TRANSISTOR PLACEMENT FOR NONDUAL CMOS CELL SYNTHESIS

Mário C. B. Osório, Renato P. Ribas, André I. Reis

Nangate Research Lab – Instituto de Informática, UFRGS
Av. Bento Gonçalves, 9500 – CEP 91501-970, Porto Alegre, Brazil
{osorio, rpribas, andreis}@inf.ufrgs.br

ABSTRACT

This paper presents a new algorithm for transistor placement of nondual cells. These cells do not have complementary series-parallel planes. The solution consists of an integration of 3 other algorithms, the Fleury, the Backtracking and the String Matching. In this paper, the proposed algorithm is applied to a class of cells with nondual disjoint planes and compared to the layout of CMOS standard cells, also produced by the same algorithm.

1. INTRODUCTION

Some logic functions implemented in CMOS Non-Complementary Series-Parallel logic (NCSP) present fewer transistors in series when compared to standard CMOS complementary series-parallel (CSP) [1]. That feature can lead to circuits with shorter delay time. On the other hand, the lack of duality between the two logical complementary planes results in more irregular topologies. Hence, NCSP cells might not be physically synthesizable by the traditional tools.

Most of the automatic layout generation techniques have been based on the first methodology proposed by [12]. The transistors are placed in two parallel rows of diffusion (for p- and n-type transistors) and vertically aligned by the gate inputs. CSP cell generation was explored in [2], [3], and [4].

Among previous works for layout of nondual circuits, [5] was the first to present an algorithm implemented in a commercial tool. In [6], a new tree representation for circuits with arbitrary topology was presented. Thus, with minor modifications, the algorithms to treat CSP circuits could be used to order cells with other topologies. [7] has developed an exact minimum-width heuristic for transistor placement in nondual cells, based on Boolean Satisfiability. Other works include [8] and [9].

This paper proposes a new approach for the resolution of the transistor placement problem in nondual CMOS cells. The main objective of this work was to minimize the layout area of NCSP cells. However, the algorithm can be easily modified to accept other kinds of cells.

2. PROBLEM DESCRIPTION

Given a spice netlist, a placer needs to find an order for the transistors within an area as small as possible. This depends mainly on the number of breaks in the diffusion rows, which can be minimized by finding Euler paths for the complementary planes. If both paths have the transistors in the same order (with respect to the input signal), the pairing of complementary transistors is also achieved. The rules adopted for the layout generation were (see Fig.1):

a) Cells with two disjoint networks and without series-parallel constraint or equal number of transistors.
b) Transistors in two parallel horizontal rows. Upper row for PMOS and lower for NMOS.
c) Complementary transistors with gate terminals aligned, when possible.
d) Two transistors without pair can also be aligned.

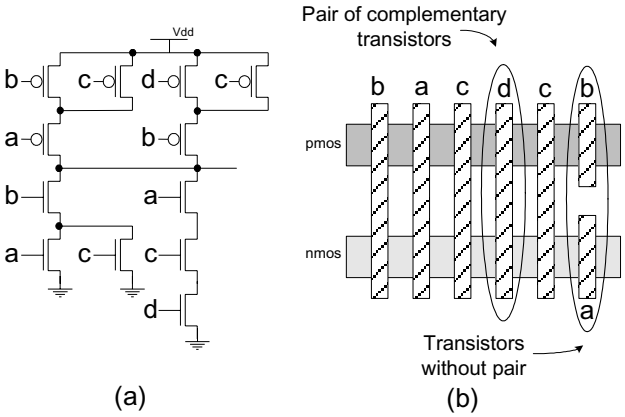

Fig.1. (a) NCSP cell. (b) A possible order.

3. PROPOSED ALGORITHM

The solution implements three other well-known algorithms. The Fleury algorithm finds an Euler path of a graph that has at least one. The algorithm is described in [10]. The Backtracking algorithm makes an exhaustive search over a data structure containing the solution space [11]. This technique can be used to find a list of Euler paths for a graph if combined with the Fleury algorithm. The String Matching algorithm compares two strings. The inputs are a 'pattern' string 'P' and a 'text' string 'T'. The output is the Levenshtein distance between 'P' and 'T' (that is equal to the number of operations to change 'T' into 'P') [10]. Given two Euler paths, the string matching algorithm defines if they are good enough (a good number of aligned transistors) to be used as the transistor order for the circuit.

Some definitions must be introduced.

Def. 1: a start-point (SP) is a vertex which is able to be used as a starting vertex for the Fleury algorithm.

Def. 2: two start-points are defined as compatible iff they belong to different planes and both have at least one edge (transistor) driven by the same signal.

The algorithm proceeds as follows:

Input: a spice netlist containing the cell's transistors.

Output: an ordered netlist defining the relative position of each transistor in order to minimize the cell area.

1. Find the start-points for each plane.
2. Find all possible compatible start-points for the cell.
3. List all Euler paths for both planes, using the backtracking and the Fleury algorithms. Those paths shall begin with the compatible start-points in order to filter some of the possible Euler paths.
4. Consider the sequence of input signals of an Euler path as a string of signals. Then, apply the string matching algorithm in the paths.
5. The best pair of Euler paths is the pair which results in the lowest distance.
6. Use the operations of the string matching algorithm to improve the alignment of the transistors, in the following way:
   a) *insert*: insert a break in the NMOS path.
   b) *delete*: insert a break in the PMOS path.

The insertion of breaks causes a shift in the corresponding plane, leading to a better alignment of the complementary transistors with the same input signal. In some cases, the length of the cell has no increase, due to two facts: a) in NCSP cells, one plane may have more transistors than the other one; or b) one plane does not have an Euler path, so it will need a break to connect the paths. Thus, the path for this plane might be longer than that of the other plane.

4. RESULTS

A test of the algorithm was carried out with a group that represents all the 4-input cells with different electrical topology [13], for both CSP and NCSP families. This test was chosen because it encompasses a great amount of feasible implementations of logic CMOS cells.

Table I compares the execution of the proposed algorithm through the two groups of cells. Here, minimal solution is the shortest length for the layout of a cell (considering the sum of transistors and diffusion breaks). For CSP cells, it is based on [2] and it considers a complete alignment of all gate inputs. On the other hand, a NCSP cell may have unpaired transistors. Then, the minimal solution is the length of the plane with higher number of transistors. The algorithm has run somewhat quickly, but the results were indeed far from the expected. Only for 62% of the CSP cells the minimal solution was achieved and 55% for NCSP.

Table I. Results of placement of 4-input cells

	CSP	**NCSP**
# of cells	3982	3982
# of minimal solutions	3126 (62%)	2183 (55%)
run time	6 319s	3 934s
time / cell	1.59s	0.99s

In the graph of Fig.2, the circuits are grouped according to the number of breaks they presented beyond the minimal solution. That means that the placer has found the best solution for the circuits in the '0' column.

Fig.3 groups the circuits with respect to the number of misaligned gate signals, that is, the number of pairs of PMOS and NMOS transistors with the same gate input.

5. CONCLUSION

A new algorithm for transistor placement of nondual cells was proposed. The new algorithm is a combination of three other common algorithms. The experimental results have shown that the approach here described has reached only an average transistor ordering, considering the minimum feasible solution.

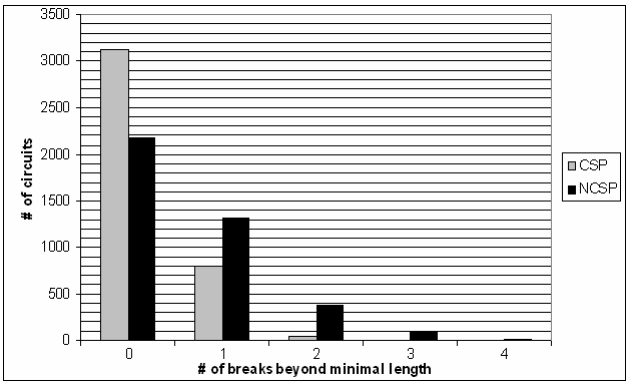

Fig.2. Circuits grouped according to the number of extra breaks beyond the minimal solution.

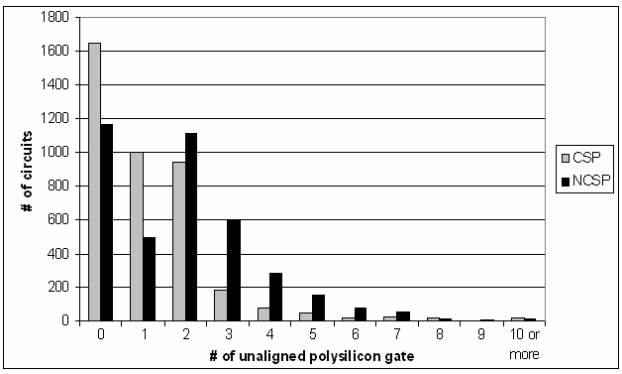

Fig.3. Number of unaligned polysilicon gates per circuit

6. REFERENCES

[1] Schneider, F.R. et al, "Exact lower bound for the number of switches in series to implement a combinational logic cell", *Proc. of the ICCD*, pp. 357-362, 2005.

[2] Hwang, C.Y. et al, "A Fast Transistor-Chaining Algorithm for CMOS Cell Layout", *IEEE Trans. on CAD*, vol.9, no.7, pp. 781-786, 1990.

[3] Maziasz, R.L. et al, "Exact Width and Height Minimization of CMOS Cells", *Proc. of the 28th DAC*, pp. 487-493, 1991.

[4] Iizuka, T. et al, "High Speed Layout Synthesis for Minimum-Width CMOS Logic Cells via Boolean Satisfiability", *Proc of the. ASP-DAC*, pp. 149-154, 2004.

[5] Bar-Yehuda, R. et al, "Depth-First-Search and Dynamic Programming Algorithms for Efficient CMOS Cell Generation", *IEEE Trans. on CAD*, vol.8, no.7, pp. 737-743, 1989.

[6] Carlson, B.S. et al, "Dual Independent Layout Generation of Arbitrary Circuit Topologies", *35th MWSCAS*, pp.528-531, 1992.

[7] Iizuka, T. et al, "Exact Minimum-Width Transistor Placement Without Dual Constraint for CMOS Cells", *Proc. of the 15th GLSVLSI*, pp. 74-77, 2005.

[8] Gupta, A. et al, "A Cell Layout Generator with Integrated Transistor Folding", *Proc. of the ED&TC,* pp. 393-400, 1996.

[9] Riepe, M.A. et al, "Transistor Placement for Non-complementary Digital VLSI Cell Synthesis", *ACM TODAES,* pp. 81-107, 2003.

[10] Skiena, S.S., *The Algorithm Design Manual*, Springer-Verlag, New York, 1998.

[11] Gerez, S.H., *Algorithms for VLSI Design Automation*, John Wiley & Sons, England, 1999.

[12] Uehara, T. et al, "Optimal Layout of CMOS Functional Arrays", *IEEE Trans. on Computers,* vol.30, no.5, pp.305-312, 1981.

[13] Sasao, T., *Switching Theory for Logic Synthesis*, 2nd ed, Kluwer Academic Publishers, 2000.