

UNIVERSIDADE FEDERAL DE PELOTAS  
Faculdade de Ciência da Computação



Trabalho Acadêmico

**Caracterização Automática de Atrasos de Portas  
Lógicas em Circuitos Combinacionais CMOS**

**Gustavo Pereira Mateus**

Pelotas, 2006



**UNIVERSIDADE FEDERAL DE PELOTAS  
INSTITUTO DE FÍSICA E MATEMÁTICA  
DEPARTAMENTO DE INFORMÁTICA  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**CARACTERIZAÇÃO AUTOMÁTICA DE ATRASOS DE PORTAS LÓGICAS EM  
CIRCUITOS COMBINACIONAIS CMOS**

**GUSTAVO PEREIRA MATEUS**

**PELOTAS, 2006**

**GUSTAVO PEREIRA MATEUS**

**CARACTERIZAÇÃO AUTOMÁTICA DE ATRASOS DE PORTAS LÓGICAS EM  
CIRCUITOS COMBINACIONAIS CMOS**

Trabalho acadêmico apresentado ao Curso de Bacharelado em Ciência da Computação da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. José Luís Almada  
Güntzel

**PELOTAS, 2006**

Dados de catalogação na fonte:  
Ubirajara Buddin Cruz – CRB-10/901  
Biblioteca de Ciência & Tecnologia - UFPel

M425c Mateus, Gustavo Pereira

Caracterização automática de atrasos de portas lógicas em circuitos combinacionais CMOS / Gustavo Pereira Mateus ; orientador José Luís Almada Güntzel. – Pelotas, 2006. – 55f. – Monografia (Conclusão de curso). Curso de Bacharelado em Ciência da Computação. Departamento de Informática. Instituto de Física e Matemática. Universidade Federal de Pelotas. Pelotas, 2006.

1.Informática. 2.Microeletônica. 3.Simulação no nível elétrico. 4.Caracterização automática. 5.Atrasos de portas lógicas. 6.SETs. 7.CMOS I.Güntzel, José Luís Almada.  
II.Título.

CDD: 621.3817

**Banca examinadora:**

---

Prof. José Luís Almada Güntzel, Dr. (Orientador)

---

Prof. Luciano Volcan Agostini, MSc.

---

Prof. Marcello da Rocha Macarthy, MSc.

---

Prof. Reginaldo da Nóbrega Tavares, MSc.

## Resumo

A simulação de um circuito no nível elétrico necessita de grande interação humana. Além disso, o tempo de simulação aumenta exponencialmente com o aumento do número de entradas e de portas lógicas no circuito. Este trabalho tem por objetivo propor e analisar um método para automatizar a caracterização de atrasos de portas lógicas em circuitos CMOS usando simulação em nível elétrico e potencialmente, considerando a influência de efeitos transientes, permitindo a simulação das portas lógicas e suas vizinhanças separadas do circuito original. Para isso, são gerados vários subcircuitos com menos portas lógicas, tendo o objetivo de fazer com que o tempo de simulação cresça de forma linear. Além disso, pretendeu-se dar prosseguimento ao desenvolvimento de um método para análise automática da propagação de SETs (*Single Event Transient*), desenvolvido em um trabalho anterior (NEVES, 2006). A ferramenta AGDA (*Automatic Gate Delay Analyzer*), implementada para validar o método proposto no presente trabalho, tem ainda a finalidade de proporcionar um aumento na precisão dos resultados das ferramentas de análise de *timing* desenvolvidas nos últimos anos por membros do Grupo de Arquiteturas e Circuitos Integrados (GACI) da UFPel e do Grupo de Microeletrônica (GME) da UFRGS. De acordo com os resultados desta pesquisa é possível garantir que a ferramenta AGDA possui um bom desempenho, com relação à precisão e à velocidade, na realização de uma análise, mantendo as características elétricas da porta lógica simulada e assegurando sua capacidade de operar em circuitos com maior número de elementos.

Palavras-Chave: Microeletrônica. Simulação no nível elétrico. Caracterização automática dos atrasos de portas lógicas. SETs.

## **Abstract**

Electric-level simulation of a circuit demands high human interaction. Moreover, the simulation time increases exponentially with the increase in the number of inputs and in the number of gates in a circuit. Therefore, this work proposes and evaluates a method to automate gate delay characterization in CMOS combinational circuits. Such method is also able to take into account the influence of transient effects, allowing the simulation of the logic gate to be characterized along with its surrounding gates, separately from the whole circuit. Thus, less complex sub-circuits are generated allowing simulation time to grow linearly. Moreover, it is intended to expand a method for automatic analysis of SET (Single Event Transient) propagation developed by a previously developed work. An Automatic Gate Delay Analyzer (AGDA) tool was implemented to validate the method considered in this work. Such tool is also intended to increase the accuracy of the results provided by timing analysis tools developed at both UFPel's Group of Architectures and Integrated Circuits (GACI) and UFRGS' Microelectronics Group (GME). According to the results achieved by this research it is possible to assure that AGDA keeps a good performance, in terms of precision and speed, to accomplish a full circuit analysis, while keeping the electric characteristics of the characterized gate, assuring its capacity to operate in circuits with more elements.

Keywords: Microelectronics. Electric-level simulation. Automatic characterization of gate delays. SET.

## Lista de Figuras

Figura 1 - Definição dos atrasos de propagação, tempo de descida e tempo de subida.....	16
Figura 2 - Cadeias de inversores com <i>fanouts</i> diferentes .....	17
Figura 3 - Relação entre o atraso de propagação e a capacitância .....	18
Figura 4 - Colisão de um íon pesado gerando um pulso transiente de tensão .....	19
Figura 5 - <i>Single-event transient</i> (SET) e sua possível propagação em um bloco combinacional .....	20
Figura 6 - Atenuação de um pulso devido ao mascaramento elétrico.....	20
Figura 7 - Mascaramento lógico .....	21
Figura 8 - Mascaramento por <i>latching window</i> .....	21
Figura 9 - Modelagem do mecanismo de geração de um SET e formas de onda típicas .....	22
Figura 10 - Gerenciamento das simulações .....	24
Figura 11 - Extração do subcircuito.....	25
Figura 12 - Subcircuito utilizado para caracterizar os atrasos da porta destacada (a) e formas de onda nas entradas e saídas dos inversores de entrada (b).....	26
Figura 13 - Portas lógicas extras no subcircuito.....	27
Figura 14 - Pinos que recebem valores não-controlantes .....	28
Figura 15 - Transistores que geram a transição mais lenta .....	29
Figura 16 - Caracterização do atraso de subida ( $tp_{LH}$ ) .....	29
Figura 17 - Caracterização do atraso de descida ( $tp_{HL}$ ) .....	30
Figura 18 - Subcircuito com inversores nas entradas e saídas .....	30
Figura 19 - Formas de onda geradas pelo simulador elétrico .....	31
Figura 20 - Fonte de tensão <i>Spice</i> .....	32
Figura 21 - Grafo Acíclico Direcionado.....	33
Figura 22 - Regiões afetadas pela ocorrência de SETs de acordo com as entradas <i>a</i> e <i>b</i> .....	40
Figura 23 - Aplicação de um SET em um subcircuito.....	41
Figura 24 - Formas de onda geradas pela aplicação do SET em <i>i5</i> .....	42
Figura 25 - Somador completo de 1 <i>bit</i> .....	43
Figura 26 - Comparação entre os atrasos de subida ( $tp_{LH}$ ) .....	45
Figura 27 - Comparação entre os atrasos de descida ( $tp_{HL}$ ) .....	46
Figura 28 - Comparação entre os atrasos de subida com presença de SET .....	48
Figura 29 - Comparação entre os atrasos de descida com a presença de SET .....	49
Figura 30 - Transistor com tensão de <i>threshold</i> alterada .....	49



## Lista de Tabelas

Tabela 1 - Atrasos de subida e descida para os pinos <i>a</i> e <i>b</i> .....	30
Tabela 2 - Pseudo código do protótipo da ferramenta .....	34
Tabela 3 - Comandos da ferramenta AGDA.....	35
Tabela 4 - Comandos do simulador .....	36

## **Lista de Abreviaturas e Siglas**

*ASPA – Automatic SET Propagation Analyzer*

*AGDA – Automatic Gate Delay Analyzer*

*CMOS – Complementary Metal-Oxide Silicon*

*DAG – Direct Acyclic Graph*

*EDA – Electronic Design Automation*

*FTA – Functional Timing Analysis*

*NMOS – N-Channel MOS*

*PMOS – P-Channel MOS*

*SET – Single Event Transient*

$t_f$  – tempo de fim

$t_i$  – tempo de início

$tp_{HL}$  – *topological delay high-low*

$tp_{LH}$  – *topological delay low-high*

*TSP – Topological SET Propagation*

*TTA – Topological Timing Analysis*

## Sumário

1	Introdução .....	11
1.1	Objetivos .....	14
1.2	Organização da Monografia .....	15
2	Conceitos e Definições.....	16
2.1	Atraso de propagação.....	16
2.2	Relação entre o atraso de propagação e a capacitância .....	17
2.3	<i>Single Event Transients</i> .....	18
3	Um Método para a Caracterização Automática do Atraso de Portas CMOS Usando Simulação no Nível Elétrico .....	23
3.1	Composição do Subcircuito .....	24
3.2	Aplicando Estímulos aos pinos de entrada .....	27
3.3	Caracterização dos atrasos das portas lógicas.....	29
3.4	Implementação do método.....	33
4	Um Método para a Caracterização Automática do Atraso de Portas CMOS Sob a Influência de SET .....	38
4.1	Modelagem do pulso de tensão .....	38
4.2	Composição do Subcircuito .....	41
5	Resultados Experimentais.....	43
5.1	Primeiro Experimento: simulação do circuito completo <i>versus</i> caracterização por particionamento do circuito com a ferramenta AGDA .....	44
5.2	Segundo Experimento: caracterização dos atrasos das portas lógicas sob a influência de SETs.....	47
6	Conclusões .....	51
7	Referências Bibliográficas.....	53

# 1 Introdução

Com a redução nas dimensões dos transistores, proporcionada pela evolução da tecnologia CMOS, cada vez mais transistores podem ser integrados em uma única pastilha de silício (*chip*). Em função disto, os projetistas têm desenvolvido sistemas cada vez mais complexos, contendo quase uma centena de milhão de transistores. Tais projetos requerem o uso intensivo de ferramentas de EDA (*Electronic Design Automation*) eficientes e precisas. Além disso, e mais recentemente, com a redução dos transistores, os circuitos se tornaram mais vulneráveis a ruídos, devido ao efeito de radiação eletromagnética e radiação de origem extraterrestre. Basicamente, as pequenas variações de tensão causadas podem ser interpretadas como inversões do sinal lógico, diminuindo a confiabilidade dos circuitos integrados.

Esses efeitos colaterais devem ser considerados para a análise dos atrasos de portas lógicas CMOS em circuitos com tecnologia nanométrica, portando, neste trabalho foram desenvolvidos métodos para análise de atrasos de portas lógicas com o intuito de aumentar a precisão de ferramentas de EDA existentes, levando em conta a possibilidade de ocorrência de SETs (*Single Event Transient*) na lógica combinacional.

Em trabalhos anteriores do Grupo de Arquiteturas e Circuitos Integrados (GACI) já havia sido desenvolvido o protótipo de ferramenta de EDA com o objetivo de facilitar o projeto de circuitos integrados. No trabalho de Ferrão (2003) foi desenvolvida uma ferramenta de análise de timing funcional baseada em enumeração de caminhos e sensibilização individual, já no trabalho de Neves (2006) foi proposto um método para analisar automaticamente a propagação de SETs em circuitos combinacionais com uso de cálculo lógico-temporal. A fim de complementar estes trabalhos, essa pesquisa propõe o desenvolvimento de um método para automatizar a caracterização de portas lógicas em circuitos CMOS, com a finalidade de tornar mais precisos os resultados das ferramentas citadas.

Sabe-se que as violações temporais em um circuito digital podem causar o armazenamento de um valor não desejado em um elemento de memória. Neste caso, o circuito produzirá um resultado inesperado. Para garantir que os dados capturados pelos elementos de memória sejam corretos em qualquer momento durante a operação do circuito, é necessário que os dados de entrada estejam estáveis em seus valores finais antes da borda ativa do sinal de relógio e que mantenham estes valores por um período de tempo após a borda do sinal.

Entre as técnicas de avaliação de desempenho em circuitos CMOS, evidenciam-se as de análise de *timing*, entre elas: topológica e funcional, além das simulações no nível lógico e no nível elétrico.

As primeiras ferramentas de análise de *timing* seguiam uma abordagem que utilizava apenas a topologia do circuito para estimar o atraso. Esta abordagem é referenciada como análise de *timing* estática ou topológica (*Topological Timing Analysis* - TTA) (DEVADAS, 1994). Entretanto, estas estimativas podem ser muito pessimistas, uma vez que muitos dos caminhos mais longos do circuito, ditos caminhos críticos, podem não ser capazes de propagar transições. Caminhos que nunca propagam transições são chamados de caminhos falsos ou caminhos não sensibilizáveis.

A análise de *timing* funcional (*Functional Timing Analysis* - FTA), por sua vez, considera não somente a topologia do circuito, mas também as relações temporais e funcionais entre seus elementos, tratando, dessa forma, o problema dos caminhos falsos e fornecendo uma estimativa mais precisa (CHEN, 1993; DEVADAS, 1993; DEVADAS, 1994; KUKIMOTO, 1997; MCGEER, 1989). A análise de *timing* funcional conduz a estimativas mais precisas do que a análise de *timing* topológica, uma vez que considera a funcionalidade dos elementos do circuito durante a análise.

A simulação lógica de um circuito é um processo interativo que permite avaliar se a operação do circuito descrito em alto nível vai ao encontro dos objetivos esperados ou se são necessárias modificações nessa descrição. A avaliação é baseada na aplicação de estímulos às entradas do circuito e na observação das suas saídas. Então, é verificado se as saídas estão de acordo com a forma esperada, cumprindo as especificações iniciais em termos de funcionalidade.

A simulação descrita acima faz uma verificação funcional do circuito recorrendo a seus componentes básicos, tais como, inversores e portas lógicas 'E'

ou 'OU' de duas entradas, modeladas através da sua tabela verdade ou usando as instruções que implementam as funções lógicas Booleanas do próprio processador. Em termos temporais, é considerado um atraso unitário, igual para todos os elementos. Considerando portas lógicas constituídas de duas entradas, o seu número tende a ser muito elevado, mesmo em circuitos de baixa complexidade, associado a um extenso conjunto de estímulos de entrada e à natureza computacional do teste o que torna o processo de simulação bastante demorado (MURGAI et al., 1999).

A simulação no nível elétrico é a forma de verificação que conduz a resultados muito precisos. Simuladores elétricos, como o *Spice* (NAGEL, 1975), representam o circuito como uma rede de elementos passivos e ativos (resistores, capacitores, indutores e fontes controladas), montando sistemas de equações diferenciais lineares, os quais são resolvidos para cada passo de simulação.

Embora a simulação seja a abordagem que garante estimativas mais precisas, ela se torna proibitiva, uma vez que, para determinar o atraso crítico do circuito, é necessário simular todas as possibilidades de vetores (ou pares de vetores) de entrada, o que não é factível para o grau de complexidade dos projetos atuais.

Com a evolução da tecnologia de fabricação CMOS, está ficando cada vez mais difícil garantir um funcionamento contínuo livre de falhas para os circuitos integrados, pois para assegurar o funcionamento dos circuitos nas tecnologias de fabricação mais recentes (ditas submicrônicas ou nanométricas), a tensão de alimentação recomendada tem sido reduzida. Esses fatores trazem, como consequência, o aumento da suscetibilidade dos circuitos a falhas causadas por partículas ionizantes. A colisão de partículas carregadas, antes um problema apenas em aplicações espaciais, está se tornando um problema para circuitos operando na superfície terrestre. Atualmente, já é necessário proteger memórias com códigos de detecção e correção de erros. Por outro lado, os pulsos transientes originados nos blocos lógicos, também denominados de *Single Event Transients* (SETs), estão se tornando cada vez mais freqüentes e com maior probabilidade de serem capturados pelos elementos de memória, em função do aumento da freqüência de operação dos circuitos (KASTENSMIDT, 2003).

Para tanto, fazem-se necessários o desenvolvimento de métodos eficientes para a análise da suscetibilidade dos circuitos a SETs. Para a análise da

suscetibilidade, dois aspectos precisam ser considerados: o mecanismo de geração do pulso no local em que ocorreu a colisão da partícula e o fenômeno da propagação do pulso resultante ao longo do circuito, em direção às saídas primárias e, por conseguinte, ao registrador de saída.

A base deste trabalho foi um método desenvolvido por Neves (2006), o qual tem como objetivo definir quais vetores de entrada terão seu resultado modificado pela ação de um transiente no circuito. Para estimar os casos onde ocorre a propagação do pulso foi necessário definir o tempo decorrido desde a geração do transiente até a variação na saída do circuito, já que o mesmo pode sofrer mascaramento por *latch-window* (SHIVAKUMAR et al., 2002), atingindo o elemento de memória fora da janela de *clock*. Esse método visa apresentar informações relevantes aos projetistas de circuitos integrados, no sentido de identificar os componentes mais suscetíveis a tais efeitos, possibilitando proteger as partes mais sensíveis do circuito ou usando técnicas de tolerância à falhas.

O método desenvolvido por Neves (2006) realiza a propagação de valores lógico-temporais, utilizando uma adaptação do conhecido algoritmo *Topological Sort* (CORMEN et. al., 2002) tendo, por isso, recebido o nome de TSP (*Topological SET Propagation*).

Para validar o método TSP foi desenvolvido, durante o Trabalho de Conclusão de Neves (2006), o protótipo de uma ferramenta na linguagem C, chamada de *Automatic SET Propagation Analyzer – ASPA*. Este protótipo assume, inicialmente, apenas portas lógicas NAND, NOR e inversores e adota uma modelagem semelhante a conhecidos métodos de análise de timing (FERRÃO, 2003; WILKE, 2003) modelando o circuito como um Grafo Acíclico Direcionado (*Direct Acyclic Graph - DAG*).

## 1.1 Objetivos

A precisão da análise da propagação de SETs depende da precisão dos atrasos de cada porta do circuito e do número de atrasos por porta. Logo, quanto mais preciso for o modelo de atraso das portas, mais preciso tende a ser o resultado da análise (NEVES, 2006).

O método de Neves (2006) considera os atrasos de propagação de uma porta lógica, segundo o modelo par de atrasos ( $tp_{HL}$  – atraso para propagação de um “0” lógico e  $tp_{LH}$  – atraso para propagação de um “1” lógico), porém necessitando que os atrasos das portas lógicas sejam fornecidos.

Este trabalho tem por objetivo permitir a caracterização automática dos atrasos de portas, sem a necessidade de simular o circuito inteiro. O método desenvolvido gerencia as simulações para a caracterização dos atrasos de portas lógicas em circuitos CMOS por meio de simulação elétrica

Levando-se em conta as diferenças entre atrasos sem a influência de SET e com influência do mesmo, desenvolveram-se, neste trabalho, procedimentos que foram acrescentados ao protótipo do método proposto acima, os quais permitem a caracterização de atrasos onde existe a atuação de transientes.

## 1.2 Organização da Monografia

No capítulo 2 são abordados aspectos relativos a conceitos e definições utilizados no desenvolvimento deste trabalho.

O capítulo 3 apresenta a descrição do método para a caracterização automática do atraso de portas CMOS usando simulação no nível elétrico. Neste capítulo também são apresentados detalhes de implementação, bem como pseudocódigos.

O capítulo 4 descreve um método para a caracterização automática do atraso de portas CMOS sob a influência de SET. Justificando a influência de SETs sobre portas CMOS e descrevendo os procedimentos acrescentados ao método proposto.

O capítulo 5 discute os resultados do desenvolvimento destes métodos, enfatizando as análises obtidas através da realização deste trabalho.

Por fim, no capítulo 6, são apresentadas as conclusões obtidas com a implementação e as análises dos métodos expostos acima, discutindo a contribuição oferecida e a aplicabilidade dos métodos.



## 2 Conceitos e Definições

O objetivo deste Capítulo é dar ao leitor uma visão geral acerca de conceitos relativos a atrasos de propagação e sobre a abordagem de SETs entre outros, de forma que, ao final dela, o leitor esteja familiarizado com as definições utilizadas neste trabalho.

### 2.1 Atraso de propagação

De acordo com RABAEY, CHANDRAKASAN e NIKOLIC (2003), o atraso de propagação de uma porta lógica é definido pela velocidade com que a porta lógica responde a uma mudança nas suas entradas. Normalmente, esse atraso é medido desde o momento em que o sinal de entrada causador da transição atinge 50% do seu valor final até o momento em que o sinal na saída atinge 50% do seu valor final. Esta medida está representada na Figura 1.

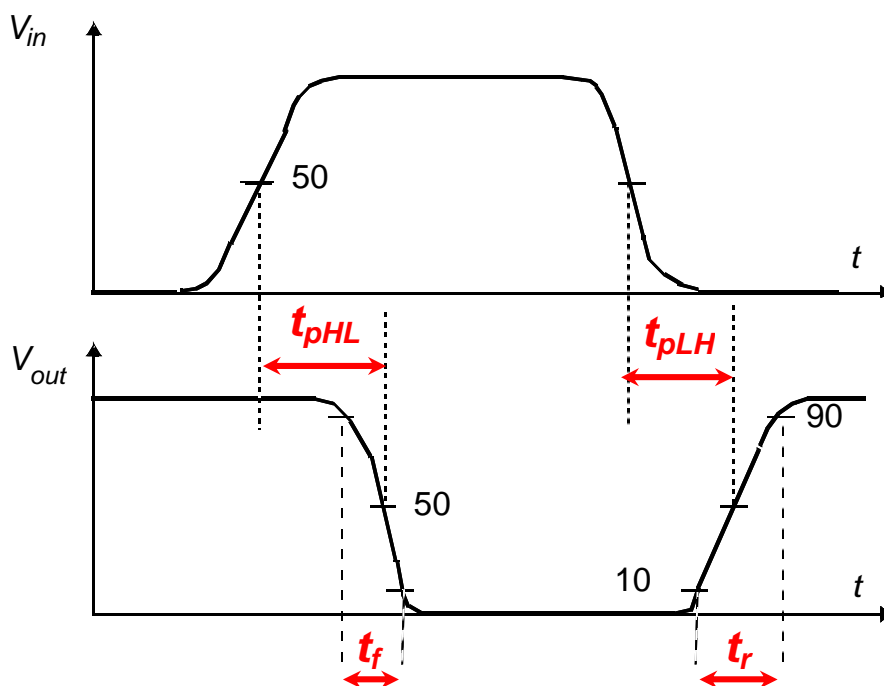


Figura 1 - Definição dos atrasos de propagação, tempo de descida e tempo de subida

A medida do atraso a 50% do valor de tensão é uma convenção amplamente adotada na indústria e na academia. Entretanto, como o formato do sinal de saída de uma porta lógica depende fortemente do formato dos sinais aplicados em suas entradas, outro critério utilizado é o tempo de transição medido sobre o próprio sinal de saída, entre 10% e 90% do valor final, conforme mostrado na Figura 1. O  $t_{pLH}$  define o tempo de resposta para uma transição de subida na saída, enquanto que o  $t_{pHL}$  se refere a uma transição descida. Estes tempos são determinados pela capacidade que uma porta lógica tem de alterar sua saída de acordo com a carga conectada a ela.

## 2.2 Relação entre o atraso de propagação e a capacitância

A Figura 2 apresenta um circuito composto por inversores, em que todos os transistores têm as mesmas dimensões, onde a porta lógica  $X2$  tem *fanout* igual a 1 e a porta lógica  $X5$  tem *fanout* igual a 5. Um pulso é inserido na entrada  $i0$  e seu comportamento é analisado no gráfico da Figura 3, nele é representado o efeito da variação da capacitância em uma cadeia de inversores, onde a transição completa do sinal se torna mais lenta ao serem colocadas mais portas lógicas conectadas a um ponto do circuito ( $i5$ ), se comparado com a forma de onda no ponto  $i2$ . O efeito é semelhante quando se aumentam as dimensões de uma porta lógica nesse ponto.

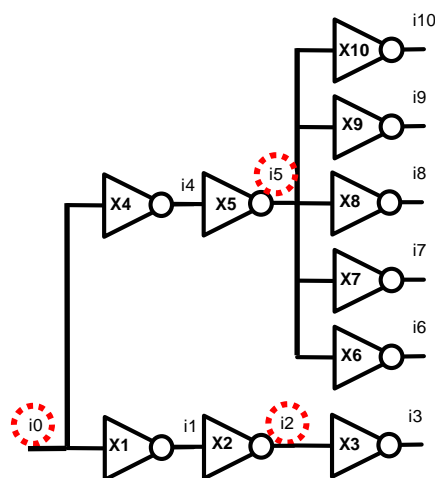


Figura 2 - Cadeias de inversores com *fanouts* diferentes

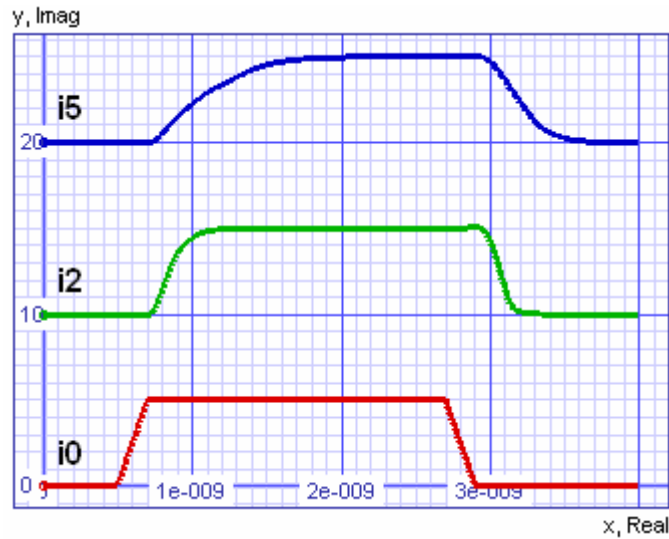


Figura 3 - Relação entre o atraso de propagação e a capacitância

### 2.3 *Single Event Transients*

Um *Single Event Transient* (SET) ocorre quando uma partícula carregada atinge um elemento da lógica combinacional, podendo resultar na injeção de pares elétron-lacuna livres em alguma área ativa do circuito.

Ao nível do mar as principais fontes de radiação capazes de gerar falhas transientes são as partículas alfa, oriundas de impurezas presentes no encapsulamento e no próprio circuito integrado, e nêutrons, oriundos da interação entre raios cósmicos de alta energia e átomos da atmosfera terrestre (NORMAND, 1996; BAUMANN, 2005).

Com a colisão da partícula, uma trilha de ionização poderá formar-se, permitindo a geração de um pulso de corrente, o qual pode transformar-se em um pulso de tensão, como apresentado na Figura 4.

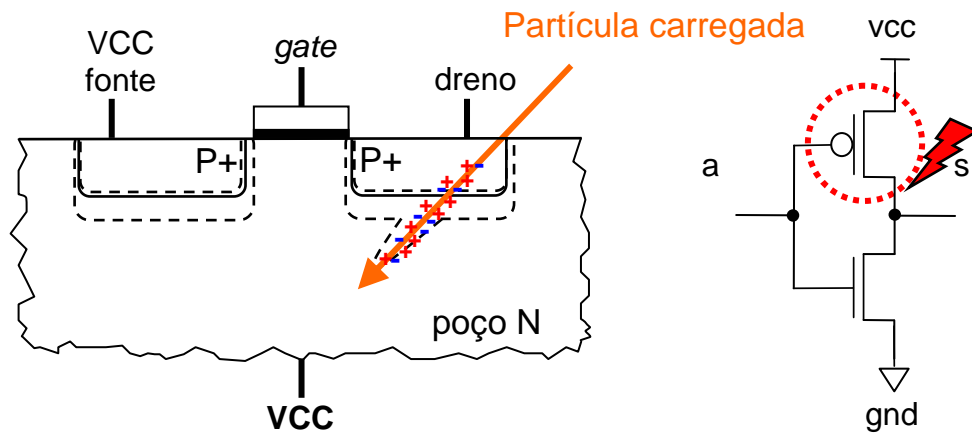


Figura 4 - Colisão de um íon pesado gerando um pulso transiente de tensão

Tipicamente, as regiões sensíveis de um circuito correspondem aos drenos de transistores que se encontram em estado *off* (BAUMANN, 2001). Quando uma partícula carregada colide com o dreno de um transistor PMOS que se encontra em estado *off* (desligado) em uma porta lógica, o pulso de tensão gerado pode carregar a capacitância associada a este dreno, ligando assim o transistor por ele controlado.

Se o pulso de tensão gerado for interpretado como um sinal lógico válido, poderá propagar-se, caso não seja filtrado, pela lógica combinacional até ser capturado por um elemento de memória, causando uma falha no circuito.

A Figura 5 mostra um bloco combinacional limitado por registradores, sendo  $R_i$  o registrador de entrada e  $R_o$  o registrador de saída. Caso um SET ocorra em uma porta que não esteja conectada diretamente a  $R_o$ , então é necessário avaliar se o pulso gerado irá se propagar até  $R_o$  ou não.

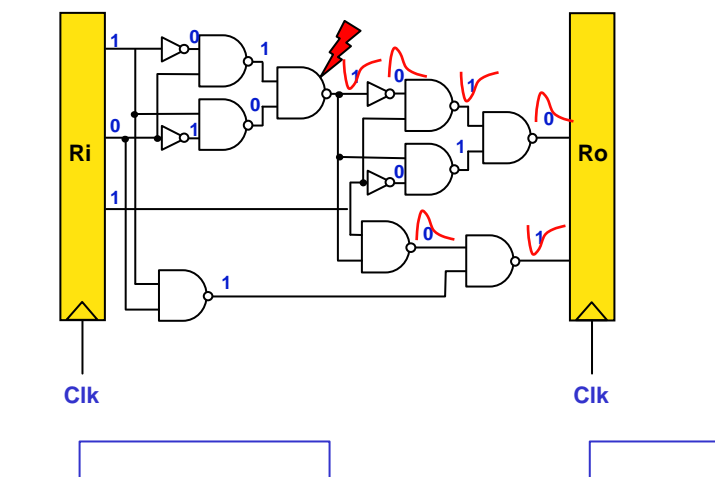


Figura 5 - *Single-event transient* (SET) e sua possível propagação em um bloco combinacional

O circuito da Figura 5 encontrava-se estável sob o vetor de entrada “101” no momento em que um SET ocorreu na saída da porta  $g$ , causando um pulso  $1 \rightarrow 0 \rightarrow 1$  em sua saída. Considerando-se apenas os valores lógicos provocados pelo vetor de entrada “101”, o pulso irá propagar-se pelos caminhos  $\{h, i, j\}$  e  $\{m, n\}$ . É possível observar que o pulso não irá se propagar pela porta  $l$  devido ao “0” presente na saída do inversor  $k$ . Portanto, nota-se que determinadas situações da lógica podem determinar o bloqueio do pulso. Assim, neste exemplo, o pulso gerado por um SET em  $g$  propagou-se por mais de um caminho, atingindo mais de uma saída primária do bloco combinacional. É interessante observar ainda que o SET não afetou todo o circuito, mas apenas o cone lógico que inicia na porta  $g$  e termina no registrador  $Ro$ .

A atenuação total (ou filtragem) de um pulso recebe o nome de mascaramento elétrico (ou temporal) (Figura 6), enquanto que o bloqueio devido ao conjunto de valores lógicos nas entradas da porta recebe o nome de mascaramento lógico (Figura 7). Além destes dois tipos de mascaramentos, há ainda o mascaramento por *latching window* que ocorre quando o pulso atinge o elemento de memória fora da janela de amostragem (Figura 8) (SHIVAKUMAR et al., 2002; WIRTH et al., 2005).

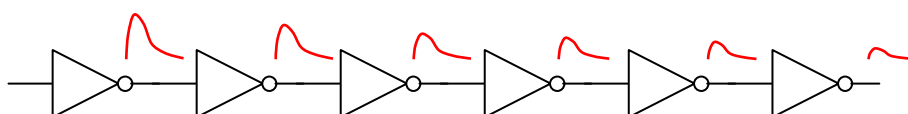


Figura 6 - Atenuação de um pulso devido ao mascaramento elétrico

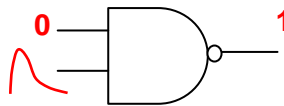
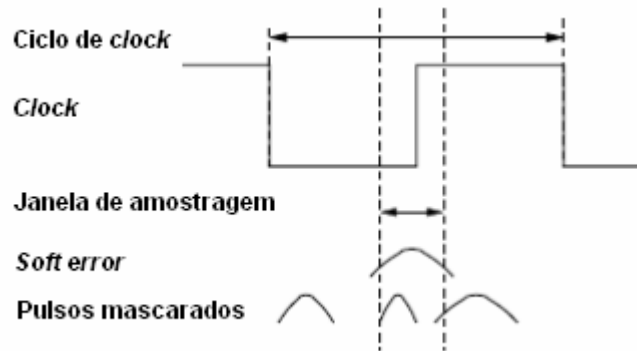


Figura 7 - Mascaramento lógico

Figura 8 - Mascaramento por *latching window*

De acordo com Messenger (1982), uma forma de caracterizar-se um SET em um circuito integrado é modelar o mecanismo de deposição de carga na saída da porta lógica como uma fonte de corrente elétrica descrita por uma dupla-exponencial, conforme mostrado pela equação (1):

$$I(t) = I_0 (e^{(-t/\tau\alpha)} - e^{(-t/\tau\beta)}) \quad (1)$$

onde  $I_0$  é aproximadamente a carga máxima de corrente,  $\tau_\alpha$  é a constante de tempo da junção e  $\tau_\beta$  é a constante de tempo para estabilizar o impacto inicial da partícula.

A Figura 9 mostra a modelagem proposta por Messenger e as formas de onda típicas para uma tecnologia de 100nm, considerando diversos valores de  $I_0$  e  $\tau_\beta$ .

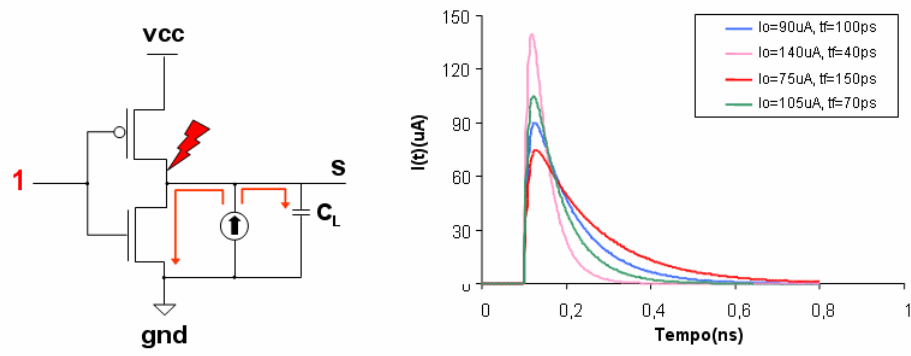


Figura 9 - Modelagem do mecanismo de geração de um SET e formas de onda típicas

### 3 Um Método para a Caracterização Automática do Atraso de Portas CMOS Usando Simulação no Nível Elétrico

A simulação elétrica apresenta estimativas mais precisas em relação a outros métodos para caracterização de circuitos, porém seu custo computacional é muito elevado, restringindo sua aplicabilidade a circuitos pequenos. Sabe-se, que os resultados da simulação são dependentes do conjunto de vetores de entrada, sendo necessária a simulação exaustiva de todos os vetores de entrada para garantir que o pior caso será estimulado. Entretanto, uma simulação exaustiva não é possível nem ao menos para circuitos de tamanho moderado, o que pode ocasionar subestimativas no atraso do circuito.

Considerando que o número de simulações necessárias para determinar os atrasos das portas lógicas de um circuito de tamanho razoável é relativamente grande, desenvolveu-se um método para a caracterização automática dos atrasos das portas.

Esse método baseou-se na idéia de dividir o circuito a ser analisado em subcircuitos, os quais serão submetidos a simulações elétricas para caracterizar os atrasos de cada porta. Esta divisão do problema justifica-se na medida em que o tempo de simulação do circuito completo cresce exponencialmente de acordo com o número de componentes. Assim, o tempo de simulação de vários subcircuitos de menor complexidade (menos portas lógicas) crescerá de maneira linear.

No contexto do presente trabalho, a caracterização do atraso de cada porta é realizada utilizando-se o ambiente *SpiceOpus* (SPICEOPUS, 2006), desenvolvido na Universidade de *Ljubljana*. Este ambiente, por sua vez, tem como base a versão 3f4 do simulador no nível elétrico *Spice* da Universidade de *Berkeley* e oferece algumas facilidades para a visualização das formas de onda resultantes. Todas estas ferramentas são de uso livre. Por outro lado, o método pode ser facilmente adaptado para utilizar outros simuladores similares, tal como o *HSpice*, da empresa *Synopsys* (SYNOPSYS, 2006).

Supondo um circuito  $C$  constituído por  $n$  portas lógicas CMOS, descritas no nível de transistores, então, a caracterização das portas deste circuito exigirá, no



mínimo,  $n$  simulações no nível elétrico. Cada uma das simulações terá como entrada a descrição *Spice* do subcircuito que contém a porta lógica a ser caracterizada, mais as portas lógicas vizinhas que podem influenciar no comportamento elétrico da primeira.

A Figura 10 representa o gerenciamento das simulações, onde o arquivo que descreve o circuito é dividido em vários arquivos que contém os subcircuitos. Então, o simulador elétrico é chamado para cada um dos arquivos gerados. Os resultados da simulação são coletados do arquivo de saída gerado pelo simulador e organizados no formato especificado na ferramenta ASPA.

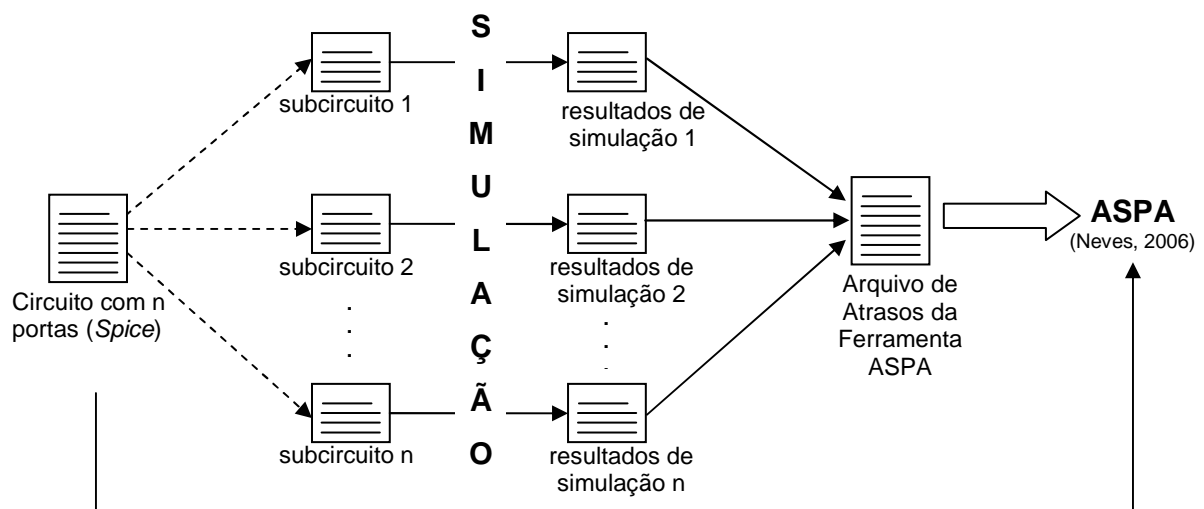


Figura 10 - Gerenciamento das simulações

### 3.1 Composição do Subcircuito

A fim de modelar minimamente os efeitos das portas vizinhas sobre a porta a ser caracterizada, optou-se por compor cada subcircuito com três níveis lógicos: a porta lógica a ser caracterizada, as portas que a precedem (*fanin*) e as portas que a sucedem (*fanout*). Portanto, o subcircuito extraído será composto pelo mínimo de portas lógicas necessárias para a simulação, preservando as características de capacitância vistas na entrada e na saída da porta lógica a ser caracterizada, como é mostrado na Figura 11.

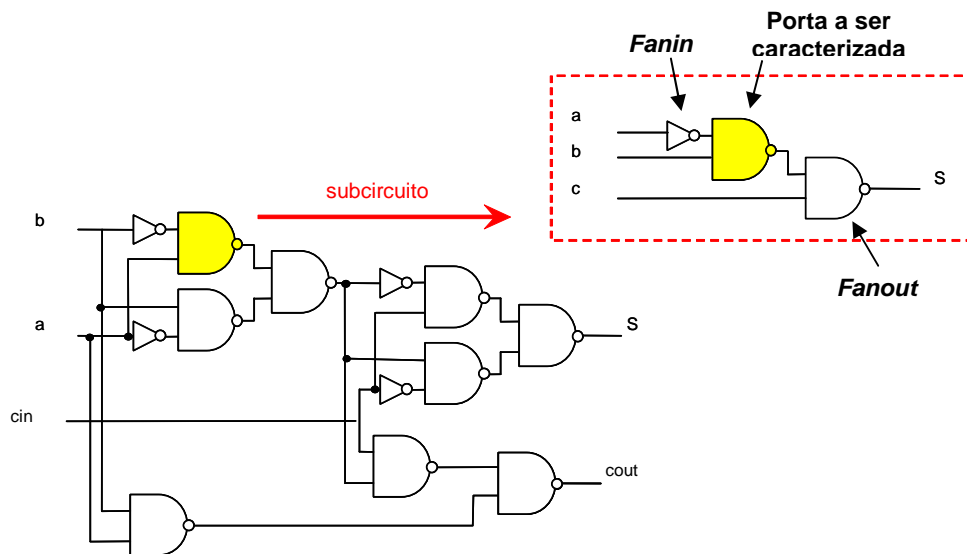
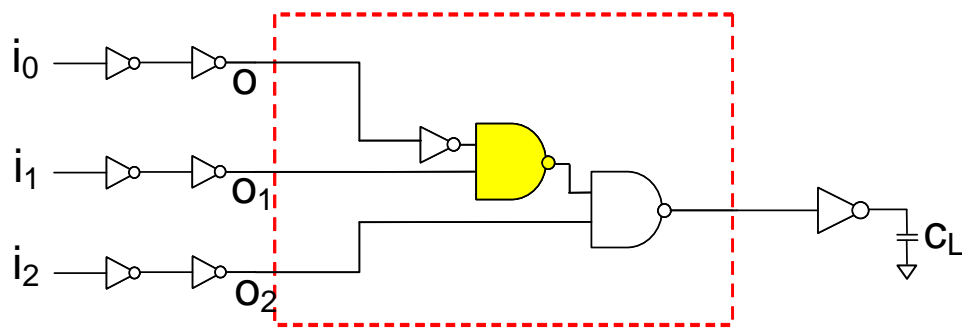
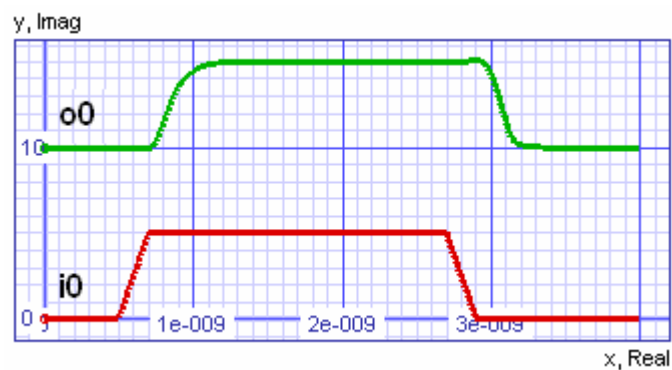


Figura 11 - Extração do subcircuito

A fim de modelar de maneira realista as formas de onda dentro do subcircuito que contém a porta a ser caracterizada, inversores são inseridos nas entradas, dois por entrada. Assim, a forma de onda trapezoidal, aplicada nas entradas dos inversores ( $i_0$ ), tem suas bordas arredondadas, como as de uma transição natural do circuito ( $o_0$ ), como apresentado na Figura 12 (b). Também é acrescentado um inversor em cada saída do subcircuito, a fim de modelar uma carga realista para cada porta de saída, conforme ilustrado na Figura 12 (a).



(a)



(b)

Figura 12 - Subcircuito utilizado para caracterizar os atrasos da porta destacada (a) e formas de onda nas entradas e saídas dos inversores de entrada (b)

Além das portas lógicas que são *fanin* para a porta a ser caracterizada, são consideradas também todas as portas lógicas que são *fanout* das primeiras, já que elas contribuem para compor a capacitância associada a cada nó de entrada da porta a ser caracterizada, influenciando na transição do sinal nesse ponto. A situação descrita anteriormente é ilustrada pela Figura 13, na qual X3 é a porta a ser caracterizada. Além de X3, X7 e X9 contribuem com carga capacitiva para o nó *i5*, e portanto o sinal neste nó (que corresponde à saída da porta X2), transicionará de forma lenta. Semelhantemente, as portas X6 e X8 contribuem com carga capacitiva para o nó *cin*.

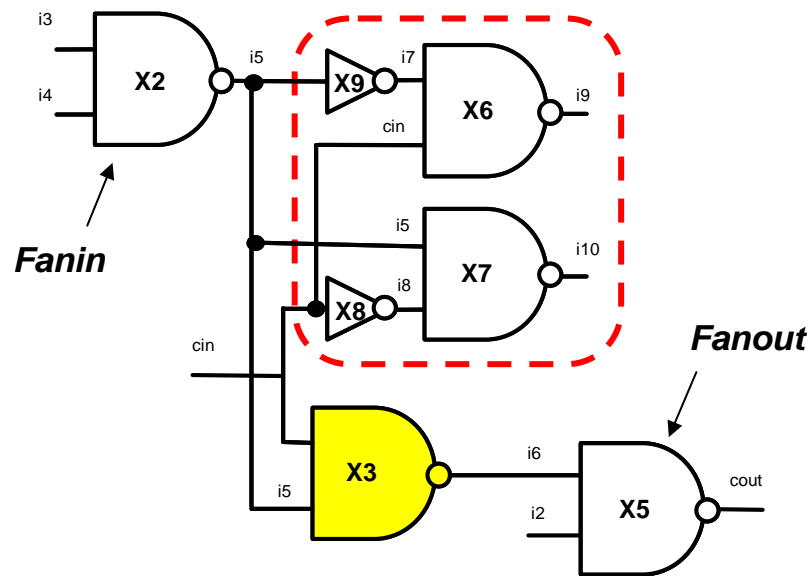


Figura 13 - Portas lógicas extras no subcircuito

### 3.2 Aplicando Estímulos aos pinos de entrada

Conforme já mencionado, um dos objetivos deste trabalho é automatizar a caracterização dos atrasos de descida ( $tp_{HL}$ ) e de subida ( $tp_{LH}$ ) pino-a-pino das portas lógicas de um circuito combinacional CMOS. Para se atingir este objetivo, faz-se necessário aplicar estímulos às entradas de cada porta, de modo que se possa determinar  $tp_{HL}$  e  $tp_{LH}$  com relação a cada uma das entradas. Ao mesmo tempo, deseja-se ser pessimista nesta caracterização. Então, o procedimento normalmente adotado para a caracterização pino-a-pino é transicionar somente uma entrada por vez, mantendo-se todas as demais entradas fixas com o valor não-controlante da porta que se está caracterizando.

O valor controlante de uma porta lógica é definido como sendo o valor que, quando aplicado a alguma de suas entradas, determina, independente dos valores nas outras entradas, o valor lógico na saída da porta. O valor não-controlante é insuficiente para definir o valor lógico na saída da porta.

Neste método, o valor não-controlante é aplicado a todas as entradas que não receberão pulso. A Figura 14 mostra o subcircuito para caracterização da porta lógica X3, com os pinos  $i2$  e  $i3$  recebendo valores não-controlantes.

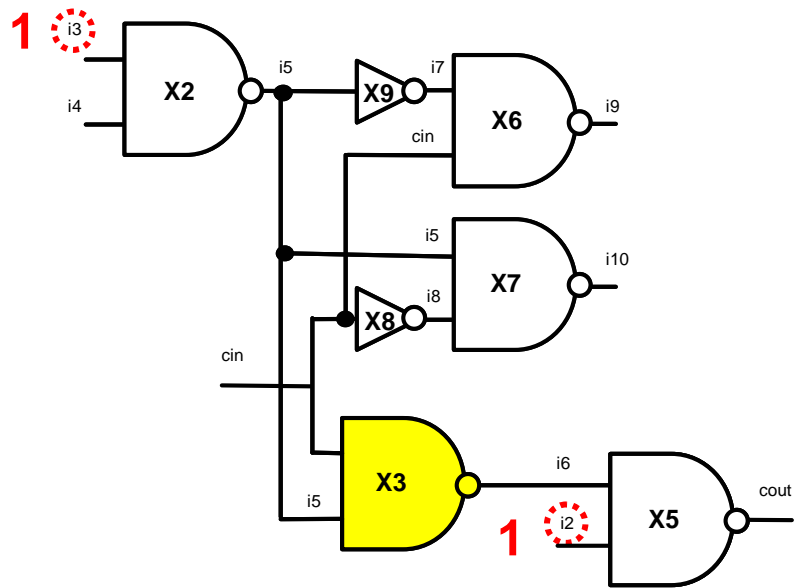


Figura 14 - Pinos que recebem valores não-controlantes

Existem casos onde um subcircuito pode ter várias entradas. Então, nesta situação é fundamental identificar quais são as entradas que geram a transição mais lenta na saída do primeiro nível lógico, já que considerar todas as combinações de entrada tenderia a aumentar consideravelmente o tempo de simulação necessário para caracterizar a porta de um subcircuito.

A identificação dos pinos de entrada do subcircuito que recebem os pulsos é feita através da escolha do pino de entrada nas portas lógicas que são *fanin* à porta que se está caracterizando. Para isso, é essencial realizar uma análise da estrutura interna das portas lógicas envolvidas. Por uma questão de simplicidade, na primeira versão deste método, não foram consideradas portas CMOS complexas.

Conforme demonstrado por Rabaey et al. (2003), a entrada mais lenta de uma porta CMOS corresponde ao *gate* do transistor mais próximo da alimentação, em se considerando associações série. No caso de associações em paralelo, pode-se considerar, em uma primeira aproximação, que todas as entradas possuem o mesmo atraso. A Figura 15 ilustra o critério citado, assumindo-se que os transistores da rede NMOS (*N-Channel MOS*) e os transistores da rede PMOS (*P-Channel MOS*) possuem as mesmas dimensões.

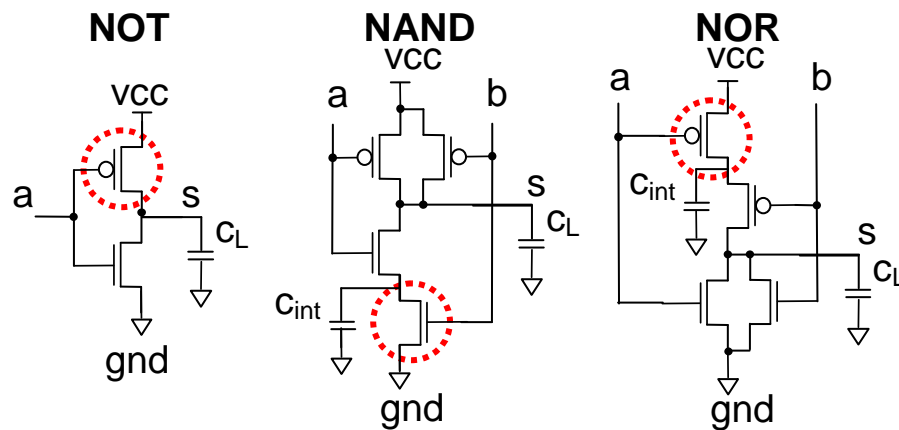


Figura 15 - Transistores que geram a transição mais lenta

De acordo com os exemplos da Figura 15 o pino escolhido é aquele que controla o transistor circulado, isto é, o transistor mais próximo de VCC ou GND e que está associado em série com outros, ou seja, será o caminho com maior resistência e capacitância.

Caso a porta lógica analisada tenha algum pino que pertence à entrada do subcircuito, ele obrigatoriamente deverá receber um pulso, pois o método verifica os atrasos pino-a-pino para cada uma das portas lógicas do circuito.

### 3.3 Caracterização dos atrasos das portas lógicas

Para a caracterização do atraso em um pino de uma porta lógica é necessário medir a diferença entre a transição de sinal na entrada e a transição correspondente no sinal de saída, ambas medidas a  $V_{CC}/2$ . A Figura 16 e a Figura 17 representam a caracterização do atraso de subida ( $t_{pLH}$ ) e descida ( $t_{pHL}$ ) respectivamente, onde  $t_i$  e  $t_f$  são os tempos inicial e final.

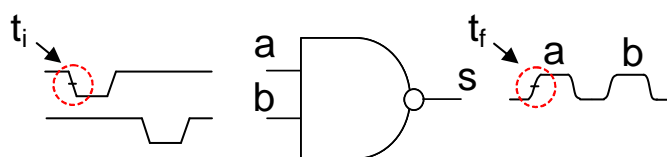


Figura 16 - Caracterização do atraso de subida ( $t_{pLH}$ )

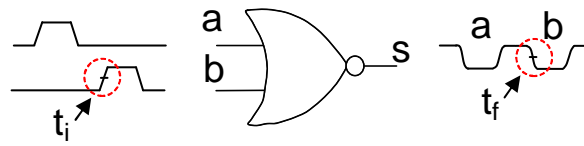


Figura 17 - Caracterização do atraso de descida ( $tp_{HL}$ )

Ao final da simulação, os resultados dos atrasos de subida ( $tp_{LH}$ ) e descida ( $tp_{HL}$ ) são atribuídos a uma tabela, onde é possível observar o atraso pino-a-pino de cada porta do circuito (Tabela 1).

Tabela 1 - Atrasos de subida e descida para os pinos  $a$  e  $b$

	$tp_{LH}$	$tp_{HL}$
$a$	$t_f - t_i$	...
$b$	...	...

O circuito da Figura 18 é utilizado para simular os atrasos da porta lógica  $X3$ , onde os pinos  $i4$  e  $cin$  devem receber pulsos, através dos inversores de entrada. Para que seja possível a medição dos atrasos em  $X3$ , é necessário que os pulsos sejam deslocados em relação ao tempo, caso contrário um mascaramento lógico poderia ocorrer, impedindo a medição. Nesta figura, a região demarcada pelo tracejado quadrado representa a carga conectada as entradas ( $i5$  e  $cin$ ) da porta lógica a ser caracterizada, conforme já mencionado anteriormente.

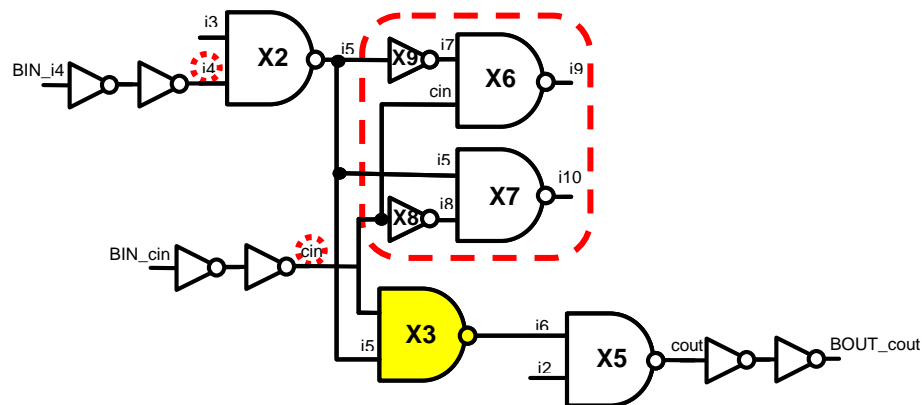


Figura 18 - Subcircuito com inversores nas entradas e saídas

O circuito da figura acima foi simulado no *SpiceOpus* (SPICEOPUS, 2006) para que sejam identificados os pulsos gerados pela ferramenta e os atrasos da porta lógica em questão. Seus resultados são apresentados na Figura 19, onde é possível observar todas as formas de ondas usadas em sua caracterização. A forma de onda no vetor *i6* descreve a saída da porta lógica X3, onde os tempos de subida e descida serão comparados com os tempos nas entradas (*cin* e *i5*), obtendo o  $tp_{LH}$  e  $tp_{HL}$  para cada uma das entradas nessa porta lógica.

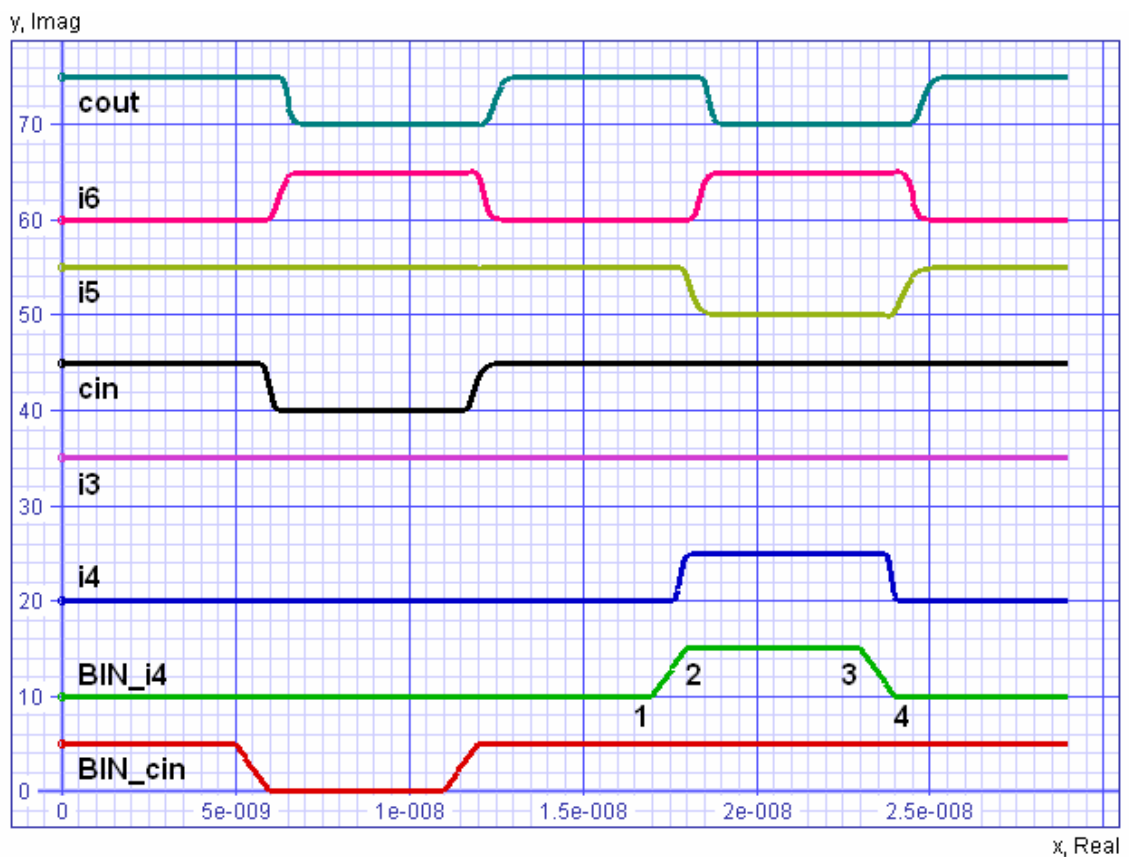


Figura 19 - Formas de onda geradas pelo simulador elétrico

A forma de onda aplicada no pino *BIN\_i4*, na Figura 19, é obtida a partir de um pulso na entrada do circuito gerado pela expressão apresentada na Figura 20, que define uma fonte de tensão denominada *V4*. A seqüência de números no vetor *BIN\_i4* indicados na Figura 19 e na Figura 20 identificam os valores de tempo e tensão do pulso especificado pela fonte de tensão *V4*.



V4 BIN\_i4 0 dc pwl( <sup>1</sup> 17000p 0v <sup>2</sup> 18000p 5v <sup>3</sup> 23000p 5v <sup>4</sup> 24000p 0v )

The diagram shows a SPICE pulse waveform definition: V4 BIN\_i4 0 dc pwl( 17000p 0v 18000p 5v 23000p 5v 24000p 0v ). Four curly braces are used to label segments of the waveform: brace 1 is above the first segment (17000p 0v), brace 2 is below the second segment (18000p 5v), brace 3 is above the third segment (23000p 5v), and brace 4 is below the fourth segment (24000p 0v).

Figura 20 - Fonte de tensão *Spice*

### 3.4 Implementação do método

Para validar o método, foi desenvolvido em linguagem C o protótipo de uma ferramenta denominado de *Automatic Gate Delay Analyzer* (AGDA). Este protótipo considera portas lógicas NAND, NOR e inversores e utiliza uma estrutura de dados desenvolvida por Güntzel (2000), semelhante aos métodos de análise de timing modelando o circuito como um grafo acíclico direcionado - DAG (*Direct Acyclic Graph*). A Figura 21 apresenta um somador completo de 1 *bit* modelado através da estrutura mencionada.

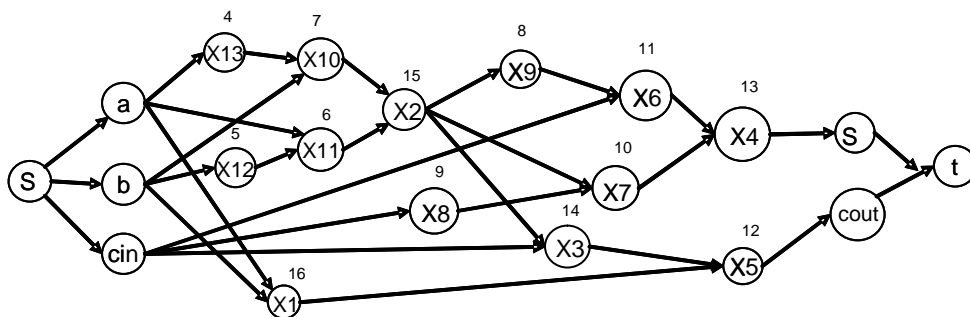


Figura 21 - Grafo Acíclico Direcionado

Cada porta lógica do circuito é representada por um nodo, e as conexões entre portas são representadas por arestas. Entradas e saídas do circuito são representadas por nodos *dummy* e arestas conectando estes nodos ao circuito. No intuito de facilitar a travessia do grafo, a este são acrescentados ainda dois outros nodos *dummy* de polarização, representando fonte (*source s*) e *terminal (t)*. O nodo *s* é conectado por meio de arestas às entradas do circuito e o nodo *t* é conectado por meio de arestas às saídas do circuito.

Um pseudocódigo descrevendo o funcionamento desta ferramenta é mostrado na Tabela 2.

Tabela 2 - Pseudo código do protótipo da ferramenta

Linhas	Comandos
1	<b><u>início</u></b>
2	cria o grafo para o circuito;
3	<b><u>para</u></b> cada porta lógica do circuito
4	desmarca todas as portas lógicas;
5	desmarca todas as redes;
6	<b><u>para</u></b> cada nodo na entrada da porta lógica atual
7	marca o primeiro nível lógico;
8	<b><u>fim para;</u></b>
9	assinala o segundo nível lógico para o nodo atual;
10	<b><u>para</u></b> cada nodo na saída da porta lógica atual
11	marca o terceiro nível lógico;
12	<b><u>fim para;</u></b>
13	<b><u>para</u></b> cada nodo marcado
14	registra as redes de acordo com o nível lógico;
15	<b><u>fim para;</u></b>
16	<b><u>para</u></b> cada rede registrada
17	determina as redes que recebem pulsos;
18	<b><u>fim para;</u></b>
19	escreve o subcircuito no arquivo;
20	escreve os comandos do simulador;
21	chama o simulador;
22	lê os atrasos do arquivo gerado pelo simulador;
23	escreve os resultados no arquivo de saída;
24	<b><u>fim para;</u></b>
25	<b><u>fim.</u></b>

Na ferramenta AGDA foram implementados os comandos descritos na Tabela 3. Essas opções foram desenvolvidas com o intuito de facilitar o processo de configuração e entrada dos dados no protótipo.

Tabela 3 - Comandos da ferramenta AGDA

Opção	Comando	Ajuda
1	-f	<i>Circuit file</i>
2	-tf	<i>Thech file</i>
3	-spth	<i>Simulator path</i>
4	-vcc	<i>VCC</i>
5	-pw	<i>Pulse wide</i>
6	-fsw	<i>Pulse fullswing wide</i>
7	-isc	<i>Input scale</i>
8	-p	<i>PMOS name</i>
9	-n	<i>NMOS name</i>
10	-stp	<i>Simulation step</i>
11	-rm	<i>Remove temp files</i>
12	-sim	<i>Call simulator</i>
13	-osc	<i>Output scale</i>
14	-stop	<i>Stop on end</i>
15	-h	<i>Help</i>

A Tabela 3 foi dividida em três colunas, onde a segunda coluna representa o código que deve ser utilizado na ferramenta e, ao lado, sua descrição. A função correspondente a cada um dos códigos da tabela acima é listada detalhadamente a seguir:

- f** Utilizado para informar o arquivo de descrição *spice* do circuito que será processado.
- tf** Informa o arquivo de tecnologia, assim a ferramenta se torna independente da tecnologia utilizada.
- spth** Diretório onde o arquivo executável do simulador é encontrado.
- vcc** Tensão máxima aplicada no circuito para a simulação.
- pw** Largura do pulso necessária para que o circuito estabilize.
- fsw** Tempo decorrido para que o nível do sinal execute uma transição completa.
- isc** Escala dos tempos de entrada.

- p** Sobrescrever o nome da rede PMOS, caso o arquivo de tecnologia seja diferente do arquivo de descrição *spice*.
- n** Sobrescrever o nome da rede NMOS, caso o arquivo de tecnologia seja diferente do arquivo de descrição *spice*.
- stp** Passo de simulação utilizado pelo simulador elétrico (o tempo total de simulação é calculado automaticamente).
- rm** Apaga os arquivos temporários gerados durante a execução da ferramenta.
- sim** Chama o simulador e executa as simulações.
- osc** Escala dos resultados.
- stop** Mantém o console aberto após a execução da ferramenta.
- h** Chama a tabela de ajuda.

O simulador elétrico utilizado para analisar os atrasos das portas lógicas foi o *SpiceOpus* (SPICEOPUS, 2006). Independe da escolha feita para este trabalho, pode ser utilizado qualquer outro simulador, para isso basta simplesmente adaptar os comandos utilizados. A Tabela 4 abaixo apresenta alguns dos comandos aplicados especificamente para este simulador elétrico.

Tabela 4 - Comandos do simulador

Linhas	Comandos
1	let index_in = 0
2	let index_out = 0
3	cursor index_in right cin 5/2 1 falling
4	cursor index_out right i8 5/2 1 rising
5	echo t <sub>plh_cin_X8</sub> = {time[%index_out] - time[%index_in]}

As linhas 1 e 2 da Tabela 4 declaram variáveis para indexar os vetores que serão usados para a medição. Nas linhas 3 e 4, a variável de índice é retornada contendo a primeira posição em que o vetor indicado (*cin* e *i8*) atinge o valor de  $VCC/2$ , sendo percorrido para a direita, na borda de descida e na borda de subida,

respectivamente. A linha 5 escreve, na saída padrão, o resultado da diferença entre os tempos de simulação apontados pelas variáveis de índice. O exemplo da Tabela 4 retorna o tempo de propagação de subida para a entrada *cin* da porta lógica X8 e saída *i8*. Conjuntos de comandos semelhantes são montados automaticamente pela ferramenta para diferentes casos.

O arquivo de entrada da ferramenta AGDA, que contém o circuito original, tem o formato padrão de descrição de circuitos *spice*. Os arquivos gerados pela ferramenta são acrescidos, além do subcircuito, de informações de identificação de pinos e portas lógicas pertencentes aos subcircuitos; apresentando, também, comandos para descrever o comportamento da tensão nos pinos de entrada e comandos específicos para o simulador elétrico calcular os atrasos das portas lógicas. Os resultados da simulação são salvos em arquivos, que posteriormente serão lidos pela ferramenta AGDA e reunidos em um único arquivo. Este arquivo de atrasos contém a identificação de cada porta lógica do circuito original, com os respectivos atrasos pino-a-pino para a subida e descida do sinal.

## 4 Um Método para a Caracterização Automática do Atraso de Portas CMOS Sob a Influência de SET

Segundo Neves (2006a), o atraso de uma porta lógica sob influência de SET é distinto do atraso da mesma porta quando o circuito opera livre de SET. Isto ocorre porque o pulso gerado por um SET apresenta declividades muito distintas em relação àquelas de uma transição no circuito operando livre de SET. Como o atraso das portas depende do fator de *fanout* e da declividade do sinal de entrada, SETs ocorrendo na entrada de alguma porta irão influenciar seu atraso.

Sabe-se que à medida que um pulso originado de um SET vai se propagando ao longo de uma cadeia, o formato do pulso vai se aproximando ao formato de uma transição natural (NEVES, 2006a), i.e., de uma transição no circuito sem SET. Portanto, a análise dos atrasos na porta lógica sob influência direta do SET é a única considerada neste trabalho, já que a variação dos atrasos nos níveis seguintes é significativamente menor.

Neste trabalho buscou-se considerar atrasos com a presença de SETs no protótipo da ferramenta AGDA. Para isso alguns procedimentos foram acrescentados ao modelo original com o intuito de validar a diferença que caracteriza o atraso de uma porta lógica sob a influência de efeitos transientes.

### 4.1 Modelagem do pulso de tensão

Neste trabalho utilizou-se uma variação da equação de Messenger (1982), a equação (2) apresentada abaixo, onde não é necessário aplicar  $\tau_{\beta}$ , e a constante  $RC$  pode ser diretamente extraída do circuito. Então, para modelar um pulso transiente na simulação de nível elétrico foi utilizada uma fonte de tensão, onde o valor de tensão varia em função do tempo de acordo com a fórmula proposta por Wirth et al. (2005).

$$V(t) = \frac{I_0 \tau_\alpha R}{\tau_\alpha - RC} \left( e^{-\frac{t}{\tau_\alpha}} - e^{-\frac{t}{RC}} \right) \quad (2)$$

Na equação (2), a variável  $V$  é a tensão,  $t$  é o tempo atual na simulação,  $I_0$  é aproximadamente a carga máxima de corrente,  $\tau_\alpha$  é a constante de tempo da junção,  $e$  representa a constante de *Euler*,  $R$  é a resistência equivalente dos transistores ativos e  $C$  é a capacitância associada ao ponto do circuito onde o pulso é inserido.

As variáveis  $I_0$  e  $\tau_\alpha$  são entradas do usuário, pois descrevem o comportamento de um SET e não são extraídas de informações no circuito. A capacitância é obtida pelo somatório das áreas dos *gates* conectados no dreno do transistor que, na simulação, é atingido pelo SET.

Para calcular a resistência equivalente ao caminho por onde um SET é descarregado é necessário observar onde a partícula pode atingir uma porta lógica e qual a configuração dos seus pinos de entrada no momento da colisão. Na Figura 22 são apresentadas as regiões onde é possível ocorrer um SET para um inversor, uma NAND e uma NOR, ambas com duas entradas, bem como as possíveis configurações de entrada para cada caso. As diferentes portas lógicas, de acordo com o valor lógico presente nos pinos de entrada, apresentam diferentes resistências equivalentes pela associação de transistores em série ou paralelo para a carga ou descarga de sua saída.



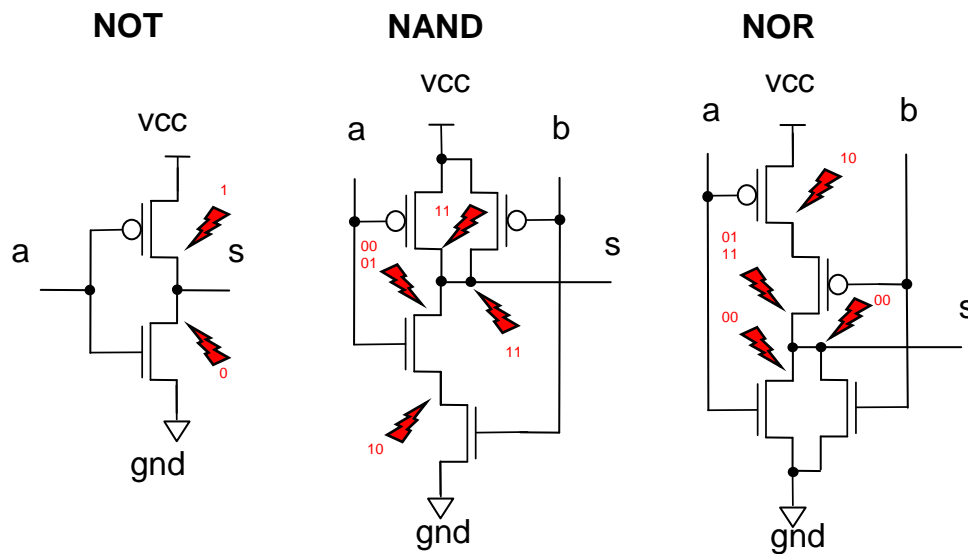


Figura 22 - Regiões afetadas pela ocorrência de SETs de acordo com as entradas a e b

Por motivos de simplificação do modelo, a resistência equivalente foi adaptada conforme a de um inversor, isto é, existem apenas dois casos, a resistência equivalente de um transistor na rede PMOS e a resistência equivalente de um transistor na rede NMOS.

A fórmula utilizada para o cálculo da resistência equivalente é apresentada na equação (3) (WESTE, 1993).

$$R_{eq} = 1 / ((u_0 * (e_{ox} / t_{ox}) * (w / l)) * (v_{gs} - v_t)) \quad (3)$$

A resistência equivalente é representada por  $R_{eq}$ ,  $u_0$  indica a mobilidade dos portadores,  $e_{ox}$  é a permissividade do óxido de silício,  $t_{ox}$  é a espessura do óxido de silício sob o *gate*,  $w$  e  $l$  são as dimensões do transistor em questão,  $v_{gs}$  é a tensão *gate source* (considerada quando o transistor está conduzindo plenamente) e  $v_t$  é a tensão de *threshold*. Os parâmetros  $u_0$ ,  $t_{ox}$ ,  $v_{gs}$ ,  $v_t$  são extraídos do arquivo de tecnologia e  $e_{ox}$  é definido como  $3.5e-11 F/m$ .

## 4.2 Composição do Subcircuito

A Figura 23 apresenta o subcircuito onde a porta lógica circutada X3 é analisada com a presença de um SET na entrada  $i5$ . As outras portas lógicas conectadas em  $i5$  não são mais necessárias, porque suas capacitâncias já são simuladas pela equação da tensão. As entradas circutadas,  $i2$  e  $cin$ , recebem valores não-controlantes para que o efeito do pulso possa ser observado na saída das portas lógicas.

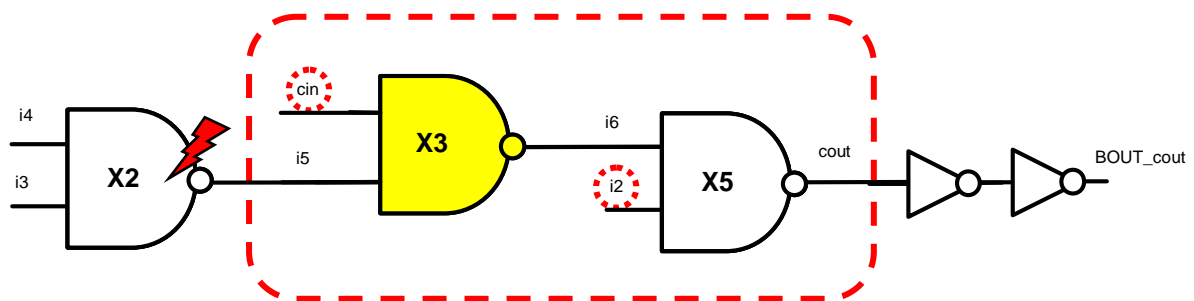


Figura 23 - Aplicação de um SET em um subcircuito

O sinal  $i5$  na Figura 24 representa o pulso de tensão gerado no simulador pela equação que descreve um SET, onde  $I_0$  corresponde a amplitude do pulso e  $\tau_\alpha$  por sua largura. O tempo final de simulação é calculado automaticamente pela ferramenta, ocorrendo quando o pulso chega a valores muito próximos de zero ou VCC. Em  $i6$  é possível visualizar o efeito causado por um SET na porta lógica conectada a X3 e no pino  $cout$  os atrasos já são semelhantes aos de uma transição natural.

Quando uma porta lógica recebe um pulso para VCC (ex. Figura 24), provocado por um transiente em uma de suas entradas, seu atraso de descida é menor se comparado com uma transição natural, porém o atraso de subida é maior. Em casos onde o pulso transiente ocasione uma transição para zero ocorre o efeito inverso.

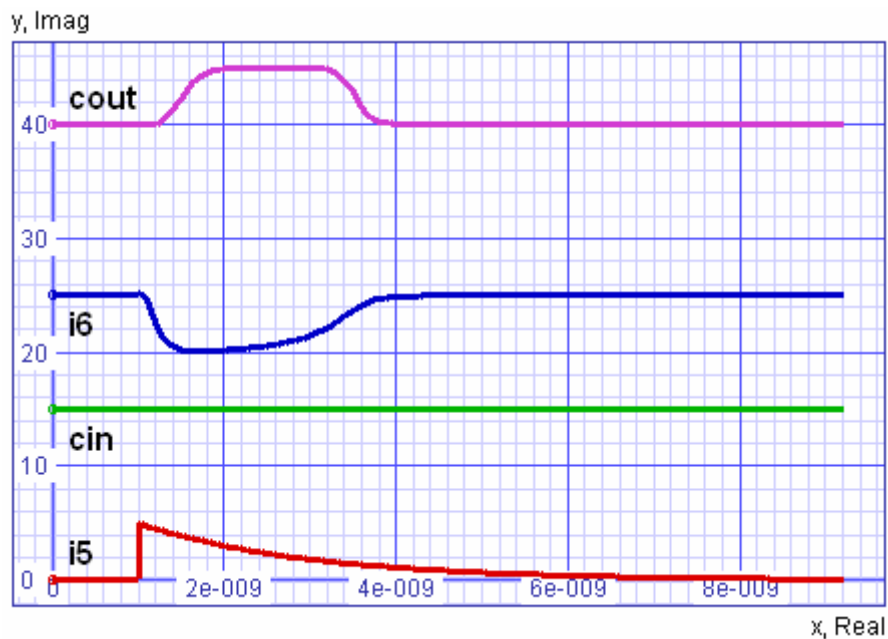


Figura 24 - Formas de onda geradas pela aplicação do SET em  $i_5$

Como os dois casos geram atrasos diferentes para a subida e descida do sinal, o protótipo da ferramenta AGDA executa as duas simulações para cada uma das entradas das portas lógicas. Assim, são montados dois arquivos com o mesmo subcircuito para cada entrada da porta lógica simulada, cobrindo estes dois casos.

## 5 Resultados Experimentais

A fim de validar a ferramenta AGDA, foram realizados dois experimentos. O primeiro experimento visou validar a ferramenta para a caracterização das portas, em termos de atrasos pino-a-pino, considerando circuitos livres de SETs, enquanto que o segundo experimento visou validar o procedimento de caracterização dos atrasos das portas sob influência direta de SETs. Ambos experimentos utilizaram o circuito somador completo de 1 bit que é apresentado na Figura 25. Além disso, foi assumida a tecnologia *AMI05UM*, fornecida pelo serviço de prototipação MOSIS (MOSIS, 2006). Nesta tecnologia, a tensão de alimentação (*VCC*) é 5V.

Na descrição *Spice* do circuito-exemplo, todos os transistores possuem comprimento de canal mínimo ( $L=0,6\mu\text{m}$ ), ao passo que os transistores NMOS e PMOS possuem largura de canal  $W=1,2\mu\text{m}$ .

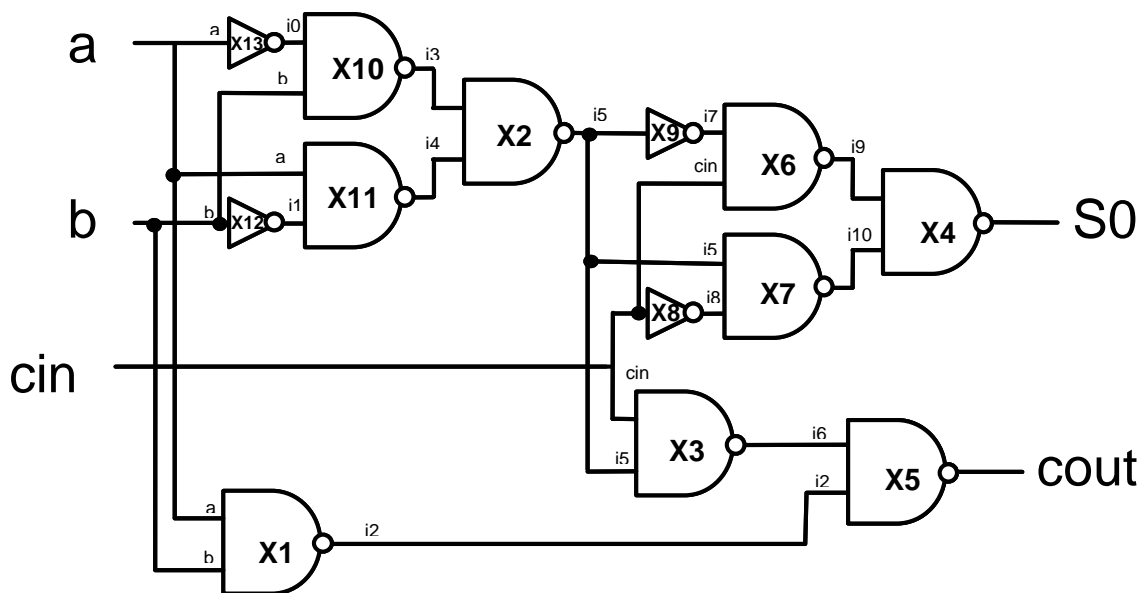


Figura 25 - Somador completo de 1 bit

## 5.1 Primeiro Experimento: simulação do circuito completo *versus* caracterização por particionamento do circuito com a ferramenta AGDA

Este primeiro experimento consistiu na execução de dois métodos: caracterização das portas por simulação elétrica do circuito completo e caracterização das portas uma a uma, através do particionamento do circuito em subcircuitos, utilizando o protótipo de ferramenta AGDA, descrito no capítulo 3. Em ambos os casos, a simulação elétrica foi levada a cabo por meio do simulador *SpiceOpus*.

Para o primeiro método, adotou-se um conjunto de vetores gerados de forma aleatória para estimular as entradas do circuito, uma vez que o número de vetores de entrada pode ser muito elevado. Teoricamente, dever-se-ia simular todos os  $2^n \times (2^n - 1)$  pares de vetores existentes (com  $n$  sendo o número de entradas do circuito). Porém, mesmo para circuitos com um número moderado de entradas, o total de pares de vetores a serem simulados pode ser proibitivamente alto. Por isso utilizam-se conjuntos de vetores gerados aleatoriamente.

A duração do pulso gerado, necessária para a estabilização do circuito, e o tempo decorrido para ocasionar uma transição completa foram inicialmente determinados como sendo de 5ns e 200ps, respectivamente. Estes valores foram os parâmetros aplicados para realizar as simulações.

Os resultados obtidos para os dois métodos podem ser observados nos gráficos ilustrados na Figura 26 (para  $tp_{LH}$ ) e na Figura 27 (para  $tp_{HL}$ ). Os gráficos demonstram a relação entre os atrasos absolutos para cada um dos pinos de entrada de todas as portas do circuito, onde para cada dupla de barras verticais, a barra da esquerda representa o atraso obtido pelo primeiro método (simulação elétrica do circuito completo) e a barra da direita mostra o atraso medido pelo segundo método (particionamento do circuito e simulação, gerenciados pelo protótipo de ferramenta AGDA). Nestes gráficos, a ordenada indica o atraso, em pico segundos, enquanto que a abscissa lista todas as portas e, para cada porta, cada pino de entrada. Por exemplo, no gráfico da Figura 26, a porta *X11* aparece duas vezes: uma vez para o pino de entrada “*i1*” e outra vez para o pino de entrada “*a*”.

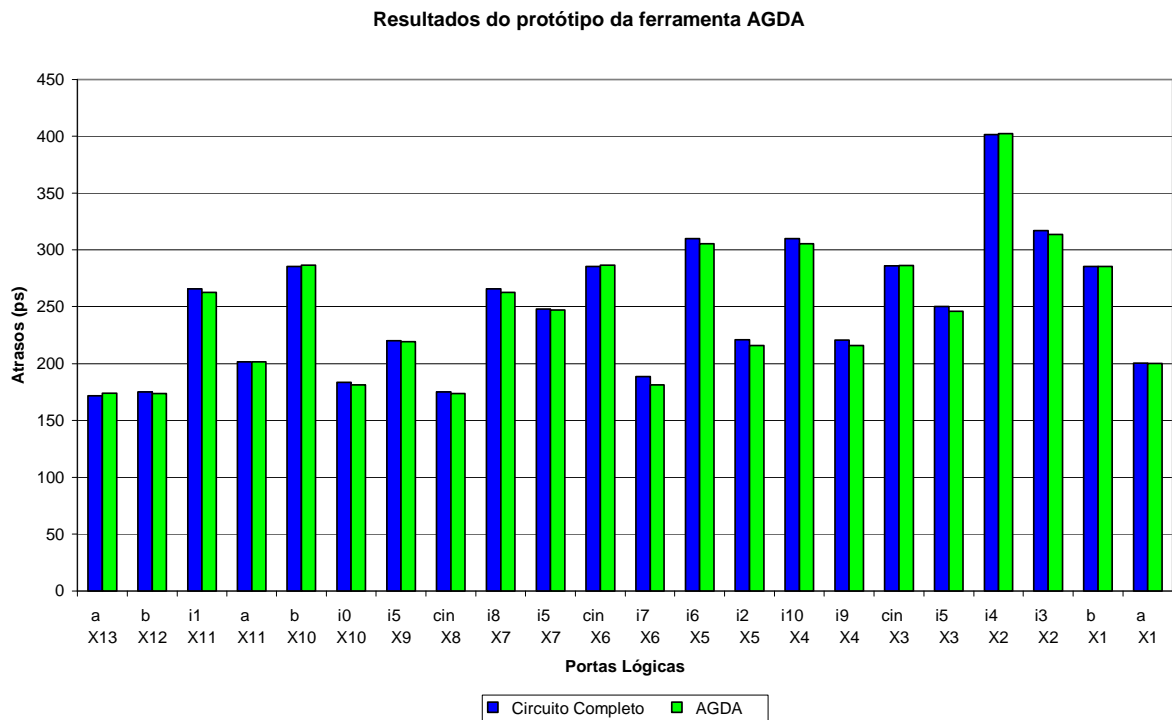


Figura 26 - Comparação entre os atrasos de subida ( $tp_{LH}$ )

Observando o gráfico da Figura 26 é possível notar que os resultados da comparação são muito próximos, assegurando que o protótipo da ferramenta AGDA tem um grau de precisão considerado satisfatório para o cálculo dos atrasos de subida.

A comparação entre os resultados dos conjuntos de atrasos para transições de descida do sinal ( $tp_{HL}$ ) são exibidos na Figura 27.

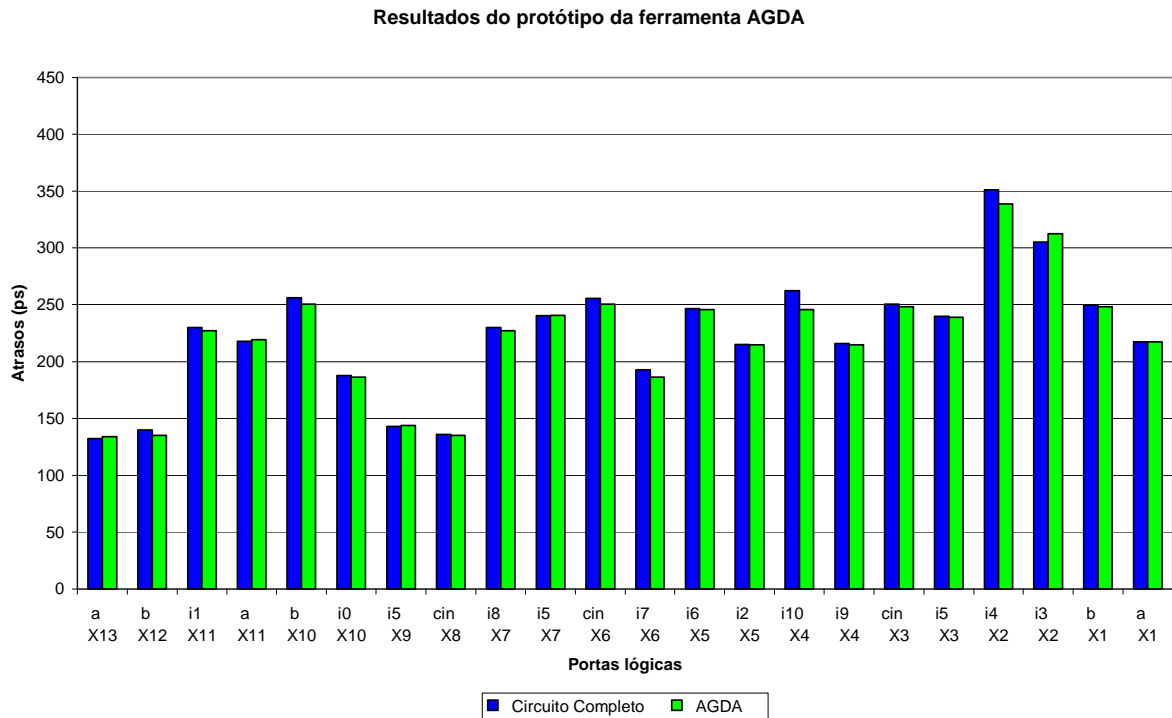


Figura 27 - Comparação entre os atrasos de descida ( $tp_{HL}$ )

Analisando o gráfico da Figura 27 é possível observar que nem todos os resultados são tão próximos quanto os do gráfico na Figura 26, porém na grande maioria dos casos o grau de precisão do protótipo da ferramenta AGDA ainda foi significativamente aceitável, mantendo a média acima de 98% para os dois casos ( $tp_{HL}$  e  $tp_{LH}$ ) e com erro percentual máximo inferior a 7% para o pior caso ( $tp_{HL}$  em  $X4$ ,  $i10$ ) e 4% para  $tp_{LH}$  em  $X6$ ,  $i7$ .

O tempo de execução para a simulação do circuito completo (aproximadamente 60 combinações de vetores possíveis para 3 entradas) foi de 93 segundos, enquanto que, para o protótipo da ferramenta AGDA, foi de apenas 25 segundos, sendo 3,72 vezes mais rápido e comprovando sua eficiência até mesmo para circuitos pouco complexos. A avaliação de desempenho das duas versões do método foi executada em um microcomputador utilizando o processador Intel(R) Celeron(R) com velocidade de 1.7 GHz e 1Gb de memória RAM.

## 5.2 Segundo Experimento: caracterização dos atrasos das portas lógicas sob a influência de SETs

Para a realização do segundo experimento, foram considerados os parâmetros  $I_{0n}$ ,  $I_{0p}$ ,  $t_{an}$  e  $t_{ap}$  com os seguintes valores na mesma seqüência: 50uA, 32uA, 2ns e 3ns.

Para analisar o efeito de um transiente no atraso das portas lógicas foram utilizados como referência os atrasos das portas no circuito livre de SET. Os gráficos da Figura 28 e da Figura 29 apresentam as comparações relativas ao atraso de subida e ao atraso de descida, considerando SETs do tipo 1→0→1 e SETs do tipo 0→1→0. Nestes gráficos, a ordenada representa o atraso das portas, em pico segundos, enquanto que a abscissa apresenta as portas lógicas e seus pinos de entrada. As três barras representam o atraso para SETs do tipo 1→0→1, 0→1→0 e sem a presença de SETs, respectivamente.

Conforme é possível observar na Figura 28, o atraso de subida para SETs que levam o sinal até VCC é significativamente maior do que o atraso de uma transição natural, isso se deve ao fato de que a transição completa de descida do sinal é muito lenta em relação à de subida.



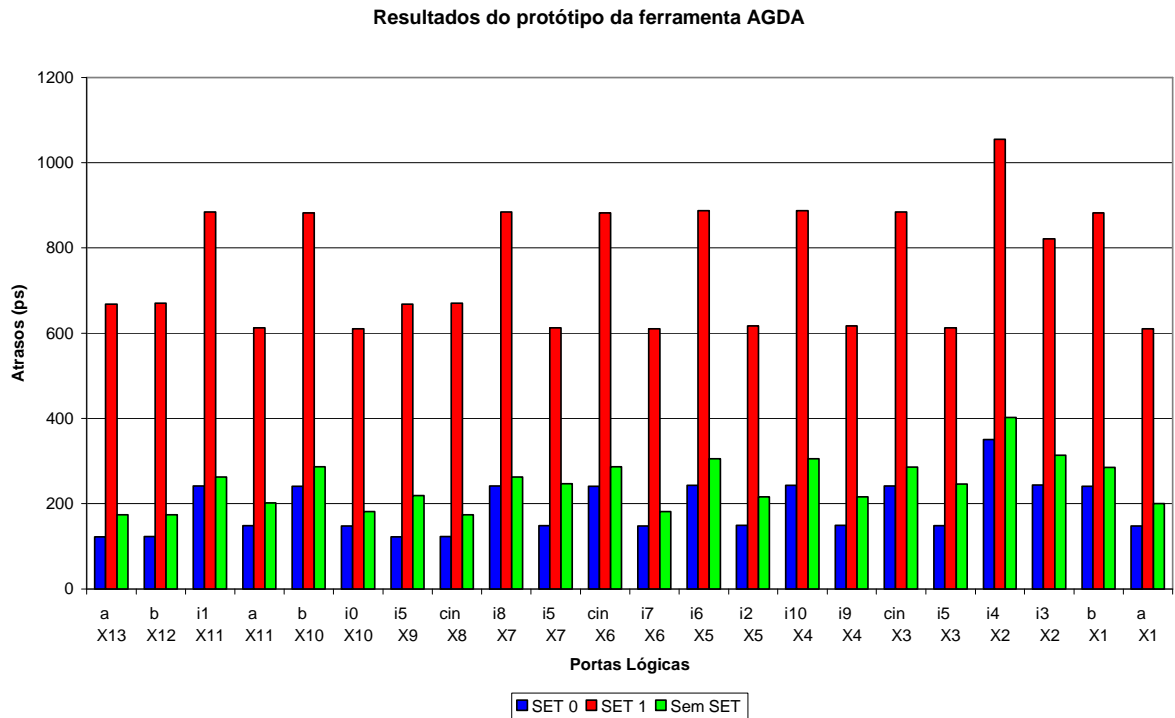


Figura 28 - Comparação entre os atrasos de subida com presença de SET

O atraso de subida para SETs do tipo  $1 \rightarrow 0 \rightarrow 1$  é menor pois a inclinação de descida de um SET, neste caso, é maior do que a de uma transição natural. O mesmo efeito ocorre para SETs do tipo  $0 \rightarrow 1 \rightarrow 0$  quando o atraso de descida é medido, já que a inclinação de subida deste tipo SET é maior do que a de uma transição natural.

Observa-se, no gráfico da Figura 29, que o atraso de descida para SETs em zero nas portas lógicas NAND é muito diferente para cada pino.

Resultados do protótipo da ferramenta AGDA

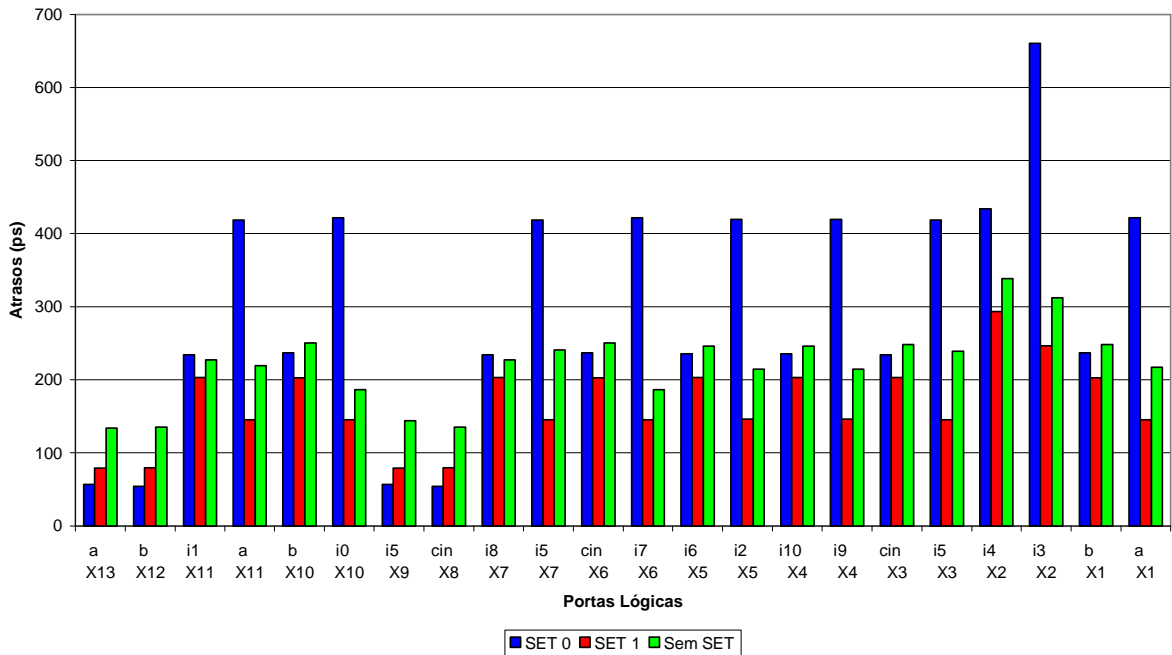


Figura 29 - Comparação entre os atrasos de descida com a presença de SET

Essa diferença dos atrasos entre os pinos das portas lógicas NAND acontece porque os transistores que têm o *gate* ligado a entradas com maior atraso apresentam uma tensão de *threshold* mais elevada por não estarem diretamente conectados ao GND, como é mostrado na Figura 30.

NAND X5

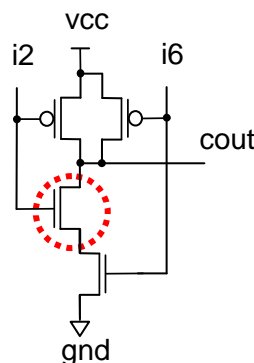


Figura 30 - Transistor com tensão de *threshold* alterada

O atraso em pinos conectados a transistores com tensão de *threshold* mais elevada é maior, já que um SET em zero apresenta uma pequena inclinação de

subida do sinal se comparada com a descida, então o tempo necessário para que o sinal se aproxime da tensão de *threshold* e ocorra uma transição na saída da porta lógica é maior.

## 6 Conclusões

Este trabalho propôs um método, bem como sua implementação, para caracterizar o atraso de portas lógicas em circuitos combinacionais CMOS de forma automática. O método fornece uma forma de determinar os atrasos de portas lógicas por meio da divisão do circuito original em diversos subcircuitos, com menor número de portas lógicas, para serem submetidos à simulação elétrica.

Sabe-se que a simulação de um circuito de tamanho moderado sem particionamento acarreta em onerosos tempos de simulação e diversas iterações com o usuário para gerar um conjunto de vetores de entrada. Além disso, não é garantido que o pior caso para cada entrada de todas as portas lógicas seja estimulado.

Embora a simulação no nível elétrico gere os resultados mais precisos, ela também apresenta os maiores tempos de execução, de modo que seu uso acaba ficando restrito a circuitos muito simples, fora da realidade. A simulação em níveis mais abstratos tende a ser mais rápida, mas também menos precisa. Assim, o método proposto neste trabalho assegura um bom compromisso entre precisão e velocidade na realização da análise, já que o circuito original é decomposto em subcircuitos com o menor conjunto de portas lógicas possível. Isso faz com que os tempos de simulação cresçam de forma linear, garantindo, assim, que este método é capaz de operar em circuitos com um número significativo de portas lógicas. Mesmo com o particionamento do circuito, a porta simulada mantém suas características em relação aos atrasos, indicando que a modelagem escolhida é adequada.

Para validar o método foi desenvolvido, em linguagem C, o protótipo de uma ferramenta, chamada de AGDA. O protótipo desta ferramenta apresentou excelentes resultados ao ser aplicado a um circuito de teste para comparações entre as simulações. Com precisão média acima de 98% e com erro percentual máximo inferior a 7% para o pior caso, se comparado a métodos de geração aleatória de vetores de entrada para um circuito sem particionamento, este método apresenta-se

como uma alternativa viável para a caracterização de atrasos de portas lógicas em circuitos CMOS.

Os resultados das simulações gerenciadas pela ferramenta compõem um arquivo-relatório, contendo os atrasos a serem utilizados na ferramenta ASPA.

Neste trabalho, também se implementaram procedimentos na ferramenta AGDA para obter a caracterização de atrasos em portas lógicas sob a influência de SETs de modo automático, além da redução dos parâmetros necessários para modelar pulsos transientes, facilitando a análise de circuitos operando nestas condições.

Como trabalhos futuros podem ser implementados procedimentos para que a ferramenta utilize o simulador elétrico *HSpice*, alterando os comandos gerados pela ferramenta de forma que se ajustem a este simulador. Assim, é possível utilizar parâmetros tecnológicos desenvolvidos para este simulador. Também podem ser executados mais testes com circuitos de diferentes níveis de complexidade. Ainda podem ser acrescentados outros modelos para o cálculo da resistência equivalente, tornando a análise dos efeitos transientes mais abrangente. Podem ser levados em consideração os efeitos das capacitâncias de interconexões, pois elas também influem no atraso das portas. Para isso, a estrutura de dados utilizada deve ser atualizada para armazenar esses valores. Como o arquivo de atrasos gerado ao final da computação foi especificado pelo formato lido pela ferramenta ASPA, este pode ser alterado de forma que a ferramenta crie arquivos de atrasos no mesmo padrão utilizado pela indústria, sendo necessárias alterações na geração do relatório de saída da ferramenta. Portanto, estas modificações aumentam a precisão e as possibilidades de uso da ferramenta AGDA.

Apesar das dificuldades enfrentadas sobre como definir os pinos que receberiam os pulsos para estimular o pior caso na porta lógica a ser caracterizada e os problemas relacionados à inserção de pulsos para modelar efeitos transientes, a avaliação final desta pesquisa foi satisfatória, pois atendeu a todos os requisitos estabelecidos em seus objetivos, garantindo, principalmente, boa precisão nos resultados.

## 7 Referências Bibliográficas

BAUMANN, R. C. Soft Errors in Advanced Semiconductor Devices - Part I: The Three Radiation Sources. **IEEE Transactions on Device and Materials Reliability**, v. 1, n. 1, March 2001.

BAUMANN, R. C. Radiation-Induced Soft Errors in Advanced Semiconductor Technologies. **IEEE Transactions on Devices and Materials Reliability**, New York, v.5, n.3, p.305-316, September, 2005.

CHEN, H-C.; DU, D. Path Sensitization in Critical Path Problem. **IEEE Transactions on CAD of integrated Circuits and Systems**, Los Alamitos, California, v.12, n.2, p.196-207, February 1993.

CORMEN, T.; LEISERSON, C.; RIVEST, R.; STEIN, C. **Algoritmos: Teoria e Prática**. 2 ed. Rio de Janeiro, Brasil: Editora Campus, 2002.

DEVADAS, S.; KEUTZER, K.; MALIK, S.; WANG, A. Computation of Floating Mode Delay in Combinational Circuits: Practice and Implementation. **IEEE Transactions on Computed-Aided Design of Integrated Circuits and Systems**, Los Alamitos, California, v.12, n.12, p.1924-1936, December 1993.

DEVADAS, S.; GHOSH, A.; KEUTZER, K. **Logic Synthesis**. New York: McGraw-Hill, 1994. 404p.

FERRÃO, Daniel Lima. **Desenvolvimento De Uma Ferramenta De Análise De Timing De Circuitos Combinacionais Baseada Em Sensibilização De Caminhos**. Trabalho de conclusão de curso. Pelotas: UFPel, abril, 2003.

GÜNTZEL, José L. **Functional Timing Analysis of VLSI Circuits Containing Complex Gates**. Tese de Doutorado. Porto Alegre: PPGC da UFRGS, novembro 2000.

KASTENSMIDT, Fernanda. **Designing Single Event Upset Mitigation Techniques for Large SRAM-based FPGA Components**. Tese de Doutorado. Porto Alegre: PPGC da UFRGS, 2003.

KUKIMOTO, Y. et al. Approximate Timing Analysis of Combinational Circuits under the XBD0 Model. In: ACM/IEEE INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN, 1997, San Jose, California. **Proceedings...** Los Alamitos, California: IEEE Computer Society Press, 1997. p.176-181.

MESSENGER, C. G., "Collection of Charge on Junction Nodes from Ion Tracks", **IEEE Transactions on Nuclear Science, Proceedings...** 1982. Vol. NS-29, Dec. 1982, pp. 2024-2031.

MCGEER, P.; BRAYTON, R. Efficient Algorithms for Computing the Longest Viable Path in a Combinational Network. In: ACM/IEEE DESIGN AUTOMATION CONFERENCE, 26., 1989, Las Vegas, Nevada. **Proceedings...** Los Alamitos, California: IEEE Computer Press, 1989. p.561-567.

MOSIS. Disponível em: <<http://www.mosis.org/>>. Acesso em: 19 out. 2006.

MURGAI, R. and Fujita, M. 1999. Speeding up Look-up Table Driven Logic Simulation. Proc. of the 10th International Conference on Very Large Scale Integration, 1999, p. 385-397.

NAGEL, W. **SPICE2: A Computer Program to Simulate Semiconductor Circuits**. Berkeley, California: University of California, Department of Electrical Engineering and Computer Sciences, 1975. 63p. (UCB/ERL M75/520).

NEVES, Carolina. **Análise Automática da Propagação Single Events Transients em Circuitos Combinacionais CMOS**. Trabalho de Conclusão de Curso. Pelotas: UFPel, abril 2006.

NEVES, C. G., MATEUS, Gustavo Pereira, AGOSTINI, L. V., LIMA-KASTENSMIDT, F., GUNTZEL, J. L. On the influence of single event transient on gate delays In: 21th South Symposium on Microelectronics, 2006, Porto Alegre, RS. **Proceedings** of the 21th South Symposium on Microelectronics. Porto Alegre, RS: Gráfica Guarani, 2006.

NORMAND, E. Single event upset at ground level. **IEEE Transactions on Nuclear Science**, New York, v.43, n.6, p. 2742 -2750, December, 1996.

RABAEY, Jam M.; CHANDRAKASAN, Anantha; NIKOLIC, Borivoje. **Digital Integrated Circuits: A design Perspective**. 2 ed. New Jersey: Pearson Education, 2003. 27p.

SHIVAKUMAR, P. et al. Modeling the Effect of Technology Trends on the Soft Error Rate of Combinational Logic. In: **International Conference On Dependable Systems And Networks**, 2002. P.389-398.

SPICEOPUS. Disponível em: <<http://fides.fe.uni-lj.si/spice/>>. Acesso em: 26 set. 2006.

SYNOPSYS. Disponível em: < <http://www.synopsys.com/>>. Acesso em: 19 out. 2006.

WESTE, Neil; ESHRAGHIAN, Kamran. **Principles of CMOS VLSI Design: a systems perspective**. Addison-Wesley, 2nd Edition, 1993.

WIRTH, G. I. ; VIEIRA, Michele G ; HENES NETO, Egas ; KASTENSIMIDT, Fernanda G L . Single Event Transients in Combinatorial Circuits. In: 18th International Symposium on Integrated Circuits and Systms Design, 2005,

Florianópolis. **Proceedings** of the SBCCI 2006. New York (USA): ACM: Association for Computing Machinery, 2005. p. 121-126.