A Signal-Flow Driven Automatic Physical Synthesis Methodology for Analog Circuits *

Long Di, Hong Xianlong, and Dong Sheqin

(*Institute of Software, Tsinghua University, Beijing* 100084, *China*)

Abstract: This paper introduces a novel automatic physical synthesis methodology for analog circuits based on the signal-flow analysis. Circuit analysis sub-system adopts the newly advanced methodology, circuit topology analysis, and circuit sensitivity analysis to generate layout constraints and control performance degradations. Considering the heuristic information about signal-flow, complexity of the methodology is less than the pure performance-driven methodology. And then these constraints are implemented in device generation, placement, and routing sub-systems separately, which makes the different constraints be satisfied at most easily implemented stages. Excellent circuit performance obtained by the methodology is demonstrated by practical circuit examples.

Key words: analog layout automation; signal-flow; circuit analysis; device generation; placement; routing
EEACC: 1205
CLC number: TN431.1 **Document code:** A **Article ID:** 0253-4177(2005)04-0689-08

1 Introduction

Nowadays, SOC integrates all of the circuits on one chip, including digital and analog parts. Whereas the physical design of digital circuits is automated to a large extent, the layout of analog circuits is still a manual, time-consuming, and error-prone task. This is mainly due to the continuous nature of analog signals, which causes analog circuit performance to be very sensitive to layout parasitics and mismatch. To obtain high-performance analog circuit layouts we must consider device matches, parasitics, substrate and thermal effects and so on. Performance degradation is subject to occurring due to so many parasitics, which are generally difficult to estimate accurately before the actual layout is completed. If the layout suffers severe performance degradation, it is possible that the circuit does not work. Most of the existing CAD tools for analog layout can be classified as follows.

(1) Special layout systems can automatically synthesize most analog functional modules, such as the synthesis of opamps[1,2], comparators[3], switched-capacitor filters[4], and data converters[5]. But these systems can not cope with arbitrary circuit architecture and full custom layout.

(2) Pre-defined language layout systems heavily rely on the user's expertise. Users must input a layout description file to guide the layout generation[6].

(3) Full custom layout generation based on

* Project supported by National Natural Science Foundation of China (No. 90307005), International Cooperation Project for National Natural Science Foundation of China (No. 60121120706), National Natural Science Foundation of USA (No. CCR-0096383), and National High Technology Research & Development Program of China (No. 2004AA1Z1050)

Long Di male, was born in 1980, PhD candidate. His research interests focus on analog layout automation. Email: longd02@mails.tsinghua.edu.cn

Hong Xianlong male, was born in 1940, professor. His research interests focus on VLSI layout theory and algorithm.

Dong Sheqin male, was born in 1964, associate professor. His research interests focus on VLSI layout algorithm and system integration.

Received 30 April 2004, revised manuscript received 6 December 2004

the minimization of cost function. In KOAN/ANAGRAM[7], area routing and unconstrained placement with abutment capability provide layout with high flexibility and area performance. But they cannot guarantee that the created layout can meet all the specifications.

(4) Performance-driven analog layout generation tools. In Ref. [8], high-level performance constraints are automatically translated into a set of low-level bounds on the layout parameters (i. e. parasitics, outline and location of geometry) by sensitivities analysis, which can effectively control the performance degradation at the stage of layout synthesis. But pure sensitivities analysis has very high algorithmic complex and the algorithm is not convergent in some special situations.

This paper presents a novel methodology for analog layout automation based on the signal-flow analysis. The aim is still to optimize the circuit performance and especially signal-flow circuits are always considered at each stage of layout generation, where the layout generation algorithms for signal-flow circuits are different from those for non-signal-flow circuits. The circuit analysis sub-system adopts not only the method of sensitivities analysis but also the method of circuit topology analysis and circuit feature parameters calculation (i. e. currents and voltages) to obtain the layout constraints. The device generation sub-system can generate 1D and 2D common-centroid structures for transistors with strict match constraints. The placement sub-system adopts determinate symmetrical placement algorithm to optimize layout performance of the signal-flow circuits and annealing simulation to optimize layout of the bias-circuits. The routing sub-system can realize symmetrical routing, multiple-layer routing and variable wire width routing according to the values of currents, which can greatly optimize circuit performance.

The architecture of SDAPS (signal driven analog physical synthesis) system is shown in Fig. 1. The inputs consist of a netlist file, a performance specification file, and a technology file, which describe the whole circuit, the upper limits of the performance degradations, and all process dependent information, respectively. Using different technology description files, SDAPS is able to generate layouts in different CMOS processes. The outputs of SDAPS are the final layouts in CIF format. The whole system consists of four sub-systems, which are circuit analysis sub-system (CA), device generation sub-system (DG), placement sub-system (PL) and routing sub-system (RT). Details of each sub-system will be discussed in the following sections.

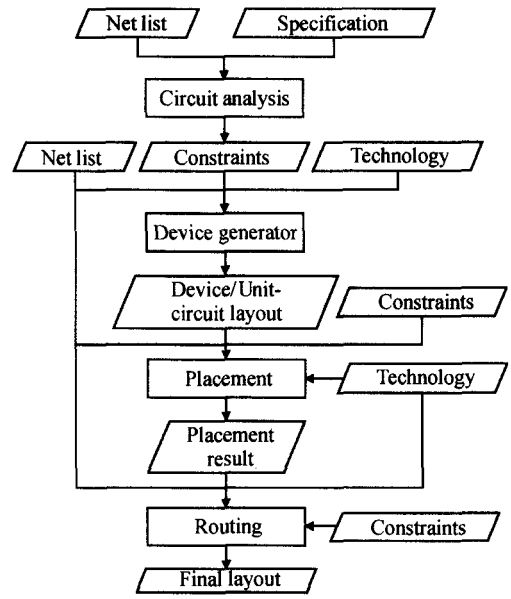

Fig. 1 Architecture of SDAPS

2 Circuit analysis sub-system

Different layouts will introduce different parasitics and mismatches. To guarantee the final layout has approximate performance with the ideal circuit we must have some instructive knowledge on layout to guide the translation from schematics to layouts, which is the task of CA.

Definition 1: performance characteristic p is a real number that quantifies some aspect of the performance of a circuit (i. e. gain-bandwidth, phase-margin and delay).

Definition 2: performance characteristic vector

P is a real vector to describe the whole performance of a circuit. Each dimension of P is a performance characteristic.

Definition 3 :performance specification is a real vector interval in a real vector space that specifies a set of acceptable vectors.

Each performance characteristic is influenced by three kinds of parameters.

(1) Design parameter :circuit designer directly control the values of these parameters (i. e. the width and length of MOS transistor ,the values of capacitance and resistance) .

(2) Process parameter :foundry specifies these values ,which cannot be controlled by circuit designer. Anisotropic process steps and silicon substrate cause asymmetries layout parameter ,so the devices that should have the same electrical properties may be different. The mismatch will cause circuit performance degradation.

(3) Layout parasitics :circuit designer does not intend these parameters ,but they are inevitable in the stage of physical synthesis. Different layouts will generate different layout parasitics. The more little the values are ,the better the performance of layout is. Parasitics mainly include device parasitics and interconnection parasitics.

Among the above three types of parameters , layout designer can control the last two types. To ensure the final layouts satisfy the performance constraints we adopt the following three methods to obtain the layout constraints.

2. 1　Sensitivity analysis

We use $L = [l_1, l_2, \cdots, L_n]^T$ to denote the layout parameter vector; $P = [p_1, p_2, \cdots, p_m]^T$ to denote the performance characteristic vector.

$$K = \begin{bmatrix} k_{11}, k_{12}, \cdots, k_{1n} \\ k_{21}, k_{22}, \cdots, k_{2n} \\ \cdots, \cdots, \cdots, \cdots \\ k_{m1}, k_{m2}, \cdots, k_{mn} \end{bmatrix} \quad (1)$$

$$s_{ij} = \frac{\partial k_{ij}}{\partial l_j} \quad (2)$$

$$S = \begin{bmatrix} s_{11}, s_{12}, \cdots, s_{1n} \\ s_{21}, s_{22}, \cdots, s_{2n} \\ \cdots, \cdots, \cdots, \cdots \\ s_{m1}, s_{m2}, \cdots, s_{mn} \end{bmatrix} \quad (3)$$

$$s_{ij} = \frac{\Delta p_i}{\Delta l_j} \quad (4)$$

Matrix K in Eq. (1) is a transform from the vector L to the vector P. Element k_{ij} in the matrix is a mapping from the layout parameter l_j to the performance characteristic p_i. We can use formula $p_i = k_{ij}(l_j)$ to denote each mapping. So $P = KL$. Each mapping is a nonlinear continuously differentiable function. So we can define the sensitivity of p_i to l_j as s_{ij} ,which is expressed in Eq. (2) . And then the sensitivity matrix S of K is defined in Eq. (3) . It is impossible to calculate the value of Eq. (2) directly ,but we can use Eq. (4) to approximate it. The correctness of linearization method has been demonstrated in Ref. [9]. In fact ,the value of Eq. (4) can be calculated by SPICE simulation. By the sensitivity analysis , CA can transform the upper limit of circuit performances to the upper limit of parasitics parameters ,such as interconnection parasitics and transistor diffusion parasitics ,which can be further transformed to geometry constraints such as length and area by parasitics models. For example ,by sensitivity analysis CA can distinguish critical nets information and interconnection parasitics constraints.

2. 2　Circuit topology analysis

Definition 4 : diffusion graph is a graph obtained by mapping each node in a MOS circuit to a vertex of a graph and mapping each source and drain connection of transistor to multiple-edge in the graph ,the number of which is the same with the finger number of the transistor.

CA converts circuit schematics to diffusion graph and then adopts some graph algorithms to recognise some special circuit units ,such as differential pairs ,current mirrors and cascade architectures. DG can directly generate layouts for them in a single unit-circuit. We use a graph-based search

algorithm in Ref. [10] to detect all symmetry constraints for placement and routing.

2.3 Signal-flow analysis

The methodology of signal-flow analysis is to partition the original circuit into core-circuit and bias-circuit. Core-circuit carries signals and bias-circuit supplies bias-voltage for the core-circuit. The core-circuit is composed of several stage-circuits among which there are obvious relationship for input and output.

CA can also calculate the values of currents on wires, which is useful to the variable wire width routing.

3 Device generator sub-system

Performance-driven analog device generation: given a circuit specified as a set of devices (transistors, capacitors and resistors) and a netlist interconnecting these devices and the constraints of devices level, we define the set of devices as D and the set ψ as a partition of the set D generated according to the results of CA, and then for $\forall e \in \psi$, we generate the layout of e, which satisfies the constraints of devices level.

DG is divided into four functional sub-systems. Firstly, DG generates common-centroid structures for MOS transistors with strict matches demand; the second is stage-circuits; the third is varieties of a single MOS transistor; the forth is single analog devices.

3.1 Generating the single analog device

CA can directly generate transistor, resistor and capacitor, the layouts of which are illustrated in Fig. 2.

3.2 Generating varieties for a single MOS transistor

Because the finger number of a transistor can be adjusted, the transistor has many varieties. Any variety_m and variety_n must hold the equation $W_m F_m = W_n F_n$, where the character W and F stand for channel width and the finger number respectively. Different layouts for the same transistor are illustrated in Fig. 3.

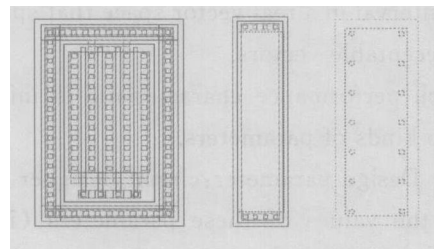

Fig. 2 Layout of the single devices (transistor, resistor and capacitor)

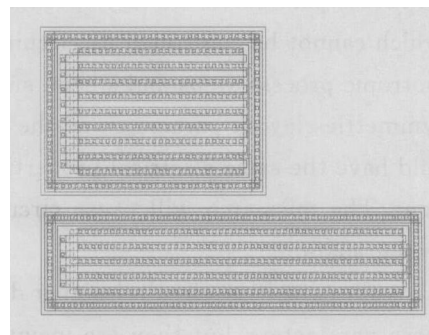

Fig. 3 Different varieties of the same MOS transistor

Different layouts for the same transistor have different parasitic parameters, which mainly include parasitic capacitors of source/drain regions. We can use Eq. (5) to evaluate the parasitics of diffusion regions, where C_{jSBt} refers to the total source/drain bulk capacitance, A_S and P_S is the source/drain area and perimeter, C_j and C_{jsw} is the bottom and sidewall junction capacitances in absence of any junction voltage and ϕ_j is the built-in junction potential. m_j and m_{jsw} depend on the doping profile of the junction. Equation (5) reveals that capacitance can be reduced by minimizing the size of diffusion regions.

$$C_{jSBt} = \frac{A_S C_j}{\left(1 - \frac{V_{BS}}{\phi_j}\right)^{m_j}} + \frac{P_S C_{jsw}}{\left(1 - \frac{V_{BS}}{\phi_j}\right)^{m_{jsw}}} \qquad (5)$$

3.3 Generating common-centroid structure

A group of devices with strict match constraints can be generated together in a single unit-circuit. We satisfy these constraints at the stage of

DG rather than leave them to PL so as to satisfy the constraints better and reduce calculation time of placement. Common-centroid structure is the best match style for the devices[11]. There is an example displayed in Fig. 4. We adopt the improved algorithm based on dual Euler-loop[12] to generate 1D common-centroid structure. But if the ratio *L*/*W* of transistors is small and the finger number is large, the outline of layout will be very narrow and long, which is not expected by PL and RT. So we must generate 2D common-centroid structure[13], the outline of which will approximate a square. The layout is illustrated in Fig. 5.

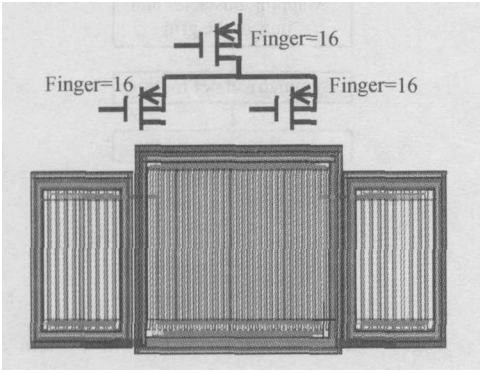

Fig. 4 Schematics of differential pairs and 1D common centroid layout

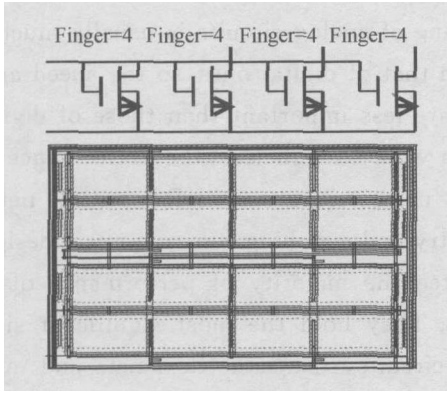

Fig. 5 Schematics of current mirror and 2D common centroid layout

3.4 Generating stage-circuit

We generate stage-circuits for the core-circuit. Each stage-circuit is a functional unit processing signal, which is surrounded by guard ring to eliminate latch-up effects. Each transistor in stage-circuit is added by dummy fingers to eliminate fringe effects. In Fig. 8 we can see all the stage-circuits clearly.

4 Placement sub-system

Performance-driven analog circuit level placement: given a circuit specified as a set of devices layouts and ports location of each device/unit-circuit and interconnection relationship and the placement constraints, we select an optimal variety and position for each device/unit-circuit, with restriction to design rules and placement constraints (i.e. symmetry and outline ratio) so as to minimize layout area.

At the stage of placement we mainly pay attention to the following three constraints:

(1) Symmetry constraints: two devices with symmetry constraints must have identical variants and mirrored orientations. Self-symmetric device must be placed on the symmetry axis. Couples and self-symmetric devices with the same symmetry axis must share the same symmetry axis.

(2) Matching constraints: in fact many match constraints have been satisfied at the stage of device generation. So, we only place devices with the same orientations such that the current flow is strictly parallel, which can reduce asymmetries caused by the anisotropic process steps and silicon substrate.

(3) Interconnect parasitics: although the actual parasitic capacitance and resistance of interconnection are calculated at the stage of routing, their minimum achievable values are determined after placement, so at the stage of placement we must estimate these values.

PL adopts different algorithms to implement placement for core-circuit and bias-circuit. The algorithm firstly completes placement for core-circuit and then for bias-circuit. Floorplan representation is based on CBL[14].

4.1 Core-circuit placement

We adopt determinate algorithm to implement

the core-circuit placement because CA gives strict and complete constraints and these stage-circuits have evident relationship that the output of the front stage-circuit is the input of the back stage-circuit, so we place all the stage-circuits from left to right and constrain the height of the stage-circuits to a certain proper value to obtain layout of the core-circuit close to a rectangle. In Fig. 8, we can clearly recognise these stage-circuits. Constraints of alinement and symmetry are illustrated in Fig. 6.

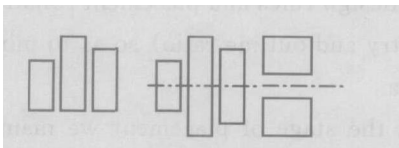

Fig. 6 Alinement and symmetry constraints

4.2 Bias-circuit placement

This part of circuit has more slack constraints except differential pairs and current mirrors, the strict constraints of which have been satisfied by DG. So we only need to consider the constraints of interconnection parasitics in the bias-circuit. Our optimization algorithm is based on simulation annealing[15]. The cost function is listed in Eq. (6), where C_{area} minimizes the layout area, C_{net_length} minimizes total length of all nets, C_{perf} can keep the performance of the circuit within the allowable range (we mainly consider the performance degradation induced by interconnect parasitics for the device mismatch has been solved by DG), C_{width} constraints the width of bias-circuit close to that of core-circuit, which can make the whole layout close to a rectangle. The weighting coefficients are experiential values obtained by many experiments.

$$C = \alpha C_{area} + \beta C_{net\_length} + \delta C_{perf} + \gamma C_{width} \quad (6)$$

5 Routing sub-system

Performance-driven analog circuit routing: given a placement of a circuit specified as a set of devices layout with their positions and orientations and the interconnection relationship of terminals, we generate the layouts of all the nets such that the circuit is interconnected correctly and layouts of all the nets must satisfy the design rules and routing constraints.

In Fig. 7, we classify the nets into three types that are symmetry nets, matching nets, power/ground nets and general nets. RT uses different routing algorithms for different types of nets.

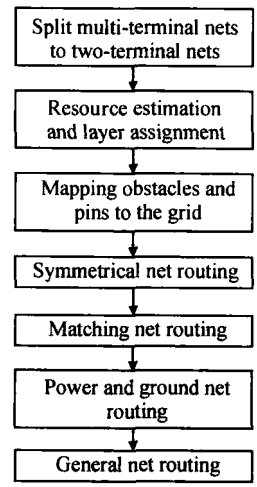

Fig. 7 Algorithm flow of RT sub-system

5.1 Symmetry nets and matching nets

It is known that the number of interconnection in routing of analog circuits is usually much smaller than that of digital ones, so the speed and efficiency are less important than those of digital circuits as well. Nonetheless, the performance of circuits is more critical, so in RT critical nets with symmetry and matching constraints are designed to guarantee the majority of performance of analog circuits. They hold the most significant signal in analog circuits, so symmetrical nets and matching nets should be routed prior to any other nets. Furthermore, symmetrical nets that are processed firstly have a higher priority than matching ones. Although these two types of critical nets have different priorities and they are routed sequentially, the core algorithms used to search paths are the same. Herein lays the A* algorithm, a heuristic algorithm referred from the field of AI, which is really practical and efficient.

5.2 Power/ground nets

RT uses two pieces of wide horizontal metal to implement ground net layout. One is placed on the top of layout, the other on the bottom of layout. RT places one piece of wide horizontal metal to implement power net layout placed in the middle of core-circuit and bias-circuit. In Fig. 8 they are all illustrated clearly.

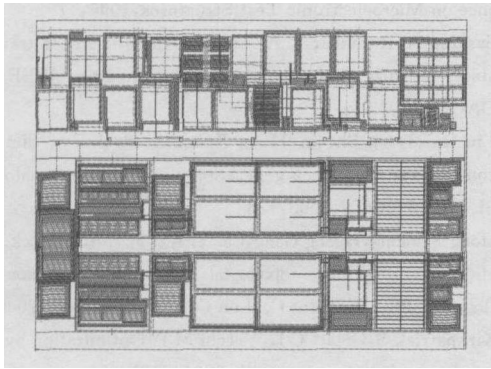

Fig. 8 Final layout of the opamp circuit

5.3 General nets

Compared with critical nets, general nets have no constraints such as symmetry and matching, other than the function as connection. Therefore, the goal of general net routing is similar to that of routing for digital circuits. It means that the success rate of routing, as well as the wire length and the count of vias should be paid more attention when the performance of an analog circuit is guaranteed. Because of the features of general nets, the classical maze routing algorithm[16] is chosen. Moreover, target-oriented expansion strategy and bi-direction expansion strategy are added to the algorithm to improve the routing efficiency. Rip-up re-routing is also integrated to obtain the better routing results so as to ensure the routing completion rate.

For all types of nets, minimization of interconnection parasitics must be considered, which RT uses RC circuit module to approximate. Different parts of an analog circuit often carry significantly different currents. To avoid electro-migration in high-current wires, RT must adjust the width of wires according to the values of currents on them. Nowadays, technology processes have six or more metal layers. RT must take advantage all of the available routing layers to improve the performance of the analog circuit because the further the metal layers are, the less interference the signals on them have.

6 Results and conclusion

We have implemented the whole system in C++. Some layouts of single device and unit-circuit have been displayed in Figs. 2～5. A complete example, opamp circuit, is illustrated in Fig. 8. All the results are gotten on the platform of Solaris of Sun-V880. The running time of CA is 4; that of DG is 7; that of PL is 255 and that of RT is 166.

The top part of Fig. 8 is the layout of bias-circuit, and the other part is the layout of core-circuit. The experimental results demonstrate that we have obtained high-quality layout. For core-circuit we obtain good matches and complete symmetry for placement and routing. All of the stage-circuits have approximate height. For the bias-circuit part we obtain high usage ratio of area (always greater than 90 %) and 100 % routing completion ratio. The whole layout is close to square. What's more, all the performance specifications have been satisfied.

In this paper, a signal-flow driven automatic physical synthesis methodology of SDAPS for analog circuits has been presented, correctness and effectiveness of which have been demonstrated by the experiment. But there is still much work to do in the future. For DG we want it to generate more types of unit-circuits; for PL thermal analysis of circuits should be considered; for RT it should support shielding routing and layer jumping.

Acknowledgement The authors would like to thank the researchers of Accelicon Micro-Electronics Company for many discussions and much help for implement of the system. All of the circuit ex-

amples shown in this paper are provided by Wu Yuping of the company, to whom the authors gratefully acknowledge.

References

[1] Onodera H, Kanbara H, Tamaru K. Operational amplifier compilation with performance optimization. Proc IEEE CICC, 1989:17

[2] Koh H Y, Sequin C H, Gray P R. OPASYN: a compiler for CMOS operational amplifiers. IEEE Trans Computer-Aided Design, 1990, 9:113

[3] Harjani R, Rutenbar R A, Carley L R. OASYS: A framework for analog circuit synthesis. Carnegie Mellon Univ., Pittsburgh, PA, Res Rep CMUCAD-89-65, 1989

[4] Yaghutiel H, Vincentelli A A, Gray P R. A methodology for automated layout of switched-capacitor filters. Proc IEEE ICCAD, 1986:444

[5] Jusuf G, Gray P R, Vincentelli S A. CADICS-Cyclic analog-to-digital converter synthesis. Proc IEEE ICCAD, 1990:286

[6] Rijmenants J, Litsios J B, Schwarz T R, et al. ILAC: an automatic layout tool for analog CMOS circuits. IEEE J Solid-State Circuits, 1989, 24(2):417

[7] Cohn J M, Garrod D J, Rutenbar R A, et al. KOAN/ANAGRAM Ⅱ: new tools for device-level analog placement and routing. IEEE J Solid-State Circuits, 1991, 26(3):330

[8] Chang H, Vincentelli S A, Balarin F, et al. A top-down, constraint-driven design methodology for analog integrated circuits. Proc IEEE CICC, 1992:841

[9] Choudhury U, Vincentelli S A. Use of performance sensitivities in routing of analog circuits. Proc IEEE Int Symp Circuits Syst, 1990:348

[10] Yi Su, Dong Sheqin, Hao Qingsheng, et al. Automated analog circuits symmetrical layout constraint extraction by partition. Proc ASICON, 2003:166

[11] Bastos J, Steyaert M, Graindourze B, et al. Matching of MOS transistors with different layout styles. Proc IEEE Int Conference on Microelectronic Test Structures, 1996:17

[12] Basaran A, Rutenbar R A. An O(n) algorithm for transistor stacking with performance constraints. Proc IEEE/ACM DAC, 1996:221

[13] Liu Rui, Dong Sheqin, Hong Xianlong, et al. Two-dimension common-centroid stack generation algorithms for analog VLSI. Proc ASICON, 2003:128

[14] Hong Xianlong, Huang Gang, Cai Yici, et al. Corner block list: an effective and efficient topological representation of non-slicing floorplan. Proc IEEE Int Conf on Comp-Aided Design, 2000:8

[15] Kirkpatrick S, Gelatt C D, Vecchi M P. Optimization by simulated annealing. Science, 1983, 220(4598):671

[16] Lee C Y. An algorithm for path connections and its application. IRE Trans Electron Computers, 1961, EC-10:346-365

信号流驱动的模拟电路版图综合方法*

龙 迪 洪先龙 董社勤

(清华大学软件研究所, 北京 100084)

摘要: 提出了一种新的基于信号流分析的模拟电路版图综合方法. 电路分析子系统采用新提出的信号流分析方法再结合已有的电路拓扑分析和电路灵敏性分析方法生成布图约束控制电路性能的衰减. 由于考虑了电路中有关信号流的启发式信息, 该方法的复杂性较一般的纯粹性能驱动方法小. 然后分别在器件生成子系统、布图子系统和布线子系统中实现这些约束, 使得这些约束在最容易实现的阶段得到满足. 实际的电路例子已经证明了这一方法可以获得出色的电路性能.

关键词: 模拟电路布图自动化; 信号流; 电路分析; 器件生成; 布局; 布线

EEACC: 1205

中图分类号: TN431.1 **文献标识码**: A **文章编号**: 0253-4177(2005)04-0689-08

*国家自然科学基金(批准号:90307005), 国家自然科学基金国际合作项目(批准号:60121120706), 美国国家自然科学基金(批准号:CCR-0096383), 国家高技术研究发展计划(批准号:2004AA1Z1050)资助项目

龙 迪 男, 1980 年出生, 博士研究生, 从事模拟电路布图自动化研究. Email:longd02@mails.tsinghua.edu.cn

洪先龙 男, 1940 年出生, 教授, 博士生导师, 研究方向为 VLSI 布图理论与算法.

董社勤 男, 1964 年出生, 副教授, 研究方向为 VLSI 布图理论与算法.

2004-04-30 收到, 2004-12-06 定稿