IMPLEMENTAÇÃO DE PORTAS LÓGICAS NO ESTILO LÓGICO MUX-BASED VISANDO A CONSTRUÇÃO DE SISTEMAS DIGITAIS SEGUROS IMUNES A ATAQUES DPA

RENATO SOUZA[1]; RAFAEL SOARES[1]; PAULO BUTZEN[2]; FELIPE MARQUES[1]; LEOMAR DA ROSA JR[1]

[1]*Universidade Federal de Pelotas – {rsdsouza; rafael.soares; felipem; leomarjr}@inf.ufpel.edu.br*
[2]*Universidade Federal do Rio Grande – paulobutzen@furg.br*

1. INTRODUÇÃO

Os circuitos digitais estão cada vez mais presentes no dia-a-dia, causando um grande impacto na sociedade, devido ao fato de que estes se aplicam diretamente em diferentes áreas do conhecimento. Este aumento da presença dos circuitos digitais se deve ao enorme avanço das tecnologias de concepção de circuitos integrados, que permitem a integração de um número cada vez maior de componentes. Assim, temos a cada dia circuitos cada vez maiores e mais complexos (DA ROSA JUNIOR, 2008). Consequentemente, grandes dificuldades no desenvolvimento destes circuitos acabam sendo encontradas devido à adaptação de novos parâmetros da tecnologia. Como exemplo, pode-se citar o grande número de transistores embarcados em um único chip, que nos dias atuais está na casa de milhões de transistores, e o exíguo tempo de projeto e desenvolvimento dos circuitos digitais, os quais precisam ser lançados no mercado em um curto prazo para sua rápida comercialização. Devido a estes fatores, a automatização de um projeto, através do uso de ferramentas de auxílio com o intuito de facilitar e alcançar as metas torna-se cada vez mais indispensável. Neste contexto, as ferramentas de *CAD (Computer-Aided Design)* vêm contribuindo para que os desenvolvedores aumentem a eficiência e diminuam a complexidade em um projeto.

Outro ponto importante nos dias de hoje é a necessidade pela existência de sistemas digitais que garantam o sigilo de informações, seja em seu processamento ou no armazenamento de dados. Isso ocorre, visto que cada vez mais são comuns as atividades de compra pela Internet, transações bancárias, consultas de informações pessoais, sistemas de reserva de passagens, entre outros, que exigem sistemas computacionais operando em rede de acesso global, a qual requer uma transmissão protegida de dados confidenciais. Para que o projeto do circuito digital atenda as restrições de segurança, exige-se protocolos especiais de comunicação e também o uso de criptografia, ciência que se baseia na aritmética para ocultar dados (SOARES, 2010). Porém, a tecnologia *CMOS (Complementary Metal-Oxide-Semiconductor)* favorece, através do consumo de potência do circuito, a fuga de informações por *SCA (Side Channel Attacks)*. Kocher (KOCHER, 1999) mostrou que diferentes operações aritméticas possuem diferentes características de consumo de potência e que, através do uso de métodos estatísticos, é possível estabelecer uma ligação entre o consumo de potência e os dados processados em um sistema criptográfico. Com isso, é possível descobrir os dados que estão sendo processados. Esta avaliação é denominada de Análise Diferencial de Potência (*Differential Power Analysis* - DPA). Neste sentido, é possível notar a importância de ferramentas de CAD que possuem como ponto principal a geração de redes de transistores que forneçam um consumo de potência uniforme para serem utilizadas como alternativa para evitar a fuga de informações do circuito.

Portanto, o presente trabalho apresenta um método de geração de redes de transistores no estilo lógico Mux-Based para ser utilizado na implementação de porta lógicas, visando sua incorporação em ferramentas de CAD dedicadas a geração automatizada de circuitos digitais. Assim, será realizada uma investigação sobre a dissipação de potência das redes geradas, para saber se o método pode ser adotado como uma alternativa para evitar a fuga de informações de um circuito através da dissipação de potência.

2. MÉTODO DE GERAÇÃO DE REDES DE TRANSISTORES MUX-BASED

Existem diversos trabalhos na literatura sobre estilos lógicos de redes de transistores aplicados na tecnologia CMOS. Um desses é o Mux-Based, o qual é baseado na lógica realizada por um multiplexador (DA ROSA JUNIOR, 2008). Um multiplexador de entrada 2^n pode ser utilizado para implementar qualquer função lógica com n entradas (ERCEGOVAC, 2000). Para isso, as variáveis de entrada da função são utilizadas como variáveis de controle do multiplexador e os valores de saída da função são usados como entradas binárias para serem selecionadas pelas variáveis de controle. No entanto, é possível reduzir o número de entradas de um multiplexador. Além de constantes ‘0’ e ‘1’, uma das variáveis pode ser escolhida para ser ligada nas entradas do multiplexador.

Com base nessas informações, foi desenvolvido um método de geração automática de redes de transistores no estilo lógico Mux-Based. Para realizar a geração de redes, o método recebe como entrada uma função lógica. Após, realiza os seguintes procedimentos.

Primeiramente, uma das entradas da função é escolhida como variável de passagem. Assim as outras entradas da função acabam se tornando variáveis de controle. Porém, não existe uma regra para saber qual literal presente na função deve ser escolhido como variável de passagem. Portanto, o método realiza a geração da rede de transistores para cada configuração, onde cada literal da função será uma variável de passagem. Após esta escolha, são realizadas todas as combinações possíveis com as variáveis de controle. Estas combinações são realizadas da seguinte maneira: é gerado um conjunto com as variáveis de controle, em ambas as suas polaridades. É importante notar que as combinações que apresentarem uma variável em ambas as polaridades, são descartadas. Este procedimento é feito para realizar a seguinte análise: o método seleciona cada combinação gerada e realiza uma verificação de qual linha da tabela verdade da função de entrada essa combinação se encontra, com o intuito de verificar qual é a relação do valor da variável de passagem com o valor de saída da tabela.

Após realizar esta análise, o método proposto gera a estrutura da rede de transistores. Do ponto de vista elétrico, um multiplexador pode ser implementado usando *Three-state buffer* (HOROWITZ e HILL, 1989), também conhecido por *Tristates* (WESTE e HARRIS, 2005). Esse processo é bastante simples, uma vez que a estrutura gerada seja regular. Para realizar este processo, é necessário ligar as variáveis de controle do multiplexador nas variáveis de controle do *Three-state buffer* e as variáveis de entrada do multiplexador são ligadas na entrada do *Three-state buffer.*

Por fim, o método proposto escolhe a rede que contém uma estrutura regular, onde a quantidade de transistores em cada caminho série da rede é o mesmo (*stacks* de mesmo tamanho). Assim é gerado um arquivo texto que descreve o circuito definindo seus componentes e suas interligações no formato *SPICE*. Este arquivo é gerado para ser utilizado em ferramentas de simulações

elétricas. As redes geradas são validadas pelo método BRC descrito em (POSSANI, 2012).

3. ESTUDO DE CASO

Como estudo de caso, foram utilizados dois circuitos pequenos para realizar as simulações do ataque DPA. As etapas necessárias para realizar o ataque DPA são descritas em (MANGARD, 2007).

Em um primeiro momento, as portas lógicas que compõem cada circuito foram geradas pelo método proposto, Mux-Based. Após, cada circuito foi descrito em dois arquivos *SPICE* diferentes, um utilizando as redes Mux-Based geradas e o outro utilizando portas lógicas no estilo lógico CMOS tradicional. As simulações elétricas de cada circuito foram realizadas na ferramenta NGSpice (NGSpice). Os dois circuitos utilizados são descritos pelas Equações (1) e (2).

$$Circuito\ 1 = (m0\ XNOR\ k0)\ NAND\ (m1\ XNOR\ k1) \quad (1)$$

$$Circuito\ 2 = (m0\ XNOR\ k0)\ NOR\ (m1\ XNOR\ k1) \quad (2)$$

Foram realizadas quarto simulações de ataque DPA para cada circuito. Para cada simulação realizada, utilizou-se uma chave criptográfica diferente. A Tabela 1 apresenta os dados de quando o ataque acertou ou errou a chave criptográfica utilizada pelo sistema avaliado.

Tabela 1 – Resultados dos ataques nos dois circuitos utilizados.

	Circuito 1 (1)				**Circuito 2 (2)**			
	CMOS		Mux-Based		CMOS		Mux-Based	
	Integral	Pico	Integral	Pico	Integral	Pico	Integral	Pico
Primeiro Ataque	✗	✗	✗	✓	✓	✓	✓	✗
Segundo Ataque	✗	✗	✗	✗	✗	✗	✗	✗
Terceiro Ataque	✓	✗	✗	✗	✗	✗	✓	✓
Quarto Ataque	✗	✓	✗	✗	✓	✗	✗	✗

Verificando os dados presentes na Tabela 1 é possível notar que o circuito 1 utilizado na lógica Mux-Based mostrou-se mais seguro contra o ataque em relação ao circuito 1 utilizado na lógicas CMOS. Considerando a lógica Mux-Based, em apenas um caso foi possível descobrir a chave criptográfica utilizada, enquanto na lógica CMOS, a chave criptográfica foi descoberta em dois dos quatros testes. Já para o circuito 2 a lógica Mux-Based não apresentou ganhos em relação à lógica CMOS, obtendo, assim, o mesmo número de chaves descobertas.

4. CONCLUSÕES

Este trabalho apresentou um estudo sobre o método de geração de redes de transistores no estilo lógico Mux-Based visando à construção de sistemas digitais seguros imunes a ataques DPA. O método proposto parte de uma expressão Booleana, e gera a rede de transistores baseada na lógica de um multiplexador.

Visto que a tecnologia CMOS possibilita a fuga de informações, a lógica Mux-Based foi escolhida para ser utilizada como uma alternativa para evitar a fuga de informações através do consumo de potência. Redes de transistores no estilo lógico Mux-Based apresentam uma estrutura regular, onde a quantidade de transistores em cada caminho da rede é o mesmo. Portanto, pretendia-se, devido

a esta estrutura regular, que a rede implementada apresentasse um consumo de potência uniforme independente dos dados computados pelo circuito.

Os resultados demonstram que para o primeiro circuito utilizado a lógica Mux-Based apresentou um resultado melhor quando comparada a lógica CMOS. Porém, para o segundo circuito, o resultado do ataque foi o mesmo.

Para trabalhos futuros pretende-se utilizar um circuito real de um algoritmo de criptografia. Como exemplo, é possível utilizar o sub-módulo SBox do algoritmo de criptografia DES, o qual é composto por 175 portas lógicas. Dentre elas estão: inversores, *AND*, *NAND*, *OR*, *NOR*, *XOR* e *XNOR* de duas entradas. Assim, será gerada a rede de transistores do sub-módulo SBox no estilo Mux-Based e será realizado o ataque para verificar a segurança do circuito.

5. REFERÊNCIAS BIBLIOGRÁFICAS

DA ROSA JUNIOR, L. **Automatic Generation and Evaluation of Transistor Networks in Different Logic Styles**. Tese (Programa de Pós-Graduação em Microeletrônica) - Instituto de informática, UFRGS. Porto Alegre. 2008. p. 147 f.

ERCEGOVAC, M.; LANG, T.; MORENO, J. H. **Introdução aos Sistemas Digitais**. Porto Alegre: Bookman, 2000.

HOROWITZ, P.; HILL, W. **The Art of Electronics**. 2nd Edition. Ed. Cambridge University Press, 1989. 487–490 p.

KOCHER, P.; JAFFE, J.; JUN, B. **Differential Power Analysis**. 19th International Cryptology Conference on Advances in Cryptology. 1999. p. 388-397.

MANGARD, S.; OSWALD, E.; POPP, T. **Power Analysis Attacks:** Revealing the Secrets of Smart Cards. 2007.

NGSPICE. Disponível em: <http://ngspice.sourceforge.net/>. Acesso em: Julho 2014.

POSSANI, V. N.; SOUZA, R. S.; DOMINGUES JÚNIOR, J. S.; MARQUES, F. S.; DA ROSA JUNIOR, L. S. **Boolean Representation Code - An Efficient Method to Represent Boolean Functions**. In: 12th Microelectronics Students Forum, 2012, Brasília. Proceedings SForum, 2012.

SOARES, R. L. **Arquitetura GALS pipeline para criptografia robusta a ataques DPA e DEMA**. Faculdade de Informática, Pontifícia Universidade do Rio Grande do Sul. Porto Alegre, p. 145 f. 2010.

WESTE, N.; HARRIS, D. **CMOS VLSI Design:** A Circuits and Systems Perspective. 3nd Edition. 2005.