

DESENVOLVIMENTO DE UMA FERRAMENTA PARA AVALIAR O COMPORTAMENTO DE PORTAS LÓGICAS CMOS NA PRESENÇA DE FALHAS *STUCK-OPEN*

Alexandra L. Zimpeck

Acadêmica do curso de Engenharia de Computação – FURG
alexandra.zimpeck@furg.br

Msc. Cristina Meinhardt, Dr. Paulo F. Butzen

Professores/Pesquisadores do Centro de Ciências Computacionais – FURG
{cristinameinhardt, paulobutzen}@furg.br

Resumo. *O comportamento clássico das falhas Stuck-Open (SOFs) foi afetado devido às altas correntes de fuga presentes em circuitos nanométricos. Em função disso, ferramentas que auxiliem o projeto de circuitos integrados e que identifiquem sua robustez quanto a este tipo de falha são indispensáveis nos dias de hoje. Este trabalho apresenta uma ferramenta que avalia o comportamento e a cobertura de falhas em portas lógicas CMOS com SOFs.*

Palavras-chave: *Circuitos integrados. Tecnologias nanométricas. Falhas Stuck-Open.*

1. INTRODUÇÃO

Os avanços tecnológicos em circuitos integrados tem como foco principal a diminuição das dimensões dos transistores. Com essa redução, é possível inserir mais transistores em uma mesma área, ampliando as funcionalidades e o desempenho dos circuitos. Contudo, essa redução traz também consequências indesejáveis, como efeitos de envelhecimento, variabilidade, correntes de fuga e um aumento no número de possíveis falhas (ANGHEL et al., 2007). As falhas podem ser consideradas uma operação incorreta no sistema, que pode ter sido originada de erros de projeto, defeitos de fabricação ou interferência externa (FRANCO, 2008).

Neste trabalho iremos focar em um tipo de falha que ocorre em transistores MOS, a

qual voltou a ter importância atualmente devido às altas correntes de fuga verificadas em tecnologias nanométricas. Ela é conhecida como Falha *Stuck-Open*. Esta falha se caracteriza por fazer com que o transistor opere sempre como uma chave aberta, independente da tensão aplicada ao seu terminal de controle, o *gate*.

Neste contexto, é importante o uso de ferramentas que auxiliem o projeto (*Computer-aid-design* - CAD) de portas lógicas e avaliem o comportamento de circuitos em tecnologias nanométricas.

Este trabalho apresenta uma ferramenta que permite a simulação de SOFs em portas lógicas CMOS. Através dos pares de vetores de teste utilizados em cada simulação, é possível calcular a porcentagem de falhas detectadas durante a aplicação do teste.

Este artigo está organizado como segue: A seção 2 apresenta o funcionamento das SOFs e como as correntes de fuga influenciam as tecnologias nanométricas. A seção 3 apresenta a descrição da ferramenta proposta. Por fim, a seção 4 apresenta as conclusões.

2. FALHAS *STUCK-OPEN*

Os transistores MOS podem ser vistos como dispositivos de três terminais (*source*, *gate* e *drain*) que se comportam como chaves lógicas. O valor lógico aplicado ao terminal de controle (*gate*) indica se existe conexão ou não entre os outros dois terminais (MATTOS et al., 2009).

Se uma SOF ocorrer em um transistor do sistema, isto significa que a conexão entre os dois terminais nunca irá existir, independente do sinal aplicado em *gate*.

Para exemplificar o efeito das SOFs, a Fig. 1 mostra a porta lógica NOR2 e sua tabela verdade correta. Também são mostradas as saídas com uma falha ocorrendo no transistor B da rede *Pull-Down* da porta lógica NOR2.

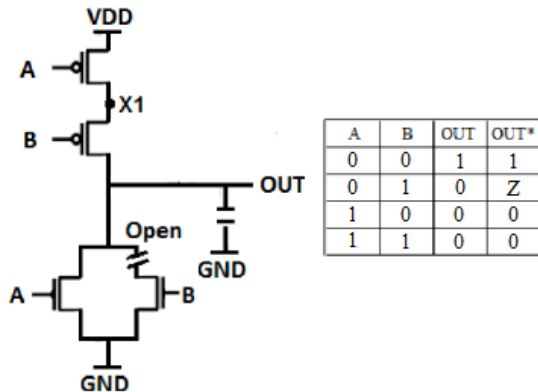


Figura 1. Falha *Stuck-Open* na Porta NOR2

Analisando alguns estados da tabela verdade, podemos chegar a algumas conclusões. Para o vetor $AB=00$, ambos os transistores PMOS estão ligados, levando a saída a um valor lógico correto. No vetor $AB=01$, o transistor “bom” na rede *pull-down* está desligado e o transistor com falha está ligado, mas não pode fornecer corrente para a saída por causa da SOF. Sendo assim, ambos os caminhos da rede *pull-down* ficam bloqueados, fazendo a saída flutuar em um estado de alta impedância (Z).

Quando isto acontece, o valor de tensão do estado anterior fica armazenado na capacitância de carga. Por isso, falhas *Stuck-Open* são difíceis de testar, pois sempre que vamos analisá-las é necessário um par específico de vetores considerando o estado anterior e o estado atual.

Estas propriedades elétricas clássicas das falhas *Stuck-Open* devem ser reexaminadas em tecnologias atuais, pois os transistores em escala de nanômetros apresentam correntes de fuga elevadas. Este comportamento se torna mais crítico durante

o estado de alta impedância causado pela SOF (GOMEZ et al., 2009).

Na Fig. 2, é mostrada a influência das correntes de fuga com a redução da tecnologia. O experimento foi realizado considerando o par de vetores $AB=00/AB=10$ e as tecnologias preditivas de 16nm, 22nm, 32nm e 45nm (CAO, 2013).

Podemos perceber que o tempo em que a saída é mantida no estado anterior vai diminuindo a cada nova tecnologia. Isto acontece porque quanto mais reduzimos as dimensões dos transistores, maiores serão as correntes de fuga presentes nos circuitos. Porém, para verificarmos a atuação do circuito com a SOF, precisamos de um período de tempo suficiente para a análise, a qual se torna cada vez mais crítica.

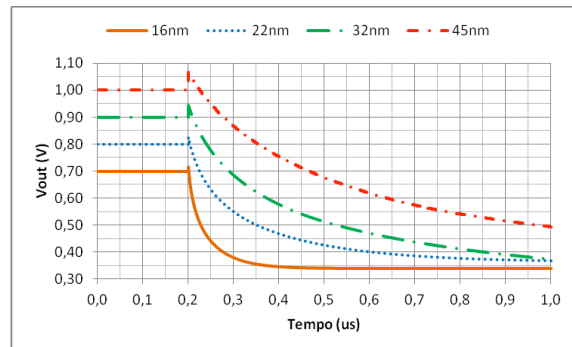


Figura 2. Resultado obtido com a porta lógica NOR2 com SOF

3. FERRAMENTA DESENVOLVIDA

Com a redução do dimensionamento dos transistores, os circuitos se tornam mais propensos a falhas. Junto com esta tendência, as altas correntes de fuga também influenciam no comportamento dos circuitos na presença das SOFs. Sendo assim, este trabalho apresenta o desenvolvimento de uma ferramenta com o propósito de auxiliar o projeto de circuitos em tecnologias nanométricas. Como resultados, são mostradas as saídas com falha que foram propagadas ou mantidas erradas por apenas um curto período de tempo, e também a cobertura de falhas.

A ferramenta foi desenvolvida na linguagem de programação JAVA devido a

sua portabilidade e independência de plataforma.

Todas as simulações elétricas necessárias são realizadas pelo simulador elétrico NGSpice (NGSPICE, 2013). Estas simulações vão ocorrer em duas etapas do processamento. A primeira ocorrerá no momento da validação da função lógica e a segunda quando a falha é inserida no circuito.

A ferramenta é dividida em seis módulos principais de acordo com o diagrama de blocos da Fig. 3. Ao total, a ferramenta apresenta seis classes e um interface que implementa os tipos de falhas. As simulações de falha duram entre dez e sessenta segundos. O tempo de duração depende do número de pares de vetores de teste que serão utilizados. Cada um dos módulos terão suas funções específicas descritas a seguir:

Interface com o usuário. A interface gráfica (Fig. 4) é o meio de comunicação com o usuário do programa. No lado esquerdo da Fig. 4 estão os campos de entrada da ferramenta, que são: o número de entradas e saídas do circuito, a tecnologia, o valor da tensão, o tipo de falha a ser inserida e a descrição do circuito em linguagem SPICE. À direita, os resultados do programa são exibidos nos campos: resultado esperado, vetores de teste e resultado obtido.

A descrição SPICE pode ser digitada manualmente ou inserida através da busca de um arquivo pelo computador. Caso o usuário decida por inserir um arquivo já existente, basta clicar no botão “Adicionar Descrição do Circuito” para acessar os diretórios.

A ferramenta oferece todo o suporte para simular circuitos com tecnologias preditivas de 16nm, 22nm, 32nm e 45nm. Se o usuário desejar inserir outra tecnologia para simulação, basta selecionar o item “Outra tecnologia”.

O botão “Limpar tudo” é o responsável por apagar todos os dados da tela e cancelar a simulação caso ela já tenha sido feita.

Validação dos dados. Antes de simular a falha é necessário fazer uma validação dos dados inseridos.

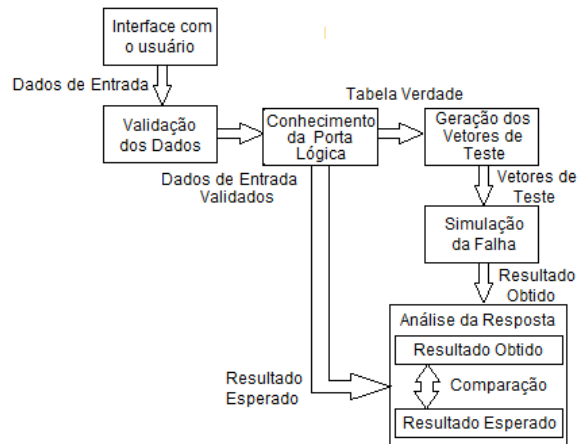


Figura 3. Diagrama de Blocos do Simulador de Falhas *Stuck-Open*



Figura 4. Interface Gráfica da Ferramenta

Se algum dos campos estiver no formato incorreto ou sem preenchimento, será mostrada uma mensagem de erro até que o usuário entre com novos dados que passem na etapa de validação. Quando todos os campos estiverem validados, a ferramenta identifica os transistores da porta lógica descrita na linguagem SPICE. Assim, aparecerão caixas de seleção com os transistores para o usuário escolher em qual será inserida a falha.

Conhecimento da porta lógica. Após a validação, a ferramenta irá juntar todos os dados concedidos na interface gráfica e transformá-los em um arquivo compatível com o Simulador NGSpice, considerando o circuito sem a falha. Serão avaliados todos os estados de entrada, gerando a tabela verdade correta para a descrição do circuito.

Geração dos vetores de teste. Para realizar um teste é necessário pelo menos um par de vetores. O número total de vetores de teste possíveis, A_S , para uma porta lógica de E entradas é dado pela Eq. (1), onde $n = 2^{(E)}$ e $p=2$, como utilizamos sempre um par de vetores.

$$A_S(n, p) = \frac{n!}{(n-p)!} \quad (1)$$

Para circuitos com o número de entradas menores ou iguais a três, será realizado um teste exaustivo, ou seja, serão geradas todas as combinações possíveis. Já para circuitos com total de entradas maior do que três, será realizado um teste aleatório, onde o usuário irá escolher quantos vetores de teste serão utilizados.

Simulação da SOF. Realiza a inserção de falha no circuito, ou seja, modifica o arquivo já existente de modo a fazer com que o transistor tenha um comportamento anormal. Falhas SOFs caracterizam-se por manter o transistor afetado sempre aberto, ou seja, sem passagem de corrente. Para a rede *pull-up*, inserir a SOF consiste em manter o terminal *gate* do transistor PMOS sempre com o valor lógico '1', enquanto que para a rede *pull-down*, formada por transistores NMOS, o terminal *gate* deve ser mantido no valor lógico '0'. Após essa alteração, o arquivo obtido será submetido a uma nova simulação elétrica.

Análise da resposta. A ferramenta compara o resultado obtido de cada par de vetores de teste com o resultado esperado. Também é feita uma contagem para saber quantos resultados deram o mesmo valor e quantos

foram diferentes, calculando a robustez do circuito a falhas do tipo SOF.

4. CONCLUSÕES

A ferramenta desenvolvida é útil para avaliar a robustez das portas lógicas CMOS na presença de SOFs. Como resultados, a ferramenta apresenta o número total de pares de vetores de teste. Ela diferencia pares de vetores que provocaram a propagação das falhas daqueles que não possuem falha ou que a falha não chega a ser percebida.

Agradecimentos

Este projeto foi apoiado financeiramente pelo programa de Iniciação Científica (PIBIC) da Universidade Federal do Rio Grande (FURG).

5. REFERÊNCIAS

ANGHEL L.; NICOLAIDIS M. *Defects Tolerant Logic Gates for Unreliable Future Nanotechnologies, 9th international work conference on Artificial neural networks*, Alemanha, 2007.

CAO, Y. *New paradigm of predictive MOSFET and interconnect modeling for early circuit simulation*, Custom Integrated Circuits Conference, EUA, 2000.

FRANCO, D. T.. *Fiabilité Du Signal des Circuits Logiques Combinatoires sous Fautes Simultanées Multiples*, Doctoral Thesis, l'École Nationale Supérieure des Télécommunications, França, 2008.

GOMEZ, M.; CHAMPAC, V.; HAWKINS, C.; SEGURA, J. *A Modern Look at the CMOS Stuck-Open Fault*, 10th LATW, 2009.

MATTOS, J.; ROSA Jr, L.; PILLA, M (org.). *Desafios e avanços em computação: o estado da arte*, Pelotas, pp. 221-240, 2009.

NGSPICE. Disponível em: <<http://ngspice.sourceforge.net/>>. Acesso em 10 de jun, 2013.