The Effects of Physical Design Characteristics on the Quality of Synthesized Designs

BY

Alice C. Parker, Jen-Pin Weng,
Pravil Gupta and Agha Hussain

Technical Report 90-24

Electrical Engineering Systems

University of Southern California

Los Angeles, CA. 90089-0781

The Effects of Physical Design Characteristics on the Quality of Synthesized Designs*

Alice C. Parker, Jen-Pin Weng, Pravil Gupta and Agha Hussain
Department of Electrical Engineering
University of Southern California
Los Angeles, CA, USA

August 31, 1990

Abstract

This paper describes a set of experiments designed to show the effects of wiring area and delay and unused area on final chip characteristics. Floorplanning experiments which show the impact of wiring delay on cycle time are presented, along with an analysis of actual chip layouts. Finally, conclusions about the impact of scaling on the problem are drawn, and recommendations for future high-level synthesis programs are given.

1 Introduction

When high-level synthesis research began, there were no VLSI chips. Design was assumed to be done with a fixed set of available modules [1]. In at least one case, these modules were assumed to be TTL chips [2]. Such chips and modules had a fixed cost, and wiring delays between chips were minimal compared to the processing delays on chip. Power consumption could easily be computed as the sum of the power consumption of individual chips, and hot chips could be cooled with a heatsink. Now, however, we are faced with a situation where high-level synthesis programs must design datapaths and controllers to fit on one or more VLSI chips. For such chips, a large portion of the chip area is consumed by wiring. Wire delays are important, and hot spots on the chip can cause reliability problems. Given this situation, high-level synthesis programs must take a number of factors into account that were by and large ignored previously.

*Research described in this study was funded by the Semiconductor Research Corporation under SRC Contract Number 89-DJ-075.

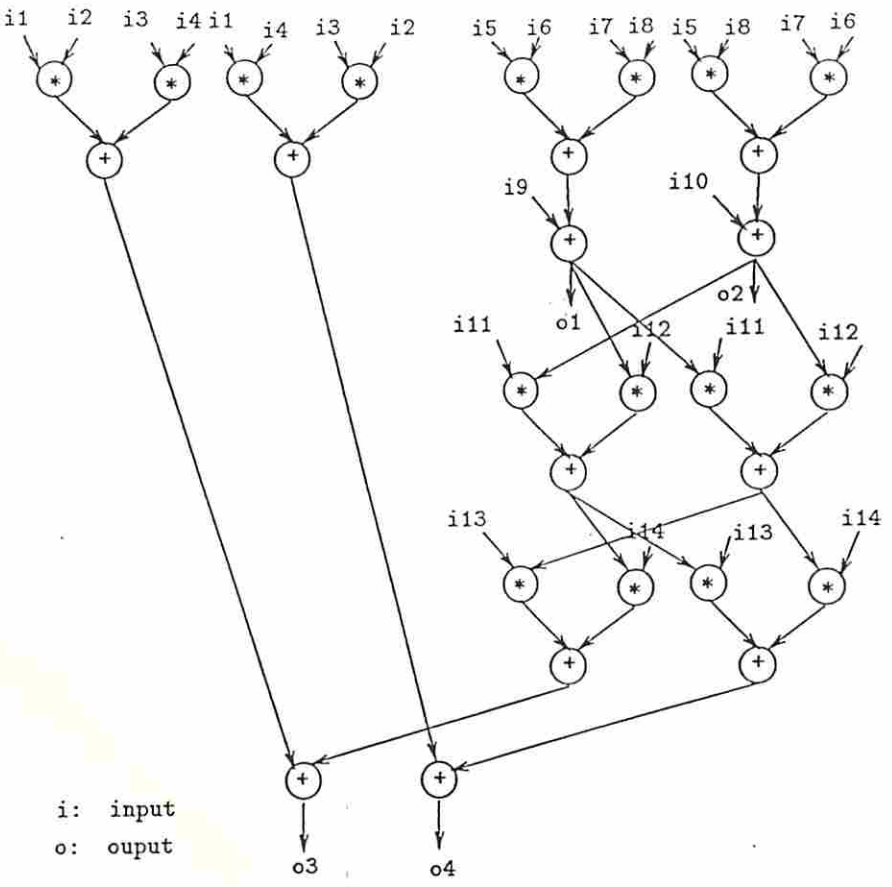

Figure 1: The AR Filter Dataflow Graph

This paper describes a set of experiments designed to determine the impact of some physical design parameters on the high-level synthesis process. Example filter specifications were used to synthesize a number of implementations at the register-transfer level with varying cost and performance. Cost was measured as total active area and performance as the delay through the active area of the chip.

One example chosen for this study was the AR lattice filter, a design with a clear cost-performance tradeoff curve at the register-transfer level. Designs with inner loops and conditional branches are expected to have less well-behaved tradeoff curves at the register-transfer level, and the physical impacts are expected to be worse. The dataflow graph for this AR filter is shown in Figure 1.

A second example used for this study was an FIR filter. The dataflow graph for this filter is shown in Figure 2.

In the first experiment we floorplanned two FIR-filter designs in order to minimize

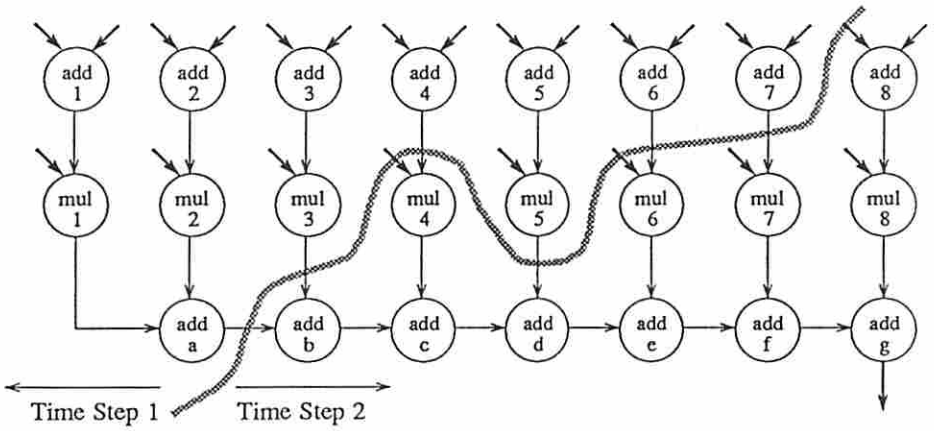

Figure 2: The FIR Filter Dataflow Graph

wiring delays along the critical path. The first design was done with the minimum number of resources to meet the schedule which the synthesis program produced. The second design was a similar design with an extra adder inserted in order to avoid long wiring delays along the critical path. We then measured the impact of the modification on chip area and critical path delay.

In a second experiment we took two of the AR-filter designs and we generated layouts using the Seattle Silicon Chipcrafter silicon compiler. We then assessed whether these layouts still fit our cost-speed tradeoff curve.

In the next two sections we describe the experiments and results, and then in the final section we draw conclusions.

2 Experiment 1: Wiring Delay

The FIR filter design was synthesized and floorplanned in order to minimize wiring delays.[1] The floorplan is shown in Figure 3. In this figure we can see the critical path highlighted with bold interconnections. A similar design was synthesized with an extra adder as shown in Figure 4. Here, an extra adder reduced wiring delays significantly as shown. The wiring delay decreased by 6%. In this case, the floorplan topology allowed an extra adder to be inserted with no area penalty. In other cases, the area penalty must be taken into account. A comparison floorplan was produced using a quadratic-based min-cut algorithm

[1]Registers and multiplexers were omitted in order to simplify the problem, but should be considered in an actual floorplan.

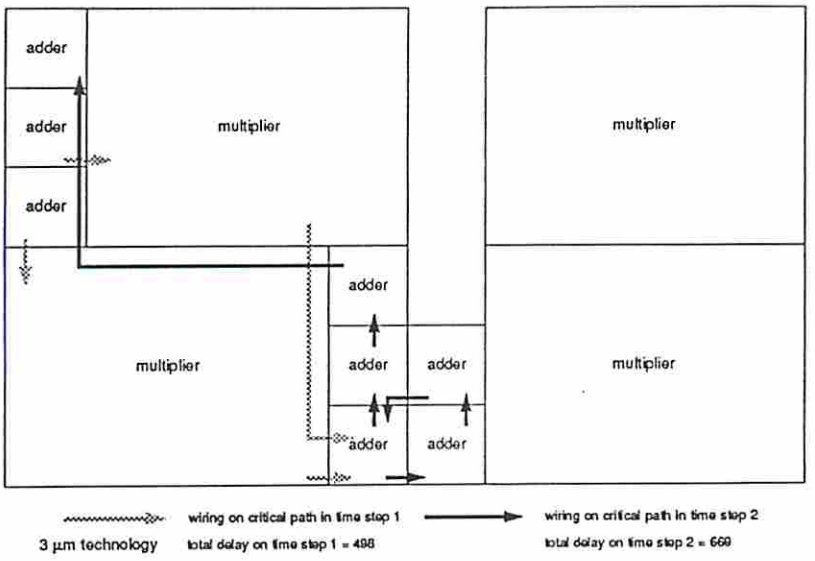

Figure 3: Floorplan of FIR Filter Showing Critical Paths

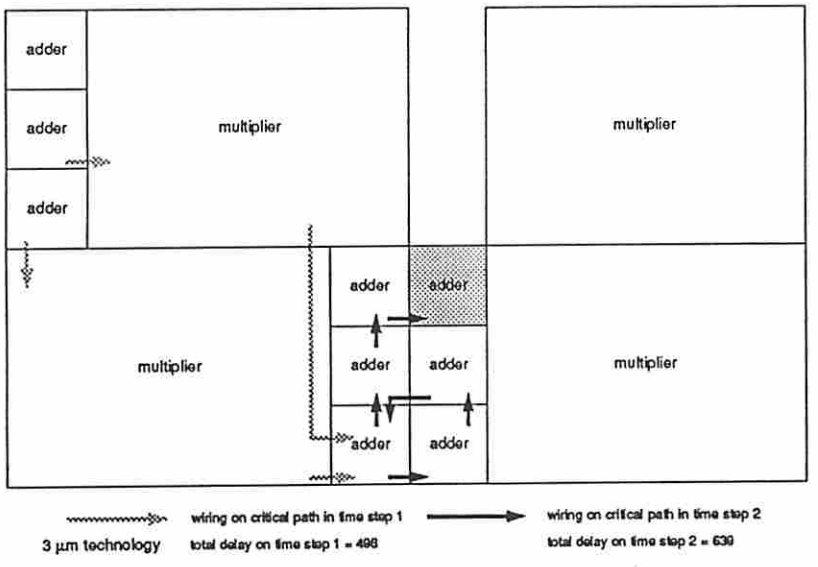

Figure 4: Floorplan of FIR Filter With Extra Adder Showing Critical Paths

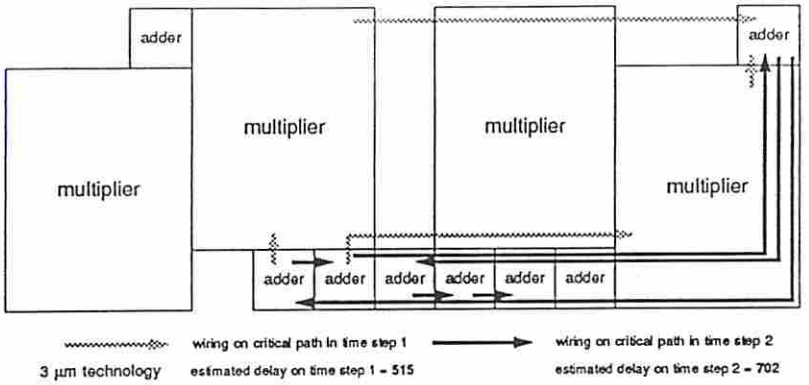

Figure 5: Floorplan Produced by Quadratic Assignment

with sequential assignment [4], and is shown here in Figure 5. Note the critical path wiring is extensive throughout the entire floorplan.

3 Experiment 2: Layout Measurements

For experiment 2, we took the most parallel and most serial designs of the FIR lattice filter and we hand translated our net list to the Seattle Silicon Chipcrafter format. We ran Chipcrafter with OKI 1.2 Micron, Twin-Well, Double-Layer Metal CMOS Ruleset and achieved layouts of 10,000 and 15,000 transistors respectively. The two layouts are plotted on the cost-speed tradeoff curve shown in Figures 6 and 7.

A table of information about the chip is shown in Table 1.

4 Conclusions

The floorplanning experiments, again performed with 3 micron feature sizes, show clearly the impact of wiring delay. As devices are scaled down, on a large chip, the wiring delays might exceed the functional delays. In this situation, the high-level scheduling process must be able to take into account the wiring delays.

The layouts showed a cost-performance tradeoff between the most serial and most parallel designs. However, the cheaper design was far slower than the faster one, but not proportionately smaller. Some intermediate designs with partial serialization might not be significantly smaller than the parallel design.

These observations point to the conclusion that high-level synthesis programs must somehow take into account the effects of layout if they are to produce designs of high

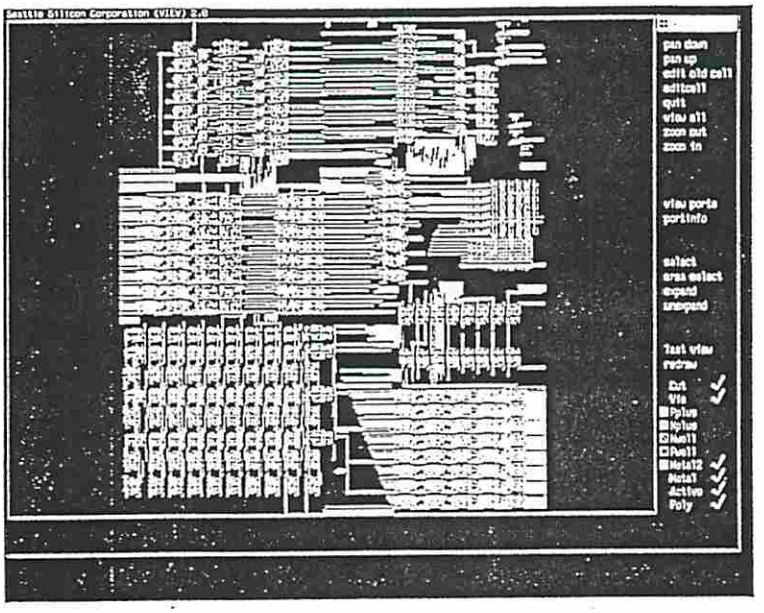

Figure 6: Layout of Serial Design

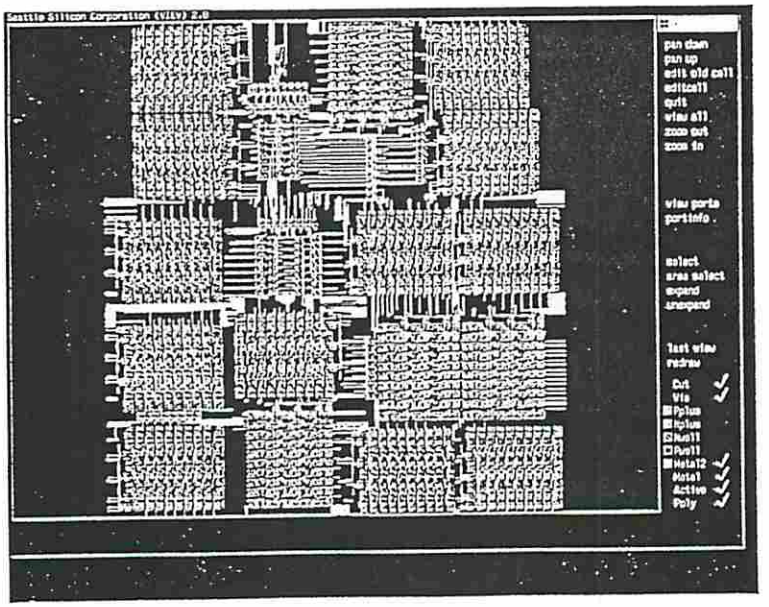

Figure 7: Layout of Parallel Design

	Parallel Design			Serial Design		
Type of Module	No. of Modules	Raw Cell Area μm^2	Functional Block Area μm^2	No. of Modules	Raw Cell Area μm^2	Functional Block Area μm^2
Multiplier	16	4562176.00	8381041.92	1	285136.00	523815.12
Adder	12	357408.00	425578.72	1	29784.00	34441.56
Register	2	72048.00	82268.00	5	180120.00	205520.60
Mux				7†	461056.00	618577.07
Controller		22000.00	32079.44		130067.00	187164.87
Total Chip Area	13519416.00μm^2			3107516.00μm^2		
Execution Time	80.66 ns			1183.88ns		

Table 1: Area Statistics for Actual Layouts

quality.

5 Acknowledgements

The authors would like to acknowledge Chao-Tsung Young and Kayhan Kucukcakar for their help in running these experiments, and the guidance provided by Michael McFarland during discussions of these ideas.

References

[1] M. Barbacci and D. Siewiorek. The cmu RT-CAD system: An innovative approach to computer aided design. In *American Federation of Information Processing Societies Conference Proceedings, Vol. 45*, pages 643–655. Amer. Fed. of Information Processing Societies, June 1976.

[2] L. Hafer and A. Parker. Automated synthesis of digital hardware. *IEEE Transactions on Computers*, C-31(2):93–109, February 1981.

[3] Y. Lai and S. Leinwand. Algorithms for floorplan design via rectangluar dualization. *IEEE Trans. on Computer-Aided Design*, Dec 1988.

†Consists of one 16-to-1 mux, one 11-to-1 mux, one 5-to-1 mux, one 4-to-1 mux and three 2-to-1 muxes.