e 86 apresentam as formas de onda de tensão e corrente da carga CLTPEQE-XCLTPEQE539727 com as melhores e as piores condições de regulação de tensão para a comutação simultânea nas três fases do TAP no regulador de tensão (RT), isto é, na figura 85 as posições dos TAP A, B e C no regulador de tensão foram situadas todas em “-2” (melhor condição de regulação de tensão) e, na figura 86 as posições dos TAP A, B e C no regulador de tensão foram situadas todas em “8” (pior condição de regulação de tensão).

Figura 85- Melhor condição de regulação de tensão para a comutação simultânea nas três fases do TAP no RT.



Fonte: Próprio autor.

Figura 86- Pior condição de regulação de tensão para a comutação simultânea nas três fases do TAP no RT.



Fonte: Próprio autor.

Os resultados do desenvolvimento da nova arquitetura de simulação em “tempo real” e controle (ASTR&C), para um alimentador de distribuição de energia elétrica, utilizando as linguagens de descrição de hardware VHDL-AMS e VHDL, mostrados nas figuras 70 até 82, e 84, podem ser sumarizados na tabela 22.

TABELA 22- RESULTADOS EXEMPLOS DO DESENVOLVIMENTO DA NOVA ARQUITETURA DE SIMULAÇÃO EM “TEMPO REAL” E CONTROLE PARA UM ALIMENTADOR DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA.

Linguagem VHDL-AMS																		Linguagem VHDL				
Cenários	Posições dos TAPs no RT			Magnitudes das Regulações de Tensão das Cinco Cargas mais críticas (%)															Sistema de Gerenciamento da Distribuição e Controle (DMS&C)			
				CLTPEQE-XCLTPEQE539727			CLTPEQE-XCLTPEQE2047531			CLTPEQE-XCLTPEQE1525339			CLTPEQE-XCLTPEQE1525345			CLTPEQE-XCLTPEQE4781628						
	TAP A	TAP B	TAP C	Fase A	Fase B	Fase C	Fase A	Fase B	Fase C	Fase A	Fase B	Fase C	Fase A	Fase B	Fase C	Fase A	Fase B	Fase C	FASE A	FASE B	FASE C	Alimentador
0	-5	8	-5	-6,1	7,0	-3,5	9,5	2,0	6,2	-14,0	-3,2	2,0	-8,0	2,0	4,0	-9,1	3,0	5,0	↓ TAP	ANB	ANB	NAN, RMT
1	-2	-2	-2	2,7	1,8	3,6	2,7	1,8	3,6	2,7	2,7	2,7	2,6	2,7	2,6	2,6	1,8	3,5	ANB	ANB	ANB	ANB
2	-8	-7	8	-5,0	5,0	7,0	7,0	2,0	7,0	-4,0	5,7	7,0	8,0	7,0	-4,0	-5,0	-3,0	5,0	NAN	ANB	ANB	NAN, RMT
3	5	1	1	8,2	3,3	3,3	8,2	3,3	3,3	8,2	3,2	3,3	8,2	3,2	3,2	8,1	3,2	3,2	↓ TAP	ANB	ANB	NAN, RMT
4	8	-7	8	-5,0	5,0	7,0	-5,1	2,0	7,0	-4,0	5,7	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	NAN	ANB	ANB	NAN, RMT
5	6	-7	8	-5,0	5,0	7,0	-5,1	2,0	7,0	-4,0	5,7	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	↑ TAP	ANB	ANB	NAN, RMT
6	6	-8	8	-5,0	7,1	7,0	-5,0	2,0	7,0	-4,0	5,7	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	NAN	ANB	NAN, RMT
7	6	5	8	-5,0	7,1	7,0	-5,0	2,0	7,0	-4,0	5,7	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	↓ TAP	ANB	NAN, RMT
8	6	8	8	-5,0	7,0	7,0	-5,0	2,0	7,0	-4,0	-5,7	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	NAN	ANB	NAN, RMT
9	6	3	8	-5,0	7,0	7,0	-5,0	2,0	7,0	-4,0	-5,7	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	↑ TAP	ANB	NAN, RMT
10	6	3	-8	-5,0	7,0	7,0	-5,0	2,0	7,2	-4,0	-5,0	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	ANB	NAN	NAN, RMT
11	6	3	-6	-5,0	7,0	7,0	-5,0	2,0	7,2	-4,0	-5,0	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	ANB	↓ TAP	NAN, RMT
12	6	3	8	-5,0	7,0	7,0	-5,0	2,0	-6,2	-4,0	-5,0	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	ANB	NAN	NAN, RMT
13	6	3	5	-5,0	7,0	7,0	-5,0	2,0	-6,2	-4,0	-5,0	7,0	4,0	7,0	-4,0	2,1	-3,0	5,0	ANB	ANB	↑ TAP	NAN, RMT
14	8	8	8	11,8	11,1	12,0	11,8	11,1	12,0	11,8	11,6	11,8	11,8	11,2	11,8	11,8	11,0	12,0	↓ TAP	↓ TAP	↓ TAP	NMB, RCT

Fonte: Próprio autor

Convenções da Tabela: Embora os valores sublinhados com azul apresentam uma condição atípica, na tabela existem as seguintes convenções:

Cenários: Cenários de Simulação em Ambiente VHDL-AMS (segundo a Tabela 4-1).	↑ TAP: Aumentar TAP em uma Posição.	ANB: Atende às normas brasileiras.
RMT: Recomenda-se manobrar o Transformador a Montante do Alimentador para Melhorar a Regulação de Tensão.	↓ TAP: Diminuir TAP em uma Posição.	NAN: Não Atende às normas brasileiras.

4.4 CONCLUSÕES

Neste capítulo foi apresentado o desenvolvimento de uma Arquitetura de Simulação em “Tempo Real” e Controle (ASTR&C) para um alimentador de distribuição de energia elétrica de 14,4 kV de tensão de linha (média tensão), utilizando as linguagens de descrição de hardware VHDL-AMS e VHDL. A ASTR&C utilizou uma plataforma VHDL-AMS como GUI para desenvolver a simulação do sistema elétrico de distribuição e a linguagem VHDL para o desenvolvimento do sistema de gerenciamento e controle (DMS&C), através de um dispositivo ZYNQ SOC XC7Z020-CLG484, inserido em uma placa de desenvolvimento ZedBoard Zynq™-7000. Ambas linguagens de descrição de hardware VHDL e VHDL-AMS, juntamente com as informações do alimentador elétrico de distribuição, estudo de caso, tornaram possível a simulação em “tempo real” e controle do alimentador, considerando-se o parâmetro de regulação de tensão para o mesmo.

Os resultados de simulação e experimentais apresentados, para a ASTR&C, permitem concluir que o emprego das linguagens VHDL e VHDL-AMS é eficaz e apropriado para o desenvolvimento e implementação desta nova arquitetura de simulação e controle. Os recursos lógicos utilizados não ultrapassaram a 3,3 % dos recursos disponíveis no dispositivo ZYNQ SOC XC7Z020-CLG484 (para o desenvolvimento do DMS&C), além de atender a máxima frequência de operação permitida pelo sistema.

Quanto ao sistema elétrico de distribuição utilizado como estudo de caso, pôde-se comprovar que, através do DMS&C foi possível controlar o perfil de tensão (segundo as normas brasileiras estipuladas pela Agência Nacional de Energia Elétrica - ANEEL), indicando os ajustes necessários para os TAPs do regulador de tensão deste alimentador, segundo as advertências e recomendações feitas pelo sistema de gerenciamento e controle (DMS&C). Obviamente, o controlador digital não atuará diretamente no sistema real (alimentador real de distribuição elétrica), considerando-se as hierarquias de controle e proteção deste alimentador, dadas suas variações dinâmicas/operacionais. Logo, a ação real a ser operada, por exemplo, em um regulador de tensão para alteração de TAP será advinda de uma lógica adicional (não implementada nesta tese) entre as indicações do controlador digital (DMS&C) e os reais controladores de rede, tendo-se ainda em conta, para o parâmetro de regulação de tensão, o melhor equilíbrio em valores eficazes para as tensões reguladas.

Finalmente, no capítulo 5 apresentam-se as conclusões gerais para a tese e, adicionalmente, algumas propostas de continuidade para o trabalho desenvolvido.

CONCLUSÕES GERAIS E TRABALHOS FUTUROS

Com base no exposto nos capítulos anteriores, podem-se apresentar as seguintes conclusões gerais e propostas para trabalhos futuros.

5.1 CONCLUSÕES GERAIS

Foi desenvolvida uma nova Arquitetura de Simulação em ‘Tempo Real’ e Controle (ASTR&C) para alimentadores elétricos de distribuição, com o intuito de analisar a qualidade da energia e melhorar as ações de controle da regulação de tensão para sistemas de distribuição de energia elétrica, envolvendo, especificamente, o controle de regulador de tensão, com base na mudança/comutação de TAP. Esta ASTR&C utiliza uma plataforma SystemVision™ (VHDL-AMS) como interface gráfica do usuário (GUI), para desenvolver a simulação do sistema elétrico e a linguagem VHDL para o desenvolvimento do sistema de gerenciamento e controle, através de um dispositivo ZYNQ SoC XC7Z020-CLG484, que encapsula um FPGA Artix-7 e um processador ARM Cortex-A9 de núcleo duplo. Ambas linguagens de descrição de hardware VHDL e VHDL-AMS, juntamente com as informações do sistema elétrico de distribuição, tornaram possível a simulação em “tempo real” e controle de um alimentador de distribuição de energia elétrica estudo de caso, com dados obtidos de um alimentador real de distribuição. O alimentador estudo de caso resultou da aplicação de uma ferramenta de redução de rede e de concentração de cargas, para os dados reais do alimentador considerado. Neste contexto, pode-se chegar às seguintes conclusões gerais derivadas do desenvolvimento da pesquisa:

- A ampla revisão bibliográfica, detalhada no capítulo 1 e o apêndice C, permitiu observar o estado da arte para a simulação em tempo real e controle de sistemas de distribuição de energia elétrica. Neste contexto foi analisada a viabilidade da inovação principal deste trabalho de tese, a qual se constituiu no desenvolvimento de uma nova arquitetura de simulação em “tempo real” e controle para sistemas elétricos de distribuição, como uma nova ferramenta voltada à avaliação da qualidade da energia elétrica e ao melhoramento das ações de controle de regulação de tensão para

alimentadores de distribuição, utilizando-se as linguagens de descrição de hardware VHDL e VHDL-AMS. Além disso, a revisão bibliográfica preliminar permitiu estabelecer uma metodologia de projeto para o desenvolvimento da pesquisa e tese.

- Levando-se em consideração que as redes dos sistemas elétricos de potência podem ser computacionalmente resolvidas com base em métodos computacionais no domínio do tempo e no domínio da frequência, a linguagem VHDL-AMS esta concebida para especificar o comportamento de sistemas analógicos no domínio do tempo, através da descrição de sistemas de equações diferenciais ordinárias e equações algébricas.
- Na implementação da simulação do sistema elétrico de distribuição escolhido como estudo de caso, realizada em plataforma SystemVision™ (VHDL-AMS), verificou-se que uma das vantagens desta linguagem, que além de permitir a modelação de sistemas analógicos, digitais e de sinal misto, é a sua flexibilidade para a criação de elementos e componentes circuitais, assim como códigos VHDL-AMS diretamente. Isto é facilitado pelo System Vision™, uma vez que o mesmo inclui um editor de modelos de elementos em VHDL-AMS (disponíveis em bibliotecas especializadas), que ajuda o projetista a integrar várias características do esquema do projeto.
- Todas as simulações realizadas em VHDL-AMS foram confrontadas com simulações nas quais foram utilizados softwares amplamente conhecidos: Orcad/Pspice ou Matlab/Simulink. As comparações destas simulações foram desenvolvidas com sistemas elétricos idênticos. Neste contexto, segundo os resultados das simulações em VHDL-AMS (tabela 16 no capítulo 2), podemos afirmar que o Orcad/Pspice apresentou maiores similaridades com os resultados das simulações em VHDL-AMS, em comparação com o Matlab/Simulink.
- No contexto do desenvolvimento das simulações em VHDL-AMS, para o alimentador escolhido como estudo de caso, o intervalo de tempo de simulação utilizado para os cálculos dos valores eficazes das grandezas de interesse foi de 500 até 600 milissegundos, resultando em dados muito satisfatórios (em regime permanente).
- Para o desenvolvimento do sistema de gerenciamento (DMS) e controle, foi considerado o trabalho do (SCHWIEGELSHOHN e HUBNER, 2014), para a escolha da placa de desenvolvimento ZedBoard Zynq™-7000, a qual se confirmou como adequada para a aplicação implementada e para atender aos objetivos deste trabalho.
- O dispositivo ZYNQ SoC XC7Z020-CLG484, que contém um FPGA Artix-7 e um processador ARM Cortex-A9 de núcleo duplo, em conjunto com as ferramentas de síntese (**Vivado Design Suite**) e a linguagem de descrição de hardware VHDL, foram

elementos essenciais na implementação do sistema digital de gerenciamento e controle do alimentador de distribuição estudo de caso, de forma rápida e com elevada flexibilidade, permitindo um projeto no estilo “**top-down**”, baseado em funções com elevado nível de abstração.

- O sistema de gerenciamento e controle desenvolvido apresenta uma independência de tecnologia, na qual, diante da necessidade da mudança de plataforma do sistema para outro dispositivo lógico programável com maior capacidade ou mais sofisticado, o processo de migração é automático e controlado pela ferramenta de síntese **Vivado Design Suite**, em conjunto com o ambiente de desenvolvimento integrado, o kit de desenvolvimento de software - SDK (**Software Development Kit**).
- Os resultados de simulação e experimentais apresentados permitem concluir que o emprego do dispositivo ZYNQ SoC XC7Z020-CLG484 e o desenvolvimento da lógica de controle digital, empregando o kit de desenvolvimento de software – SDK, utilizando a linguagem de descrição de hardware VHDL, foram eficazes e apropriados para o desenvolvimento e implementação do sistema de gerenciamento e controle do alimentador de distribuição estudo de caso. Restando ainda a quase totalidade de seus recursos computacionais para o desenvolvimento dos trabalhos futuros, considerando-se as continuidades demandadas para a pesquisa.
- Pode-se observar, através dos resultados de simulação e experimentais, que a nova arquitetura de simulação e controle proposta apresentou as características desejadas no que se refere ao parâmetro de controle: Regulação de Tensão do alimentador. Dentre estas características podem-se mencionar:
 - ✓ Implementa o conceito de simulação em “tempo real” e controle, segundo o qual um sistema elétrico de distribuição real é reproduzido identicamente em uma interface gráfica do usuário (plataforma SystemVision™), em linguagem VHDL-AMS, e dependendo dos resultados destas simulações, um sistema de gerenciamento toma ações de controle, através de indicações operacionais que devem ser implementadas hierarquicamente no sistema elétrico real.
 - ✓ De maneira básica, pode-se visualizar a aplicação do conceito de teste de equipamento em malha fechada (Hardware-In-the-Loop), no qual, o sistema integrado em teste seria o controlador digital programável, que está formado pela placa de desenvolvimento ZedBoard Zynq™-7000 e o software de interface para a transferência de dados entre o PC e a placa.

Apesar dos avanços alcançados nesta tese, considerando-se a modelação desenvolvida em VHD-AMS e controlador digital em VHDL, os resultados da pesquisa não estão ainda completamente embarcados no dispositivo FPGA. Ou seja, os modelos desenvolvidos em VHDL-AMS, para elementos das redes de distribuição de energia elétrica, ainda deverão ser embarcados em dispositivo FPGA, de tal forma que a modelação seja automática em função destes modelos estabelecidos, assim como, que as simulações ocorram também no ambiente FPGA, resultando em enorme redução para os tempos de simulação, aproximando ainda mais os resultados do conceito ótimo desejado para Simulação Digital em Tempo Real.

Portanto, estabelecem-se as seguintes propostas de continuidade para a pesquisa tema desta tese.

5.2 TRABALHOS FUTUROS

Em função dos resultados obtidos neste trabalho, diferentes linhas de pesquisas podem ser determinadas para a continuidade dos trabalhos, quais sejam:

- 5.2.1 Inserção dos modelos em linguagem VHDL-AMS dentro de um dispositivo lógico programável, embarcando num dispositivo de hardware programável os algoritmos de simulação para redes elétricas de distribuição.
- 5.2.2 Refinamentos para o sistema de controle digital da operação (ASTR&C), envolvendo diversos elementos de controle dos sistemas elétricos de distribuição (múltiplos reguladores de tensão, bancos de capacitores chaveados, filtros passivos ressonantes chaveados, dentre outros para as redes elétricas do futuro), com as finalidades de regulação ótima de tensão (ajustes V/var) e ainda de mitigação harmônica.
- 5.2.3 Inserção de modelos de geradores distribuídos (fotovoltaicos e eólicos, por exemplo), envolvendo ainda alterações associadas ao controlador digital da operação, permitindo ainda o futuro gerenciamento de microrredes isoladas.
- 5.2.4 Projeto, desenvolvimento e montagem de um protótipo em escala reduzida para um sistema elétrico de distribuição, inteligente (*smart grid*), que possua:
 - 5.2.4.1 Controladores da rede elétrica, os quais farão o controle elétrico local dos principais equipamentos elétricos que pertencem à rede. Estes controladores são dispositivos de hardware que se constituem como uma interface entre o exterior do equipamento elétrico e o seu funcionamento interno, pela qual é possível comandar o equipamento individualmente, na forma local ou remota.

- 5.2.4.2** Sistema de aquisição e armazenamento de dados da rede, relacionado com as medições das principais grandezas do sistema elétrico, com vistas a avaliar os principais indicadores de qualidade da energia elétrica na rede de distribuição.
- 5.2.5 Desenvolvimento de uma interface dinâmica que opere em tempo real com o sistema real de distribuição em escala reduzida, proposto anteriormente. Essa interface deve ser capaz de comandar os controladores da rede elétrica, considerando-se lógicas de controle com as hierarquias devidas e estabelecidas para a rede em escala reduzida.
- 5.2.6 Desenvolvimento de uma interface entre o controlador lógico programável (PLC) de um sistema elétrico de distribuição (localizado na subestação a montante do alimentador elétrico), e a plataforma VHDL-AMS, com o intuito de modelar nesta linguagem, qualquer sistema de distribuição real (ou projetado no laboratório), para o qual se possa acessar a sua topologia, configuração e medições significativas em tempo real por meio de módulos de entradas e saída (analógico-digitais).
- 5.2.7 Inserção no controlador digital, além do gerenciamento do perfil de tensão e da otimização da regulação (V/var), as seguintes funcionalidades: 1) Ilhamento para localização de faltas e restauração do serviço, 2) Rejeição de carga de emergência, 3) Gerenciamento de carregamento de veículos elétricos, 4) Gerenciamento de múltiplas microrredes, 5) Gerenciamento do fluxo de potências bidirecionais e em quatro quadrantes no alimentador, dentre outras.

Referências

- AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA – ANEEL. **Procedimentos de distribuição de energia elétrica no sistema elétrico nacional, módulo 8:** qualidade da energia elétrica. [S.l.: s.n], 2011. Disponível em: <http://www.aneel.gov.br/arquivos/pdf/modulo8_revisao_2.pdf>. Acesso em: 10 jan. 2014.
- AL-KASHEF, A.; ZAKY, M.M.; DESSOUKY, M.; EL-GHITANI, H. A case-based reasoning approach for the automatic generation of VHDL-AMS models. In: IEEE INTERNATIONAL BEHAVIORAL MODELING AND SIMULATION WORKSHOP-(BMAS, 12., 2008, San Jose. **Proceedings...** San Jose: IEEE, 2008. p. 100-105.
- ALVES, F. **Deteção de falhas em sistema de distribuição de energia elétrica usando dispositivos programáveis.** 2008. 197 f. Dissertação (Mestrado em Engenharia Elétrica) – Faculdade de Engenharia, Universidade Estadual Paulista, Ilha Solteira, 2008.
- ARAMIZU, J.; VIEIRA, J. C. M. Analysis of PV generation impacts on voltage imbalance and on voltage regulation in distribution networks. In: POWER AND ENERGY SOCIETY GENERAL MEETING- PES, 11., 2013, Vancouver. **Proceedings...** Vancouver: IEEE, 2013. p. 1-5.
- ATTIA, H. A. Optimal voltage profile control and losses minimization of radial distribution feeders. In: INTERNATIONAL MIDDLE-EAST POWER SYSTEM CONFERENCE - MEPCON, 12., 2008, Aswan. **Proceedings...** Aswan: IEEE, 2008. p. 453-458.
- AUNG, M.; DINAVAHI, V. FPGA-based real-time emulation of power electronic systems with detailed representation of device characteristics. In: POWER AND ENERGY SOCIETY GENERAL MEETING (PES), 9., 2011, San Diego. **Proceedings...** San Diego: IEEE, 2011. p. 1-11.
- BACHIR, T. O.; DAVID, J. P. ; DUFOUR, C.; BÉLANGER, J. Effective FPGA-based electric motor modeling with floating-point cores. In: ANNUAL CONFERENCE ON IEEE INDUSTRIAL ELECTRONICS SOCIETY - IECON, 36., 2010, Glendale. **Proceedings...** Glendale: IEEE, 2010. p. 829-834.
- BALDWIN, J. D.; CHENG, P.; GARRAULT, P. Geração de um projeto específico de documento modelo de entrada/Saída (E/S). **Patente dos Estados Unidos.** Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetathtml%2FFPTO%2Fsrchnu m.htm&r=1&f=G&l=50&s1=7882484.PN.&OS=PN/7882484&RS=PN/7882484>>. Acesso em: 25 set. 2012.
- BISWAS, P.; MAYILADUTHURAI RAMESH, S.; CHETPUT, C. L.; KOLPEKWAR, A. Conexão de componentes em Verilog-AMS e VHDL-AMS num projeto de sinal misto com uma linguagem mista. **Patente dos Estados Unidos.** Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetathtml%2FFPTO%2Fsrchnu m.htm&r=1&f=G&l=50&s1=7882484.PN.&OS=PN/7882484&RS=PN/7882484>>. Acesso em: 25 set. 2012.

m.htm&r=1&f=G&l=50&s1=7251795.PN.&OS=PN/7251795&RS=PN/7251795>. Acesso em: 30 set. 2012.

BOUSSETTA, H.; MARZENCKI, M.; BASROUR, S.; SOUDANI, A. Efficient physical modeling of MEMS energy harvesting devices with VHDL-AMS. **Sensors Journal of IEEE**, New York, v. 10, n. 9, p. 1427-1437, 2010.

BREWTON, N. E. Estrutura variável em modelagem física. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnethtml%2FPTO%2Fsrchnu m.htm&r=1&f=G&l=50&s1=7398192.PN.&OS=PN/7398192&RS=PN/7398192>>. Acesso em: 30 ago. 2012.

CARROLL, M. VHDL-panacea or hype?. **IEEE Spectrum**, New York, v. 30, n. 6, p. 34-37, 1993.

CASTELLAZZI, A.; CIAPPA, M.; MERMET-GUYENNET, M.; FICHTNER, W. VHDL-AMS simulation of integrated power systems: a unified solution for multi-domain multi-level abstraction analysis. In: CONFERENCE INTEGRATED POWER SYSTEMS- CIPS, 5., 2008, Nuremberg. **Proceedings...** Nuremberg: IEEE, 2008. p. 1-6.

CASTRO, A. de.; ZUMEL, P.; GARCÍA, O.; RIESGO, T.; UCEDA, J. Concurrent and simple digital controller of an AC/DC converter with power factor correction based on an FPGA. **IEEE Transactions on Power Electronics**, New York, v. 18, n. 1, p. 334-343, 2003.

CASU, M. R.; CREPALDI, M.; GRAZIANO, M. A VHDL-AMS Simulation Environment for an UWB Impulse Radio Transceiver. **IEEE Transactions on Circuits and Systems I**, New York, v. 55, n. 5, p. 1368-1381, 2008.

CHRISTEN, E.; BAKALAR, K. VHDL-AMS-a hardware description language for analog and mixed-signal applications. **IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing**, New York, v. 46, n. 10, p. 1263-1272, 1999.

CHRISTEN, E.; VREUGDENHIL G. J.; VLACH, M. Classificação das variáveis em um sistema de equações simultâneas descrito por linguagens de descrição de hardware. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnethtml%2FPTO%2Fsrchnu m.htm&r=1&f=G&l=50&s1=6532569.PN.&OS=PN/6532569&RS=PN/6532569>>. Acesso em: 10 out. 2012.

CHUNG-HSING, C.; KARNIK, T.; SAAB, D. G. Structural and behavioral synthesis for testability techniques. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 13, n. 6, p. 777-785, 1994.

CLUNE, M. I.; APTE, A. D. Processamento de eventos de combinação de entidades em ambientes de modelagem. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnethtml%2FPTO%2Fsrchnu>>

m.htm&r=1&f=G&l=50&s1=7966161.PN.&OS=PN/7966161&RS=PN/7966161>. Acesso em: 25 set. 2012.

COOKE, L. H.; VENKATRAMANI KUMAR; SHYR JIN-SHENG. Bloco baseado em metodologia de projeto com componentes programáveis. **Patente da World Intellectual Property Organization (WIPO)**. International and National Patent Collections. Disponível em:

<<http://patentscope.wipo.int/search/en/detail.jsf?docId=WO2002075611&recNum=1&docAn=US2002008310&queryString=PCT/US02/08310&maxRec=1>>. Acesso em: 15 out. 2012.

DAMON, D.; CHRISTEN, E. Introduction to VHDL-AMS. 1: structural and discrete time concepts. In: IEEE INTERNATIONAL SYMPOSIUM ON COMPUTER-AIDED CONTROL SYSTEM DESIGN, 4., 1996, Dearborn. **Proceedings...** Dearborn: IEEE, 1996. p. 264-269.

DE LOS ANGELES GOMEZ LOPEZ, M.; GOY, C. B.; HERRERA, M. C. Design, implementation and evaluation of a FPGA embedded digital systems course at the university level. **IEEE Latin America Transactions (Revista IEEE de America Latina)**, New York, v. 11, n. 1, p. 137-142, 2013.

DEWEY, A.; DE GEUS, A. J. VHDL: toward a unified view of design. **IEEE Design & Test of Computers**, New York, v. 9, n. 2, p. 8-17, 1992.

DIGILENT. **ZedBoard Zynq™-7000 Development Board**. [S.l: s.n], 2015. Disponível em: <<https://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,1028&Prod=ZEDBOARD>>. Acesso em: 10 jan. 2015.

DINAVAH, V.; IRAVANI, R.; BONERT, R. Design of a real-time digital Simulator for a D-STATCOM system. **IEEE Transactions on Industrial Electronics**, New York, v. 51, n. 5, p. 1001-1008, 2004.

DUFOUR, C.; BELANGER, J.; ABOURIDA, S.; LAPOINTE, V. FPGA-Based Real-Time Simulation of Finite-Element Analysis Permanent Magnet Synchronous Machine Drives. In: POWER ELECTRONICS SPECIALISTS CONFERENCE- PESC, 38., 2007, Orlando. **Proceedings...** Orlando: IEEE, 2007. p. 909-915.

ELECTRIC POWER RESEARCH INSTITUTE - EPRI. **Integrating smart distributed energy resources with distribution management systems**. [S.l: s.n], 2012. Disponível em: <<http://tdworld.com/go-grid-optimization/integrating-smart.pdf>>. Acesso em: 15 dez. 2014.

FAES PHILIPPE PAUL HENRI; ELECKHAUT HENDRIK RICHARD PIETER. Dispositivo e método para a refatoração do código de hardware. **Patente da World Intellectual Property Organization (WIPO)**. International and National Patent Collections. Disponível em: <[http://patentscope.wipo.int/search/en/detail.jsf?docId=WO2010040567&recNum=1&docAn=EP2009007307&queryString=FP:\(PCT/EP2009/007307\)&maxRec=1](http://patentscope.wipo.int/search/en/detail.jsf?docId=WO2010040567&recNum=1&docAn=EP2009007307&queryString=FP:(PCT/EP2009/007307)&maxRec=1)>. Acesso em: 15 nov. 2012.

FATTOUH FARAG; ABDENNADHER SALEM. Modelagem de um circuito. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em:

<<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnethtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=6868534.PN.&OS=PN/6868534&RS=PN/68685341&u=%2Fnethtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=6868534.PN.>>. Acesso em: 12 ago. 2012.

FAVORETTO, C. **Protótipo Virtual da Estratégia DTC (Controle Direto de Torque) Aplicada a Motores de Indução Usando Linguagem VHDL**. 2006. 187 f. Dissertação (Mestrado em Engenharia Elétrica) – Escola de Engenharia de São Carlos da Universidade de São Paulo, Universidade de São Paulo, São Carlos, 2006.

FAZANARO, F. **Estudos e Implementações de Dinâmica Caótica utilizando Dispositivos Analógicos Reconfiguráveis**. 2007. 196 f. Dissertação (Mestrado em Engenharia Elétrica) – Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2007.

FREEMAN, R. User-programmable gate arrays. **IEEE Spectrum**, New York, v. 25, n. 13, p. 32-35, 1988.

FREY, P.; NELLAYAPPAN, K.; SHANMUGASUNDARAM, V.; MAYILADUTHURAI, R.S.; CHANDRASHEKAR, C.L.; CARTER, H.W. SEAMS: simulation environment for VHDL-AMS. In: SIMULATION CONFERENCE- WSC, 9., 1998, Washington. **Proceedings...** Washington: IEEE, 1998. p. 539-546.

GRITTKER UDO; HUMPERT AXEL; FRÜHAUF DIETMAR; GIRARDEY ROMUALDO; BECKER JÜRGEN; PAULSSON KATARINA; HÜBNER MICHAEL. Sistema para a configuração flexível de módulos funcionais. **Patente da World Intellectual Property Organization (WIPO)**. International and National Patent Collections. Disponível em: <<http://patentscope.wipo.int/search/en/detail.jsf?docId=WO2008046696&recNum=1&docAn=EP2007059442&queryString=PCT/EP2007/059442&maxRec=1>>. Acesso em: 5 nov. 2012.

GOYAL, S.; LEDWICH, G.; GHOSH, A. Power Network in Loop: A Paradigm for Real-Time Simulation and Hardware Testing. **IEEE Transactions on Power Delivery**, New York, v. 25, n. 2, p. 1083 - 1092, 2010.

GREEN, R. C.; LINGFENG WANG ; ALAM, M. Applications and Trends of High Performance Computing for Electric Power Systems: Focusing on Smart Grid. **IEEE Transactions on Smart Grid**, New York, v. 4, n. 2, p. 922 - 931, 2013.

GUESSAB, S.; OUDINOT, J. Modeling of a piezoelectric device with shocks management using VHDL-AMS. In: BEHAVIORAL MODELING AND SIMULATION CONFERENCE, 8., 2004, Orlando. **Proceedings...** Orlando: IEEE, 2004. p. 19-24.

GUIHAL, D.; ANDRIEUX, L.; ESTEVE, D.; CAZARRE, A. VHDL-AMS Model Creation. In: MIXED DESIGN OF INTEGRATED CIRCUITS AND SYSTEM (MIXDES), 13., 2006, Gdynia. **Proceedings...** Gdynia: IEEE, 2006. p. 549-554.

GUNGOR, V.C.; SAHIN, D.; KOCAK, T.; ERGUT, S.; BUCCELLA, C.; CECATI, C.; HANCKE, G.P. Smart Grid Technologies: Communication Technologies and Standards. **IEEE Transactions on Industrial Informatics**, New York, v. 7, n. 4, p. 529-539, 2011.

HIRSCH, H.; CHAWLA, P.; CARTER, H. W. Parallel simulation of VHDL-AMS models. In: AEROSPACE AND ELECTRONICS CONFERENCE- NAECON, 11., 1998, Dayton. **Proceedings...** Dayton: IEEE, 1998. p. 545-551.

HOLLINGWORTH, P. The rise of VHDL 1076 and all that. **IEE Review**, New York, v. 37, n. 4, p. 139-142, 1991.

HUABIAO, Q.; FEI, W. Modeling of Operational Amplifier based on VHDL-AMS. In: IEEE INTERNATIONAL CONFERENCE ON ELECTRONICS, Circuits and Systems- ICECS, 13., 2006, Nice. **Proceedings...** Nice: IEEE, 2006. p. 894-897.

HUANG H. J.; FANG C. H.; FAN C. P. Very-large-scale integration design of a low-power and cost-effective context-based adaptive variable length coding decoder for H.264/AVC portable applications. **IEEE Image Processing IET**, New York, v. 6, n. 2, p. 104-114, 2012.

IBARRA HERNANDEZ, F.; CANESIN, C. A.; ZAMORA, R.; SRIVASTAVA, A. K. Active power management in multiple microgrids using a multi-agent system with JADE. In: IEEE/IAS INTERNATIONAL CONFERENCE ON INDUSTRY APPLICATIONS- INDUSCON, 11., 2014, Juiz de Fora. **Proceedings...** Juiz de Fora: IEEE, 2014. p. 1-8.

IBARRA HERNANDEZ, F.; CANESIN, C. A.; ZAMORA, R.; MARTINA, F.; SRIVASTAVA, A. K. Energy management and control for islanded microgrid using multi-agents. In: NORTH AMERICAN POWER SYMPOSIUM (NAPS), 11., 2013, Manhattan. **Proceedings...** Manhattan: IEEE, 2013. p. 1-6.

IBARRA HERNANDEZ, F.; CANESIN, C. A. Electrical power distribution system modeling with VHDL-AMS for the construction of a Real-Time Digital Simulator using FPGAS devices. In: IEEE/IAS INTERNATIONAL CONFERENCE ON INDUSTRY APPLICATIONS- INDUSCON, 10., 2012, Fortaleza. **Proceedings...** Fortaleza: IEEE, 2012. p. 1-7.

JABER, K.; BEN SALEH, B.; FAKHFAKH, A.; NEJI, R. Modeling and simulation of electrical vehicle in VHDL-AMS. In: IEEE INTERNATIONAL CONFERENCE ON ELECTRONICS, CIRCUITS, AND SYSTEMS- ICECS, 16., 2009, Yasmine Hammamet. **Proceedings...** Yasmine Hammamet: IEEE, 2009. p. 908-911.

JESUS CHAVEZ, J. de; RAMIREZ, A.I.; DINAVAH, V.; IRAVANI, R.; MARTINEZ, J.A.; JATSKEVICH, J.; CHANG, G.W. Interfacing Techniques for Time-Domain and Frequency-Domain Simulation Methods. **IEEE Transactions on Power Delivery**, New York, v. 25, n. 3, p. 1796-1807, 2010.

JIN-HONG, J.; JONG-YUL, K.; HAK-MAN, K. SEUL-KI, K.; CHANGHEE, C.; JANG-MOK, K.; JONG-BO, A.; KEO-YAMG, N. Development of hardware in-the-loop simulation system for testing operation and control functions of microgrid. **IEEE Transactions on Power Electronics**, New York, v. 25, n. 12, p. 2919-2929, 2010.

JOSIFOVSKA, S. The father of labview. **IEE Review**, New York, v. 49, n. 9, p. 30-33, 2003.

JOVANOVIĆ, S.; POURE, P. Design of power electronic digital controller based on FPGA/SOC using VHDL-AMS language. In: IEEE INTERNATIONAL SYMPOSIUM ON INDUSTRIAL ELECTRONICS- ISIE, 16., 2007, Vigo. **Proceedings...** Vigo: IEEE, 2007. p. 2301-2306.

KAZMIERSKI, T. A formal description of VHDL-AMS analogue systems. In: DESIGN, AUTOMATION AND TEST IN EUROPE, 1., 1998, Paris. **Proceedings...** Paris: IEEE, 1998. p. 916-920.

KAZMIERSKI, T. J.; HAMID, F. A. Analogue integrated circuit synthesis from VHDL-AMS behavioural specifications. In: INTERNATIONAL CONFERENCE ON MICROELECTRONICS- MIEL, 23., 2002, Nis. **Proceedings...** Nis: IEEE, 2002. p. 585-588.

KERSTING, W. H. Radial distribution test feeders. Power Systems. **IEEE Transactions on Power Systems**, New York, v. 6, n. 3, p. 975-985, 1991.

KRAEMER, M.; DRAGOMIRESCU, D.; PUYAL, V.; PLANA, R. A VHDL-AMS model of RF MEMS capacitive shunt switches. In: INTERNATIONAL SEMICONDUCTOR CONFERENCE- CAS, 15., 2009, Sinaia. **Proceedings...** Sinaia: IEEE, 2009. p. 307-310.

KUO YEA, Z.; YANCEY JERRY, W. Métodos e sistemas para modelar comportamento concorrente. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetacgi%2FPTO%2Fsrchnu.htm&r=1&f=G&l=50&s1=7603656.PN.&OS=PN/7603656&RS=PN/7603656>>. Acesso em: 20 set. 2012.

LASSETER, R. H.; ETO, J. H.; SCHENKMAN, B.; STEVENS, J.; VOLLKOMMER, H.; KLAPP, D.; LINTON, E.; HURTADO, H.; ROY, J. CERTS Microgrid laboratory test bed. **IEEE Transactions on Power Delivery**, New York, v. 26, n. 1, p. 325-332, 2011.

LIMA, A. de. **Reconfigurabilidade dinâmica e remota de FPGAs**. 2002. 167 f. Dissertação (Mestrado em Engenharia Elétrica) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Paulo, 2002.

LOK-FU, P.; FARUQUE, M. O.; XIN NIE; DINAVAH, V. A versatile cluster-based real-time digital simulator for power engineering research. **IEEE Transactions on Systems**, New York, v. 21, n. 2, p. 455-465, 2006.

MACHACA, W. **Modelamento em VHDL de um Sistema de Simulação de Circuitos**. 1994. 118 f. Dissertação (Mestrado em Engenharia Elétrica) – Faculdade de Engenharia, Universidade Estadual de Campinas, Campinas, 1994.

MARTIN, E.; FERRER, C. DC DC converter modelling with VHDL-AMS. In: IEEE INTERNATIONAL CONFERENCE ON INDUSTRIAL TECHNOLOGY, 5., 2003, Maribor. **Proceedings...** Maribor: IEEE, 2003. p. 766-771.

MATAR, M.; IRAVANI, R. Massively Parallel Implementation of AC Machine Models for FPGA-Based Real-Time Simulation of Electromagnetic Transients. **IEEE Transactions on Power Delivery**, New York, v. 26, n. 2, p. 830-840, 2011.

MCCLURE C. L. Top-down, bottom-up, and structured programming. **IEEE Transactions on Software Engineering**, New York, v. SE-1, n. 4, p. 397-403, 1975.

MCDERMOTT, T. E.; JUCHEM, R.; DEVARAJAN, D. Distribution Feeder and Induction Motor Modeling with VHDL-AMS. In: IEEE PES TRANSMISSION AND DISTRIBUTION CONFERENCE AND EXHIBITION 2005/2006, 8., 2006, Dallas. **Proceedings...** Dallas: IEEE, 2006. p. 141-146.

MCGRANAGHAN, M. F. Quantifying Reliability and Service Quality for Distribution Systems. **IEEE Transactions on Industry Applications**, New York, v. 43, n. 1, p. 188-195, 2007.

MOSER, E. Automotive model exchange using VHDL-AMS: a benchmark. In: IEEE INTERNATIONAL SYMPOSIUM ON COMPUTER-AIDED CONTROL SYSTEM DESIGN, 4., 1996, Dearborn. **Proceedings...** Dearborn: IEEE, 1996. p. 276-281.

NAGASAMY, V.; BERRY, N.; DANGELO, C. Specification, planning, and synthesis in a VHDL design environment. **IEEE Design & Test of Computers**, New York, v. 9, n. 2, p. 58-68, 1992.

NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION – NEMA. **Programmable Controllers (PLC), Part 1: General Information**. [S.l: s.n], 2005. Disponível em: <http://www.nema.org/Standards/Pages/Programmable-Controllers-Part-1-General-Information.aspx>

NIANNIAN, C.; XUFENG, X.; MITRA, J. A hierarchical multi-agent control scheme for a black start-capable microgrid. In: POWER AND ENERGY SOCIETY GENERAL MEETING- PES, 9., 2011, San Diego. **Proceedings...** San Diego: IEEE, 2011. p. 1-7.

OLIVEIRA, M. de. **Metodologias de teste para FPGAs (Field Programmable Gate Arrays) integradas em sistemas reconfiguráveis**. 2003. 190 f. Dissertação (Mestrado em Engenharia Elétrica) – Faculdade de Engenharia, Universidade do Porto, Porto, 2003.

OLIVEIRA, L. C. O.; E MELO, G. A.; SOUZA, J. B.; CANESIN, C. A.; BONATTO, B. D.; BELCHIOR, F. N.; OLIVEIRA, M.; MERTENS, E. A. Harmonic propagation analysis in electric energy distribution systems. In: INTERNATIONAL CONFERENCE ON ELECTRICAL POWER QUALITY AND UTILISATION- EPQU, 11., 2011, Lisbon. **Proceedings...** Lisbon: IEEE, 2011. p. 1-6.

O'RIORDAN, D. J.; TRIHY, R. Método de depuração de modelos comportamentais de sinal misto e analógicos durante a simulação. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahml%2FPTO%2Fsrchnu.htm&r=1&f=G&l=50&s1=7085700.PN.&OS=PN/7085700&RS=PN/7085700>>. Acesso em: 15 ago. 2012.

RIESGO T.; TORROJA Y.; DE LA TORRE E. Design methodologies based on hardware description languages. **IEEE Transactions on Industrial Electronics**, New York, v. 46, n. 1, p. 3-12, 1999.

ROYTELMAN, I.; WEE, B. K.; LUGTU, R. L. Volt/var control algorithm for modern distribution management system. **IEEE Transactions on Power Systems**, New York, v. 10, n. 3, p. 1454-1460, 1995.

SCHWIEGELSHOHN, F.; HUBNER, M. Design of an attention detection system on the Zynq-7000 SoC. In: INTERNATIONAL CONFERENCE ON RECONFIGURABLE COMPUTING AND FPGAS- RECONFIG, 9., 2014, Cancun. **Proceedings...** Cancun: IEEE, 2014. p. 1-6.

SHOUFAN, A. A compact course on VHDL-AMS. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS- ISCAS, 25., 2010, Paris. **Proceedings...** Paris: IEEE, 2010. p. 89-92.

SHOUFAN, A.; HUSS, S. Understanding physical models in VHDL-AMS. In: FORUM ON SPECIFICATION & DESIGN LANGUAGES- FDL, 2., 2009, Sophia Antipolis. **Proceedings...** Sophia Antipolis: IEEE, 2009. p. 1-4.

SILVA, T. da. **Ambiente computacional para projetos de sistemas com tecnologia mista**. 2009. 182 f. Dissertação (Mestrado em Engenharia Elétrica) – Faculdade de Engenharia, Universidade Estadual Paulista, Ilha Solteira, 2009.

SLUSARCZYK, K.; NAPIERALSKI, A. Nonlinear Model of Electrokinetic Phenomena in VHDL-AMS. In: INTERNATIONAL CONFERENCE ON MIXED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS- MIXDES, 14., 2007, Ciechocinek. **Proceedings...** Ciechocinek: IEEE, 2007. p. 460-463.

TRIMBERGER S. M.; MOORE J. J. FPGA Security: motivations, features, and applications. **IEEE Spectrum**, New York, v. 102, n. 8, p. 1248-1265, 2014.

UENO, S.; TSUJI, K.; ABE, T.; HIGUCHI, T.; SHIGEMATSU, K. Fundamental modeling of vehicle power network system using VHDL-AMS. In: INTERNATIONAL SYMPOSIUM ON POWER ELECTRONICS ELECTRICAL DRIVES AUTOMATION AND MOTION-SPEEDAM, 4., 2010, Pisa. **Proceedings...** Pisa: IEEE, 2010. p. 1046-1051.

VANDOORN, T.L.; MEERSMAN, B.; DEGROOTE, L.; RENDERS, B.; VANDEVELDE, L. A Control Strategy for Islanded Microgrids with DC-Link Voltage Control. **IEEE Transactions on Power Delivery**, New York, v. 26, n. 2, p. 703-713, 2011.

VANEGAS, M. **Arquitetura baseada em tecnologia FPGA para a estimação e análise de informação de fluxo óptico em tempo real**. 2010. 125 f. Dissertação (Mestrado em Engenharia Elétrica) – Departamento de Arquitetura e Tecnologia de Computadores, Universidade de Granada, Granada, 2010.

VENKATA, R. **Simulação digital em tempo real de circuitos de alimentação chaveadas**. 2000. 187 f. These (Doctor in Engineering Electrical) – University of Toronto, Toronto, 2000.

VENKATARAMANAN, G.; MARNAY, C. A larger role for microgrids. **IEEE Power and Energy Magazine**, New York, v. 6, n. 3, p. 78-82, 2008.

VORBACH, M.; BECKER, J.; WEINHARDT, M.; BAUMGARTE, V.; MAY, F. Dispositivo e método de processamento de dados. **Patente da World Intellectual Property Organization (WIPO)**. International and National Patent Collections. Disponível em: <<http://patentscope.wipo.int/search/en/detail.jsf?docId=WO2004015568&recNum=1&docAn=EP2003008080&queryString=PCT/EP2003/008080&maxRec=1>>. Acesso em: 25 out. 2012.

VREUGDENHIL, G. J.; CHRISTEN, E. T.; VLACH, M. Associação dinâmica de equações para incógnitas durante simulações de sistemas descritos por linguagens de descrição de hardware. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnethtml%2FPTO%2Fsrchnu.htm&r=1&f=G&l=50&s1=6975972.PN.&OS=PN/6975972&RS=PN/6975972>>. Acesso em: 5 out. 2012.

WALTER, D. **Verificação de circuitos analógicos e de sinal misto usando métodos simbólicos**. 2007. 167 f. Tese (Doutorado em Engenharia Elétrica) – Faculdade de Engenharia, University of Utah, Utah, 2007.

WASHINGTON, C.; DOLMAN, J. Creating next generation HIL simulators with FPGA technology. In: IEEE AUTOTESTCON, 24., 2010, Orlando. **Proceedings...** Orlando: IEEE, 2010. p. 1-6.

WILLIS, J. C. Geração semiautomática de modelos comportamentais de valores contínuos usando a sondagem iterativa de um dispositivo ou modelo de componente existente. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnethtml%2FPTO%2Fsrchnu.htm&r=1&f=G&l=50&s1=7328195.PN.&OS=PN/7328195&RS=PN/7328195>>. Acesso em: 20 ago. 2012.

WILLIS, J. C. Simulação acelerada de sistemas de equações diferenciais que tem comportamento contínuo. **Patente dos Estados Unidos**. Escritório de Marcas e Patentes de EUA. Disponível em: <<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnethtml%2FPTO%2Fsrchnu.htm&r=1&f=G&l=50&s1=7539602.PN.&OS=PN/7539602&RS=PN/7539602>>. Acesso em: 10 set. 2012.

WILLIS, J. C.; JOHNSON, J. A.; BETCHER, R. A. Simulação de projetos usando lógica reconfigurável. **Patente da World Intellectual Property Organization (WIPO)**. International and National Patent Collections. Disponível em: <<http://patentscope.wipo.int/search/en/detail.jsf?docId=WO2003046776&recNum=1&docAn=US2002037352&queryString=PCT/US02/37352&maxRec=1>>. Acesso em: 20 out. 2012.

XILINX. **Zynq-7000 all programmable SoC first generation architecture**. [S.l: s.n], 2014. Disponível em: <http://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf>. Acesso em: 28 fev. 2015.

YANG, D.; SHI, X.; YANG, D.; YU, L. Distributed transmission-line behavioral modeling and analyzing using VHDL-AMS. In: MULTICONFERENCE ON COMPUTATIONAL ENGINEERING IN SYSTEMS APPLICATIONS- IMACS, 1., 2006, Beijing. **Proceedings...** Beijing: IEEE, 2006. p. 500-503.

YAO, L.; ISKANDER, R.; LOUERAT, M. M. Modeling, design and verification platform using SystemC AMS. In: INTERNATIONAL SYMPOSIUM ON QUALITY ELECTRONIC DESIGN- ISQED, 15., 2014, Santa Clara. **Proceedings...** Santa Clara: IEEE, 2014. p. 39-46.

YING-YU, T.; HAU-JEAN, H. FPGA realization of space-vector PWM control IC for three-phase PWM inverters. **IEEE Transactions on Power Electronics**, New York, v. 12, n. 6, p. 953-963, 1997.

YOUNG-SU, K.; CHONG-MIN, K. Performance-driven event-based synchronization for multi-FPGA simulation accelerator with event time-multiplexing bus. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, v. 24, n. 9, p. 1444-1456, 2005.

YUAN, C.; DINAVAHI, V. FPGA-Based Real-Time EMTP. **IEEE Transactions on Power Delivery**, New York, v. 24, n. 2, p. 892-902, 2009.

YUKISHIMA, C. **Desenvolvimento de Ferramenta de Engenharia para Aplicação de Simulação em Tempo Real em Plantas Industriais Automatizadas “FASTR”**. 2006. 201 f. Tese (Mestrado em Engenharia Elétrica) – Escola Politécnica da Universidade de São Paulo, Universidade de São Paulo, São Paulo, 2006.

ZHANG, H.; SIEBERT, K.; FREI, S.; WENZEL, T.; MICKISCH, W. Multiconductor transmission line modeling with VHDL-AMS for EMC applications. In: IEEE INTERNATIONAL SYMPOSIUM ON ELECTROMAGNETIC COMPATIBILITY- EMC, 26., 2008, Detroit. **Proceedings...** Detroit: IEEE, 2008. p. 1-6.

ZORZI, M.; FRANZE, F.; SPECIALE, N. Construction of VHDL-AMS simulator in Matlab. In: IEEE INTERNATIONAL BEHAVIORAL MODELING AND SIMULATION WORKSHOP- BMAS, 7., 2003, San Jose. **Proceedings...** San Jose: IEEE, 2003. p. 113-117.

PRODUÇÃO DE ARTIGOS CIENTÍFICOS

A.1 PUBLICAÇÕES EM CONGRESSOS:

1. IBARRA HERNANDEZ FRANK; CANESIN CARLOS ALBERTO; ZAMORA RAMON; SRIVASTAVA ANURAG K. Active power management in multiple microgrids using a multi-agent system with JADE. In: IEEE/IAS INTERNATIONAL CONFERENCE ON INDUSTRY APPLICATIONS (INDUSCON), 11., 2014, Juiz de Fora. **Proceedings...** Juiz de Fora: IEEE, 2014. p. 1-8.
2. IBARRA HERNANDEZ FRANK; CANESIN CARLOS ALBERTO; ZAMORA RAMON; MARTINA FRANSISKA; SRIVASTAVA ANURAG K. Energy management and control for islanded microgrid using multi-agents. In: NORTH AMERICAN POWER SYMPOSIUM (NAPS), 11., 2013, Manhattan. **Proceedings...** Manhattan: IEEE, 2013. p. 1-6.
3. IBARRA HERNANDEZ FRANK; CANESIN CARLOS A. Electrical Power Distribution System modeling with VHDL-AMS for the construction of a Real-Time Digital Simulator using FPGAS devices. In: IEEE/IAS INTERNATIONAL CONFERENCE ON INDUSTRY APPLICATIONS (INDUSCON), 10., 2012, Fortaleza. **Proceedings...** Fortaleza: IEEE, 2012. p. 1-7.

A.2 ARTIGOS ACEITOS PARA PUBLICAÇÃO:

1. IBARRA HERNANDEZ, FRANK. A. I.; CANESIN C. A. Modelação de sistema de distribuição de energia elétrica com carregamento não linear, utilizando a linguagem VHDL-AMS. In: CONGRESSO BRASILEIRO DE QUALIDADE DE ENERGIA-CBQEE, 2015, Campina Grande. **Anais...** Campina Grande: Sociedade Brasileira de Qualidade de Energia, 2015. p. 1-7.

A.3 ARTIGO PRELIMINARMENTE ACEITO EM REVISTAS (JOURNALS):

1. HERNANDEZ, F. A. I.; CANESIN C. A. Real-Time simulation for electrical distribution feeders through hardware description languages, sob revisão na **IEEE Transactions on Smart Grid.**, 2015. p. 1-12.

Apêndice B

DESENVOLVIMENTO COM RTDS EM PARCERIA COM A WSU – ESTADOS UNIDOS DE AMÉRICA DO NORTE

Tendo em consideração a figura 2 (Esquema geral da arquitetura proposta para simulação em “tempo real” e controle de alimentadores de distribuição de energia elétrica) do Capítulo 1, podemos observar que esta pesquisa de doutorado pode ser dividida em sete conceitos-chave:

- a. Simulação digital em tempo real (***Real Time Digital Simulation***),
- b. Alimentadores de distribuição de energia elétrica (***Electrical Distribution Feeders***),
- c. Dispositivos FPGAs (***Field Programmable Gate Array***),
- d. Linguagem de descrição de hardware (***Hardware Description Language - HDL***),
- e. Teste de equipamento em malha fechada (***Hardware in the Loop - HIL***),
- f. Micro-redes com Geração Distribuída (***Microgrids with Distributed Generation***) e,
- g. Redes elétricas inteligentes (***Smart-Grids***),

Para estudar aprofundadamente cinco dos anteriores sete conceitos, foi desenvolvida uma parceria com a Universidade Estadual de Washington (WSU) nos Estados Unidos, na qual se teve a oportunidade de afundar os conceitos: um (1), dois (2), cinco (5), seis (6) e sete (7).

Neste apêndice se descreve a interação e o trabalho com um Simulador digital em tempo real comercial (RTDS®), desenvolvido para um estudo de caso de concessionária de distribuição, fornecido pela empresa canadense RTDS Technologies e, que pertence à WSU.

Com o desenvolvimento da parceria conjunta entre o nosso Laboratório de Eletrônica de Potência (LEP) da UNESP – FE/IS e o Laboratório de Investigação, Pesquisa e Demonstração de Redes elétricas inteligentes (SGDRIL¹) na WSU na cidade de Pullman – WA/EUA (Ver figura 87), este apêndice esta composto pelos seguintes itens: 1) Especificações técnicas relevantes do RTDS utilizado, 2) Resultados experimentais da simulação e modelação em tempo real e, 3) Códigos gerados em JADE para o Gerenciamento da Potência Ativa em Várias Micro-redes.

¹ O SGDRIL é um laboratório reconhecido na Faculdade de Engenharia Elétrica e Ciência da Computação da Washington State University (WSU), na qual o Professor Dr. Anurag K. Srivastava é o diretor do laboratório e, líder pesquisador no campo da simulação e modelagem de sistemas de potência em tempo real nos Estados Unidos de America (EUA).

Figura 87- Laboratório de investigação, pesquisa e demonstração de redes elétricas inteligentes - **Smart Grid Demonstration and Research Investigation Lab** (SGDRIL).



Fonte: Arquivos do SGDRIL.

B.1 ESPECIFICAÇÕES TÉCNICAS RELEVANTES DO RTDS® UTILIZADO

Tomando em consideração os manuais técnicos atualizados em dezembro de 2012 do Simulador Digital em Tempo Real (RTDS), fornecido pela empresa canadense RTDS Technologies, podemos extrair as seguintes especificações técnicas relevantes do RTDS utilizado na parceria com a WSU, úteis para o desenvolvimento desta pesquisa:

- a. O Simulador Digital em Tempo Real (RTDS) é um computador de propósito especial projetado para estudar em tempo real o fenômeno dos transitórios eletromagnéticos (EMT). O RTDS constitui-se numa ferramenta utilizada para analisar sistemas elétricos de alta e média tensão, tanto DC como AC. Especificamente o RTDS é usado para avaliar equipamentos de proteção e controle, analisando o desempenho da rede do sistema elétrico e, também para formar aos operadores, técnicos e estudantes.
- b. A arquitetura do hardware de processamento paralelo totalmente digital do Simulador RTDS é capaz de simular redes complexas utilizando um passo de tempo de simulação de 50 microssegundos. O hardware é agrupado em unidades modulares chamadas armários (*racks*), que permitem uma fácil expansão da capacidade de computação do simulador conforme seja necessário. Com a rápida velocidade de computo do hardware do RTDS, podem ser realizados estudos analíticos muito mais rapidamente do que programas de simulação do tipo EMT.

- c. Como o RTDS é especialmente projetado para simulações de transitórios eletromagnéticos (EMT) nos sistemas elétricos de potência em tempo real, ele funciona de maneira contínua em tempo real, fornecendo resultados precisos dentro de uma faixa de frequências que variam desde zero Hz (DC) até três kHz. Esta gama oferece maior profundidade de análise em comparação com os programas tradicionais de fluxo de carga ou de estabilidade, que analisam os fenômenos dentro de uma faixa de frequência limitada.
- d. O RTDS é composto por hardware e software, os quais em resumo, são projetados especialmente assim:
 - O hardware do RTDS está baseado num Processador Digital de Sinal (DSP) e um Computador com um Conjunto Reduzido de Instruções (RISC), os quais utilizam técnicas avançadas de processamento paralelo, a fim de alcançar as velocidades de computação necessárias para manter a operação contínua em tempo real. É importante notar que em arquitetura de computadores, RISC é um tipo de projeto de CPU (A Unidade Central de Processamento-CPU ou processador é o principal componente do computador e outros dispositivos programáveis, que interpreta as instruções em programas e, além disso, esta unidade processa múltiplos dados), geralmente usado em microprocessadores ou microcontroladores com as seguintes características principais:
 - ✓ Tem instruções de tamanho fixo e apresentadas num número reduzido de formatos.
 - ✓ Só as instruções de carga e armazenamento acessam à memória de dados.
 - ✓ Além do hardware anterior, o RTDS tem um cartão GTWIF (Giga Transceiver Workstation InterFace), ou seja, um cartão de interface entre a estação de trabalho e o giga transceptor (transmissor-receptor). O GTWIF é um processador cuja função principal é estabelecer a comunicação entre o RTDS e o computador hospedeiro (que executa o software RSCAD). Isso é feito usando as comunicações baseadas em tecnologia Ethernet TCP/IP (10/100baseT). Os protocolos de Internet TCP/IP são: Protocolo de Controle de Transmissão (TCP) e Protocolo de Internet (IP), os quais são a base da Internet, e são usados para ligar computadores que utilizam diferentes sistemas operacionais, incluindo PCs, minicomputadores e computadores de grande porte em redes de área local (LAN) e de área ampla (WAN). Por outro lado o 100baseT é o padrão mais comum da Ethernet de 100 megabit também chamada Fast Ethernet que é um termo para vários padrões da Ethernet que levam o tráfego de dados à taxa nominal de 100 Mbit/s. Cada rack do RTDS inclui um cartão de GTWIF.
 - O software do RTDS inclui modelos precisos de componentes do sistema elétrico de potência, necessários para representar muitos dos complexos elementos que compõem os sistemas elétricos de potência físicos (ou reais). A solução geral de rede utilizada na técnica do RTDS é baseada na análise nodal. Os algoritmos de solução subjacentes são aqueles introduzidos no clássico artigo “Solução para Computador Digital de Transitórios Eletromagnéticos em redes monofásicas e polifásicas” pelo H.W. Dommel. O algoritmo de solução do Dommel é utilizado em praticamente

todos os programas de simulação digitais concebidos para o estudo de transientes eletromagnéticos. O software do RTDS também inclui uma poderosa e fácil de usar interface gráfica do usuário (**Graphical User Interface** - GUI), conhecida como RSCAD, através da qual o usuário é capaz de construir, executar e analisar os casos de simulação. RSCAD contém uma biblioteca de componentes do sistema de controle e do sistema elétrico que permite desenhar o sistema simulado num entorno CAD (**Computer-Aided Design**), ou seja, projeto assistido por computador. Neste contexto a GUI do RTDS, chamada RSCAD, permite desenhar e projetar qualquer sistema elétrico de potência ou controle, segundo as restrições do hardware (número de armários). Em termos gerais, um RTDS pode ter um ou mais armários (**racks**) os quais, pela escalabilidade do RTDS, podem aumentar o desempenho deste com o acréscimo de hardware (**racks**), proporcionalmente à capacidade acrescida. Duas soluções de rede (subsistemas) são permitidas por um rack e cada solução de rede pode resolver até 21 nós. Como em nosso caso o RTDS utilizado no laboratório da WSU tem um (1) Rack, então se pode resolver um sistema elétrico de até 42 nós com um máximo de dois subsistemas que conformam este sistema elétrico.

- Ademais das capacidades de software e hardware oferecidas pelo RTDS, com a sua capacidade de simulação em tempo real, ele se pode conectar diretamente a uns dispositivos de controle e/ou relés de proteção reais para realizar testes destes equipamentos em malha fechada (Hardware in the Loop - HIL). O HIL é uma técnica usada para desenvolver e testar sistemas que tenham um controlador. A meta é verificar e certificar a funcionalidade, rendimento, qualidade e segurança do programa de software do controlador. Para fazer isto o comportamento dinâmico do sistema físico real é convertido num modelo de software que roda no Simulador Digital em Tempo Real (através do RSCAD). O modelo simula exatamente o comportamento do sistema físico, em condições de estado estável e transitório. O controlador a testar é conectado às entradas e saídas do simulador (digitais e/ou analógicas), o qual emula os sensores e atuadores reais. Recriando a realidade, o controlador do equipamento a testar é "enganado" em acreditar que é ligado ao sistema físico real. Então se torna possível a obtenção de toda a flexibilidade necessária para testar o controlador sob quaisquer condições de funcionamento.
- Ao contrário dos simuladores analógicos, os quais os sinais de saída são contínuos com relação ao tempo, os simuladores digitais computam o estado do modelo de um sistema de potência, somente em instantes discretos no tempo. O tempo entre esses instantes discretos é referido como o passo de tempo da simulação (Δt). Muitas centenas de milhares de cálculos devem ser realizadas durante cada passo de tempo, a fim de calcular o estado do sistema nesse instante.
- A classe de transitórios temporários estudados para o qual o RTDS é mais frequentemente usado requer um passo de tempo (Δt) na ordem de 50 até 60 μsec (resposta de frequência exata para aproximadamente 3000 Hz). Por definição, a fim de operar em tempo real é necessário um passo

de tempo da simulação de 50 μ sec, ou seja, que todos os cálculos para a solução de sistema sejam completados em menos de 50 microssegundos do tempo real.

- A fim de alcançar os passos de tempo necessários e, manter as taxas de computação adequadas para a operação em tempo real, muitos processadores de alta velocidade operando em paralelo são utilizados pelo RTDS. Em cada armário (rack) do RTDS são instalados dois tipos de cartões de processador:
 - ✓ O Cartão de Processador Triplo (3PC) que contém três processadores digitais de sinais analógicos (dispositivos ADSP 21062). A velocidade de relógio do ADSP 21062 é de 40 MHz.
 - ✓ O Cartão de Processador RISC (Computador com um Conjunto Reduzido de Instruções), chamado RPC, que contém dois processadores PowerPC 750CXe RISC os quais operam com uma velocidade de relógio de 600 MHz.
- Um armário (rack) do RTDS contém 22 placas de circuito impresso. As configurações típicas de um rack incluem o seguinte número de cartões de processador:
 - ✓ Vinte (20) cartões de Processador Triplo (3PC).
 - ✓ Um cartão de Processador RPC.
 - ✓ Um cartão GTWIF.
- Através da interface gráfica de usuário do simulador digital em tempo real (RTDS) o software deste simulador traz modelos precisos de componentes do sistema elétrico de potência, prontos para o usuário, os quais são:
 - ✓ Fonte trifásica AC de tensão: sem harmônicos, com harmônicos (até 4 harmônicos), com frequência e/ou modulação da magnitude.
 - ✓ Fonte trifásica DC de tensão.
 - ✓ Fonte trifásica de tensão com a entrada de controles para definir o tipo de onda.
 - ✓ Fonte monofásica AC de tensão: sem harmônicos, com harmônicos (até 4 harmônicos), com frequência e/ou modulação da magnitude.
 - ✓ Transformadores trifásicos monofásicos.
 - ✓ Transformadores com falhas internas.
 - ✓ Transformador UMEC (Circuito Equivalente Magnético Unificado): baseado principalmente na geometria do núcleo.
 - ✓ Linhas de transmissão e de distribuição.
 - ✓ Máquinas elétricas monofásicas e trifásicas.
 - ✓ Transdutores de medição: transformadores de tensão capacitivos (CVT) e transformadores de corrente (TC).
 - ✓ Compensação série controlada por tiristores.
 - ✓ Matriz Fotovoltaica (PV).
 - ✓ Bateria de íons de lítio.

- ✓ Entre outros de proposito mais especifico.
- Além do anteriormente descrito, o RTDS® utilizado no SGDRIL da **Washington State University** tem outros equipamentos ancilares que estão direcionados à pesquisa em redes elétricas inteligentes (**Smart Grids**), os quais são:
 - ✓ Relógio GPS: Fornece o sinal de sincronização de tempo para todos os dispositivos sincrofasores, independentemente da sua localização geográfica.
 - ✓ Relés associados às Unidades de Medição Fasorial (PMU) e ao Gravador de falha perturbação (DFR).
 - ✓ Amplificadores de Corrente e Tensão.
 - ✓ Unidades de Medição Fasorial (PMU).
 - ✓ Concentrador de Dados Fasoriais (PDC).
 - ✓ Controlador de Automação em Tempo Real (SEL-3530).
 - ✓ Computador de automação de subestações (SEL-3354).

B.2 RESULTADOS EXPERIMENTAIS DA SIMULAÇÃO E MODELAÇÃO EM TEMPO REAL COM UM SIMULADOR DIGITAL EM TEMPO REAL COMERCIAL (RTDS®)

B.2.1 Gerenciamento da Energia e Controle para uma Micro-Rede Ilhada Usando Multi-Agentes

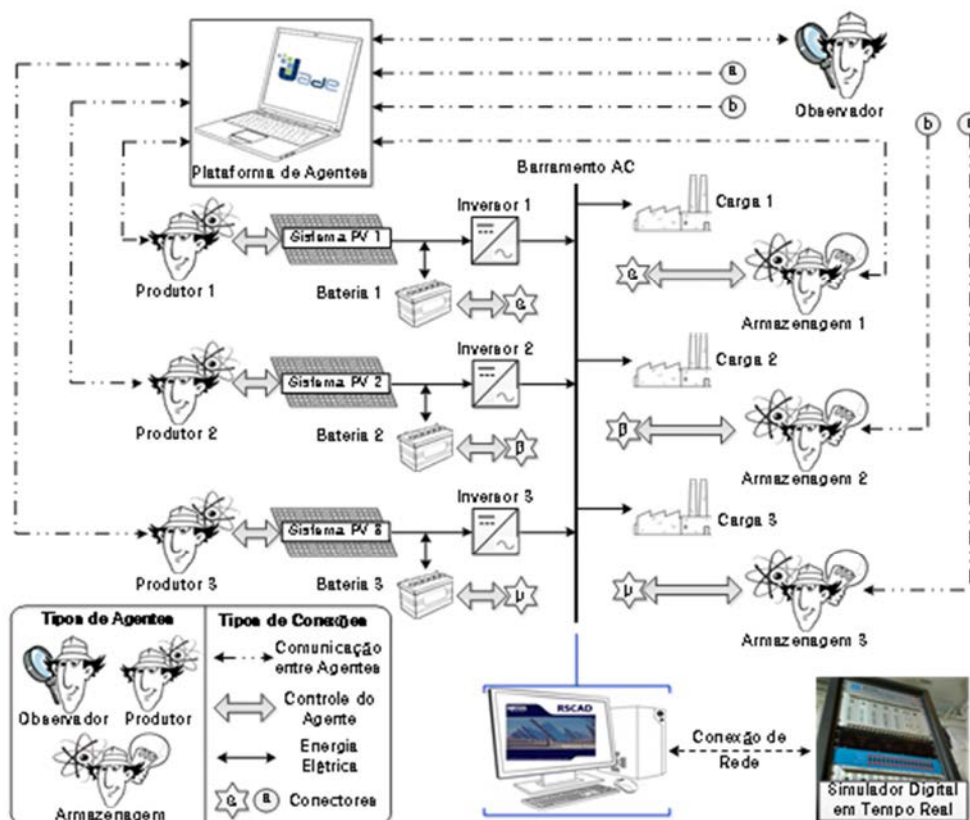
Segundo o trabalho publicado (IBARRA HERNANDEZ et al., 2013), a figura 88 mostra o diagrama da utilização de um Simulador Digital em Tempo Real - RTDS comercial e um sistema multi-agentes, o qual foi desenvolvido em JADE (**Java Agent DEvelopment Framework**), para o controle e gerenciamento da energia elétrica, com o intuito de regular as micro-fontes para atingir as suas capacidades máximas. JADE é uma camada de software dentro de um único aplicativo que permite que diferentes aspectos do programa trabalhem em conjunto com o aplicativo, conhecido como middleware. Este único aplicativo é o sistema operacional denominado máquina virtual de Java (**Java Virtual Machine-JVM**). O funcionamento do middleware JADE é baseado no desenvolvimento de códigos para os agentes escritos em Java. Os agentes se podem estender por vários computadores ou estar num único computador, mas para a implementação, o código para o envio e recebimento das mensagens é o mesmo.

O anterior gerenciamento da energia elétrica foi feito com o objetivo de melhorar o funcionamento operacional de uma micro-rede ilhada formada exclusivamente por fontes de energia fotovoltaica, para o qual foi feita a modelação e simulação digital em tempo real da micro-rede com os seguintes elementos: 3 sistemas fotovoltaicos de 60 kW cada um (com sistema de armazenamento através de baterias), 3 alimentadores radiais com a carga variável “sensível” entre 0 e 100 kW, um

computador com o software em JADE e, um simulador digital em tempo real comercial (com a sua interface de usuário – RSCAD). Todas as cargas combinam cargas puramente resistivas e máquinas rotativas. Os resultados da simulação mostraram que:

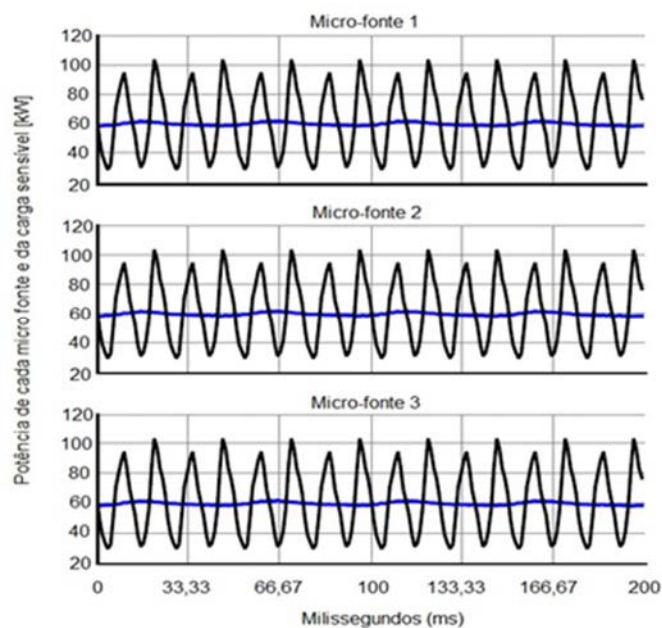
- O DSM ou sistema de gerenciamento de energia e controle é capaz de regular as micro-fontes fotovoltaicas para atingir suas capacidades máximas.
- Como a natureza das cargas é oscilante ($60 \text{ kW} \pm 40 \text{ kW}$ a cada 6,67 milissegundos) e, a potência destas em alguns momentos na simulação é maior (ou menor) que a capacidade dos sistemas fotovoltaicos da micro-rede, o sistema de armazenamento de energia (baterias) equilibra a falta (ou excesso) da energia ministrada a todas as cargas pelos painéis solares (Figura 89). Os resultados da experimentação mostraram que o carregamento (ou descarregamento) das baterias flutua com diferentes taxas (aproximadamente desde 74,386% até 74,366% em 0,2 segundos), para compensar as flutuações das cargas (Figura 89). Além disso, considerando que a tensão das baterias é ajustada em 400 V nominalmente, quando a taxa de carregamento (ou descarregamento) destas muda; em forma periódica, a tensão da bateria aumenta (ou diminui) abruptamente e, depois volta ao valor nominal (400 V), aproximadamente $\pm 200 \text{ V}$ a cada 66,7 milissegundos, em 200 milissegundos (Figura 90). Com a mesma análise, considerando que a corrente das baterias em condições de não carregamento (ou descarregamento) deve ser zero, quando a taxa de carregamento (ou descarregamento) destas muda, em forma periódica, a corrente da bateria diminui (ou aumenta) abruptamente e, depois volta ao valor de não carregamento (ou descarregamento), ou seja, 0,0 A. A mudança da corrente da bateria é de aproximadamente $\pm 70 \text{ A}$, a cada 66,7 milissegundos, em 200 milissegundos (Figura 88). Em conclusão, quando a taxa de descarga da bateria é elevada, a tensão desta diminui, enquanto que a corrente aumenta.
- As micro-fontes, as cargas e as baterias são bem coordenadas pelo sistema de gerenciamento e controle desenvolvido através de sistemas multi-agentes em JADE, no qual o excesso (ou defeito) de energia a partir das e micro-fontes é armazenado (ou compensado) em (ou pelas) baterias.
- Nesta modelação e simulação em tempo real foi mostrada a viabilidade e utilidade das aplicações dos sistemas multi-agentes (MAS) na gestão de energia e controle de micro-redes ilhadas com alta penetração de sistemas fotovoltaicos.
- O resultado dos estudos de simulação demonstra a eficácia do método de controle e a técnica proposta e, mostra a possibilidade de funcionamento autónomo incorporando a operação de uma micro-rede ilhada com um sistema multi-agentes.

Figura 88- Simulação em tempo real de um sistema fotovoltaico numa micro-rede ilhada.



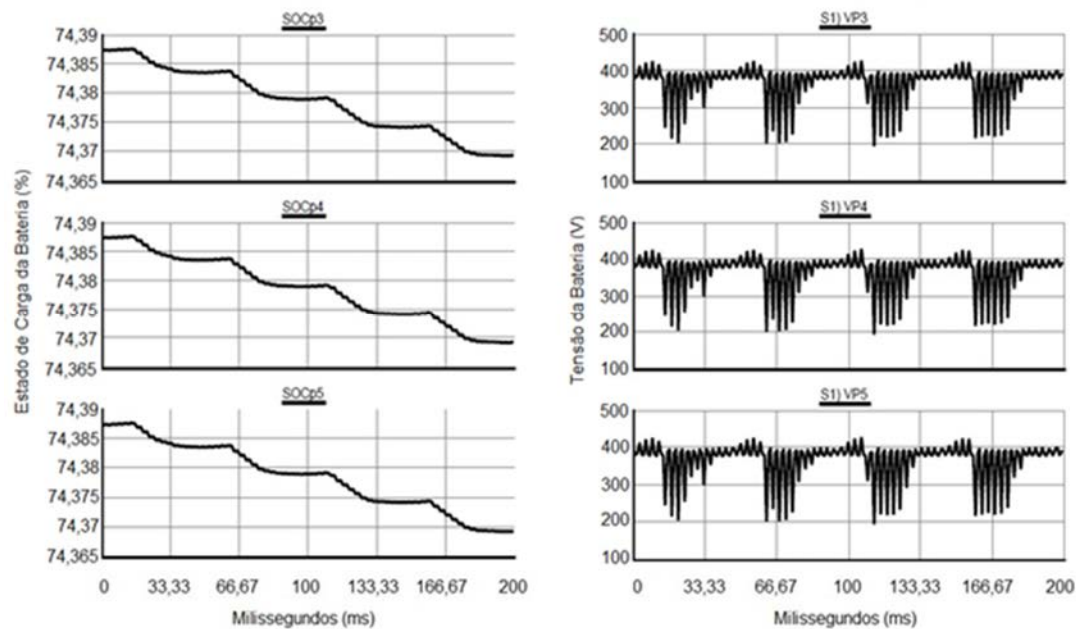
Fonte: (IBARRA HERNANDEZ et al., 2013).

Figura 89- Potência de cada micro fonte (azul) [kW] e da carga sensível (preta) [kW].



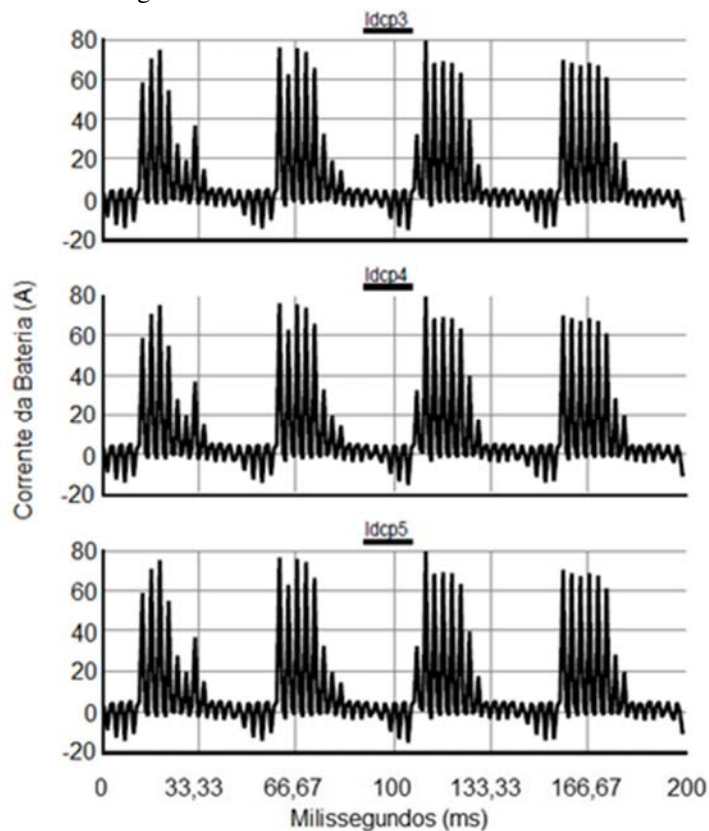
Fonte: (IBARRA HERNANDEZ et al., 2013).

Figura 90- Descarregamento da bateria de cada micro fonte PV e Tensão da Bateria.



Fonte: (IBARRA HERNANDEZ et al., 2013).

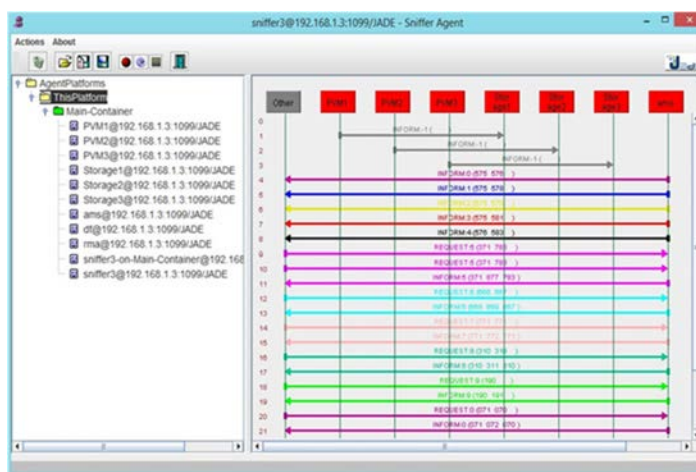
Figura 91- Corrente de descarregamento da Bateria.



Fonte: (IBARRA HERNANDEZ et al., 2013).

- Finalmente, o diagrama *sniffer* para a modelação e implementação do sistema multi-agentes pode ser vista na figura 92. Esta ferramenta é utilizada para documentar as conversações entre os agentes. Os resultados experimentais da simulação e modelação em tempo real mostraram que os agentes produtores (PVM1, PVM2 e PVM3) enviam uma mensagem específica para os agentes de armazenamento (Armazenagem1, Armazenagem2 e Armazenagem3); esta comunicação é a potência máxima que pode receber a bateria para que seja carregada. Nós também podemos ver que o agente do Sistema de Gestão de Agentes (AMS), que é responsável por gerenciar a operação de um agente da plataforma JADE, está em constante comunicação com o agente *Sniffer*, que é responsável para monitorar todos os requisitos dos agentes da plataforma desenvolvida em JADE.

Figura 92- Diagrama sniffer da modelação e implementação do sistema multi-agentes.



Fonte: (IBARRA HERNANDEZ et al., 2013).

B.2.2 Gerenciamento da Potência Ativa em Várias Micro-redes Usando um Sistema Multi-agentes com JADE

Segundo o trabalho publicado (HERNANDEZ et al., 2014) a figura 93 mostra o cenário modelado para o gerenciamento da potência ativa em várias micro-redes, segundo o qual um sistema multi-agentes (MAS) é utilizado para modelar e permitir um gerenciamento da potência ativa em um sistema de múltiplas micro-redes, consistindo em baterias, sistemas fotovoltaicos e micro fontes diesel, em operação ilhada ou conectada à rede. O sistema multi-agente foi desenvolvido em JADE (Java Agent DEvelopment Framework). Todos os seis estudos de caso apresentados neste trabalho foram modelados utilizando JADE (favor ver os códigos no Apêndice C). Os casos foram escolhidos considerando-se cenários possíveis da vida real, tais como: descarregar a bateria (modo boost), exportação de energia a partir da micro-rede, carregamento da bateria (modo buck), importação de energia do gerador diesel, importação de energia da rede elétrica e, parada do sistema elétrico. No entanto, considerando-se que os

resultados para os dois primeiros casos podem demonstrar o desempenho e as características do MAS proposto, este trabalho centra-se nos resultados desses casos. Os resultados demonstraram a efetividade dos algoritmos de controle propostos, necessários para o controle coordenado e gerenciamento de energia, e mostrar a possibilidade de operação built-in autônomo de um microgrid com um sistema multi-agente usando JADE. O trabalho utilizou o conceito da micro-rede CERTS o qual significa: Consorcio para as Soluções Tecnológicas em Confiabilidade Elétrica (Consortium for Electric Reliability Technology Solutions).

TABELA 23-. RESULTADOS DAS SIMULAÇÕES PARA O GERENCIAMENTO DA POTÊNCIA ATIVA EM VÁRIAS MICRO-REDES USANDO UM SISTEMA MULTI-AGENTES.

Casos de Teste Micro-Rede	Agentes	Variáveis Calculadas	Micro-Rede 1	Micro-Rede 2	Micro-Rede 3	Sistema Elétrico Global
Caso 1. Modo de operação: Sistema PV fornece à carga local, descarregando bateria (Modo Boost), importar energia de Diesel e importar energia de Grid.	Calculado pelo Agentes Gerenciadores	Potência do Sistema PV:	100.0 kW	60.0 kW	20.0 kW	N. A.
		Potência do descarregamento da bateria:	77.0 kW	35.0 kW	92.0 kW	N. A.
		Importação de Potência do Diesel:	10.0 kW	55.0 kW	10.0 kW	N. A.
		Importação de Potência da rede:	63.0 kW [Enviado para o Agente GES]	70.0 kW [Enviado para o Agente GES]	28.0 kW [Enviado para o Agente GES]	N. A.
		Potência Fornecida à Carga Local:	250.0 kW	220.0 kW	150.0 kW	N. A.
	Agente GES	Exportação da Potência para o Sistema Elétrico (Rede) desde o Sistema das Micro-redes:	N. A.	N. A.	N. A.	-261.0 kW
		Potência fornecida ao Alimentador C:	N. A.	N. A.	N. A.	100.0 kW
Caso 2. Modo de operação: Sistema PV fornece à carga local e Exportação de Potência à rede (Desde o Sistema PV e Bateria).	Calculado pelo Agentes Gerenciadores	Potência do Sistema PV:	120.0 kW	100.0 kW	120.0 kW	N. A.
		Exportação de Potência à rede:	150.58 kW [Enviado para o Agente GES]	125.17 kW [Enviado para o Agente GES]	149.77 kW [Enviado para o Agente GES]	N. A.
		Potência Fornecida à Carga Local:	1.0 kW	2.0 kW	1.0 kW	N. A.
	Agente GES	Exportação da Potência para o Sistema Elétrico (Rede) desde o Sistema das Micro-redes:	N. A.	N. A.	N. A.	125.52 kW
		Potência fornecida ao Alimentador C:	N. A.	N. A.	N. A.	300.0k W

Fonte: Próprio autor

B.2.2.1 Casos de Teste das Micro-redes:

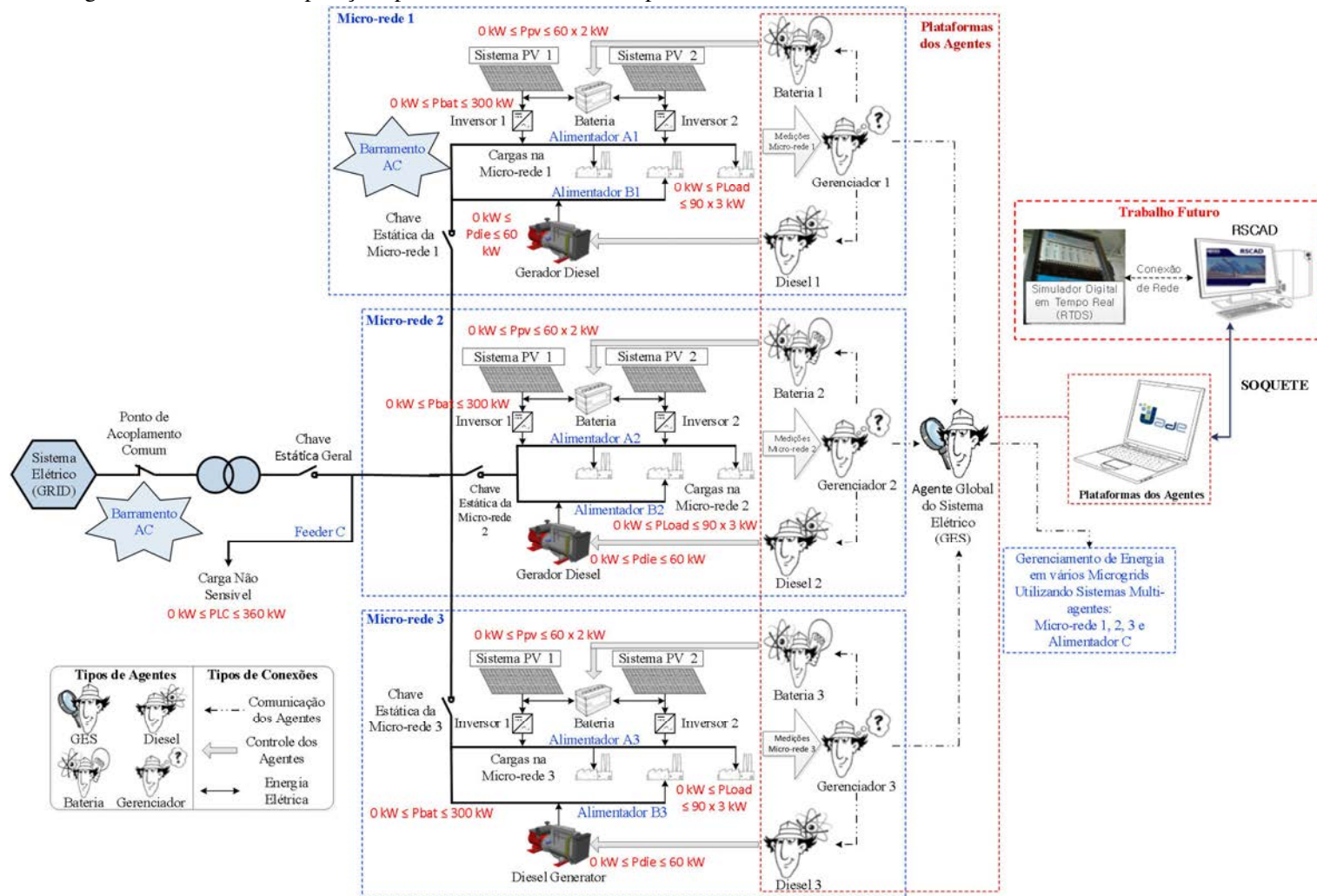
Na simulação de um sistema de micro-redes, no início de sua operação, a verificação do balanço da potência ativa é realizada a cada passo de tempo de simulação discreta. A fim de resolver o problema de gerenciamento de energia ativa, propõe-se um sistema multi-agentes (MAS) para fazer o gerenciamento de um sistema elétrico, o qual é composto por três micro-redes e um alimentador com cargas não-sensíveis, de acordo com o conceito apresentado na figura 93. Neste contexto, o número de parâmetros de entrada requeridos pelo MAS são:

- Seis entradas para cada microgrid: 1) Potência do Sistema Fotovoltaico (kW), 2) Potência da bateria (kW), 3) Potência do Gerador Diesel (kW), 4) Potência Ativa da Carga (kW), 5) Estado de Carga da Bateria (SOC) e 6) Posição da Chave Estática da Micro-rede (ON para micro-rede conectada à rede ou OFF para micro-rede ilhada).
- Três entradas para o sistema elétrico global: 1) Potência total da Carga no alimentador com cargas não-sensível (Alimentador C), 2) Posição da chave estática geral (ON para o sistema das micro-redes conectado à rede ou OFF para as três micro-redes ilhadas da rede elétrica) e 3) Posição do Ponto de acoplamento comum (ON para o sistema das micro-redes e o alimentador C conectados à rede ou OFF para as três micro-redes e o Alimentador C ilhadas da rede).

B.2.2.2 Conclusão:

O sistema multiagente proposto com JADE é capaz de controlar o gerenciamento da potência ativa no cenário de várias micro-redes, em condições conectadas à rede ou com o sistema das micro-redes ilhado e, os algoritmos apresentados podem ser usados para mais de três micro-redes, independentemente do tamanho da micro-rede.

Figura 93- Conceito de operações para o Cenário Modelado para o Gerenciamento da Potência Ativa em Várias Micro-redes.



Fonte: (HERNANDEZ et al., 2014).

DADOS QUANTITATIVOS REFERENTES À REVISÃO GLOBAL DO ESTADO DA ARTE DA PESQUISA

Considerando-se que o estado da arte é o nível mais alto de desenvolvimento numa área científica, procurado num momento determinado e num período definido, para a revisão global do estado da arte da pesquisa que se propõe desenvolver no doutoramento, utilizou-se a seguinte metodologia:

C.1 METODOLOGIA

Esta revisão foi feita considerando as seguintes setes palavras-chave: 1) Simulação digital em tempo real, 2) Sistemas de distribuição, 3) Linguagens de descrição de hardware em software, 4) Dispositivos FPGA, 5) Hardware in the Loop (HIL), 6) Micro-redes com geração distribuída e, 7) Redes de distribuição inteligentes.

Baseado no anterior, foi realizado um primeiro levantamento bibliográfico e, depois de uma ampla pesquisa na internet, abrangendo um período de 16 anos (desde 1994 até 2012), foram escolhidos para leitura e análise:

- ✓ Quatorze (14) trabalhos de pós-graduação de bases de dados de teses e dissertações dos seguintes países do mundo: Estados Unidos (2), Canadá (1), Portugal (1), Espanha (1) e Brasil (9). Foram escolhidas seis (6) Teses e oito (8) Dissertações. (Ver figura 2.1.)
- ✓ Vinte e sete (27) artigos do site web do **IEEE Xplore Digital Library**, dos quais sete (7) artigos nomeados com a letra [J] pertencem a revistas indexadas.
- ✓ Vinte e cinco (25) patentes de quatro (4) webs sites de patentes no mundo, incluindo Estados Unidos e Europa.

Com base no anterior, foram elaboradas 94 perguntas (com as suas respectivas respostas), derivadas de todo o material lido e revisado e, procurando obter a programação adequada das atividades subsequentes que compõem o projeto de pesquisa, assim:

- ✓ Trinta e duas (32), perguntas-respostas dos trabalhos de pós-graduação.
- ✓ Quarenta e cinco (45), perguntas-respostas dos artigos escolhidos.
- ✓ Dezessete (17), perguntas-respostas das patentes no mundo.

C.2 ESQUEMA DE REFERÊNCIA DA VISÃO GLOBAL DA PESQUISA PROPOSTA

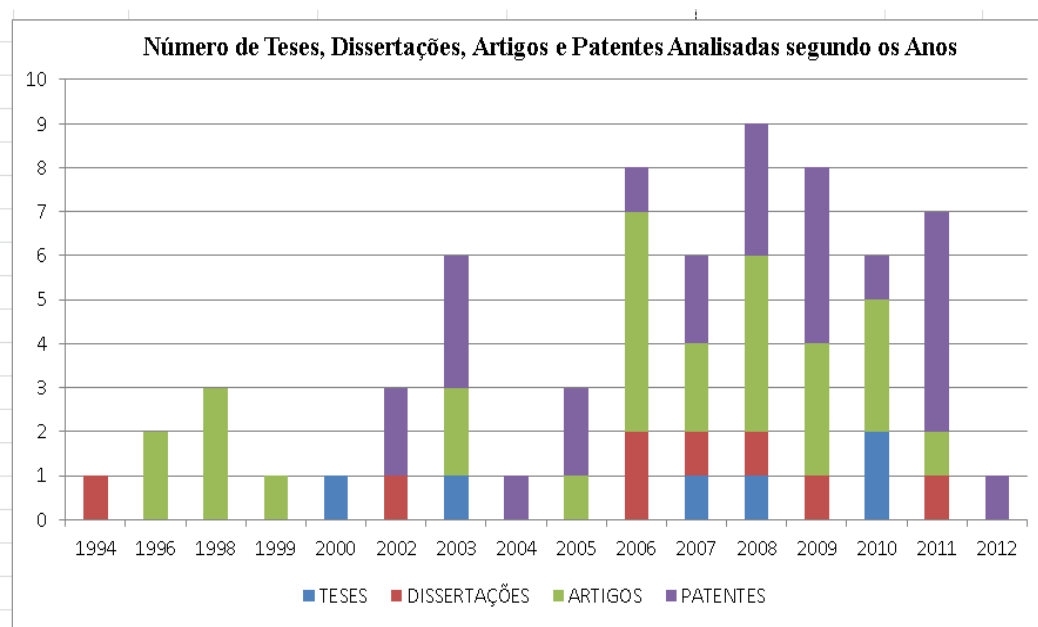
Esta primeira revisão global foi elaborada de acordo como seguinte esquema de referência:

- ✓ A área científica da pesquisa (busca) foi delimitada pelas seguintes setes (7) palavras-chave: 1) Simulação digital em tempo real, 2) Sistemas de distribuição, 3) Linguagens de descrição de

hardware em software, 4) Dispositivos FPGA, 5) Hardware in the Loop (HIL), 6) Micro-redes com geração distribuída e, 7) Redes de distribuição inteligentes. Estas palavras-chave foram utilizadas para fazer toda a pesquisa abrangida nesta tese.

- ✓ As datas das publicações das referências podem-se classificar segundo o seguinte gráfico (Figura 94):

Figura 94- Número de teses, dissertações, artigos e patentes analisadas para a primeira visão global do estado da arte segundo os anos de publicação.



Fonte: Elaboração do próprio autor.

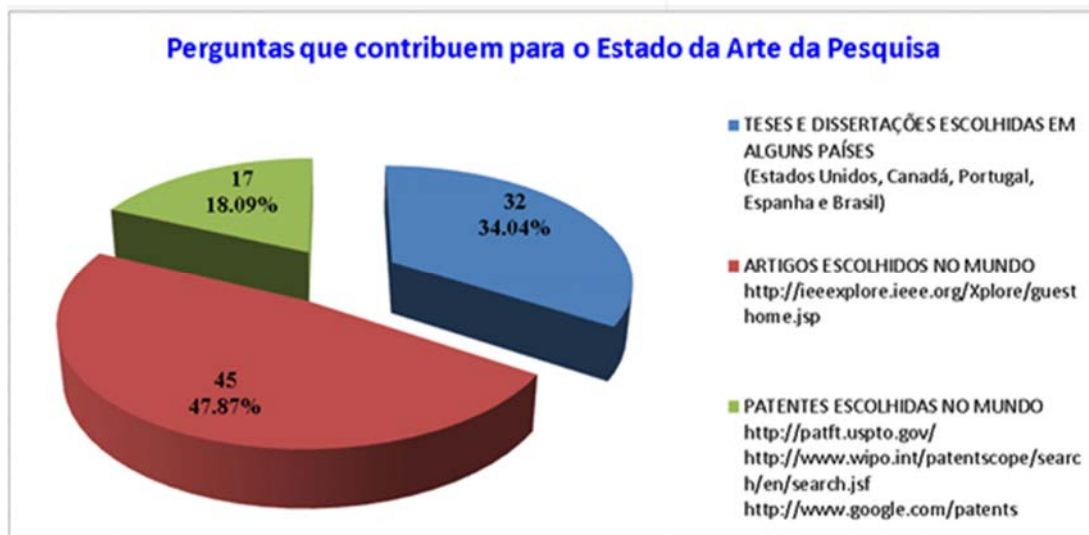
C.3 ESTRATÉGIA DE ANÁLISE DO MATERIAL ENCONTRADO NA REVISÃO DO ESTADO DA ARTE

Tendo em conta o esquema geral da pesquisa desenvolvida (Figura 2), como estratégia principal desta revisão bibliográfica, foi utilizada a técnica de perguntas-respostas para analisar todo o material encontrado. Foram elaboradas 94 perguntas com as suas respectivas respostas, derivadas dos trabalhos de pós-graduação, artigos e patentes escolhidas, visando obter:

- ✓ Um primeiro cenário do nível mais alto de desenvolvimento desta proposta de pesquisa científica através do conhecimento, sistematização e, hierarquização da produção científica achada.
- ✓ A programação adequada das atividades subsequentes que compõem este projeto de pesquisa, identificando debilidades e fortalezas, segundo a visão global da pesquisa proposta.

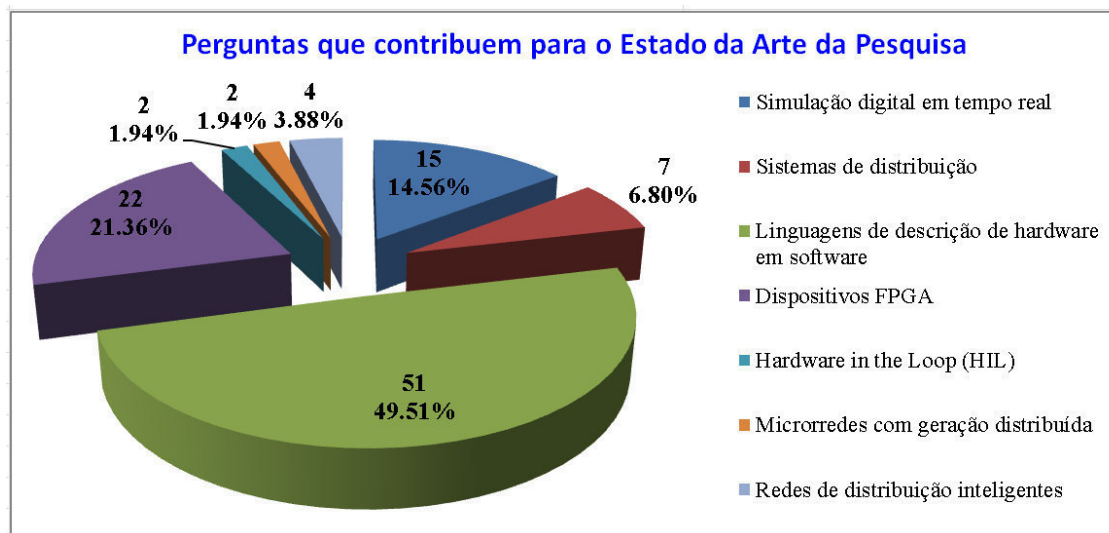
A visão geral das perguntas-respostas elaboradas pode-se ver de acordo às seguintes figuras 95 e 96.

Figura 95- Perguntas-respostas classificadas segundo os alvos da revisão global do estado da arte.



Fonte: Próprio autor.

Figura 96- Perguntas-respostas classificadas segundo as palavras-chave utilizadas.



Fonte: Próprio autor.

Apêndice D

BIBLIOTECAS DA LINGUAGEM VHDL-AMS, SIMULAÇÕES DAS CARGAS E LINHAS NÃO CRÍTICAS DO ALIMENTADOR ESCOLHIDO, CÓDIGOS DE PROGRAMAÇÃO E, HARDWARE RESULTANTE NO DISPOSITIVO FPGA

Sumário do Apêndice D

Item	Pag.
Bibliotecas da Linguagem VHDL-AMS	
D.1- BIBLIOTECAS MODELOS DA LINGUAGEM VHDL-AMS	184
D.1.1. Biblioteca dos Sistemas de Controle	184
D.1.2. Biblioteca Digital	186
D.1.4. Biblioteca Elétrica	187
Simulações em VHDL-AMS do Alimentador de Distribuição	
D.2- SIMULAÇÕES DE 7 LINHAS E DAS 13 CARGAS TOTAIS DO SISTEMA DE DISTRIBUIÇÃO PRIMÁRIO A 14,4 KV	188
Códigos gerados em VHDL-AMS	
D.3- CÓDIGO VHDL-AMS DO ALIMENTADOR DE BT DE 26 NÓS	208
D.4- CÓDIGO VHDL-AMS DO ALIMENTADOR DE MT DE 925 NÓS	214
Códigos gerados em VHDL	
D.5- CÓDIGO VHDL: BLOCO “Regulacao_Transf_v1_0.vhd”	232
D.6- Código VHDL: BLOCO “Regulacao_Transf_v1_0_S00_AXI_TRASNf_parameterized0.vhd”	236
D.7- CÓDIGO VHDL: BLOCO “Regulacao.vhd”	248
Códigos em Outras Linguagens de Programação	
D.8- CÓDIGO EM LINGUAGEM DE PROGRAMAÇÃO C: “main.c”	252
D.9- CÓDIGO EM MICROSOFT VISUAL BASIC: “mainModuleChk.bas”	257
Hardware Resultante no Dispositivo FPGA	
D.10- HARDWARE RESULTANTE NO DISPOSITIVO FPGA (ESQUEMAS COMPLETOS)	265

D.1- BIBLIOTECAS MODELOS DA LINGUAGEM VHDL-AMS

Na linguagem VHDL-AMS as bibliotecas modelos são aquelas coleções padronizadas e classificadas por áreas do conhecimento, disponíveis para serem utilizadas nas modelações e simulações de sistemas de sinal analógico, digital ou misto. Segundo o padrão IEEE 1076, as seguintes bibliotecas estão atualmente na Linguagem VHDL-MAS. Estas bibliotecas estão apresentadas na tabela 24.

TABELA 24- BIBLIOTECAS MODELOS DA LINGUAGEM VHDL-AMS.

1. Controls Library	6. Mixed Signal Library	11. VDA Spice2VHD Library
2. Digital Library	7. Mixed Technology Library	12. Fundamentals_VDA Library
3. Electrical Library	8. Rotational Library	13. VDA Automotive Library
4. Hydraulic (Fluidic) Library	9. Translational Library	14. VDA Megma Library
5. Magnetic Library	10. Thermal Library	

Tendo em conta que esta tese está focalizada aos sistemas elétricos, faremos um exame detalhado das três primeiras bibliotecas na linguagem VHDL-AMS, apresentando o seguinte conjunto símbolos ou modelos parametrizáveis pelo usuário que podem ser usados diretamente nas simulações de sistemas com sinais analógicas ou digitais:

D.1.1. Biblioteca dos Sistemas de Controle:

A biblioteca dos Sistemas de Controle da Linguagem VHDL-AMS contém um conjunto símbolos ou modelos VHDL-AMS parametrizáveis pelo usuário que podem ser usados diretamente em simulações de sistemas. Estes modelos têm tipicamente duas arquiteturas, um para o domínio-S (contínua) e um para o domínio-Z (discreta). Os seguintes símbolos ou modelos estão disponíveis na biblioteca de controle VHDL-AMS, de acordo com a tabela 25.

TABELA 25- BIBLIOTECA DOS SISTEMAS DE CONTROLE.

No.	Nome da Biblioteca	Significado
1	e_BPF	Filtro passa-banda com conexões elétricas.
2	e_BRF	Rejeitar banda (Notch). Filtro com conexões elétricas.
3	e_Deriv	Bloco derivativo ideal com conexões elétricas.
4	e_DerivLag	Bloco derivativo com pólo. Conexões elétricas.
5	e_Difference	Junção com conexões elétricas - diferenciação (in2 in1).
6	e_Gain	Bloco de Ganho com conexões elétricas.
7	e_Integ	Bloco Integrador com conexões elétricas.
8	e_LPF_1st	Filtro de primeira ordem passa baixo com conexões elétricas.
9	e_LPF_2nd	Filtro de segunda ordem passa baixo com conexões elétricas.
10	e_LeadLag	Filtro de adianto e atraso com conexões elétricas.
11	e_Limit	Limitador com conexões elétricas. Limites têm uma pequena inclinação.
12	e_Mult	Multiplificador com duas entradas com conexões elétricas.
13	e_NthOrderPoly	Função de transferência polinomial de enésimo grau no domínio S, com conexões elétricas.
14	e_PWL_tf	Bloco de função linear por partes elétricas com transferência.
15	e_Sqrt	Função de raiz quadrada com conexões elétricas.
16	e_Sum	Junção soma (+ in1 in2) com conexões elétricas.
17	q_BPF	Filtro passa-banda com conexões de quantidade real.
18	q_BRF	Rejeitar banda (Notch) Filtro com conexões de quantidade real.
19	q_Deriv	Bloco derivativo com conexões de quantidade real.

20	q_DerivLag	Bloco derivativo com Atraso, conexões de quantidade real.
21	q_Difference	Junção de diferenciação com quantidade real (in2 in1).
22	q_Gain	Bloco de Ganho com conexões de quantidade real.
23	q_Integ	Bloco Integrador com conexões de quantidade real.
24	q_LPF_1st	Filtro de primeira ordem passa baixo com conexões de quantidade real.
25	q_LPF_2nd	Filtro de segunda ordem passa baixo com conexões de quantidade real.
26	q_LeadLag	Filtro de adianto e atraso com conexões de quantidade real.
27	q_Limit	Limitador de conexões com a quantidade real.
28	q_Mult	Multiplicador com duas entradas com conexões de quantidade real.
29	q_NthOrderPoly	Função de transferência polinomial de enésimo grau no domínio S.
30	q_PWL_tf	Bloco de função linear por partes com quantidade real.
31	q_Sqrt	Função de raiz quadrada com conexões de quantidade real.
32	q_Sum	Soma de junção (in1 + in2) com conexões de quantidade real.
33	q_from_angle	Modelo de conversor ideal, a partir de um ângulo de rotação (rad) de entrada.
34	q_from_current	Modelo de conversor ideal, de uma entrada de corrente elétrica.
35	q_from_displacement	Modelo de conversor ideal, a partir de um deslocamento de translação (em metros).
36	q_from_flux	Modelo de conversor ideal, a partir da entrada de fluxo magnético (Wb).
37	q_from_force	Modelo de conversor ideal, de uma força de entrada de translação (N).
38	q_from_heat_flow	Modelo de conversor ideal, a partir da entrada do fluxo de calor térmico (Watts).
39	q_from_mmf	Modelo de conversor ideal, a partir da de entrada de um mmf magnético ($A \cdot t$).
40	q_from_pressure	Modelo de conversor ideal, a partir de uma entrada de pressão fluídica (Pa).
41	q_from_temperature	Modelo de conversor ideal, a partir de uma temperatura (K) de entrada.
42	q_from_torque_r	Modelo de conversor ideal, a partir de um torque de rotação ($N \cdot m$) de entrada.
43	q_from_torque_rv	Modelo de conversor ideal, a partir de um torque ($N \cdot m$) de entrada para a quantidade.
44	q_from_voltage	Modelo de conversor ideal, a partir de uma tensão elétrica de entrada.
45	q_to_ang_vel	Modelo de conversor ideal, a partir da entrada para uma quantidade de rotação.
46	q_to_angle	Modelo de conversor ideal, a partir da entrada quantidade a uma rotação.
47	q_to_current	Modelo de conversor ideal, a partir da entrada de uma quantidade elétrica.
48	q_to_displacement	Modelo de conversor ideal, a partir da entrada para uma quantidade de translação.
49	q_to_flux	Modelo de conversor ideal, a partir da entrada para uma quantidade de um fluxo magnético.
50	q_to_force	Modelo de conversor ideal, a partir da entrada para uma quantidade de translação.
51	q_to_heat_flow	Modelo de conversor ideal, a partir da entrada para uma quantidade térmica.
52	q_to_mmf	Modelo de conversor ideal, a partir da entrada para uma quantidade de campo magnético.
53	q_to_pressure	Modelo de conversor ideal, a partir da entrada para uma quantidade fluídica de pressão.
54	q_to_temperature	Modelo de conversor ideal, a partir da entrada para uma quantidade de temperatura (K).
55	q_to_torque_r	Modelo de conversor ideal, a partir da entrada para uma quantidade de rotação.
56	q_to_torque_rv	Modelo de conversor ideal, a partir da entrada para uma quantidade de torque ($N \cdot m$).
57	q_to_vflow_rate	Modelo de conversor ideal, a partir da entrada para uma quantidade fluídica.
58	q_to_voltage	Modelo de conversor ideal, a partir da entrada de uma quantidade elétrica.
59	qsrc_Constant	Fonte constante com conexões de quantidade real.
60	qsrc_PWL	Fonte de quantidade real por partes lineares.
61	qsrc_Pulse	Fonte de pulso com conexões de quantidade real.
62	qsrc_Sine	Fonte de tensão senoidal com conexões de quantidade real.

D.1.2. Biblioteca Digital

A Biblioteca Digital da linguagem VHDL-AMS contém um conjunto símbolos ou modelos VHDL-AMS parametrizáveis pelo usuário que podem ser usados diretamente em simulações de sistemas digitais. Os seguintes símbolos ou modelos estão disponíveis na Biblioteca Digital da linguagem VHDL-AMS, de acordo com a tabela 26.

TABELA 26- BIBLIOTECA DOS SISTEMAS DIGITAIS.

No.	Nome da Biblioteca	Significado
1	DLatch	Trava com reset.
2	JKFF	Flip flop com preset e clear.
3	LevelSet	Nível de saída constante digital.
4	and2	Entrada AND e duas comportas de entradas NAND.
5	AND	Duas comportas de entradas NAND.
6	and3	Comporta de Entrada AND e três comportas de entradas NAND.
7	and4	Comporta de Entrada AND e 4 comportas de entradas NAND.
8	and8	Comporta de Entrada AND e 8 comportas de entradas NAND.
9	buf	Buffer com atraso de buffer Tri-afirmar com tempo de atraso.
10	clock	Relógio com 0,5 ciclos.
11	clock_en	Relógio com 0,5 ciclos de trabalho e disponível.
12	clock_freq	Relógio com 0,5 ciclos de trabalho (especificar frequência).
13	clock_jitter	Fonte de relógio com uma função de vibração na borda (distribuição uniforme).
14	clock_pulse	Fonte de pulso com o Período e largura genéricos.
15	d_bus_cmp	Comparador de barramento.
16	data_logic_vector	Fonte de dados. Sequência de dados é especificado como um std_logic_vector.
17	dig_prbs	Pseudo-Random Bit Stream (PRBS) Dados.
18	dig_prbs_jitter	Bit Stream pseudoaleatórios (PRBS) de Dados com construção na Borda.
19	dig_pulse	Fonte de pulso com Período e Ciclo genéricos.
20	inverter	Inversor com tempo de atraso.
21	mux4	Multiplexador com permissão.
22	nand2	Comporta de entrada NAND.
23	nand3	Comporta de entrada NAND.
24	nand4	Comporta de entrada NAND.
25	nand8	Comporta de entrada NAND
26	nor2	Comporta de entrada NOR
27	nor3	Comporta de entrada NOR.
28	nor4	Comporta de entrada NOR.
29	nor8	Comporta de entrada NOR.
30	or2	Comporta de entrada NOR e duas comportas de entradas OR.
31	gate	Duas comportas de entradas OR e duas comportas de entrada XNOR.
32	or3	Comporta de entrada NOR e três comportas de entrada XNOR
33	Three	Comporta de entrada XOR.
34	or4	Comporta de entrada NOR e quatro comportas de entrada XNOR.
35	input	Comporta de entrada.
36	or8	Comporta de entrada NOR e oito comportas de entrada XNOR.
37	Eight	Comporta de entrada XOR.
38	p2s_shift_reg	Registro serial de deslocamento com carga assíncrona.
39	pattern_det	Dispositivo de detecção.
40	s2p_shift_reg	Registro paralelo de deslocamento com carga assíncrona.
41	triand2	Duas entradas porta AND.
42	tribuf	Buffer com tempo de atraso.
43	triinv	Inversor com tempo de atraso.
44	trinand2	Duas entradas NAND.
45	trinator2	Duas entradas porta NOR.

46	trior2	Duas entradas porta OR.
47	xnor2	Comporta de entrada XNOR.
48	xnor3	Comporta de entrada XNOR.
49	xnor4	Comporta de entrada XNOR.
50	xnor8	Comporta de entrada XNOR.
51	xor2	XOR comporta de entrada.
52	xor3	XOR comporta de entrada.
53	xor4	XOR comporta de entrada.
54	xor8	XOR comporta de entrada.

D.1.4. Biblioteca Elétrica

A Biblioteca Elétrica da linguagem VHDL-AMS contém um conjunto símbolos ou modelos VHDL-AMS parametrizáveis pelo usuário que podem ser usados diretamente em simulações de sistemas elétricos. Os seguintes símbolos ou modelos estão disponíveis na Biblioteca Elétrica da linguagem VHDL-AMS, de acordo com a tabela 27.

TABELA 27- BIBLIOTECA DOS SISTEMAS ELÉTRICOS.

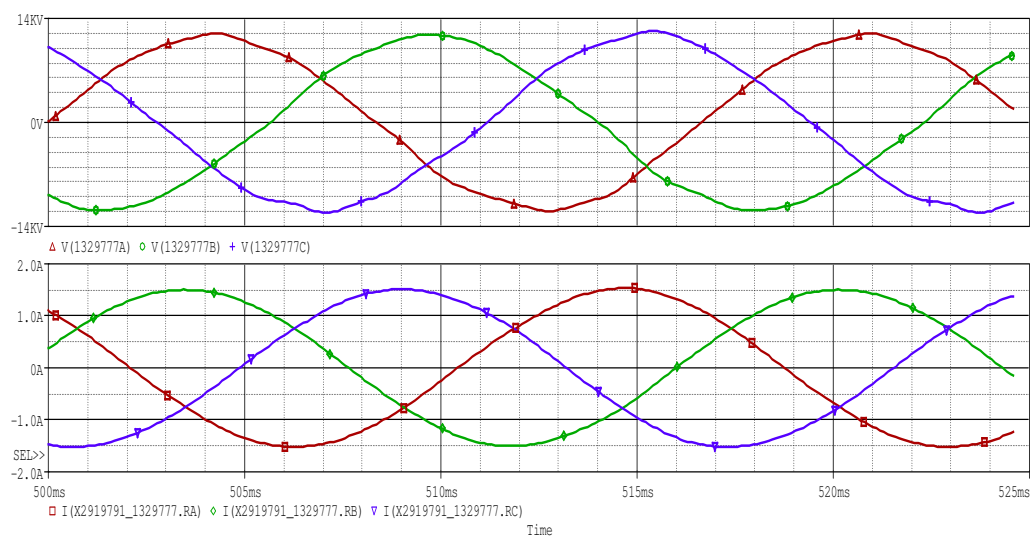
No.	Nome da Biblioteca	Significado
1	BandPassFilter	Filtro passa-banda com conexões elétricas. A entrada e a saída.
2	BandRejectFilter	Filtro Rejeitar banda com conexões elétricas. A entrada e a saída.
3	DifferenceAmp	Duas entradas simples de amplificador diferencial com conexões elétricas.
4	GainAmp	Amplificador de Ganho simples de com conexões elétricas.
5	HighPassFilter	Filtro passa alta de segunda ordem com conexões elétricas.
6	IntegratingAmp	Amplificador de integração simples com conexões elétricas.
7	LimitingAmp	Amplificador Limitador com conexões elétricas.
8	LowPassFilter	Filtro passa baixo de segunda ordem com conexões elétricas.
9	MeasFreq	Medição da frequência de um sinal elétrico de entrada (tensão).
10	Multiplier	Multiplicador simples de duas entradas com conexões elétricas.
11	OpAmp_3p	Modelo de AmpOp de 3-pin com arquitetura comportamental.
12	SummingAmp	Amplificador com duas entradas com conexões elétricas.
13	TxLine	Linha de Transmissão Elétrica Ideal sem perdas.
14	VCOAnalog	Oscilador analógico de tensão controlado.
15	VoltageClamp	Modelo de pinça simples de tensão.
16	capacitor	Capacitor elétrico.
17	diode	Modelos Básicos do Diodo Zener.
18	i_constant	Fonte constante de corrente elétrica.
19	i_pulse	Fonte do pulso de corrente elétrica.
20	i_sine	Fonte de corrente elétrica sinusoidal.
21	inductor	Indutor elétrico.
22	resistor	Resistor elétrico.
23	sw_1p2t	Interruptor elétrico de um pólo e duplo deslocamento.
24	sw_1p3t	Interruptor elétrico de um pólo e triplo deslocamento.
25	sw_2p2t	Interruptor elétrico de pólo duplo e duplo deslocamento.
26	sw_ctrl	Controlador de interruptor com sinal de saída inteiro.
27	sw_matrix	Comutação de matriz elétrica.
28	v_PWL	Fonte de tensão linear por partes.
29	v_SweptSine	Fonte de tensão senoidal Swept.
30	v_constant	Fonte de tensão constante.
31	v_pulse	Fonte de pulso de tensão.
32	v_sine	Fonte de tensão elétrica sinusoidal.
33	zener_diode	Diodo Zener.

D.2- SIMULAÇÕES DE 7 LINHAS E DAS 13 CARGAS TOTAIS DO SISTEMA DE DISTRIBUIÇÃO PRIMÁRIO A 14,4 KV

LINHA_PRI-X2919791_1329777

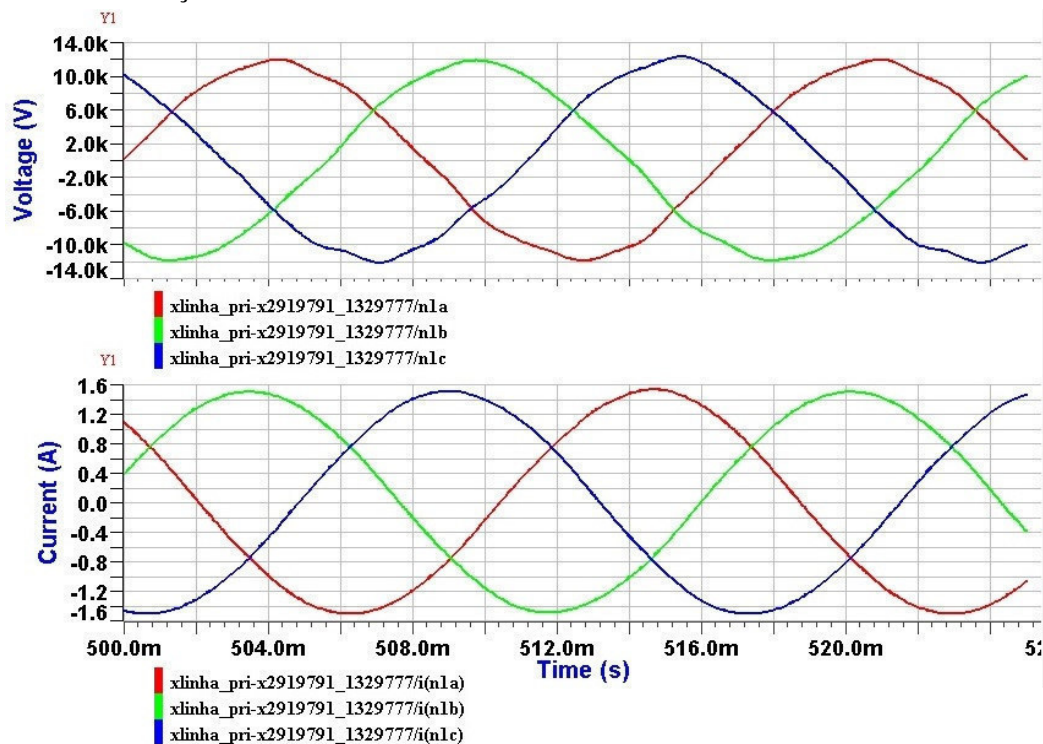
Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 97- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X2919791_1329777.



Fonte: Próprio autor.

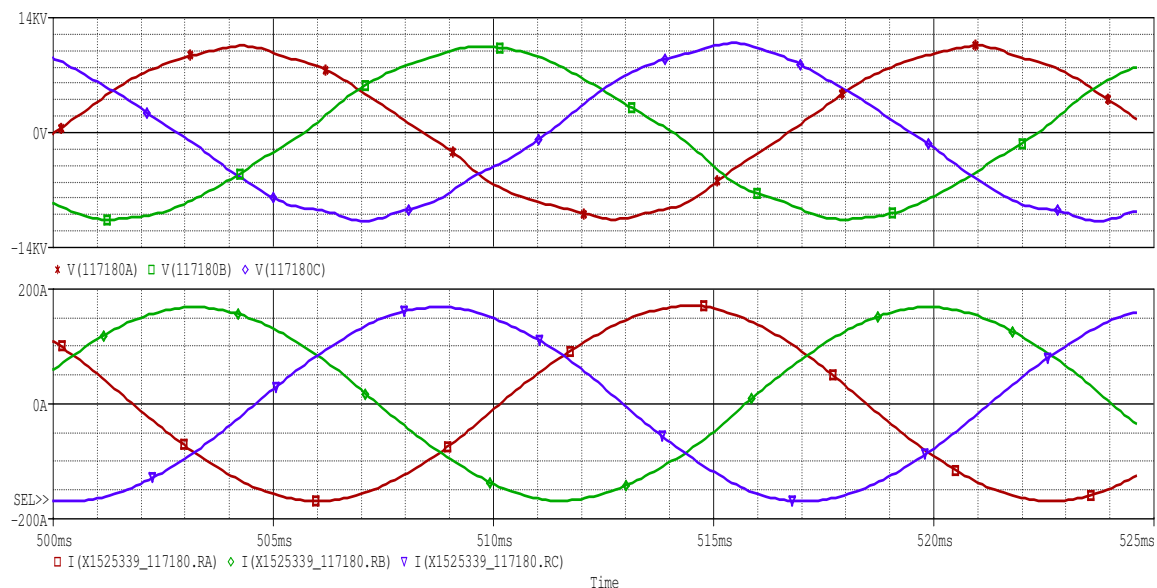
Figura 98- Simulação em ambiente VHDL-AMS da LINHA_PRI-X2919791_1329777.



Fonte: Próprio autor.

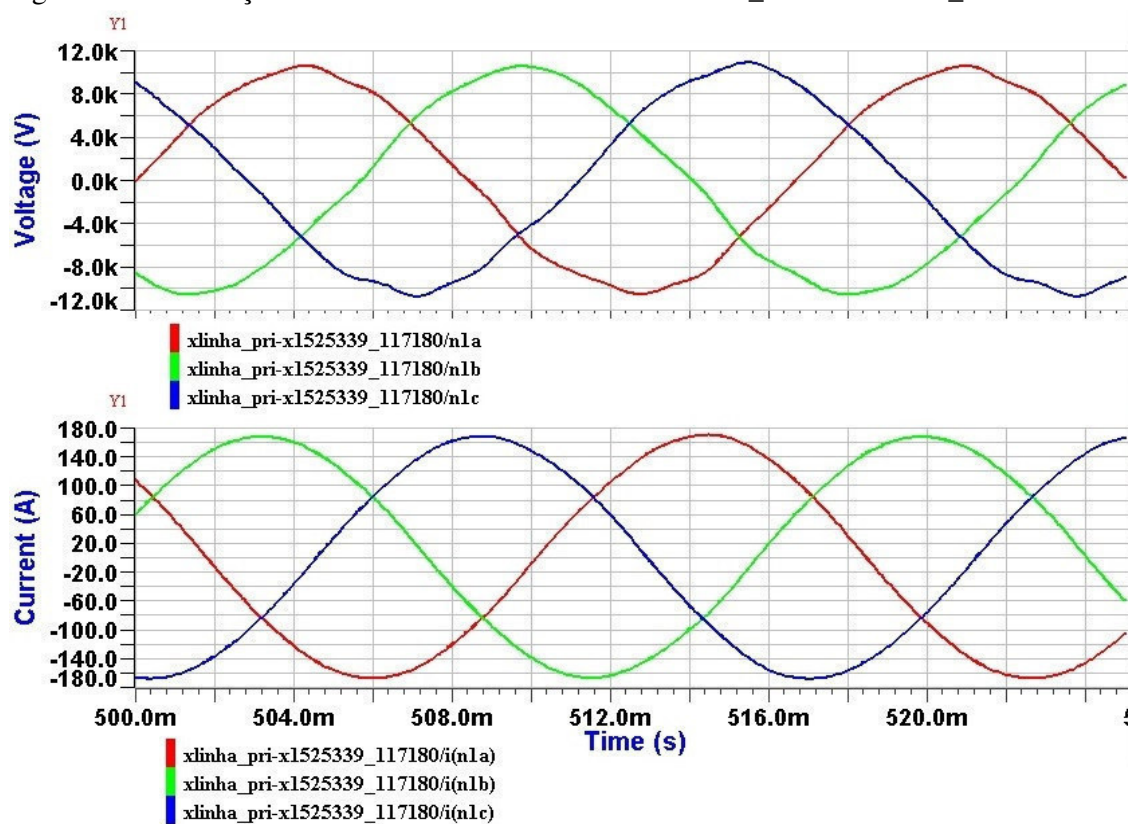
LINHA_PRI-X1525339_117180
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 99- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X1525339_117180.



Fonte: Próprio autor.

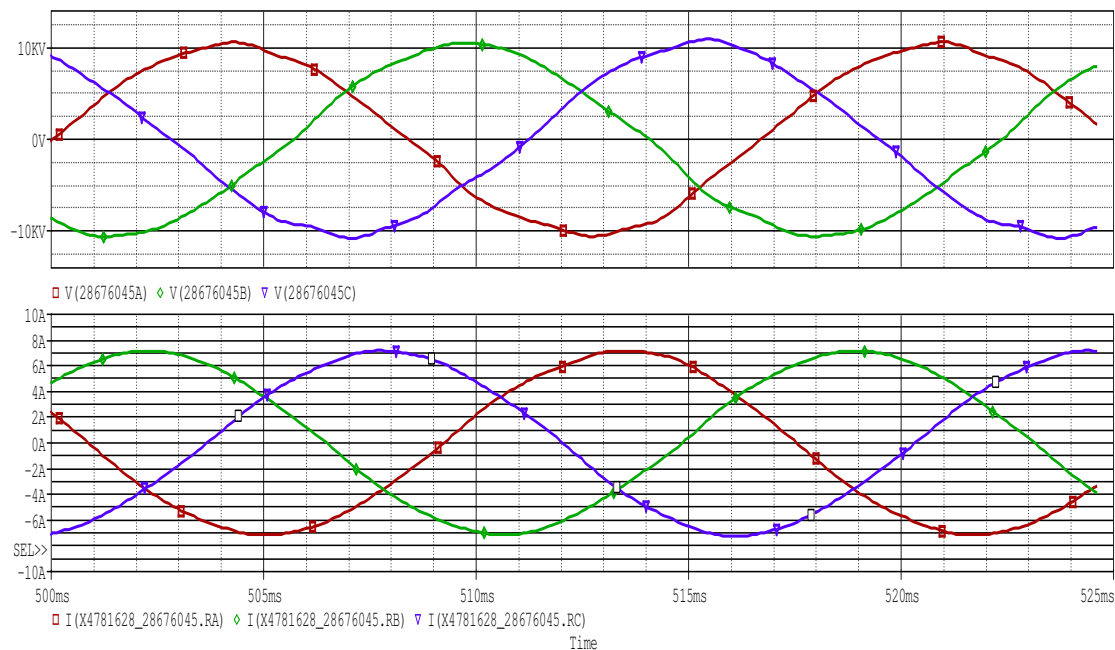
Figura 100- Simulação em ambiente VHDL-AMS da LINHA_PRI-X1525339_117180.



Fonte: Próprio autor.

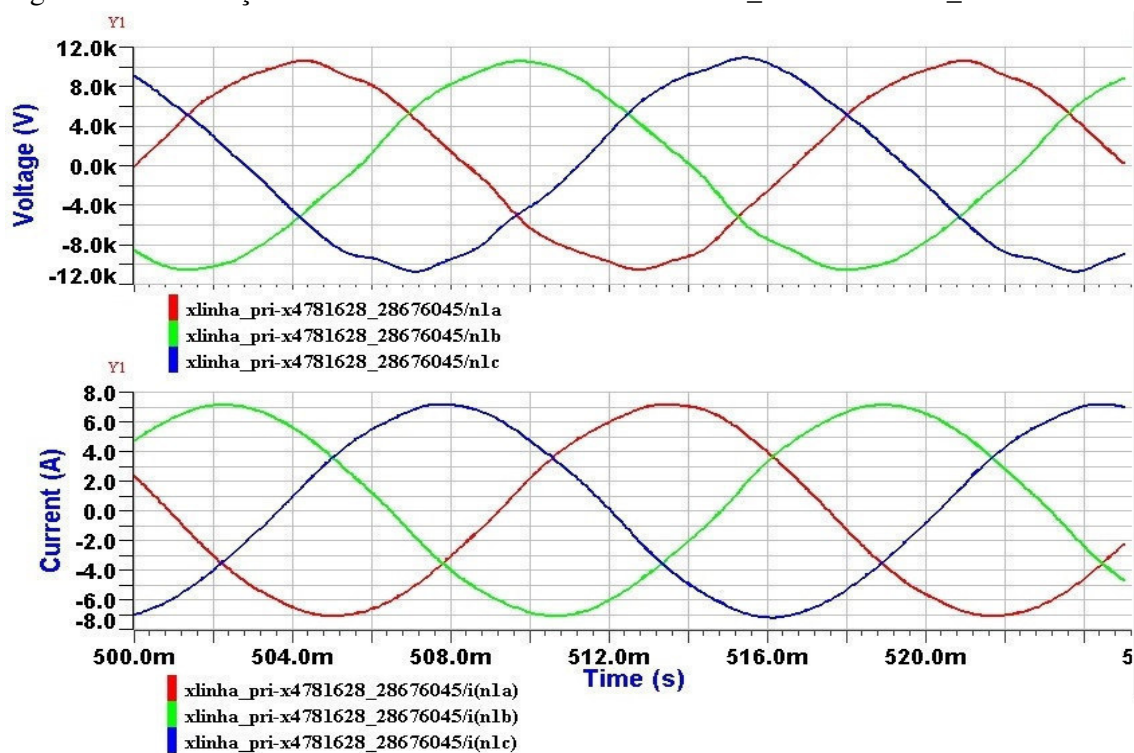
LINHA_PRI-X4781628_28676045
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 101- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X4781628_28676045.



Fonte: Próprio autor.

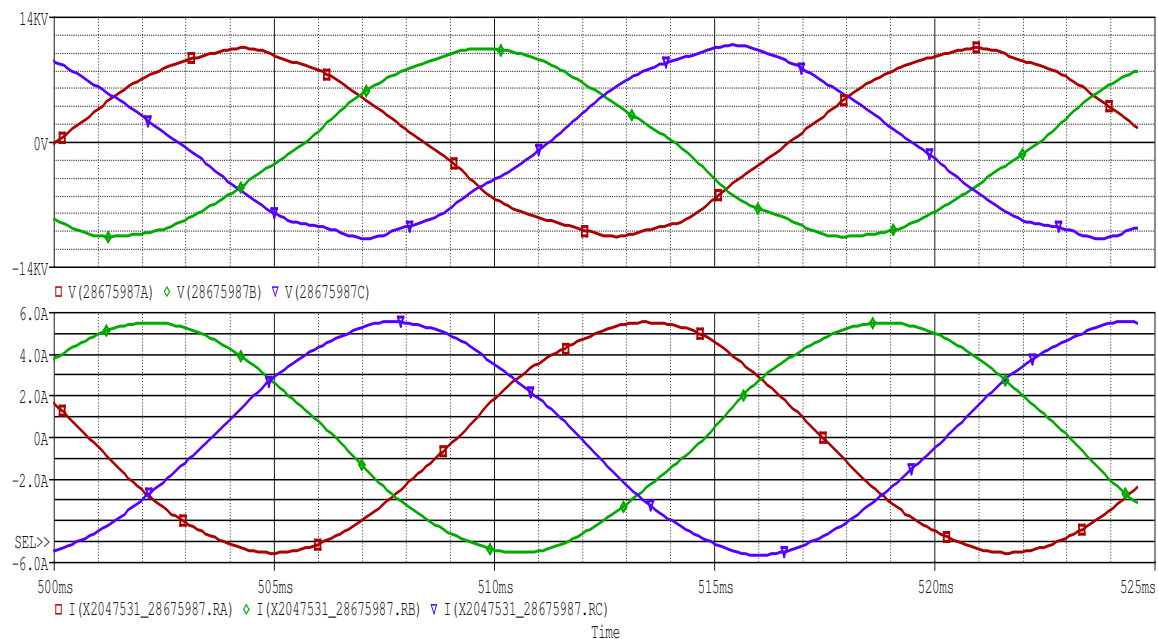
Figura 102- Simulação em ambiente VHDL-AMS da LINHA_PRI-X4781628_28676045.



Fonte: Próprio autor.

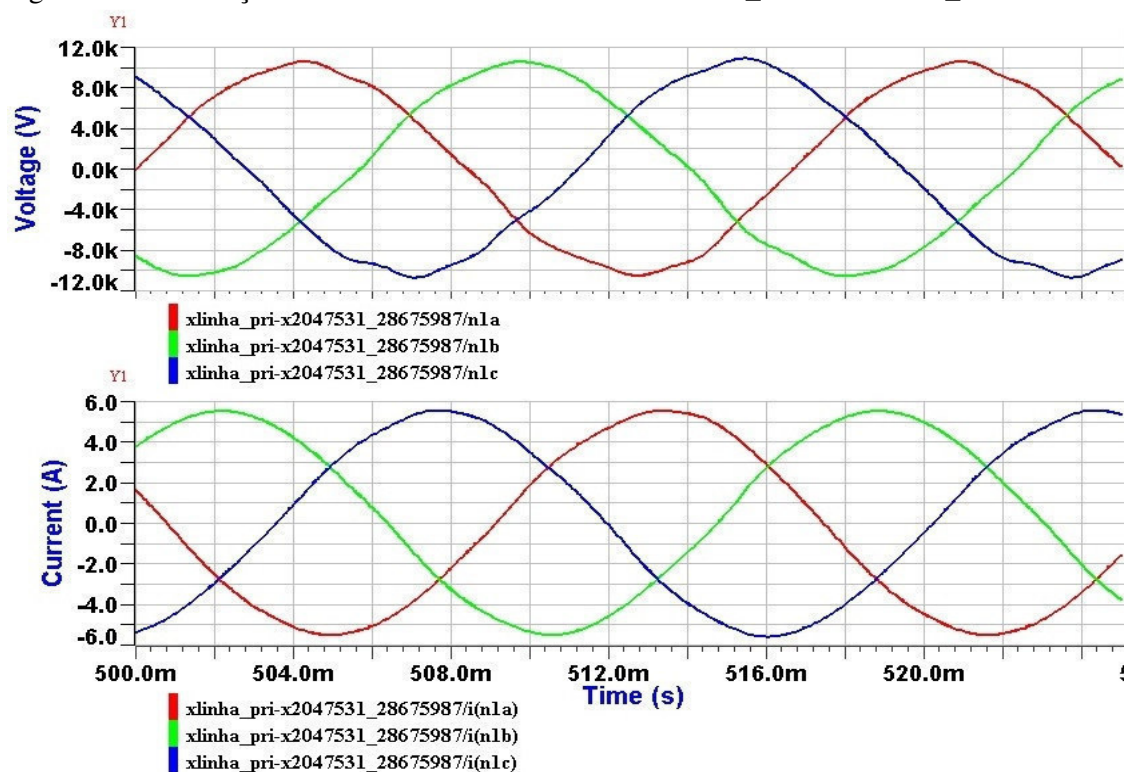
LINHA_PRI-X2047531_28675987
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 103- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X2047531_28675987.



Fonte: Próprio autor.

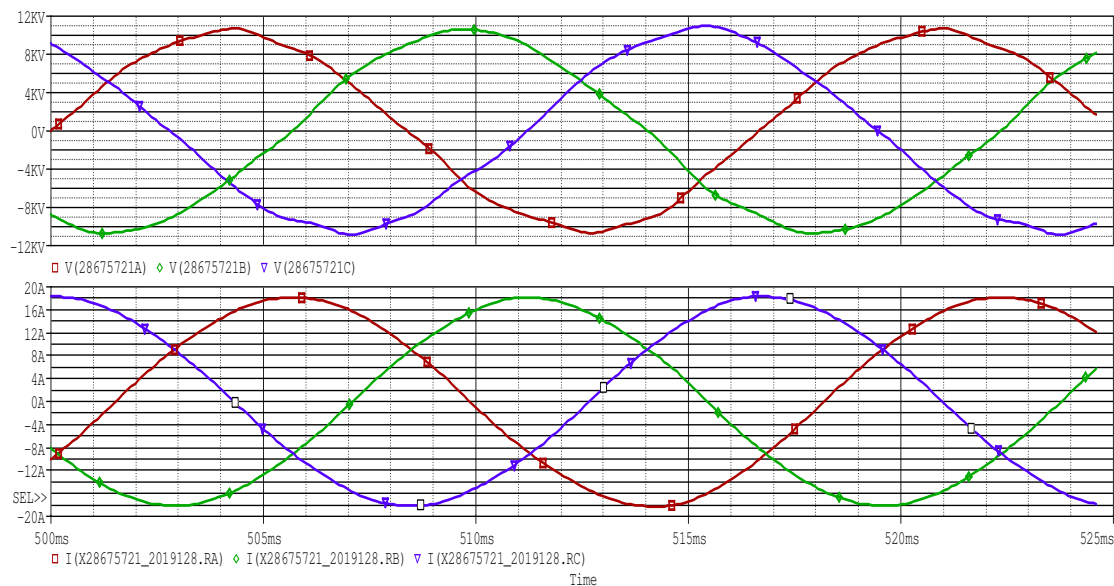
Figura 104- Simulação em ambiente VHDL-AMS da LINHA_PRI-X2047531_28675987.



Fonte: Próprio autor.

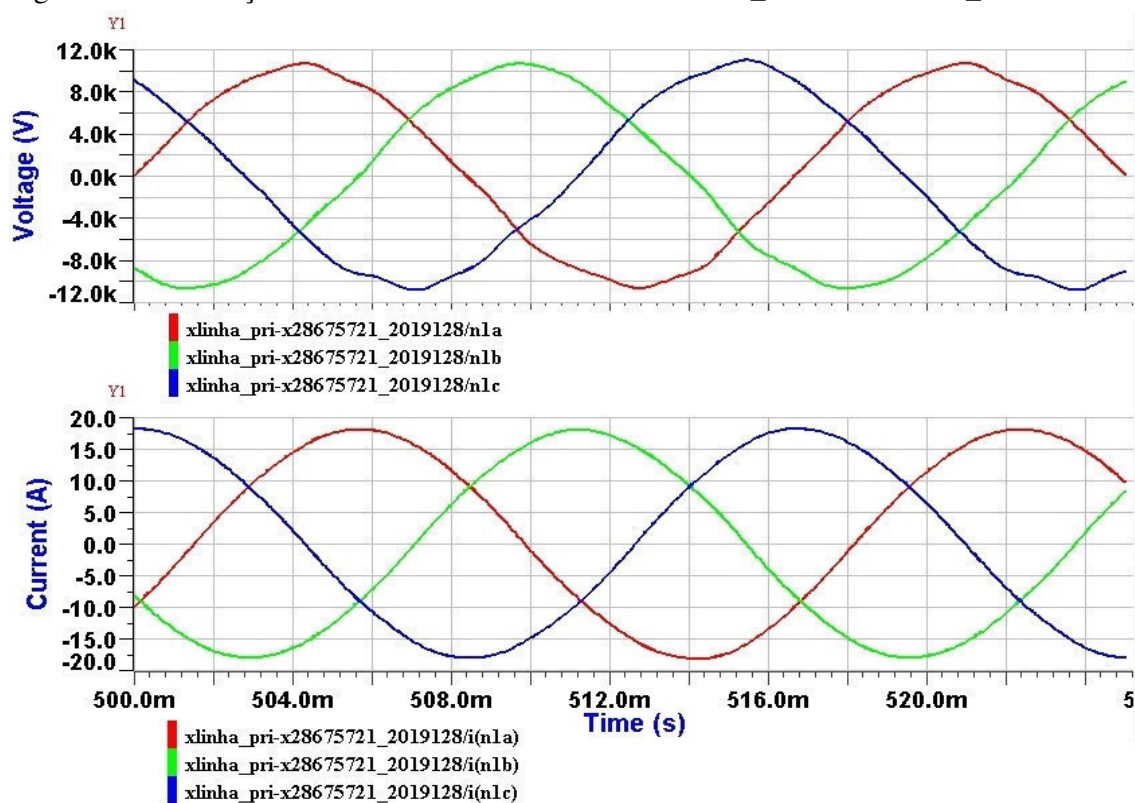
LINHA_PRI-X28675721_2019128
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 105- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X28675721_2019128.



Fonte: Próprio autor.

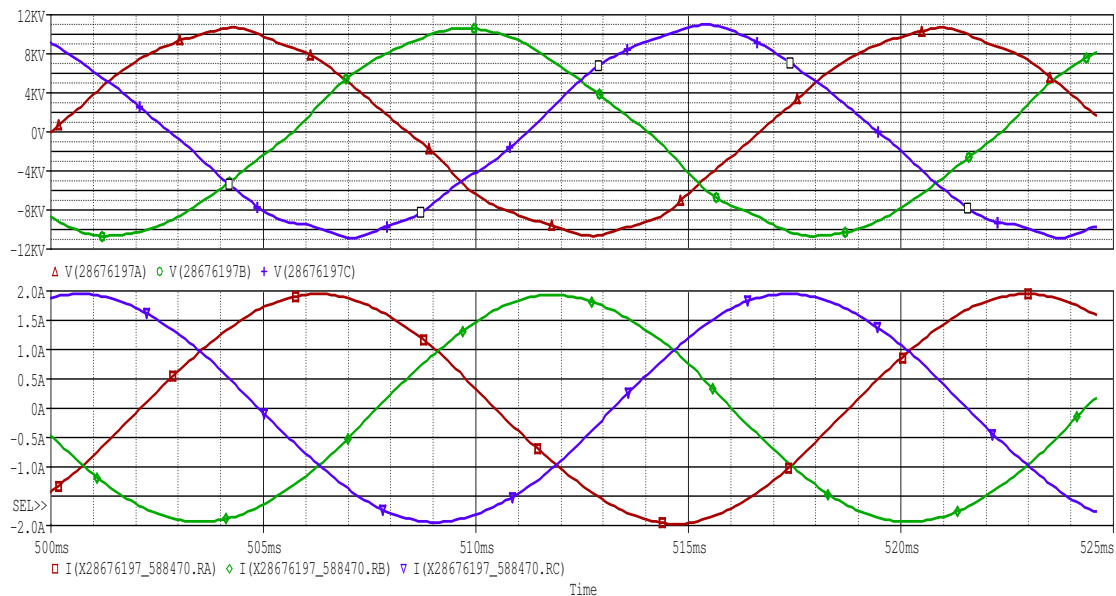
Figura 106- Simulação em ambiente VHDL-AMS da LINHA_PRI-X28675721_2019128.



Fonte: Próprio autor.

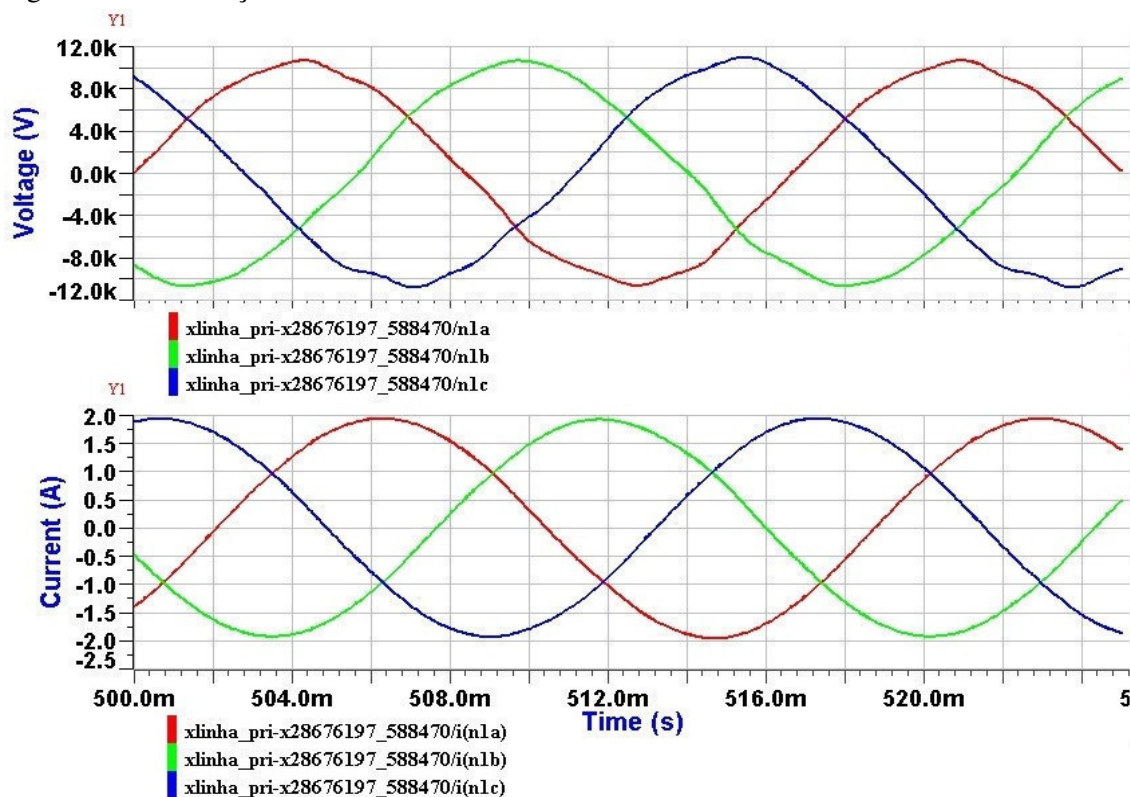
LINHA_PRI-X28676197_588470
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 107- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X28676197_588470.



Fonte: Próprio autor.

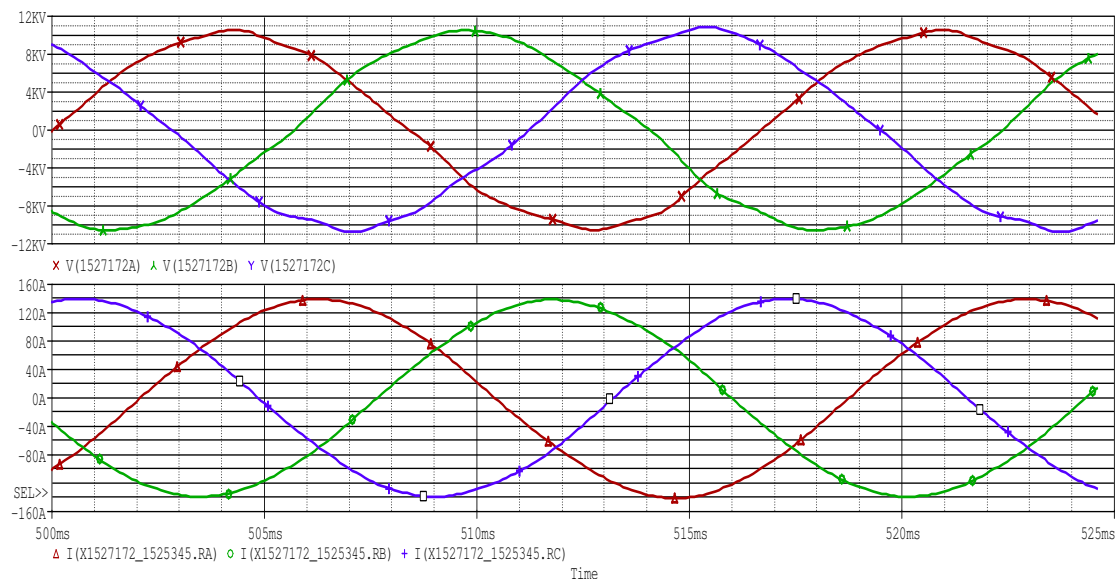
Figura 108- Simulação em ambiente VHDL-AMS da LINHA_PRI-X28676197_588470.



Fonte: Próprio autor.

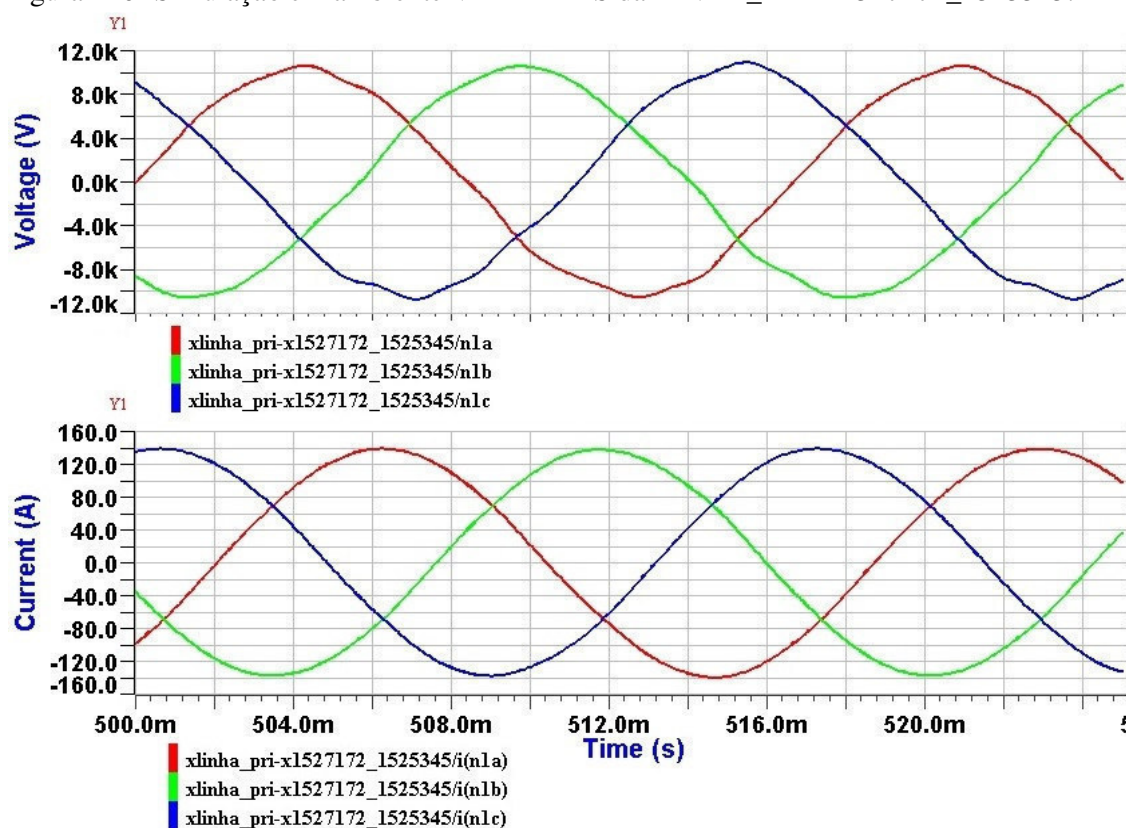
LINHA_PRI-X1527172_1525345
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 109- Simulação em ambiente Orcad/Pspice da LINHA_PRI-X1527172_1525345.



Fonte: Próprio autor.

Figura 110- Simulação em ambiente VHDL-AMS da LINHA_PRI-X1527172_1525345.



Fonte: Próprio autor.

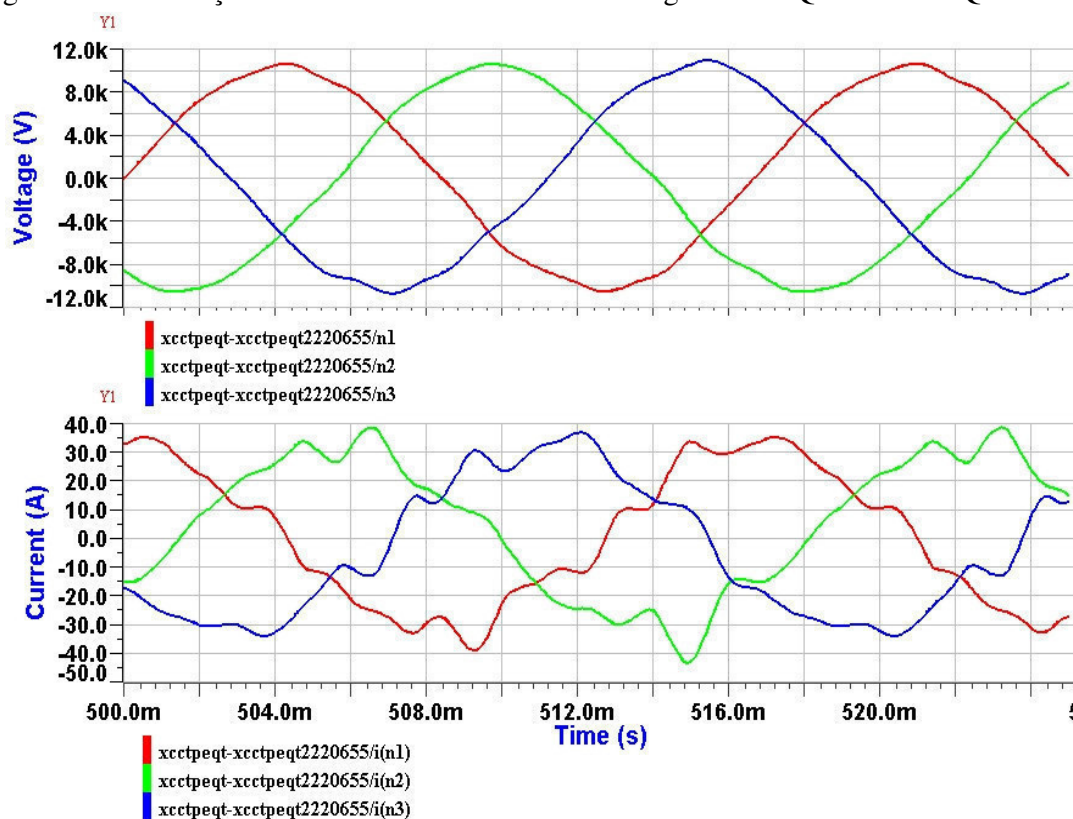
CCTPEQT-XCCTPEQT2220655
 Carga Capacitiva Trifásica Primária Equilibrada (Triângulo)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 111- Simulação em ambiente Orcad/Pspice da carga CCTPEQT-XCCTPEQT2220655.



Fonte: Próprio autor.

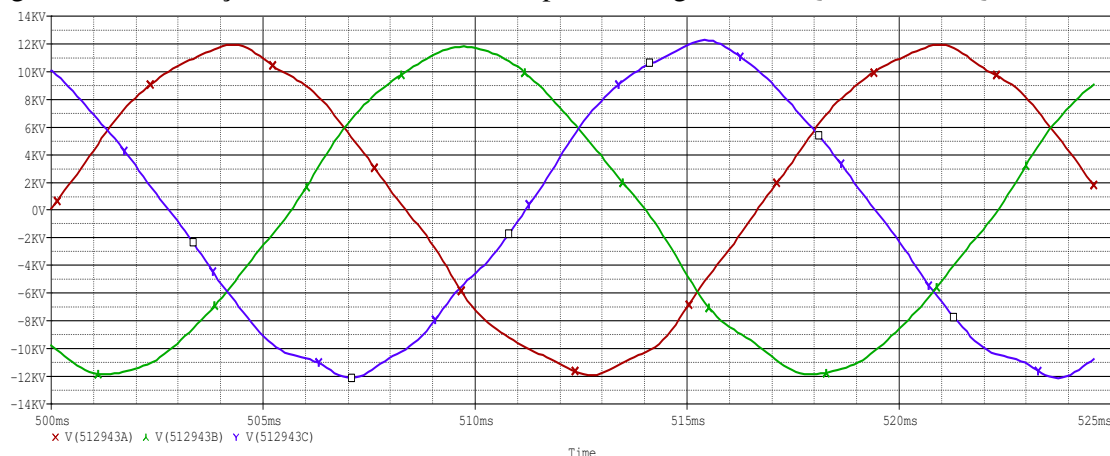
Figura 112- Simulação em ambiente VHDL-AMS da carga CCTPEQT-XCCTPEQT2220655.



Fonte: Próprio autor.

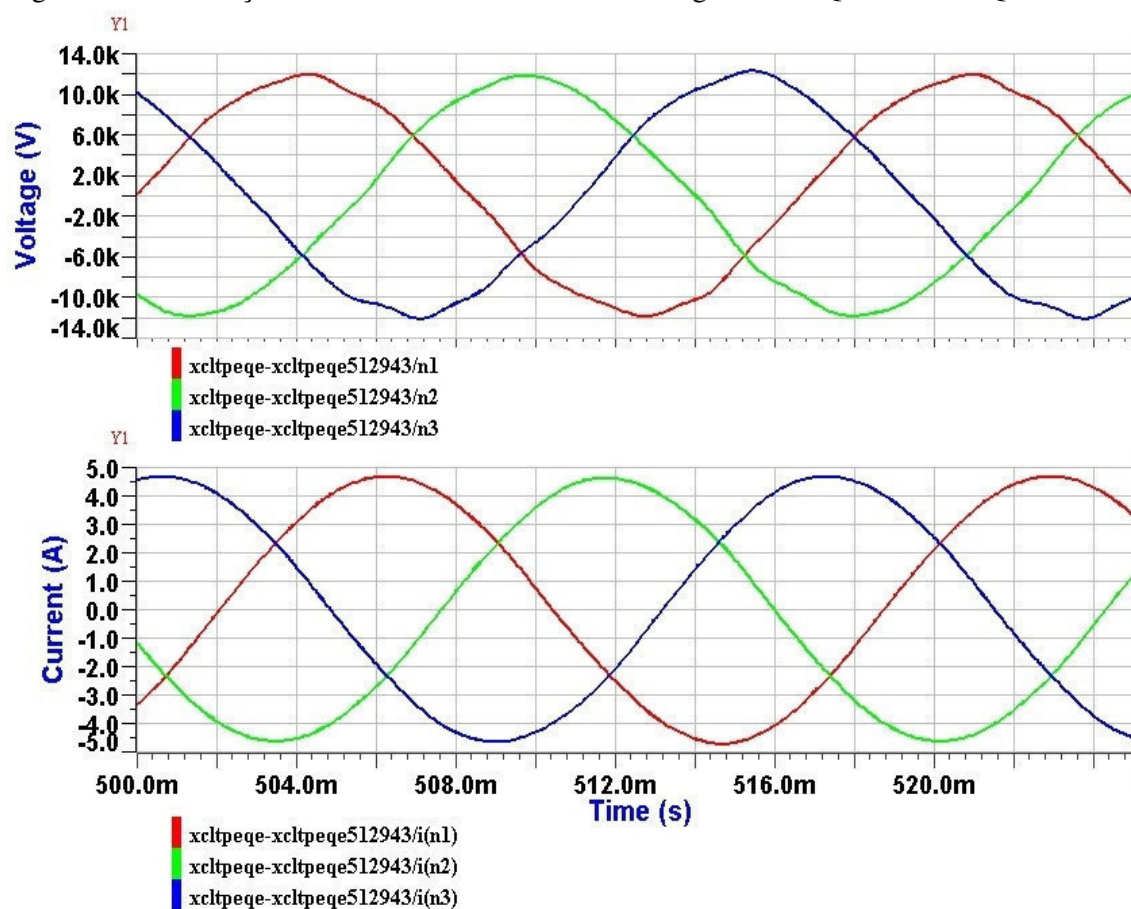
CLTPEQE-XCLTPEQE512943
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 113- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE512943.



Fonte: Próprio autor.

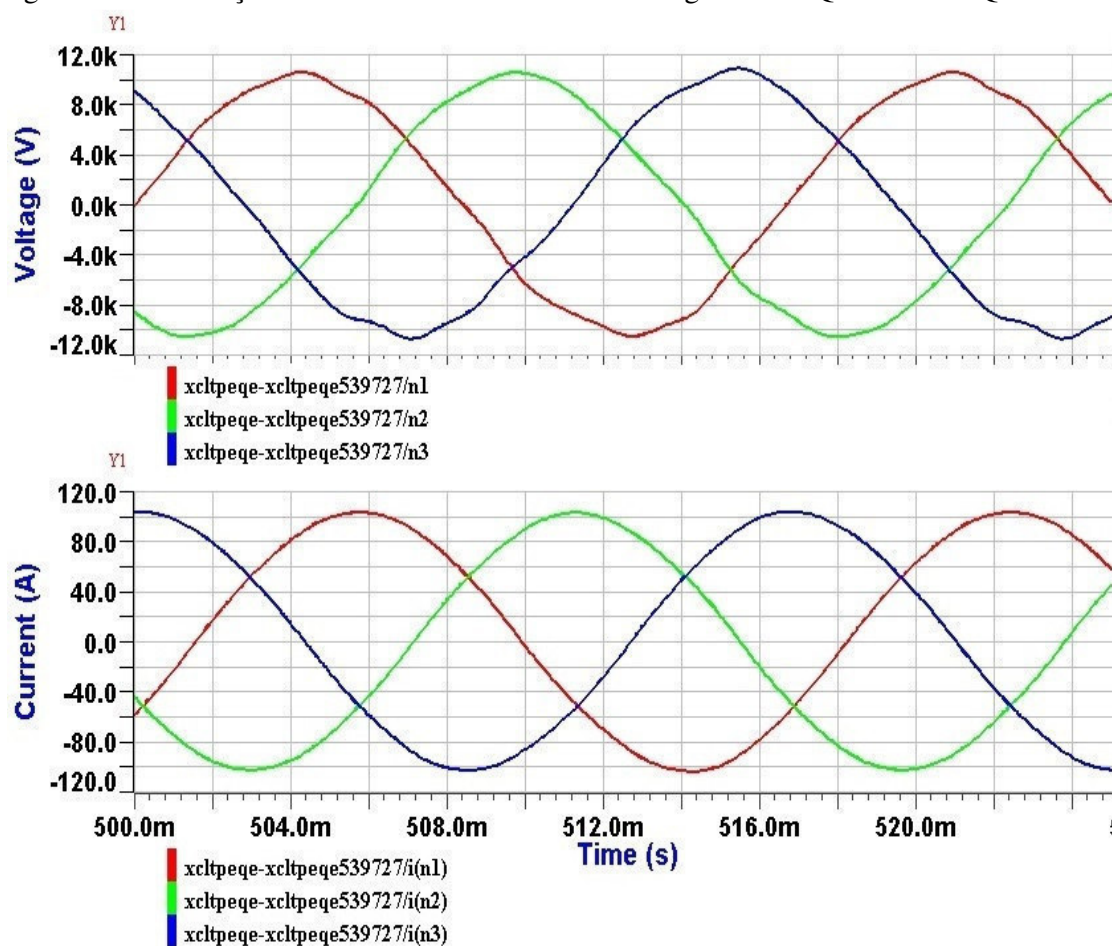
Figura 114- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE512943.



Fonte: Próprio autor.

CLTPEQE-XCLTPEQE539727
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

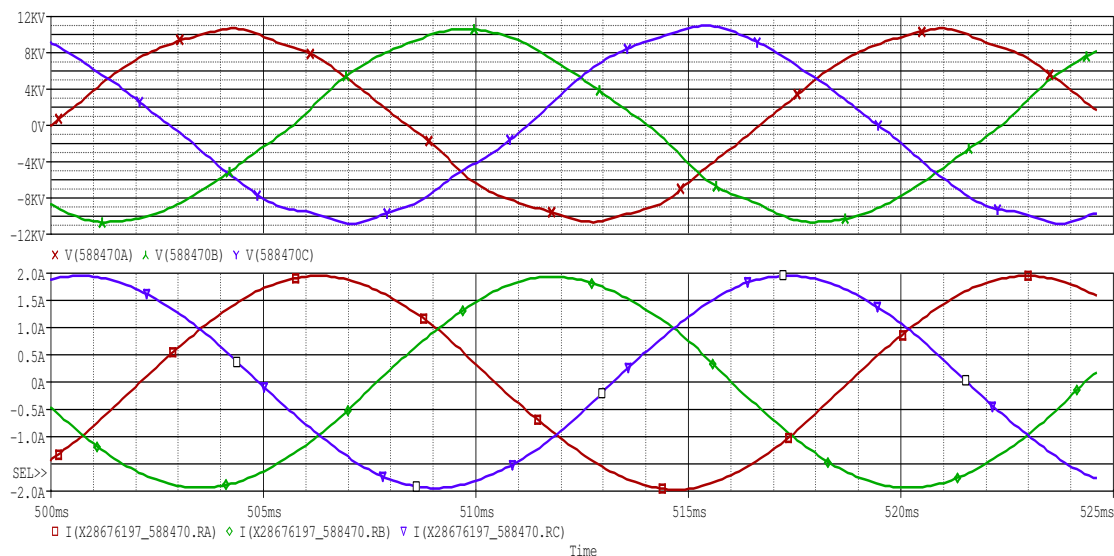
Figura 115- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE539727.



Fonte: Próprio autor.

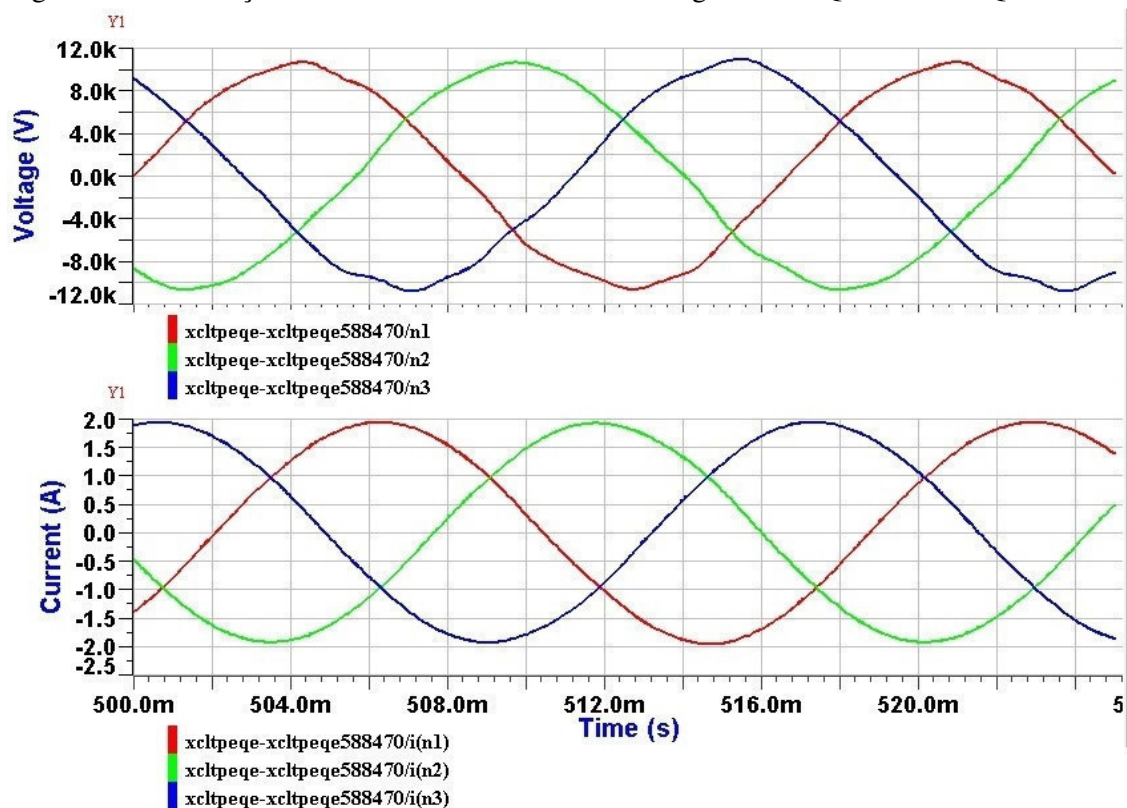
CLTPEQE-XCLTPEQE588470
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 116- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE588470.



Fonte: Próprio autor.

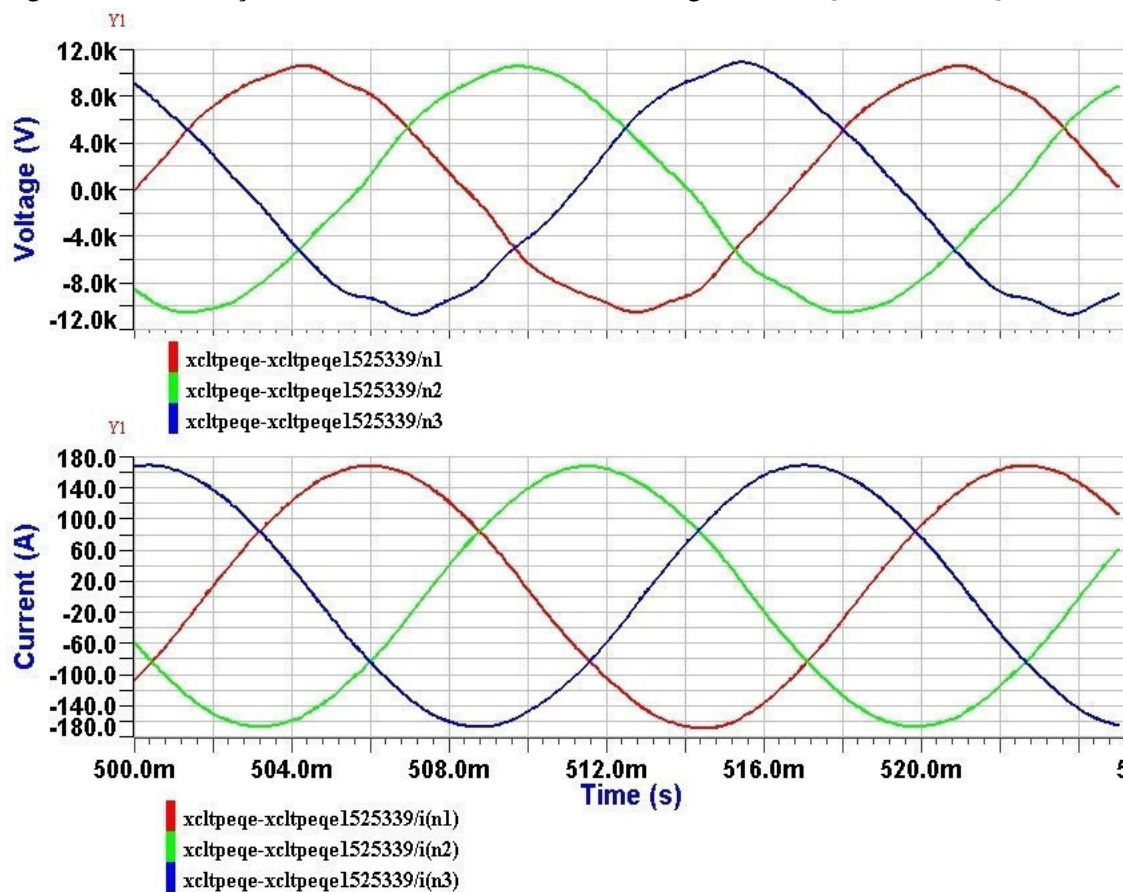
Figura 117- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE588470.



Fonte: Próprio autor.

CLTPEQE-XCLTPEQE1525339
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

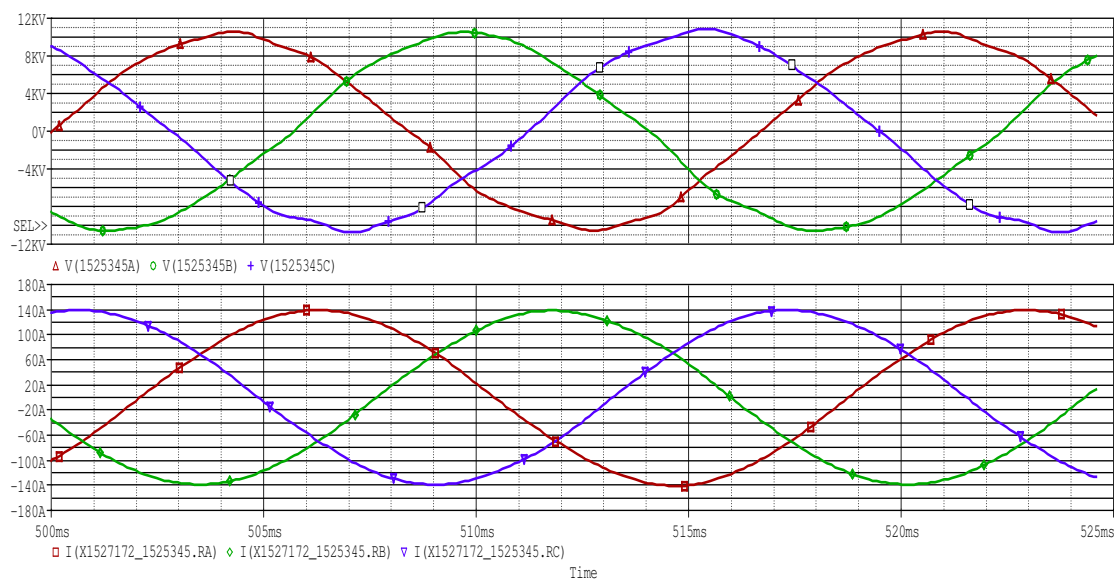
Figura 118- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE1525339.



Fonte: Próprio autor.

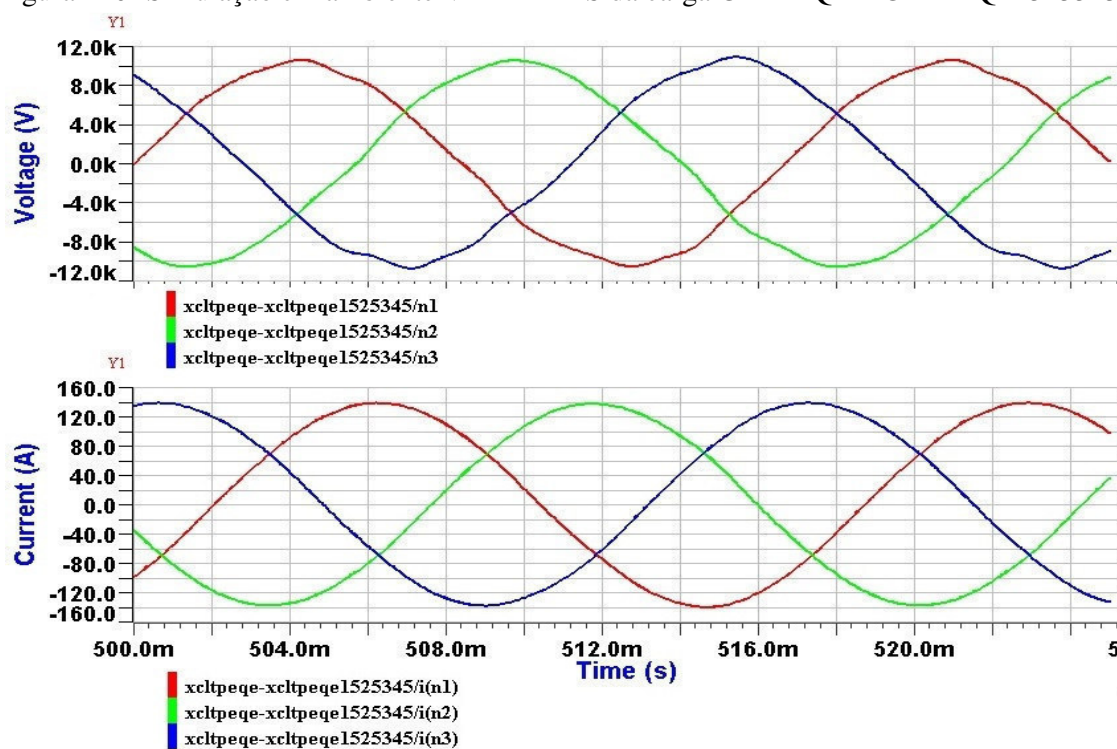
CLTPEQE-XCLTPEQE1525345
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 119- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE1525345.



Fonte: Próprio autor.

Figura 120- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE1525345.



Fonte: Próprio autor.

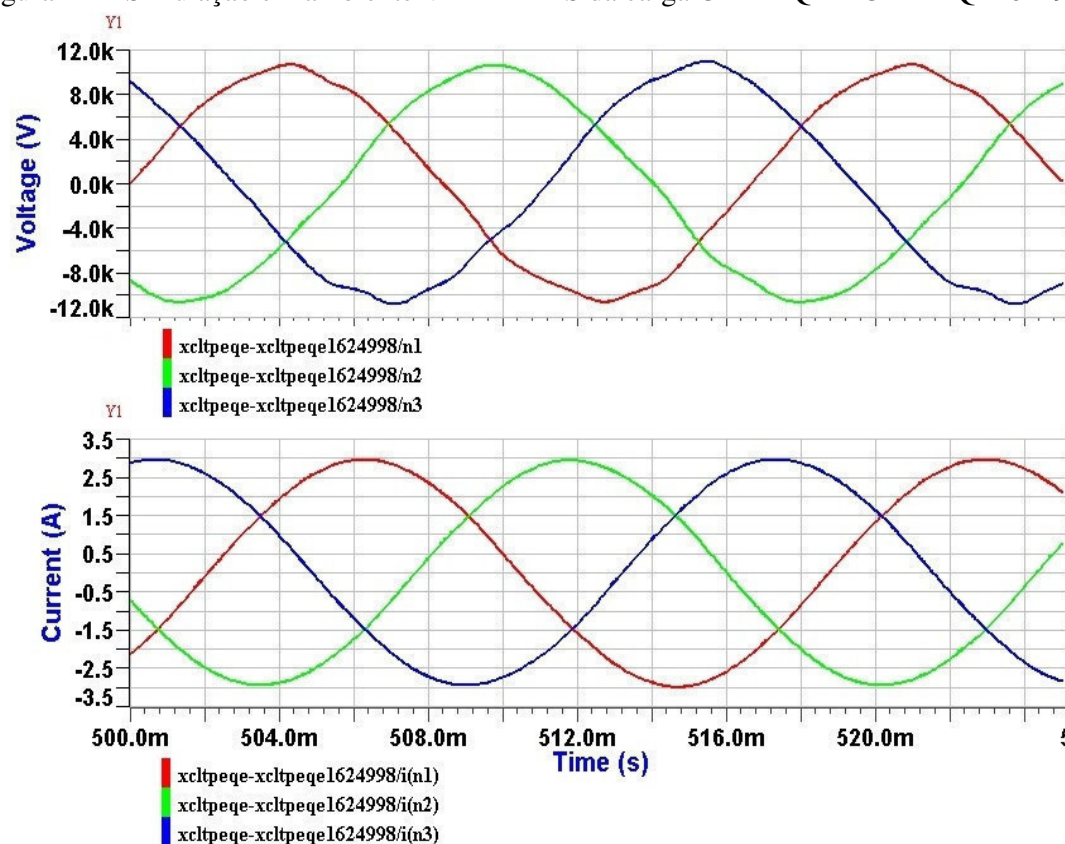
CLTPEQE-XCLTPEQE1624998
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 121- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE1624998.



Fonte: Próprio autor.

Figura 122- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE1624998.



Fonte: Próprio autor.

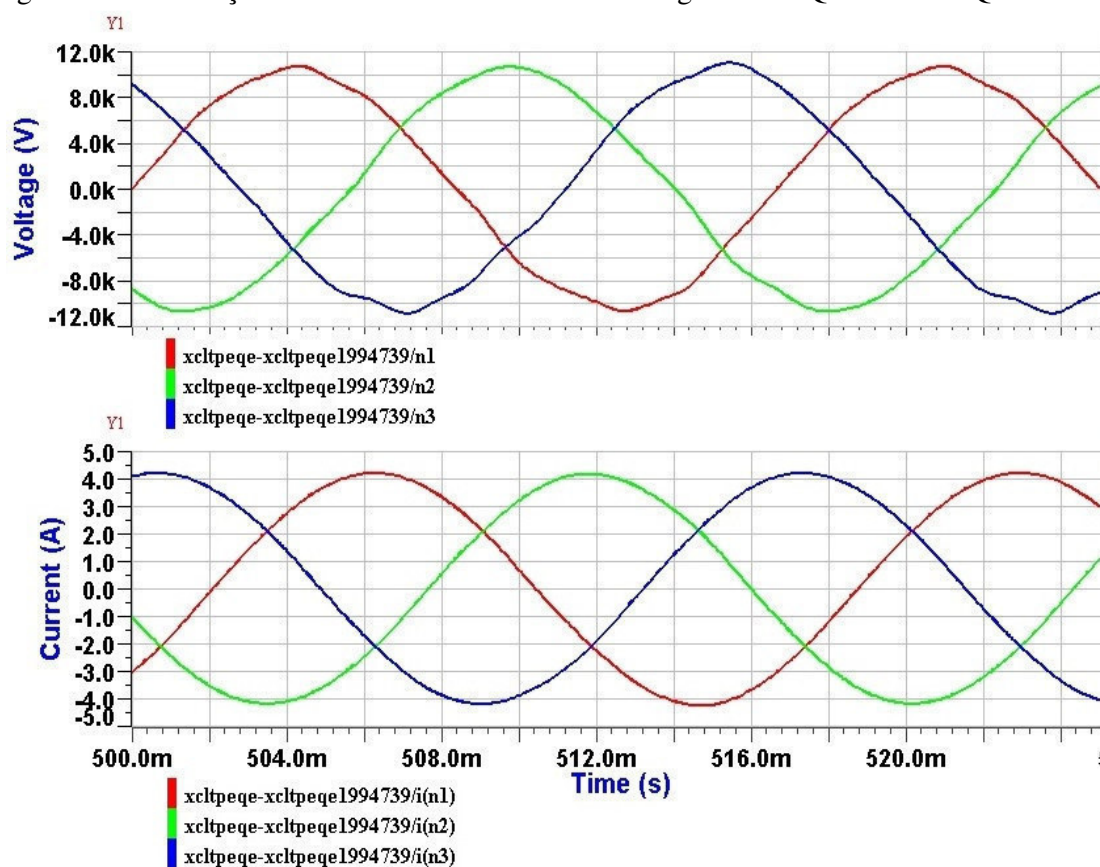
CLTPEQE-XCLTPEQE1994739
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 123- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE1994739.



Fonte: Próprio autor.

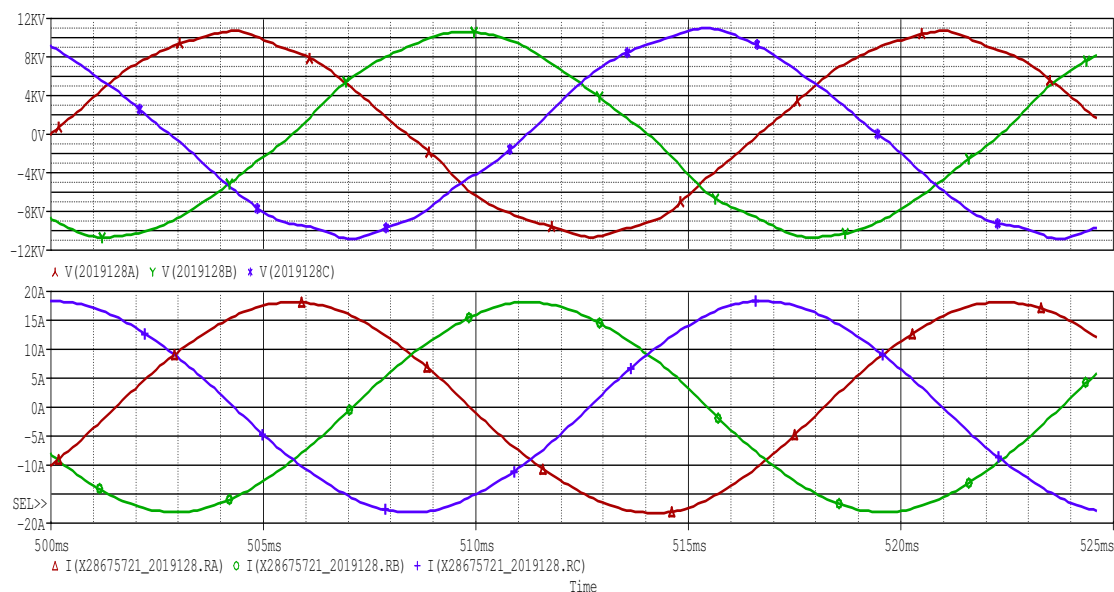
Figura 124- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE1994739.



Fonte: Próprio autor.

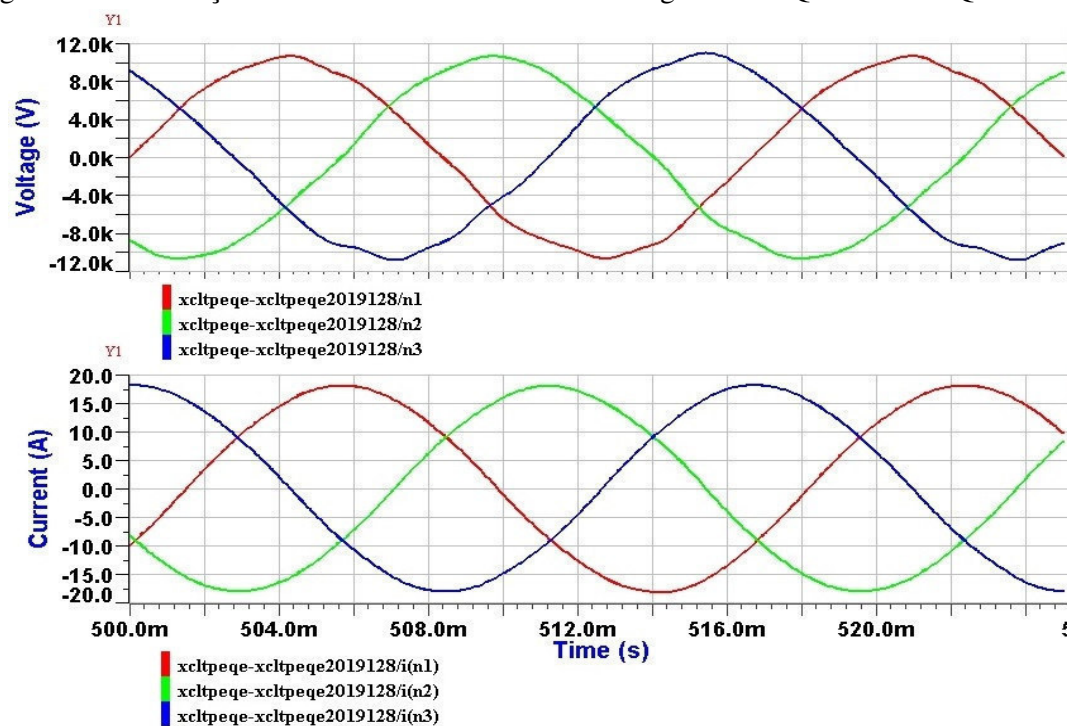
CLTPEQE-XCLTPEQE2019128
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 125- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE2019128.



Fonte: Próprio autor.

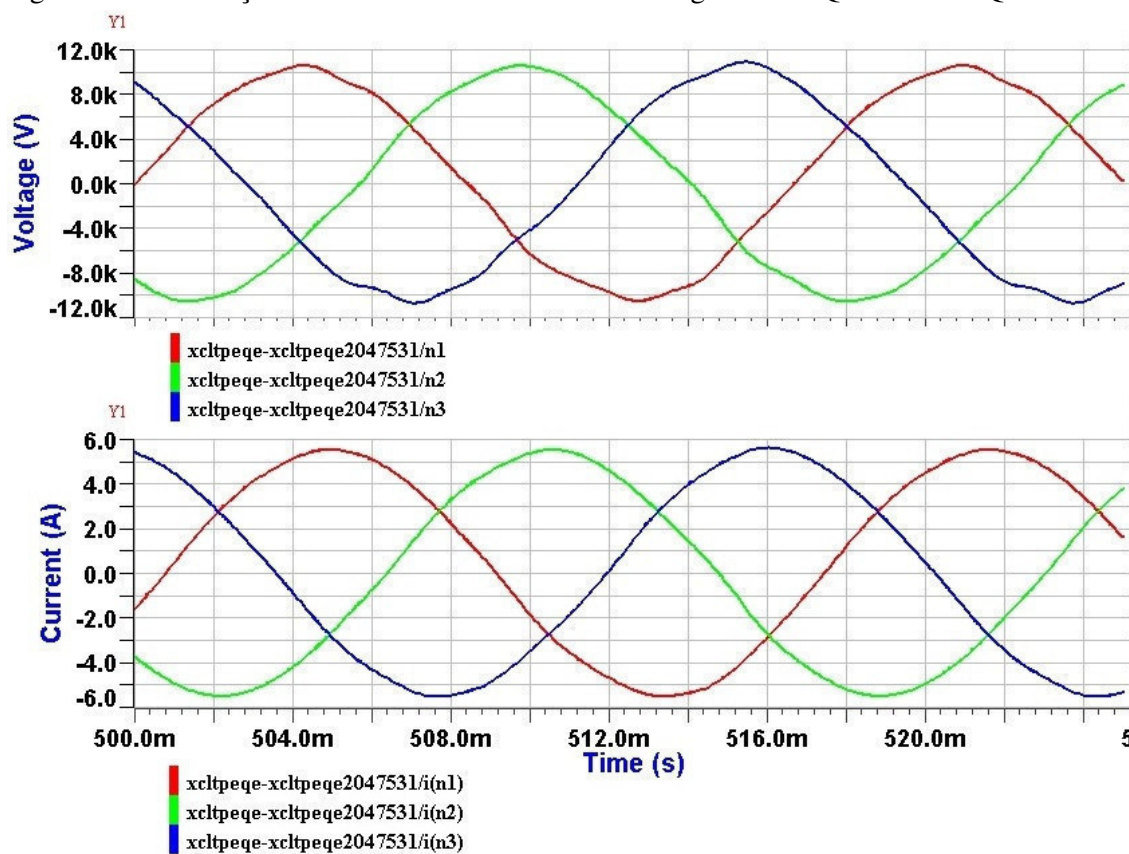
Figura 126- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE2019128.



Fonte: Próprio autor.

CLTPEQE-XCLTPEQE2047531
 Carga indutiva trifásica primária equilibrada (estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

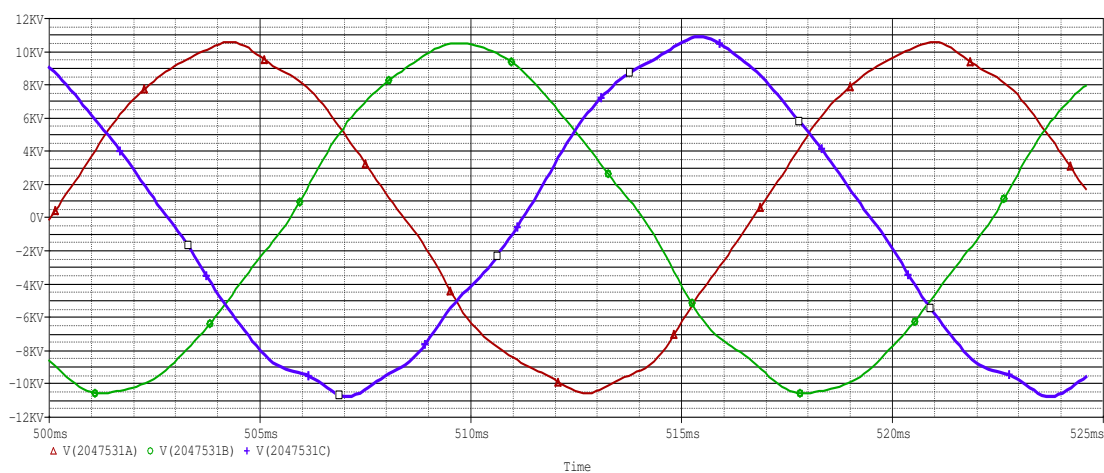
Figura 127- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE2047531.



Fonte: Próprio autor.

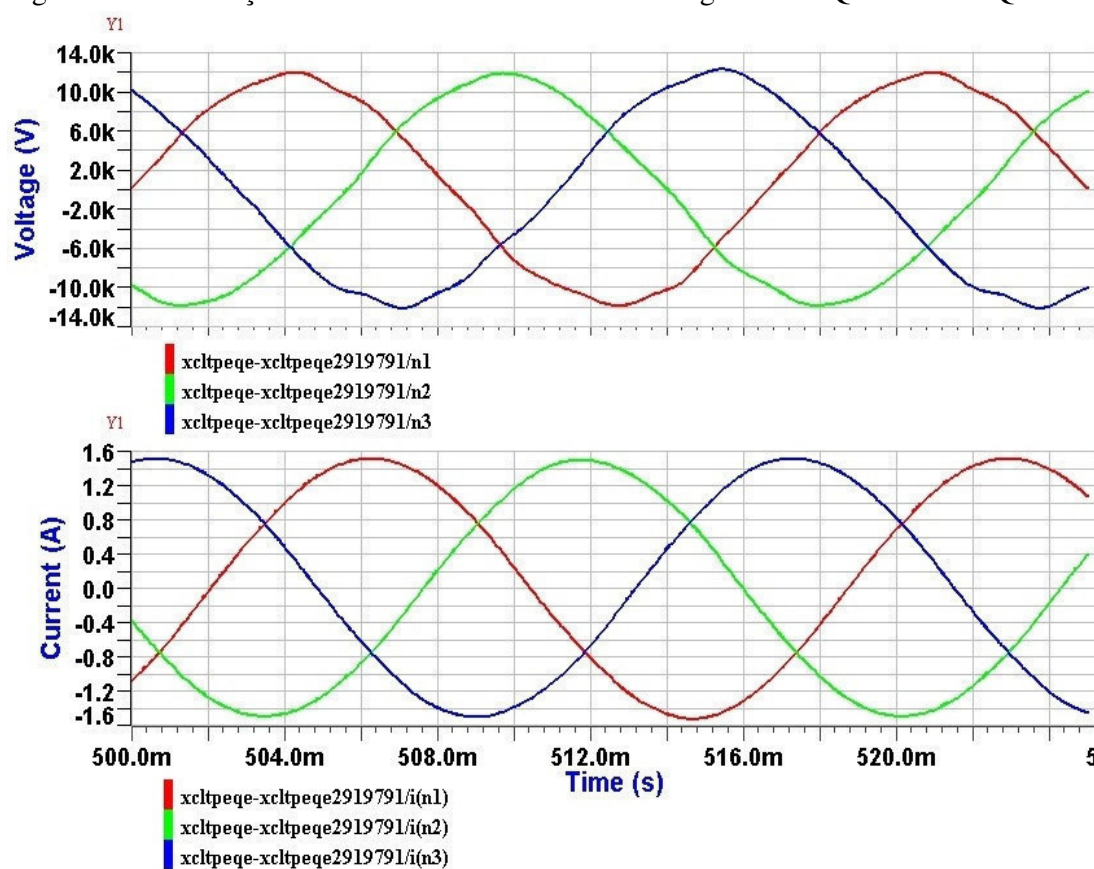
CLTPEQE-XCLTPEQE2919791
 Carga indutiva trifásica primária equilibrada (estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 128- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE2919791.



Fonte: Próprio autor.

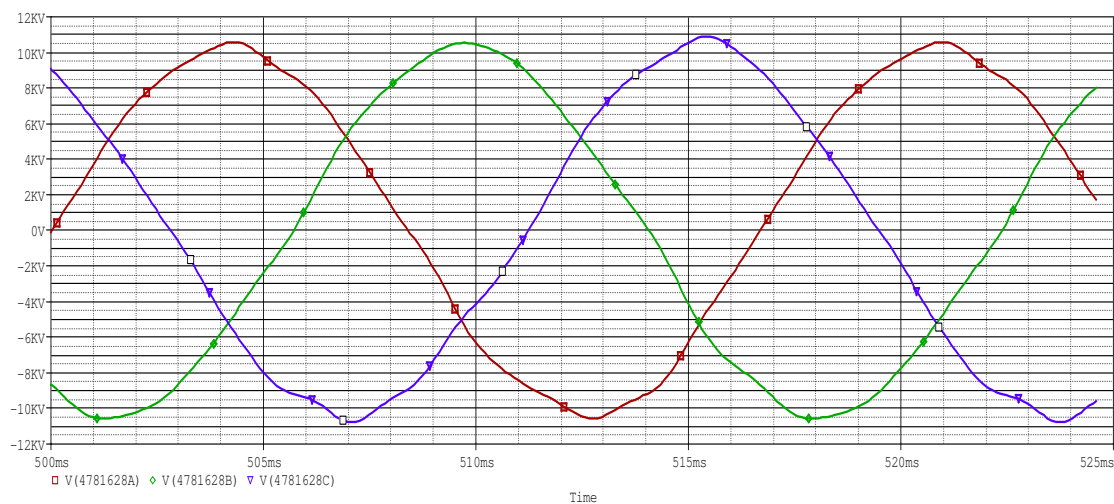
Figura 129- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE2919791.



Fonte: Próprio autor.

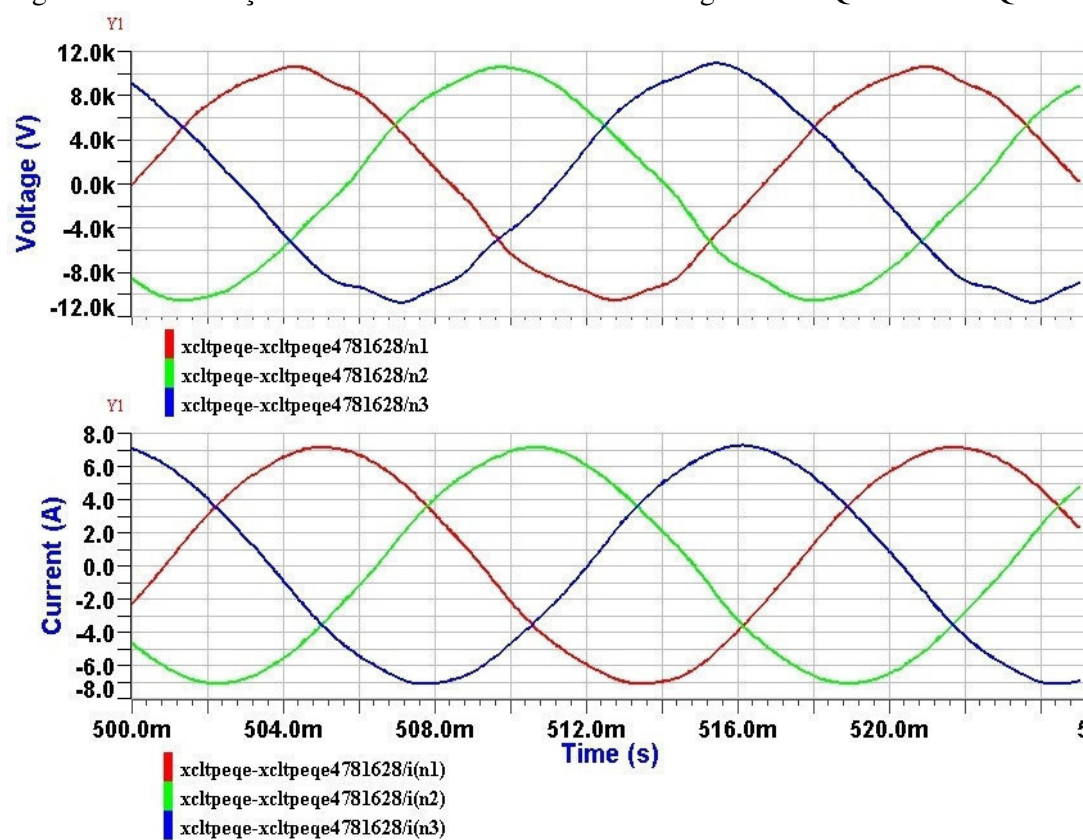
CLTPEQE-XCLTPEQE4781628
 Carga Indutiva Trifásica Primária Equilibrada (Estrela)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 130- Simulação em ambiente Orcad/Pspice da carga CLTPEQE-XCLTPEQE4781628.



Fonte: Próprio autor.

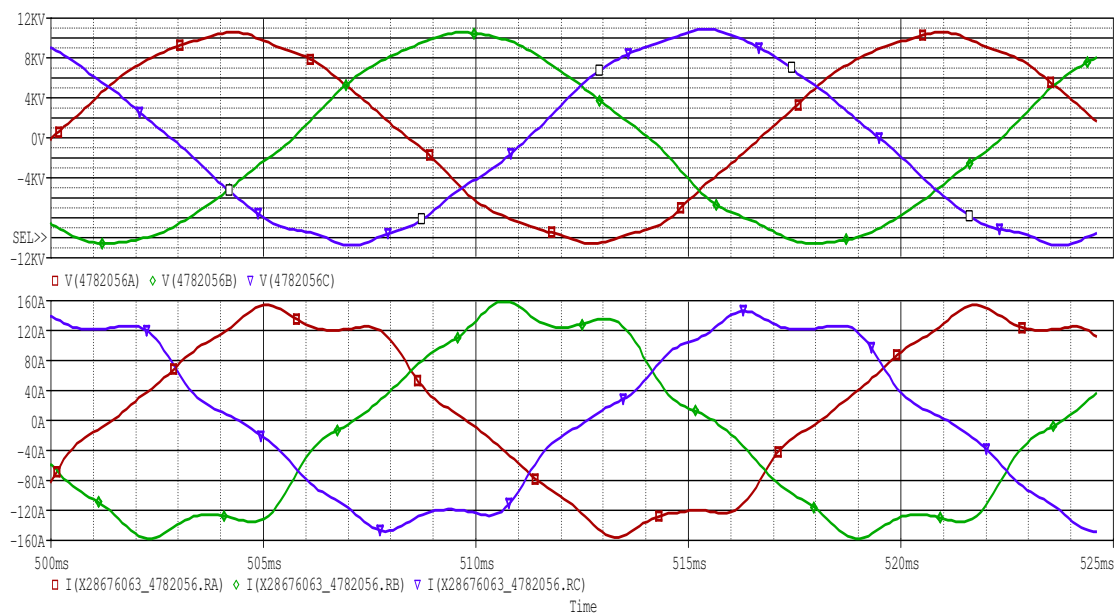
Figura 131- Simulação em ambiente VHDL-AMS da carga CLTPEQE-XCLTPEQE4781628.



Fonte: Próprio autor.

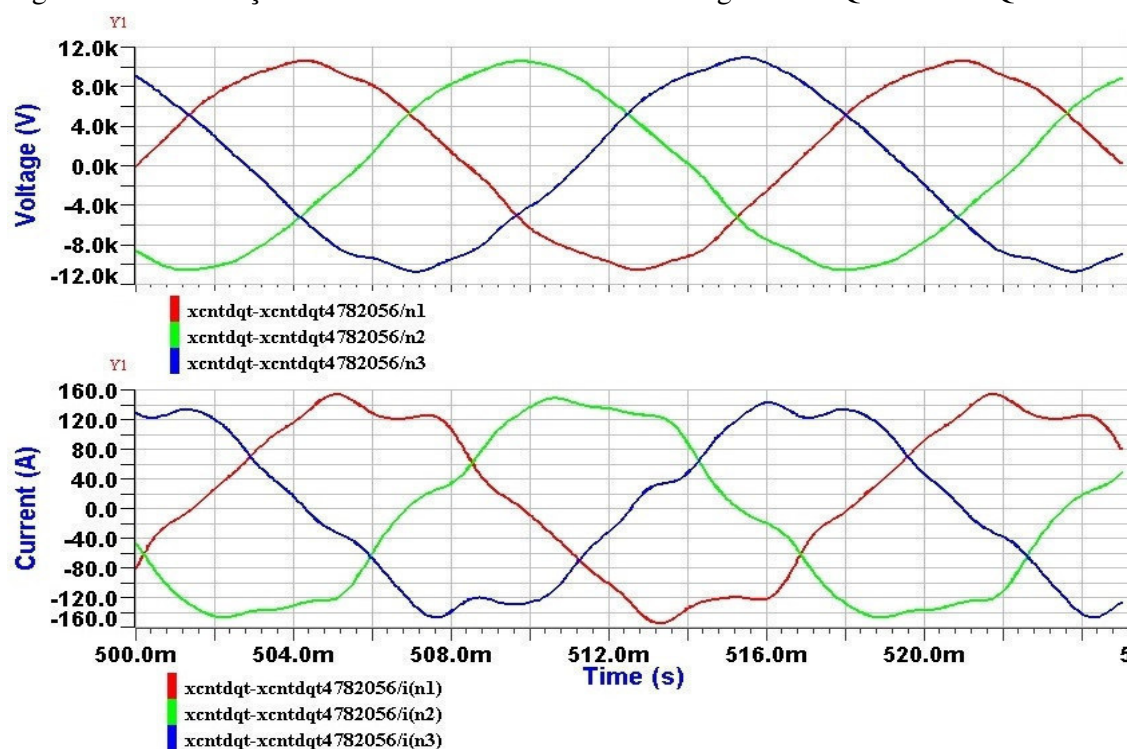
CNTDQT-XCNTDQT4782056
 Carga Não-Linear Trifásica Desequilibrada (Triângulo)
 Simulação para TAP A=8, TAP B=8 e TAP C=8

Figura 132- Simulação em ambiente Orcad/Pspice da carga CNTDQT-XCNTDQT4782056.



Fonte: Próprio autor.

Figura 133- Simulação em ambiente VHDL-AMS da carga CNTDQT-XCNTDQT4782056.



Fonte: Próprio autor.

D.3- CÓDIGO VHDL-AMS DO ALIMENTADOR DE BT DE 26 NÓS

O seguinte código VHDL-AMS foi desenvolvido na plataforma do software SystemVision™ e representa um sistema de múltiplas micro-redes baseado no conceito CERTS, esquematizado na figura 10.

```

001  library ieee;
002  library edulib;
003  library svlib;
004  use ieee.std_logic_1164.all;
005  use ieee.electrical_systems.all;
006  use ieee.mechanical_systems.all;
007  use ieee.fluidic_systems.all;
008  use ieee.thermal_systems.all;
009  use ieee.radiant_systems.all;
010  use work.all;
011  entity MICROGRIDS_4 is
012  end entity MICROGRIDS_4;
013  architecture arch_MICROGRIDS_4 of MICROGRIDS_4 is
014      terminal \$1N14\ : ELECTRICAL;
015      terminal \$1N15\ : ELECTRICAL;
016      terminal \$1N111\ : ELECTRICAL;
017      terminal FASE_A : ELECTRICAL;
018      terminal \$1N101\ : ELECTRICAL;
019      terminal FASE_B : ELECTRICAL;
020      terminal \$1N113\ : ELECTRICAL;
021      terminal FASE_C : ELECTRICAL;
022      terminal \$1N18\ : ELECTRICAL;
023      terminal \$1N5\ : ELECTRICAL;
024      terminal \$1N19\ : ELECTRICAL;
025      terminal \$1N80\ : ELECTRICAL;
026      terminal \$1N6\ : ELECTRICAL;
027      terminal \$1N7\ : ELECTRICAL;
028      terminal \$1N106\ : ELECTRICAL;
029      terminal \$1N9\ : ELECTRICAL;
030      terminal \$1N85\ : ELECTRICAL;
031      terminal \$1N97\ : ELECTRICAL;
032      terminal \$1N20\ : ELECTRICAL;
033      terminal \$1N10\ : ELECTRICAL;
034      terminal \$1N99\ : ELECTRICAL;
035      terminal \$1N11\ : ELECTRICAL;
036      terminal \$1N89\ : ELECTRICAL;
037      terminal \$1N13\ : ELECTRICAL;
038  begin
039      R3A : entity EDULIB.RESISTOR(IDEAL)
040          generic map ( RES => 10000 )
041          port map ( P1 => \$1N11\,
042                  P2 => FASE_A );
043      R3B : entity EDULIB.RESISTOR(IDEAL)
044          generic map ( RES => 10000 )

```

```

045     port map ( P1 => \$1N9\,
046                 P2 => FASE_B );
047 R3C : entity EDULIB.RESISTOR(IDEAL)
048     generic map ( RES => 10000 )
049     port map ( P1 => \$1N10\,
050                 P2 => FASE_C );
051 R2B : entity EDULIB.RESISTOR(IDEAL)
052     generic map ( RES => 10000 )
053     port map ( P1 => \$1N13\,
054                 P2 => FASE_B );
055 R1B : entity EDULIB.RESISTOR(IDEAL)
056     generic map ( RES => 10000 )
057     port map ( P1 => \$1N18\,
058                 P2 => FASE_B );
059 R1C : entity EDULIB.RESISTOR(IDEAL)
060     generic map ( RES => 10000 )
061     port map ( P1 => \$1N19\,
062                 P2 => FASE_C );
063 R2C : entity EDULIB.RESISTOR(IDEAL)
064     generic map ( RES => 10000 )
065     port map ( P1 => \$1N14\,
066                 P2 => FASE_C );
067 L1A : entity EDULIB.INDUCTOR(IDEAL)
068     generic map ( IND => 3950.225,
069                 I_IC => 0.0 )
070     port map ( P1 => ELECTRICAL_REF,
071                 P2 => \$1N20\ );
072 L1B : entity EDULIB.INDUCTOR(IDEAL)
073     generic map ( IND => 3950.225,
074                 I_IC => 0.0 )
075     port map ( P1 => ELECTRICAL_REF,
076                 P2 => \$1N18\ );
077 L1C : entity EDULIB.INDUCTOR(IDEAL)
078     generic map ( IND => 3950.225,
079                 I_IC => 0.0 )
080     port map ( P1 => ELECTRICAL_REF,
081                 P2 => \$1N19\ );
082 L2B : entity EDULIB.INDUCTOR(IDEAL)
083     generic map ( IND => 3950.225,
084                 I_IC => 0.0 )
085     port map ( P1 => ELECTRICAL_REF,
086                 P2 => \$1N13\ );
087 L2C : entity EDULIB.INDUCTOR(IDEAL)
088     generic map ( IND => 3950.225,
089                 I_IC => 0.0 )
090     port map ( P1 => ELECTRICAL_REF,
091                 P2 => \$1N14\ );
092 L3A : entity EDULIB.INDUCTOR(IDEAL)
093     generic map ( IND => 3950.225,
094                 I_IC => 0.0 )
095     port map ( P1 => ELECTRICAL_REF,
096                 P2 => \$1N11\ );
097 L3B : entity EDULIB.INDUCTOR(IDEAL)

```

```

1098     generic map ( IND => 3950.225,
1099                   I_IC => 0.0 )
1100     port map ( P1 => ELECTRICAL_REF,
1101               P2 => \$1N9\ );
1102 L3C : entity EDULIB.INDUCTOR(IDEAL)
1103     generic map ( IND => 3950.225,
1104                   I_IC => 0.0 )
1105     port map ( P1 => ELECTRICAL_REF,
1106               P2 => \$1N10\ );
1107 R2A : entity EDULIB.RESISTOR(IDEAL)
1108     generic map ( RES => 10000 )
1109     port map ( P1 => \$1N15\,
1110               P2 => FASE_A );
1111 L2A : entity EDULIB.INDUCTOR(IDEAL)
1112     generic map ( IND => 3950.225,
1113                   I_IC => 0.0 )
1114     port map ( P1 => ELECTRICAL_REF,
1115               P2 => \$1N15\ );
1116 RCA : entity EDULIB.RESISTOR(IDEAL)
1117     generic map ( RES => 10000 )
1118     port map ( P1 => \$1N5\,
1119               P2 => FASE_A );
1120 LCA : entity EDULIB.INDUCTOR(IDEAL)
1121     generic map ( IND => 3950.225,
1122                   I_IC => 0.0 )
1123     port map ( P1 => ELECTRICAL_REF,
1124               P2 => \$1N5\ );
1125 RCB : entity EDULIB.RESISTOR(IDEAL)
1126     generic map ( RES => 10000 )
1127     port map ( P1 => \$1N7\,
1128               P2 => FASE_B );
1129 RCC : entity EDULIB.RESISTOR(IDEAL)
1130     generic map ( RES => 10000 )
1131     port map ( P1 => \$1N6\,
1132               P2 => FASE_C );
1133 LCB : entity EDULIB.INDUCTOR(IDEAL)
1134     generic map ( IND => 3950.225,
1135                   I_IC => 0.0 )
1136     port map ( P1 => ELECTRICAL_REF,
1137               P2 => \$1N7\ );
1138 LCC : entity EDULIB.INDUCTOR(IDEAL)
1139     generic map ( IND => 3950.225,
1140                   I_IC => 0.0 )
1141     port map ( P1 => ELECTRICAL_REF,
1142               P2 => \$1N6\ );
1143 R1A : entity EDULIB.RESISTOR(IDEAL)
1144     generic map ( RES => 10000 )
1145     port map ( P1 => \$1N20\,
1146               P2 => FASE_A );
1147 PV1A : entity EDULIB.I_SINE(IDEAL)
1148     generic map ( AMPLITUDE => 100.0,
1149                 FREQ => 60.0,
1150                 PHASE => 0.0 )

```

```

151     port map ( POS => FASE_A,
152               NEG => ELECTRICAL_REF );
153 PV2A : entity EDULIB.I_SINE(IDEAL)
154     generic map ( AMPLITUDE => 100.0,
155                 FREQ => 60.0,
156                 PHASE => 0.0 )
157     port map ( POS => FASE_A,
158               NEG => ELECTRICAL_REF );
159 PV3A : entity EDULIB.I_SINE(IDEAL)
160     generic map ( AMPLITUDE => 100.0,
161                 FREQ => 60.0,
162                 PHASE => 0.0 )
163     port map ( POS => FASE_A,
164               NEG => ELECTRICAL_REF );
165 GD2A : entity EDULIB.V_SINE(IDEAL)
166     generic map ( AMPLITUDE => 450.0,
167                 FREQ => 60.0,
168                 PHASE => 0.0 )
169     port map ( POS => \$1N97\,
170               NEG => ELECTRICAL_REF );
171 GD3A : entity EDULIB.V_SINE(IDEAL)
172     generic map ( AMPLITUDE => 450.0,
173                 FREQ => 60.0,
174                 PHASE => 0.0 )
175     port map ( POS => \$1N106\,
176               NEG => ELECTRICAL_REF );
177 GD1B : entity EDULIB.V_SINE(IDEAL)
178     generic map ( AMPLITUDE => 450.0,
179                 FREQ => 60.0,
180                 PHASE => -120.0 )
181     port map ( POS => \$1N85\,
182               NEG => ELECTRICAL_REF );
183 PV1B : entity EDULIB.I_SINE(IDEAL)
184     generic map ( AMPLITUDE => 100.0,
185                 FREQ => 60.0,
186                 PHASE => -120.0 )
187     port map ( POS => FASE_B,
188               NEG => ELECTRICAL_REF );
189 GD3B : entity EDULIB.V_SINE(IDEAL)
190     generic map ( AMPLITUDE => 450.0,
191                 FREQ => 60.0,
192                 PHASE => -120.0 )
193     port map ( POS => \$1N113\,
194               NEG => ELECTRICAL_REF );
195 GD2B : entity EDULIB.V_SINE(IDEAL)
196     generic map ( AMPLITUDE => 450.0,
197                 FREQ => 60.0,
198                 PHASE => -120.0 )
199     port map ( POS => \$1N99\,
200               NEG => ELECTRICAL_REF );
201 PV3B : entity EDULIB.I_SINE(IDEAL)
202     generic map ( AMPLITUDE => 100.0,
203                 FREQ => 60.0,

```

```

204         PHASE => -120.0 )
205     port map ( POS => FASE_B,
206               NEG => ELECTRICAL_REF );
207 PV2B : entity EDULIB.I_SINE(IDEAL)
208     generic map ( AMPLITUDE => 100.0,
209                 FREQ => 60.0,
210                 PHASE => -120.0 )
211     port map ( POS => FASE_B,
212               NEG => ELECTRICAL_REF );
213 GD1C : entity EDULIB.V_SINE(IDEAL)
214     generic map ( AMPLITUDE => 450.0,
215                 FREQ => 60.0,
216                 PHASE => 120.0 )
217     port map ( POS => \$1N89\,
218               NEG => ELECTRICAL_REF );
219
220 GD2C : entity EDULIB.V_SINE(IDEAL)
221     generic map ( AMPLITUDE => 450.0,
222                 FREQ => 60.0,
223                 PHASE => 120.0 )
224     port map ( POS => \$1N101\,
225               NEG => ELECTRICAL_REF );
226 GD3C : entity EDULIB.V_SINE(IDEAL)
227     generic map ( AMPLITUDE => 450.0,
228                 FREQ => 60.0,
229                 PHASE => 120.0 )
230     port map ( POS => \$1N111\,
231               NEG => ELECTRICAL_REF );
232 PV1C : entity EDULIB.I_SINE(IDEAL)
233     generic map ( AMPLITUDE => 100.0,
234                 FREQ => 60.0,
235                 PHASE => 120.0 )
236     port map ( POS => FASE_C,
237               NEG => ELECTRICAL_REF );
238 PV2C : entity EDULIB.I_SINE(IDEAL)
239     generic map ( AMPLITUDE => 100.0,
240                 FREQ => 60.0,
241                 PHASE => 120.0 )
242     port map ( POS => FASE_C,
243               NEG => ELECTRICAL_REF );
244 PV3C : entity EDULIB.I_SINE(IDEAL)
245     generic map ( AMPLITUDE => 100.0,
246                 FREQ => 60.0,
247                 PHASE => 120.0 )
248     port map ( POS => FASE_C,
249               NEG => ELECTRICAL_REF );
250 GD1A : entity EDULIB.V_SINE(IDEAL)
251     generic map ( AMPLITUDE => 450.0,
252                 FREQ => 60.0,
253                 PHASE => 0.0 )
254     port map ( POS => \$1N80\,
255               NEG => ELECTRICAL_REF );
256 R123A : entity EDULIB.RESISTOR(IDEAL)

```

```

257     generic map ( RES => 0.3 )
258     port map ( P1 => FASE_A,
259               P2 => \$1N80\ );
260 R123B : entity EDULIB.RESISTOR(IDEAL)
261     generic map ( RES => 0.3 )
262     port map ( P1 => FASE_B,
263               P2 => \$1N85\ );
264 R123C : entity EDULIB.RESISTOR(IDEAL)
265     generic map ( RES => 0.3 )
266     port map ( P1 => FASE_C,
267               P2 => \$1N89\ );
268 R223A : entity EDULIB.RESISTOR(IDEAL)
269     generic map ( RES => 0.3 )
270     port map ( P1 => FASE_A,
271               P2 => \$1N97\ );
272 R223B : entity EDULIB.RESISTOR(IDEAL)
273     generic map ( RES => 0.3 )
274     port map ( P1 => FASE_B,
275               P2 => \$1N99\ );
276 R223C : entity EDULIB.RESISTOR(IDEAL)
277     generic map ( RES => 0.3 )
278     port map ( P1 => FASE_C,
279               P2 => \$1N101\ );
280 R333A : entity EDULIB.RESISTOR(IDEAL)
281     generic map ( RES => 0.3 )
282     port map ( P1 => FASE_A,
283               P2 => \$1N106\ );
284 R333B : entity EDULIB.RESISTOR(IDEAL)
285     generic map ( RES => 0.3 )
286     port map ( P1 => FASE_B,
287               P2 => \$1N113\ );
288 R333C : entity EDULIB.RESISTOR(IDEAL)
289     generic map ( RES => 0.3 )
290     port map ( P1 => FASE_C,
291               P2 => \$1N111\ );
292 end architecture arch_MICROGRIDS_4;

```

D.4- CÓDIGO VHDL-AMS DO ALIMENTADOR DE MT DE 925 NÓS

O seguinte código fonte VHDL-AMS foi desenvolvido na plataforma do software SystemVision™ e representa um alimentador de uma rede de distribuição primaria de 14,4 kV (Caso de Estudo), esquematizado na figura 11.

```

001  library ieee;
002  library edulib;
003  library svlib;
004  use ieee.std_logic_1164.all;
005  use ieee.electrical_systems.all;
006  use ieee.mechanical_systems.all;
007  use ieee.fluidic_systems.all;
008  use ieee.thermal_systems.all;
009  use ieee.radiant_systems.all;
010  use work.all;
011
012  entity FEDEER is
013  end entity FEDEER;
014
015  architecture arch_FEDEER of FEDEER is
016      terminal \$1N69\ : ELECTRICAL;
017      terminal \$1N58\ : ELECTRICAL;
018      terminal \$1N47\ : ELECTRICAL;
019      terminal \$1N36\ : ELECTRICAL;
020      terminal \$1N275\ : ELECTRICAL;
021      terminal \$1N25\ : ELECTRICAL;
022      terminal \$1N187\ : ELECTRICAL;
023      terminal \$1N176\ : ELECTRICAL;
024      terminal \$1N165\ : ELECTRICAL;
025      terminal \$1N154\ : ELECTRICAL;
026      terminal \$1N143\ : ELECTRICAL;
027      terminal \$1N14\ : ELECTRICAL;
028      terminal \$1N132\ : ELECTRICAL;
029      terminal \$1N121\ : ELECTRICAL;
030      terminal \$1N110\ : ELECTRICAL;
031      terminal \$1N59\ : ELECTRICAL;
032      terminal \$1N48\ : ELECTRICAL;
033      terminal \$1N37\ : ELECTRICAL;
034      terminal \$1N276\ : ELECTRICAL;
035      terminal \$1N26\ : ELECTRICAL;
036      terminal \$1N188\ : ELECTRICAL;
037      terminal \$1N177\ : ELECTRICAL;
038      terminal \$1N166\ : ELECTRICAL;
039      terminal \$1N155\ : ELECTRICAL;
040      terminal \$1N15\ : ELECTRICAL;
041      terminal \$1N144\ : ELECTRICAL;
042      terminal \$1N133\ : ELECTRICAL;
043      terminal \$1N122\ : ELECTRICAL;
044      terminal \$1N111\ : ELECTRICAL;
045      terminal \$1N100\ : ELECTRICAL;
046      terminal \$1N1\ : ELECTRICAL;

```

```
047      terminal \S1N49\ : ELECTRICAL;
048      terminal \S1N38\ : ELECTRICAL;
049      terminal \S1N277\ : ELECTRICAL;
050      terminal \S1N27\ : ELECTRICAL;
051      terminal \S1N2\ : ELECTRICAL;
052      terminal \S1N189\ : ELECTRICAL;
053      terminal \S1N178\ : ELECTRICAL;
054      terminal \S1N167\ : ELECTRICAL;
055      terminal \S1N16\ : ELECTRICAL;
056      terminal \S1N156\ : ELECTRICAL;
057      terminal \S1N145\ : ELECTRICAL;
058      terminal \S1N134\ : ELECTRICAL;
059      terminal \S1N123\ : ELECTRICAL;
060      terminal \S1N112\ : ELECTRICAL;
061      terminal \S1N101\ : ELECTRICAL;
062      terminal \S1N39\ : ELECTRICAL;
063      terminal \S1N3\ : ELECTRICAL;
064      terminal \S1N28\ : ELECTRICAL;
065      terminal \S1N179\ : ELECTRICAL;
066      terminal \S1N17\ : ELECTRICAL;
067      terminal \S1N168\ : ELECTRICAL;
068      terminal \S1N157\ : ELECTRICAL;
069      terminal \S1N146\ : ELECTRICAL;
070      terminal \S1N135\ : ELECTRICAL;
071      terminal \S1N124\ : ELECTRICAL;
072      terminal \S1N113\ : ELECTRICAL;
073      terminal \S1N102\ : ELECTRICAL;
074      terminal \S1N4\ : ELECTRICAL;
075      terminal \S1N29\ : ELECTRICAL;
076      terminal \S1N279\ : ELECTRICAL;
077      terminal \S1N18\ : ELECTRICAL;
078      terminal \S1N169\ : ELECTRICAL;
079      terminal \S1N158\ : ELECTRICAL;
080      terminal \S1N147\ : ELECTRICAL;
081      terminal \S1N136\ : ELECTRICAL;
082      terminal \S1N125\ : ELECTRICAL;
083      terminal \S1N114\ : ELECTRICAL;
084      terminal \S1N103\ : ELECTRICAL;
085      terminal \S1N90\ : ELECTRICAL;
086      terminal \S1N5\ : ELECTRICAL;
087      terminal \S1N19\ : ELECTRICAL;
088      terminal \S1N159\ : ELECTRICAL;
089      terminal \S1N148\ : ELECTRICAL;
090      terminal \S1N137\ : ELECTRICAL;
091      terminal \S1N126\ : ELECTRICAL;
092      terminal \S1N115\ : ELECTRICAL;
093      terminal \S1N104\ : ELECTRICAL;
094      terminal \S1N91\ : ELECTRICAL;
095      terminal \S1N80\ : ELECTRICAL;
096      terminal \S1N6\ : ELECTRICAL;
097      terminal \S1N149\ : ELECTRICAL;
098      terminal \S1N138\ : ELECTRICAL;
099      terminal \S1N127\ : ELECTRICAL;
```

```
100    terminal \ $1N116\ : ELECTRICAL;
101    terminal \ $1N105\ : ELECTRICAL;
102    terminal \ $1N92\ : ELECTRICAL;
103    terminal \ $1N81\ : ELECTRICAL;
104    terminal \ $1N70\ : ELECTRICAL;
105    terminal \ $1N7\ : ELECTRICAL;
106    terminal \ $1N139\ : ELECTRICAL;
107    terminal \ $1N128\ : ELECTRICAL;
108    terminal \ $1N117\ : ELECTRICAL;
109    terminal \ $1N106\ : ELECTRICAL;
110    terminal \ $1N93\ : ELECTRICAL;
111    terminal \ $1N82\ : ELECTRICAL;
112    terminal \ $1N8\ : ELECTRICAL;
113    terminal \ $1N71\ : ELECTRICAL;
114    terminal \ $1N60\ : ELECTRICAL;
115    terminal \ $1N129\ : ELECTRICAL;
116    terminal \ $1N118\ : ELECTRICAL;
117    terminal \ $1N107\ : ELECTRICAL;
118    terminal \ $1N94\ : ELECTRICAL;
119    terminal \ $1N9\ : ELECTRICAL;
120    terminal \ $1N83\ : ELECTRICAL;
121    terminal \ $1N72\ : ELECTRICAL;
122    terminal \ $1N61\ : ELECTRICAL;
123    terminal \ $1N50\ : ELECTRICAL;
124    terminal \ $1N190\ : ELECTRICAL;
125    terminal \ $1N119\ : ELECTRICAL;
126    terminal \ $1N108\ : ELECTRICAL;
127    terminal \ $1N95\ : ELECTRICAL;
128    terminal \ $1N84\ : ELECTRICAL;
129    terminal \ $1N73\ : ELECTRICAL;
130    terminal \ $1N62\ : ELECTRICAL;
131    terminal \ $1N51\ : ELECTRICAL;
132    terminal \ $1N40\ : ELECTRICAL;
133    terminal \ $1N191\ : ELECTRICAL;
134    terminal \ $1N180\ : ELECTRICAL;
135    terminal \ $1N109\ : ELECTRICAL;
136    terminal VFASE_A: ELECTRICAL;
137    terminal \ $1N96\ : ELECTRICAL;
138    terminal \ $1N85\ : ELECTRICAL;
139    terminal \ $1N74\ : ELECTRICAL;
140    terminal \ $1N63\ : ELECTRICAL;
141    terminal \ $1N52\ : ELECTRICAL;
142    terminal \ $1N41\ : ELECTRICAL;
143    terminal \ $1N30\ : ELECTRICAL;
144    terminal \ $1N280\ : ELECTRICAL;
145    terminal \ $1N192\ : ELECTRICAL;
146    terminal \ $1N170\ : ELECTRICAL;
147    terminal VFASE_B: ELECTRICAL;
148    terminal \ $1N97\ : ELECTRICAL;
149    terminal \ $1N86\ : ELECTRICAL;
150    terminal \ $1N75\ : ELECTRICAL;
151    terminal \ $1N64\ : ELECTRICAL;
152    terminal \ $1N53\ : ELECTRICAL;
```

```
153    terminal \ $1N42\ : ELECTRICAL;
154    terminal \ $1N31\ : ELECTRICAL;
155    terminal \ $1N281\ : ELECTRICAL;
156    terminal \ $1N20\ : ELECTRICAL;
157    terminal \ $1N193\ : ELECTRICAL;
158    terminal \ $1N182\ : ELECTRICAL;
159    terminal \ $1N171\ : ELECTRICAL;
160    terminal \ $1N160\ : ELECTRICAL;
161    terminal VFASE_C: ELECTRICAL;
162    terminal \ $1N98\ : ELECTRICAL;
163    terminal \ $1N87\ : ELECTRICAL;
164    terminal \ $1N76\ : ELECTRICAL;
165    terminal \ $1N65\ : ELECTRICAL;
166    terminal \ $1N54\ : ELECTRICAL;
167    terminal \ $1N43\ : ELECTRICAL;
168    terminal \ $1N32\ : ELECTRICAL;
169    terminal \ $1N282\ : ELECTRICAL;
170    terminal \ $1N21\ : ELECTRICAL;
171    terminal \ $1N183\ : ELECTRICAL;
172    terminal \ $1N172\ : ELECTRICAL;
173    terminal \ $1N161\ : ELECTRICAL;
174    terminal \ $1N150\ : ELECTRICAL;
175    terminal \ $1N10\ : ELECTRICAL;
176    terminal \ $1N99\ : ELECTRICAL;
177    terminal \ $1N88\ : ELECTRICAL;
178    terminal \ $1N77\ : ELECTRICAL;
179    terminal \ $1N66\ : ELECTRICAL;
180    terminal \ $1N55\ : ELECTRICAL;
181    terminal \ $1N44\ : ELECTRICAL;
182    terminal \ $1N33\ : ELECTRICAL;
183    terminal \ $1N283\ : ELECTRICAL;
184    terminal \ $1N184\ : ELECTRICAL;
185    terminal \ $1N173\ : ELECTRICAL;
186    terminal \ $1N162\ : ELECTRICAL;
187    terminal \ $1N151\ : ELECTRICAL;
188    terminal \ $1N140\ : ELECTRICAL;
189    terminal \ $1N11\ : ELECTRICAL;
190    terminal \ $1N89\ : ELECTRICAL;
191    terminal \ $1N78\ : ELECTRICAL;
192    terminal \ $1N67\ : ELECTRICAL;
193    terminal \ $1N56\ : ELECTRICAL;
194    terminal \ $1N45\ : ELECTRICAL;
195    terminal \ $1N34\ : ELECTRICAL;
196    terminal \ $1N284\ : ELECTRICAL;
197    terminal \ $1N23\ : ELECTRICAL;
198    terminal \ $1N185\ : ELECTRICAL;
199    terminal \ $1N174\ : ELECTRICAL;
200    terminal \ $1N163\ : ELECTRICAL;
201    terminal \ $1N152\ : ELECTRICAL;
202    terminal \ $1N141\ : ELECTRICAL;
203    terminal \ $1N130\ : ELECTRICAL;
204    terminal \ $1N12\ : ELECTRICAL;
205    terminal \ $1N79\ : ELECTRICAL;
```

```

206     terminal \$1N68\ : ELECTRICAL;
207     terminal \$1N57\ : ELECTRICAL;
208     terminal \$1N46\ : ELECTRICAL;
209     terminal \$1N35\ : ELECTRICAL;
210     terminal \$1N285\ : ELECTRICAL;
211     terminal \$1N24\ : ELECTRICAL;
212     terminal \$1N186\ : ELECTRICAL;
213     terminal \$1N175\ : ELECTRICAL;
214     terminal \$1N164\ : ELECTRICAL;
215     terminal \$1N153\ : ELECTRICAL;
216     terminal \$1N142\ : ELECTRICAL;
217     terminal \$1N131\ : ELECTRICAL;
218     terminal \$1N13\ : ELECTRICAL;
219     terminal \$1N120\ : ELECTRICAL;
220
221 begin
222
223     \CCTPEQT-XCCTPEQT2220655\ : entity WORK.CCTPEQT-
224 XCCTPEQT2220655 (ARCH_CCTPEQT-XCCTPEQT2220655)
225         port map ( \1\ => \$1N54\,
226                   \2\ => \$1N55\,
227                   \3\ => \$1N56\ );
228
229     \LINHA_PRI-X1329777_512944\ : entity WORK.LINHA_PRI-
230 X1329777_512944 (ARCH_LINHA_PRI-X1329777_512944)
231         port map ( \1A\ => \$1N79\,
232                   \1B\ => \$1N80\,
233                   \1C\ => \$1N71\,
234                   \2A\ => \$1N184\,
235                   \2B\ => \$1N183\,
236                   \2C\ => \$1N182\ );
237
238     \LINHA_PRI-X512943_512944\ : entity WORK.LINHA_PRI-
239 X512943_512944 (ARCH_LINHA_PRI-X512943_512944)
240         port map ( \1A\ => \$1N187\,
241                   \1B\ => \$1N186\,
242                   \1C\ => \$1N185\,
243                   \2A\ => \$1N184\,
244                   \2B\ => \$1N183\,
245                   \2C\ => \$1N182\ );
246
247     \LINHA_PRI-X512943_1747204\ : entity WORK.LINHA_PRI-
248 X512943_1747204 (ARCH_LINHA_PRI-X512943_1747204)
249         port map ( \1A\ => \$1N187\,
250                   \1B\ => \$1N186\,
251                   \1C\ => \$1N185\,
252                   \2A\ => \$1N92\,
253                   \2B\ => \$1N93\,
254                   \2C\ => \$1N91\ );
255
256     \LINHA_PRI-X2919791_1329777\ : entity WORK.LINHA_PRI-
257 X2919791_1329777 (ARCH_LINHA_PRI-X2919791_1329777)
258         port map ( \1A\ => \$1N109\,

```

```

259          \1B\ => \$1N111\,
260          \1C\ => \$1N110\,
261          \2A\ => \$1N79\,
262          \2B\ => \$1N80\,
263          \2C\ => \$1N71\ );
264
265      \LINHA_PRI-X296189_388608\ : entity WORK.LINHA_PRI-
266      X296189_388608 (ARCH_LINHA_PRI-X296189_388608)
267      port map ( \1A\ => \$1N21\,
268                \1B\ => \$1N20\,
269                \1C\ => \$1N50\,
270                \2A\ => \$1N112\,
271                \2B\ => \$1N113\,
272                \2C\ => \$1N114\ );
273
274      \LINHA_PRI-X960320_1994738\ : entity WORK.LINHA_PRI-
275      X960320_1994738 (ARCH_LINHA_PRI-X960320_1994738)
276      port map ( \1A\ => \$1N192\,
277                \1B\ => \$1N191\,
278                \1C\ => \$1N190\,
279                \2A\ => \$1N115\,
280                \2B\ => \$1N116\,
281                \2C\ => \$1N117\ );
282
283      \LINHA_PRI-X1994738_1994739\ : entity WORK.LINHA_PRI-
284      X1994738_1994739 (ARCH_LINHA_PRI-X1994738_1994739)
285      port map ( \1A\ => \$1N115\,
286                \1B\ => \$1N116\,
287                \1C\ => \$1N117\,
288                \2A\ => \$1N9\,
289                \2B\ => \$1N8\,
290                \2C\ => \$1N7\ );
291
292      \LINHA_PRI-X3193575_1178293\ : entity WORK.LINHA_PRI-
293      X3193575_1178293 (ARCH_LINHA_PRI-X3193575_1178293)
294      port map ( \1A\ => \$1N99\,
295                \1B\ => \$1N98\,
296                \1C\ => \$1N97\,
297                \2A\ => \$1N1\,
298                \2B\ => \$1N2\,
299                \2C\ => \$1N3\ );
300
301      \LINHA_PRI-X28760967_1060058\ : entity WORK.LINHA_PRI-
302      X28760967_1060058 (ARCH_LINHA_PRI-X28760967_1060058)
303      port map ( \1A\ => \$1N96\,
304                \1B\ => \$1N95\,
305                \1C\ => \$1N94\,
306                \2A\ => \$1N118\,
307                \2B\ => \$1N119\,
308                \2C\ => \$1N120\ );
309
310      \LINHA_PRI-X489128_1060058\ : entity WORK.LINHA_PRI-
311      X489128_1060058 (ARCH_LINHA_PRI-X489128_1060058)

```

```

312     port map ( \1A\ => \$1N100\,
313               \1B\ => \$1N101\,
314               \1C\ => \$1N102\,
315               \2A\ => \$1N118\,
316               \2B\ => \$1N119\,
317               \2C\ => \$1N120\ );
318
319     \LINHA_PRI-X2045999_1527172\ : entity WORK.LINHA_PRI-
320 X2045999_1527172 (ARCH_LINHA_PRI-X2045999_1527172)
321     port map ( \1A\ => \$1N121\,
322               \1B\ => \$1N122\,
323               \1C\ => \$1N123\,
324               \2A\ => \$1N105\,
325               \2B\ => \$1N104\,
326               \2C\ => \$1N103\ );
327
328     \LINHA_PRI-X489128_489129\ : entity WORK.LINHA_PRI-
329 X489128_489129 (ARCH_LINHA_PRI-X489128_489129)
330     port map ( \1A\ => \$1N100\,
331               \1B\ => \$1N101\,
332               \1C\ => \$1N102\,
333               \2A\ => \$1N124\,
334               \2B\ => \$1N125\,
335               \2C\ => \$1N126\ );
336
337     \LINHA_PRI-X489129_3193575\ : entity WORK.LINHA_PRI-
338 X489129_3193575 (ARCH_LINHA_PRI-X489129_3193575)
339     port map ( \1A\ => \$1N124\,
340               \1B\ => \$1N125\,
341               \1C\ => \$1N126\,
342               \2A\ => \$1N99\,
343               \2B\ => \$1N98\,
344               \2C\ => \$1N97\ );
345
346     \LINHA_PRI-X2045998_2045999\ : entity WORK.LINHA_PRI-
347 X2045998_2045999 (ARCH_LINHA_PRI-X2045998_2045999)
348     port map ( \1A\ => \$1N129\,
349               \1B\ => \$1N128\,
350               \1C\ => \$1N127\,
351               \2A\ => \$1N121\,
352               \2B\ => \$1N122\,
353               \2C\ => \$1N123\ );
354
355     \LINHA_PRI-X1527172_1525345\ : entity WORK.LINHA_PRI-
356 X1527172_1525345 (ARCH_LINHA_PRI-X1527172_1525345)
357     port map ( \1A\ => \$1N105\,
358               \1B\ => \$1N104\,
359               \1C\ => \$1N103\,
360               \2A\ => \$1N107\,
361               \2B\ => \$1N108\,
362               \2C\ => \$1N106\ );
363
364

```

```

365 \LINHA_PRI-X1001458_2045998\ : entity WORK.LINHA_PRI-
366 X1001458_2045998 (ARCH_LINHA_PRI-X1001458_2045998)
367     port map ( \1A\ => \$1N130\,
368                \1B\ => \$1N131\,
369                \1C\ => \$1N132\,
370                \2A\ => \$1N129\,
371                \2B\ => \$1N128\,
372                \2C\ => \$1N127\ );
373
374 \LINHA_PRI-X1624998_28760967\ : entity WORK.LINHA_PRI-
375 X1624998_28760967 (ARCH_LINHA_PRI-X1624998_28760967)
376     port map ( \1A\ => \$1N70\,
377                \1B\ => \$1N81\,
378                \1C\ => \$1N180\,
379                \2A\ => \$1N96\,
380                \2B\ => \$1N95\,
381                \2C\ => \$1N94\ );
382
383 \LINHA_PRI-X1001458_1477529\ : entity WORK.LINHA_PRI-
384 X1001458_1477529 (ARCH_LINHA_PRI-X1001458_1477529)
385     port map ( \1A\ => \$1N130\,
386                \1B\ => \$1N131\,
387                \1C\ => \$1N132\,
388                \2A\ => \$1N36\,
389                \2B\ => \$1N37\,
390                \2C\ => \$1N62\ );
391
392 \LINHA_PRI-X1477529_46650023\ : entity WORK.LINHA_PRI-
393 X1477529_46650023 (ARCH_LINHA_PRI-X1477529_46650023)
394     port map ( \1A\ => \$1N36\,
395                \1B\ => \$1N37\,
396                \1C\ => \$1N62\,
397                \2A\ => \$1N133\,
398                \2B\ => \$1N134\,
399                \2C\ => \$1N135\ );
400
401 \LINHA_PRI-X46650023_296189\ : entity WORK.LINHA_PRI-
402 X46650023_296189 (ARCH_LINHA_PRI-X46650023_296189)
403     port map ( \1A\ => \$1N133\,
404                \1B\ => \$1N134\,
405                \1C\ => \$1N135\,
406                \2A\ => \$1N21\,
407                \2B\ => \$1N20\,
408                \2C\ => \$1N50\ );
409
410 \LINHA_PRI-X1525339_117180\ : entity WORK.LINHA_PRI-
411 X1525339_117180 (ARCH_LINHA_PRI-X1525339_117180)
412     port map ( \1A\ => \$1N88\,
413                \1B\ => \$1N89\,
414                \1C\ => \$1N90\,
415                \2A\ => \$1N136\,
416                \2B\ => \$1N60\,
417                \2C\ => \$1N53\ );

```

```

418
419     \LINHA_PRI-X1747204_28760949\ : entity WORK.LINHA_PRI-
420 X1747204_28760949 (ARCH_LINHA_PRI-X1747204_28760949)
421     port map ( \1A\ => \$1N92\,
422                \1B\ => \$1N93\,
423                \1C\ => \$1N91\,
424                \2A\ => \$1N18\,
425                \2B\ => \$1N17\,
426                \2C\ => \$1N16\ );
427
428     \LINHA_PRI-X1994739_486412\ : entity WORK.LINHA_PRI-
429 X1994739_486412 (ARCH_LINHA_PRI-X1994739_486412)
430     port map ( \1A\ => \$1N9\,
431                \1B\ => \$1N8\,
432                \1C\ => \$1N7\,
433                \2A\ => \$1N33\,
434                \2B\ => \$1N68\,
435                \2C\ => \$1N86\ );
436
437     \LINHA_PRI-X486412_486413\ : entity WORK.LINHA_PRI-
438 X486412_486413 (ARCH_LINHA_PRI-X486412_486413)
439     port map ( \1A\ => \$1N33\,
440                \1B\ => \$1N68\,
441                \1C\ => \$1N86\,
442                \2A\ => \$1N87\,
443                \2B\ => \$1N137\,
444                \2C\ => \$1N59\ );
445
446     \LINHA_PRI-X486413_28675721\ : entity WORK.LINHA_PRI-
447 X486413_28675721 (ARCH_LINHA_PRI-X486413_28675721)
448     port map ( \1A\ => \$1N87\,
449                \1B\ => \$1N137\,
450                \1C\ => \$1N59\,
451                \2A\ => \$1N84\,
452                \2B\ => \$1N57\,
453                \2C\ => \$1N58\ );
454
455     \LINHA_PRI-X117180_960103\ : entity WORK.LINHA_PRI-
456 X117180_960103 (ARCH_LINHA_PRI-X117180_960103)
457     port map ( \1A\ => \$1N136\,
458                \1B\ => \$1N60\,
459                \1C\ => \$1N53\,
460                \2A\ => \$1N41\,
461                \2B\ => \$1N42\,
462                \2C\ => \$1N78\ );
463
464     \LINHA_PRI-X3166792_296189\ : entity WORK.LINHA_PRI-
465 X3166792_296189 (ARCH_LINHA_PRI-X3166792_296189)
466     port map ( \1A\ => \$1N29\,
467                \1B\ => \$1N52\,
468                \1C\ => \$1N51\,
469                \2A\ => \$1N21\,
470                \2B\ => \$1N20\,

```

```

471         \2C\ => \$1N50\ );
472
473     \LINHA_PRI-X28675721_2019128\ : entity WORK.LINHA_PRI-
474     X28675721_2019128 (ARCH_LINHA_PRI-X28675721_2019128)
475         port map ( \1A\ => \$1N84\,
476                   \1B\ => \$1N57\,
477                   \1C\ => \$1N58\,
478                   \2A\ => \$1N85\,
479                   \2B\ => \$1N144\,
480                   \2C\ => \$1N145\ );
481
482     \LINHA_PRI-X28760882_5173569\ : entity WORK.LINHA_PRI-
483     X28760882_5173569 (ARCH_LINHA_PRI-X28760882_5173569)
484         port map ( \1A\ => \$1N138\,
485                   \1B\ => \$1N139\,
486                   \1C\ => \$1N140\,
487                   \2A\ => \$1N143\,
488                   \2B\ => \$1N142\,
489                   \2C\ => \$1N141\ );
490
491     \LINHA_PRI-X2220655_28760882\ : entity WORK.LINHA_PRI-
492     X2220655_28760882 (ARCH_LINHA_PRI-X2220655_28760882)
493         port map ( \1A\ => \$1N55\,
494                   \1B\ => \$1N54\,
495                   \1C\ => \$1N56\,
496                   \2A\ => \$1N138\,
497                   \2B\ => \$1N139\,
498                   \2C\ => \$1N140\ );
499
500     \LINHA_PRI-X38673_28676197\ : entity WORK.LINHA_PRI-
501     X38673_28676197 (ARCH_LINHA_PRI-X38673_28676197)
502         port map ( \1A\ => \$1N46\,
503                   \1B\ => \$1N44\,
504                   \1C\ => \$1N45\,
505                   \2A\ => \$1N165\,
506                   \2B\ => \$1N166\,
507                   \2C\ => \$1N167\ );
508
509     \LINHA_PRI-X2047531_28675987\ : entity WORK.LINHA_PRI-
510     X2047531_28675987 (ARCH_LINHA_PRI-X2047531_28675987)
511         port map ( \1A\ => \$1N19\,
512                   \1B\ => \$1N27\,
513                   \1C\ => \$1N72\,
514                   \2A\ => \$1N74\,
515                   \2B\ => \$1N162\,
516                   \2C\ => \$1N163\ );
517
518     \LINHA_PRI-X28675987_2047532\ : entity WORK.LINHA_PRI-
519     X28675987_2047532 (ARCH_LINHA_PRI-X28675987_2047532)
520         port map ( \1A\ => \$1N74\,
521                   \1B\ => \$1N162\,
522                   \1C\ => \$1N163\,
523                   \2A\ => \$1N159\,

```

```

524             \2B\ => \$1N160\,
525             \2C\ => \$1N161\ );
526
527     \LINHA_PRI-X2047532_1429563\ : entity WORK.LINHA_PRI-
528     X2047532_1429563 (ARCH_LINHA_PRI-X2047532_1429563)
529         port map ( \1A\ => \$1N159\,
530                 \1B\ => \$1N160\,
531                 \1C\ => \$1N161\,
532                 \2A\ => \$1N156\,
533                 \2B\ => \$1N157\,
534                 \2C\ => \$1N158\ );
535
536     \LINHA_PRI-X1429563_1605338\ : entity WORK.LINHA_PRI-
537     X1429563_1605338 (ARCH_LINHA_PRI-X1429563_1605338)
538         port map ( \1A\ => \$1N156\,
539                 \1B\ => \$1N157\,
540                 \1C\ => \$1N158\,
541                 \2A\ => \$1N153\,
542                 \2B\ => \$1N154\,
543                 \2C\ => \$1N155\ );
544
545     \LINHA_PRI-X2418037_28760998\ : entity WORK.LINHA_PRI-
546     X2418037_28760998 (ARCH_LINHA_PRI-X2418037_28760998)
547         port map ( \1A\ => \$1N147\,
548                 \1B\ => \$1N148\,
549                 \1C\ => \$1N149\,
550                 \2A\ => \$1N24\,
551                 \2B\ => \$1N49\,
552                 \2C\ => \$1N63\ );
553
554     \LINHA_PRI-X2418037_1973768\ : entity WORK.LINHA_PRI-
555     X2418037_1973768 (ARCH_LINHA_PRI-X2418037_1973768)
556         port map ( \1A\ => \$1N147\,
557                 \1B\ => \$1N148\,
558                 \1C\ => \$1N149\,
559                 \2A\ => \$1N77\,
560                 \2B\ => \$1N75\,
561                 \2C\ => \$1N76\ );
562
563     \LINHA_PRI-X1605338_907090\ : entity WORK.LINHA_PRI-
564     X1605338_907090 (ARCH_LINHA_PRI-X1605338_907090)
565         port map ( \1A\ => \$1N153\,
566                 \1B\ => \$1N154\,
567                 \1C\ => \$1N155\,
568                 \2A\ => \$1N282\,
569                 \2B\ => \$1N280\,
570                 \2C\ => \$1N281\ );
571
572     \LINHA_PRI-X539727_1605338\ : entity WORK.LINHA_PRI-
573     X539727_1605338 (ARCH_LINHA_PRI-X539727_1605338)
574         port map ( \1A\ => \$1N175\,
575                 \1B\ => \$1N176\,
576                 \1C\ => \$1N177\,

```

```

577             \2A\ => \$1N153\,
578             \2B\ => \$1N154\,
579             \2C\ => \$1N155\ );
580
581     \LINHA_PRI-X4781629_1743843\ : entity WORK.LINHA_PRI-
582     X4781629_1743843 (ARCH_LINHA_PRI-X4781629_1743843)
583     port map ( \1A\ => \$1N283\,
584               \1B\ => \$1N188\,
585               \1C\ => \$1N189\,
586               \2A\ => \$1N279\,
587               \2B\ => \$1N284\,
588               \2C\ => \$1N285\ );
589
590     \LINHA_PRI-X28676045_4781629\ : entity WORK.LINHA_PRI-
591     X28676045_4781629 (ARCH_LINHA_PRI-X28676045_4781629)
592     port map ( \1A\ => \$1N172\,
593               \1B\ => \$1N173\,
594               \1C\ => \$1N174\,
595               \2A\ => \$1N283\,
596               \2B\ => \$1N188\,
597               \2C\ => \$1N189\ );
598
599     \LINHA_PRI-X4781628_28676045\ : entity WORK.LINHA_PRI-
600     X4781628_28676045 (ARCH_LINHA_PRI-X4781628_28676045)
601     port map ( \1A\ => \$1N67\,
602               \1B\ => \$1N171\,
603               \1C\ => \$1N66\,
604               \2A\ => \$1N172\,
605               \2B\ => \$1N173\,
606               \2C\ => \$1N174\ );
607
608     \LINHA_PRI-X1973768_1973769\ : entity WORK.LINHA_PRI-
609     X1973768_1973769 (ARCH_LINHA_PRI-X1973768_1973769)
610     port map ( \1A\ => \$1N77\,
611               \1B\ => \$1N75\,
612               \1C\ => \$1N76\,
613               \2A\ => \$1N150\,
614               \2B\ => \$1N151\,
615               \2C\ => \$1N152\ );
616
617     \LINHA_PRI-XSE_1973769\ : entity WORK.LINHA_PRI-
618     XSE_1973769 (ARCH_LINHA_PRI-XSE_1973769)
619     port map ( \1A\ => VFASE_A,
620               \1B\ => VFASE_B,
621               \1C\ => VFASE_C,
622               \2A\ => \$1N150\,
623               \2B\ => \$1N151\,
624               \2C\ => \$1N152\ );
625
626     \LINHA_PRI-X1329777_1228633\ : entity WORK.LINHA_PRI-
627     X1329777_1228633 (ARCH_LINHA_PRI-X1329777_1228633)
628     port map ( \1A\ => \$1N79\,
629               \1B\ => \$1N80\,

```

```

630          \1C\ => \$1N71\,
631          \2A\ => \$1N47\,
632          \2B\ => \$1N65\,
633          \2C\ => \$1N64\ );
634
635      \LINHA_PRI-X28676063_4782056\ : entity WORK.LINHA_PRI-
636      X28676063_4782056 (ARCH_LINHA_PRI-X28676063_4782056)
637          port map ( \1A\ => \$1N43\,
638                    \1B\ => \$1N48\,
639                    \1C\ => \$1N23\,
640                    \2A\ => \$1N178\,
641                    \2B\ => \$1N179\,
642                    \2C\ => \$1N25\ );
643
644      \LINHA_PRI-X38673_1624998\ : entity WORK.LINHA_PRI-
645      X38673_1624998 (ARCH_LINHA_PRI-X38673_1624998)
646          port map ( \1A\ => \$1N46\,
647                    \1B\ => \$1N44\,
648                    \1C\ => \$1N45\,
649                    \2A\ => \$1N70\,
650                    \2B\ => \$1N81\,
651                    \2C\ => \$1N180\ );
652
653      \LINHA_PRI-X28760978_38673\ : entity WORK.LINHA_PRI-
654      X28760978_38673 (ARCH_LINHA_PRI-X28760978_38673)
655          port map ( \1A\ => \$1N69\,
656                    \1B\ => \$1N168\,
657                    \1C\ => \$1N30\,
658                    \2A\ => \$1N46\,
659                    \2B\ => \$1N44\,
660                    \2C\ => \$1N45\ );
661
662      \LINHA_PRI-X1003989_28760978\ : entity WORK.LINHA_PRI-
663      X1003989_28760978 (ARCH_LINHA_PRI-X1003989_28760978)
664          port map ( \1A\ => \$1N38\,
665                    \1B\ => \$1N40\,
666                    \1C\ => \$1N169\,
667                    \2A\ => \$1N69\,
668                    \2B\ => \$1N168\,
669                    \2C\ => \$1N30\ );
670
671      \LINHA_PRI-X486412_1003989\ : entity WORK.LINHA_PRI-
672      X486412_1003989 (ARCH_LINHA_PRI-X486412_1003989)
673          port map ( \1A\ => \$1N33\,
674                    \1B\ => \$1N68\,
675                    \1C\ => \$1N86\,
676                    \2A\ => \$1N38\,
677                    \2B\ => \$1N40\,
678                    \2C\ => \$1N169\ );
679
680      \LINHA_PRI-X1477528_1477529\ : entity WORK.LINHA_PRI-
681      X1477528_1477529 (ARCH_LINHA_PRI-X1477528_1477529)
682          port map ( \1A\ => \$1N61\,

```

```

683          \1B\ => \$1N35\,
684          \1C\ => \$1N34\,
685          \2A\ => \$1N36\,
686          \2B\ => \$1N37\,
687          \2C\ => \$1N62\ );
688
689      \LINHA_PRI-X1477528_28676063\ : entity WORK.LINHA_PRI-
690      X1477528_28676063 (ARCH_LINHA_PRI-X1477528_28676063)
691          port map ( \1A\ => \$1N61\,
692                    \1B\ => \$1N35\,
693                    \1C\ => \$1N34\,
694                    \2A\ => \$1N43\,
695                    \2B\ => \$1N48\,
696                    \2C\ => \$1N23\ );
697
698      \LINHA_PRI-X2019891_1477528\ : entity WORK.LINHA_PRI-
699      X2019891_1477528 (ARCH_LINHA_PRI-X2019891_1477528)
700          port map ( \1A\ => \$1N31\,
701                    \1B\ => \$1N32\,
702                    \1C\ => \$1N170\,
703                    \2A\ => \$1N61\,
704                    \2B\ => \$1N35\,
705                    \2C\ => \$1N34\ );
706
707      \LINHA_PRI-X1790994_2019891\ : entity WORK.LINHA_PRI-
708      X1790994_2019891 (ARCH_LINHA_PRI-X1790994_2019891)
709          port map ( \1A\ => \$1N39\,
710                    \1B\ => \$1N83\,
711                    \1C\ => \$1N82\,
712                    \2A\ => \$1N31\,
713                    \2B\ => \$1N32\,
714                    \2C\ => \$1N170\ );
715
716      \LINHA_PRI-X154347841_1790994\ : entity WORK.LINHA_PRI-
717      X154347841_1790994 (ARCH_LINHA_PRI-X154347841_1790994)
718          port map ( \1A\ => \$1N28\,
719                    \1B\ => \$1N277\,
720                    \1C\ => \$1N146\,
721                    \2A\ => \$1N39\,
722                    \2B\ => \$1N83\,
723                    \2C\ => \$1N82\ );
724
725      \LINHA_PRI-X960103_2220655\ : entity WORK.LINHA_PRI-
726      X960103_2220655 (ARCH_LINHA_PRI-X960103_2220655)
727          port map ( \1A\ => \$1N41\,
728                    \1B\ => \$1N42\,
729                    \1C\ => \$1N78\,
730                    \2A\ => \$1N55\,
731                    \2B\ => \$1N54\,
732                    \2C\ => \$1N56\ );
733
734      \$1I251\ : entity WORK.FONTETENS (ARCH_FONTETENS)
735          port map ( VA => VFASE_A,

```

```

731             VB => VFASE_B,
732             VC => VFASE_C );
733
734     \CLTPEQE-XCLTPEQE512943\ : entity WORK.CLTPEQE-
735     XCLTPEQE512943 (ARCH_CLTPEQE-XCLTPEQE512943)
736         port map ( \1\ => \$1N187\,
737                 \2\ => \$1N186\,
738                 \3\ => \$1N185\ );
739
740     \CLTPEQE-XCLTPEQE1525339\ : entity WORK.CLTPEQE-
741     XCLTPEQE1525339 (ARCH_CLTPEQE-XCLTPEQE1525339)
742         port map ( \1\ => \$1N88\,
743                 \2\ => \$1N89\,
744                 \3\ => \$1N90\ );
745
746     \CLTPEQE-XCLTPEQE2919791\ : entity WORK.CLTPEQE-
747     XCLTPEQE2919791 (ARCH_CLTPEQE-XCLTPEQE2919791)
748         port map ( \1\ => \$1N109\,
749                 \2\ => \$1N111\,
750                 \3\ => \$1N110\ );
751
752     \CLTPEQE-XCLTPEQE1624998\ : entity WORK.CLTPEQE-
753     XCLTPEQE1624998 (ARCH_CLTPEQE-XCLTPEQE1624998)
754         port map ( \1\ => \$1N70\,
755                 \2\ => \$1N81\,
756                 \3\ => \$1N180\ );
757
758     \CLTPEQE-XCLTPEQE1525345\ : entity WORK.CLTPEQE-
759     XCLTPEQE1525345 (ARCH_CLTPEQE-XCLTPEQE1525345)
760         port map ( \1\ => \$1N107\,
761                 \2\ => \$1N108\,
762                 \3\ => \$1N106\ );
763
764     \CLTPEQE-XCLTPEQE2019128\ : entity WORK.CLTPEQE-
765     XCLTPEQE2019128 (ARCH_CLTPEQE-XCLTPEQE2019128)
766         port map ( \1\ => \$1N85\,
767                 \2\ => \$1N144\,
768                 \3\ => \$1N145\ );
769
770     \CLTPEQE-XCLTPEQE588470\ : entity WORK.CLTPEQE-
771     XCLTPEQE588470 (ARCH_CLTPEQE-XCLTPEQE588470)
772         port map ( \1\ => \$1N26\,
773                 \2\ => \$1N73\,
774                 \3\ => \$1N164\ );
775
776     \CLTPEQE-XCLTPEQE4781628\ : entity WORK.CLTPEQE-
777     XCLTPEQE4781628 (ARCH_CLTPEQE-XCLTPEQE4781628)
778         port map ( \1\ => \$1N67\,
779                 \2\ => \$1N171\,
780                 \3\ => \$1N66\ );
781
782     \CLTPEQE-XCLTPEQE539727\ : entity WORK.CLTPEQE-
783     XCLTPEQE539727 (ARCH_CLTPEQE-XCLTPEQE539727)

```

```

784     port map ( \1\ => \$1N175\,
785               \2\ => \$1N176\,
786               \3\ => \$1N177\ );
787
788     \LINHA_PRI-X28760998_1228633\ : entity WORK.LINHA_PRI-
789     X28760998_1228633 (ARCH_LINHA_PRI-X28760998_1228633)
790     port map ( \1A\ => \$1N24\,
791               \1B\ => \$1N49\,
792               \1C\ => \$1N63\,
793               \2A\ => \$1N47\,
794               \2B\ => \$1N65\,
795               \2C\ => \$1N64\ );
796
797     \CNTDQT-XCNTDQT4782056\ : entity WORK.CNTDQT-
798     XCNTDQT4782056 (ARCH_CNTDQT-XCNTDQT4782056)
799     port map ( \1\ => \$1N178\,
800               \2\ => \$1N179\,
801               \3\ => \$1N25\ );
802
803     \LINHA_PRI-X5173569_3166792\ : entity WORK.LINHA_PRI-
804     X5173569_3166792 (ARCH_LINHA_PRI-X5173569_3166792)
805     port map ( \1A\ => \$1N143\,
806               \1B\ => \$1N142\,
807               \1C\ => \$1N141\,
808               \2A\ => \$1N29\,
809               \2B\ => \$1N52\,
810               \2C\ => \$1N51\ );
811
812     \CLTPEQE-XCLTPEQE2047531\ : entity WORK.CLTPEQE-
813     XCLTPEQE2047531 (ARCH_CLTPEQE-XCLTPEQE2047531)
814     port map ( \1\ => \$1N19\,
815               \2\ => \$1N27\,
816               \3\ => \$1N72\ );
817
818     \LINHA_PRI-X28676197_588470\ : entity WORK.LINHA_PRI-
819     X28676197_588470 (ARCH_LINHA_PRI-X28676197_588470)
820     port map ( \1A\ => \$1N165\,
821               \1B\ => \$1N166\,
822               \1C\ => \$1N167\,
823               \2A\ => \$1N26\,
824               \2B\ => \$1N73\,
825               \2C\ => \$1N164\ );
826
827     \LINHA_PRI-X28760949_960319\ : entity WORK.LINHA_PRI-
828     X28760949_960319 (ARCH_LINHA_PRI-X28760949_960319)
829     port map ( \1A\ => \$1N18\,
830               \1B\ => \$1N17\,
831               \1C\ => \$1N16\,
832               \2A\ => \$1N15\,
833               \2B\ => \$1N14\,
834               \2C\ => \$1N13\ );

```

```

833 \LINHA_PRI-X1178294_388608\ : entity WORK.LINHA_PRI-
834 X1178294_388608 (ARCH_LINHA_PRI-X1178294_388608)
835     port map ( \1A\ => \$1N4\,
836                \1B\ => \$1N5\,
837                \1C\ => \$1N6\,
838                \2A\ => \$1N112\,
839                \2B\ => \$1N113\,
840                \2C\ => \$1N114\ );
841
842 \LINHA_PRI-X9603201_960320\ : entity WORK.LINHA_PRI-
843 X9603201_960320 (ARCH_LINHA_PRI-X9603201_960320)
844     port map ( \1A\ => \$1N12\,
845                \1B\ => \$1N11\,
846                \1C\ => \$1N10\,
847                \2A\ => \$1N192\,
848                \2B\ => \$1N191\,
849                \2C\ => \$1N190\ );
850
851 \CLTPEQE-XCLTPEQE1994739\ : entity WORK.CLTPEQE-
852 XCLTPEQE1994739 (ARCH_CLTPEQE-XCLTPEQE1994739)
853     port map ( \1\ => \$1N9\,
854                \2\ => \$1N8\,
855                \3\ => \$1N7\ );
856
857 \LINHA_PRI-X1178293_1178294\ : entity WORK.LINHA_PRI-
858 X1178293_1178294 (ARCH_LINHA_PRI-X1178293_1178294)
859     port map ( \1A\ => \$1N1\,
860                \1B\ => \$1N2\,
861                \1C\ => \$1N3\,
862                \2A\ => \$1N4\,
863                \2B\ => \$1N5\,
864                \2C\ => \$1N6\ );
865
866 \REGDF_3F-X960319_9603201\ : entity WORK.REGDF_3F-
867 X960319_9603201 (ARCH_REGDF_3F-X960319_9603201)
868     port map ( \1A\ => \$1N15\,
869                \1B\ => \$1N14\,
870                \1C\ => \$1N13\,
871                \2A\ => \$1N12\,
872                \2B\ => \$1N11\,
873                \2C\ => \$1N10\ );
874
875 \LINHA_PRI-X1743843_2894747\ : entity WORK.LINHA_PRI-
876 X1743843_2894747 (ARCH_LINHA_PRI-X1743843_2894747)
877     port map ( \1A\ => \$1N279\,
878                \1B\ => \$1N284\,
879                \1C\ => \$1N285\,
880                \2A\ => \$1N275\,
881                \2B\ => \$1N276\,
882                \2C\ => \$1N193\ );
883
884 \LINHA_PRI-X2894747_154347841\ : entity WORK.LINHA_PRI-
885 X2894747_154347841 (ARCH_LINHA_PRI-X2894747_154347841)

```

```
886      port map ( \1A\ => \$1N275\,  
887                \1B\ => \$1N276\,  
888                \1C\ => \$1N193\,  
889                \2A\ => \$1N28\,  
890                \2B\ => \$1N277\,  
891                \2C\ => \$1N146\ );  
892  
893      \LINHA_PRI-X907090_1743843\ : entity WORK.LINHA_PRI-  
894      X907090_1743843 (ARCH_LINHA_PRI-X907090_1743843)  
895      port map ( \1A\ => \$1N282\,  
896                \1B\ => \$1N280\,  
897                \1C\ => \$1N281\,  
898                \2A\ => \$1N279\,  
899                \2B\ => \$1N284\,  
900                \2C\ => \$1N285\ );  
901  
902  end architecture arch_FEDEER;
```

D.5- CÓDIGO VHDL: BLOCO “Regulacao_Transf_v1_0.vhd”

Este código fonte é responsável pela conexão entre o bloco “Regulacao.vhd” e o bloco “Regulacao_Transf_v1_0_S00_AXI_TRANSF_parameterized0.vhd” dentro do FPGA. Neste código é feita a conexão dos barramentos de dados criados dos dois anteriores arquivos VHDL e também este código representa o bloco “Regulacao_Transf_v1.0” no projeto geral do **Vivado Design Suite**. Este código fonte foi gerado por um assistente de configuração e criação da interface AXI-Lite e, posteriormente modificado para as necessidades do projeto. Este código é sintetizado pelo **Vivado Design Suite** e implementado no FPGA Artix-7.

```

001 library ieee;
002 use ieee.std_logic_1164.all;
003 use ieee.std_logic_arith.all;
004 use ieee.std_logic_unsigned.all;
005
006 entity Regulacao_Transf_v1_0 is
007     generic (
008         -- Users to add parameters here
009         -- User parameters ends
010         -- Do not modify the parameters beyond this line
011         -- Parameters of Axi Slave Bus Interface S00_AXI_TRANSF
012         C_S00_AXI_TRANSF_DATA_WIDTH : integer := 32;
013         C_S00_AXI_TRANSF_ADDR_WIDTH : integer := 7
014     );
015     port (
016         -- Users to add ports here
017         -- User ports ends
018         -- Do not modify the ports beyond this line
019         -- Ports of Axi Slave Bus Interface S00_AXI_TRANSF
020         s00_axi_transf_aclk : in std_logic;
021         s00_axi_transf_aresetn : in std_logic;
022         s00_axi_transf_awaddr : in
023         std_logic_vector(C_S00_AXI_TRANSF_ADDR_WIDTH-1 downto 0);
024         s00_axi_transf_awprot : in std_logic_vector(2 downto 0);
025         s00_axi_transf_awvalid : in std_logic;
026         s00_axi_transf_awready : out std_logic;
027         s00_axi_transf_wdata : in
028         std_logic_vector(C_S00_AXI_TRANSF_DATA_WIDTH-1 downto 0);
029         s00_axi_transf_wstrb : in
030         std_logic_vector((C_S00_AXI_TRANSF_DATA_WIDTH/8)-1 downto 0);
031         s00_axi_transf_wvalid : in std_logic;
032         s00_axi_transf_wready : out std_logic;
033         s00_axi_transf_bresp : out std_logic_vector(1 downto 0);
034         s00_axi_transf_bvalid : out std_logic;
035         s00_axi_transf_bready : in std_logic;
036         s00_axi_transf_araddr : in
037         std_logic_vector(C_S00_AXI_TRANSF_ADDR_WIDTH-1 downto 0);
038         s00_axi_transf_arprot : in std_logic_vector(2 downto 0);
039         s00_axi_transf_arvalid : in std_logic;
040         s00_axi_transf_arready : out std_logic;
041         s00_axi_transf_rdata : out
042         std_logic_vector(C_S00_AXI_TRANSF_DATA_WIDTH-1 downto 0);
043         s00_axi_transf_rresp : out std_logic_vector(1 downto 0);
044         s00_axi_transf_rvalid : out std_logic;
045         s00_axi_transf_rready : in std_logic
046     );
047 end Regulacao_Transf_v1_0;

```

```

048 architecture arch_imp of Regulacao_Transf_v1_0 is
049     -- component declaration
050     component Regulacao_Transf_v1_0_S00_AXI_TRANSF is
051         generic (
052             C_S_AXI_DATA_WIDTH    : integer    := 32;
053             C_S_AXI_ADDR_WIDTH    : integer    := 7
054         );
055         port (
056             tapA_reg0 : out std_logic_vector(4 downto 0);
057             tapB_reg1 : out std_logic_vector(4 downto 0);
058             tapC_reg2 : out std_logic_vector(4 downto 0);
059             r1a_reg3  : out std_logic_vector(10 downto 0);
060             r2a_reg4  : out std_logic_vector(10 downto 0);
061             r3a_reg5  : out std_logic_vector(10 downto 0);
062             r4a_reg6  : out std_logic_vector(10 downto 0);
063             r5a_reg7  : out std_logic_vector(10 downto 0);
064             r1b_reg8  : out std_logic_vector(10 downto 0);
065             r2b_reg9  : out std_logic_vector(10 downto 0);
066             r3b_reg10 : out std_logic_vector(10 downto 0);
067             r4b_reg11 : out std_logic_vector(10 downto 0);
068             r5b_reg12 : out std_logic_vector(10 downto 0);
069             r1c_reg13 : out std_logic_vector(10 downto 0);
070             r2c_reg14 : out std_logic_vector(10 downto 0);
071             r3c_reg15 : out std_logic_vector(10 downto 0);
072             r4c_reg16 : out std_logic_vector(10 downto 0);
073             r5c_reg17 : out std_logic_vector(10 downto 0);
074             resp_reg18 : in std_logic_vector(6 downto 0);
075
076             S_AXI_ACLK    : in std_logic;
077             S_AXI_ARESETN : in std_logic;
078             S_AXI_AWADDR  : in std_logic_vector(C_S_AXI_ADDR_WIDTH-1
079 downto 0);
080             S_AXI_AWPROT  : in std_logic_vector(2 downto 0);
081             S_AXI_AWVALID : in std_logic;
082             S_AXI_AWREADY : out std_logic;
083             S_AXI_WDATA   : in std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
084 0);
085             S_AXI_WSTRB   : in std_logic_vector((C_S_AXI_DATA_WIDTH/8)-1
086 downto 0);
087             S_AXI_WVALID  : in std_logic;
088             S_AXI_WREADY  : out std_logic;
089             S_AXI_BRESP   : out std_logic_vector(1 downto 0);
090             S_AXI_BVALID  : out std_logic;
091             S_AXI_BREADY  : in std_logic;
092             S_AXI_ARADDR  : in std_logic_vector(C_S_AXI_ADDR_WIDTH-1
093 downto 0);
094             S_AXI_ARPROT  : in std_logic_vector(2 downto 0);
095             S_AXI_ARVALID : in std_logic;
096             S_AXI_ARREADY : out std_logic;
097             S_AXI_RDATA   : out std_logic_vector(C_S_AXI_DATA_WIDTH-1
098 downto 0);
099             S_AXI_RRESP   : out std_logic_vector(1 downto 0);
100             S_AXI_RVALID  : out std_logic;
101             S_AXI_RREADY  : in std_logic
102         );
103     end component Regulacao_Transf_v1_0_S00_AXI_TRANSF;
104
105     component Regulacao is
106     port(
107         clk : in STD_LOGIC;                -- Clock do Sistema
108         tap_A : in STD_LOGIC_VECTOR(4 downto 0); -- -8 to 8

```

```

109      tap_B : in STD_LOGIC_VECTOR(4 downto 0); -- -8 to 8
110      tap_C : in STD_LOGIC_VECTOR(4 downto 0); -- -8 to 8
111      R5a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
112      R4a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
113      R3a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
114      R2a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
115      R1a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
116      R5b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
117      R4b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
118      R3b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
119      R2b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
120      R1b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
121      R5c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
122      R4c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
123      R3c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
124      R2c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
125      R1c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
126      rec_Ra_Rb_Rc_AtN : out STD_LOGIC_VECTOR(6 downto 0)
127      );
128  end component;
129
130  signal sig_tap_A : STD_LOGIC_VECTOR(4 downto 0);
131  signal sig_tap_B : STD_LOGIC_VECTOR(4 downto 0);
132  signal sig_tap_C : STD_LOGIC_VECTOR(4 downto 0);
133  signal sig_R5a : STD_LOGIC_VECTOR(10 downto 0);
134  signal sig_R4a : STD_LOGIC_VECTOR(10 downto 0);
135  signal sig_R3a : STD_LOGIC_VECTOR(10 downto 0);
136  signal sig_R2a : STD_LOGIC_VECTOR(10 downto 0);
137  signal sig_R1a : STD_LOGIC_VECTOR(10 downto 0);
138  signal sig_R5b : STD_LOGIC_VECTOR(10 downto 0);
139  signal sig_R4b : STD_LOGIC_VECTOR(10 downto 0);
140  signal sig_R3b : STD_LOGIC_VECTOR(10 downto 0);
141  signal sig_R2b : STD_LOGIC_VECTOR(10 downto 0);
142  signal sig_R1b : STD_LOGIC_VECTOR(10 downto 0);
143  signal sig_R5c : STD_LOGIC_VECTOR(10 downto 0);
144  signal sig_R4c : STD_LOGIC_VECTOR(10 downto 0);
145  signal sig_R3c : STD_LOGIC_VECTOR(10 downto 0);
146  signal sig_R2c : STD_LOGIC_VECTOR(10 downto 0);
147  signal sig_R1c : STD_LOGIC_VECTOR(10 downto 0);
148  signal sig_rec_Ra_Rb_Rc_AtN : STD_LOGIC_VECTOR(6 downto 0);
149
150  begin
151
152  -- Instantiation of Axi Bus Interface S00_AXI_TRANSF
153  Regulacao_Transf_v1_0_S00_AXI_TRANSF_inst :
154  Regulacao_Transf_v1_0_S00_AXI_TRANSF
155      generic map (
156          C_S_AXI_DATA_WIDTH    => C_S00_AXI_TRANSF_DATA_WIDTH,
157          C_S_AXI_ADDR_WIDTH    => C_S00_AXI_TRANSF_ADDR_WIDTH
158      )
159      port map (
160          tapA_reg0 => sig_tap_A,
161          tapB_reg1 => sig_tap_B,
162          tapC_reg2 => sig_tap_C,
163          r1a_reg3  => sig_R1a,
164          r2a_reg4  => sig_R2a,
165          r3a_reg5  => sig_R3a,
166          r4a_reg6  => sig_R4a,
167          r5a_reg7  => sig_R5a,
168          r1b_reg8  => sig_R1b,
169          r2b_reg9  => sig_R2b,

```

```

170         r3b_reg10 => sig_R3b,
171         r4b_reg11 => sig_R4b,
172         r5b_reg12 => sig_R5b,
173         r1c_reg13 => sig_R1c,
174         r2c_reg14 => sig_R2c,
175         r3c_reg15 => sig_R3c,
176         r4c_reg16 => sig_R4c,
177         r5c_reg17 => sig_R5c,
178         resp_reg18 => sig_rec_Ra_Rb_Rc_AtN,
179         S_AXI_ACLK => s00_axi_transf_aclk,
180         S_AXI_ARESETN => s00_axi_transf_aresetn,
181         S_AXI_AWADDR => s00_axi_transf_awaddr,
182         S_AXI_AWPROT => s00_axi_transf_awprot,
183         S_AXI_AWVALID => s00_axi_transf_awvalid,
184         S_AXI_AWREADY => s00_axi_transf_awready,
185         S_AXI_WDATA => s00_axi_transf_wdata,
186         S_AXI_WSTRB => s00_axi_transf_wstrb,
187         S_AXI_WVALID => s00_axi_transf_wvalid,
188         S_AXI_WREADY => s00_axi_transf_wready,
189         S_AXI_BRESP => s00_axi_transf_bresp,
190         S_AXI_BVALID => s00_axi_transf_bvalid,
191         S_AXI_BREADY => s00_axi_transf_bready,
192         S_AXI_ARADDR => s00_axi_transf_araddr,
193         S_AXI_ARPROT => s00_axi_transf_arprot,
194         S_AXI_ARVALID => s00_axi_transf_arvalid,
195         S_AXI_ARREADY => s00_axi_transf_arready,
196         S_AXI_RDATA => s00_axi_transf_rdata,
197         S_AXI_RRESP => s00_axi_transf_rresp,
198         S_AXI_RVALID => s00_axi_transf_rvalid,
199         S_AXI_RREADY => s00_axi_transf_rready
200     );
201     -- Add user logic here
202     Regulacao_inst : Regulacao
203     port map(
204         clk => s00_axi_transf_aclk,
205         Tap_A => sig_tap_A,
206         Tap_B => sig_tap_B,
207         Tap_C => sig_tap_C,
208         R1a => sig_R1a,
209         R2a => sig_R2a,
210         R3a => sig_R3a,
211         R4a => sig_R4a,
212         R5a => sig_R5a,
213         R1b => sig_R1b,
214         R2b => sig_R2b,
215         R3b => sig_R3b,
216         R4b => sig_R4b,
217         R5b => sig_R5b,
218         R1c => sig_R1c,
219         R2c => sig_R2c,
220         R3c => sig_R3c,
221         R4c => sig_R4c,
222         R5c => sig_R5c,
223         rec_Ra_Rb_Rc_AtN => sig_rec_Ra_Rb_Rc_AtN
224     );
225     -- User logic ends
226 end arch_imp;

```

D.6- Código VHDL: BLOCO

“Regulacao_Transf_v1_0_S00_AXI_TRASNF_parameterized0.vhd”

Este código fonte disponibiliza o acesso aos dados dos registradores em forma de barramentos paralelos de dados. A função principal é trabalhar com a comunicação AXI-Lite pela parte do FPGA Artix-7, para a leitura e escrita de dados nos registradores da interface. Este arquivo foi gerado por um assistente de configuração e criação da interface AXI-Lite e posteriormente modificado para as necessidades do projeto. Este código é sintetizado pelo **Vivado Design Suite** e implementado no FPGA Artix-7.

```

001 library ieee;
002 use ieee.std_logic_1164.all;
003 use ieee.std_logic_arith.all;
004 use ieee.std_logic_unsigned.all;
005
006 entity Regulacao_Transf_v1_0_S00_AXI_TRANSF is
007     generic (
008         -- Users to add parameters here
009
010         -- User parameters ends
011         -- Do not modify the parameters beyond this line
012
013         -- Width of S_AXI data bus
014         C_S_AXI_DATA_WIDTH  : integer    := 32;
015         -- Width of S_AXI address bus
016         C_S_AXI_ADDR_WIDTH  : integer    := 7
017     );
018     port (
019         -- Users to add ports here
020         tapA_reg0 : out std_logic_vector(4 downto 0);
021         tapB_reg1 : out std_logic_vector(4 downto 0);
022         tapC_reg2 : out std_logic_vector(4 downto 0);
023
024         r1a_reg3 : out std_logic_vector(10 downto 0);
025         r2a_reg4 : out std_logic_vector(10 downto 0);
026         r3a_reg5 : out std_logic_vector(10 downto 0);
027         r4a_reg6 : out std_logic_vector(10 downto 0);
028         r5a_reg7 : out std_logic_vector(10 downto 0);
029         r1b_reg8 : out std_logic_vector(10 downto 0);
030         r2b_reg9 : out std_logic_vector(10 downto 0);
031         r3b_reg10 : out std_logic_vector(10 downto 0);
032         r4b_reg11 : out std_logic_vector(10 downto 0);
033         r5b_reg12 : out std_logic_vector(10 downto 0);
034         r1c_reg13 : out std_logic_vector(10 downto 0);
035         r2c_reg14 : out std_logic_vector(10 downto 0);
036         r3c_reg15 : out std_logic_vector(10 downto 0);
037         r4c_reg16 : out std_logic_vector(10 downto 0);
038         r5c_reg17 : out std_logic_vector(10 downto 0);
039
040         resp_reg18 : in std_logic_vector(6 downto 0);
041         -- User ports ends
042         -- Do not modify the ports beyond this line
043
044         -- Global Clock Signal
045         S_AXI_ACLK : in std_logic;
046         -- Global Reset Signal. This Signal is Active LOW
047         S_AXI_ARESETN : in std_logic;

```

```

048      -- Write address (issued by master, accepted by Slave)
049      S_AXI_AWADDR      : in std_logic_vector(C_S_AXI_ADDR_WIDTH-1
050  downto 0);
051      -- Write channel Protection type. This signal indicates the
052      -- privilege and security level of the transaction, and
053  whether
054      -- the transaction is a data access or an instruction
055  access.
056      S_AXI_AWPROT      : in std_logic_vector(2 downto 0);
057      -- Write address valid. This signal indicates that the master
058  signaling
059      -- valid write address and control information.
060      S_AXI_AWVALID     : in std_logic;
061      -- Write address ready. This signal indicates that the slave
062  is ready
063      -- to accept an address and associated control signals.
064      S_AXI_AWREADY     : out std_logic;
065      -- Write data (issued by master, accepted by Slave)
066      S_AXI_WDATA      : in std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
067  0);
068      -- Write strobes. This signal indicates which byte lanes hold
069      -- valid data. There is one write strobe bit for each
070  eight
071      -- bits of the write data bus.
072      S_AXI_WSTRB      : in std_logic_vector((C_S_AXI_DATA_WIDTH/8)-1
073  downto 0);
074      -- Write valid. This signal indicates that valid write
075      -- data and strobes are available.
076      S_AXI_WVALID     : in std_logic;
077      -- Write ready. This signal indicates that the slave
078      -- can accept the write data.
079      S_AXI_WREADY     : out std_logic;
080      -- Write response. This signal indicates the status
081      -- of the write transaction.
082      S_AXI_BRESP      : out std_logic_vector(1 downto 0);
083      -- Write response valid. This signal indicates that the
084  channel
085      -- is signaling a valid write response.
086      S_AXI_BVALID     : out std_logic;
087      -- Response ready. This signal indicates that the master
088      -- can accept a write response.
089      S_AXI_BREADY     : in std_logic;
090      -- Read address (issued by master, accepted by Slave)
091      S_AXI_ARADDR      : in std_logic_vector(C_S_AXI_ADDR_WIDTH-1
092  downto 0);
093      -- Protection type. This signal indicates the privilege
094      -- and security level of the transaction, and whether the
095      -- transaction is a data access or an instruction access.
096      S_AXI_ARPROT     : in std_logic_vector(2 downto 0);
097      -- Read address valid. This signal indicates that the channel
098      -- is signaling valid read address and control
099  information.
100      S_AXI_ARVALID     : in std_logic;
101      -- Read address ready. This signal indicates that the slave
102  is
103      -- ready to accept an address and associated control
104  signals.
105      S_AXI_ARREADY     : out std_logic;
106      -- Read data (issued by slave)
107      S_AXI_RDATA      : out std_logic_vector(C_S_AXI_DATA_WIDTH-1
108  downto 0);

```

```

109         -- Read response. This signal indicates the status of the
110         -- read transfer.
111         S_AXI_RRESP : out std_logic_vector(1 downto 0);
112         -- Read valid. This signal indicates that the channel is
113         -- signaling the required read data.
114         S_AXI_RVALID : out std_logic;
115         -- Read ready. This signal indicates that the master can
116         -- accept the read data and response information.
117         S_AXI_RREADY : in std_logic
118     );
119 end Regulacao_Transf_v1_0_S00_AXI_TRANSF;
120
121 architecture arch_imp of Regulacao_Transf_v1_0_S00_AXI_TRANSF is
122
123     -- AXI4LITE signals
124     signal axi_awaddr : std_logic_vector(C_S_AXI_ADDR_WIDTH-1
125     downto 0);
126     signal axi_awready : std_logic;
127     signal axi_wready : std_logic;
128     signal axi_bresp : std_logic_vector(1 downto 0);
129     signal axi_bvalid : std_logic;
130     signal axi_araddr : std_logic_vector(C_S_AXI_ADDR_WIDTH-1
131     downto 0);
132     signal axi_arready : std_logic;
133     signal axi_rdata : std_logic_vector(C_S_AXI_DATA_WIDTH-1
134     downto 0);
135     signal axi_rresp : std_logic_vector(1 downto 0);
136     signal axi_rvalid : std_logic;
137
138     -- Example-specific design signals
139     -- local parameter for addressing 32 bit / 64 bit
140     C_S_AXI_DATA_WIDTH
141     -- ADDR_LSB is used for addressing 32/64 bit registers/memories
142     -- ADDR_LSB = 2 for 32 bits (n downto 2)
143     -- ADDR_LSB = 3 for 64 bits (n downto 3)
144     constant ADDR_LSB : integer := (C_S_AXI_DATA_WIDTH/32)+ 1;
145     constant OPT_MEM_ADDR_BITS : integer := 4;
146     -----
147     ----- Signals for user logic register space example
148     -----
149     ----- Number of Slave Registers 19
150     signal slv_reg0 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
151     signal slv_reg1 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
152     signal slv_reg2 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
153     signal slv_reg3 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
154     signal slv_reg4 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
155     signal slv_reg5 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
156     signal slv_reg6 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
157     signal slv_reg7 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
158     signal slv_reg8 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
159     signal slv_reg9 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto 0);
160     signal slv_reg10 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
161     0);
162     signal slv_reg11 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
163     0);
164     signal slv_reg12 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
165     0);
166     signal slv_reg13 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
167     0);
168     signal slv_reg14 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
169     0);

```

```

170 signal slv_reg15 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
171 0);
172 signal slv_reg16 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
173 0);
174 signal slv_reg17 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
175 0);
176 signal slv_reg18 :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
177 0);
178 signal slv_reg_rden : std_logic;
179 signal slv_reg_wren : std_logic;
180 signal reg_data_out :std_logic_vector(C_S_AXI_DATA_WIDTH-1 downto
181 0);
182 signal byte_index : integer;
183
184 begin
185     -- I/O Connections assignments
186     tapA_reg0 <= slv_reg0(4 downto 0);
187     tapB_reg1 <= slv_reg1(4 downto 0);
188     tapC_reg2 <= slv_reg2(4 downto 0);
189     r1a_reg3 <= slv_reg3(10 downto 0);
190     r2a_reg4 <= slv_reg4(10 downto 0);
191     r3a_reg5 <= slv_reg5(10 downto 0);
192     r4a_reg6 <= slv_reg6(10 downto 0);
193     r5a_reg7 <= slv_reg7(10 downto 0);
194     r1b_reg8 <= slv_reg8(10 downto 0);
195     r2b_reg9 <= slv_reg9(10 downto 0);
196     r3b_reg10 <= slv_reg10(10 downto 0);
197     r4b_reg11 <= slv_reg11(10 downto 0);
198     r5b_reg12 <= slv_reg12(10 downto 0);
199     r1c_reg13 <= slv_reg13(10 downto 0);
200     r2c_reg14 <= slv_reg14(10 downto 0);
201     r3c_reg15 <= slv_reg15(10 downto 0);
202     r4c_reg16 <= slv_reg16(10 downto 0);
203     r5c_reg17 <= slv_reg17(10 downto 0);
204     slv_reg18(6 downto 0) <= resp_reg18;
205
206     S_AXI_AWREADY <= axi_awready;
207     S_AXI_WREADY <= axi_wready;
208     S_AXI_BRESP <= axi_bresp;
209     S_AXI_BVALID <= axi_bvalid;
210     S_AXI_ARREADY <= axi_arready;
211     S_AXI_RDATA <= axi_rdata;
212     S_AXI_RRESP <= axi_rresp;
213     S_AXI_RVALID <= axi_rvalid;
214     -- Implement axi_awready generation
215     -- axi_awready is asserted for one S_AXI_ACLK clock cycle when
216 both
217     -- S_AXI_AWVALID and S_AXI_WVALID are asserted. axi_awready is
218     -- de-asserted when reset is low.
219
220     process (S_AXI_ACLK)
221     begin
222         if rising_edge(S_AXI_ACLK) then
223             if S_AXI_ARESETN = '0' then
224                 axi_awready <= '0';
225             else
226                 if (axi_awready = '0' and S_AXI_AWVALID = '1' and
227 S_AXI_WVALID = '1') then
228                     -- slave is ready to accept write address when
229                     -- there is a valid write address and write data
230                     -- on the write address and data bus. This design

```

```

231         -- expects no outstanding transactions.
232         axi_awready <= '1';
233     else
234         axi_awready <= '0';
235     end if;
236 end if;
237 end if;
238 end process;
239
240 -- Implement axi_awaddr latching
241 -- This process is used to latch the address when both
242 -- S_AXI_AWVALID and S_AXI_WVALID are valid.
243
244 process (S_AXI_ACLK)
245 begin
246     if rising_edge(S_AXI_ACLK) then
247         if S_AXI_ARESETN = '0' then
248             axi_awaddr <= (others => '0');
249         else
250             if (axi_awready = '0' and S_AXI_AWVALID = '1' and
251 S_AXI_WVALID = '1') then
252                 -- Write Address latching
253                 axi_awaddr <= S_AXI_AWADDR;
254             end if;
255         end if;
256     end if;
257 end process;
258
259 -- Implement axi_wready generation
260 -- axi_wready is asserted for one S_AXI_ACLK clock cycle when
261 both
262 -- S_AXI_AWVALID and S_AXI_WVALID are asserted. axi_wready is
263 -- de-asserted when reset is low.
264
265 process (S_AXI_ACLK)
266 begin
267     if rising_edge(S_AXI_ACLK) then
268         if S_AXI_ARESETN = '0' then
269             axi_wready <= '0';
270         else
271             if (axi_wready = '0' and S_AXI_WVALID = '1' and
272 S_AXI_AWVALID = '1') then
273                 -- slave is ready to accept write data when
274                 -- there is a valid write address and write data
275                 -- on the write address and data bus. This design
276                 -- expects no outstanding transactions.
277                 axi_wready <= '1';
278             else
279                 axi_wready <= '0';
280             end if;
281         end if;
282     end if;
283 end process;
284
285 -- Implement memory mapped register select and write logic
286 generation
287 -- The write data is accepted and written to memory mapped
288 registers when
289 -- axi_awready, S_AXI_WVALID, axi_wready and S_AXI_WVALID are
290 asserted. Write strobes are used to
291 -- select byte enables of slave registers while writing.

```

```

292 -- These registers are cleared when reset (active low) is
293 applied.
294 -- Slave register write enable is asserted when valid address and
295 data are available
296 -- and the slave is ready to accept the write address and write
297 data.
298 slv_reg_wren <= axi_wready and S_AXI_WVALID and axi_awready and
299 S_AXI_AWVALID ;
300
301 process (S_AXI_ACLK)
302 variable loc_addr :std_logic_vector(OPT_MEM_ADDR_BITS downto 0);
303 begin
304   if rising_edge(S_AXI_ACLK) then
305     if S_AXI_ARESETN = '0' then
306       slv_reg0 <= (others => '0');
307       slv_reg1 <= (others => '0');
308       slv_reg2 <= (others => '0');
309       slv_reg3 <= (others => '0');
310       slv_reg4 <= (others => '0');
311       slv_reg5 <= (others => '0');
312       slv_reg6 <= (others => '0');
313       slv_reg7 <= (others => '0');
314       slv_reg8 <= (others => '0');
315       slv_reg9 <= (others => '0');
316       slv_reg10 <= (others => '0');
317       slv_reg11 <= (others => '0');
318       slv_reg12 <= (others => '0');
319       slv_reg13 <= (others => '0');
320       slv_reg14 <= (others => '0');
321       slv_reg15 <= (others => '0');
322       slv_reg16 <= (others => '0');
323       slv_reg17 <= (others => '0');
324       --slv_reg18 <= (others => '0');
325     else
326       loc_addr := axi_awaddr(ADDR_LSB + OPT_MEM_ADDR_BITS downto
327 ADDR_LSB);
328       if (slv_reg_wren = '1') then
329         case loc_addr is
330           when b"00000" =>
331             for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
332               if ( S_AXI_WSTRB(byte_index) = '1' ) then
333                 -- Respective byte enables are asserted as per
334 write strobes
335                 -- slave register 0
336                 slv_reg0(byte_index*8+7 downto byte_index*8) <=
337 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
338               end if;
339             end loop;
340           when b"00001" =>
341             for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
342               if ( S_AXI_WSTRB(byte_index) = '1' ) then
343                 -- Respective byte enables are asserted as per
344 write strobes
345                 -- slave register 1
346                 slv_reg1(byte_index*8+7 downto byte_index*8) <=
347 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
348               end if;
349             end loop;
350           when b"00010" =>
351             for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
352               if ( S_AXI_WSTRB(byte_index) = '1' ) then

```

```

353         -- Respective byte enables are asserted as per
354 write strobes
355         -- slave register 2
356         slv_reg2(byte_index*8+7 downto byte_index*8) <=
357 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
358     end if;
359 end loop;
360 when b"00011" =>
361     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
362         if ( S_AXI_WSTRB(byte_index) = '1' ) then
363             -- Respective byte enables are asserted as per
364 write strobes
365             -- slave register 3
366             slv_reg3(byte_index*8+7 downto byte_index*8) <=
367 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
368         end if;
369     end loop;
370 when b"00100" =>
371     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
372         if ( S_AXI_WSTRB(byte_index) = '1' ) then
373             -- Respective byte enables are asserted as per
374 write strobes
375             -- slave register 4
376             slv_reg4(byte_index*8+7 downto byte_index*8) <=
377 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
378         end if;
379     end loop;
380 when b"00101" =>
381     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
382         if ( S_AXI_WSTRB(byte_index) = '1' ) then
383             -- Respective byte enables are asserted as per
384 write strobes
385             -- slave register 5
386             slv_reg5(byte_index*8+7 downto byte_index*8) <=
387 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
388         end if;
389     end loop;
390 when b"00110" =>
391     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
392         if ( S_AXI_WSTRB(byte_index) = '1' ) then
393             -- Respective byte enables are asserted as per
394 write strobes
395             -- slave register 6
396             slv_reg6(byte_index*8+7 downto byte_index*8) <=
397 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
398         end if;
399     end loop;
400 when b"00111" =>
401     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
402         if ( S_AXI_WSTRB(byte_index) = '1' ) then
403             -- Respective byte enables are asserted as per
404 write strobes
405             -- slave register 7
406             slv_reg7(byte_index*8+7 downto byte_index*8) <=
407 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
408         end if;
409     end loop;
410 when b"01000" =>
411     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
412         if ( S_AXI_WSTRB(byte_index) = '1' ) then
413

```

```

414         -- Respective byte enables are asserted as per
415 write strobes
416         -- slave register 8
417         slv_reg8(byte_index*8+7 downto byte_index*8) <=
418 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
419     end if;
420 end loop;
421 when b"01001" =>
422     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
423         if ( S_AXI_WSTRB(byte_index) = '1' ) then
424             -- Respective byte enables are asserted as per
425 write strobes
426             -- slave register 9
427             slv_reg9(byte_index*8+7 downto byte_index*8) <=
428 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
429         end if;
430     end loop;
431 when b"01010" =>
432     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
433         if ( S_AXI_WSTRB(byte_index) = '1' ) then
434             -- Respective byte enables are asserted as per
435 write strobes
436             -- slave register 10
437             slv_reg10(byte_index*8+7 downto byte_index*8) <=
438 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
439         end if;
440     end loop;
441 when b"01011" =>
442     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
443         if ( S_AXI_WSTRB(byte_index) = '1' ) then
444             -- Respective byte enables are asserted as per
445 write strobes
446             -- slave register 11
447             slv_reg11(byte_index*8+7 downto byte_index*8) <=
448 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
449         end if;
450     end loop;
451 when b"01100" =>
452     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
453         if ( S_AXI_WSTRB(byte_index) = '1' ) then
454             -- Respective byte enables are asserted as per
455 write strobes
456             -- slave register 12
457             slv_reg12(byte_index*8+7 downto byte_index*8) <=
458 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
459         end if;
460     end loop;
461 when b"01101" =>
462     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
463         if ( S_AXI_WSTRB(byte_index) = '1' ) then
464             -- Respective byte enables are asserted as per
465 write strobes
466             -- slave register 13
467             slv_reg13(byte_index*8+7 downto byte_index*8) <=
468 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
469         end if;
470     end loop;
471 when b"01110" =>
472     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
473         if ( S_AXI_WSTRB(byte_index) = '1' ) then
474

```

```

475         -- Respective byte enables are asserted as per
476 write strobes
477         -- slave register 14
478         slv_reg14(byte_index*8+7 downto byte_index*8) <=
479 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
480     end if;
481 end loop;
482 when b"01111" =>
483     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
484         if ( S_AXI_WSTRB(byte_index) = '1' ) then
485             -- Respective byte enables are asserted as per
486 write strobes
487             -- slave register 15
488             slv_reg15(byte_index*8+7 downto byte_index*8) <=
489 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
490         end if;
491     end loop;
492 when b"10000" =>
493     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
494         if ( S_AXI_WSTRB(byte_index) = '1' ) then
495             -- Respective byte enables are asserted as per
496 write strobes
497             -- slave register 16
498             slv_reg16(byte_index*8+7 downto byte_index*8) <=
499 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
500         end if;
501     end loop;
502 when b"10001" =>
503     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
504         if ( S_AXI_WSTRB(byte_index) = '1' ) then
505             -- Respective byte enables are asserted as per
506 write strobes
507             -- slave register 17
508             slv_reg17(byte_index*8+7 downto byte_index*8) <=
509 S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
510         end if;
511     end loop;
512 when b"10010" =>
513     for byte_index in 0 to (C_S_AXI_DATA_WIDTH/8-1) loop
514         if ( S_AXI_WSTRB(byte_index) = '1' ) then
515             -- Respective byte enables are asserted as per
516 write strobes
517             -- slave register 18
518             --slv_reg18(byte_index*8+7 downto byte_index*8)
519 <= S_AXI_WDATA(byte_index*8+7 downto byte_index*8);
520         end if;
521     end loop;
522 when others =>
523     slv_reg0 <= slv_reg0;
524     slv_reg1 <= slv_reg1;
525     slv_reg2 <= slv_reg2;
526     slv_reg3 <= slv_reg3;
527     slv_reg4 <= slv_reg4;
528     slv_reg5 <= slv_reg5;
529     slv_reg6 <= slv_reg6;
530     slv_reg7 <= slv_reg7;
531     slv_reg8 <= slv_reg8;
532     slv_reg9 <= slv_reg9;
533     slv_reg10 <= slv_reg10;
534     slv_reg11 <= slv_reg11;
535     slv_reg12 <= slv_reg12;

```

```

536         slv_reg13 <= slv_reg13;
537         slv_reg14 <= slv_reg14;
538         slv_reg15 <= slv_reg15;
539         slv_reg16 <= slv_reg16;
540         slv_reg17 <= slv_reg17;
541         --slv_reg18 <= slv_reg18;
542     end case;
543 end if;
544 end if;
545 end if;
546 end process;
547
548     -- Implement write response logic generation
549     -- The write response and response valid signals are asserted by
550 the slave
551     -- when axi_wready, S_AXI_WVALID, axi_wready and S_AXI_WVALID are
552 asserted.
553     -- This marks the acceptance of address and indicates the status
554 of
555     -- write transaction.
556
557 process (S_AXI_ACLK)
558 begin
559     if rising_edge(S_AXI_ACLK) then
560         if S_AXI_ARESETN = '0' then
561             axi_bvalid <= '0';
562             axi_bresp <= "00"; --need to work more on the responses
563         else
564             if (axi_awready = '1' and S_AXI_AWVALID = '1' and
565 axi_wready = '1' and S_AXI_WVALID = '1' and axi_bvalid = '0' ) then
566                 axi_bvalid <= '1';
567                 axi_bresp <= "00";
568             elsif (S_AXI_BREADY = '1' and axi_bvalid = '1') then --
569 check if bready is asserted while bvalid is high)
570                 axi_bvalid <= '0'; --
571 (there is a possibility that bready is always asserted high)
572             end if;
573         end if;
574     end if;
575 end process;
576
577     -- Implement axi_arready generation
578     -- axi_arready is asserted for one S_AXI_ACLK clock cycle when
579     -- S_AXI_ARVALID is asserted. axi_arready is
580     -- de-asserted when reset (active low) is asserted.
581     -- The read address is also latched when S_AXI_ARVALID is
582     -- asserted. axi_araddr is reset to zero on reset assertion.
583 process (S_AXI_ACLK)
584 begin
585     if rising_edge(S_AXI_ACLK) then
586         if S_AXI_ARESETN = '0' then
587             axi_arready <= '0';
588             axi_araddr <= (others => '1');
589         else
590             if (axi_arready = '0' and S_AXI_ARVALID = '1') then
591 address
592                 axi_arready <= '1';
593                 -- Read Address latching
594                 axi_araddr <= S_AXI_ARADDR;
595             else
596                 axi_arready <= '0';

```

```

597         end if;
598     end if;
599     end if;
600 end process;
601 -- Implement axi_arvalid generation
602 -- axi_rvalid is asserted for one S_AXI_ACLK clock cycle when
603 both
604 -- S_AXI_ARVALID and axi_arready are asserted. The slave
605 registers
606 -- data are available on the axi_rdata bus at this instance. The
607 -- assertion of axi_rvalid marks the validity of read data on the
608 -- bus and axi_rresp indicates the status of read
609 transaction.axi_rvalid
610 -- is deasserted on reset (active low). axi_rresp and axi_rdata
611 are
612 -- cleared to zero on reset (active low).
613 process (S_AXI_ACLK)
614 begin
615     if rising_edge(S_AXI_ACLK) then
616         if S_AXI_ARESETN = '0' then
617             axi_rvalid <= '0';
618             axi_rresp <= "00";
619         else
620             if (axi_arready = '1' and S_AXI_ARVALID = '1' and
621 axi_rvalid = '0') then
622                 -- Valid read data is available at the read data bus
623                 axi_rvalid <= '1';
624                 axi_rresp <= "00"; -- 'OKAY' response
625             elsif (axi_rvalid = '1' and S_AXI_RREADY = '1') then
626                 -- Read data is accepted by the master
627                 axi_rvalid <= '0';
628             end if;
629         end if;
630     end if;
631 end process;
632 -- Implement memory mapped register select and read logic
633 generation
634 -- Slave register read enable is asserted when valid address is
635 available
636 -- and the slave is ready to accept the read address.
637 slv_reg_rden <= axi_arready and S_AXI_ARVALID and (not
638 axi_rvalid) ;
639 process (slv_reg0, slv_reg1, slv_reg2, slv_reg3, slv_reg4,
640 slv_reg5, slv_reg6, slv_reg7, slv_reg8, slv_reg9, slv_reg10,
641 slv_reg11, slv_reg12, slv_reg13, slv_reg14, slv_reg15, slv_reg16,
642 slv_reg17, slv_reg18, axi_araddr, S_AXI_ARESETN, slv_reg_rden)
643     variable loc_addr :std_logic_vector(OPT_MEM_ADDR_BITS downto 0);
644     begin
645         if S_AXI_ARESETN = '0' then
646             reg_data_out <= (others => '1');
647         else
648             -- Address decoding for reading registers
649             loc_addr := axi_araddr(ADDR_LSB + OPT_MEM_ADDR_BITS downto
650 ADDR_LSB);
651             case loc_addr is
652                 when b"00000" =>
653                     reg_data_out <= slv_reg0;
654                 when b"00001" =>
655                     reg_data_out <= slv_reg1;
656                 when b"00010" =>
657                     reg_data_out <= slv_reg2;

```

```

658         when b"00011" =>
659             reg_data_out <= slv_reg3;
660         when b"00100" =>
661             reg_data_out <= slv_reg4;
662         when b"00101" =>
663             reg_data_out <= slv_reg5;
664         when b"00110" =>
665             reg_data_out <= slv_reg6;
666         when b"00111" =>
667             reg_data_out <= slv_reg7;
668         when b"01000" =>
669             reg_data_out <= slv_reg8;
670         when b"01001" =>
671             reg_data_out <= slv_reg9;
672         when b"01010" =>
673             reg_data_out <= slv_reg10;
674         when b"01011" =>
675             reg_data_out <= slv_reg11;
676         when b"01100" =>
677             reg_data_out <= slv_reg12;
678         when b"01101" =>
679             reg_data_out <= slv_reg13;
680         when b"01110" =>
681             reg_data_out <= slv_reg14;
682         when b"01111" =>
683             reg_data_out <= slv_reg15;
684         when b"10000" =>
685             reg_data_out <= slv_reg16;
686         when b"10001" =>
687             reg_data_out <= slv_reg17;
688         when b"10010" =>
689             reg_data_out <= slv_reg18;
690         when others =>
691             reg_data_out <= (others => '0');
692     end case;
693 end if;
694 end process;
695 -- Output register or memory read data
696 process( S_AXI_ACLK ) is
697 begin
698     if (rising_edge (S_AXI_ACLK)) then
699         if ( S_AXI_ARESETN = '0' ) then
700             axi_rdata <= (others => '0');
701         else
702             if (slv_reg_rden = '1') then
703                 -- When there is a valid read address (S_AXI_ARVALID)
704                 with
705                     -- acceptance of read address by the slave (axi_arready),
706                     -- output the read data
707                     -- Read address mux
708                     axi_rdata <= reg_data_out;      -- register read data
709                 end if;
710             end if;
711         end if;
712     end process;
713     -- Add user logic here
714     -- User logic ends
715 end arch_imp;
716
717

```

D.7- Código VHDL: BLOCO “Regulacao.vhd”

Este código fonte descreve o funcionamento do algoritmo para o sistema de gerenciamento da distribuição e controle (DMS&C) feito pelo FPGA, esquematizado na figura 3.2 (no capítulo 3). Os dados são recebidos por barramentos paralelos de dados (posições dos TAP no regulador de tensão e regulação das 5 cargas críticas, para as fases A, B e C), processados no FPGA e então tem como saída um dado em outro barramento paralelo de dados (resposta do controle feito pelo FPGA). Este arquivo é sintetizado pelo *Vivado Design Suite* e implementado no FPGA Artix-7.

```

001 library IEEE;
002 use IEEE.STD_LOGIC_1164.ALL;
003 use IEEE.NUMERIC_STD.ALL;
004
005 entity Regulacao is
006     Port ( clk : in STD_LOGIC;      -- Clock do Sistema
007           tap_A : in STD_LOGIC_VECTOR(4 downto 0); -- -8 to 8
008           tap_B : in STD_LOGIC_VECTOR(4 downto 0); -- -8 to 8
009           tap_C : in STD_LOGIC_VECTOR(4 downto 0); -- -8 to 8
010           R5a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
011           R4a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
012           R3a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
013           R2a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
014           R1a : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
015           R5b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
016           R4b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
017           R3b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
018           R2b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
019           R1b : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
020           R5c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
021           R4c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
022           R3c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
023           R2c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
024           R1c : in STD_LOGIC_VECTOR(10 downto 0); -- -1000 to 1000
025           rec_Ra_Rb_Rc_AtN : out STD_LOGIC_VECTOR(6 downto 0)); --
026     dados de saída combinados com record
027     -- rec_Ra_Rb_Rc_AtN[6..5] = resposta da fase A
028     -- rec_Ra_Rb_Rc_AtN[4..3] = resposta da fase B
029     -- rec_Ra_Rb_Rc_AtN[2..1] = resposta da fase C
030     -- rec_Ra_Rb_Rc_AtN[0]    = resposta geral do alimentador
031     -- 0=Aumentar TAP/ 1=Diminuir TAP/ 2=Atende As Normas/3=Nao Atende As
032     Normas
033     -- 0=Nao Atende As Normas / 1=Atende As Normas
034
035 end Regulacao;
036
037 architecture Behavioral of Regulacao is
038
039     signal int_tap_A : integer range -8 to 8;
040     signal int_tap_B : integer range -8 to 8;
041     signal int_tap_C : integer range -8 to 8;
042     signal int_R5a : integer range -1000 to 1000;
043     signal int_R4a : integer range -1000 to 1000;
044     signal int_R3a : integer range -1000 to 1000;
045     signal int_R2a : integer range -1000 to 1000;
046     signal int_R1a : integer range -1000 to 1000;
047     signal int_R5b : integer range -1000 to 1000;

```

```

048     signal int_R4b      : integer range -1000 to 1000;
049     signal int_R3b      : integer range -1000 to 1000;
050     signal int_R2b      : integer range -1000 to 1000;
051     signal int_R1b      : integer range -1000 to 1000;
052     signal int_R5c      : integer range -1000 to 1000;
053     signal int_R4c      : integer range -1000 to 1000;
054     signal int_R3c      : integer range -1000 to 1000;
055     signal int_R2c      : integer range -1000 to 1000;
056     signal int_R1c      : integer range -1000 to 1000;
057
058     type pacote_saida is record
059         rec_Resp_A : std_logic_vector(1 downto 0);
060         rec_Resp_B : std_logic_vector(1 downto 0);
061         rec_Resp_C : std_logic_vector(1 downto 0);
062     end record;
063     signal resp_saida : pacote_saida;
064
065 begin
066
067     int_tap_A <= to_integer(signed(tap_A));
068     int_tap_B <= to_integer(signed(tap_B));
069     int_tap_C <= to_integer(signed(tap_C));
070     int_R5a <= to_integer(signed(R5a));
071     int_R4a <= to_integer(signed(R4a));
072     int_R3a <= to_integer(signed(R3a));
073     int_R2a <= to_integer(signed(R2a));
074     int_R1a <= to_integer(signed(R1a));
075     int_R5b <= to_integer(signed(R5b));
076     int_R4b <= to_integer(signed(R4b));
077     int_R3b <= to_integer(signed(R3b));
078     int_R2b <= to_integer(signed(R2b));
079     int_R1b <= to_integer(signed(R1b));
080     int_R5c <= to_integer(signed(R5c));
081     int_R4c <= to_integer(signed(R4c));
082     int_R3c <= to_integer(signed(R3c));
083     int_R2c <= to_integer(signed(R2c));
084     int_R1c <= to_integer(signed(R1c));
085
086     Verificacao_Taps:process(clk)
087     begin
088         if(clk'event and clk='1') then -- Borda de subida do clock
089
090             -- Verificacoes no enrolamento A
091             if(int_R5a <= 70 and int_R4a <= 70 and int_R3a <= 70 and
092 int_R2a <= 70 and int_R1a <= 70) then
093                 if(int_R5a >= -50 and int_R4a >= -50 and int_R3a >= -
094 50 and int_R2a >= -50 and int_R1a >= -50) then
095                     resp_saida.rec_Resp_A <= "10"; --Atende As Normas
096                 else
097                     if(int_tap_A = 8) then
098                         resp_saida.rec_Resp_A <= "11"; --Nao Atende
099 As Normas
100                     else
101                         resp_saida.rec_Resp_A <= "00"; --Aumentar TAP
102                     end if;
103                 end if;
104             else
105                 if(int_tap_A = -8) then
106                     resp_saida.rec_Resp_A <= "11"; --Nao Atende As
107 Normas
108                 else

```

```

109         resp_saida.rec_Resp_A <= "01"; -- Diminuir TAP
110     end if;
111 end if;
112
113     -- Verificacoes no enrolamento B
114     if(int_R5b <= 70 and int_R4b <= 70 and int_R3b <= 70 and
115 int_R2b <= 70 and int_R1b <= 70) then
116         if(int_R5b >= -50 and int_R4b >= -50 and int_R3b >= -
117 50 and int_R2b >= -50 and int_R1b >= -50) then
118             resp_saida.rec_Resp_B <= "10"; --Atende As Normas
119         else
120             if(int_tap_B = 8) then
121                 resp_saida.rec_Resp_B <= "11"; --Nao Atende
122 As Normas
123             else
124                 resp_saida.rec_Resp_B <= "00"; --Aumentar TAP
125             end if;
126         end if;
127     else
128         if(int_tap_B = -8) then
129             resp_saida.rec_Resp_B <= "11"; --Nao Atende As
130 Normas
131         else
132             resp_saida.rec_Resp_B <= "01"; --Diminuir TAP
133         end if;
134     end if;
135
136     -- Verificacoes no enrolamento C
137     if(int_R5c <= 70 and int_R4c <= 70 and int_R3c <= 70 and
138 int_R2c <= 70 and int_R1c <= 70) then
139         if(int_R5c >= -50 and int_R4c >= -50 and int_R3c >= -
140 50 and int_R2c >= -50 and int_R1c >= -50) then
141             resp_saida.rec_Resp_C <= "10"; --Atende As Normas
142         else
143             if(int_tap_C = 8) then
144                 resp_saida.rec_Resp_C <= "11"; --Nao Atende
145 As Normas
146             else
147                 resp_saida.rec_Resp_C <= "00"; --Aumentar TAP
148             end if;
149         end if;
150     else
151         if(int_tap_C = -8) then
152             resp_saida.rec_Resp_C <= "11"; --Nao Atende As
153 Normas
154         else
155             resp_saida.rec_Resp_C <= "01"; -- Diminuir TAP
156         end if;
157     end if;
158 end if;
159 end process;
160
161 Verificacao_Alimentador:process(resp_saida)
162 begin
163     if(resp_saida.rec_Resp_A = "10" and resp_saida.rec_Resp_B =
164 "10" and resp_saida.rec_Resp_C = "10") then
165         rec_Ra_Rb_Rc_AtN(6 downto 5) <= resp_saida.rec_Resp_A;
166         rec_Ra_Rb_Rc_AtN(4 downto 3) <= resp_saida.rec_Resp_B;
167         rec_Ra_Rb_Rc_AtN(2 downto 1) <= resp_saida.rec_Resp_C;
168         rec_Ra_Rb_Rc_AtN(0) <= '1'; -- Atende As Normas
169     else

```

```
170         rec_Ra_Rb_Rc_AtN(6 downto 5) <= resp_saida.rec_Resp_A;
171         rec_Ra_Rb_Rc_AtN(4 downto 3) <= resp_saida.rec_Resp_B;
172         rec_Ra_Rb_Rc_AtN(2 downto 1) <= resp_saida.rec_Resp_C;
173         rec_Ra_Rb_Rc_AtN(0) <= '0';      -- Nao Atende As Normas
174     end if;
175 end process;
176
177 end Behavioral;
```

D.8- CÓDIGO EM LINGUAGEM DE PROGRAMAÇÃO C: “main.c”

Tomando em consideração que o projeto do DMS&C utiliza um dispositivo ZYNQ SoC XC7Z020-CLG484, que encapsula um FPGA Artix-7 e um processador ARM Cortex-A9 de núcleo duplo, foi escolhido o processador ARM como o master do sistema de controle. Baseados nisso, este código fonte tem a função de verificar o status de uma porta serial que é emulada no PC pela placa de desenvolvimento ZedBoard Zynq™-7000, coletando assim todos os dados recebidos por ela e verificando se os mesmos são dados de um pacote válido. Sendo um pacote válido o próximo passo é extrair os dados do pacote e enviá-los para o FPGA, através da interface AXI-Lite. Após o processamento dos dados pelo FPGA o resultado é lido através da interface AXI-Lite, para depois ser montado um pacote de dados de resposta e, enviado à porta serial. Este arquivo é compilado pelo kit de desenvolvimento de software - SDK e executado pelo processador ARM Cortex-A9.

```

001 #include "xparameters.h"
002 #include "xstatus.h"
003 #include "xil_types.h"
004 #include "xil_assert.h"
005 #include "xuartps_hw.h"
006 #include "xil_printf.h"
007 #include "sleep.h"
008 #include "platform.h"
009 #include "Regulacao_Transf.h"
010 #include "CRC16.h"
011 #include <string.h>
012 #include <stdlib.h>
013 #include <stdio.h>
014 /***** Constant Definitions
015 *****/
016 #define REGULACAO_TRANSF_BASEADDR    0x43C00000
017 /*
018  * The following constants map to the XPAR parameters created in the
019  * xparameters.h file. They are defined here such that a user can
020 easily
021  * change all the needed parameters in one place.
022  */
023 #define UART_BASEADDR                XPAR_XUARTPS_0_BASEADDR
024 #define UART_CLOCK_HZ                XPAR_XUARTPS_0_CLOCK_HZ
025 /*
026  * The following constant controls the length of the buffers to be
027 sent
028  * and received with the device. This constant must be 32 bytes or
029 less so the
030  * entire buffer will fit into the TX and RX FIFOs of the device.
031  */
032 #define SEND_BUFFER_SIZE             100
033 #define RECV_BUFFER_SIZE             100
034 #define NUN_MAX_LEN                   6
035 /***** Type Definitions
036 *****/
037 /***** Macros (Inline Functions) Definitions
038 *****/
039 /***** Function Prototypes
040 *****/
041 void init_my_uart(void);
042 u32 get_serial_string(void);

```

```

043 void split_data(void);
044 void IP_regulacao(void);
045 void responde_resultado(void);
046
047 /***** Variable Definitions
048 *****/
049 unsigned char SendBuffer[SEND_BUFFER_SIZE];
050 unsigned char RecvBuffer[RECV_BUFFER_SIZE];
051 unsigned char RecvChar;
052 unsigned char num_convert[20][NUN_MAX_LEN];
053 short Tap[3];
054 short R[5][3];
055 short IP_resposta;
056
057 int main(void)
058 {
059     u32 recepcao_valida = FALSE;           // check de recepcao valida
060
061     init_platform();                       // configura o ARM
062     init_my_uart();                       // inicializa a serial
063
064     while(TRUE)
065     {
066         recepcao_valida = get_serial_string(); // recebe string pela
067         serial
068
069         if(recepcao_valida) //verifica se e um pacote de dados valido
070         {
071             split_data();                 // extrai os dados do pacote
072             IP_regulacao();               // envia os dados para o FPGA
073             responde_resultado();         //monta o pacote para envio da
074             serial
075             recepcao_valida = FALSE;
076         }
077     }
078     return XST_SUCCESS;
079 }
080
081 void IP_regulacao()
082 {
083     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
084 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG0_OFFSET, Tap[0]);
085     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
086 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG1_OFFSET, Tap[1]);
087     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
088 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG2_OFFSET, Tap[2]);
089
090     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
091 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG3_OFFSET, R[0][0]);
092     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
093 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG4_OFFSET, R[1][0]);
094     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
095 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG5_OFFSET, R[2][0]);
096     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
097 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG6_OFFSET, R[3][0]);
098     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
099 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG7_OFFSET, R[4][0]);
100
101     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
102 REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG8_OFFSET, R[0][1]);
103

```

```

104     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
105     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG9_OFFSET, R[1][1]);
106     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
107     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG10_OFFSET, R[2][1]);
108     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
109     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG11_OFFSET, R[3][1]);
110     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
111     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG12_OFFSET, R[4][1]);
112
113     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
114     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG13_OFFSET, R[0][2]);
115     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
116     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG14_OFFSET, R[1][2]);
117     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
118     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG15_OFFSET, R[2][2]);
119     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
120     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG16_OFFSET, R[3][2]);
121     REGULACAO_TRANSF_mWriteReg(REGULACAO_TRANSF_BASEADDR,
122     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG17_OFFSET, R[4][2]);
123
124     IP_resposta =
125     REGULACAO_TRANSF_mReadReg(REGULACAO_TRANSF_BASEADDR,
126     REGULACAO_TRANSF_S00_AXI_TRANSF_SLV_REG18_OFFSET);
127 }
128
129 void responde_resultado()
130 {
131     u16 CRC_check;
132     u8 count = 0;
133
134     SendBuffer[0] = 'B';
135     SendBuffer[1] = 'C';
136     SendBuffer[2] = 4;
137     SendBuffer[3] = IP_resposta&0xFF;
138     CRC_check = FindCRC(&SendBuffer[0], SendBuffer[2]);
139     SendBuffer[4] = (unsigned char)(CRC_check&0xFF);
140     SendBuffer[5] = (unsigned char)((CRC_check&0xFF00)>>8);
141     SendBuffer[6] = '\0';
142
143     while(count < 6)
144     {
145         XUartPs_WriteReg(UART_BASEADDR, XUARTPS_FIFO_OFFSET,
146         SendBuffer[count]);
147         count++;
148     }
149 }
150
151 void init_my_uart()
152 {
153     int Index;
154     u32 CntrlRegister;
155
156     CntrlRegister = XUartPs_ReadReg(UART_BASEADDR,
157     XUARTPS_CR_OFFSET);
158     /*
159     * Enable TX and RX for the device
160     */
161     XUartPs_WriteReg(UART_BASEADDR, XUARTPS_CR_OFFSET,
162     ((CntrlRegister & ~XUARTPS_CR_EN_DIS_MASK) |
163     XUARTPS_CR_TX_EN | XUARTPS_CR_RX_EN));
164

```

```

165     for (Index = 0; Index < SEND_BUFFER_SIZE; Index++) {
166         SendBuffer[Index] = '\0';
167     }
168 }
169
170 u32 get_serial_string()
171 {
172     u8 string_count = 0;
173     u32 fim_string = 0;
174     u16 CRC_check;
175     u16 CRC_sent;
176
177     while (!XUartPs_IsReceiveData(UART_BASEADDR));
178     // <-- Espera receber algum dado pela serial
179     RecvBuffer[0] =
180 XUartPs_ReadReg(UART_BASEADDR, XUARTPS_FIFO_OFFSET); // <-- Recebe
181 o dado em RecvChar
182
183     if(RecvBuffer[0] == 'A')
184     {
185         while (!XUartPs_IsReceiveData(UART_BASEADDR));
186         RecvBuffer[1] =
187 XUartPs_ReadReg(UART_BASEADDR, XUARTPS_FIFO_OFFSET);
188         if(RecvBuffer[1] == 'B')
189         {
190             string_count = 3;
191             while (!XUartPs_IsReceiveData(UART_BASEADDR));
192             RecvBuffer[2] =
193 XUartPs_ReadReg(UART_BASEADDR, XUARTPS_FIFO_OFFSET);
194
195             while((string_count < RECV_BUFFER_SIZE) && (fim_string <
196 RecvBuffer[2]-1))
197             {
198                 while (!XUartPs_IsReceiveData(UART_BASEADDR));
199                 RecvBuffer[string_count] =
200 XUartPs_ReadReg(UART_BASEADDR, XUARTPS_FIFO_OFFSET);
201                 string_count++;
202                 fim_string++;
203
204                 if(RecvBuffer[2]-1 == fim_string)
205                 {
206                     CRC_check = FindCRC(&RecvBuffer[0],
207 RecvBuffer[2]);
208                     CRC_sent =
209 ((RecvBuffer[RecvBuffer[2]+1]<<8)&0xFF00) +
210 (RecvBuffer[RecvBuffer[2]]&0x00FF);
211
212                     if(CRC_check == CRC_sent)
213                     {
214                         return TRUE;
215                     }
216                 }
217             }
218         }
219     }
220     return FALSE;
221 }
222
223 void split_data()
224 {
225     Tap[0] = ((RecvBuffer[3]<<8)&0xFF00) + (RecvBuffer[4]&0x00FF);

```

```

226     Tap[1] = ((RecvBuffer[5]<<8)&0xFF00) + (RecvBuffer[6]&0x00FF);
227     Tap[2] = ((RecvBuffer[7]<<8)&0xFF00) + (RecvBuffer[8]&0x00FF);
228
229     R[0][0] = ((RecvBuffer[9]<<8)&0xFF00) + (RecvBuffer[10]&0x00FF);
230     R[0][1] = ((RecvBuffer[11]<<8)&0xFF00) + (RecvBuffer[12]&0x00FF);
231     R[0][2] = ((RecvBuffer[13]<<8)&0xFF00) + (RecvBuffer[14]&0x00FF);
232
233     R[1][0] = ((RecvBuffer[15]<<8)&0xFF00) + (RecvBuffer[16]&0x00FF);
234     R[1][1] = ((RecvBuffer[17]<<8)&0xFF00) + (RecvBuffer[18]&0x00FF);
235     R[1][2] = ((RecvBuffer[19]<<8)&0xFF00) + (RecvBuffer[20]&0x00FF);
236
237     R[2][0] = ((RecvBuffer[21]<<8)&0xFF00) + (RecvBuffer[22]&0x00FF);
238     R[2][1] = ((RecvBuffer[23]<<8)&0xFF00) + (RecvBuffer[24]&0x00FF);
239     R[2][2] = ((RecvBuffer[25]<<8)&0xFF00) + (RecvBuffer[26]&0x00FF);
240
241     R[3][0] = ((RecvBuffer[27]<<8)&0xFF00) + (RecvBuffer[28]&0x00FF);
242     R[3][1] = ((RecvBuffer[29]<<8)&0xFF00) + (RecvBuffer[30]&0x00FF);
243     R[3][2] = ((RecvBuffer[31]<<8)&0xFF00) + (RecvBuffer[32]&0x00FF);
244
245     R[4][0] = ((RecvBuffer[33]<<8)&0xFF00) + (RecvBuffer[34]&0x00FF);
246     R[4][1] = ((RecvBuffer[35]<<8)&0xFF00) + (RecvBuffer[36]&0x00FF);
247     R[4][2] = ((RecvBuffer[37]<<8)&0xFF00) + (RecvBuffer[38]&0x00FF);
    }

```

D.9- CÓDIGO EM MICROSOFT VISUAL BASIC: “mainModuleChk.bas”

Este código fonte tem a função de estabelecer uma interface de comunicação entre o PC e a placa ZedBoard. Neste sentido, o software: 1) Comunica a placa ZedBoard com uma porta serial que é emulada no PC pela mesma placa, utilizando o **Universal Serial Bus** (USB) do PC, 2) Envia os resultados da simulação em VHDL-AMS (em um arquivo de texto) para a porta serial (emulada no PC), 3) Estabelece um tempo de espera para a execução do algoritmo digital no FPGA Artix-7 e, 4) Visualiza, no PC, os resultados da simulação digital em VHDL, feita pelo dispositivo FPGA.

```

001 Attribute VB_Name = "MainModule"
002 Option Explicit
003
004 Private OnBits(0 To 31) As Long
005 Public LoginSucceeded As Boolean
006 Public isBegin As Boolean
007
008 Sub SerialEnvio(o As Form)
009 On Error GoTo err_SerialEnvio
010     Dim resposta As String
011     Dim pergunta As String
012     o.MSComm1.CommPort = o.txtPortaSerial
013     o.MSComm1.Settings = Mid(o.lblSerialSettings, 13)
014     o.MSComm1.PortOpen = True
015     o.MSComm1.InputLen = 0
016     EscreveMessage o, "Acessando usando " & o.lblSerialSettings &
017 String(16, " ") & "Aguardando resposta...", True
018     pergunta = SerialEscrita(o, "AB")
019     resposta = SerialLeitura(o, "BC", 26)
020     o.MSComm1.PortOpen = False
021 Exit Sub
022 err_SerialEnvio:
023     If Err.Number = 8002 Or Err.Number = 380 Then EscreveMessage o,
024 "Porta Serial errada ou desabilitada", True
025     If Err.Number = 8005 Then EscreveMessage o, "Porta Serial Aberta",
026 True
027 Exit Sub
028 End Sub
029
030 Private Function SerialEscrita(o As Form, Prefixo As String)
031 Dim tmpStr As String
032 Select Case True
033     Case Prefixo = "AB"
034         tmpStr = LeituraDadosFPGA(o)
035         Step 2
036         If tmpStr <> "" Then
037             o.MSComm1.Output = tmpStr
038         End If
039 End Select
040 End Function
041
042 Function LeituraDadosFPGA(o As Form) As String
043 Dim i As Integer
044 Dim indice As Integer
045 Dim valLen1, valLen2 As Integer
046 Dim dadoLido As String
047 Dim textLine As String

```

```

048 Dim lineNumber As Integer
049 Dim nrDeCLTP As Integer
050 Dim tmpStr As String
051 Dim x() As String
052     dadoLido = "AB*"
053     lineNumber = 1
054     EscreveMessage o, "", True
055     Open o.txtArquivoDeDados For Input As #1
056     Do Until EOF(1)
057         Line Input #1, textLine
058         If lineNumber = 1 And Not (Mid(textLine, 1, 4) = "Posi"
059 Or Mid(textLine, 1, 4) = ""Pos") Then
060             EscreveMessage o, "Erro na primeira linha do arquivo
061 de dados.", True
062             Exit Function
063         ElseIf lineNumber = 2 And (InStr(textLine, "Tap A") = 0
064 Or InStr(textLine, "Tap B") = 0 Or InStr(textLine, "Tap C") = 0) Then
065             EscreveMessage o, "Erro na segunda linha do arquivo
066 de dados.", True
067             Exit Function
068         ElseIf lineNumber = 3 Then
069             textLine = eliminaEspaçosDuplos(textLine)
070             x = Split(textLine, " ")
071             If Not IsNumeric(x(0)) Or Not IsNumeric(x(1)) Or Not
072 IsNumeric(x(2)) Then
073                 EscreveMessage o, "Erro em um dos TAPs.", True
074                 Exit Function
075             End If
076             If x(0) < -8 Or x(1) < -8 Or x(2) < -8 Then
077                 EscreveMessage o, "Erro em um dos TAPs lidos é
078 menor que -8 (ref. dado nr.: " & nrDeCLTP & ").", True
079                 Exit Function
080             End If
081             If x(0) > 8 Or x(1) > 8 Or x(2) > 8 Then
082                 EscreveMessage o, "Erro em um dos TAPs lidos é
083 maior que 8 (ref. dado nr.: " & nrDeCLTP & ").", True
084                 Exit Function
085             End If
086             o.Linha(1).Tag = String(60, "-")
087             o.Linha(11).Tag = String(60, "-")
088             o.Linha(2).Tag = Right(String(60, " ") & "Tap A
089 Tap B          Tap C", 60)
090             valLen1 = Len("Posição dos TAPs no RT:")
091             valLen2 = Len(Format(Format(x(0), "#####"),
092 String(12, "@")) & Format(Format(x(1), "#####"), String(12, "@")) &
093 Format(Format(x(2), "#####"), String(12, "@")))
094             o.Linha(3).Tag = Right(String(60, " ") & "Posição dos
095 TAPs no RT:" & String(60 - valLen1 - valLen2, " ") &
096 Format(Format(x(0), "#####"), String(12, "@")) & Format(Format(x(1),
097 "#####"), String(12, "@")) & Format(Format(x(2), "#####"), String(12,
098 "@")), 60)
099             o.Linha(4).Tag = String(60, "-")
100             valLen1 = Len("CARGA CRÍTICA")
101             valLen2 = Len("FASE A          FASE B          FASE C")
102             o.Linha(5).Tag = Right(String(60, " ") & "CARGA
103 CRÍTICA" & String(60 - valLen1 - valLen2, " ") & "FASE A          FASE B
104 FASE C", 60)
105             tmpStr = Right("0000" & Hex(x(0)), 4)
106             dadoLido = dadoLido & Chr(Val("&H" & Left(tmpStr,
107 2))) & Chr(Val("&H" & Right(tmpStr, 2)))
108             tmpStr = Right("0000" & Hex(x(1)), 4)

```

```

109         dadoLido = dadoLido & Chr(Val("&H" & Left(tmpStr,
110 2))) & Chr(Val("&H" & Right(tmpStr, 2)))
111         tmpStr = Right("0000" & Hex(x(2)), 4)
112         dadoLido = dadoLido & Chr(Val("&H" & Left(tmpStr,
113 2))) & Chr(Val("&H" & Right(tmpStr, 2)))
114         ElseIf lineNumber = 5 And Mid(textLine, 1, 6) <> "Regula"
115 Then
116             EscreveMessage o, "Erro na quinta linha do arquivo de
117 dados.", True
118             Exit Function
119         ElseIf lineNumber = 6 And Mid(textLine, 1, 5) <> "CARGA"
120 Then
121             EscreveMessage o, "Erro na sexta linha do arquivo de
122 dados.", True
123             Exit Function
124         ElseIf lineNumber = 7 And textLine <> "" Then
125             textLine = eliminaEspacosDuplos(textLine)
126             textLine = Replace(textLine, "%", ".")
127             'textLine = Replace(textLine, ",", ".")
128             If Len(textLine) > 1 Then
129                 x = Split(textLine, " ")
130                 If o.Linha(5).Tag = "" Then
131                     indice = 5
132                 ElseIf o.Linha(6).Tag = "" Then
133                     indice = 6
134                 ElseIf o.Linha(7).Tag = "" Then
135                     indice = 7
136                 ElseIf o.Linha(8).Tag = "" Then
137                     indice = 8
138                 ElseIf o.Linha(9).Tag = "" Then
139                     indice = 9
140                 ElseIf o.Linha(10).Tag = "" Then
141                     indice = 10
142                 End If
143                 valLen1 = Len(x(0))
144                 valLen2 = Len(Format(Format(x(1),
145 "#####.0###\%"), String(12, "@")) & Format(Format(x(2),
146 "#####.0###\%"), String(12, "@")) & Format(Format(x(3),
147 "#####.0###\%"), String(12, "@")))
148                 o.Linha(indice).Tag = Right(String(60, " ") &
149 x(0) & String(60 - valLen1 - valLen2, " ") & Format(Format(x(1),
150 "#####.0###\%"), String(12, "@")) & Format(Format(x(2),
151 "#####.0###\%"), String(12, "@")) & Format(Format(x(3),
152 "#####.0###\%"), String(12, "@")), 60)
153                 If Not IsNumeric(x(1)) Or Not IsNumeric(x(2)) Or
154 Not IsNumeric(x(3)) Then
155                     EscreveMessage o, "Erro em um dos valores
156 lidos (ref. dado nr.: " & nrDeCLTP & ").", True
157                     Exit Function
158                 End If
159                 If x(1) < -1000 Or x(2) < -1000 Or x(3) < -1000
160 Then
161                     EscreveMessage o, "Erro em um dos valores
162 lidos é menor que -1000 (ref. dado nr.: " & nrDeCLTP & ").", True
163                     Exit Function
164                 End If
165                 If x(1) > 1000 Or x(2) > 1000 Or x(3) > 1000 Then
166                     EscreveMessage o, "Erro em um dos valores
167 lidos é maior que 1000 (ref. dado nr.: " & nrDeCLTP & ").", True
168                     Exit Function
169                 End If

```

```

170         x(1) = CInt(x(1) * 10)
171         x(2) = CInt(x(2) * 10)
172         x(3) = CInt(x(3) * 10)
173         tmpStr = Right("0000" & Hex(x(1)), 4)
174         dadoLido = dadoLido & Chr(Val("&H" & Left(tmpStr,
175 2))) & Chr(Val("&H" & Right(tmpStr, 2)))
176         tmpStr = Right("0000" & Hex(x(2)), 4)
177         dadoLido = dadoLido & Chr(Val("&H" & Left(tmpStr,
178 2))) & Chr(Val("&H" & Right(tmpStr, 2)))
179         tmpStr = Right("0000" & Hex(x(3)), 4)
180         dadoLido = dadoLido & Chr(Val("&H" & Left(tmpStr,
181 2))) & Chr(Val("&H" & Right(tmpStr, 2)))
182         nrDeCLTP = nrDeCLTP + 1
183         End If
184     End If
185     If lineNumber < 7 Then lineNumber = lineNumber + 1
186     Loop
187     Close #1
188     dadoLido = Left(dadoLido, 2) & Chr(2 + 1 + (3 * 2) + (nrDeCLTP *
189 3 * 2)) & Mid(dadoLido, 4)
190     dadoLido = dadoLido & CRCModBus(dadoLido)
191     LeituraDadosFPGA = dadoLido
192 End Function
193
194 Function eliminaEspaçosDuplos(s As String)
195     s = Replace(s, vbTab, " ")
196     Do While InStr(s, " ")
197         s = Replace(s, " ", " ")
198     Loop
199     eliminaEspaçosDuplos = s
200 End Function
201
202 Sub Step(Value As Integer)
203     Dim d As Date
204     d = Now
205     Dim i As Integer
206     Do While Abs(DateDiff("s", Now, d)) < Value
207     Loop
208 End Sub
209
210 Sub iniciarInterface(o As Form)
211 On Error GoTo err_Form_Load
212     Dim i As Integer
213     o.Image1.Move 0, 0, o.Width, o.Height
214     o.Image2.Move 0, 0, o.Width, o.Height
215     o.Image3.Move 0, 0, o.Width, o.Height
216     o.Image4.Move 0, 0, o.Width, o.Height
217     o.Refresh
218     isBegin = True
219     o.Picture1.Left = 0
220     o.Picture1.Top = 0
221     o.Drive1 = GetSetting("TRIFASICO", "Caminho", "Drive", "C:")
222     o.Dir1 = o.Drive1
223     o.Dir1.Path = GetSetting("TRIFASICO", "Caminho", "Dir", "")
224     o.File1.Path = o.Dir1
225     For i = 0 To o.File1.ListCount - 1
226         If o.File1.List(i) = GetSetting("TRIFASICO", "Caminho",
227 "File", "") Then
228             o.File1.Selected(i) = True
229         End If
230     Next

```

```

231     o.txtArquivoDeDados = GetSetting("TRIFASICO", "Caminho",
232     "File&Path", "")
233     SaveSetting "TRIFASICO", "Caminho", "Drive", o.Drive1
234     SaveSetting "TRIFASICO", "Caminho", "Dir", o.Dir1
235     SaveSetting "TRIFASICO", "Caminho", "File", o.File1
236     SaveSetting "TRIFASICO", "Caminho", "File&Path",
237     o.txtArquivoDeDados
238     'Inicia Serial Settings:
239     o.txtPortaSerial = GetSetting("TRIFASICO", "Serial", "Porta",
240     "1")
241     o.txtBaudrate = GetSetting("TRIFASICO", "Serial", "Baudrate",
242     "115200")
243     o.txtBitDeDados = GetSetting("TRIFASICO", "Serial", "BitDados",
244     8)
245     o.txtParidade = GetSetting("TRIFASICO", "Serial", "Paridade",
246     "n")
247     o.txtBitDeParada = GetSetting("TRIFASICO", "Serial", "Parada",
248     "1")
249     o.lblSerialSettings = "Serial Com" & o.txtPortaSerial & _
250     " " & o.txtBaudrate & _
251     ", " & o.txtParidade & _
252     ", " & o.txtBitDeDados & _
253     ", " & o.txtBitDeParada
254     EscreveMessage o, "", True
255     o.MessageTop.Left = 390: o.MessageTop.Top = 5550
256     o.MessageMiddle.Left = 400: o.MessageMiddle.Top = 5565
257     o.MessageBottom.Left = 420: o.MessageBottom.Top = 5580
258     isBegin = False
259     Exit Sub
260 err_Form_Load:
261     If Err.Number = 76 Then
262         o.Dir1.Path = ""
263         Resume Next
264     End If
265 End Sub
266
267 Sub EscondeEscolhaSerialSettings(o As Form, Esconder As Boolean)
268     o.FrameSerialSettings.Visible = Not Esconder
269     o.txtArquivoDeDados.Visible = Esconder
270     o.cmdEnviarArquivoDeDados.Visible = Esconder
271     o.lblSerialSettings.Visible = Esconder
272     EscreveMessage o, "", True
273 End Sub
274
275 Sub EscondeEscolhaArquivoDeDados(o As Form, Esconder As Boolean)
276     o.FrameConfirmarArquivoDeDados.Visible = Not Esconder
277     o.txtArquivoDeDados.Visible = Esconder
278     o.cmdEnviarArquivoDeDados.Visible = Esconder
279     o.lblSerialSettings.Visible = Esconder
280     EscreveMessage o, "", True
281 End Sub
282
283 Private Function vCRC16(Message() As Byte) As Integer
284     Dim i As Long, j As Long, Odd As Boolean
285     vCRC16 = -1
286     For i = 0 To UBound(Message)
287         vCRC16 = (vCRC16 And &HFF00) Or ((vCRC16 And 255) Xor
288         Message(i))
289         For j = 1 To 8
290             Odd = vCRC16 And 1
291

```

```

292         vCRC16 = ((vCRC16 And &HFFFF) \ 2) And &H7FFF 'shift
293     right
294         If Odd Then vCRC16 = &HA001 Xor vCRC16
295     Next
296 Next
297 End Function
298 Function CRCModBus(ByVal txtHexInput As String) As String
299     Dim x As Long
300     Dim mask, i, j, nC, Crc As Integer
301     Dim c As String
302     Dim txt As String
303     txt = Replace(cToSpacedHexa(txtHexInput), " ", "") ' "CA00F7"
304     Crc = &HFFFF ' crc mit $ffff initialisieren
305     For nC = 1 To Len(txt) Step 2
306         j = Val("&H" + Mid(txt, nC, 2)) 'im HEX-Format
307         Crc = Crc Xor j
308         For j = 1 To 8
309             mask = 0
310             If Crc / 2 <> Int(Crc / 2) Then mask = &HA001
311             Crc = Int(Crc / 2) And &H7FFF: Crc = Crc Xor mask
312         Next j
313     Next nC
314     txt = Right("0000" & Hex$(Crc), 4)
315     CRCModBus = Chr(Val("&H" & Right(txt, 2))) & Chr(Val("&H" &
316 Left(txt, 2)))
317 End Function
318
319 Function cToSpacedHexa(Palavra As String) As String
320     Dim mstr As String
321     Dim i As Integer
322     For i = 1 To Len(Palavra)
323         mstr = mstr & Right("00" & Hex(Asc(Mid(Palavra, i, 1))), 2) &
324 " "
325     Next i
326     cToSpacedHexa = mstr
327 End Function
328
329 Function cFromSpacedHexa(Palavra As String) As String
330     Dim mstr As String
331     Dim i As Integer
332     Dim sVar() As String
333     Palavra = Trim(Palavra)
334     sVar = Split(Palavra, " ")
335     For i = 0 To UBound(sVar)
336         mstr = mstr & Chr(Val("&H" & sVar(i)))
337     Next i
338     cFromSpacedHexa = mstr
339 End Function
340
341 Public Function RShiftLong(ByVal Value As Long, _
342     ByVal Shift As Integer) As Long
343     Dim hi As Long
344     MakeOnBits
345     If (Value And &H80000000) Then hi = &H40000000
346     RShiftLong = (Value And &H7FFFFFFF) \ (2 ^ Shift)
347     RShiftLong = (RShiftLong Or (hi \ (2 ^ (Shift - 1))))
348 End Function
349
350 Private Sub MakeOnBits()
351     Dim j As Integer, _
352     v As Long

```

```

353     For j = 0 To 30
354         v = v + (2 ^ j)
355         OnBits(j) = v
356
357     Next j
358     OnBits(j) = v + &H80000000
359 End Sub
360
361 Sub EscreveMessage(o As Form, msg As String, apagaLinhas As Boolean)
362 Dim i As Integer
363 If apagaLinhas Then
364     For i = 0 To 14
365         o.Linha(i) = ""
366         o.Linha(i).Tag = ""
367     Next i
368 End If
369 o.MessageTop = msg
370 o.MessageTop.Refresh
371 End Sub
372
373 Private Function SerialLeitura(o As Form, Prefixo As String, Tempo As
374 Integer)
375 Dim Timer1
376 Dim i As Long
377 Dim strttemp As String
378 Dim start As Integer
379 Dim mstr As String
380 Dim isPrefixed As Boolean
381 Dim d As Date
382 Dim msgInicial As String
383 msgInicial = "Lendo dados"
384 d = Now
385 mstr = ""
386 EscreveMessage o, msgInicial, False
387 LeSerial:
388 Do While Abs(DateDiff("s", Now, d)) < Tempo And
389 o.MSComm1.InBufferCount < 6
390     i = i + 1
391     If i > 65000 Then
392         msgInicial = msgInicial & "."
393         EscreveMessage o, msgInicial, False
394         o.Refresh
395         i = 0
396     End If
397     DoEvents
398 Loop
399 If o.MSComm1.InBufferCount = 0 Then
400     EscreveMessage o, "Porta COM - tempo esgotado", False:
401     '''BEGIN SOMENTE EM TESTES
402     '''cmdTeste.Visible = True
403     '''END SOMENTE EM TESTES
404 Exit Function
405 End If
406 strttemp = o.MSComm1.Input
407 mstr = mstr & strttemp
408 If InStr(mstr, "BC") Then
409     If Right(mstr, 2) = CRCModBus(Left(mstr, 4)) Then
410         Call SerialProcessa(o, Asc(Mid(mstr, 4, 1)))
411     Else
412         GoTo LeSerial
413     End If

```

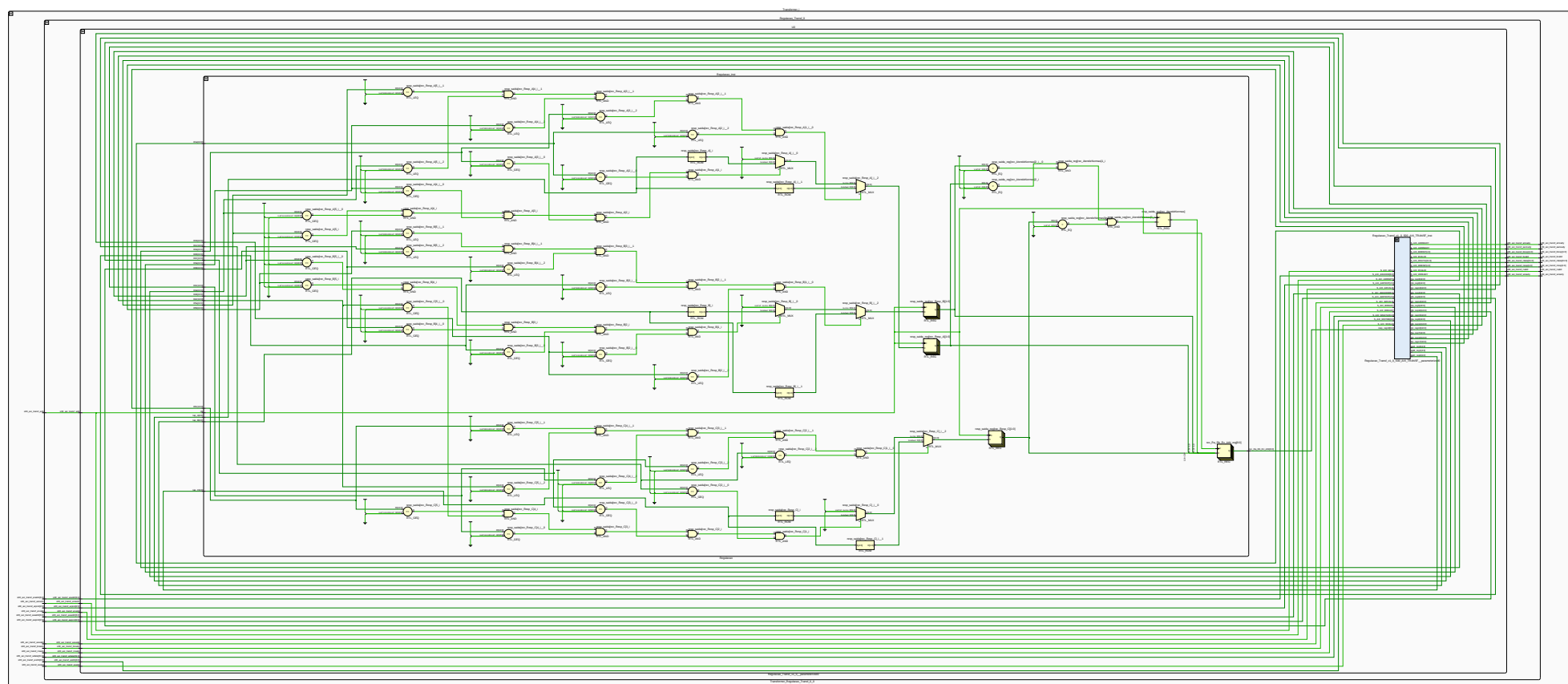
```

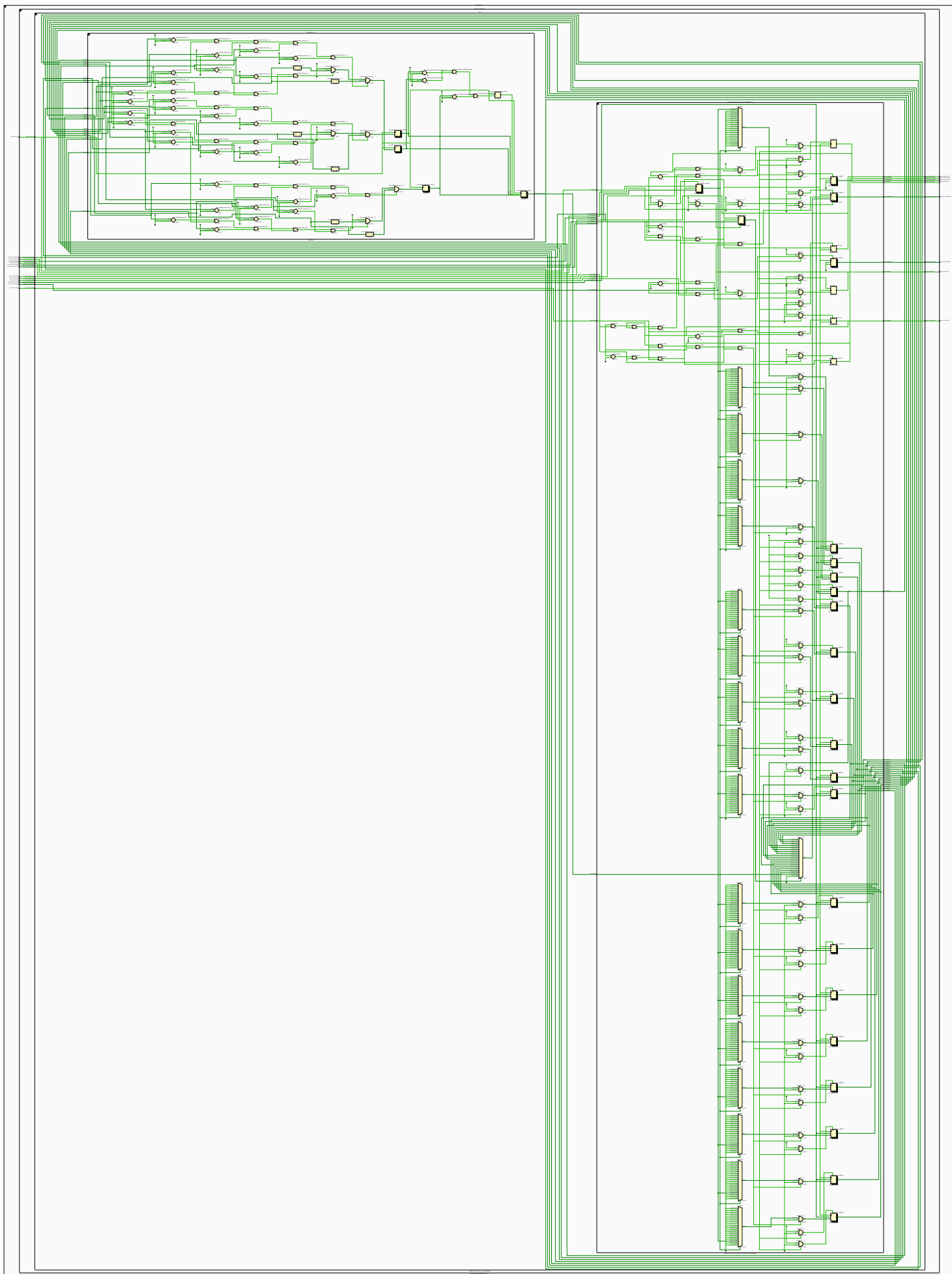
414     Else
415         GoTo LeSerial
416     End If
417 End Function
418
419 Sub SerialProcessa(o As Form, mstr As String)
420     Dim resp As String
421     Dim i As Integer
422     Dim dic(4) As String
423     dic(0) = "Aumentar TAP"
424     dic(1) = "Diminuir TAP"
425     dic(2) = "Atende as Normas Brasileiras"
426     dic(3) = "Não atende as Normas Brasileiras"
427     EscreveMessage o, "", False
428     If (mstr And 1) = 1 Then
429         o.Linha(0) = String(60, " ") & vbCrLf & String(60, " ") &
430         vbCrLf & "O alimentador atende as Normas Brasileiras"
431     Else
432         o.Linha(0) = Left("O alimentador não atende as Normas
433 Brasileiras" & String(60, " "), 60)
434         o.Linha(0) = o.Linha(0) & vbCrLf & Left("Recomenda-se
435 manobrar o transformador a montante para" & String(60, " "), 60)
436         o.Linha(0) = o.Linha(0) & vbCrLf & Left("melhorar a
437 Regulação" & String(60, " "), 60)
438     End If
439     For i = 1 To 11
440         o.Linha(i) = o.Linha(i).Tag
441     Next i
442     resp = RShiftLong(mstr, 5) And 3
443     o.Linha(12) = "TAP-A = " & dic(resp)
444     resp = RShiftLong(mstr, 3) And 3
445     o.Linha(13) = "TAP-B = " & dic(resp)
446     resp = RShiftLong(mstr, 1) And 3
447     o.Linha(14) = "TAP-C = " & dic(resp)
448 End Sub
449

```

D.10- HARDWARE RESULTANTE NO DISPOSITIVO FPGA (ESQUEMAS COMPLETOS)

Os seguintes dois esquemas foram criados automaticamente pela ferramenta de síntese do **Vivado Design Suite**, e representam o circuito digital gerado automaticamente a partir das descrições feitas na Linguagem VHDL. O primeiro esquema é mais simplificado do que o segundo.





Apêndice E

CÓDIGO EM ORCAD/PSPICE ORIGINAL, USADO PARA O CASO DE ESTUDO NAS SIMULAÇÕES EM VHDL-AMS PARA UM ALIMENTADOR EM MÉDIA TENSÃO A 14,4 KV

O seguinte código escrito em Orcad/Pspice foi derivado da aplicação de uma ferramenta de redução da rede e concentração de cargas a um alimentador de média tensão com 14,4 kV de tensão de linha nominal, considerando um cenário típico real de uma distribuidora nacional de energia elétrica. A partir desse código foi desenhado o circuito esquemático da figura 11.

Código Orcad/Pspice Original (.cir)

```

001  VA1 4376414ASE1 0 sin(0 11749.6110 60 0 0 {0.00})
002  VB1 4376414BSE1 0 sin(0 11719.8060 60 0 0 {-120.18})
003  VC1 4376414CSE1 0 sin(0 11821.3040 60 0 0 {120.68})
004  VA2 4376414ASE2 4376414ASE1 sin(0 130.0682 120 0 0 {21.43})
005  VB2 4376414BSE2 4376414BSE1 sin(0 48.7544 120 0 0 {162.25})
006  VC2 4376414CSE2 4376414CSE1 sin(0 100.4811 120 0 0 {156.46})
007  VA3 4376414ASE3 4376414ASE2 sin(0 110.4463 180 0 0 {62.54})
008  VB3 4376414BSE3 4376414BSE2 sin(0 124.9331 180 0 0 {-118.83})
009  VC3 4376414CSE3 4376414CSE2 sin(0 150.9581 180 0 0 {-142.97})
010  VA4 4376414ASE4 4376414ASE3 sin(0 138.9979 240 0 0 {134.88})
011  VB4 4376414BSE4 4376414BSE3 sin(0 29.8855 240 0 0 {-72.50})
012  VC4 4376414CSE4 4376414CSE3 sin(0 48.3491 240 0 0 {104.11})
013  VA5 4376414ASE5 4376414ASE4 sin(0 109.0364 300 0 0 {-110.72})
014  VB5 4376414BSE5 4376414BSE4 sin(0 197.8303 300 0 0 {40.65})
015  VC5 4376414CSE5 4376414CSE4 sin(0 238.7903 300 0 0 {157.84})
016  VA6 4376414ASE6 4376414ASE5 sin(0 41.1236 360 0 0 {-39.70})
017  VB6 4376414BSE6 4376414BSE5 sin(0 28.2447 360 0 0 {74.32})
018  VC6 4376414CSE6 4376414CSE5 sin(0 87.3594 360 0 0 {-72.04})
019  VA7 4376414ASE7 4376414ASE6 sin(0 186.1138 420 0 0 {-179.33})
020  VB7 4376414BSE7 4376414BSE6 sin(0 140.6377 420 0 0 {99.10})
021  VC7 4376414CSE7 4376414CSE6 sin(0 90.4330 420 0 0 {-94.11})
022  VA8 4376414ASE8 4376414ASE7 sin(0 59.9230 480 0 0 {-24.91})
023  VB8 4376414BSE8 4376414BSE7 sin(0 43.4805 480 0 0 {102.00})
024  VC8 4376414CSE8 4376414CSE7 sin(0 48.4673 480 0 0 {170.84})
025  VA9 4376414ASE9 4376414ASE8 sin(0 60.5105 540 0 0 {-18.83})
026  VB9 4376414BSE9 4376414BSE8 sin(0 52.6219 540 0 0 {145.02})
027  VC9 4376414CSE9 4376414CSE8 sin(0 88.0687 540 0 0 {-8.42})
028  VA10 4376414ASE10 4376414ASE9 sin(0 21.1493 600 0 0 {13.70})
029  VB10 4376414BSE10 4376414BSE9 sin(0 37.7378 600 0 0 {141.68})
030  VC10 4376414CSE10 4376414CSE9 sin(0 62.0618 600 0 0 {-178.13})
031  VA 4376414ASE 4376414ASE10 sin(0 92.5869 660 0 0 {93.73})
032  VB 4376414BSE 4376414BSE10 sin(0 19.6893 660 0 0 {28.79})

```

```

033 VC 4376414CSE 4376414CSE10 sin(0 92.0880 660 0 0 {21.97})
034 X9603201_960320 9603201A 9603201B 9603201C 960320A 960320B 960320C
035 LINHA_PRI
036 + Params: R=470.075700u, L=9.539426u,
037 X2047532_1429563 2047532A 2047532B 2047532C 1429563A 1429563B 1429563C
038 LINHA_PRI
039 + Params: R=20527.314667u, L=70.654308u,
040 X28676063_4782056 28676063A 28676063B 28676063C 4782056A 4782056B 4782056C
041 LINHA_PRI
042 + Params: R=696.458900u, L=10.353549u,
043 X489129_3193575 489129A 489129B 489129C 3193575A 3193575B 3193575C
044 LINHA_PRI
045 + Params: R=11473.875800u, L=39.492684u,
046 X1527172_1525345 1527172A 1527172B 1527172C 1525345A 1525345B 1525345C
047 LINHA_PRI
048 + Params: R=5241.102000u, L=106.359683u,
049 X1605338_907090 1605338A 1605338B 1605338C 907090A 907090B 907090C
050 LINHA_PRI
051 + Params: R=8539.230000u, L=44.854643u,
052 X486412_1003989 486412A 486412B 486412C 1003989A 1003989B 1003989C
053 LINHA_PRI
054 + Params: R=653.401800u, L=13.259732u,
055 X960320_1994738 960320A 960320B 960320C 1994738A 1994738B 1994738C
056 LINHA_PRI
057 + Params: R=1448.613600u, L=29.397269u,
058 X1994738_1994739 1994738A 1994738B 1994738C 1994739A 1994739B 1994739C
059 LINHA_PRI
060 + Params: R=720.052500u, L=14.612300u,
061 X2045998_2045999 2045998A 2045998B 2045998C 2045999A 2045999B 2045999C
062 LINHA_PRI
063 + Params: R=760.030900u, L=11.298610u,
064 X1477528_1477529 1477528A 1477528B 1477528C 1477529A 1477529B 1477529C
065 LINHA_PRI
066 + Params: R=2550.066400u, L=37.909255u,
067 X1329777_512944 1329777A 1329777B 1329777C 512944A 512944B 512944C
068 LINHA_PRI
069 + Params: R=4561.881000u, L=92.575992u,
070 X1003989_28760978 1003989A 1003989B 1003989C 28760978A 28760978B 28760978C
071 LINHA_PRI
072 + Params: R=4318.848000u, L=87.644031u,
073 X117180_960103 117180A 117180B 117180C 960103A 960103B 960103C LINHA_PRI
074 + Params: R=2778.009000u, L=56.375197u,
075 X296189_388608 296189A 296189B 296189C 388608A 388608B 388608C LINHA_PRI
076 + Params: R=1997.858400u, L=40.543303u,
077 X1994739_486412 1994739A 1994739B 1994739C 486412A 486412B 486412C
078 LINHA_PRI
079 + Params: R=5510.052000u, L=111.817588u,
080 X3166792_296189 3166792A 3166792B 3166792C 296189A 296189B 296189C
081 LINHA_PRI
082 + Params: R=3850.875000u, L=78.147276u,
083 X1743843_2894747 1743843A 1743843B 1743843C 2894747A 2894747B 2894747C
084 LINHA_PRI
085 + Params: R=10891.090000u, L=57.208431u,
086 X512943_1747204 512943A 512943B 512943C 1747204A 1747204B 1747204C
087 LINHA_PRI
088 + Params: R=12655.320000u, L=256.819239u,
089

```

```

090 X1329777_1228633 1329777A 1329777B 1329777C 1228633A 1228633B 1228633C
091 LINHA_PRI
092 + Params: R=5237.190000u, L=106.280296u,
093 X486412_486413 486412A 486412B 486412C 486413A 486413B 486413C LINHA_PRI
094 + Params: R=649.929900u, L=13.189276u,
095 X5173569_3166792 5173569A 5173569B 5173569C 3166792A 3166792B 3166792C
096 LINHA_PRI
097 + Params: R=1716.243300u, L=34.828380u,
098 X154347841_1790994 154347841A 154347841B 154347841C 1790994A 1790994B
099 1790994C LINHA_PRI
100 + Params: R=4612.722800u, L=24.229589u,
101 X539727_1605338 539727A 539727B 539727C 1605338A 1605338B 1605338C
102 LINHA_PRI
103 + Params: R=10763.606000u, L=56.538787u,
104 X2220655_28760882 2220655A 2220655B 2220655C 28760882A 28760882B 28760882C
105 LINHA_PRI
106 + Params: R=5328.633000u, L=108.135983u,
107 X28676197_588470 28676197A 28676197B 28676197C 588470A 588470B 588470C
108 LINHA_PRI
109 + Params: R=23789.218000u, L=81.881667u,
110 X960103_2220655 960103A 960103B 960103C 2220655A 2220655B 2220655C
111 LINHA_PRI
112 + Params: R=8988.501200u, L=47.214563u,
113 X2418037_28760998 2418037A 2418037B 2418037C 28760998A 28760998B 28760998C
114 LINHA_PRI
115 + Params: R=7342.858600u, L=38.570375u,
116 X46650023_296189 46650023A 46650023B 46650023C 296189A 296189B 296189C
117 LINHA_PRI
118 + Params: R=137.785400u, L=2.048316u,
119 X1001458_2045998 1001458A 1001458B 1001458C 2045998A 2045998B 2045998C
120 LINHA_PRI
121 + Params: R=5437.479000u, L=80.833494u,
122 X1429563_1605338 1429563A 1429563B 1429563C 1605338A 1605338B 1605338C
123 LINHA_PRI
124 + Params: R=2976.751400u, L=15.636202u,
125 X4781629_1743843 4781629A 4781629B 4781629C 1743843A 1743843B 1743843C
126 LINHA_PRI
127 + Params: R=11847.220000u, L=62.230766u,
128 X1973768_1973769 1973768A 1973768B 1973768C 1973769A 1973769B 1973769C
129 LINHA_PRI
130 + Params: R=217.457700u, L=3.232723u,
131 X4781628_28676045 4781628A 4781628B 4781628C 28676045A 28676045B 28676045C
132 LINHA_PRI
133 + Params: R=2274.490400u, L=11.947383u,
134 XSE_1973769 4376414ASE 4376414BSE 4376414CSE 1973769A 1973769B 1973769C
135 LINHA_PRI
136 + Params: R=6190.886800u, L=32.519327u,
137 X3193575_1178293 3193575A 3193575B 3193575C 1178293A 1178293B 1178293C
138 LINHA_PRI
139 + Params: R=11164.960800u, L=58.647013u,
140 X2045999_1527172 2045999A 2045999B 2045999C 1527172A 1527172B 1527172C
141 LINHA_PRI
142 + Params: R=930.615900u, L=18.885344u,
143 X2418037_1973768 2418037A 2418037B 2418037C 1973768A 1973768B 1973768C
144 LINHA_PRI
145 + Params: R=1337.757300u, L=27.147620u,
146

```

```

147 X1790994_2019891 1790994A 1790994B 1790994C 2019891A 2019891B 2019891C
148 LINHA_PRI
149 + Params: R=8335.533000u, L=123.915928u,
150 X1178293_1178294 1178293A 1178293B 1178293C 1178294A 1178294B 1178294C
151 LINHA_PRI
152 + Params: R=8980.148800u, L=47.170690u,
153 X907090_1743843 907090A 907090B 907090C 1743843A 1743843B 1743843C
154 LINHA_PRI
155 + Params: R=27233.220000u, L=143.049942u,
156 X489128_489129 489128A 489128B 489128C 489129A 489129B 489129C LINHA_PRI
157 + Params: R=5376.308000u, L=28.240529u,
158 X1624998_28760967 1624998A 1624998B 1624998C 28760967A 28760967B 28760967C
159 LINHA_PRI
160 + Params: R=6863.474800u, L=36.052280u,
161 X489128_1060058 489128A 489128B 489128C 1060058A 1060058B 1060058C
162 LINHA_PRI
163 + Params: R=1228.857000u, L=24.937664u,
164 X1001458_1477529 1001458A 1001458B 1001458C 1477529A 1477529B 1477529C
165 LINHA_PRI
166 + Params: R=5680.711000u, L=84.449378u,
167 X2919791_1329777 2919791A 2919791B 2919791C 1329777A 1329777B 1329777C
168 LINHA_PRI
169 + Params: R=17227.036000u, L=59.294863u,
170 X28760949_960319 28760949A 28760949B 28760949C 960319A 960319B 960319C
171 LINHA_PRI
172 + Params: R=700.101300u, L=14.207423u,
173 X2019891_1477528 2019891A 2019891B 2019891C 1477528A 1477528B 1477528C
174 LINHA_PRI
175 + Params: R=2891.627700u, L=42.986901u,
176 X512943_512944 512943A 512943B 512943C 512944A 512944B 512944C LINHA_PRI
177 + Params: R=1357.023900u, L=27.538604u,
178 X1178294_388608 1178294A 1178294B 1178294C 388608A 388608B 388608C
179 LINHA_PRI
180 + Params: R=2813.559300u, L=57.096633u,
181 X38673_1624998 38673A 38673B 38673C 1624998A 1624998B 1624998C LINHA_PRI
182 + Params: R=18222.031100u, L=40.896090u,
183 X1525339_117180 1525339A 1525339B 1525339C 117180A 117180B 117180C
184 LINHA_PRI
185 + Params: R=1705.143000u, L=34.603118u,
186 X28675721_2019128 28675721A 28675721B 28675721C 2019128A 2019128B 2019128C
187 LINHA_PRI
188 + Params: R=6022.739800u, L=31.636089u,
189 X2047531_28675987 2047531A 2047531B 2047531C 28675987A 28675987B 28675987C
190 LINHA_PRI
191 + Params: R=5057.171200u, L=17.406609u,
192 X1747204_28760949 1747204A 1747204B 1747204C 28760949A 28760949B 28760949C
193 LINHA_PRI
194 + Params: R=1u, L=0.01p,
195 X486413_28675721 486413A 486413B 486413C 28675721A 28675721B 28675721C
196 LINHA_PRI
197 + Params: R=1u, L=0.01p,
198 X28760978_38673 28760978A 28760978B 28760978C 38673A 38673B 38673C
199 LINHA_PRI
200 + Params: R=1u, L=0.01p,
201 X1477528_28676063 1477528A 1477528B 1477528C 28676063A 28676063B 28676063C
202 LINHA_PRI
203 + Params: R=1u, L=0.01p,

```

204 X2894747_154347841 2894747A 2894747B 2894747C 154347841A 154347841B
 205 154347841C LINHA_PRI
 206 + Params: R=1u, L=0.01p,
 207 X28760882_5173569 28760882A 28760882B 28760882C 5173569A 5173569B 5173569C
 208 LINHA_PRI
 209 + Params: R=1u, L=0.01p,
 210 X28676045_4781629 28676045A 28676045B 28676045C 4781629A 4781629B 4781629C
 211 LINHA_PRI
 212 + Params: R=1u, L=0.01p,
 213 X28675987_2047532 28675987A 28675987B 28675987C 2047532A 2047532B 2047532C
 214 LINHA_PRI
 215 + Params: R=1u, L=0.01p,
 216 X28760967_1060058 28760967A 28760967B 28760967C 1060058A 1060058B 1060058C
 217 LINHA_PRI
 218 + Params: R=1u, L=0.01p,
 219 X28760998_1228633 28760998A 28760998B 28760998C 1228633A 1228633B 1228633C
 220 LINHA_PRI
 221 + Params: R=1u, L=0.01p,
 222 X38673_28676197 38673A 38673B 38673C 28676197A 28676197B 28676197C
 223 LINHA_PRI
 224 + Params: R=1u, L=0.01p,
 225 X1477529_46650023 1477529A 1477529B 1477529C 46650023A 46650023B 46650023C
 226 LINHA_PRI
 227 + Params: R=1u, L=0.01p,
 228 XCLTPEqE588470 588470A 588470B 588470C CLTPEqE
 229 + Params: R=3776.109679, L=10218.814826m,
 230 XCLTPEqE2919791 2919791A 2919791B 2919791C CLTPEqE
 231 + Params: R=5421.448615, L=14671.390451m,
 232 XCLTPEqE1994739 1994739A 1994739B 1994739C CLTPEqE
 233 + Params: R=1745.668827, L=4724.085901m,
 234 XCLTPEqE2019128 2019128A 2019128B 2019128C CLTPEqE
 235 + Params: R=486.436782, L=833.460274m,
 236 XCLTPEqE512943 512943A 512943B 512943C CLTPEqE
 237 + Params: R=1750.206782, L=4736.366403m,
 238 XCLTPEqE4781628 4781628A 4781628B 4781628C CLTPEqE
 239 + Params: R=1386.068676, L=1208.460147m,
 240 XCLTPEqE1624998 1624998A 1624998B 1624998C CLTPEqE
 241 + Params: R=2471.137804, L=6687.332143m,
 242 XCLTPEqE539727 539727A 539727B 539727C CLTPEqE
 243 + Params: R=83.496136, L=148.835721m,
 244 XCLTPEqE1525339 1525339A 1525339B 1525339C CLTPEqE
 245 + Params: R=48.125186, L=102.416055m,
 246 XCLTPEqE2047531 2047531A 2047531B 2047531C CLTPEqE
 247 + Params: R=1809.370361, L=1399.855318m,
 248 XCLTPEqE1525345 1525345A 1525345B 1525345C CLTPEqE
 249 + Params: R=53.166707, L=139.877335m,
 250 XCCTPEqT2220655 2220655A 2220655B 2220655C CCTPEqT
 251 + Params: C=0.002786m, IC_A=11749.6110, IC_B=-5874.8055, IC_C=-5874.8055
 252 XCNTDqT4782056 4782056A 4782056B 4782056C CNTDqT
 253 + Params: DEF_V=0
 254 + MG1A=85.397, MG2A=0.225, MG3A=0.400, MG4A=0.344, MG5A=6.926, MG6A=0.396,
 255 MG7A=5.314,
 256 + MG8A=0.322, MG9A=0.308, MG10A=0.338, MG11A=1.169,
 257 + PH1A=-2.927, PH2A=-5.710, PH3A=162.347, PH4A=-40.684, PH5A=-68.678, PH6A=9.891,
 258 PH7A=46.822,
 259 + PH8A=-148.778, PH9A=8.903, PH10A=9.414, PH11A=-36.980,
 260

```

261 + MG1B=82.692, MG2B=0.499, MG3B=0.443, MG4B=0.681, MG5B=7.336, MG6B=0.407,
262 MG7B=5.432,
263 + MG8B=0.250, MG9B=0.505, MG10B=0.380, MG11B=1.392,
264 + PH1B=-125.950, PH2B=138.780, PH3B=-117.400, PH4B=-168.150, PH5B=58.400,
265 PH6B=157.210, PH7B=-91.160,
266 + PH8B=-115.640, PH9B=-87.730, PH10B=-49.780, PH11B=70.070,
267 + MG1C=80.211, MG2C=0.343, MG3C=0.646, MG4C=0.545, MG5C=6.366, MG6C=0.226,
268 MG7C=3.854,
269 + MG8C=0.549, MG9C=0.561, MG10C=0.624, MG11C=1.533,
270 + PH1C=117.260, PH2C=-63.600, PH3C=24.920, PH4C=41.860, PH5C=178.160, PH6C=-93.890,
271 PH7C=156.190,
272 + PH8C=45.630, PH9C=125.380, PH10C=157.920, PH11C=-156.750,
273 X960319_9603201 960319A 960319B 960319C 9603201A 9603201B 9603201C REGDF_3F
274 + Params: RD=1m, LD=1u, RM=100meg,
275 + TAPA=8, SIGA=-1, TAPB=8, SIGB=-1, TAPC=8, SIGC=-1
276 *.lib LINHAS.lib
277 .subckt LINHA_PRI 1A 1B 1C 2A 2B 2C
278 + Params: R=1, L=1m
279 RA 1A 1MA {R}
280 LA 1MA 2A {L}
281 RB 1B 1MB {R}
282 LB 1MB 2B {L}
283 RC 1C 1MC {R}
284 LC 1MC 2C {L}
285 .ends LINHA_PRI
286 *.lib CARGAS.lib
287 * Carga Indutiva Trifasica Primaria Equilibrada (Estrela) *****
288 .subckt CLTPEqE 1 2 3
289 + Params: R=1, L=1m
290 Rr 1 1a {R}
291 Lr 1a 4 {L}
292 Rs 2 2a {R}
293 Ls 2a 4 {L}
294 Rt 3 3a {R}
295 Lt 3a 4 {L}
296 .ends CLTPEqE
297 * Carga Capacitiva Trifasica Primaria Equilibrada (Triangulo) *****
298 .subckt CCTPEqT 1 2 3
299 + Params: R=10m, C=1m,
300 + IC_A=1m, IC_B=1m, IC_C=1m,
301 Rr 1 1a {R}
302 Cr 1a 2 {C} IC={IC_A}
303 Rs 2 2a {R}
304 Cs 2a 3 {C} IC={IC_B}
305 Rt 3 3a {R}
306 Ct 3a 1 {C} IC={IC_C}
307 .ends CCTPEqT
308 * Carga Nao-Linear Trifasica Desequilibrada (Triangulo) *****
309 .subckt CNTDqT 1 2 3
310 + Params:
311 + DEF_V=0
312 + MG1A=0, MG2A=0, MG3A=0, MG4A=0, MG5A=0, MG6A=0, MG7A=0, MG8A=0, MG9A=0,
313 MG10A=0, MG11A=0
314 + MG1B=0, MG2B=0, MG3B=0, MG4B=0, MG5B=0, MG6B=0, MG7B=0, MG8B=0, MG9B=0,
315 MG10B=0, MG11B=0
316 + MG1C=0, MG2C=0, MG3C=0, MG4C=0, MG5C=0, MG6C=0, MG7C=0, MG8C=0, MG9C=0,
317 MG10C=0, MG11C=0

```

```

318 + PH1A=0, PH2A=0, PH3A=0, PH4A=0, PH5A=0, PH6A=0, PH7A=0, PH8A=0, PH9A=0, PH10A=0,
319 PH11A=0
320 + PH1B=0, PH2B=0, PH3B=0, PH4B=0, PH5B=0, PH6B=0, PH7B=0, PH8B=0, PH9B=0, PH10B=0,
321 PH11B=0
322 + PH1C=0, PH2C=0, PH3C=0, PH4C=0, PH5C=0, PH6C=0, PH7C=0, PH8C=0, PH9C=0, PH10C=0,
323 PH11C=0
324 I1A 1 2 sin(0 {MG1A} 60 0 0 {DEF_V+PH1A})
325 I2A 1 2 sin(0 {MG2A} 120 0 0 {DEF_V+PH2A})
326 I3A 1 2 sin(0 {MG3A} 180 0 0 {DEF_V+PH3A})
327 I4A 1 2 sin(0 {MG4A} 240 0 0 {DEF_V+PH4A})
328 I5A 1 2 sin(0 {MG5A} 300 0 0 {DEF_V+PH5A})
329 I6A 1 2 sin(0 {MG6A} 360 0 0 {DEF_V+PH6A})
330 I7A 1 2 sin(0 {MG7A} 420 0 0 {DEF_V+PH7A})
331 I8A 1 2 sin(0 {MG8A} 480 0 0 {DEF_V+PH8A})
332 I9A 1 2 sin(0 {MG9A} 540 0 0 {DEF_V+PH9A})
333 I10A 1 2 sin(0 {MG10A} 600 0 0 {DEF_V+PH10A})
334 I11A 1 2 sin(0 {MG11A} 660 0 0 {DEF_V+PH11A})
335 I1B 2 3 sin(0 {MG1B} 60 0 0 {DEF_V+PH1B})
336 I2B 2 3 sin(0 {MG2B} 120 0 0 {DEF_V+PH2B})
337 I3B 2 3 sin(0 {MG3B} 180 0 0 {DEF_V+PH3B})
338 I4B 2 3 sin(0 {MG4B} 240 0 0 {DEF_V+PH4B})
339 I5B 2 3 sin(0 {MG5B} 300 0 0 {DEF_V+PH5B})
340 I6B 2 3 sin(0 {MG6B} 360 0 0 {DEF_V+PH6B})
341 I7B 2 3 sin(0 {MG7B} 420 0 0 {DEF_V+PH7B})
342 I8B 2 3 sin(0 {MG8B} 480 0 0 {DEF_V+PH8B})
343 I9B 2 3 sin(0 {MG9B} 540 0 0 {DEF_V+PH9B})
344 I10B 2 3 sin(0 {MG10B} 600 0 0 {DEF_V+PH10B})
345 I11B 2 3 sin(0 {MG11B} 660 0 0 {DEF_V+PH11B})
346 I1C 3 1 sin(0 {MG1C} 60 0 0 {DEF_V+PH1C})
347 I2C 3 1 sin(0 {MG2C} 120 0 0 {DEF_V+PH2C})
348 I3C 3 1 sin(0 {MG3C} 180 0 0 {DEF_V+PH3C})
349 I4C 3 1 sin(0 {MG4C} 240 0 0 {DEF_V+PH4C})
350 I5C 3 1 sin(0 {MG5C} 300 0 0 {DEF_V+PH5C})
351 I6C 3 1 sin(0 {MG6C} 360 0 0 {DEF_V+PH6C})
352 I7C 3 1 sin(0 {MG7C} 420 0 0 {DEF_V+PH7C})
353 I8C 3 1 sin(0 {MG8C} 480 0 0 {DEF_V+PH8C})
354 I9C 3 1 sin(0 {MG9C} 540 0 0 {DEF_V+PH9C})
355 I10C 3 1 sin(0 {MG10C} 600 0 0 {DEF_V+PH10C})
356 I11C 3 1 sin(0 {MG11C} 660 0 0 {DEF_V+PH11C})
357 .ends CNTDqT
358 .subckt REGDF_3F 1A 1B 1C 2A 2B 2C
359 * Primário: 1A-1B-1C
360 * Secundário: 2A-2B-2C
361 * Controles de TAP: 1TAPA 1TAPB 1TAPC
362 * Controles de Sinal Aditivo ou Subtrativo: 1SIGA 1SIGB 1SIGC
363 * Impedância de Disperção: RD+j*LD
364 * Impedância de Magnetização: Rm+j*Lm
365 + PARAMS: RD=1u, LD=1u, RM=100meg, TAPA=0, TAPB=0, TAPC=0, SIGA=1, SIGB=1, SIGC=1
366 * Fase A //////////////////////////////////////
367 RdA 1A 1rdA {RD}
368 LdA 1rdA 1LdA {LD}
369 RmA 1LdA 2B {RM}
370 VTAPA 1TAPA 0 {TAPA}
371 VSIGA 1SIGA 0 {SIGA}
372 E1FA 2A 1LdA Value = {V(1RegA)*V(1A)*V(1SIGA)}
373 EregA 1RegA 0 TABLE { V(1TAPA) } = (0,0) (1,0.00625) (2,0.0125) (3,0.01875)
374 (4,0.025) (5,0.03125)

```

```

375 + (6,0.0375) (7,0.04375) (8,0.05) (9,0.05625) (10,0.0625) (11,0.06875) (12,0.075) (13,0.08125)
376 + (14,0.0875) (15,0.09375) (16,0.1)
377 RTAPA 1TAPA 0 10k
378 RSIGA 1SIGA 0 10k
379 RregA 1RegA 0 10k
380 * Fase B //////////////////////////////////////
381 RdB 1B 1rdB {RD}
382 LdB 1rdB 1LdB {LD}
383 RmB 1LdB 2C {RM}
384 VTAPB 1TAPB 0 {TAPB}
385 VSIGB 1SIGB 0 {SIGB}
386 E1FB 2B 1LdB Value = {V(1RegB)*V(1B)*V(1SIGB)}
387 EregB 1RegB 0 TABLE { V(1TAPB) } = (0,0) (1,0.00625) (2,0.0125) (3,0.01875)
388 (4,0.025) (5,0.03125)
389 + (6,0.0375) (7,0.04375) (8,0.05) (9,0.05625) (10,0.0625) (11,0.06875) (12,0.075) (13,0.08125)
390 + (14,0.0875) (15,0.09375) (16,0.1)
391 RTAPB 1TAPB 0 10k
392 RSIGB 1SIGB 0 10k
393 RregB 1RegB 0 10k
394 * Fase C //////////////////////////////////////
395 RdC 1C 1rdC {RD}
396 LdC 1rdC 1LdC {LD}
397 RmC 1LdC 2A {RM}
398 VTAPC 1TAPC 0 {TAPC}
399 VSIGC 1SIGC 0 {SIGC}
400 E1FC 2C 1LdC Value = {V(1RegC)*V(1C)*V(1SIGC)}
401 EregC 1RegC 0 TABLE { V(1TAPC) } = (0,0) (1,0.00625) (2,0.0125) (3,0.01875)
402 (4,0.025) (5,0.03125)
403 + (6,0.0375) (7,0.04375) (8,0.05) (9,0.05625) (10,0.0625) (11,0.06875) (12,0.075) (13,0.08125)
404 + (14,0.0875) (15,0.09375) (16,0.1)
405 RTAPC 1TAPC 0 10k
406 RSIGC 1SIGC 0 10k
407 RregC 1RegC 0 10k
408 .ends REGDF_3F
409 .tran 98.03921569u 0.52458 0.49958 98.03921569u uic ; *ipsp*
410 .options itl5=0 reltol=0.100; *ipsp*
411 *.probe
412 .probe/csd I(VA) I(VB) I(VC) V(4376414ASE) V(4376414BSE) V(4376414CSE)
413 +(X9603201_960320.RA) I(X9603201_960320.RB) I(X9603201_960320.RC)
414 +(X2047532_1429563.RA) I(X2047532_1429563.RB) I(X2047532_1429563.RC)
415 +(X28676063_4782056.RA) I(X28676063_4782056.RB) I(X28676063_4782056.RC)
416 +(X489129_3193575.RA) I(X489129_3193575.RB) I(X489129_3193575.RC)
417 +(X1527172_1525345.RA) I(X1527172_1525345.RB) I(X1527172_1525345.RC)
418 +(X1605338_907090.RA) I(X1605338_907090.RB) I(X1605338_907090.RC)
419 +(X486412_1003989.RA) I(X486412_1003989.RB) I(X486412_1003989.RC)
420 +(X960320_1994738.RA) I(X960320_1994738.RB) I(X960320_1994738.RC)
421 +(X1994738_1994739.RA) I(X1994738_1994739.RB) I(X1994738_1994739.RC)
422 +(X2045998_2045999.RA) I(X2045998_2045999.RB) I(X2045998_2045999.RC)
423 +(X1477528_1477529.RA) I(X1477528_1477529.RB) I(X1477528_1477529.RC)
424 +(X1329777_512944.RA) I(X1329777_512944.RB) I(X1329777_512944.RC)
425 +(X1003989_28760978.RA) I(X1003989_28760978.RB) I(X1003989_28760978.RC)
426 +(X117180_960103.RA) I(X117180_960103.RB) I(X117180_960103.RC)
427 +(X296189_388608.RA) I(X296189_388608.RB) I(X296189_388608.RC)
428 +(X1994739_486412.RA) I(X1994739_486412.RB) I(X1994739_486412.RC)
429 +(X3166792_296189.RA) I(X3166792_296189.RB) I(X3166792_296189.RC)
430 +(X1743843_2894747.RA) I(X1743843_2894747.RB) I(X1743843_2894747.RC)
431 +(X512943_1747204.RA) I(X512943_1747204.RB) I(X512943_1747204.RC)

```

432 +(X1329777_1228633.RA) I(X1329777_1228633.RB) I(X1329777_1228633.RC)
 433 +(X486412_486413.RA) I(X486412_486413.RB) I(X486412_486413.RC)
 434 +(X5173569_3166792.RA) I(X5173569_3166792.RB) I(X5173569_3166792.RC)
 435 +(X154347841_1790994.RA) I(X154347841_1790994.RB) I(X154347841_1790994.RC)
 436 +(X539727_1605338.RA) I(X539727_1605338.RB) I(X539727_1605338.RC)
 437 +(X2220655_28760882.RA) I(X2220655_28760882.RB) I(X2220655_28760882.RC)
 438 +(X28676197_588470.RA) I(X28676197_588470.RB) I(X28676197_588470.RC)
 439 +(X960103_2220655.RA) I(X960103_2220655.RB) I(X960103_2220655.RC)
 440 +(X2418037_28760998.RA) I(X2418037_28760998.RB) I(X2418037_28760998.RC)
 441 +(X46650023_296189.RA) I(X46650023_296189.RB) I(X46650023_296189.RC)
 442 +(X1001458_2045998.RA) I(X1001458_2045998.RB) I(X1001458_2045998.RC)
 443 +(X1429563_1605338.RA) I(X1429563_1605338.RB) I(X1429563_1605338.RC)
 444 +(X4781629_1743843.RA) I(X4781629_1743843.RB) I(X4781629_1743843.RC)
 445 +(X1973768_1973769.RA) I(X1973768_1973769.RB) I(X1973768_1973769.RC)
 446 +(X4781628_28676045.RA) I(X4781628_28676045.RB) I(X4781628_28676045.RC)
 447 +(XSE_1973769.RA) I(XSE_1973769.RB) I(XSE_1973769.RC)
 448 +(X3193575_1178293.RA) I(X3193575_1178293.RB) I(X3193575_1178293.RC)
 449 +(X2045999_1527172.RA) I(X2045999_1527172.RB) I(X2045999_1527172.RC)
 450 +(X2418037_1973768.RA) I(X2418037_1973768.RB) I(X2418037_1973768.RC)
 451 +(X1790994_2019891.RA) I(X1790994_2019891.RB) I(X1790994_2019891.RC)
 452 +(X1178293_1178294.RA) I(X1178293_1178294.RB) I(X1178293_1178294.RC)
 453 +(X907090_1743843.RA) I(X907090_1743843.RB) I(X907090_1743843.RC)
 454 +(X489128_489129.RA) I(X489128_489129.RB) I(X489128_489129.RC)
 455 +(X1624998_28760967.RA) I(X1624998_28760967.RB) I(X1624998_28760967.RC)
 456 +(X489128_1060058.RA) I(X489128_1060058.RB) I(X489128_1060058.RC)
 457 +(X1001458_1477529.RA) I(X1001458_1477529.RB) I(X1001458_1477529.RC)
 458 +(X2919791_1329777.RA) I(X2919791_1329777.RB) I(X2919791_1329777.RC)
 459 +(X28760949_960319.RA) I(X28760949_960319.RB) I(X28760949_960319.RC)
 460 +(X2019891_1477528.RA) I(X2019891_1477528.RB) I(X2019891_1477528.RC)
 461 +(X512943_512944.RA) I(X512943_512944.RB) I(X512943_512944.RC)
 462 +(X1178294_388608.RA) I(X1178294_388608.RB) I(X1178294_388608.RC)
 463 +(X38673_1624998.RA) I(X38673_1624998.RB) I(X38673_1624998.RC)
 464 +(X1525339_117180.RA) I(X1525339_117180.RB) I(X1525339_117180.RC)
 465 +(X28675721_2019128.RA) I(X28675721_2019128.RB) I(X28675721_2019128.RC)
 466 +(X2047531_28675987.RA) I(X2047531_28675987.RB) I(X2047531_28675987.RC)
 467 +(X1747204_28760949.RA) I(X1747204_28760949.RB) I(X1747204_28760949.RC)
 468 +(X486413_28675721.RA) I(X486413_28675721.RB) I(X486413_28675721.RC)
 469 +(X28760978_38673.RA) I(X28760978_38673.RB) I(X28760978_38673.RC)
 470 +(X1477528_28676063.RA) I(X1477528_28676063.RB) I(X1477528_28676063.RC)
 471 +(X2894747_154347841.RA) I(X2894747_154347841.RB) I(X2894747_154347841.RC)
 472 +(X28760882_5173569.RA) I(X28760882_5173569.RB) I(X28760882_5173569.RC)
 473 +(X28676045_4781629.RA) I(X28676045_4781629.RB) I(X28676045_4781629.RC)
 474 +(X28675987_2047532.RA) I(X28675987_2047532.RB) I(X28675987_2047532.RC)
 475 +(X28760967_1060058.RA) I(X28760967_1060058.RB) I(X28760967_1060058.RC)
 476 +(X28760998_1228633.RA) I(X28760998_1228633.RB) I(X28760998_1228633.RC)
 477 +(X38673_28676197.RA) I(X38673_28676197.RB) I(X38673_28676197.RC)
 478 +(X1477529_46650023.RA) I(X1477529_46650023.RB) I(X1477529_46650023.RC)
 479 +V(9603201A) V(9603201B) V(9603201C)
 480 +V(388608A) V(388608B) V(388608C)
 481 +V(46650023A) V(46650023B) V(46650023C)
 482 +V(1329777A) V(1329777B) V(1329777C)
 483 +V(28676063A) V(28676063B) V(28676063C)
 484 +V(28760998A) V(28760998B) V(28760998C)
 485 +V(1477529A) V(1477529B) V(1477529C)
 486 +V(28760949A) V(28760949B) V(28760949C)
 487 +V(1429563A) V(1429563B) V(1429563C)
 488 +V(2894747A) V(2894747B) V(2894747C)

```

489 +V(489129A) V(489129B) V(489129C)
490 +V(1228633A) V(1228633B) V(1228633C)
491 +V(960103A) V(960103B) V(960103C)
492 +V(2047532A) V(2047532B) V(2047532C)
493 +V(1527172A) V(1527172B) V(1527172C)
494 +V(3193575A) V(3193575B) V(3193575C)
495 +V(512943A) V(512943B) V(512943C)
496 +V(1477528A) V(1477528B) V(1477528C)
497 +V(2418037A) V(2418037B) V(2418037C)
498 +V(960319A) V(960319B) V(960319C)
499 +V(2220655A) V(2220655B) V(2220655C)
500 +V(489128A) V(489128B) V(489128C)
501 +V(4781629A) V(4781629B) V(4781629C)
502 +V(1973768A) V(1973768B) V(1973768C)
503 +V(1178294A) V(1178294B) V(1178294C)
504 +V(486413A) V(486413B) V(486413C)
505 +V(486412A) V(486412B) V(486412C)
506 +V(1001458A) V(1001458B) V(1001458C)
507 +V(296189A) V(296189B) V(296189C)
508 +V(1790994A) V(1790994B) V(1790994C)
509 +V(1994739A) V(1994739B) V(1994739C)
510 +V(1624998A) V(1624998B) V(1624998C)
511 +V(1973769A) V(1973769B) V(1973769C)
512 +V(1994738A) V(1994738B) V(1994738C)
513 +V(3166792A) V(3166792B) V(3166792C)
514 +V(117180A) V(117180B) V(117180C)
515 +V(38673A) V(38673B) V(38673C)
516 +V(1060058A) V(1060058B) V(1060058C)
517 +V(512944A) V(512944B) V(512944C)
518 +V(907090A) V(907090B) V(907090C)
519 +V(28760967A) V(28760967B) V(28760967C)
520 +V(28760978A) V(28760978B) V(28760978C)
521 +V(28675721A) V(28675721B) V(28675721C)
522 +V(960320A) V(960320B) V(960320C)
523 +V(28676197A) V(28676197B) V(28676197C)
524 +V(1003989A) V(1003989B) V(1003989C)
525 +V(1605338A) V(1605338B) V(1605338C)
526 +V(1178293A) V(1178293B) V(1178293C)
527 +V(2045998A) V(2045998B) V(2045998C)
528 +V(2019891A) V(2019891B) V(2019891C)
529 +V(28676045A) V(28676045B) V(28676045C)
530 +V(28760882A) V(28760882B) V(28760882C)
531 +V(2045999A) V(2045999B) V(2045999C)
532 +V(1747204A) V(1747204B) V(1747204C)
533 +V(28675987A) V(28675987B) V(28675987C)
534 +V(1743843A) V(1743843B) V(1743843C)
535 +V(5173569A) V(5173569B) V(5173569C)
536 +V(154347841A) V(154347841B) V(154347841C)
537 +V(4782056A) V(4782056B) V(4782056C)
538 ; *ipsp*
539 .end

```