

Technical Report NCSU-ERL-94-16
(presented at 1995 IEEE MCM conf.)
System Design Optimization for MCM

Paul D. Franzon Andrew Stanaski Yusuf Tekmen Sanjeev Banerjia

Department of Electrical and Computer Engineering
North Carolina State University
Box 7911, Raleigh, NC 27695

Ph. 919 515 7351, Fax. 919 515 7382, paulf@ncsu.edu

http://www2.ncsu.edu/eos/info/ece_info/www/centers/erl_home.html

Abstract

Many performance/cost advantages can be gained if a chip-set is optimally redesigned to take advantage of the high wire density, fast interconnect delays, and high pin-counts available in MCM-D/flip-chip technology. Examples are given showing the conditions where the cost of the system can be reduced through chip partitioning and how the performance/cost of a computer core can be increased by 81%.

from such a redesign. A number of issues related to such a redesign are investigated. First, the paradigm shift necessary for this redesign is explored. Second, some case studies presenting the cost advantages gained from partitioning for such a redesign are presented. Third, interconnect delay and power aspects are explored. Fourth, an example is given illustrating the potential performance/cost advantage to be gained. Finally, some risk issues are discussed and conclusions drawn.

1 Introduction

The combination of MCM-D substrates and flip chip area attachment provides a very high density packaging technology. However, there are very few commercial digital system designs that have migrated to this technology. The reason is quite simple – digital chips designed for single chip packaging (SCP) grossly under-utilize the potential of MCM-D/flip-chip technology to the point where such a technology is difficult to justify. Chips designed for single chip packaging are limited in pin-count (as pins on SCPs are expensive relative to solder bumps on an MCM), and are designed assuming that the off-chip delays are relatively large. The last assumption leads to large chips, with integrated memory and logic, to large current drivers and to large off-chip designed timing slacks. When transplanted onto an MCM-D, such chips underutilize the routing density on the MCM-D and, unless the clock can be profitably redesigned, gain no speed advantage. The only possible advantage becomes one of small size.

In this paper, we suggest that if the chips in a high performance system are redesigned specifically for MCM-D/flip-chip technology, then there are significant performance and cost advantages that arise

2 Interconnect-Driven Paradigm Shift

Chips designed for single chip packaging have the following design constraints:

- Pads are limited to edges and pins are expensive (up to 10 cents per pin, just for the package). As a result, I/O count is limited to the several hundred range.
- Between-chip delays are significantly larger than on-chip delays.

As a result the designer is forced to make the following compromises:

- Bus widths are often sub-optimal and many signals might be multiplexed.
- Chip size is Maximized.
- Memory is placed on the chip, with the logic. This memory is not fabricated in a (dense) memory process but in a modified logic process.
- High-current, high-power drivers are used.

- Allowances are made for significant off-chip delays in the timing design. Often, these delay allowances can not be reclaimed (for higher clock speed), if an MCM is used instead.

On the other hand, chips designed for MCM-D/flip-chip technology have very different constraints:

- The chips can have 1000's of I/Os. Thus bus-widths, etc. can be made more optimal.
- Inter-chip delay is comparable to and often faster than intra-chip delay.

As a result, the designer can enjoy the following advantages if the system is redesigned:

- Chips can be partitioned to improve chip yield and reduce cost.
- The technology mix can be optimized. In particular, RAM can be built in a RAM process, analog in an analog process etc.
- Busses (and other I/O count) can be optimally sized.
- Drivers can be down-sized to reduce power while meeting timing and noise constraints.

Essentially, the designer needs to move from a chip-size/pad-limited constrained design approach to an *Interconnect-Driven* design approach, where-in the primary constraint comes about when you run out of wires.

In order to conduct this redesign optimally, the MCM needs to be designed concurrently with the ICs therein. ie. Floorplanning, circuit design, test and other issues need to be decided concurrently for the MCM and IC.

3 Cost-Driven Partitioning

System design issues must be decided with system-wide, whole-life-cycle cost models. Components to such a cost model include the following [3]: (1) Design and prototyping costs; (2) Manufacturing costs, including component procurement, test, assembly, inventory, yield, repair, etc.; (3) Cost of product support, including warranty and non-warranty repair; (4) Appropriately proportioned overhead, including expenses shared amongst projects, marketing, etc. (and NOT just stated as a flat percentage on other costs).

The most appropriate approach to such modeling is to use a Technical Cost Model [6]. We are in the process of constructing such a model. However, in

this paper, the model used is at a slightly higher level of abstraction (in order to make it more useful to others).

For illustration, we start with a single chip implementation of a microprocessor (assuming an area of 17×17 mm² and 400 signal I/Os) and compare it with a number of multichip implementations.

The cost models used here contain the following assumptions:

1. A six-inch, sub-micron silicon wafer costs \$2,000 to \$5,000 in volume. (Why the wide range? This covers a number of technologies, and partially reflects the difference between internal cost for vertically integrated companies and procurement cost for design companies.)
2. 35% of the original die area is used for memory (e.g. cache for a CPU).
3. The defect density ranges from 0.9 defects/cm² (immature process) to 0.3 defects/cm² (mature process). Poisson yield models are used with random defects only. This on-chip memory can withstand two manufacturing defects, improving the yield accordingly.
4. SRAM-only, logic-only, and logic-SRAM processes cost the same per wafer. SRAM implemented in an SRAM-only process is two times denser than SRAM implemented in a logic-SRAM process.
5. Chip test cost is 10 cents per chip signal I/O for packaged die and 15 cents per chip signal I/O for bare die. (This is the most questionable assumption. Test cost is a function of many parameters including logic count, I/O count, coverage, use of BIST, etc.)
6. A ball grid array costs \$50.
7. A tested 9 cm² MCM-D substrate costs \$94 and \$20 to package.
8. 2% of the chips are discovered to be faulty after assembly into the MCM. Faulty MCMs are repaired at a cost of \$30 plus the cost of the replacement part. (A preliminary analysis indicated that for the high value MCMs described here, repair was cheaper than discarding the MCM.)
9. When a chip is divided an additional 1 mm² is added for clock circuits, etc. and 0.005 mm² is added for each driver. 30% of drivers are replicated when partitioned. In a flip-chip technology, power and grounds are essentially free.

Partit.	logic size (mm)	mem. size (mm)	cost
LM	13.8×13.8	7.2×7.2	\$881
L2M	9.7×9.7		\$471
L2M2		5.1×5.1	\$477
L3M	8×8		\$356
L4M	6.9×6.9		\$394
L5M	6.2×6.2		\$393

Table 1: Partitioning results for a 17×17 die built with \$5,000 wafers at a defect density of 0.9 defects/cm². Single chip cost is \$963.

- Additional ‘overhead’ costs associated with increased chip-count are ignored for now. This ‘overhead’ includes additional prototyping costs for each additional chip, and additional inventory costs for each additional component, etc.

The first question we address in this paper is that of when does it make sense to partition onto an MCM?

Table 1 shows results obtained for the case for a 17×17 mm. die, with a wafer cost of \$5,000 and a defect density of 0.9 defects per square cm (the most ‘pro-MCM’ case). The packaged cost of the 17×17 die was \$963. The sizes of the partitioned die are also shown (as square die solely for convenience). The notation ‘LxMy’ means that the logic has been partitioned into x die and the memory into y die.

The results shown in Table 1 show that, under the conditions presented, partitioning is very cost-advantageous but that it does make sense to partition into die as small as 7×7 mm². Finer partitioning would only make sense if cost advantages were gained elsewhere; for example, if the smaller die could be used in many designs.

The results of a larger number of studies are given in Tables 2 and 3. Some general conclusions can be drawn from these studies:

- If the choice is between a large die in a Single Chip Package (SCP) and a partitioned die set on an MCM, then the partitioned set only makes sense if the original die has a particularly high manufacturing cost. A high cost die occurs when some combination of high wafer cost, large die area and high defect count takes effect. Such conditions occur often during the initial life of a performance driven chip.
- If an MCM is justified for another reason (e.g. size reduction, mixed signal advantages, etc.), then partitioning might make sense for a die

wafer cost, defect density, SCP cost, MCM cost		
Partit.	MCM Cost	
	15 c/pin test cost	10 c/pin test cost
\$5,000; 0.9; \$963; \$1024		
LM	\$881	\$768
L2M	\$471	\$411
L3M	\$410	\$356
L4M	\$394	\$340
L5M	\$393	\$336
\$5,000; 0.3; \$327; \$388		
LM	\$377	\$340
L2M	\$326	\$291
L3M	\$323	\$285
L4M	\$330	\$289
\$2,000; 0.9; \$520; \$581		
LM	\$622	\$511
L2M	\$366	\$306
L3M	\$330	\$277
L4M	\$325	\$271
\$2,000; 0.3; \$192; \$253		
LM	\$287	\$250
L2M	\$262	\$227
L3M	\$266	\$229
L4M	\$276	\$235

Table 2: Results of a range of partitioning studies for a 17×17 mm² die.

wafer cost; defect density; SCP cost; MCM cost		
Partit.	MCM Cost	
	15 c/pin test cost	10 c/pin test cost
\$5,000; 0.9; \$244; \$305		
LM	\$349	\$300
L2M	\$292	\$252
L3M	\$289	\$248
L4M	\$296	\$252

Table 3: Partitioning results with a smaller starting die size - 12×12 mm².

IC area cm ²	Wafer cost	defect density	SCP \$	L	M	eff. area cm ²	MCM \$
2.89	\$5,000	0.9	\$963	8	4	9.4	\$934
2.89	\$2,000	0.9	\$519	5	2	5.2	\$480
2.25	\$2,000	0.9	\$308	3	2	3.1	\$308

Table 4: Examples of the effective area increase possible through partitioning.

larger than a square centimetre, even if the percentage of memory on that die is small.

- The relative advantage of partitioning onto an MCM increases somewhat whenever any of the following conditions occur: (1) The extra test cost associated with MCM use is reduced; (2) the relative percentage used for memory increases; and (3) the relative cost of the MCM package is reduced. However, the effects of these factors are small. The existence of an expensive die is the main motivator for partitioning.

However, there is a very valid alternative view of the value of this cost-driven partitioning paradigm. The alternative view is expressed as the second question addressed in the form of a question: Instead of reducing cost, can partitioning be used to obtain more total silicon area at the same cost?

Some examples are given in Table 4. For the larger $17 \times 17 \text{ mm}^2 = 2.89 \text{ cm}^2$ chip, almost twice the effective silicon area can be obtained for the same price as the original IC. For the smaller $15 \times 15 \text{ mm}^2 = 2.25 \text{ cm}^2$ chip in a low-cost process, the effective silicon area increase is only 10%. This additional area can be used to improve performance. Before, presenting a case study exploring how performance can be improved, we will give results demonstrating the relative performance of on-MCM and on-IC interconnections. Unless, the performance of the two interconnections are comparable, the potential performance advantages to be gained through partitioning are reduced.

4 Interconnect Circuit Delay and Power

In this section, we show that on-chip and off-chip delays are comparable in an MCM environment and explore some of the tradeoffs in speed, driver area and power dissipation.

Simulation results for three of the drivers described in Table 5 are shown in Figures 1, 2 to 3. In each

Area (μm^2)	0.8 micron		1 micron	
	conv.	diff.	conv.	diff.
small	2010	5540	3141	8656
medium	3348	7167	5231	11198
large	4284	9324	6694	14568

Table 5: Driver layout areas.

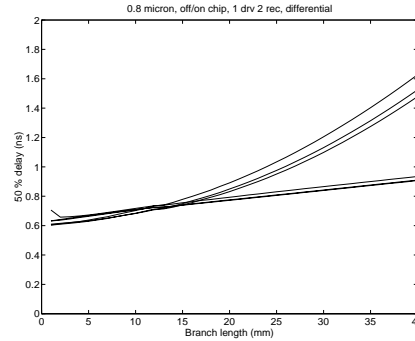


Figure 1: 50 % delay vs branch length.

figure, the curved lines represent on-chip delays for the different driver strengths and the straighter lines are for the off-chip delays. ‘Branch length’ is the length to the farthest receiver. It can be seen that for the lengths of main interest (less than 10 mm), on- and off- chip delays are comparable, more so for sub-micron than for micron technology drivers. For larger distances, off- chip delays are smaller than on-chip delays due to the large resistance of long on-chip interconnect. The power dissipation of the different sized drivers was also simulated. It was found that the power dissipation varied by up to 100% between the small and large drivers. Often large amounts of power can be traded for small amounts of delay in the MCM environment.

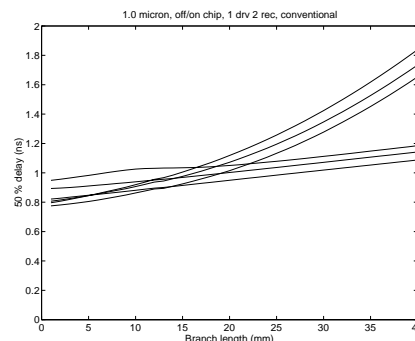


Figure 2: 50 % delay vs branch length.

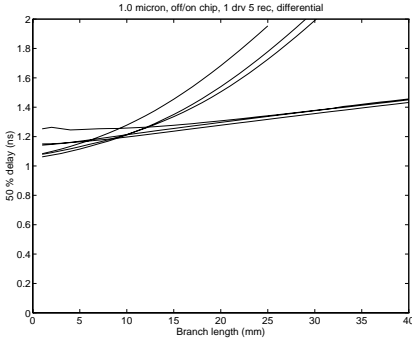


Figure 3: 50 % delay vs branch length.

5 Illustrative Example – A Computer Core

In this example, we illustrate the performance/cost advantages possible by conducting a paper redesign of a computer core for MCM-D/flip-chip technology. The baseline model is one very similar to DEC Alpha 21064 [2]. The 21064 is a two-issue superscalar design running at 200 Mhz and capable of a peak of 400 MIPs. Using a modified version of Mike Johnson’s `ssim` simulator [5], simulations were run using several SPECint program¹ as benchmarks to gauge the performance of the baseline model, using cycles per instruction (CPI) as the metric. CPIs for the baseline model are shown in Table 6.

5.1 Performance Optimizations

An MCM frees us from having to pack all of the features and, hence, all of the transistors onto only one die. We thus have more latitude in increasing the functionality and size of individual sections of logic and memory because we are not constrained to placing all of the functionality onto only one die. With respect to processor organization, we ran simulations across a variety of microarchitecture configurations in order to have empirical data to justify our design decisions; this is similar to the strategy presented in Johnson [4]. Our results are available in a technical report [1].

We assume that our optimized design will run at the same clock rate as the baseline processor. Initially, we concentrated on enhancing the issue/decode logic and adding register renaming capabilities. The baseline model used an in-order issue and completion algorithm; we changed this to a more aggressive out-of-order issue and completion strategy. We also changed the issue/decode rate to 4 instructions

¹ Several of the SPECint programs were not compiled with full optimizations, due to incompatibilities with the compilers on our system. Please contact the authors for details.

Simulated program	CPI	
	Baseline design	Optimized design
026.compress	1.87	1.41
023.eqntott	2.09	1.34
022.li	2.29	1.36
008.espresso	1.99	1.39
Geom. Mean	2.05	1.37

Table 6: Simulated CPIs for unoptimized and optimized processors.

per cycle and imposed no restrictions on the combinations of instructions can be issued together. The execution model now uses reservation stations as introduced by the Tomasulo model. The number of functional units is kept the same (we concentrate on *supplying* instructions at a pace that utilizes the functional units at hopefully near 100%).

Speculative execution support hardware was increased: the optimized design can pursue multiple branch paths and also pursue multiple branch paths per cycle. Branch prediction accuracy was greatly increased through the use of an enlarged branch target buffer, up from 64 entries to 512 entries. The first-level caches were increased from 8 KB to 64 KB each. Other memory improvements were also made.

Our modified version of `ssim` was used to estimate the performance of the optimized model. The CPIs for these simulations are shown alongside those for the baseline model in Table 6.

As in the baseline model, a two-level cache was used. However, the fetch size between the two cache levels was increased from 128 to 256 bits. (This was the largest increase possible with off-the-shelf memory parts – with wider memory parts, the fetch size could be increased further.) This improvement is *not* reflected in Table 6. Based on available data [7], we estimate the wider fetch size would lead to an additional 0.1 CPI improvement. *The total performance improvement then is 61%.*

As we are no longer limited by the die size of a single chip, we can add extra functionality that often is not considered for most designs. For example, we could consider multimedia support [8], hardware memory disambiguation, high-bandwidth I/O, etc.

5.2 Cost Optimization

For the baseline design, at \$5,000 per wafer and 0.9 defects/mm², the yielded, packaged cost of the 15×15 mm² CPU chip is \$520, and the sixteen second level cache chips, \$80 each. On a printed circuit board, this

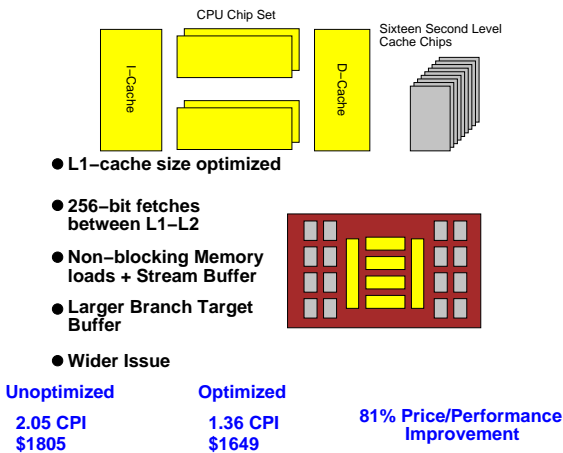


Figure 4: Optimized Computer Design on an MCM.

assembly would cost about \$1805. On a multichip module, it would cost \$1886.

With the performance enhancements described above, we estimate a requirement for a total silicon area of $18 \times 20 \text{ mm}^2$. We partitioned this component into four logic units and two first-level cache SRAMs for a new MCM-packaged CPU price of \$369. With the sixteen second level cache chips, the on-MCM CPU core would cost \$1649, a 13% price improvement. *The total performance/price improvement is thus 81%.*

6 Risk Issues

There are a number of risk issues involved in the redesigns proposed here: (1) The optimally designed chip-set assumes a reasonably close parametric matching of the different ICs. This requirement can increase test complexity. (The ICs must be binned.) (2) Test complexity might also be increased by the resulting requirement to test partial chip-sets. (3) First-pass success is required on the module design. (4) The redesigned chips can only be implemented with MCMs. (5) In some cases (such as the first level cache above), non-standard SRAM parts will be needed from SRAM manufacturers.

7 Conclusions

In high performance, applications optimally redesigning the chip set for MCM-D/flip-chip technology might lead to advantages that justify the use of that technology. The justification is particularly strong when the single-chip packaged chips are expensive to manufacturer. In this case, repartitioning the large chip into higher-yield smaller die and intercon-

necting them on an MCM (where the off-chip penalty is very small) will lead to lower cost systems. We gave an example in which a CPU core is redesigned and the final design provides 61% more performance at 13% less cost.

The cost model spreadsheet, and several relevant reports can be found on our WWW server listed at the top of this article.

Acknowledgements

The authors wish to thank the following funding and support sources: ARPA under contract DAAH04-94-G-003-P2, NSF under grants MIP-901704 and DDM-9215755, NSF for Dr. Franzon's NSF Young Investigator's Award, Cadence Design Systems, Tektronix and Hewlett Packard.

The authors also wish to thank the following individuals for their thoughts on this topic: Robert Frye, Thad Gabara, Eric Bogatin, David Salzman, Howard Sachs, Vern Bretheor, David Tuckerman, David Carroll, Real Pommerleau, Wentai Liu, Frederico E. de los Santos, Evan Davidson, Donald Benson, Pat Sullivan, Frank Swiatowec, and Bob Parker.

References

- [1] Sanjeev Banerjia, Eric Schweitz, Mark Vilas, and Saurabh Misra. Performance Prediction for Superscalar Processors. Technical Report NCSU-ERL-94-18, Department of Electrical and Computer Engineering, North Carolina State University, November 1994.
- [2] Daniel W. Dobberpuhl et al. A 200-Mhz 64-b Dual-Issue CMOS Microprocessor. *IEEE Journal of Solid-State Circuits*, 27(11):1555-1567, November 1992.
- [3] P.D. Franzon. MCM package selection: A system's need perspective. In D.A. Doane and P.D. Franzon (ed), editors, *Multichip Module Technologies and Alternatives: The Basics*, chapter 3. Van Nostrand Reinhold (New York), 1992.
- [4] M. Johnson. *Superscalar Microprocessor Design*. Prentice Hall, 1991.
- [5] Mike Johnson and Michael D. Smith. **ssim**: A Superscalar Simulator. Available via ftp from velox.stanford.edu in pub/ssim/manual.ps.
- [6] L.H. Ng. Mcm package selection: Cost issues. In D.A. Doane and P.D. Franzon (ed), editors, *Multichip Module Technologies and Alternatives*:

The Basics, chapter 4. Van Nostrand Reinhold (New York), 1992.

[7] Steven A. Przybylski. *Cache and Memory Hierarchy Design*. Morgan Kaufmann, 1990.

[8] Peter Wayner. SPARC Strikes Back. *BYTE*, 19(11):105–112, November 1994.