

SIMULAÇÃO DE TRANSITÓRIOS ELETROMAGNÉTICOS EM TEMPO REAL VIA FPGA

FILIFE RAFAEL FARIAS DE SÁ¹, ALEXANDRE CUNHA OLIVEIRA², WASHINGTON LUIZ ARAÚJO NEVES²

¹Mestrando no Programa de Pós-Graduação em Engenharia Elétrica, PPgEE/UFMG

²Departamento de Engenharia Elétrica, UFG

Rua Aprígio Veloso, 882 – Universitário, 58429-900, Campina Grande – PB, Brasil

E-mails: *filipe.sa@ee.ufcg.edu.br*, *aco@dee.ufcg.edu.br*, *waneves@dee.ufcg.edu.br*

Abstract— This paper presents the procedures for the development of a simulation platform for electromagnetic transient studies, implemented in FPGA. Some theoretical aspects are presented regarding the operation of these devices and the models used in the simulations. Then is presented the methodology for implementing the simulation environment in FPGA. In the end, some case studies involving transient analysis of lossless transmission lines simulated in FPGA are presented and they are compared with simulations performed in other platforms.

Keywords—FPGA, fixed point, floating point, simulations, transmission lines.

Resumo— Este trabalho apresenta os procedimentos para desenvolvimento de uma plataforma de simulação para estudos de transitórios eletromagnéticos, implementada em FPGA. São apresentados alguns aspectos teóricos referentes ao funcionamento destes dispositivos e aos modelos utilizados nas simulações. Na sequência é apresentada a metodologia de implementação do ambiente de simulação em FPGA. Por fim são apresentados alguns estudos de caso envolvendo análise de transitórios de linhas de transmissão sem perdas, simuladas em FPGA, e comparadas com simulações efetuadas em outras plataformas.

Palavras-chave—FPGA, linhas de transmissão, ponto fixo, ponto flutuante, simulações.

1 Introdução

Ao longo do último século, com o advento da computação digital, os estudos de transitórios eletromagnéticos em sistemas de potência ganharam ferramentas de análise gradativamente mais robustas que foram evoluindo com o passar do tempo. Os simuladores digitais de transitórios eletromagnéticos foram sendo otimizados de forma que apresentassem níveis cada vez melhores de exatidão dentro dos resultados fornecidos nas simulações. Esses simuladores, denominados simuladores do tipo EMTP (Electromagnetic Transients Program), são baseados nos modelos computacionais propostos em [1].

Embora sejam instrumentos de grande valia para estudos de transitórios, os simuladores do tipo EMTP são simuladores offline e não são ferramentas propriamente aplicáveis a determinadas situações em que a velocidade da obtenção dos resultados da simulação é relevante. Para isso são utilizados simuladores de tempo real, tendo estes simuladores hardwares de processamento que executam as rotinas de simulação com velocidades que permitam que o tempo de execução da simulação seja tão rápido quanto à resposta natural do sistema simulado, considerando ser um sistema real.

Durante certo período a procura por velocidades de processamento maiores significou restringir o tamanho mínimo do passo de cálculo utilizado nas simulações. Uma solução encontrada para esta restrição foi acoplar *Field Programmable Gate Arrays* (FPGAs) ao hardware de simuladores de tempo real, permitindo a execução de simulações com passos de cálculos menores, sem comprometer a velocidade de processamento requerida.

A escolha de dispositivos FPGA para este fim é feita devido aos recursos que este tipo de hardware

oferece para otimizar a velocidade de execução das simulações, especialmente a capacidade de processamento paralelo, onde em um único ciclo de clock o FPGA é capaz de executar múltiplas tarefas de forma simultânea.

Outra vantagem de um dispositivo FPGA é possuir um formato mais compacto em termos de espaço ocupado pelo hardware, facilitando a utilização de simuladores acoplados a sistemas reais. Diante disto propõe-se o desenvolvimento de estudos que buscam demonstrar a viabilidade de implementar simulações processadas integralmente em dispositivos FPGA, não apenas parte dela.

Embora apresente as vantagens descritas, dependendo dos recursos oferecidos pelo dispositivo FPGA utilizado, nem sempre é possível comportar todo o sistema a ser simulado em um único chip. Sendo assim, tem-se como objetivo final da pesquisa demonstrar a viabilidade de comunicação de múltiplos dispositivos para a execução de simulações onde o sistema é subdividido e simulado cada trecho em um dispositivo FPGA diferente. De forma mais específica, neste trabalho são apresentados os resultados de simulações de sistemas representados apenas por modelos clássicos de linhas de transmissão, como análise preliminar do método, com todos os trechos simulados ainda em um único dispositivo FPGA e os resultados parciais obtidos mediante simulações utilizando-se dois dispositivos FPGA comunicados entre si.

2 Estruturação do Ambiente de Simulação

2.1 Dispositivos FPGA

FPGAs são dispositivos eletrônicos programáveis e reconfiguráveis, fabricados a base de Silício, tendo

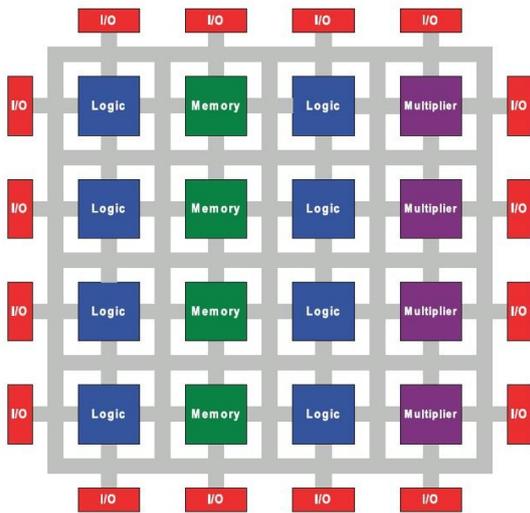


Figura 1. Estrutura de um dispositivo FPGA. [5]

neste elemento semiconductor a capacidade de constituir milhões de circuitos de lógica combinacional e sequencial. Estruturalmente a arquitetura de um FPGA é organizada em blocos, sendo estes: blocos de entrada e saída, blocos lógicos, blocos de multiplicadores e blocos de memória, conforme é mostrado na Figura 1.

Esses dispositivos possuem uma vasta gama de aplicações nas diversas áreas da engenharia devido as suas vantagens em relação a outros dispositivos eletrônicos programáveis, sendo escolhido para este trabalho devido às vantagens já citadas.

Em termos de velocidade de execução de tarefas, ela se torna maior em dispositivos FPGA em comparação com dispositivos de processamento de rotinas, pois por meio de um FPGA se torna possível projetar um hardware otimizado para realização de uma tarefa de processamento de dados específica. A arquitetura de um FPGA permite que operações aritméticas sejam processadas através de somadores e multiplicadores operando de forma simultânea o dado a ser calculado pela operação, propiciando a otimização no tempo de processamento em comparação com um processamento sequencial.

Para programar ou reconfigurar um FPGA para alguma aplicação específica utiliza-se uma linguagem de descrição de hardware. Este tipo de linguagem é utilizada para descrever o funcionamento e a conectividade do circuito a ser configurado, ou seja, modela o hardware para atender os requisitos específicos da aplicação que vai ser dada a ele.

Dentre as linguagens de descrição de hardware existentes selecionou-se a linguagem Verilog para a implementação proposta neste trabalho, pelo fato de possuir e um formato mais prático para descrição rápida de modelos e também uma estruturação mais próxima da linguagem C, utilizada em outra plataforma de simulação discutida na seção 3, cujos resultados são tomados como parâmetro de comparação.

A fim de realizar a modelagem de nível comportamental utilizando Verilog deve-se implementar descrições de hardware ao mais alto nível de abstração possível. Isto é obtido com a

criação de módulos que comunicam para o hardware as ligações lógicas necessárias para configurar cada aplicação desejada pelo usuário. Todos os módulos implementados podem ser instanciados em um módulo principal que faz a comunicação direta com o FPGA.

2.2 Aritmética de Ponto Fixo e de Ponto Flutuante

Processamentos de dados podem ser executados utilizando dois tipos de aritmética: a de ponto fixo e a de ponto flutuante. A aritmética de ponto flutuante trabalha com números com a possibilidade de representar partes fracionárias, ou seja, dependendo da quantidade de casas decimais do número é capaz de representá-lo em toda sua totalidade, sem arredondamentos ou truncamentos, desde que não seja uma dízima infinita ou um número fracionário com casas decimais de ordem excessivamente grande. Isto implica em maior exatidão nos cálculos, mas também maior consumo dos recursos disponíveis para o processamento dos dados.

Apesar das vantagens apresentadas do FPGA, os mesmos não contam com recursos nativos de ponto flutuante, havendo a necessidade da construção de um módulo dedicado para este fim. Diante disto trabalha-se utilizando aritmética de ponto fixo.

Na aritmética de ponto fixo, números são representados apenas por sua parte inteira sendo processados utilizando arredondamentos em algumas operações aritméticas como divisão, ou quaisquer outras que possam gerar números fracionários mesmo com dois operandos inteiros. Com isso percebe-se que inevitavelmente parte das informações é perdida nos cálculos.

Uma forma de minimizar essas perdas é converter o número a ser processado em um número inteiro utilizando um formato, número inteiro em potência de base 2 que vai fornecer a quantidade de bits com a qual o número será representado. Multiplica-se este formato pela razão entre o número a ser convertido e um valor base. Pretende-se que a perda de casas decimais ao final do processamento seja a mínima possível, comparando com o mesmo processamento em ponto flutuante. Matematicamente:

$$N_{PONTOFIXO} = N_{NORMALIZADO} \cdot 2^B. \quad (1)$$

2.3 Simuladores em FPGA

A maioria dos ambientes de simulação para sistemas de potência implementados em FPGA que se têm registros na literatura possuem características semelhantes na forma como exploram os recursos fornecidos por este dispositivo. Em [2]-[4] é proposta a criação de módulos para processar diferentes estágios sequenciais de um algoritmo padrão de solução, com a possibilidade de operações ocorrendo em paralelo nos módulos do simulador.

Basicamente os estágios sequenciais são: condicionamento de dados (formação da matriz admitância nodal e do vetor de correntes injetadas nos nós), cálculo das tensões nodais e atualização das fontes de corrente. Uma ressalva é feita no primeiro destes estágios, pois dependendo da aplicação a ser

dada se torna inviável, em termos de recursos disponíveis e eficiência, processar inversões de matrizes admitância nodal em FPGA durante a simulação dos modelos. Uma solução para isso é realizar essa parte da simulação em outra plataforma e transferir os dados obtidos para o FPGA utilizado.

A forma mais comum de implementar simulações em FPGA é descrevendo todo o sistema em um único dispositivo ou utilizando-se de modelos equivalentes caso este dispositivo não comporte todo o sistema representado de forma integral [4, 7]. Porém não havendo disponibilidade de um dispositivo que abarque um sistema de grande porte, mas tendo disponível mais de um dispositivo, é possível segmentar a simulação utilizando-se estes dispositivos disponíveis.

Em [8] é apresentada uma metodologia para este tipo de simulação segmentada, denominada pelo autor de decomposição funcional, onde cada dispositivo utilizado na simulação fica dedicado a executar exclusivamente equações referentes ao modelo de cada componente presente no sistema simulado. São analisados sistemas de larga escala com um deles sendo necessários 10 dispositivos FPGA para possibilitar a representação do sistema. Com isso são implementados modelos de todos os elementos elétricos presentes no sistema como máquinas, elementos não lineares e modelos de linha com perdas e dependentes da frequência. Utilizou-se aritmética de ponto flutuante.

Com relação ao desenvolvimento do atual trabalho aqui apresentado, objetiva-se também atingir a segmentação de sistemas, processando trechos de cada subdivisão de um sistema em um FPGA distinto, independentemente de cada uma dessas subdivisões conter elementos elétricos iguais ou não. Além disso, devido aos recursos disponíveis serem mais limitados, projeta-se a utilização de uma unidade de processamento externa para formação da matriz admitância nodal e cálculo da matriz inversa, sincronizado com o FPGA. A implementação é realizada em ponto fixo, conforme já mencionado.

2.4 Modelo Computacional para Linhas de Transmissão sem Perdas

Neste trabalho são utilizados modelos de linhas de transmissão sem perdas representando os sistemas elétricos simulados. A formulação matemática deste modelo parte dos princípios encontrados na Teoria das Ondas Viajantes. O tempo necessário para uma onda percorrer toda uma linha é definido pela constante τ , que é a razão entre o comprimento da linha e a velocidade da onda eletromagnética. Quando uma onda parte de uma das extremidades da linha em um tempo $t - \tau$, percorre toda a linha com uma velocidade constante, atingindo a outra extremidade em um tempo t . A expressão que relaciona as grandezas de corrente e tensão, de acordo com o modelo desenvolvido por [1] é:

$$i_{km}(t) = Y_c \cdot v_k(t) - [Y_c \cdot v_m(t - \tau) + i_{mk}(t - \tau)]. \quad (2)$$

Em que k e m são os pontos correspondentes às duas extremidades da linha.

Fontes de corrente fictícias são definidas em (3) e

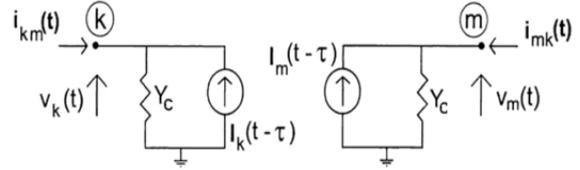


Figura 2. Modelo clássico de linhas de transmissão monofásicas sem perdas. [6]

(4). Estas fontes são dependentes de termos relacionados a tensões e correntes correspondentes a instantes de tempo passados e fazem parte do modelo computacional apresentado na Figura 2.

$$I_k(t - \tau) = Y_c \cdot v_m(t - \tau) + i_{mk}(t - \tau); \quad (3)$$

$$I_m(t - \tau) = Y_c \cdot v_k(t - \tau) + i_{km}(t - \tau). \quad (4)$$

Para linhas polifásicas, as equações de onda possuem a mesma forma das equações para linhas monofásicas, onde as grandezas unidimensionais das equações são substituídas por matrizes de ordem igual ao número de fases do sistema. Devido à presença de elementos de acoplamento mútuo nas matrizes de indutância e capacitância das linhas, a solução das equações se torna mais complexa que a mostrada para sistemas monofásicos. Esta dificuldade pode ser contornada utilizando-se o artifício das transformações modais, em que as matrizes do sistema são transformadas em matrizes associadas a modos desacoplados entre si, podendo cada equação desses modos ser resolvida de forma similar ao apresentado para linhas monofásicas:

$$[v_{fase}] = [S][V_{mod}]; \quad (5)$$

$$[i_{fase}] = [Q][J_{mod}]. \quad (6)$$

[S] e [Q] são as matrizes de transformação. Para este trabalho foi tomada a matriz de Karrenbauer como matriz de transformação.

$$[S] = [Q] = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \end{bmatrix}. \quad (7)$$

As correntes modais são calculadas de forma análoga ao cálculo para a corrente em um sistema monofásico para cada modo μ :

$$J_{km}^\mu(t) = Y_c^\mu \cdot V_k^\mu(t) - J_k^\mu(t - \tau); \quad (8)$$

$$J_{mk}^\mu(t) = Y_c^\mu \cdot V_m^\mu(t) - J_m^\mu(t - \tau). \quad (9)$$

Em que:

$$J_k^\mu(t - \tau) = Y_c^\mu \cdot V_m^\mu(t - \tau) + J_{mk}^\mu(t - \tau); \quad (10)$$

$$J_m^\mu(t - \tau) = Y_c^\mu \cdot V_k^\mu(t - \tau) + J_{km}^\mu(t - \tau). \quad (11)$$

Em termos de implementação computacional deve-se seguir os estágios sequenciais apresentados na seção anterior, com as tensões sendo calculadas pelo método da matriz admitância nodal para cálculo de tensões de redes. Como está se trabalhando em aritmética de ponto fixo deve-se atentar para a precedência das operações durante a conversão dos dados para a representação desejada, pois isto se mostra uma grande fonte de erro de cálculo quando executado de modo incorreto.

3 Implementação das Simulações

Para a implementação de um ambiente de simulação processado em FPGA foram utilizadas uma placa DE2 da Altera com chip Cyclone II e uma placa DE2-115 também da Altera com chip Cyclone IV. Por meio do Quartus II, software que realiza a sintetização dos circuitos a serem configurados no dispositivo, são construídos os módulos que permitem o processamento das equações que descrevem o modelo a ser simulado.

É criado um módulo que descreve o modelo de linhas monofásicas, processado no dispositivo contendo o chip Cyclone II. A intenção inicial era que quaisquer modelos pudessem ser processados neste dispositivo, porém o módulo criado para descrever modelos trifásicos apresentou problemas de limitação de recursos, conforme vai se aumentando o número de trechos e consequentemente o número de instâncias do módulo, comportando apenas uma instância. Desta forma optou-se por utilizar um dispositivo com maiores recursos, contendo um chip Cyclone IV para simulações envolvendo sistemas trifásicos.

Na descrição destes módulos, além de escrever as equações que representam o modelo, é importante atentar para o armazenamento das variáveis intermediárias dos cálculos, normalmente feita em registradores. Uma escolha equivocada do tamanho dos registradores, caso seja de um tamanho menor do que maior tamanho de um dado a ser armazenado pode implicar em estouro de armazenamento e consequentemente erro nos cálculos se propagando indefinidamente a partir deste erro. Ainda em termos de armazenamento, constatou-se que para definição de um número de ordem muito grande, para representar uma impedância correspondente a um circuito aberto (carga infinita) caso este número fosse definido no formato decimal, o Quartus não permitia a representação correta do dado. Assim percebeu-se que, obrigatoriamente, este dado teria que ser definido no formato binário.

Outro ponto importante sobre implementação é com relação ao módulo de linhas trifásicas. Devido à necessidade de inversão de matrizes em ponto fixo para o cálculo da matriz impedância do sistema, que consumiria uma quantidade considerável de recursos disponíveis, optou-se por processar o cálculo desta matriz em outro ambiente de simulação, para uma quantidade pré definida de valores de carga acoplada a cada barra, e, inicialmente, transferir e armazenar estes valores em um conjunto de registradores definido no módulo.

Além disso, ainda é necessária a criação de outros módulos, alguns deles servindo como ferramentas aos módulos de processamento das equações dos modelos e outros para condicionamento dos dados obtidos e transmissão destes. Assim, é construído um módulo de geração digital de sinal senoidal, utilizando look-up table com valores de seno representados em ponto fixo para cada ângulo inteiro em graus de 0 a 90 armazenados em um conjunto de registradores. Para os demais ângulos faz-se uso de propriedades trigonométricas no cálculo de seus respectivos senos. Outro módulo que é construído é o

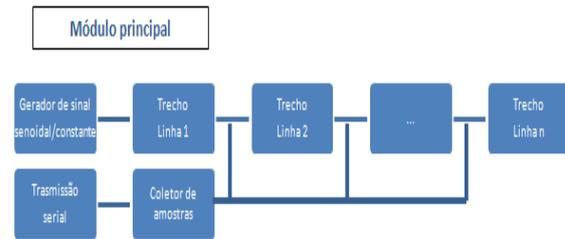


Figura 3. Diagrama esquemático do módulo principal.

responsável por selecionar entre a aplicação deste sinal senoidal ou um sinal constante a um dos terminais de uma linha, assim como a amplitude do sinal selecionado. Para seleção do passo de cálculo utilizado na simulação é construído outro módulo que gera um clock alternativo ao fornecido pelo FPGA, cuja frequência está relacionada com o passo de cálculo desejado, a cada subida deste clock.

Um módulo para transmissão serial de bits também é construído. Nele os dados são particionados em conjuntos de bytes e enviados para outro módulo que realiza as associações necessárias para comunicação com a porta de transmissão serial RS-232. Esta porta de transmissão leva os dados a um computador, aonde estes vão sendo impressos em tela enquanto o bit responsável por ativar a transmissão estiver em nível lógico alto. Os dados transmitidos são mostrados em formato hexadecimal e constituem um conjunto de 200 amostras correspondentes a 16.67 ms do sinal.

Todos estes módulos descritos são instanciados em um módulo principal, que também tem a função de realizar as associações das entradas e saídas definidas com os pinos do dispositivo FPGA. Um diagrama esquemático contendo as associações entre estes módulos descritos é apresentado na Figura 3.

Neste esquema da Figura 3 está apresentada a configuração para simulações utilizando apenas um dispositivo. Para simulações segmentadas em mais de um dispositivo, apresentadas parcialmente neste trabalho, é proposto o procedimento de partição visto no diagrama da Figura 4. No ponto do sistema em que ocorre a segmentação, há uma troca bidirecional de informações referente à transmissão de tensões e fontes de corrente fictícias entre os dispositivos. Esta

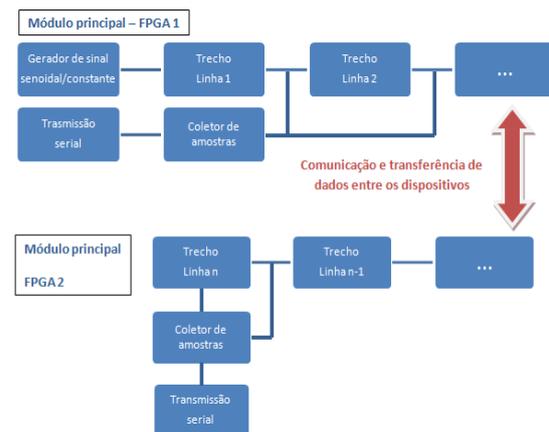


Figura 4. Diagrama esquemático da segmentação das simulações em dois dispositivos FPGA

troca de dados possibilita que dois ou mais módulos processados em dispositivos distintos operem da mesma forma como se pertencessem a um único dispositivo e assim se torne possível esta simulação segmentada.

Também são efetuadas simulações em outros ambientes que as processam de forma offline. As curvas obtidas nestas simulações offline servem de parâmetro de comparação para avaliar o nível de exatidão dos resultados simulados em FPGA. Para tal são efetuadas simulações considerando o sistema nas mesmas condições que as implementadas em FPGA, sendo simulações processadas em aritmética de ponto flutuante no software ATP® e utilizando DLLs programadas para operar em aritmética de ponto fixo por meio do software PSIM®. A DLL pode ser programada em C contendo equações que descrevem um modelo de um componente elétrico, fazendo abstração semelhante aos módulos em linguagem de descrição de hardware.

4 Cenários Avaliados e Resultados Obtidos

A fim de avaliar o que foi proposto, foi simulado e observado o comportamento de tensões e correntes em linhas de transmissão sem perdas representando sistemas elétricos em diferentes cenários variando-se as condições de carregamento e a fonte de tensão acoplada ao sistema, aplicados a testes nos modelos de linha monofásico e trifásico.

A escolha pela avaliação do modelo monofásico sem perdas acoplado a uma fonte de tensão constante, embora seja uma situação hipotética, porém explorada na literatura para fins didáticos [6, 9], foi feita objetivando analisar a representação em ponto fixo e os possíveis problemas de uma escolha inadequada. Foi avaliado um sistema contendo apenas uma linha em aberto representada por uma resistência de 1000 G Ω , conforme o diagrama unifilar construído no ATPDraw®, retratado na Figura 5. A linha foi definida com extensão de 300 km, possuindo como parâmetros: impedância característica de 500 Ω e 1 ms de tempo de trânsito. Foi definido um passo de cálculo de 1 μ s para as simulações.

Inicialmente foi selecionada uma quantidade de 16 bits para representar valores de tensão e corrente e percebeu-se que essa quantidade não seria suficiente para minimizar a perda de informações durante o processamento em ponto fixo, ocasionando um erro considerável comparado com as mesmas simulações processadas em ponto flutuante no ATP® com passo de cálculo de 0.1 μ s, cujo gráfico pode ser observado

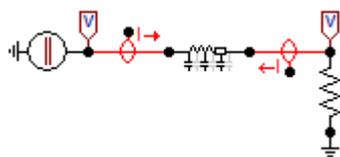


Figura 5. Diagrama unifilar de sistema monofásico representado por uma linha de transmissão em aberto.

na Figura 6. Aumentando-se a representação para 23 bits o erro se tornou desprezível, como pode ser visto na Figura 7. Um gráfico comparativo do erro entre as tensões para cada representação é apresentado na Figura 8.

A representação em 23 bits se mostrou consistente neste cenário de avaliação, ficando esta quantidade de bits definida para representar as tensões e correntes simuladas.

Para avaliação do comportamento de sistemas elétricos de alta tensão, representados pelo modelo trifásico de linhas apresentado, é simulado o sistema cujo diagrama unifilar pode ser visto na Figura 9, contendo duas linhas de transmissão em série, tendo uma carga acoplada representada por uma resistência de 2 k Ω em cada linha. As impedâncias de sequência positiva e zero são 290 Ω e 648 Ω , respectivamente, em ambas as linhas. Ambas também possuem os mesmos tempos de trânsito de sequência positiva e zero: 1 ms e 2 ms, respectivamente. O comprimento de cada linha é de 300 km.

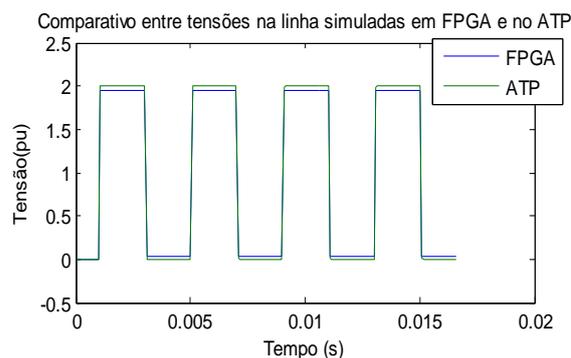


Figura 6. Tensão no terminal de um trecho de linha monofásica, com a simulação em FPGA representada em 16 bits.

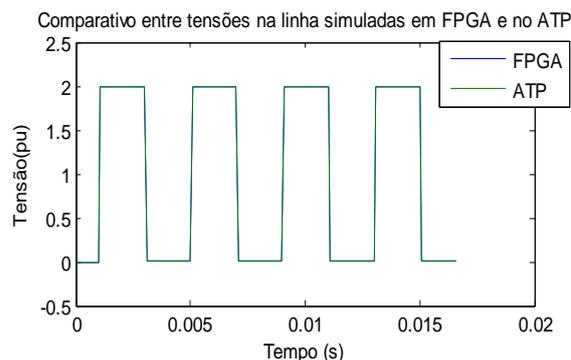


Figura 7. Tensão no terminal de um trecho de linha monofásica, com a simulação em FPGA representada em 23 bits.

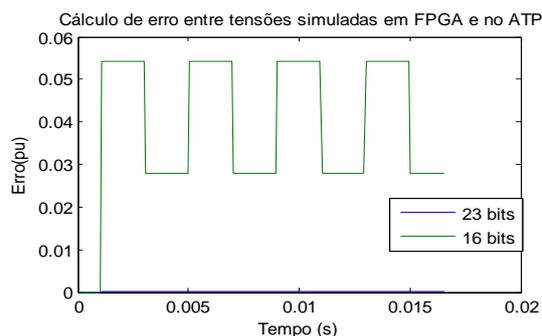


Figura 8. Erro entre as tensões representadas em 16 e em 23 bits.

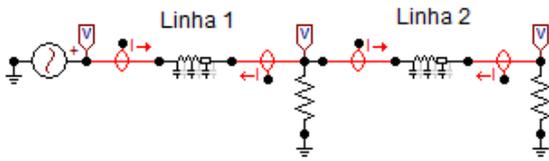


Figura 9. Diagrama unifilar de sistema trifásico representado por duas linhas de transmissão em carga.

Objetiva-se avaliar o comportamento transitório ao se inserir uma tensão de 500 kV ao sistema estando em carga, sendo a simulação processada em um único dispositivo com passo de cálculo de 1 μ s, conforme o diagrama da Figura 3, e comparam-se os resultados desta simulação com os obtidos no PSIM® e no ATP®. Para representação destas linhas trifásicas no domínio modal foram utilizadas as matrizes [S] e [Q] já apresentadas anteriormente. As curvas de tensão e corrente em uma das fases do sistema podem ser vistas na Figura 10 e na Figura 11.

Nos gráficos está representado o primeiro período do transitório da tensão no terminal receptor da linha 2 e a corrente que entra no terminal emissor da linha 2.

As curvas apresentam certo distanciamento de uma senóide pura devido a fenômenos de reflexão e refração das ondas ao atingir as extremidades das linhas, se estabelecendo ao atingir o regime permanente. Como cada linha simulada possui 300 km de extensão e isso gera um atraso da ordem de unidade de milissegundos, ocasionado pelo tempo de deslocamento da onda entre as extremidades da linha.

Estes transitórios de tensão avaliados possuem picos que chegam a atingir um acréscimo de mais da metade do valor máximo da tensão do sistema. Para minimizar estes problemas, utiliza-se alguma técnica

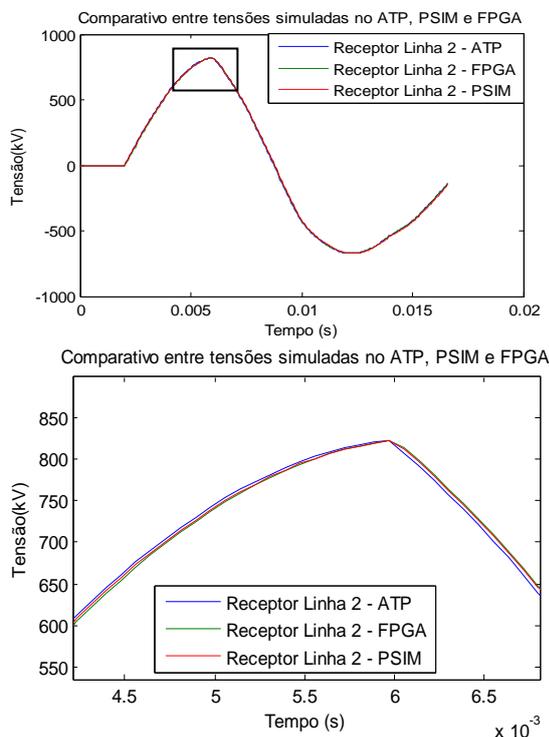


Figura 10. Curvas de tensão em uma das fases no terminal receptor da linha 2.

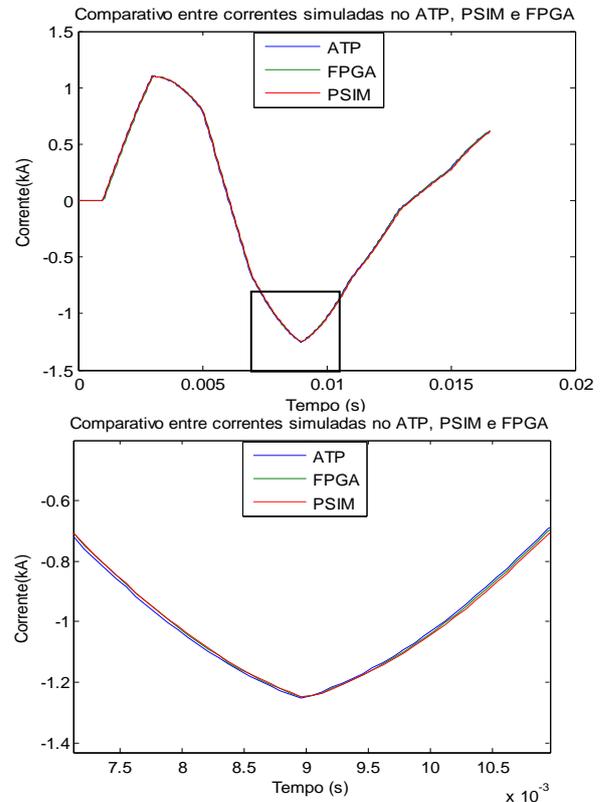


Figura 11. Curvas de corrente em uma das fases no terminal emissor da linha 2.

para redução de transitórios, dentre as mais comuns: elementos de pré inserção e chaveamento controlado [10,11]. Simulações em tempo real podem servir de auxílio à aplicação destas técnicas, como por exemplo, no tocante à escolha do instante ótimo de inserção/remoção do elemento de pré inserção ou abertura/fechamento de uma chave.

Além destas curvas apresentadas foram simuladas também as curvas de tensão no terminal receptor da Linha 1 e da corrente que entra no terminal emissor da Linha 1. Comparando estas simulações realizadas em FPGA com as realizadas no ATP®, em ponto flutuante, e no PSIM®, em ponto fixo, observa-se que os dados gerados em cada uma das plataformas formam curvas que praticamente se sobrepõem, sendo perceptíveis pequenas discrepâncias apenas com uma aproximação. Gráficos que relacionam o erro em pu, utilizando-se bases de 500 kV e 1 kA, entre ATP-FPGA e PSIM-FPGA para as grandezas simuladas são apresentados nas Figuras 12 a 15.

As curvas simuladas em FPGA apresentaram boa parte dos pontos com erros inferiores a 1%, considerando-se satisfatório. Espera-se nas próximas fases da pesquisa encontrar alternativas que possam minimizar os erros nos pontos que se mostraram relativamente maiores, sem que seja necessária uma mudança na representação de 23 bits utilizada.

A fim de se avaliar parcialmente o funcionamento de simulações implementadas utilizando-se dois dispositivos FPGA sincronizados foi selecionado um sistema contendo quatro linhas monofásicas em série, tendo cada uma os mesmos parâmetros da linha apresentada na Figura 6. O diagrama unifilar do sistema é apresentado na Figura 16. No terminal receptor de cada linha há uma carga acoplada, sendo

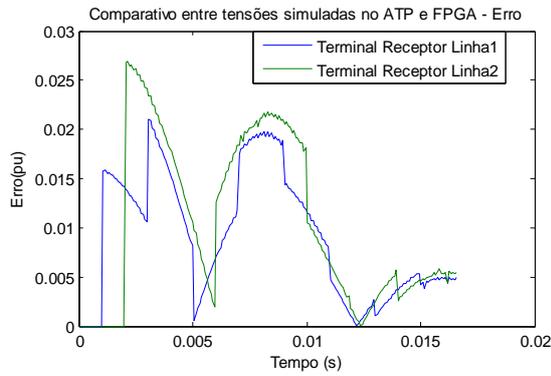


Figura 12. Erro entre ATP e FPGA no cálculo das tensões.

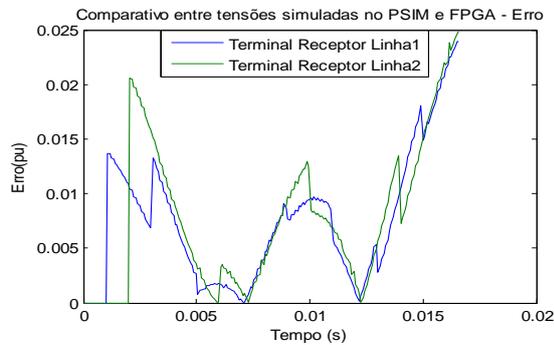


Figura 13. Erro entre PSIM e FPGA no cálculo das tensões.

representada por uma resistência de 2 k Ω nas linhas 1 e 3 e uma resistência de 1 k Ω nas linhas 2 e 4. O sistema foi dividido com as linhas 1 e 2 simuladas em um dispositivo e as linhas 3 e 4 no outro. Foi avaliado o transitório durante a energização por uma

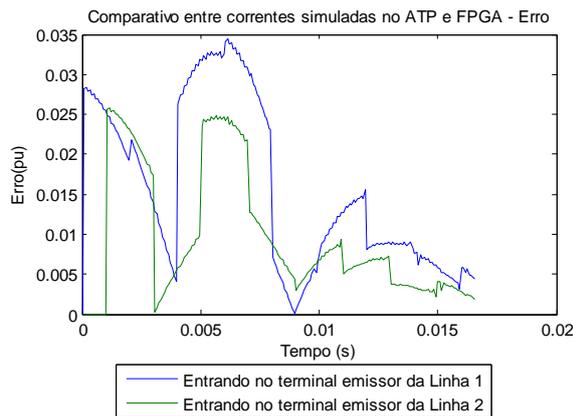


Figura 14. Erro entre ATP e FPGA no cálculo das correntes.

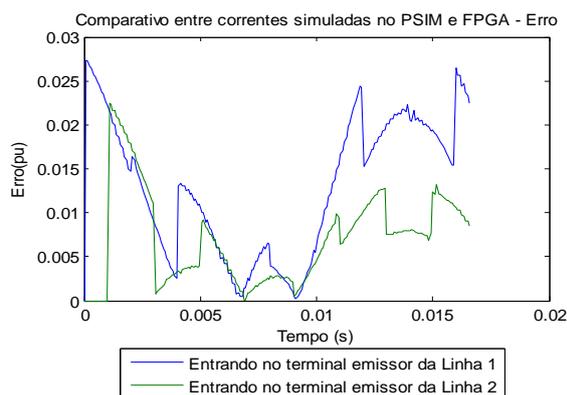


Figura 15. Erro entre PSIM e FPGA no cálculo das correntes.

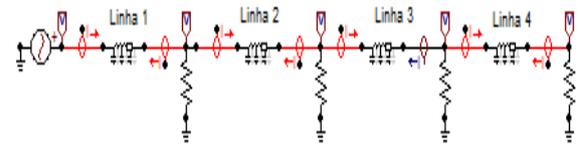


Figura 16. Diagrama unifilar de sistema monofásico representado por quatro linhas de transmissão em carga.

fonte de tensão senoidal. Para tal foram simuladas todas as tensões dos terminais receptores de cada linha, bem como cada corrente no sentido terminal emissor-receptor e terminal receptor-emissor.

Nas Figuras 17 e 18 são retratadas as tensões no terminal receptor das linhas 2 e 4, respectivamente, comparando-se as curvas simuladas em um único dispositivo com as curvas simuladas utilizando-se dois dispositivos.

Os pontos que foram amostrados para a formação das curvas apresentadas são, em sua grande maioria, iguais ou muito próximos nas duas simulações. Isto pode ser visualizado na observação das Figuras 17 e 18, onde as curvas praticamente se sobrepõem, e no gráfico de erros da Figura 19.

Neste gráfico de erros nota-se que há um ponto, na comparação entre as duas simulações para a tensão no terminal receptor da Linha 2, que apresentou um erro mais significativo que os demais. Isto pode ser explicado por um eventual atraso que ocorreu na transmissão e/ou na recepção de algum dos dados transferidos de um FPGA para o outro, dentro do instante de tempo onde é calculado o dado da tensão observada nesta análise.

Portanto, em termos de consistência dos resultados se faz necessária a observação de fatores relativos ao processo de comunicação e transferência de dados entre dispositivos, como eventuais atrasos e o qual isto pode influenciar na sincronização destes.

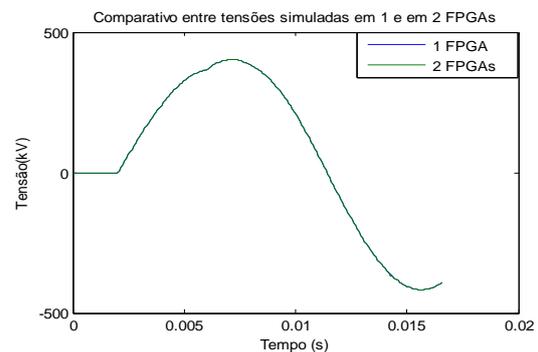


Figura 17. Curvas de tensão no terminal receptor da Linha 2.

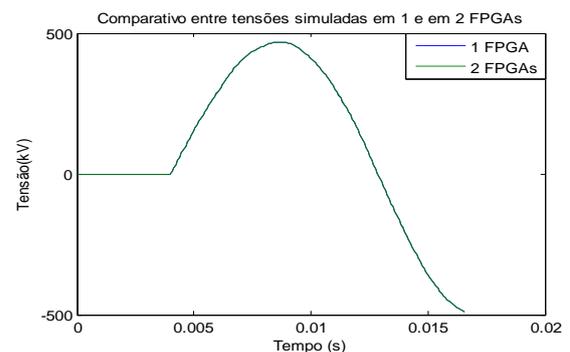


Figura 18. Curvas de tensão no terminal receptor da Linha 4.

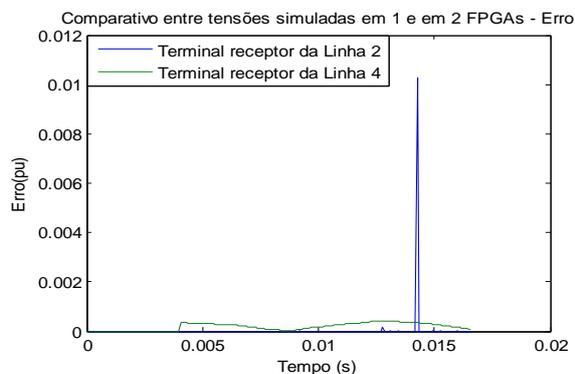


Figura 19. Erro no cálculo das tensões simuladas em 1 e em 2 dispositivos FPGA.

5 Conclusões

Em suma o desenvolvimento deste trabalho buscou apresentar motivações de se implementar simulações de transitórios em sistemas de potência processadas em tempo real integralmente em dispositivos FPGA e apresentar uma arquitetura de processamento para estas simulações. Durante o desenvolvimento do ambiente de simulação em FPGA, pôde-se constatar alguns detalhes relativos à organização estrutural dos módulos de simulação. Dentre eles vale-se ressaltar a importância da definição de registradores capazes de armazenar de forma apropriada variáveis intermediárias dos cálculos, além da escolha da representação em ponto fixo e sua influência na exatidão dos resultados calculados.

Para a representação adotada nos resultados expostos percebeu-se que os erros inerentes ao processo de cálculo realizado em ponto fixo se mostraram com uma tolerância relativamente razoável, ilustrados pelos gráficos apresentados. Assim constatou-se que é possível realizar simulações em tempo real desenvolvidas em FPGA implementadas em ponto fixo, e se obter níveis de exatidão semelhantes aos de simulações offline.

Na sequência serão finalizadas as atividades para implementação da sincronização de múltiplos dispositivos realizando uma mesma simulação de forma simultânea, a fim de se demonstrar uma alternativa viável para simulação de grandes sistemas podendo ser representados em toda sua totalidade e simulados em dispositivos conectados, na quantidade que for necessária/ disponível.

Também será realizada a conexão de uma unidade de processamento externa para cálculo da matriz impedância de sistemas simulados e transferência destes dados para utilização na simulação em FPGA. Realizando este procedimento em sincronismo com a simulação em tempo real, implica-se em mais memória disponível para o processamento das equações dos modelos simulados.

Agradecimentos

Nossos agradecimentos ao CNPq por todo aporte financeiro fornecido ao desenvolvimento desta

pesquisa, e aos revisores do CBA pela valiosa contribuição com seus comentários e sugestões sobre este trabalho.

Referências Bibliográficas

- [1] Dommel, H. W. "Digital Computer Solution of Electromagnetic Transients in Single and Multiphase Networks", IEEE Summer Power Meeting, Chicago - Ill, USA: 1968.
- [2] Y. Chen, V. Dinavahi, "FPGA-based real time EMTP". IEEE Transactions on Power Delivery, v. 24, n. 02, p. 892-901, Apr 2009, ISSN 0885-8977.
- [3] M. Matar, R. Iravani, "An FPGA-based real-time digital simulator for power electronic systems", International Conference on Power Systems Transients, Lyon, France, Jun 2007.
- [4] M. Matar, R. Iravani, "FPGA implementation of a modified two-layer network equivalent for real-time simulation of electromagnetic transients", International Conference on Power Systems Transients, Kyoto, Japan, Jun 2009.
- [5] I. Kuon, R. Tessier, J. Rose, "FPGA Architecture: Survey and Challenges", Foundation and Trends in Electronic Design Automation, Published by: Now Publishers Inc., v. 02, n. 02, p. 135-253, Hanover – MA, USA: 2008.
- [6] A. Araújo, W. Neves, "Cálculo de Transitórios Eletromagnéticos em Sistemas de Energia", Editora UFMG, Belo Horizonte – MG, Brasil: 2005.
- [7] M. Matar, M. Abdel-Rahman, A. Soliman, "Developing an FPGA coprocessor for real-time simulation of power systems", International Conference on Electrical, Electronic and Computer Engineering, Cairo, Egypt, Sep 2004, p. 791-794.
- [8] Y. Chen, V. Dinavahi, "Multi-FPGA digital hardware design for detailed large-scale real time electromagnetic transient simulation of power systems", IET Generation, Transmission & Distribution, v. 7, n. 05, p. 451-463, Jun 2013, ISSN 1751-8687.
- [9] L. Zanetta Jr., "Transitórios Eletromagnéticos em Sistemas de Potência", Edusp - Editora da Universidade de São Paulo, São Paulo – SP, Brasil: 2003.
- [10] J. Bickford, N. Mullineux, J. Reed, "Computation of Power-System Transients", Stevenage/UK: IEE Monograph Series 18, Peter Peregrinus Ltd., 1980.
- [11] K. Dantas, W. Neves, D. Fernandes Jr., G. Cardoso, L. Fonseca, "Chaveamento Controlado de Linhas de Transmissão: Uma Abordagem em Tempo Real via RTDS", Revista Controle e Automação, v. 22, n. 02, p.151-168, Março/Abril 2011.