Local vs. Global Interconnections in Pipelined Arrays: An Example of the Interaction Architecture-Technology

Eduardo Boemo and Sergio López-Buedo

School of Computer Eng., Univ. Autónoma de Madrid,
Ctra. Colmenar Km.15, Madrid, España
http://www.ii.uam.es

Nelson Acosta and Elías Todorovich

ISISTAN, Universidad Nacional del Centro,
Tandil, Argentina.
http://www.exa.unicen.edu.ar

Abstract

This paper shows the effect of local and global interconnections in the area, throughput, and latency of VLSI arrays. As an example, a binary multiplier topology is pipelined in two advantageous directions to obtain two prototypes with the same logic depth (one processor element between consecutive lines of registers) but different wiring distribution. If a purely architectural point of view were adopted, both pipelines would have identical throughput. However, from the technological side, there is an important difference between the circuits: the second pipelining direction transforms all the global communications in local ones. This fact leads to a higher throughput, in spite of the area and routing overhead of the circuits.

1. Introduction

Pipelining is an antique but efficient way to improve throughputs. One of the first technologists that experienced its advantages was Henry Ford. In 1913, he reduced the duration of the motor assembly 3 times by dividing the task in 48 operations [1]. Additionally, combining parallelism and pipelining, he diminished the manufacturing time of a complete chassis from 728 to 93 minutes [2]. Fifty years later, the construction of electronic pipelines became an active research field. In 1965, Cotten published a complete analysis of pipelining [3]. An early application was reported in [4]: a pipelined adder was included in the legendary IBM 360 FPU. One important step was the combination of pipelining and regular structures. In 1969, De Mori [5] and Guild [6] proposed array circuits for binary multiplication. In the next decade, Hallin and Flynn [7], Deverell [8], and Davio and Bioul [9] pipelined these topologies. In 1978, Jump and Aura [10] introduced the concept of direction of pipelining and Hatamian and Cash constructed a VLSI prototype [11]. However, this line of research was soon hidden by the idea of systolic arrays, a popular concept introduced by H.T. Kung in 1982 [12].

2. Global and local communications

Kung classified the arrays in two categories. *Pure-systolic*, where all the wires between processor (except clock and reset wires) are local; that is, they drive just one input. On the contrary, in the *semi-systolic* arrays some wires send data to a group of processors, usually a file or a column of the array. However, from the pipeline research line reviewed above, global and local communications were simply a consequence of the combination of two elements: the circuit topology and the direction of pipelining. This fact is illustrated in Fig. 1. It shows a generic 6x6 topology with both local and global communications to input the Y and X data respectively.

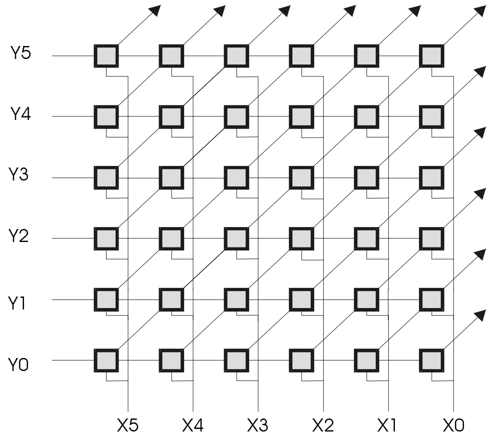


Fig. 1. A generic array with mixed communication: local to transmit the Y data and global to transmit the X data.

The array can be pipelined in the two preferential ways depicted in Fig. 2. Each intersection between the data wires and the "equitemporal" lines (dotted) indicate the points where 1-bit register must be inserted. Other pipelining directions are possible but not practical: neither the area nor the clock period is minimized.

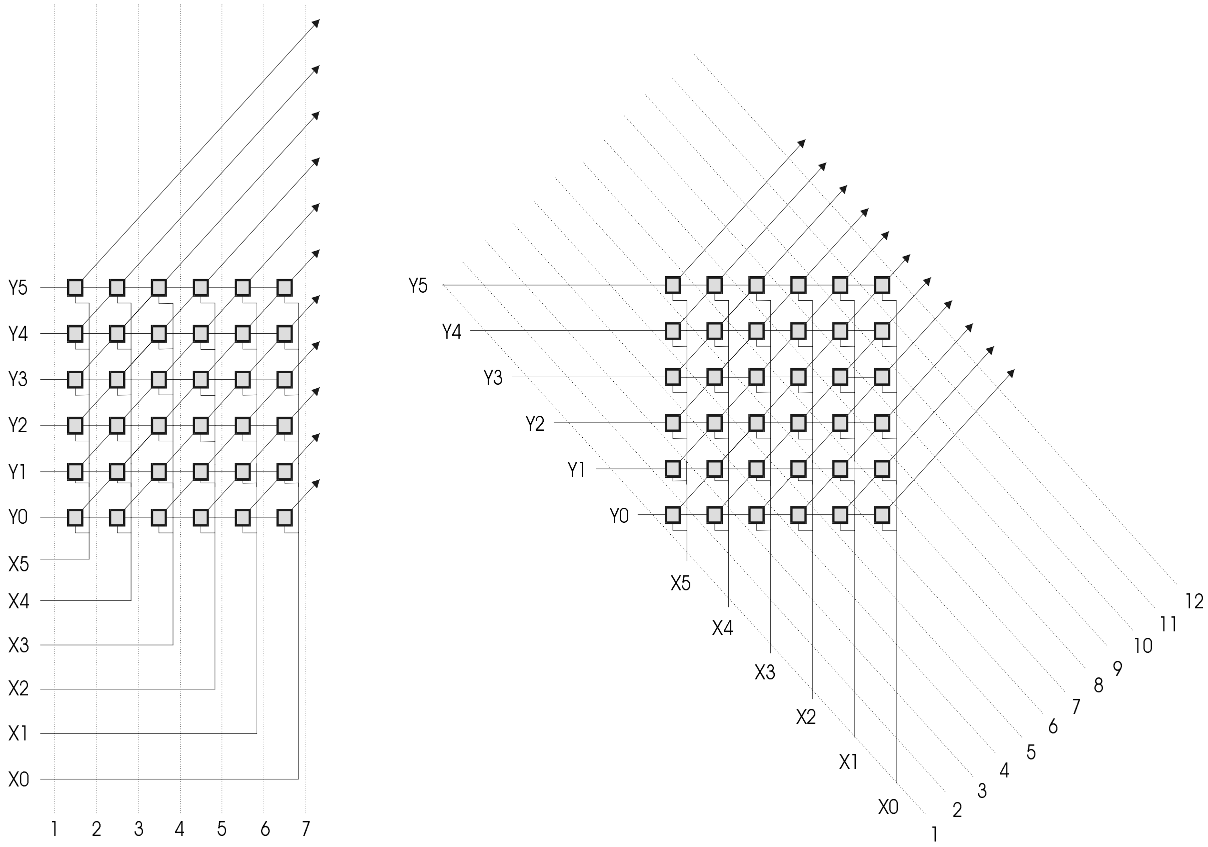


Fig. 2. Two preferential directions of pipelining for the same array.

In terms of area and latency, there is an important saving if global communication is utilized. For instance, in the circuits of Fig. 2 the numbers of registers to be inserted are 190 and 107 for global and local communication respectively, meanwhile the latency is 12 and 7 clock cycles for each case. Respect to the last variable, the throughput, there is no advantage if a purely architectural point of view is adopted. Both pipeline options have the same logic depth: one processor element between consecutive lines of registers. However, there is an important difference in terms of a VLSI implementation: in the semi-systolic version, each global line results confined inside a pipeline stage. So, its potential higher delay can significantly increment the clock period. On the contrary, this problem does not exist in the pure-systolic version. The registers that define the pipeline stage break each global line. As a consequence, the worse line pass from driving six inputs (or N in general) to drive just two: one corresponding to the PE and the other to the input of the register of the next stage.

The transformation of the data lines by the direction of pipelining is illustrated in the simplified scheme of Fig. 3. In the left graph, the global line that broadcasts the X5 data (highlighted) departs from an input register and reaches all the PEs situated in the same column. It must drive 6 inputs, and its wiring delay must be added to the combinational delay of the pipeline stage. On the contrary, in the right graph, which corresponds to the other direction, the pipeline registers broke the X5 global line in a set of lines, with a fanout of 2 inputs. As a consequence, even when the logic delay is the same in both arrays, the fraction of the stage delay corresponding to wiring is lower in the second case.

3. Experimental results

In order to illustrate the hypothesis of this paper, two arrays have been selected as case study: the 16-bit versions of the Hatamian-Cash [11] and the McCanny-McWhirter [13] multipliers. Both circuits share the same topology but the first includes a global communication, meanwhile the second transforms

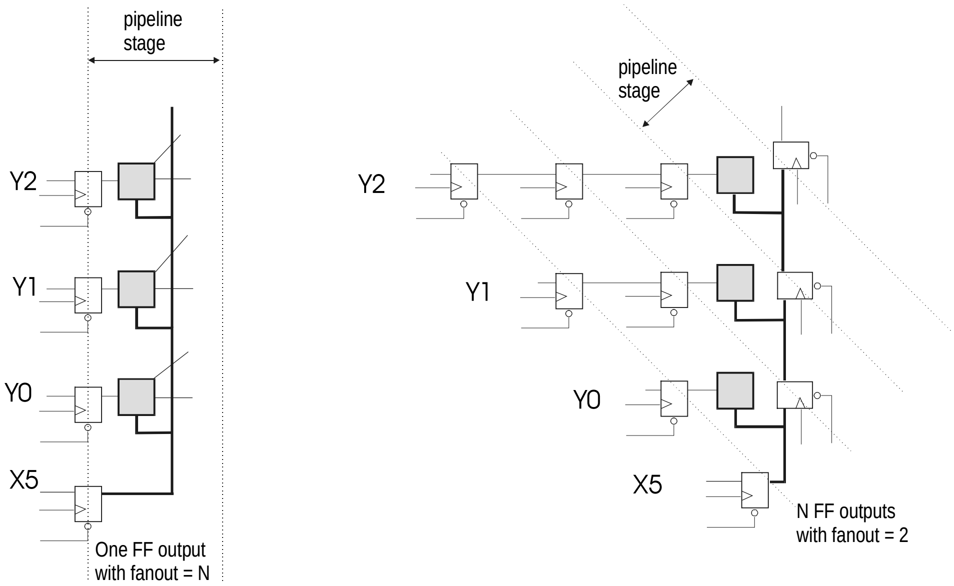


Fig. 3: Detail of the transformation of the global line (thick) that broadcasts X5 data to a column of PEs, in a set of lines with fanout=2.

these local communications in a set of “local” wires (in this paper, the definition of local wire is extended to include lines that drive up to two inputs). Two different technologies have been analyzed: a Xilinx XC4036XL-1-HQ240 FPGA and 1μ CMOS Standard Cells from the former ES2 company. The circuits were compiled using Foundation Series 1.5 and Cadence Design Framework II respectively. In order to avoid biasing the results by the experience of the designer, in all cases the default compilation options of the tools were utilized.

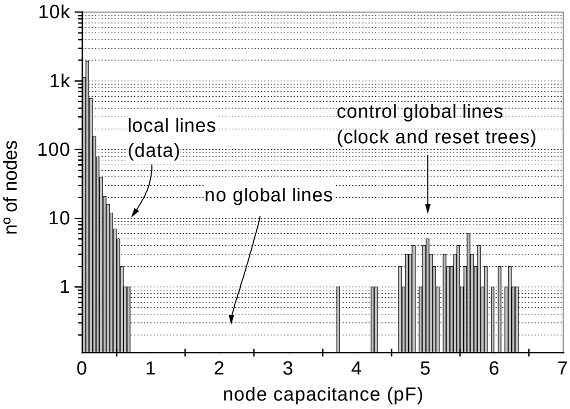
3.1 CMOS Standard Cells

The post-layout reports of two fine-grain-pipelined 16-bit Hatamian-Cash and McCanny-McWhirter multipliers prototypes developed in [14] have been utilized as a base for the comparisons. The regularity of the arrays made straightforward the identification of the different types of interconnections in the post-layout reports.

In Fig.4, the node capacitance histograms of both circuits are depicted. Three different set of wires can be distinguished in the Hatamian array (from left to right): local lines, that have the lowest values of capacitance; then, the global lines of data; and finally, the control global lines, that corresponds to the clock (TSPC) and reset trees. The direction of pipelining eliminates the global lines in the McCanny-McWhirter diagram (top histogram of Fig.4) at the cost of extra local and control lines. In this way, local or pure-systolic pipelining will produce a circuit speedup wherever the wiring degradation as well as the clock skew (produced by the extra registers) result lower than the delay of the global lines. This hypothesis has been true in the relatively antique 1μ-CMOS cell-based technology and the chip size utilized in this paper. For instance, in the McCanny topology, the worse local wire has a capacitance of 0.59 pF (0.68 pF for the whole node). Considering the extrinsic delay of the cell that drives the node, an *FADD1* [15], this solely wire contributes with 1 ns to the clock period. In the Hatamian circuit, the worst global line of data exhibits a capacitance of 0.98 pF (1.6 pF for the whole node, noticeably increased by the higher fanout). Thus, the wire and node contribution to the clock period is 1.8 ns and 2.9 ns respectively.

The second consequence of a local communication is the increment of registers and clock branches. However, the absolute value of the worst clock lines does not affect the delay of the stage: just their maximum difference or skew (in nanoseconds) must be computed in the pipeline period [16]. For instance, a maximum unbalance of 1.64 pF was measured in the pure-systolic array: its contribution to the stage delay is near to 0.3 ns.

The final result is that the array with local communication is 30% faster, even though it is 40% bigger than the other topology. In Table 1, the main characteristics of both circuits are summarized.



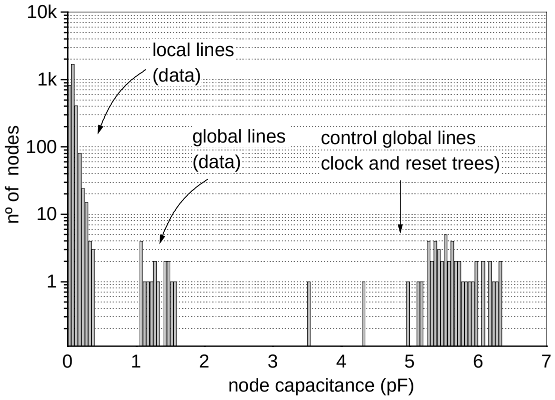


Fig.4: Wiring capacitance histogram of 16-bit cell-based multipliers: McCanny-McWinther (top) and Hatamian-Cash (bottom).

Parameter	McCanny-McWhirter	Hatamian-Cash
Throughput (typical delays)	154 MOPS	117 MOPS
Area pre-layout	10.54 mm^2	7.52 mm^2
Area nucleus (post-layout)	16.62 mm^2	11.79 mm^2
Area chip (post-layout)	26.7 mm^2	19.95 mm^2
Latency	47 T_{CLK}	32 T_{CLK}
Number of transistors	99910	71878
Number of nodes in the datapath	3986	3115
Number of 1-bit registers	2760	1584
Number of clock tree branches	29	16
Clock skew	1.64 pF	1.19 pF
Total wiring capacitance (datapath)	336.90 pF	252.86 pF
Total wiring capacitance (clock tree)	149.17 pF	105.15 pF
Worse wiring capacitance (datapath)	0.58 pF	1.55 pF

Table 1. Main characteristics of the cell-based Hatamian and McCanny prototypes.

Word length	McCanny-McWhirter			Hatamian-Cash		
	Number of CLBs	Number of 1-bit registers	Number of nodes	Number of CLBs	Number of 1-bit registers	Number of nodes
16	1199	2360	2395	832	1640	1675
14	916	1799	1830	637	1253	1284
12	671	1314	1341	468	918	945
10	464	905	928	324	635	658
8	296	572	591	207	404	423
6	164	315	330	116	225	240
4	71	134	145	51	98	109

Table 2. Resource occupation of the FPGA-based Hatamian and McCanny prototypes.

Word length	McCanny-McWhirter				Hatamian-Cash			
	Throughput, MOPS	Worst net delay,ns	Worst path, % route	Worst net fanout	Throughput, MOPS	Worst net delay,ns	Worst path, % route	Worst net fanout
16	84.1	9.3	79.9	3	46.3	19.0	88.4	32
14	71.6	11.6	83.2	1	46.3	19.1	88.5	28
12	97.6	7.9	77.2	3	53.5	16.0	86.6	24
10	93.3	8.2	76.7	3	54.5	15.9	86.4	20
8	119.3	5.8	70.0	3	78.2	10.3	80.5	16
6	127.9	5.2	67.7	2	81.5	9.8	79.7	12
4	118.7	6.0	71.9	1	97.7	7.7	75.7	8

Table 3. Performance and worst (slowest) net and path characteristics. FPGA-based Hatamian and McCanny prototypes

3.2 FPGAs

The Webgen tool, a Java-based parametrizable module generator [17], was utilized to synthesize a set of fine-grain-pipelined multipliers. Both Hatamian and McCanny arrays, with word lengths ranging from 4 bits to 16 bits were implemented on a Xilinx FPGA. The main results can be observed in Tables 2 and 3. Column labeled “worst path, % route” indicates the fraction of the stage delay corresponding to wiring. It can be up to 89 %. As occurred in the previous section, despite is greater area, the McCanny architecture is at least a 21% faster than its counterpart. This difference increases with the word length, reaching 81% for the 16 bits array. This speedup is much higher than the observed in the Standard Cells case for the same operand length.

The increased fanout in the global communications of the Hatamian array causes a degradation of the net delay. As two LUTs implement each PE, the fanout of the global communications is twice the word length. On the other hand, the maximum fanout for the McCanny array is 3, corresponding to two LUTs (one PE) and one flip-flop.

As happened in Standard Cells, the cost for an increased throughput is a higher area. For example, in the 16 bit case, the number of CLBs of the McCanny array is 44% higher than the one of the Hatamian array.

4. Conclusion

The experiments illustrate the effect of the direction of pipelining in all the characteristics of a pipeline array. Extra latency and chip size can be explained from a purely topological point of view. However, the final throughput depends on the balance between three effect of opposite sign. In one side, the wiring degradation produced by a more dense FPGA occupation (or a higher area in the cell-based prototypes) together with an increment of the skew. In the other side, the datapath delay reduction associated to the elimination of heavily loaded nodes. The work also evidences that regular structures present several advantages for technology benchmarking. The inherent high degree of extendibility and understandability facilitates the modeling of all the parameters of the circuits.

Acknowledges

Preliminary experiments of this work were founded by the CICYT of Spain, under the Project TIC92-0083 directed by Prof. Juan Meneses. The FOMEC program of the Word Bank granted the participation of Nelson Acosta and Elías Todorovich.

References

[1] G. Burrel (Ed.), "Crónica de la Técnica", pp. 524-525, Barcelona: Plaza & Janes Publishers, 1989,

[2] *Encyclopaedia Britannica,* 1996.

[3] Cotten L., "Circuit Implementation of High-Speed Pipeline Systems", *Proc. Fall Joint Computer Conference (AFIPS),* 1965.

[4] Anderson S., Earle J., Goldschmidt R., and Powers D., "The IBM system/360 model 91 floating point execution unit", *IBM Journal Res. Development*, Vol.11, pp. 34-53, January 1967.

[5] De Mori R. "Suggestion for an I.C. Fast Parallel Multiplier". *Electronic Letters*. Vol.5, Nº3, pp.50-51. Feb. 1969.

[6] Guild H., "Fully Iterative Fast Array for Binary Multplication and Addition", *Electronic Letters*, pp.263, Vol.5, Nº12, Jun. 1969.

[7] Hallin T. and Flynn M. "Pipeline of Arithmetic Functions". *IEEE Trans. on Computer*, pp.880-886. Aug. 1972.

[8] Deverell J. "Pipeline Iterative Arithmetic Arrays". *IEEE Trans. on Computers*, May 1975.

[9] Davio M. and Bioul G., "Efficiency of Pipelined Combinational Circuits", Digital Processes, Vol.4, pp.3-16, 1978.

[10] Jump R. and Ahura S. "Effective Pipeline of Digital Systems", *IEEE Trans. on Computers*, Vol. C-27, Nº9, pp.855-865, Sept. 1978.

[11] Hatamian M. and G. Cash. "A 70-MHz 8-bit x 8 bit Parallel Pipelined Multiplier in 2.5-um CMOS". *IEEE Journal of Solid-State Circuits.* Aug. 1986.

[12] Kung H. "Why Systolic Architectures". *IEEE Computer.* January 1982.

[13] McCanny J. and McWhirter J., "Completely iterative, pipelined multiplier array suitable for VLSI". *IEE Proc.* pp.40-46. Vol.129, Part G, Nº2. April 1982.

[14] Jáuregui J., *"Técnicas de Sincronización en Circuitos VLSI Standard Cells", PFC* (M.Sc. Thesis) Dept. of Electrical Engineering, *ETSI Telecomunicación, Universidad Politécnica de Madrid*. 1997.

[15] European Silicon Structures, *"ES2 ECPD10 Library Databook",* Doc. E01A09, 1993.

[16] Fishburn J., "Clock Skew Optimization", *IEEE Trans. on Computers*, Vol.39, Nº7, pp.945-951, July 1990.

[17] López-Buedo, S. *"Web-based Parametrizable Module Generator"*. Available at http://www.ii.uam.es/~eda.