# ELECTRICAL EXTRACTION TECHNIQUES FOR VLSI CIRCUITS

*Marcos Rafael Boschetti, Ricardo Augusto da Luz Reis*

Instituto de Informática – Universidade Federal do Rio Grande do Sul - UFRGS
{marcosrb, reis}@inf.ufrgs.br

**ABSTRACT**

This work presents an electrical extractor implemented using Java and the main aspects of its development. From a layout description (CIF), the tool generates a Spice electrical description ready to be compared or simulated. During the extraction process there are two different kinds of elements that the software identifies: devices and parasitic elements. In this context, a device is an element intentionally created by the composition of the available layers (for example transistors). Parasitic elements are electrical elements (typically resistances and capacitances) that represent undesired effects.

## INTRODUCTION

The performance and complexity of integrated circuits has been increasing exponentially due to device and interconnection scaling, new microarchitectures and design methodologies. This great complexity demands a rigid validation of all hardware blocks and cells that are present in the circuit. In a complete design flow a set of specialized tools for circuit analysis and test are used in every design stage. In our group it was already developed two electrical extractors, Extribo [1] [2] and Lasca [3] [4]. The electrical extracted netlist can also be used as input to a logical extractor like the Extralo [5]. The logical netlist obtained by the logical extractor can be compared with the original logical netlist used in the synthesis flow. This new electrical extractor has the characteristic that it is build using Java that allows running it in any platform. It is also more adapted to be included in the CAVE framework [6] [7].

## ELECTRICAL EXTRACTION

The extraction process can also be divided in two parts: connectivity analysis and elements extraction. The connectivity analysis goal is to build an efficient data structure that represents all the rectangles of the circuit and their interconnections. The routines for connectivity extraction are performed in different levels: local and global. Local analysis (Figure 1a) means that only the rectangles in the same layer are considered. In a similar way, in the global analysis (Figure 1b) all rectangles of the layout description are considered. As result of this process, every connected rectangle has the same connectivity label.

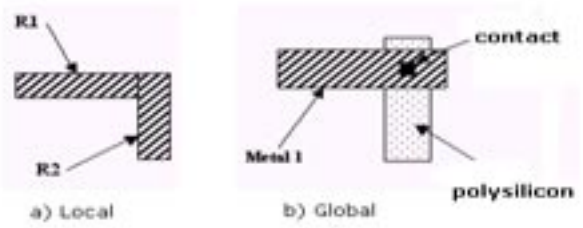


Figure 1: Connectivity Extraction

The elements extraction is the most time-consuming task. It is possible to identify transistors considering only the crossing of the layers, but a complete extraction must be able to provide some device parameters as transistor's channel width (W) and length (L). However, a transistor can be implemented using different topologies, to achieve precise results, generic source and drain areas extraction algorithms were implemented using sets theory operations like union, exclusion and intersection (Figure 2). Mathematically, integrated circuits masks description can be considered as a set of points in a flat surface. The most important operations are intersection and exclusion. Intersection is useful whenever we want to know if two rectangles are overlapped. Nevertheless, exclusion is a more critical operation because it is necessary in complex "cutting" algorithms. The drain and source areas of the transistor are the result of an exclusion operation between polysilicon and diffusion rectangles, therefore exclusion means the removal of all points that belongs to a rectangle B from a rectangle A.

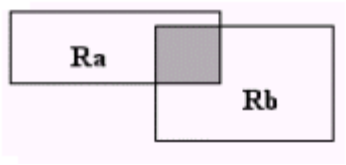


Figure 2- Intersection Operation

A huge number of algorithms about polygons geometry management can be found in the literature. The technique chosen here for internal data organization is an ordered linked list. The linked list has linear space complexity O(N) and it is well-suited for the operations performed by the routines. Besides, the characteristics of an ordered list can be exploited in order to get significant improvements in the tool performance. The information stored about the position of a rectangle are the coordinates (x,y) of its lower left corner and upper right corner (ordered by the left corner x-axis). So, given a rectangle R with coordinates $[(x_{1r}, y_{1r}) (x_{2r}, y_{2r})]$, some algorithms need to perform rectangle comparisons (connectivity extraction for example), in relation to R the routines need only to analyze rectangles where $x_{2n} >= x_{1r}$ and $x_{1n} <= x_{2r}$. Thus, the used data structure allows considerable reductions in the number of necessary operations.

The software was implemented using Java language giving flexibility to users that can run the tool in any computer. The software was built to be seen as a complete application and as a collection of geometry extraction classes. Some generic classes represent abstract entities like transistors and rectangles while other classes provide the general libraries for the topological operations. This separation makes easier the construction of geometry-based CAD tools using Java heritance mechanisms. We plan to improve the mentioned techniques to achieve better results in terms of processing time. Besides algorithms improvement, we intend to add some routines for interconnections parameters extraction. Nowadays this kind of data is very important, the scaling of the transistors due to new submicron technologies makes interconnections delay a significant factor in the circuit performance, being as or more important than active devices delays.

## REFERENCES

[1] STEMMER, Marcos Augusto; REIS, Ricardo Augusto da Luz. **Um Algoritmo para Calcular Resistências em Circuitos Integrados.** In: CONGRESSO da SOCIEDADE BRASILEIRA de MICROELETRÔNICA, 4., Porto Alegre, 12-14 jul. 1989. Anais. Porto Alegre: SBMICRO/UFRGS, 1989. v.2 p. 575-86.

[2] STEMMER, Marcos .Augusto; REIS, Ricardo Augusto da Luz. **EXTRIBO: um Extrator de Circuitos.** In: SIMPÓSIO BRASILEIRO de CONCEPÇÃO de CIRCUITOS INTEGRADOS, 4, Rio de Janeiro, 12- 14 abr. 1989, Anais. Rio de Janeiro, SBC, 1989. p. 1-9.

[3] FERREIRA, Fábio Klein; MORAES, Fernando.; REIS, Ricardo **Fast Interconnect Parasitic Extraction in Deep Submicron Using Bin-Based Algorithm**, 43th IEEE Midwest Symposium on Circuits and Systems, IEEE Circuits and Systems Society, Michigan, USA, 9-12 Agosto 2000

[4] FERREIRA, Fábio Klein; MORAES, Fernando; REIS, Ricardo. **LASCA - Interconnect Parasitic Extraction Tool for Deep-Submicron IC Design** In: Symposium on Integrated Circuits and Systems Design, 13, Manaus, AM, 18 a 24 de setembro 2000. Anais. IEEE Computer Society, 2000. Pg. 327- 332. ISBN: 0-7695-0843-X

[5] MORAES, Fernando; REIS, Ricardo. **EXTRALO - Extrator Lógico.** In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 5. Ouro Preto, 24-26 out. 1990. Anais. Ouro Preto: SBC, 1990. 334p. p. 167-176.

[6] INDRUSIAK, Leandro; REIS, Ricardo. **From a Hyperdocument-Centric to an Object-Oriented Approach for the Cave Project**. In: Symposium on Integrated Circuits and Systems Design, 13, Manaus, AM, 18 a 24 de setembro 2000. Anais. IEEE Computer Society, 2000. Pg. 125-130. ISBN: 0-7695-0843-X

[7] INDRUSIAK, L., BECKER, J., GLESNER, M., REIS, R. A. L.. **Distributed Collaborative Design Over Cave Framework** In: SOC Design Methodologies. Ed.Londres : Kluwer, 2002