



## SALÃO DE INICIAÇÃO CIENTÍFICA XXVIII SIC

paz no plural



<b>Evento</b>	Salão UFRGS 2016: SIC - XXVIII SALÃO DE INICIAÇÃO CIENTÍFICA DA UFRGS
<b>Ano</b>	2016
<b>Local</b>	Campus do Vale - UFRGS
<b>Título</b>	Automação de Modelo de Atraso para Gates CMOS Estáticos
<b>Autor</b>	LUCAS CARRARO
<b>Orientador</b>	RENATO PEREZ RIBAS

Automação de Modelo de Atraso para Gates CMOS Estáticos  
Lucas Carraro, Renato P. Ribas  
Universidade Federal do Rio Grande do Sul

A metodologia de projeto de circuitos digitais baseada em células padrão, amplamente aplicada na concepção de circuitos VLSI, utiliza uma biblioteca de células para mapear componentes físicos em um design que corresponde a um circuito lógico. A qualidade desse projeto depende diretamente das especificações das células usadas e, portanto, é importante que elas sejam previamente caracterizadas. Entretanto, a caracterização através de simulação elétrica é computacionalmente custosa. Devido a isso, modelos matemáticos são estudados com o objetivo de acelerar esse processo admitindo-se uma troca de exatidão por desempenho. A precisão dos modelos pode variar dependendo do circuito ao qual ele é aplicado e, portanto, é interessante testá-los exaustivamente a fim de garantir que eles sejam suficientemente corretos.

Este trabalho apresenta uma abordagem para automatizar a avaliação do atraso de células baseado no modelo analítico detalhado em [1]. O desafio consiste em extrair da rede de transistores todos os parâmetros usados nas equações matemáticas do modelo. Ao contrário da simulação elétrica, a extração dos parâmetros e a aplicação deles nas equações são procedimentos relativamente rápidos e de baixo custo computacional.

Pode-se dizer, de forma simplificada, que o modelo em questão depende de dois parâmetros encontrados nas redes. Inicialmente deve-se obter um transistor eletricamente equivalente aos transistores que permanecerão estáticos na análise. A largura desse transistor equivalente será utilizada para determinar as correntes e capacitâncias que aparecem nas equações - este procedimento é chamado de compressão. Em seguida são encontrados os valores de tensão em todos os nodos da rede. Com essas informações já é possível estimar o atraso de uma porta através do modelo. Caso a porta seja composta por múltiplos estágios, a mesma deve ser segmentada em unidades com pinos de entrada e de saída identificados, para que elas possam ser analisadas sequencialmente a fim de resolver as dependências.

A compressão é uma série de associações de pares de transistores, dados os nodos terminais que delimitam o conjunto que será colapsado. Se dois transistores estiverem em série o equivalente deles é definido pela razão do produto pela soma da largura dos *gates*. Caso eles estejam em paralelo, simplesmente somam-se as larguras para gerar um equivalente. Essas duas formas de associação são repetidas até que só haja um transistor entre os terminais da compressão.

Os valores nodais da rede podem ser: alta impedância, zero-forte, zero-fraco, um-forte ou um-fraco. Sendo que o nodo VDD vale um-forte e o *ground* vale zero-forte, é feita uma propagação a partir desses pontos ao longo da rede. Se um transistor NMOS conduzir um zero-forte, o valor propagado segue sendo zero-forte, mas se o conduzido for um-forte ele propaga um-fraco. No caso do PMOS, caso o conduzido seja um-forte, é propagado um-forte, e se for zero-forte o propagado é zero-fraco. Feitas as análises partindo de VDD e de *ground*, podemos mesclar os resultados para gerar a rede completa de valores nodais e analisar a existência de curtos-circuitos.

O próximo passo do método é aplicar os dados obtidos a uma implementação do modelo analítico. Assim, será possível caracterizar grandes conjuntos de portas e testar a precisão do modelo de forma automatizada. Os testes feitos com os códigos já desenvolvidos demonstraram uma eficiência bastante promissora com resultados quase imediatos.

[1] Marranghello, Felipe S., Andre I. Reis, and Renato P. Ribas. "Delay model for static CMOS complex gates." Integrated Circuits and Systems Design (SBCCI), 2013 26th Symposium on. IEEE, 2013.