LASCA – Interconnect Parasitic Extraction Tool for Deep-Submicron IC Design

Fabio Klein Ferreira
UFRGS – BRAZIL
fklein@inf.ufrgs.br

Fernando Moraes
PUCRS – BRAZIL
moraes@inf.pucrs.br

Ricardo Reis
UFRGS - BRAZIL
reis@inf.ufrgs.br

Abstract

In this paper a fast capacitance and resistance extraction tool (wire extractor) is presented. Five models are implemented: ground capacitances, ground plus coupling capacitances and RC models (L, π and T lumped). The designer can choose one of these models according some criteria, like accuracy, CPU time and circuit complexity. Comparisons between our wire extractor and Diva extractor (Cadence Design Systems) for C models, give an average difference of only 5% in the final delay of some benchmarks, being up to 30 times faster than Diva.

1. INTRODUCTION

With advancements in integrated circuits process technology, feature dimensions below 0.35 microns are currently used by the semiconductor industry. As the physical size decreases, the delay of electrical signals traveling in the interconnections is equivalent or greater than the gate delay.

For example, the interconnect capacitances between aluminum and silicon dioxide dielectric represents 50 percent of the total delay in a 0.25 μm technology. In a 0.18 μm technology this capacitance can represents up to 70 percent, and it is expected to contribute up to 80 percent in a 0.15 μm technology [1].

The parasitic capacitance of each net has two components: area and perimeter. The relationship between wire height and width increases in deep-submicron technology (1.8 for 0.25 μm technology and will reach 2.7 for 0.07 μm technology) [1], resulting in a major contribution for perimeter capacitances. Also, the number of interconnect layers is increasing to 6-7 layers. These two facts make the coupling capacitances as important as ground capacitances.

Currently, analysis and verification procedures in the design flow consume the same amount of time as synthesis, placement and routing. As the number of parasitic elements to consider during design verification increases in deep-submicron technologies (coupling capacitances and more metal layers), fast and accurate estimation tools must be developed to keep a reasonable CPU time spent by CAD tools (minutes or few hours instead days or weeks).

Usually, design verification methods use 2D extraction, which is highly inaccurate, when used in deep-submicron technology. The two dimensional parasitic extraction ignores all 3D details, assuming the geometry is uniform in one dimension, generally the signal propagation direction.

There are many 3D numerical techniques to determine the parasitic capacitances, such as finite difference [2], finite elements, boundary element methods [3] and multipole algorithms [4]. These methods are too compute intensive to be applied in CAD for VLSI design.

An alternative approach, the 2½D method presented in [5][6], using fast empirical models [7][8], represents a good trade-off between CPU time and accuracy.

Capacitance is only one of the elements needing to be extracted from the layout. To correctly model interconnections, we need to extract RC elements. When resistances and capacitances are extracted (RC elements), it is possible to accurately predict delay, skew and noise in all signals.

We present in this paper a fast and accurate wire extraction tool, called LASCA. We take the physical description of each net, computing its resistances, coupling capacitances and substrate capacitances. Using a bin-based algorithm [9] to extract the connectivity, empiric formulation and a simple 2½D methodology, we provide a good trade-off between accuracy and efficiency when compared to traditional layout extraction (2D extraction).

Pre-characterization is no more sufficient to predict delay at higher abstraction levels. Thus, tools as LASCA, are important to predict the parasitic elements, and can be used closely with other tools, e.g., layout synthesis tools.

This paper is organized as follows. Section 2 presents the connectivity extraction algorithm and an analysis of the algorithm complexity. Section 3 introduces the parasitic extraction method. Section 4 presents preliminary results, and section 5 our conclusions and future work.

0-7695-0843-X/00 $10.00 © 2000 IEEE

2. CONNECTIVITY EXTRACTION

The connectivity extraction is the first task executed by an electrical extractor. This task finds all interconnected polygons, numbering the equipotential regions.

The connectivity extraction in LASCA is done by a Bin-Based algorithm [9]. The circuit layout is mapped over a virtual grid in the bin-based algorithm. This grid splits the circuit in rectangles, called bins. This set of bins composes a NxM bidimensional matrix. The space complexity is O(bn), where b is the total number of bins, and n is the number of polygons.

Each bin will contain all polygons intersecting it, storing them in linked lists. If a polygon intersects more than one bin, it will be stored in each bin it intersects. To reduce the memory usage and to avoid redundant information, only pointers to polygons are stored into the bins. Thus, there are two main data structures: (i) the list of all polygons, (ii) the bin matrix, storing polygon pointers.

After the data structure creation, it is verified the connectivity between all polygons. This task is handled as follows: each polygon not analyzed starts a new net. This starting polygon is the first component of a new net. All non-visited polygons connected to the starting polygon are set as "new starting polygons", and added to the net. Recursively, all "new starting polygons" are used as "new polygons", until no more polygon can be added to the net.

The bin-based algorithm has better performance than linked lists, since each bin contains a small number of polygons, and consequently, the search procedures are executed in short lists. The performance of this algorithm is a function of the bin size. If the bins are too large, the linked lists will contain a huge number of polygons, increasing CPU processing time. If the bins are small, the memory usage is too important.

From our experimental data, using a 0.25 μm technology, we fix the bin size as a 2 μm x 2 μm square. This size result in short linked lists, containing 1 to 20 polygons, and consequently in a small CPU time for wire extraction.

To improve the performance of the algorithm, the matrix size (virtual grid) is a function of the circuit size. If the matrix size is fixed, an important number of empty bins occur for small circuits or we can have bins containing hundreds of polygons in large circuits, decreasing the algorithm performance.

Therefore, our bin-based algorithm implementation has a fixed bin size, 2μm x 2μm, and a flexible virtual grid.

Figure 1 presents the polygon distribution histogram for the C7552 circuit (ISCAS85 benchmarks). The C7552 benchmark was generated using the TROPIC [10] layout synthesis tool, with three metal layers, in a 0.25 μm technology. It has 14736 transistors and 265563 polygons. The resulting circuit size is 933.8 μm x 364.2 μm, corresponding to a matrix size with 466 x 182 units (84812 bins).

From Figure 1/Table 1, we observe that 86.8% of bins has less than 16 polygons, due to the choice of the bin dimension (2μm x 2μm square) and a matrix size proportional to the circuit size. Then, when we need to find connected polygons, we must search in a list containing less than 16 polygons (typical case), instead of all 265563 polygons.

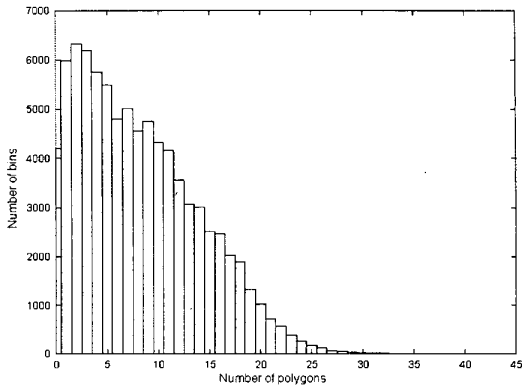

Figure 1. Polygon distribution histogram - C7552 benchmark

Table 1. Polygons distribution - C7552 benchmark (total number of bins = 466 x 182 = 84812)

Number of polygons	Number of bins	% (bins/ total nb bins)
0 (bins empty)	4199	5.0%
1-5	29716	35.0%
6-10	23399	27.6%
11-15	16282	19.2%
≥ 16	11216	13.2%

The second analysis concerns the correctness of the bin size, i.e., the average number of bins used by each polygon. In this benchmark, 97% of nets are using 1-5 bins. The other nets are long wires, using up to 100 bins (like clock and reset nets).

As showed in Figure 1/Table 1, to find connected polygons in one bin we search in a list containing less than 16 polygons (typical case). Consequently, to find all connected polygons to a given polygon, the search space is 5 times 16, or 80 polygons instead all layout polygons. Observe that this is the typical search space, there are long nets where the search space can be very huge, but these long nets represents less the 3% off all nets.

In the Figure 2 we have the CPU time as a function of number of polygons. This CPU time consider the

connectivity analysis and capacitance extraction. These results were obtained from ISCAS85 benchmarks, on a SUN Ultra Sparc10. The solid lines represents $O(n)$ and $O(\sqrt{n})$ complexity for 100 polygons/second. They not indicate fitting, they are inserted to indicate lower and upper bounds of the algorithm. These results confirm the efficiency of bin-based algorithm implementation.

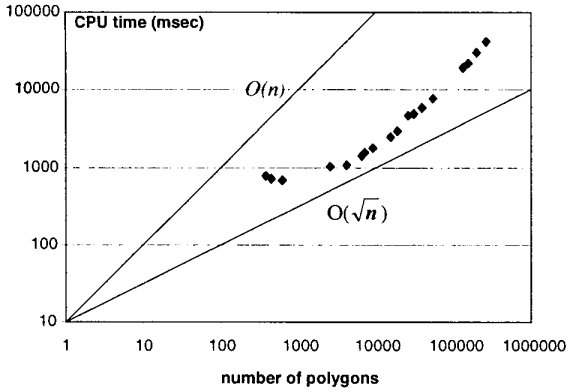

Figure 2. Extraction time *versus* number of polygons, on a SUN Ultra Sparc 10, $O(n)$ and $O(\sqrt{n})$ complexity for 100 polygons/second

3. PARASITIC EXTRACTION

Interconnect capacitance in each node in a circuit is calculated using the model shown in Figure 3. It consists of two conduction layers over the substrate, considered as a reference plane (ground plane).

There are three capacitance components at any node [5]:

- Overlap capacitance (C_{over}) - due the overlap between two conductors in different planes. They are C_{21a} and C_{23a} in the Figure 3.
- Lateral capacitance (C_{lat}) - is the capacitance between two conductors in the same plane. In the Figure 3 is C_{22lat}.
- Fringing capacitance (C_{fr}) - due the coupling between two conductors of different planes. They are C_{21fr} and C_{23fr} in the Figure 3.

The capacitances Cij (i,j=1...n, where n is the number of conductors) are calculate by an empiric formulation. It was used the 3D field solver FastCap [4] to validate the equations.

It was also used the simple and accurate 2½D methodology described in [6]. It is based on experiments

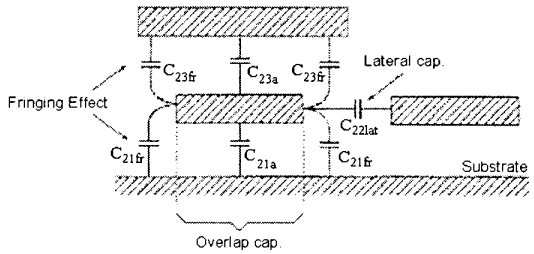

Figure 3. Capacitance model

with a 3D field solver over 0.50 μm, 0.35 μm and 0.18μm process. Briefly, for each connection in the layer i, we should analyze the immediate neighbor in the same layer, all crossunders in the layer i-1 and all crossovers in the layer i+1, treating the layers i±2 as ground planes.

3.1. Intrinsic Capacitance

The intrinsic capacitance is the capacitance between one conductor layer and the ground plane. It has two components: overlap and fringing capacitances.

Two parallel plates model the overlap capacitance. It is calculated using the traditional formulation based in the overlap area:

$$C_{over}=C_{area}.W.L$$

where C_{area} is capacitance per unit area (fF/μm^2), and W.L is the overlap area (μm^2).

The fringing capacitance is due to the edge of one conductor and the surface of the other one (in this case, the ground plane). It is calculated by:

$$C_{fr}=2.C_{length}.L$$

where C_{lenght} is the capacitance per unit length (fF/μm).

Thus, the intrinsic capacitance is the sum of these two components.

$$C_{int}=(C_{area}.W+2.C_{length}).L \qquad (1)$$

A simple structure was simulated using the FastCap [4] field solver, in order to study the accuracy of Equation 1. We assume a wire over a ground plane. The wire length is 20.Wmin, and the ground is a 40Wmin x 40Wmin plane, where Wmin is the minimum wire width for a given layer.

Table 2. Intrinsic capacitance results, in fF

	Poly	Metal1	Metal2	Metal3
FastCap	0.568	0.7441	1.155	0.9978
Equation 1	0.5683	0.7318	1.1277	0.9914
Error (%)	-	1.65	2.36	0.6

Table 2 shows the results calculated by equation (1) and the simulation in the FastCap. All parameters (wire width, wire thickness and distance to the ground plane) were obtained from a 0.25 μm technology.

The equation (1) presents a small error, less than 2.5%. It should be noted that for metal2 and metal3 layers, it was considered as the ground plane, the polysilicon and metal1 layers, respectively.

3.2. Coupling Capacitance in the Same Layer

In deep-submicron processes, the ratio between wire thickness and wire separation has increased. Consequently, coupling capacitance between wires in the same layer becomes more important than micron in technologies. The lateral coupling capacitance cause parasitic noise effects, which may affect the circuit performance in several ways: parasitic glitch propagation, gate delay increase and noisy signals inside analog cells.

The lateral coupling capacitance is a function, basically, of two factors: wire thickness and the space between wires.

The wire thickness is a function of the technology and the layer considered. The separation between wires is the main factor, depending on the layout. As the space between two wires can have a large variation, we can not use the capacitance given by the technology rules, since it is restrict to the minimum distance between wires. Thus, if we use the capacitance from the technology rules, we have a large error.

For example, in deep-submicron process, the foundries set the minimum area usage for each metal layer. This value is normally fixed to 30%, for uniformity of etch rate or CMP planarization. The maximum area usage is around 50%, when the distance between two wires is equal to the minimum distance rule.

In the LASCA, we use the empirical formulation described in [8]. Two cases can happen: just one ground plane or two ground planes, where the second plane is the i+2 plane.

Again, to validate these formulations, we simulated a test structure in the FastCap. The test structure is composed by two parallel wires in the same layer, each one has a length equal to 20.Wmin, and one or two ground planes (40Wmin x 40Wmin).

Results are presented in the Table 3. The "S" parameter in Table 3 represents the space between two wires (S=1 is the minimum space in the technology rules). All parameters used in the FastCap simulation are obtained from a 0.25μm technology, metal1 layer. Observe that the presence of a second ground plane reduce the lateral capacitance, due the interactions between the wires and this second ground plane.

The equations described in [8] have errors below than 2.5% for one ground plane, and below 10% for two ground planes. For a three-metal layer technology, the only layers having two ground planes are polysilicon (substrate and metal2) and metal1 (substrate and metal3).

Table 3. Lateral capacitance results, in fF

		S=1	S=1.5	S=2	S=3	S=4
round lane	FastCap	0.7076	0.5072	0.3954	0.2807	0.2095
	Eq. [8]	0.7127	0.5151	0.4052	0.2821	0.2129
	Error(%)	+0.7	+1.55	+2.47	+0.5	+1.62
round lanes	FastCap	0.679	0.4758	0.3619	0.2328	0.1596
	Eq. [8]	0.6183	0.448	0.3534	0.2423	0.1754
	Error(%)	-8.93	-5.84	-2.35	+4.08	+9.89

3.3. Crossover Capacitance

The crossover capacitance is only calculated if there is a common area between two wires in different planes. This common area is extracted by the AND logic operator between the two wires.

As the intrinsic capacitance, the crossover capacitance (Ccross) is modeled by two parallel plates, and is calculated using the follow equation:

$$C_{cross}=C_{area}.W_1.W_2+2.C_{length}.(W_1+W_2) \quad (2)$$

where $W_1.W_2$ is the common area between two wires.

To validate equation (2), the following structure was simulated using the FastCap: two orthogonal wires and two grounds plane, in the i±2 layers. Again, the wire lengths are 20.Wmin and the grounds are 40Wmin x 40Wmin plane, where Wmin is the minimum width of metal1 layer. Results are presented in Table 4. Despite of presenting an error greater than 10% for the crossover between metal1 and metal2, the errors induced by this parasitic capacitance will have minor effects, since the common area between metal1-metal2 is minimal (usually metal1-metal2 are orthogonal layers).

Table 4. Crossover capacitance results, in fF

	Poly-metal1	Metal1-metal2	metal2-metal3
FastCap	0.0658	0.1168	0.1386
Equation (2)	0.0649	0.1046	0.1283
Error (%)	-1.36	-10.44	-7.43

3.4. Resistance

In VLSI circuits there are two types of resistances. The first one is the contact resistance, which occurs at the interface between different materials, for example, metal and silicon. The other one is the resistance for the current

flow in a given material (e.g. polysilicon or aluminum), called wire resistance.

Contact resistance is not a negligible factor in the resistance evaluation. We used a model in which the contact resistance is assumed to be the contact material sheet resistance divided by the contact area. Usually, the technology rules inform these resistances.

The wire resistance of a uniform conductor is calculated by using the follow equation:

$$R=R_{\square}\frac{L}{W}$$

where $R_{\square}$ is the sheet resistance, in $\Omega/\square$, L is the wire length and W is the wire width.

To obtain the resistance of a wire, simply multiply the sheet resistance by the wire length divided by its width. A complete net normally has several polygons, with vertical and horizontal segments. The total resistance of the net is obtained by adding the resistances of all polygons composing this net.

If a more complex parasitic model is considered, like the π model, the nets are broken and new nodes are created. Hence, the distribution of coupling capacitances should be modified. Figure 4 presents the coupling capacitances and resistances for the L, π lumped and T lumped models between two wires. Note that no ground capacitances are presented in this Figure.

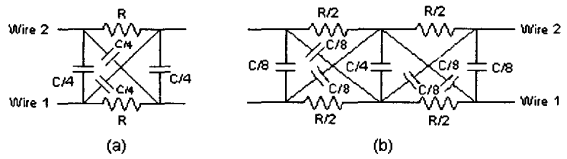

Figure 4. Coupling capacitance in a RC model (a) L or π lumped (b) T lumped

4. RESULTS

To verify the CPU time required extracting the connectivity, we first generate some benchmarks using the TROPIC layout generator [10]. The resulting layout was extracted with our tool, LASCA, and with Diva[1] [11]. Table 5 presents in the first three columns the circuit name, the number of transistors and the number of polygons. In the fourth column it was extracted only the ground capacitances, and the fifth column, the coupling and ground capacitances. The last column presents the CPU time for the Diva extractor. For a circuit containing more than 250,000 polygons we extract all capacitances in less than 45 seconds, while the Diva extractor takes more than 19 minutes (DIVA is a complete layout extractor, extraction not only wires, but also devices, such as transistors and diodes).

[1] Diva is a registered trademark of Cadence Design Systems, Inc.

Table 5. Comparison the CPU time, on a SUN Ultra Sparc 10

Circuit	Xtors	Polygons	CPU time (s)		
			C_{ground}	C + C	Diva
C17	24	386	0.75	0.77	12
C432	150	2564	0.93	1.04	19
C1355	2244	38841	4.01	5.83	132
C3540	7154	129918	10.89	19.55	481
C6288	10112	155885	12.91	21.86	595
C7552	14376	265563	23.78	42.51	1160

In the Figure 5 it is plotted the CPU time for circuits extracted by LASCA, on a SUN Ultra Sparc10. There are no differences for CPU time to RC models. The C model (ground and coupling capacitances) is slower than the Cground model, because it is necessary to find all neighbor polygons.

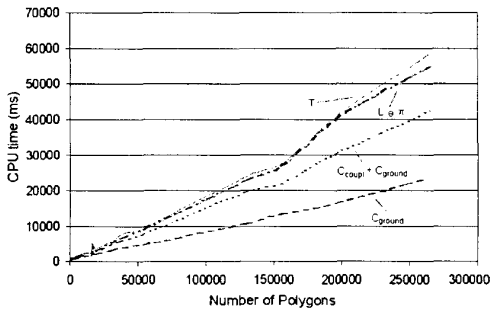

Figure 5. CPU time for circuits extracted by LASCA, on SUN Ultra Sparc10

To verify the accuracy of circuits extracted, we simulated the netlists extracted by LASCA and Diva, using the HSPICE simulator. Table 6 shows in the first four columns the circuit name, the number of transistors, number of nets and the number of polygons. The fifth and sixth column presents the circuit delays with the parasitic capacitances obtained with LASCA, considering first only capacitance to ground (Cground) and after both capacitance to ground and coupling capacitance (Cground+Ccoupling). The last column shows the delay obtained with the capacitances extracted with Diva [11].

In the Figure 6 we plot the results of Table 6 and the results RC models. Comparing the first column (ground capacitance) to Diva (our reference), we get pessimistic delay evaluation, with an average error of 8.25% (worst case: 15%). When considering all capacitances (coupling and ground) the average error is 3.63%, and the maximum

error is small than 10%. These results validate our method for a fast wire extraction. With L lumped model we obtained the upper estimation of delay, while the π and T lumped models presented a closer response.

Table 6. Comparison the time delay for circuits extracted by Diva and LASCA

Circuit	xtors	polygons	Time delay(ns)		
			C_{ground}	C + $C_{coupl.}$	Diva
Adder	28	449	0.3624	0.3847	0.36783
Addergate	40	635	0.4042	0.4304	0.4247
Alu	260	4291	0.985	1.1167	1.1056
Alugate	432	6503	1.0984	1.2401	1.2208
Rip 16bits	448	7270	3.3713	3.7872	3.8571
Cla 16bits	528	9066	2.5539	2.8737	2.6234
Rip 32 bits	896	15503	7.0158	8.2248	8.2730
Cla 32bits	1056	19051	5.3084	6.2971	5.7966

Technology: 0.25μm, 3-metal layers with stacked contacts. Transistors size: w=2μm, l=0.25μm.

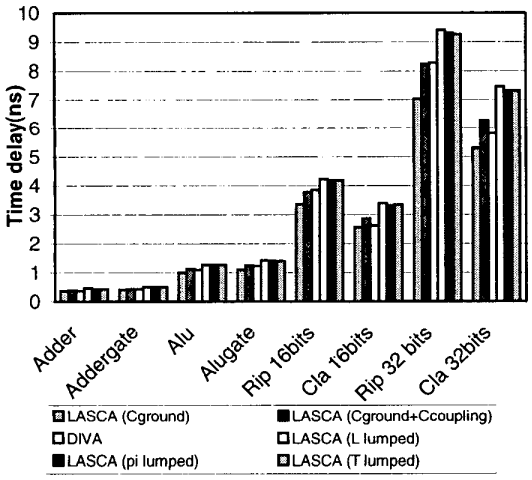

Figure 6. Comparison the delay for circuits extracted by Diva and LASCA

5. CONCLUSIONS

Our primary goal is to show that it is possible to quickly extract the coupling capacitances, ground capacitances and resistances with a reasonable accuracy. Using bin-based algorithm, empiric formulation and a simple 2½D methodology, we developed a fast wire extractor tool, called LASCA. The designer can choose the model to represent the nets (ground capacitances, coupling and ground capacitance, L lumped, π lumped and T lumped), in function of the accuracy required, CPU time and circuit complexity.

The LASCA extractor is integrated within an automatic layout generator, TROPIC [10], resulting in an integrated environment for fast on-the-fly implementation of macro-cells and accurate parasitic evaluation.

As future work, we intend to developer more accurate model for RC extraction, e.g, distributed model.

6. REFERENCES

[1] Semiconductor Industry Association. The National Technology Roadmap for Semiconductor. Available by WWW in http://notes.sematech.org/ntrs/PublNTRS.nsf, 1997

[2] R.H. Uebbing and M. Fukuna. Process Based three-dimensional capacitance – TRICEPS. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems,* CAD-5:215-220, 1986.

[3] Q. Ning, P.M. Dewilde and F.L. Neerhoff. Capacitance coefficients for VLSI multilevel metallization lines. *IEEE Transactions Electron Devices,* ED-34:644-649, 1987.

[4] K. Nabors and J. White. FastCap: A Multipole Accelerated 3D Capacitance Extraction Program. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems,* 10(11):1447-1459, Nov. 1991.

[5] N.D. Arora, K.V. Raol, R. Schumann and L.M. Richardson. Modeling and Extraction of Interconnect Capacitances for Multilayer VLSI Circuits. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems,* 15(1):58-66, Jan. 1996.

[6] J. Cong, A.B. Kahng, D. Noice, N. Shiralli and S.H. Yen. Analysis and Justification of a Simple, Practical 2 1/2D Capacitance Extraction Methodology. UCLA Computer Science Technical Report 970013, 1996.

[7] T. Sakurai. Approximation of wiring delay in Mosfet LSI. *IEEE Journal Electron Devices Letter,* 13(1):32-34, Feb. 1992.

[8] J.H. Chern, J. Huang, L. Arledge, P. C. Li and P. Yang. Multilevel Metal Capacitances Models for CAD Design Synthesis Systems. *IEEE Electron Devices Letters,* 13(1):32-34, Feb. 1992.

[9] N. Sherwani. *Algorithms for VLSI Physical Design Automation.* Kluwer Academic Publisher, 1993.

[10] F. Moraes, M. Robert and D. Auvergne. A Virtual CMOS Library Approach for Fast Layout Synthesis. In: VLSI, 1999.

[11] CADENCE™. *Diva Interactive Verification Reference.* 1998

Acknowledgements: Fernando Moraes gratefully acknowledges the support of the CNPq through grant number 522939/96-1.