

## OTIMIZAÇÃO DA TÉCNICA DE FOLDING DOS TRANSISTORES NA FERRAMENTA DE CAD ASTRAN

**GUSTAVO SMANIOTTO<sup>1</sup>; MATHEUS MOREIRA<sup>2</sup>; ADRIEL ZIESEMER JR.<sup>3</sup>;**  
**FELIPE MARQUES<sup>1</sup>; LEOMAR DA ROSA JR.<sup>1</sup>**

<sup>1</sup>*Universidade Federal de Pelotas – {ghsmaniotto, felipem, leomarjr}@inf.ufpel.edu.br* 1

<sup>2</sup>*Pontifícia Universidade Católica do Rio Grande do Sul – matheus.moreira@acad.pucrs.br* 2

<sup>3</sup>*Instituto Federal Sul-rio-grandense – adriel.ziesemer@canoas.ifrs.edu.br* 3

### 1. INTRODUÇÃO

Dispositivos eletrônicos miniaturizaram-se com o avanço da tecnologia e, de certa forma, tornaram-se indissociáveis à vida dos seres humanos. Nesta era “mobile”, onde a microeletrônica ganha importância, são raras as atividades que não utilizam algo tecnológico, pois em alguns casos o uso da tecnologia é “transparente” ao usuário. Para obter dispositivos relativamente pequenos, rápidos, e que possuam uma bateria duradoura, é necessário produzir circuitos otimizados em aspectos, como: área, atraso e consumo de energia. Neste contexto, destacam-se os circuitos integrados de aplicação específica (ou ASICs), um dos responsáveis pelo avanço da microeletrônica. Porém, projetar um ASIC otimizado não é uma tarefa trivial, ainda mais quando se trata de um processo de concepção automático.

Devido a dificuldade de projetar um ASIC, são utilizadas ferramentas de CAD (*Circuit Aided Digital*) ou EDA (*Eletronical Digital Aided*) para auxiliar nesta tarefa. Estas ferramentas percorrem um fluxo formado por várias etapas complexas, podendo dividí-las em duas principais: a síntese lógica e a síntese física. No fluxo tradicional, baseado em células padrão, a síntese lógica é responsável por transformar uma descrição RTL (*Register Transistor Level*) em uma descrição estrutural equivalente, utilizando portas lógicas básicas contidas em uma determinada biblioteca de células. Já a síntese física transforma esta descrição estrutural no layout do circuito, um desenho geométrico contendo as representações dos transistores e suas interconexões.

A maioria das ferramentas de CAD utiliza um fluxo para concepção de ASICs baseado em células padrão (*Standard Cell*). Neste tipo de abordagem, o circuito final é construído de acordo com uma biblioteca de células pré-projetadas, testadas e validadas por um projetista experiente. Porém, esta abordagem limita a otimização do circuito, visto que o projetista fica restrito a quantidade de células contidas em cada biblioteca (GUAN, 1996), quantidade que, na maioria das bibliotecas é reduzida. Além dos problemas com otimização, o fluxo de células padrão possui restrições quanto a adoção de novas tecnologias. Isto por que, para cada nova tecnologia é necessário adaptar toda biblioteca de células de forma que ela obedeça às novas regras. Como as células para a biblioteca são projetadas manualmente, adequar a biblioteca para a nova tecnologia demanda grande quantidade de tempo, algo não desejado atualmente.

Visando contornar alguns problemas relacionados ao fluxo *standard cell*, uma ferramenta automática de síntese física de circuitos VLSI (*Very Large Scale Integration*) foi desenvolvida. Chamada ASTRAN (*Automatic Synthesis of Transistor Network*) (ASTRAN, 2015), esta ferramenta acadêmica e de código aberto foi construída objetivando ser uma alternativa ao fluxo tradicional da concepção de ASICs. Com esta ferramenta, é possível otimizar as células para problemas específicos, como por exemplo: circuitos assíncronos

(MOREIRA, 2014), redução de consumo estático, NBTI (*Negative Bias Temperature Instability*), entre outros.

Uma etapa chave no fluxo do ASTRAN é a geração automática de células de redes de transistores. Esta é uma das etapas responsáveis pela diferenciação do ASTRAN para o fluxo baseado em células padrão. Isso porque, no ASTRAN, as células são geradas de acordo com a demanda do projeto, não necessitando de uma biblioteca de células. Durante a geração de células, ganha destaque a técnica de *folding* dos transistores, principal encarregada por manter a altura (definida nas regras de projeto) “padronizada” das células desenvolvidas.

A técnica de *folding* consiste em “quebrar” os transistores que excedem um determinado limite de largura, em transistores menores, conectados em paralelo. A largura dessas quebras (ou *legs*) serão menores ou iguais a altura da área de difusão e que, somadas, deverão resultar na largura do transistor original. Aplica-se o *folding* com o objetivo de reduzir e padronizar a altura das células geradas pela ferramenta, em troca de um pequeno acréscimo na largura destas células. Aplicando esta técnica, temos que a conexão das células para a construção do circuito final é facilitada, visto que a altura de todas as células será igual. Alguns trabalhos que abordam esta técnica podem ser vistos em CORTADELLA (2013), CHENG (2001), KIM (1997) e GUPTA (1998).

No ASTRAN, o *folding* é aplicado considerando cada transistor individualmente, contudo, melhores resultados podem ser obtidos quando se aplica esta técnica simultaneamente em todos transistores de uma associação série (ZIESEMER, 2014). Realizar o *folding* analisando as séries de transistores permite reduzir o número de interconexões e, consequentemente, de contatos nas células, tornando possível a redução de área desta. Baseado nesta alternativa, o presente trabalho apresenta uma proposta de otimização para a técnica de *folding* na ferramenta ASTRAN.

## 2. METODOLOGIA

O ASTRAN utiliza uma estrutura de grafos para representar a rede de transistores de cada célula. Dessa maneira, o método proposto utiliza algoritmos aplicados a este tipo de estrutura de dados. O foco deste trabalho está voltado para a associação em série dos transistores, visto que desejamos aplicar o *folding* em todo o arranjo em série e não nos transistores individualmente, como é feito atualmente no ASTRAN. A metodologia proposta divide-se em duas etapas principais:

Na primeira etapa, é necessário obter todas as associações de transistores em série no grafo. Para esta tarefa é utilizado o conceito de compactação de arestas (POSSANI, 2013). Com este método, percorre-se todo o grafo da rede de transistores da célula, procurando as associações série dos transistores, armazenando todos arranjos encontrados em uma matriz bidimensional. Nesta matriz, cada linha representa uma série de transistores e cada coluna representa um transistor da série. Em cada linha, os transistores estão ordenados de acordo com sua ordem na série encontrada.

Já a segunda etapa é responsável por aplicar o *folding* em toda a série que necessitar. Para isso, percorre-se a matriz bidimensional linha a linha, analisando se os transistores contidos em cada série, excedem o tamanho da área de difusão estabelecido nas regras de projeto. Caso hajam transistores mais largos que a área de difusão, é necessário obter a quantidade de réplicas (“quebras”) a serem feitas desta série. Esta quantidade é obtida com a divisão da largura do maior transistor da série pela altura da área de difusão da célula. Em seguida, a série é quebrada de acordo com o número obtido anteriormente. A largura de cada

transistor contido nestas quebras, será obtida de acordo com a divisão da largura do transistor original da série, pelo número de quebras realizadas. Com as séries quebradas, é necessário conectá-las em paralelo para manter a lógica do circuito. Após todas estas etapas, o método é finalizado.

Para avaliar a qualidade da técnica proposta, foi selecionado um conjunto de células retiradas de MOREIRA (2014), POSSANI (2013) e ASTRAN (2015). Para comparar as duas técnicas, cada célula foi gerada duas vezes pelo ASTRAN, uma considerando o atual método de *folding* e, a outra, o método proposto. Dois aspectos foram considerados para cada célula: a área e o número de difusões com contatos. A área foi calculada utilizando a ferramenta SightGL, já o número de difusões com contatos, foi obtido através da contagem das difusões (tanto dos transistores PMOS, quanto dos NMOS) que possuíam contatos. Estes resultados são apresentados e discutidos na próxima seção.

### 3. RESULTADOS E DISCUSSÃO

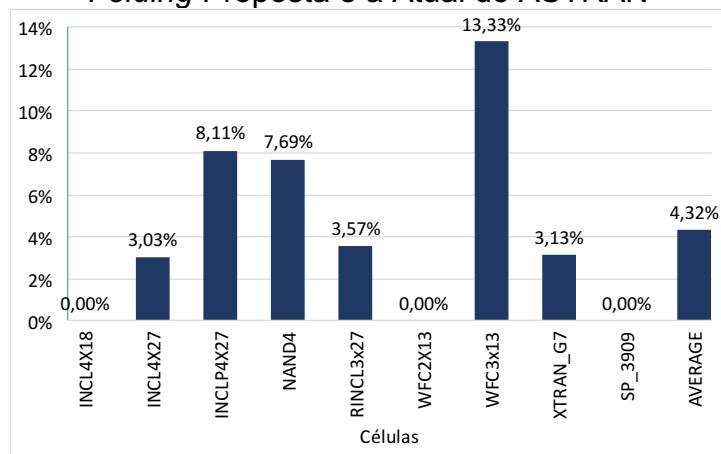
Os dados coletados dos experimentos estão representados na Tabela 1 e na Figura 1. Na Tabela 1 é comparado a quantidade de difusão com contatos, obtidas com os dois métodos e, na Figura 1, é demonstrado a porcentagem de redução da área das células comparando as duas metodologias. Tanto na Tabela 1, quanto na Figura 1, é possível notar que o método proposto apresentou melhores resultados quando comparado com o método atual implementado na ferramenta, gerando células com uma redução de 4,32% em média.

O ASTRAN é capaz de gerar células com um acréscimo de 3,6% na área destas, quando comparado com células de bibliotecas padrão (ZIESEMER, 2014). Assim, com esta otimização, torna-se possível gerar células com áreas idênticas às áreas das células geradas manualmente, contidas nas bibliotecas de células.

Tabela 1. Quantidade de Difusões com Contatos Comparando a Técnica de *Folding* Proposta e a Atual do ASTRAN

| Células               |                  | INCL 4X18 | INCL 4X27 | INCLP 4X27 | NAND 4 | RINCL 3X27 | WFC 2X13 | WFC 3X13 | XTRAN G7 | SP 3909 |
|-----------------------|------------------|-----------|-----------|------------|--------|------------|----------|----------|----------|---------|
| Difusão com contatos  | Folding Atual    | 37        | 45        | 53         | 20     | 39         | 14       | 19       | 41       | 36      |
|                       | Folding Proposto | 29        | 31        | 35         | 13     | 24         | 12       | 12       | 36       | 34      |
| Diferença entre ambos |                  | 8         | 14        | 18         | 7      | 15         | 2        | 7        | 5        | 2       |

Figura 1. Porcentagem de Redução da Área Comparando a Metodologia de *Folding* Proposta e a Atual do ASTRAN



Esta otimização é obtida pois o método proposto reduz a quantidade de interconexões entre os transistores da célula. Dessa forma, ao reduzir as interconexões, o número de contatos nas áreas de difusão também é reduzido, possibilitando a otimização da área da célula.

#### 4. CONCLUSÕES

Este trabalho apresentou uma otimização para a técnica de *folding* da ferramenta ASTRAN. Os resultados preliminares demonstram que o método proposto permite à ferramenta gerar células mais otimizadas quando comparadas com o atual método implementado. O ganho obtido nos casos analisados é notável, porém modesto. Com isso, futuramente, esta técnica será analisada em um conjunto maior de células, contendo células de vários tamanhos. Com estas novas avaliações, são esperados resultados ainda melhores do que os obtidos até o momento.

Como trabalhos futuros, planeja-se incorporar este método a ferramenta ASTRAN, além de realizar a caracterização elétrica das células geradas com este método. Esta caracterização analisará características, com: atraso, consumo de energia estático, potência, capacidade, entre outros. Assim, será possível comparar e concluir de uma forma precisa, se o método apresentado é realmente melhor que o atual.

#### 5. REFERÊNCIAS BIBLIOGRÁFICAS

GUAN, B.; SECHEN, C. Large Standard Cell Libraries and Their Impact on Layout Area and Circuit Performance. **International Conference on Computer Design, Vlsi in Computers and Processors**. Washington, DC, USA. p.378-383, 1996.

ZIESEMER JR., A. M. **Síntese Automática do Leioute de Redes de Trasistores**. 2014. Tese de Doutorado, PPGMICRO, UFRGS.

ASTRAN. **Astran – Automatic synthesis transistor networks**. Acesso em 1 jun. 2015. Online. Disponível em: <http://aziesemer.github.io/astran/>

MOREIRA, M. T., et al. Automated Synthesis of Cell Libraries for Asynchronous Circuits. **27th Symposium on Integrated Circuits and Systems Design**. p. 1-6, 2014.

CORTADELLA, J. Area-Optimal Transistor Folding 1-D Gridded Cell Design. **IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems**. v. 32, n. 11, p. 1-4, 2013.

CHENG, E. Y. C.; SAHNI, S. A fast algorithm for transistor folding. **VLSI Design**. v. 12, n. 1, p. 53-60, 2001.

KIM, J.; KANG, S. M. An efficient transistor folding algorithm for row-based CMOS layout design. **Proc. ACM/IEEE Design Autom. Conf.** p. 456-459, 1997.

GUPTA, A.; HAYES, J. P. Optimal 2-D cell layout with integrated transistor folding. **Proc. IEEE/ACM Int. Conf. Comput.-Aided Design**. p. 128-135, 1998.

POSSANI V., et al. Improving the methodology to build non-series-parallel transistor arrangements. **Symposium on Integrated Circuits and Systems Design**. p. 1-6, 2013.