Evaluation of VLSI Layout Implementation for Efficiency

Michel Robert, Joel Trauchessec, Guy Cathébras, Vincent Bonzom, Nadine Azemard, Denis Deschacht, Daniel Auvergne

EVALUATION OF VLSI LAYOUT STYLE IMPLEMENTATIONS FOR EFFICIENCY

M. Robert, J. Trauchessec, G. Cathebras,
V. Bonzom, N. Azemard, D. Deschacht, D. Auvergne

LABORATOIRE D'AUTOMATIQUE ET DE MICROÉLECTRONIQUE DE MONTPELLIER
UA CNRS D0 3710
UNIVERSITÉ MONTPELLIER II, SCIENCES ET TECHNIQUES DU LANGUEDOC
Place Eugène Bataillon, 34095 Montpellier cedex 5
FRANCE
Telex: 490944F, Fax: (+33) 67 52 14 96, Phone: (+33) 67 54 45 90

Abstract

As an attempt to define a priori mapping rules for performance driven layout, we show in this paper how an automatic module generator can be used to compare different implementation styles of regular layout.

Speed and area performances of gate and linear matrix approaches are compared. It is clearly shown that abutment of diffusions results in lower "locox" parasitic capacitances inducing higher speed performances for linear matrix style.

I - INTRODUCTION

As part of the design automation project conducted in the Microelectronics Department (LAMM) of Montpellier, we have been explored ways to automate structural synthesis of integrated circuits. As a design rule independent and flexible way to design on Silicon non regular structures, we are developing a structural compiler, PRINT /1/ based on a symbolic library associated to a technology interpreter, compaction procedures, post layout extraction, temporal evaluation and optimal sizing (Figure 1). In the way to obtain an adaptative module that generates cells automatically to satisfy loading and constraint conditions in terms of performance /2/ and layout, we present an application of the automatic layout and evaluation modules of PRINT, to compare layout style efficiencies, with parameterization of the parasitic capacitances.

II - LAYOUT STYLE RELATED WORK

Generally, cell compilers with fixed layout architectures attempt to regularize placement of randomly connected logic as found in control and glue logic. Two fixed layout methods for CMOS circuits can be categorized with respect to the growing dimension of the cell with the number of components : **linear-matrix** and **gate-matrix** arrays.

A **linear-matrix** architecture has been introduced by Uhuera and Van Cleemput /3/, to lay out complex gates in a single row of N and P transistors, aligned at the common gate connection (Figure 2-a). Minimizing the number of separations between diffusion strips, reduces the overall length of the cell. Intra cell routing is realized between rows, using vertical polysilicon and horizontal metal wires.

A **gate matrix** architecture has been introduced by Lopez and Law /4/ and widely applied to dedicated generators /5/, /6/, /7/. It extends the style of complex cell layout to

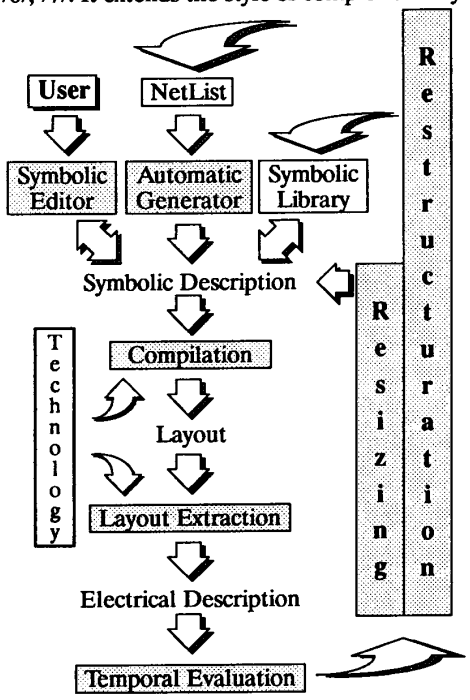

Figure 1 : Structure of the PRINT Compiler

several cells (Figure 2-b), by aligning transistors along vertical polysilicon lines following the order of the control inputs. All transistors controlled by the same gate net are placed in a single column. Connections are made by abutment of the horizontal diffusions or by vertical diffusion links between polysilicon lines, and with horizontal metal lines.

III - LAYOUT STYLE COMPARISON.

Each layout style induces the general characteristics of the cells, such as their aspect ratio and mostly their speed

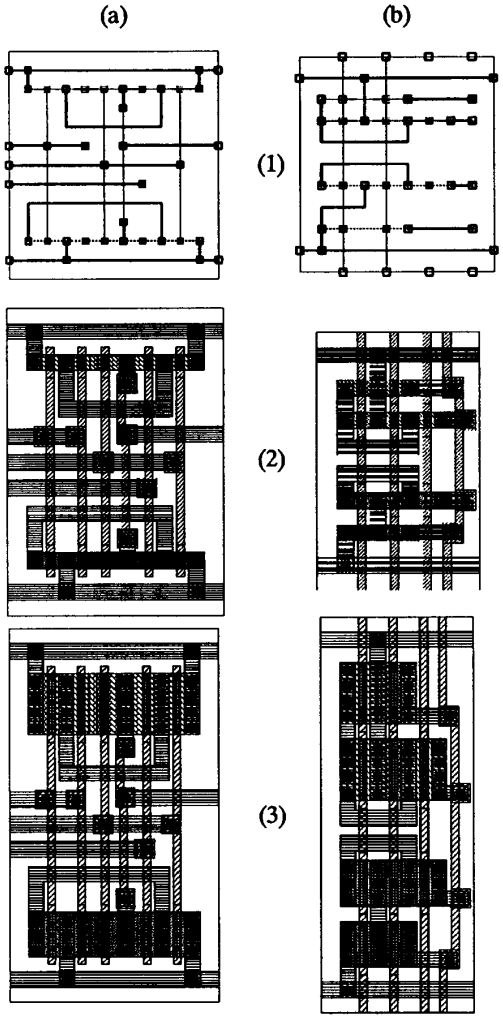

Figure 2 : Examples of Layout Styles, col.: (a) Linear Matrix, (b) Gate Matrix rows: (1) Symbolic layout, (2) layout with minimum transistors sizes, (3) layout with sized transistors

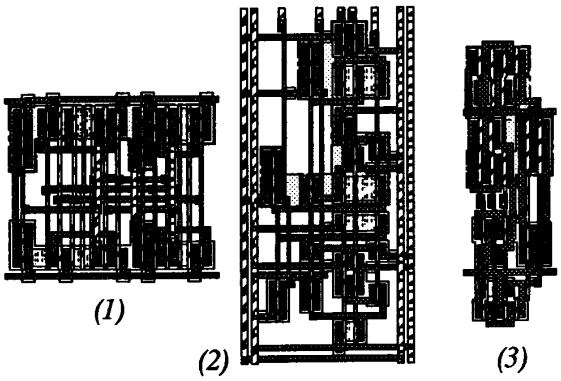

Figure 3 : Comparison between layout styles (adder slice). (1) linear matrix, (2) gate matrix style (3) manual optimization of the stick diagram.

area efficiency through the layout associated interconnect and diffusion parasitic capacitances. In order to evaluate these different styles, we used the symbolic editor of the PRINT generator and compared post layout characteristics of an automatically generated adder bit slice. Speed characteristics, are obtained from direct temporal evaluation (PATH RUNNER /8/) of the extracted layout.

Table 1 summarizes area and speed characteristics of the different layout implementations of the adder slice (Figure 3). We have compared :

- automatic linear-matrix style, obtained from PRINT compiler,
- gate-matrix with polysilicon oriented style /5/,
- by hand optimized (jog insertion) gate-matrix (referred as manual stick in the table), to minimize cell area.

Area and speed characteristics have been determined for different transistor sizing options : minimum width, constant width (standard cell approach), optimal width /2/.

From the data shown in Table 1, it appears that the best speed area trade off is obtained with the "linear-matrix" approach. As it can be observed on the parasitic capacitances extracted from the layout, this is a direct consequence of the abutment of the diffusions which minimizes the locox side wall capacitances. As shown in the Figure 4, free diffusion ends, which characterize gate-matrix style, result in locox parasitic capacitances 3 or 4 times larger than for abutted diffusions. This observation explains the rather poor speed performance of gate-matrix layout style. This is confirmed by the results of manual stick layout which gives smaller area cell with lower speed than for linear-matrix implementation.

As a minimum layout parasitic capacitance style, "linear-matrix" layout appears to be an efficient style of implementation for fast cells. This result justify the choice of "linear-matrix" style we did for the regular cell generator of PRINT compiler.

IV - PARASITIC CAPACITANCE PARAMETERIZATION

As an attempt to predict cell performances to drive higher level of design (synthesis or mapping level), we use in this part the regularity allowed in the "linear-matrix" style to parameterize the parasitic layout capacitances. For each electrical node, we can derive the diffusion parasitic capacitance from the number and the width of the transistors connected to this node and from constant parameters depending on the design rules of the selected technology.

As illustrated in Figure 4, the total diffusion capacitance of a node can be described as :

$$C_{par} = A_N.C_{jN} + L_N.C_{jSWN} + L_{chN}.C_{jSWchN} + A_P.C_{jP} + L_P.C_{jSWP} + L_{chP}.C_{jSWchP} \quad (1)$$

where C_j, C_{jSW} and C_{jSWch} are respectively the diffusion bottom area capacitance, the "locox" periphery capacitance and the "channel" periphery capacitance.

The parameters A_N, L_N, L_{chN}, A_P, L_P and L_{chP} are obtained from the widths (W_N, W_P) and the numbers (n_N, n_P) of transistors connected to the electrical node, by the following equations :

$$L_{chN} = n_N.W_N \quad (2a)$$

$$\text{if } n_N = 2 \text{ and } n_P = 0 \text{ then } \begin{cases} A_N = L_1.W_N \\ L_N = 2.L_1 \end{cases} \quad (2b)$$

$$\text{else if } n_N \text{ is even then } \begin{cases} A_N = \frac{n_N}{2} L_2 W_N \\ L_N = n_N L_2 \end{cases} \quad (2c)$$

$$\text{else if } n_N \text{ is odd then } \begin{cases} A_N = \left(\frac{n_N - 1}{2} L_2 + L_3\right) W_N \\ L_N = (n_N - 1)L_2 + 2L_3 + W_N \end{cases} \quad (2d)$$

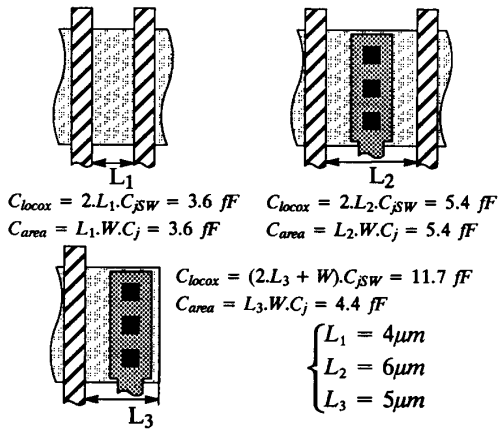

Figure 4 : Evaluation of diffusion area and periphery (locox) capacitances for different configurations (transistors N, W=16µm)

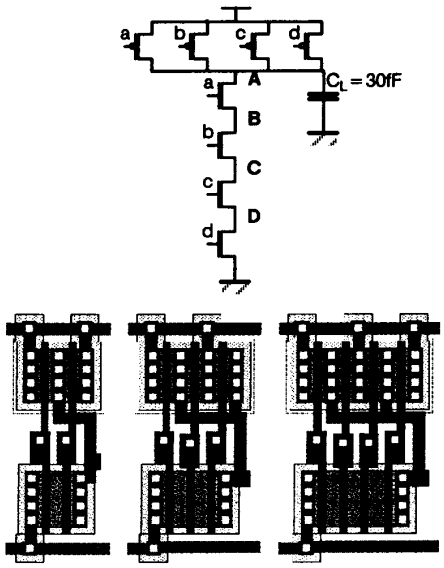

Figure 5 : Schematic and layouts of the nand cells referenced in Table 2.

Where L_1, L_2 and L_3 are constant parameters derived from design rules and defined in Figure 4. Identical expressions can be written for the P diffusions by exchanging subscript N and P respectively.

Application of equations (1) and (2) to logic gates leads to pre-layout temporal evaluation closed to post-layout ones, as shown in Table 2 where we compared, for NAND gates, the node capacitances obtained directly from the layout (post-layout) to the values calculated from equations (1) and (2) (pre-layout). Labels A, B, C and D of the left columns identify the output node and the serial array intermediate nodes, respectively (Figure 5). The good agreement obtained between calculated and measured values, confirms the validity of this parameterization. In the right column we give evaluation of delays obtained from post-layout files (t_{ref}), pre-layout estimations (t_{cpar}) and from electrical description without parasitics (t). The satisfactory agreement between t_{ref} and t_{cpar}, reinforces the validity of the proposed parameterization. However the large difference obtained for delays (t) evaluated without parasitic capacitances gives strong evidence of the necessity, for performance driven synthesis, to take account as soon as possible of the layout style of implementation.

V -CONCLUSION.

Performances of cell compilers as well as regular layout style of implementation have been mainly investigated with respect to total cell area. We showed in this paper that layout styles induced parasitic capacitances are of prime importance in determining speed performances of any implementation.

Using automatic module generator, we compared speed and area performances of gate and linear matrix approaches for regular layout. We showed that the regularity of transistor implementation, in linear style, results in lower parasitics and faster cells. Moreover this regular style allows to parameterize layout parasitic capacitances in a simple and satisfactory way. This constitutes a good starting point to define a priori mapping rules for performance driven layout. This gives also the possibility of accurate pre-layout estimation of cell delays as well as good initial sizing of transistors. Future application of these results will be the definition of technological thresholds for logical synthesis.

Table 1 : Comparison of the adder slice performances for different layout styles.			area			delay (nS) C_L = 1 pF		C_L = 100 fF	
			X (µm)	Y (µm)	S (µm²)	$t_{A,S}$	$t_{A,Carry}$	$t_{A,S}$	$t_{A,Carry}$
Table a	linear matrix		138	81	11178	17.1	14.4	8.6	5.7
L = 2µm	gate matrix	poly oriented	128	141	18048	19.5	16.1	10.9	6.9
W = 3µm		manual stick	63	114	7182	17.8	15.2	9.2	6.3
Table b	linear matrix		138	107	14766	4.7	3.7	3.0	1.9
L = 2µm	gate matrix	poly oriented	128	253	32384	5.9	4.4	4.2	2.6
W = 16µm		manual stick	63	210	13230	5.5	4.3	3.8	2.5
Table c	linear matrix		138	128	17664	4.1	2.9	3.3	1.9
L = 2µm	gate matrix	poly oriented	128	259	33152	5.3	3.4	4.4	2.4
W #		manual stick	63	210	13230	4.9	3.4	4.0	2.3

Delays are measured, with a timing analyser (PATH RUNNER /8/) after automatic extraction of layout capacitances, between the switching input (A) and the sum (S) or Carry outputs for different load conditions (C_L) : $t_{A,S} = t_{LH} + t_{HL}$. Technology parameters (2µmCMOS) : $V_{tN} = |V_{tP}| = 0.9V$, $K_{pN} = 5\ 10^{-5}\ A/V^2$, $K_{pP} = 2\ 10^{-5}\ A/V^2$.. Diffusion N capacitances : area $C_j = 57\ 10^{-6}\ pF/\mu m^2$, periphery $C_{jSWlocox} = 450\ 10^{-6}\ pF/\mu m$ $C_{jSWchannel} = 75\ 10^{-6}\ pF/\mu m$. Diffusion P capacitances : area $C_j = 175\ 10^{-6}\ pF/\mu m^2$, periphery $C_{jSWlocox} = 345\ 10^{-6}\ pF/\mu m$, $C_{jSWchannel} = 145\ 10^{-6}\ pF/\mu m$. In Table a and Table b, all transistors have the same sizes. In Table c, the transistor widths have been adjusted taking into account load conditions -Cl = 1pF- (/2/).

Table 2	Measured Cap. (fF)				Calculated Cap. (fF)				t_{ref} (nS)		t_{cpar} (nS)		t (nS)	
	A	B	C	D	A	B	C	D	rise	fall	rise	fall	rise	fall
nand2	51.82	9.65			43.52	9.65			0.16	0.16	0.15	0.15	0.07	0.08
nand3	80.78	9.65	9.65		69.85	9.65	9.65		0.33	0.22	0.30	0.21	0.12	0.08
nand4	82.30	11.46	9.65	9.65	69.58	9.65	9.65	9.65	0.52	0.25	0.45	0.24	0.20	0.11

REFERENCES :

/1/ M.Robert, D.Deschacht, G.Cathebras, S.Pravossoudovitch, D.Auvergne "PRINT methodology : a compilation approach for cell library generation", IEEE ISCAS'88, Helsinki (Finland), June 1988.

/2/ D.Auvergne, N.Azemard, V.Bonzom, D.Deschacht, M.Robert "formal sizing rules of CMOS circuits", the European Design Automation Conference, Amsterdam (Netherland), February 1991.

/3/ T.Uheara, W.M.Van Cleemput, "optimal layout of CMOS functionnal arrays", IEEE Trans., vol C-30, May 1981.

/4/ A.D.Lopez, H.F.S.Law, "a dense gate-matrix layout method for CMOS VLSI", IEEE trans. Electron Devices, vol ED-27, August 1980.

/5/ S.Huang, O.Wing, "improved gate-matrix layout", IEEE Transactions on computer-aided design, vol.8, n58, August 1989.

/6/ C.Piguet, G.Berweiler, C.Voirol, E.Dijkstra, J.Rijmenants, R.Zinszner, M.Stauffer, M.Joss, "ALADDIN : a CMOS Gate-Matrix layout system", IEEE ISCAS'88, Helsinki (Finland), June 1988.

/7/ R.L.Maziasz, J.P.Hayes "layout optimization of static CMOS fuctionnal Cells", IEEE Transactions on Computer-Aided Design, Vol.9, N57, July 1990.

/8/ D.Deschacht, P.Pinède, M.Robert, D.Auvergne, "PATH RUNNER : an accurate and fast timer analyser", the European Design Automation Conference, Glascow (Scotland), March 1990.