University of Tennessee, Knoxville

TRACE: Tennessee Research and Creative Exchange

Doctoral Dissertations

Graduate School

5-1999

Early analysis of VLSI systems with packaging considerations

Karthikeyan Ramamurthi

Follow this and additional works at: https://trace.tennessee.edu/utk_graddiss

Recommended Citation

Ramamurthi, Karthikeyan, "Early analysis of VLSI systems with packaging considerations. " PhD diss., University of Tennessee, 1999.
https://trace.tennessee.edu/utk_graddiss/8907

This Dissertation is brought to you for free and open access by the Graduate School at TRACE: Tennessee Research and Creative Exchange. It has been accepted for inclusion in Doctoral Dissertations by an authorized administrator of TRACE: Tennessee Research and Creative Exchange. For more information, please contact trace@utk.edu.

To the Graduate Council:

I am submitting herewith a dissertation written by Karthikeyan Ramamurthi entitled "Early analysis of VLSI systems with packaging considerations." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

Donald W. Bouldin, Major Professor

We have read this dissertation and recommend its acceptance:

Suzanne Lenhart, Robert Bodenheimer, Chandra Tan

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a dissertation written by Karthikeyan Ramamurthi entitled "Early Analysis of VLSI systems with Packaging Considerations." I have examined the final copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

Donald W. Bouldin

Dr. Donald W. Bouldin, Major Professor

We have read this dissertation
and recommend its acceptance:

Suzanne Lenhart

Robert Bodenheimer

Chandra Tan

Accepted for the Council:

Cwminkel

Associate Vice Chancellor
and Dean of The Graduate School

EARLY ANALYSIS OF VLSI SYSTEMS WITH PACKAGING CONSIDERATIONS

A Dissertation

Submitted for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Karthikeyan Ramamurthi

May 1999

ACKNOWLEDGMENTS

I express my deep gratitude and regard for my major professor Prof. Don Bouldin. He gave me very helpful suggestions and constantly motivated me during course of this dissertation work and was very supportive during the last three years while I continued to work on it after I started to work on my full time job at Intel. I sincerely thank Prof. Robert Bodenheimer for serving on my committee and for his valuable suggestions. He was very kind to continue to serve on my committee even after his retirement from teaching from UTK. I would like to thank Prof. Suzanne Lenhart for being on committee and for her comments and suggestions during the course of this research work. I would also like to thank Dr. Chandra Tan who agreed to serve on my committee on a short notice and for his suggestions while I was doing this research at UTK. I am very grateful to him for all his help he offered me to get all the work necessary for my defense done on time to meet the deadline dates set by UTK.

I would like to thank Dr. Peyman Dehkordi for his valuable ideas and comments which helped me to accomplish the major goals of this dissertation work. I would like to express my thanks and gratitude to Prof. Ranga Vemuri, Dr. Ram Vemuri and Dr. Nand Kumar of University of Cincinnati, for giving us a part of the software used in this dissertation work. I acknowledge with thanks the comments and support provided by Dr. Peter Sandborn and Ashish Parikh of Savantage, Austin. I value very much the suggestions by Dr. Howard Davidson of Sun Microsystems.

This thesis work would not have been possible without the support of the start-up Microelectronic System Research team at UTK. I am very much grateful to Kanagaraj and Barry Tolnas for their help in some of the implementation part of the work. I would like to thank Tim Powell for his help in the delay and power estimation models used in this work. I acknowledge the help of Quinn Devine for his size estimation models. I would also like to thank Shane Frank for his help with the schematic capture of the MicroSparc Design. I also thank Subhomoy for his comments and help.

This work would not have been possible without the help of Winston Holmes and Nina Hathaway our Unix System Administrators. I would like to thank the staff of the Interlibrary Loan Services for getting the necessary reference material for this work. I would also like to thank the administrative staff of the EE department for their assistance with all the paper work necessary for completing this dissertation.

This work was sponsored in part by the Advanced Research Projects Agency under ARPA order A879 and monitored by the Army Research Office under grant DAAH04-94-G-004to the University of Tennessee.

I would like to thank Mr. Sherif Sweha, my design manager at Intel Corp. Folsom, CA, who was very supportive of me which greatly helped me to complete this dissertation work. I would also like to thank Mr. Vishram Dalvi my manager at Intel Corp. for his supportive suggestions to get my dissertation work done. I would also like to thank Intel Corp. for reimbursing my tuition fees during the

time I had to enroll as a part time student at university and also allowing me to use some of their computer resources while writing this dissertation. I would also like to thank all my friends at Intel Corp. who constantly motivated to get this work completed. I would like to thank all my friends and relatives who had helped me realize my dream. I thank my brother and sisters for their encouragement. I thank my parents for their love and support and for the sacrifices they have made. I would also like to thank my grand aunt who taught me that the purpose of life is a life of purpose. Above all I would like to thank my Master and God, the source of every thing I have accomplished.

ABSTRACT

There is an explosive growth in the size of the VLSI (Very Large Scale Integration) systems today. Microelectronic system designers are packing millions of transistors in a single IC chip. Packaging techniques like Multi-chip module (MCM) and flip-chip bonding offer faster interconnects and IC's capable of accommodating a larger number of inputs and outputs. The complexity of today's designs and the availability of advanced packaging techniques call for an early analysis of the system based on estimation of system parameters to select from a wide choice of circuit partitioning, architecture alternatives and packaging options which give the best cost/performance.

A procedure for the early analysis of VLSI systems under packaging considerations has been developed and implemented in this dissertation work. The early analysis tool was used to evaluate the inter-relationship between partitioning and packaging and to determine the best system design considering cost, size and delays. The functional unit level description of a 750,000-transistor MicroSparc processor was studied using an exhaustive search technique. The early analysis performed on the MicroSparc design suggested that the three chip multi-chip design using flip-chip IC's interconnected on a MCM-D substrate is the most cost effective. An early bond pitch analysis performed using the tool concluded that a 250-micron bond pitch is the best choice for the multi-chip MicroSparc designs. The tool was also used to perform an early cache analysis which showed that the use of separate memory and logic processes made it feasible to design the

MicroSparc design with larger cache sizes than the use of a combined logic and memory process. The designs based on the separate processes gave equivalent or better performance than the design candidates with smaller cache sizes. Future extensions of the procedure are also outlined here.

TABLE OF CONTENTS

CHAPTER PAGE

1. INTRODUCTION . . . 1

1.1 Early Analysis . . . 2

1.2 Motivation . . . 3

1.3 Problem Statement . . . 5

1.4 Goals and Expected Contributions . . . 7

2. BACKGROUND . . . 8

2.1 Related Work . . . 8

2.2 Packaging Techniques . . . 9

2.2.1 Bonding Technology . . . 9

2.2.2 Interconnect Techniques . . . 13

2.3 Partitioning . . . 17

3. METHODOLOGY . . . 23

3.1 Early Analysis Considering System Cost . . . 23

3.1.1 Early Analysis at Functional Unit Level . . . 23

3.1.2 Early Analysis at RT Level . . . 38

3.2 Early Analysis Considering System Performance . . . 43

4. IMPLEMENTATION . . . 47

CHAPTER PAGE

4.1 Early Cache Analysis 47

4.2 Early Analysis at RT Level 50

4.3 Early Analysis Considering System Performance 52

5. RESULTS 55

5.1 Early Cache Analysis 55

5.1.1 Early Bond Pitch Analysis for Area-Array IC 64

5.2 Early Analysis at RT Level 69

5.3 Early Analysis Considering System Performance 72

6. SUMMARY, CONCLUSIONS AND FUTURE WORK 75

6.1 Summary 75

6.2 Conclusion 76

6.3 Future Work 77

BIBLIOGRAPHY 78

VITA 82

LIST OF FIGURES

FIGURE **PAGE**

1.1 *The inter-relationship of the key system performance parameters* . 4

2.1 *Wire Bonding* 10

2.2 *Flip-chip Bonding* 12

2.3 *MCM-C Interconnect Technique (after [2]).* 14

2.4 *MCM-L Interconnect Technique (after [2]).* 16

2.5 *MCM-D Interconnect Technique (after [2]).* 18

2.6 *(a) Graph Model for Partitioning (b) Clustering of Components during Partitioning (c) Physical Representation* 20

3.1 *(a) I/O limited cache die. (b) Cache size increased without increase in the die size.* 25

3.2 *The MicroSparc Functional unit block diagram* 26

3.3 *Memory Hierarchy* 27

3.4 *Critical path in a logic network* 29

3.5 *Interconnect in Monolithic chip and chips on MCM* 30

3.6 *Interconnect model for delay evaluation considering the distributed RC effects* 31

3.7 *The chip to chip interconnect and transmission line model for the interconnect* 34

3.8 *The Bond Pitch in Area-Array IC* 36

FIGURE **PAGE**

3.9 *Escape Routing for 16×16 Area-Array Bond Pad IC after [24]* . . 36

3.10 *Levels of Design Representation and Partitioning* 39

3.11 *Simulated Annealing algorithm for Partitioning* 42

3.12 *The flow diagram of the Early Analysis considering System Performance* 44

3.13 *The logic and net delays in the instruction fetch pipe line stage* . . 45

3.14 *The net delay in the fetch cycle using RC analysis* 46

4.1 *Flow Diagram of the Early Cache Analysis* 48

4.2 *Flow Diagram of the Early Analysis at RT level* 51

4.3 *The approach to evaluate the system delay performance* 54

5.1 *Typical Placement of Dies on the MCM Substrate* 58

5.2 *The Relative Performance of the Candidate Designs* 61

5.3 *The Relative Cost of the Candidate Designs* 62

5.4 *The Relative Cost/Performance Index of the Candidate Designs* . 63

5.5 *The Relative Comparison of System Cost for for different values of Bond Pitch* 65

5.6 *The Relative Comparison of Number of Chips in the System for different values of Bond Pitch* 66

5.7 *The Relative Comparison of System Size for different values of Bond Pitch* 68

5.8 *The Relative Comparison of System Power for different values of Bond Pitch* 70

FIGURE **PAGE**

5.9 *The Relative Comparison of System Noise for different values of Bond Pitch* 71

5.10 *Cost vs Performance Results of the Comparative Early Analysis for the MicroSparc Design.* 73

LIST OF TABLES

TABLE PAGE

3.1 *The time taken to generate all possible partitions* 40

5.1 *The real CPI values considering the memory stalls due to cache-memory hierarchy.* 56

5.2 *The cycle time values of the MCM designs* 59

5.3 *The Cost/MIPS performance index for the candidate designs* . . . 59

5.4 *The Cost/Performance Comparison of different Design Choices* . 60

5.5 *Results of Partitioning using Simulated Annealing at the Functional Unit level* 69

5.6 *Results of Partitioning using Simulated Annealing at the Register Transfer Level* 72

5.7 *Comparative results of the Early Analysis of MicroSparc for different design choices and processes.* 72

CHAPTER 1

INTRODUCTION

The size of VLSI systems are following an explosive growth. Single chip IC's (Integrated Circuits) packed with millions of transistors are commercially available in today's market. The design of these chips starts with a conceptual specification. At this stage the designer perceives the chip as simple functional blocks or units interconnected to one another. From this step the design proceeds through several steps before the chip is realized as the interconnection of several million transistors. The design evolves from a simple, less detailed specification at the beginning to a complex highly detailed description before the chip is fabricated.

Packaging techniques like Multi chip module (MCM) and flip-chip bonding offer faster interconnects and IC's capable of accommodating a larger number of inputs and outputs. To design better cost/performance systems, the various packaging options, cache size alternatives and circuit partitioning choices have to be evaluated concurrently during early stages of the VLSI system design cycle. This design procedure is called *early analysis*.

1.1 Early Analysis

The goal of early analysis is to emulate the detailed analysis which requires very exact design specification. The early analysis tool just needs a simpler and less detailed input design specification. The early analysis tools used to evaluate designs under packaging considerations normally depend upon the following user input:

- Design specification or description
- Packaging technology definition
- Evaluation strategy for evaluating the match between design and package

The early design description consists of net-list of functional units. Each functional unit in the design description will have attributes such as area, type of function (logic or memory) and power consumption. The packaging definition specification details issues like number of layers, wiring rules etc. The evaluation strategy is decided by the packaging and system designers. The issues in the strategy include partitioning of the design into several chips, the cache sizes and packaging constraints. The early analysis tool along with high level design specifications helps the package and VLSI system designers to find a common acceptable design point.

The system designers specify the best choice of partitioning and cache sizes while the package designers choose the best package. Issues like routing congestion is affected by the choice of packaging. Partitioning and packaging can

have impact on the package delay and hence system cycle time. Normally for a given packaging technology reducing the cycles per instruction (CPI) increases the cycle time and vice versa [1].

1.2 Motivation

Thus the aim of the early analysis of the VLSI system at a high level design description is to develop an understanding of the inter-relationship between the key system performance parameters and capabilities as shown in Figure 1.1 that affect both package and system designers. The basic characteristics of the parameter values are determined early in the design cycle even though the actual values change as the design progresses.

The key electrical parameters affecting the wiring rules are decided by the properties of packaging materials, especially the dielectric constant. The number of I/Os that can be accommodated also depend upon the package material. About 50% or more of the design cost is determined during the first 10% of the design cycle; therefore, there is a need to perform early analysis before committing to the design choices [2]. A significant mismatch between the package and the design might need a redesign of the system or package or both. It is very evident that the cost incurred to accommodate changes will be the least if the necessary changes are found in the early stages of the design cycle.

The complex relationships between packaging options, speed and cost impose conflicting demands on the system and package designers of VLSI systems. For

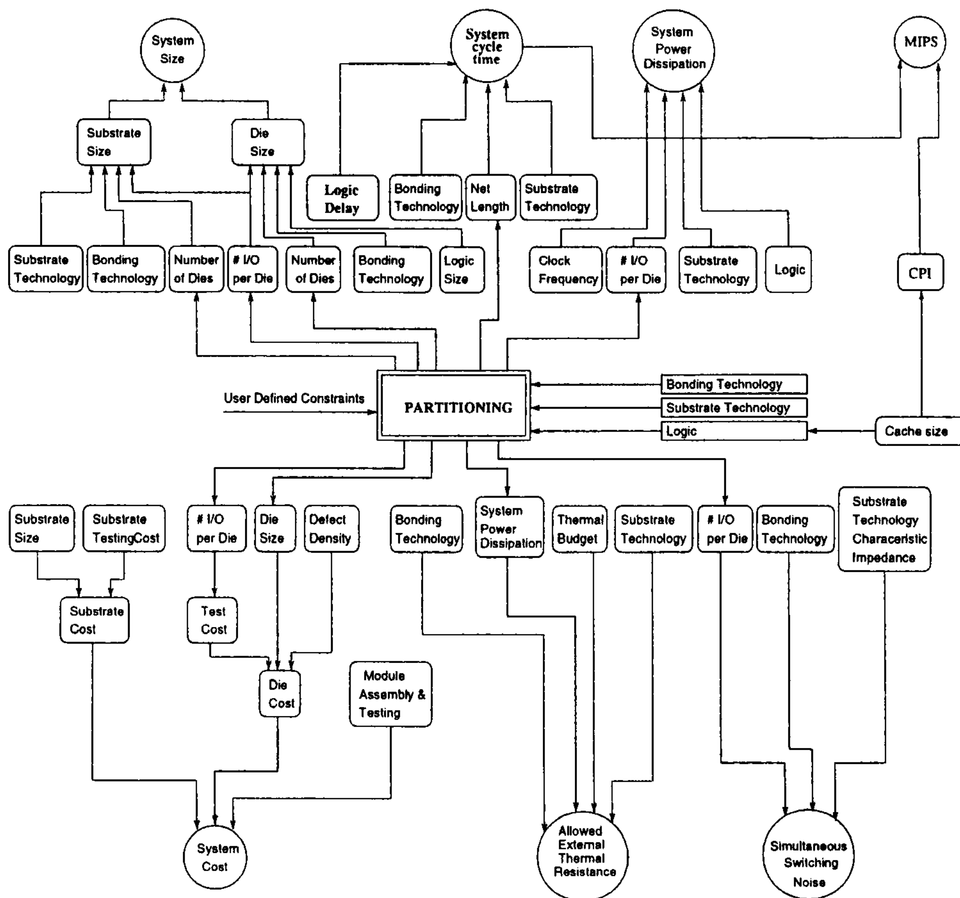

Figure 1.1: *The inter-relationship of the key system performance parameters*

example, tight interconnection pitch is desirable for maximum wiring capacity but this increases the simultaneous switching noise due to the close coupling of the interconnect lines [1]. There should be a careful balance between the choices made by the package and VLSI system designers for a successful product and *early analysis* plays a major role in achieving this goal. The early analysis tool allows the MCM packaging and VLSI system designers to consider the various cost and performance tradeoff early in the design cycle to design the best cost/performance VLSI system.

1.3 Problem Statement

Current VLSI system design methodology consists of sequential design steps. The circuit and logic designers play a major role at the top of the design cycle and package and manufacturing engineers at the bottom part of the design cycle. The chips are designed with default packaging options like the wire-bond (i.e. peripheral) pads. There have been great strides made in the area of packaging techniques like flip-chip bonding and area-array pads in the recent years.

The trend in the IC process technology is to pack millions of transistors in a single chip which results in very large size chips to improve the performance. The large size chips have a low yield and hence a higher cost. Advanced multi-chip modules contain several smaller size chips on faster interconnects compared to the current printed circuit boards.

Designing VLSI systems without considering the packaging options up front

may result in higher cost and lower performance systems. There is a need for an early analysis tool which will consider the various partitioning choices and cache size alternatives along with the packaging options to design the best cost/performance system among the different available design choices. To consider the packaging issues early in the design cycle, the VLSI system designer has to answer the following questions:-

1. Should the design be realized as single chip or multiple chips ?
2. What is the best number of partitions for the multi-chip design ?
3. What kind of substrate should be used ?
   MCM-C, MCM-D, MCM-L ?
4. What is the best choice of IC process to be adopted for chips ? Separate Logic and Memory process or Single process for both Logic and Memory chips ?
5. Which design gives the best Cost/Performance of the system.
6. What is the optimum Cache size ?
7. What is the optimum bond pitch for flip-chip IC and MCM-D substrate ?
8. What kind of search strategy should be used ? - exhaustive or heuristics at different level of design abstraction.

1.4 Goals and Expected Contributions

The main goal of this work is to establish an early analysis procedure procedure which will assist the VLSI system designers and packaging designers to explore the various partitioning, cache size and MCM packaging options to produce the best cost/performance VLSI system. This work aims to develop a computer aided tool to help the system designers with very little knowledge about the various packaging techniques to perform the early analysis to achieve this goal.

The *early analysis* procedure developed in this work will help the transition of large, high cost single chip designs to be realized as low cost multi-chip versions, made up of a cluster of multiple smaller dies on a single substrate. The tool developed in this work will allow the VLSI system designers to perform various tradeoff analysis involved in making this decision. The transition from a single chip design to multi-chip design is a paradigm shift from the traditional VLSI system design procedures.

CHAPTER 2

BACKGROUND

2.1 Related Work

The early package analysis tool Pepper is described in [1]. The Pepper tool needs a rough partitioning of the design as input specification. Thus the tool does not allow the analysis of interaction between various choice of partitioning and packaging. The analysis does not consider actual cost figures, only normalized values are considered. Multi-chip System Design Advisor (MSDA) is a software tool which concurrently computes size, thermal and cost performance metrics for a multi-chip module [2]. These metrics are evaluated using analytical models. However, MSDA lacks the setup to generate the various partitions of the design to be analyzed. Earlier work involving MSDA to evaluate various partitions and package options considers a few manually selected set of partitions [3]. The partitions considered were of equal size, i.e. homogeneous. The analysis did not consider non-homogeneous partitions.

The Audit tool [4] evaluates key system parameters like cycle time for different packaging material parameters. In this tool, partitions are supplied by the user and the tool is not useful to analyze the effect of various partition choices on the system performance.

An early analysis of cache is reported in [5]. This work is done at the pre-netlist level which is more suitable for futuristic processor designs. The work also does not include die model and die parameters in its analysis.

System optimization of a MCM is described in [6]. The work analyzes cost/performance improvement of a RISC processor implemented as a MCM. The analysis considers only a few partitioning cases in which chips are homogeneous, i.e. all chips which make up the MCM are of the same size.

2.2 Packaging Techniques

The various multi-chip packaging options available to the micro-electronic system designer depend upon (i) the techniques available for bonding chips to the package and (ii) the materials used to fabricate the interconnection layers and substrate. These are described in this chapter.

2.2.1 Bonding Technology

Wire-bond

Single chip packages (SCPs) for micro-electronic systems traditionally used wire-bond as the the bonding technique. The chips are attached to the substrate with the I/O pads facing up. A special thermally conductive adhesive is used to attach the chips to the substrate. Aluminum wires are used to attach the chip I/O pads and the corresponding metallized pads on the substrate as shown in Figure 2.1.

The primary advantage of wire-bonding is that it does not require any new

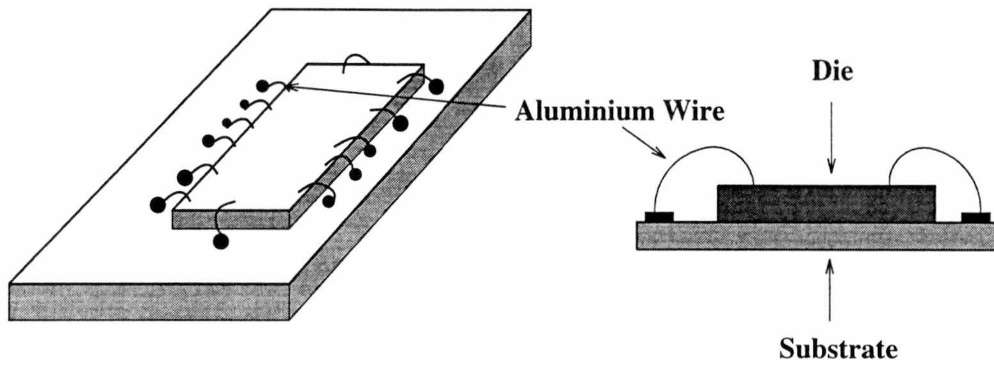

Figure 2.1: *Wire Bonding*

equipment for multi-chip modules since existing SCP wire-bonding equipment can be used. Because of this, it is very cost-effective for many low volume and low performance applications. The major disadvantage of this technique is the inefficient way of forming the chip connections since this has to be done one pin at a time. The other problem with this technique is that the I/O pads must be located on the perimeter of the chip and the area-array pads cannot be bonded with this technique. This limits the number of I/O pads that can be accommodated on a given die. The number of I/Os available are of the order of a few hundred [7]. Also, the parasitic inductance of the bonding wires degrades the electrical signals which results poor electrical performance.

Flip-chip

In the flip chip bonding technique solder bumps are formed on the I/O pads of the bare dies. The chips are placed face down on the substrate so that the bumps align with contact pads on the substrate surface. The solder is re-flowed to form all the contacts simultaneously and attach the bare dies to the metallized substrate as shown in Figure 2.2. The major advantages of flip chip are very low parasitics since there are no leads between the chip and package and very high silicon efficiency. The flip chip technology exhibits self-alignment since any small misalignments in the placement of the chip are automatically corrected due to the surface tension of the solder. In the case of flip chip bonding, all the connections between the chip and substrate are completed in parallel in one step, unlike the

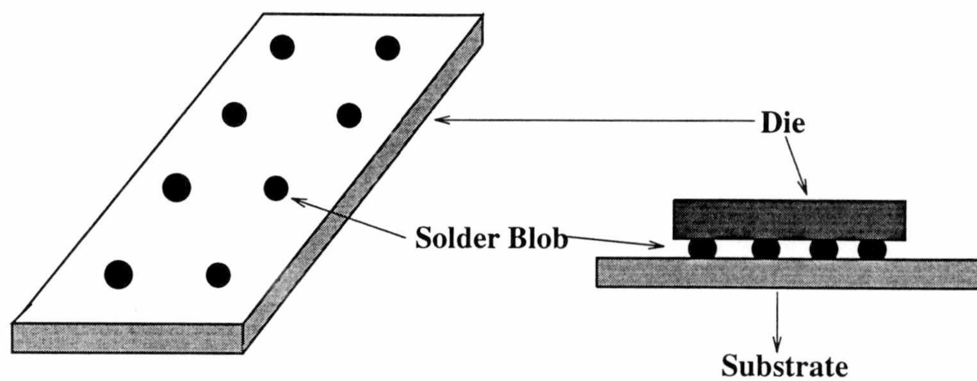

Figure 2.2: *Flip-chip Bonding*

wire-bonding where the connections are made sequentially, one connection after another. The connectivity between a matrix array of I/O pads and the substrate can be made easily with flip-chip bonding. Chips with very high pin count, of the order of a several hundreds can be handled because of these two features [7]. Faulty chips can be removed for rework without damaging the chip bond pads. The disadvantage of the flip chip technique is that the heat removal from the chips becomes difficult since the back side of the chip is not in contact with the substrate.

2.2.2 Interconnect Techniques

The interconnect techniques used in the micro-electronic systems packaged as MCMs are classified as MCM-C (multi-layer ceramic), MCM-L (laminated) or MCM-D (deposited thin-film). The features of these are mentioned in the following sections.

MCM-C

The high performance computers have used the ceramic multi-chip module (MCM-C) packaging technique for over a decade [8]. The cross section of the MCM-C substrate is shown in Figure 2.3. The top layer of the MCM-C substrate is used for bonding the chips to the substrate. The metal wiring used for routing inter-chip connections, connections to module I/O pads, and for supplying power and ground to chips is patterned on each layer. The I/O pads of the module are brazed to the bottom layer.

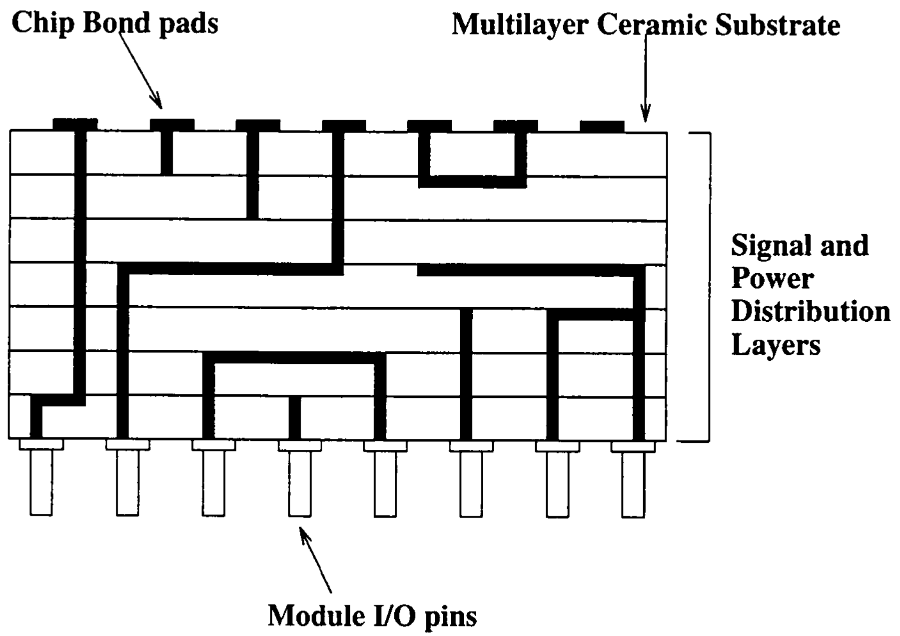

Figure 2.3: *MCM-C Interconnect Technique (after [2]).*

The ceramic modules are constructed using layers of "greensheet". The greensheet is a mixture of ceramic and glass powder suspended in an organic binder [7]. Holes are punched for the vias and the wiring patterns using molybdenum paste . A multi-layer alumina substrate is obtained after the layers are stacked together and baked. The major disadvantage of the ceramic substrate is the relatively low wiring density achievable with the screening and punching process used to form the interconnect wiring. Complex designs may require many wiring layers to complete the routing. IBM's recent ceramic module used as many as 63 layers [9]. The advantages of MCM-C are excellent dimensional stability, high reliability and module power dissipation capacity. The cost of this technique is comparable with other MCM substrate technology due to the maturity of the process.

MCM-L

Laminated MCMs basically use finer line width PCB technology. Figure 2.4 illustrates a typical MCM-L substrate. Presently, MCM-L substrates offer line width and spacings as fine as 75 microns (3 mils), which is sufficient for several low-cost micro-electronic system designs. Among the available substrate techniques MCM-L is the most cost-effective for applications which do not require very large routing resources or demand very high performance. This is because MCM-L is based on PCB technology which is a mature technique. The disadvantage of MCM-L is its poor thermal performance, due to the lower thermal conductivity of the substrate.

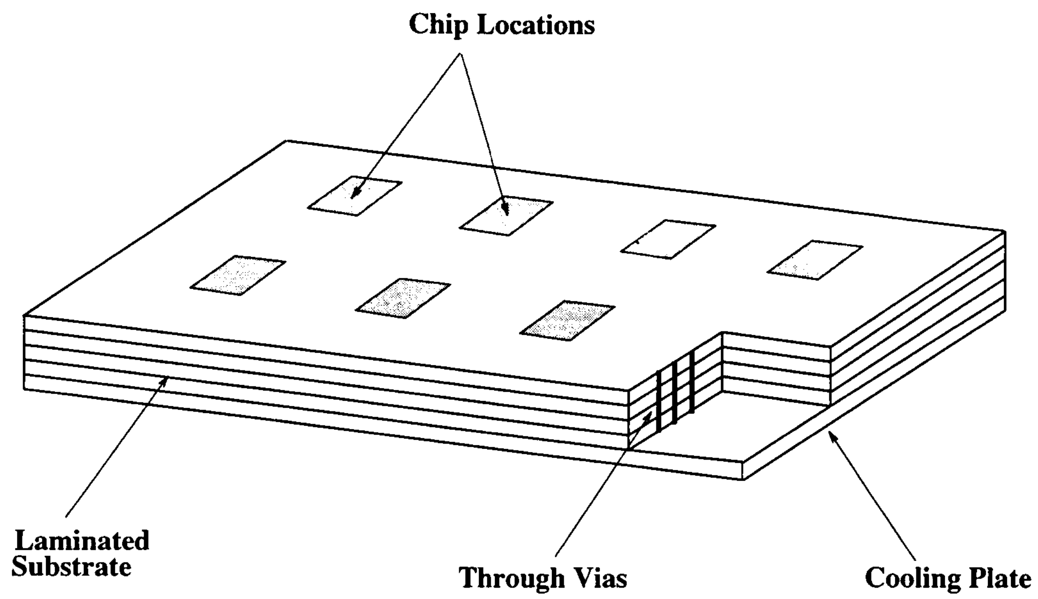

Figure 2.4: *MCM-L Interconnect Technique (after [2]).*

MCM-D

MCM-D (Deposited thin film) consists of a substrate over which alternate thin-film dielectric and metalization layers are deposited as shown in Figure 2.5. The dielectric can be polyimide or silicon. Copper or aluminum is normally used in the metalization layers. The material used for the substrate is ceramic, silicon or aluminum. The wiring capacity is in the order of 1000 inches/sq.inch. Thus the wiring density is comparable to that of on-chip interconnect. The prime disadvantage of the MCM-D technology is its high cost mainly due to the lack of vendor infrastructure. Because of the very high wiring capacity offered by the MCM-D which exceeds the requirements of most systems, the technique rarely uses more than two signal wiring layers. For applications which require very high interconnect density and operating speeds beyond 200 MHz, MCM-D may become necessary.

2.3 Partitioning

Partitioning is an important step in the design of VLSI systems. The partitioning process involves breaking a large system into smaller subsystems. One of the most important applications of partitioning is circuit packaging where partitioning is used to divide a large design into several chips to satisfy packaging constraints.

In circuit packaging, a large circuit is to be divided into several sub-circuits

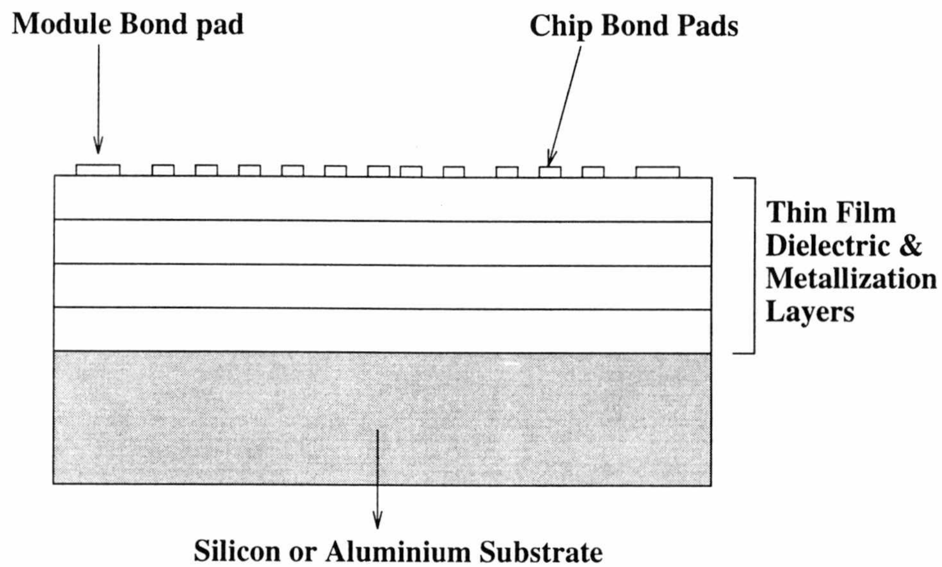

Figure 2.5: *MCM-D Interconnect Technique (after [2]).*

that can be implemented in the available MCM interconnect technology and package. The packages have a limited size and a limited number of external pins so the partitioning process must limit the size and external connections of each sub-circuit.

Several formulations of the partitioning problem, including logic partitioning, are known to be NP-complete [10]. The computation time for the exact solution of these problems increases exponentially with the number of components in the design and becomes rapidly prohibitive in terms of computation time. For this reason, most of the research in partitioning has concentrated on heuristics which aim at finding a near-optimal solution in a reasonable amount of computation time. The heuristic methods do not guarantee the global optimum solution, but a near optimal solution. Many factors have to be considered during partitioning since there are some physical constraints which must necessarily be satisfied and there are other factors that need to be optimized. Most of the partitioning techniques are based on a graph model of the design. Figure 2.6(a) shows the simplified version of the net model used in this work. In this model, each net is modeled as an edge in a hyper-graph and each functional unit component is a node in the hyper-graph [11]. The partitioning algorithm clusters the functional unit components into a specified number of separate modules as shown in Figure 2.6(b). After partitioning, each subgraph represents a cluster of components to be placed together on a separate chip as shown in Figure 2.6(c).

The main objective of partitioning is to decompose a graph into a set of sub-

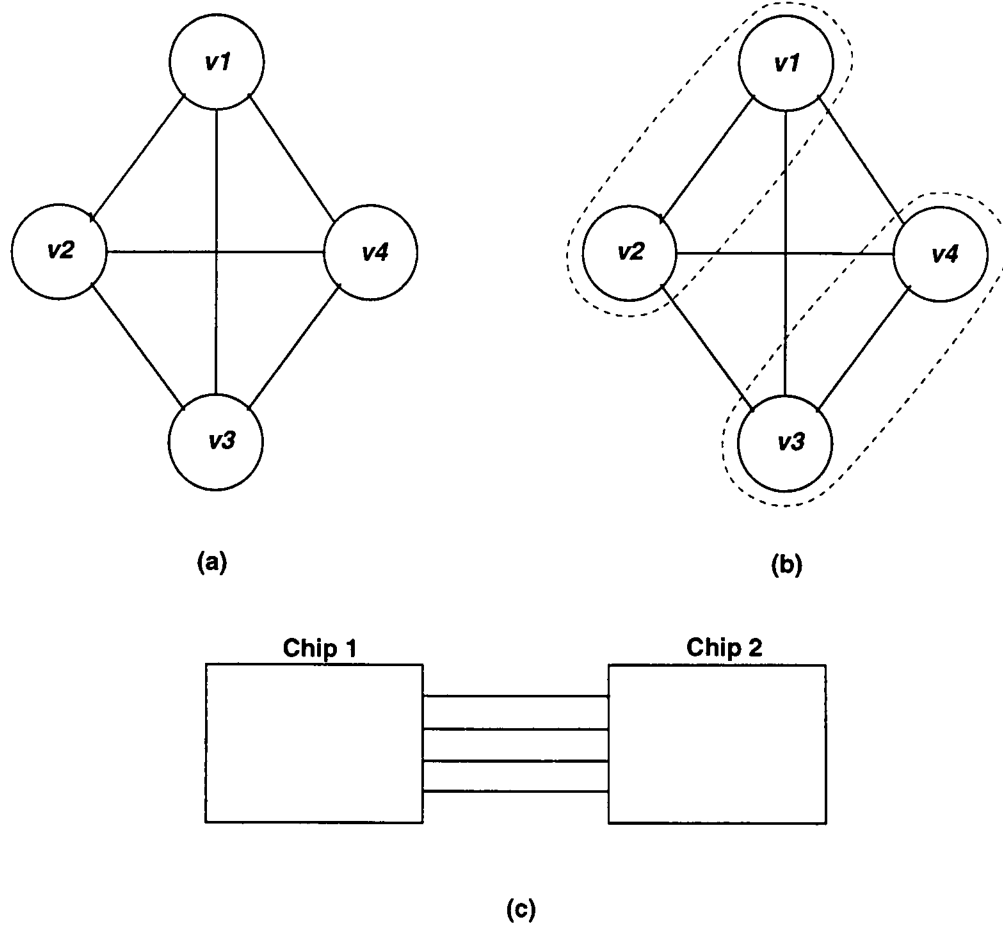

Figure 2.6: *(a) Graph Model for Partitioning (b) Clustering of Components during Partitioning (c) Physical Representation*

graphs to satisfy some given constraints while minimizing an objective function, such as the number of edges connecting two subgraphs.

The partitioning problem is formulated as a constraint satisfying problem, so the MCM partitioning can be defined as a partition of the design into a set of chips subject to constraints on area and the I/O pin count. An MCM package can be considered to contain a set of chips with each chip placed in a chip slot. Each chip has constraints on area A_{max} and the maximum number of terminals P_{max}. The system is represented as a graph given as $G = (V, E)$. Each node in the graph denotes a component in the design, and each edge represents a net connecting two or more components of the design through their I/O pins. Formally, the partitioning problem can be stated as:

Let $G = (V, E)$ be a graph, where V denotes the set of nodes (corresponding to components) and E the edges in the graph. Each node v has some cost associated with it. For example area, power, etc. can be associated as the cost for each node. If A_v is the area of the functional level unit associated with each node of V, then partition V into a set of disjoint subsets $V_1, V_2, ...V_k$ subject to the following constraints:

$$\sum_{v \epsilon V_i} P_v \leq P_{max} \qquad 1 \leq i \leq k$$

$$\sum_{v \epsilon V_i} A_v \leq A_{max} \qquad 1 \leq i \leq k$$

where P_v is the number of edges crossing from any component v in segment V_i to any other segment V_j $(j \neq i)$.

CHAPTER 3

METHODOLOGY

3.1 Early Analysis Considering System Cost

In cost-driven early analysis, we start with the single chip implementation of the MicroSparc which is a 15×15 mm square die and compare the cost of manufacturing with several multi-chip versions. The cost models described in [12] are used to evaluate the single chip and multi-chip versions. The current implementation of the MicroSparc design using a large expensive die in a single chip package is the main motivator for partitioning the design to be implemented as several smaller dies in a multi-chip design. The design candidates are basically evaluated in terms of cost. The performance is evaluated for the best cost designs and it has been found to have performance comparable with single chip designs without any significant degradation [13]. The improvement in cost/performance index is primarily due to the significant improvement in the cost with almost equal performance.

3.1.1 Early Analysis at Functional Unit Level

At the early stages of the design cycle, the designer perceives the VLSI system as interconnected functional units or blocks. An estimation-based early analysis is

performed to analyze the various design candidates when the design is represented at the functional unit level in [13]. This work assumed that the same fabrication process is used for both logic and memory blocks of the design. This work did not take advantage of the denser memory manufacturing process for the functional units which were used as memory blocks.

Early Cache Analysis

The processor system design engineers are always confronted with the issue of deciding the correct choice of cache size which gives the best price/performance system. The designers normally evaluate different cache sizes without considering the packaging issues involved in the system.

The use of MCM packaging allows VLSI system design engineers to realize the CPU as smaller size multiple chips on a single substrate. This enables us to fabricate the cache in the separate memory process. The fabrication process for manufacturing memory elements is much denser than the process for logic elements. This is due to the more regular structure of the memory elements. The size of the memory chips is much smaller when fabricated in the separate memory process. This means for the given cache size the dies are smaller. This reduces the system cost for the same performance and hence the cost/performance of the system increases. There also are situations where we can add more cache blocks without increasing the die size when the die is I/O limited. Figure 3.1 describes this situation. Considering the above facts, there is a great need and importance

to perform an early cache analysis considering the packaging options to achieve the best cost/performance system [14].

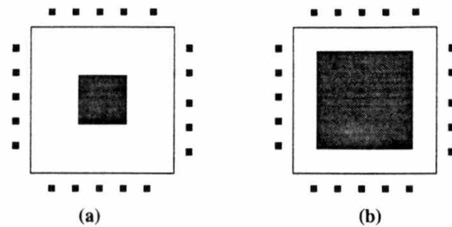

Figure 3.1: *(a) I/O limited cache die. (b) Cache size increased without increase in the die size.*

The design used in this case study was the MicroSparc design consisting of nine functional units as shown in Figure 3.2. The functional units are the instruction unit (IU), floating point unit (FU), memory management unit (MMU), instruction cache (I$), data cache (D$), S-bus controller (SBC), clock control (CLK-CTL), memory interface (MEM-INT) and reset control, clock start/stop (MISC). The monolithic baseline design consists of 750K transistors, 2K data cache and 4K instruction cache. The system operated at 40 MHz and the die measures 15mm by 15mm (0.8μ).

Evaluation of System Performance

The system performance [15] [16] is evaluated in terms of *MIPS* (Millions of Instruction Per Second) which is expressed by the equation Equation 3.1

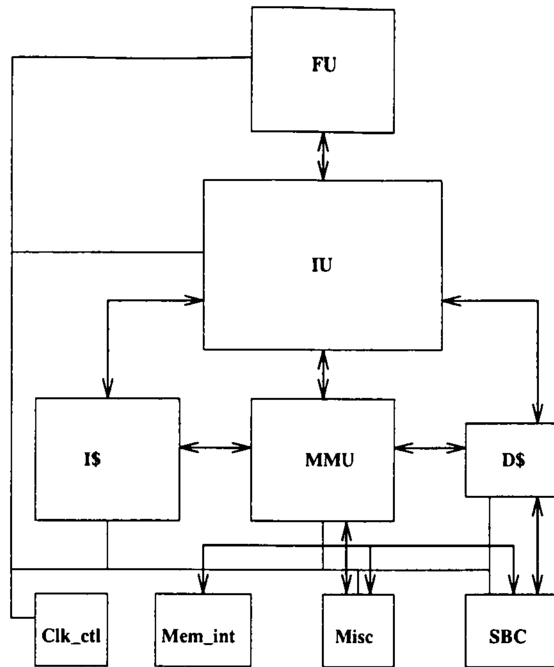

Figure 3.2: *The MicroSparc Functional unit block diagram*

$$MIPS = \frac{1000}{CPI * T_c} \tag{3.1}$$

where T_c is the cycle time in ns and CPI is the Clock Cycles Per Instruction. CPI is a figure of merit reflecting the style of instruction set and implementation. Cycle time depends upon hardware technology and organization.

Principles of Memory Hierarchy

The evaluation of CPI of a processor depends upon the memory hierarchy of the system. The general memory hierarchy in a system is shown in Figure 3.3.

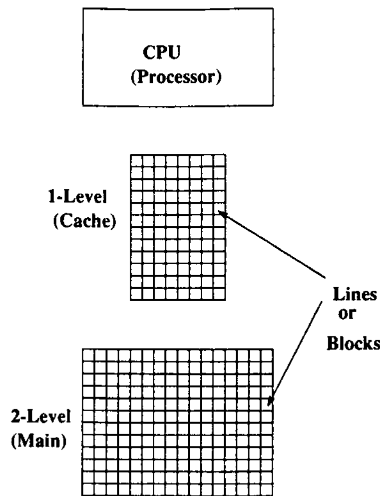

Figure 3.3: *Memory Hierarchy*

There could be several levels in the memory hierarchy but at any point of time the memory is managed between two adjacent levels. *Block* is the basic unit of information that can be present or not present in the memory hierarchy. The term *Line* is used in caches instead of *Block* . *Hit* is an access found in the first level whereas *Miss* means it is not found in that level. *Hit* and *Miss* represent the success and failure of access respectively. The ratio of memory access not found to the memory access tried is defined as *Miss Rate*. The speed of hits and miss are the key issues which determine the performance of the system.

Hit Time is defined as the time required to access the first level of memory hierarchy. The time needed to replace a line or block in the first level by the corresponding block in the second level in addition to the time to deliver the

block to the CPU is termed as *Miss Penalty.* Average time to access memory given by Eqn. 3.2 is a better measure of performance than just the *Miss rate* [16].

$$Average\ memory\ access\ time = Hit\ time + Miss\ rate * Miss\ penalty \quad (3.2)$$

The average memory access time is normally expressed in number of clock cycles. Computer designers favor the lowest average access time which reflects the product of miss rate and miss penalty rather than just the miss rate.

Performance of Cache

The CPU shares its time between executing the program and waiting for the memory system. To simplify evaluation of cache alternatives, it is assumed that all memory stalls are due to cache [16]. This assumption is true for most of the machines. The *CPI* considering the memory hierarchy is given by Eqns 3.3 and 3.4

$$CPI = CPI_{execution} + Memory\ stall\ clock\ cycles \quad (3.3)$$

$$CPI = CPI_{execution} + \frac{Memory\ access}{Instruction} * Miss\ rate * Miss\ penalty \quad (3.4)$$

Cycle Time Evaluation

The cycle time T_c is determined by the critical path of the system. Critical path is defined as the longest delay path from any input to any output in the system.

Figure 3.4 shows the critical path in the logic network. Timing analyzers are used to determine the critical path in the system [17] [18]. The critical path in the Figure 3.4 is $I_1 \rightarrow A \rightarrow B \rightarrow C \rightarrow D \rightarrow O_1$. Partitioning the dies into multiple chips will have a great impact on the cycle time T_c if the critical path runs between two different chips. Figure 3.5(a) shows the situation where the critical path interconnect lies within a single or monolithic die. Figure 3.5(b) shows the case when the critical path interconnect runs between two separate chips bonded to a MCM substrate.

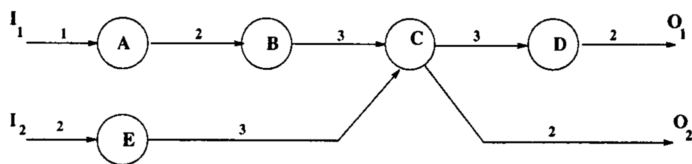

Figure 3.4: *Critical path in a logic network*

Interconnect Delays

The propagation delay on the interconnect wire depends upon the distributed capacitance, resistance and strength of the driver. For long interconnect lines

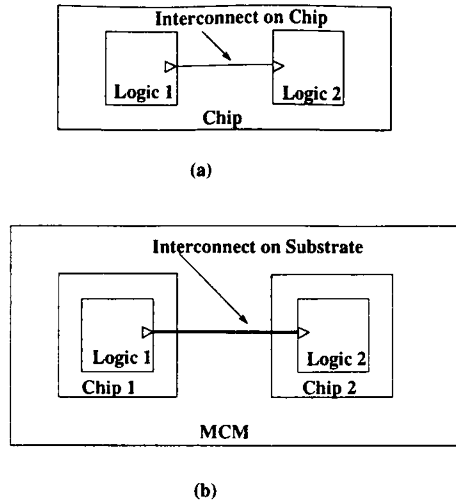

Figure 3.5: *Interconnect in Monolithic chip and chips on MCM*

the RC delays can dominate the other factors. The transmission line effect is predominant especially in interconnect wires which are highly resistive [17]. The interconnect can be modeled as several RC sections as shown in Figure 3.6. The discrete analysis of the circuit in Figure 3.6 gives an expression for the delay in the interconnect [17].

$$T_c = \frac{RC * n(n+1)}{2} \tag{3.5}$$

where n is the number of sections. The Eqn. 3.5 can be translated to an equation involving the length of interconnect wire as n becomes large.

$$T_c = \frac{rc * l^2}{2} \tag{3.6}$$

where r and c are the resistance and capacitance per unit length respectively

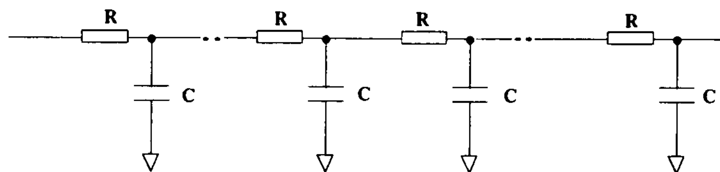

Figure 3.6: *Interconnect model for delay evaluation considering the distributed RC effects*

and l is the length of the interconnect wire. The l^2 term in the Eqn. 3.6 clearly shows that RC delays dominate the signal delays in longer interconnect wires. Normally buffers are placed in between the sections of the interconnects to reduce the delays. In some situations like clock lines the interconnect width has to widened to reduce the delays which is due to the predominant decrease in r over a slight increase in c.

Types of Interconnects

In the case of monolithic chips, all the interconnections are within the chip. The interconnect properties like trace width, resistance and capacitance are decided by the IC interconnect technology. IC interconnect technology uses shorter trace width to accommodate the high chip interconnect density. The shorter trace width results in a higher value of per unit resistance. Typically aluminum with

a resistivity of 2.37×10^{-08} ohm-m is used as the conductor material and silicon dioxide with a dielectric constant of 3.9 is used as the dielectric material.

In multi-chip modules where the multiple chips are bonded to a common substrate, the interconnection between the chips run on the substrate. There are three main substrate options available for MCM interconnects as described in Chapter 2. The MCM-D technology offers higher interconnect densities comparable to the IC interconnects but has lower per unit resistance due to larger trace widths, typically 15μ. The MCM-D substrate also allows the use of copper conductor and polyimide dielectric which offer better electrical properties than the materials used in the IC interconnects. Hence the delays of the interconnections on the MCM substrate are comparable to the delays in the IC interconnects for shorter length nets and even smaller for longer length nets [6].

Transmission Line Model for Interconnects

Multi-chip modules have chip-to-chip delays. To model accurately these delays, the bonding wires, package pins and interconnections should be modeled as transmission lines [19]. The determination of whether the interconnect carrying signals should be treated as transmission line or lumped capacitor for analysis purposes is related to the frequency spectrum of the signal. The bandwidth of the signal β is given by

$$\beta = 0.365/t_r \tag{3.7}$$

where t_r is the rise time of the signal.

The pulse design wave length λ for the signal is defined as [20, Chapter 11]

$$\lambda = \frac{\frac{c}{\sqrt{\epsilon_r * \mu_r}}}{N * \beta} \tag{3.8}$$

where c is the speed of light in vacuum, ϵ_r relative dielectric constant, μ_r relative permeability and N is an integer that signifies the quality of the signal. The interconnect is modeled and analyzed as transmission line if the length of interconnect exceeds $\lambda/8$, otherwise it is treated as lumped circuit. For the MCM-D substrate with polyimide dielectric and rise time of 1ns and N=4, the value of $\lambda/8$ limit is 14 mm. Figure 3.7 shows the simple equivalent circuit for a typical chip to chip interconnect that has to be treated as a transmission line. The chip connection leads are shorter than the $\lambda/8$ and hence treated as lumped circuit elements. The typical values of lead inductance and capacitance for flip chip bonding [20] are shown in Figure 3.7.

The buffer is placed at the output of chip1 to drive the signal through the interconnect. The buffer is also called as driver. The gate input in chip2 at the end of the interconnect is called as receiver or load. In the transmission line model the buffer is modeled by its thevenin's equivalent circuit. The receiver is replaced by its equivalent capacitance. The rise time of the signal is determined by the parasitic circuit elements in the model. The signal that leaves the output buffer travels down the transmission line at the speed of light in that medium [21]. The rise time of the signal at the receiver is further increased by the parasitics

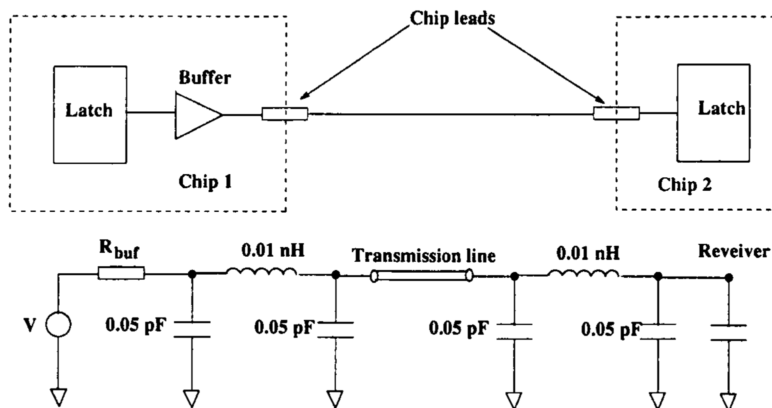

Figure 3.7: *The chip to chip interconnect and transmission line model for the interconnect*

at the receiver chip lead. The I/O buffers described in [22] are used in the timing simulations performed [23]. The I/O pads are designed to be placed in an array over the entire chip with a center to center pitch of 200 μm. The driver pad cells measure approximately 50×50 μm with a metal pad which is about 30×30 μm. Since the solder bump process protects the pad during test and die attachment the protection circuitry for the chip and the drivers can be placed directly under the pad. The model is terminated at the source end with a terminating resistor to reduce reflections [19]. The propagation delay of the signal for different interconnect length is determined by performing SPICE simulations on the model [23].

Early Bond Pitch Analysis for Area-Array IC

Traditionally, package selection and evaluation is done by the package designer, which is constrained by the packaging technology. However, some of the packaging parameters will have an impact on the other phases of the system design (i.e. IC design and partitioning) which might adversely affect the overall system design. Therefore, it is important to identify some of these parameters and evaluate their impact at the overall system design level for a given application during the early design stages to achieve an optimum system design.

Bond pitch of an area-array IC is one of the important packaging parameters which can greatly impact the other phases of the system design [24]. There is a need to evaluate the impact of the bond pitch on the system design with the help of a case study which is a representative of a large design.

For a given die area, smaller the bond pitch the higher the number of I/O that can be physically accommodated on the area-array IC as shown in Figure 3.8.

The substrate designer who designs the substrate for the MCM package has to be concerned about "escape routing" of the pads on the IC as shown in Figure 3.9. The escape routing is referred to as the routing of the area-array I/O pads on the IC from the area under or surrounding the die to vias for distribution to other layers or the top layer routing on the substrate.

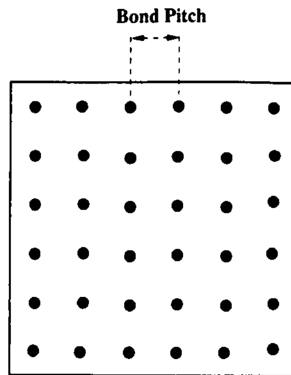

Figure 3.8: *The Bond Pitch in Area-Array IC*

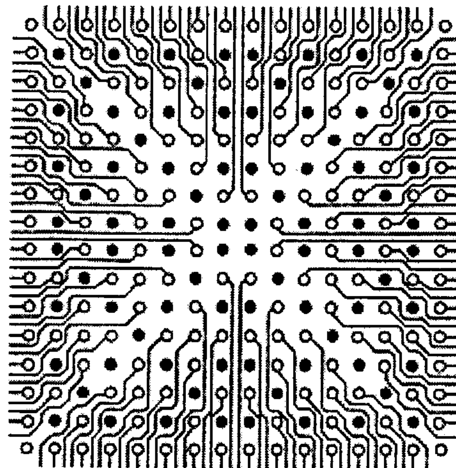

Figure 3.9: *Escape Routing for 16×16 Area-Array Bond Pad IC after [24]*

The upper bound for the number of I/O's which can be escape routed is a function of

1. Bond Pad Pitch (driven by bonding technology)

2. Wiring Pitch (driven by the substrate technology)

The upper bound for the number of I/O's which can be physically placed on the die is a function of

1. Die size (driven by IC technology)

2. Bond Pad Pitch (driven by bonding technology)

For a set of given IC, bonding and substrate technologies the maximum number of I/O pads which can be place and escaped routed on the die is the lowest of the above two upper bounds.

The upper bound for escape routing for a single layer of substrate is given by the Equations 3.9 and 3.10.

$$N = \frac{b^2 * \left(1 + \sqrt{1 + (\frac{4}{b})}\right)^2}{4} \tag{3.9}$$

$$b = -\frac{4 * (P_{top} + 1) * (2 + S : G) - 8}{S : G} \tag{3.10}$$

where N is the total number of I/O's that can be escape routed per layer, P_{top} is the number of tracks allowed between the I/O pads and $S : G$ is the signal to ground ratio.

The above discussion clearly indicates that the methodology adopted for designing the best MCM system design should consider the escape routing requirements which are dependent on the bond pad pitch at the early stages of the system design.

3.1.2 Early Analysis at RT Level

As the design of the system progresses the designers move from defining the system in terms of functional blocks to register transfer level units. This is a part of the top down design process in which designers start at the top of the abstraction hierarchy and move down to the least abstract description adding more details at each step. Figure 3.10 shows the different levels of design abstraction in VLSI system design. At the Register Transfer (RT) level of design description the system's time behavior is fully specified that is the input and output values at every clock are known. The area and delay estimates are more accurate at the RT level than at the functional unit level. The MicroSparc design consists of twenty-three synthesize-able register transfer level units. It is easy to synthesize the individual register transfer level units than the whole MicroSparc design due to the large memory requirement for the synthesis process. The exhaustive search used in the early analysis of designs at the functional unit level is not feasible for designs at the register transfer level as the number of units is large at the register transfer level. Table 3.1 shows the typical time taken to generate all possible partitions along with the number of nodes in the design. The time indicated is

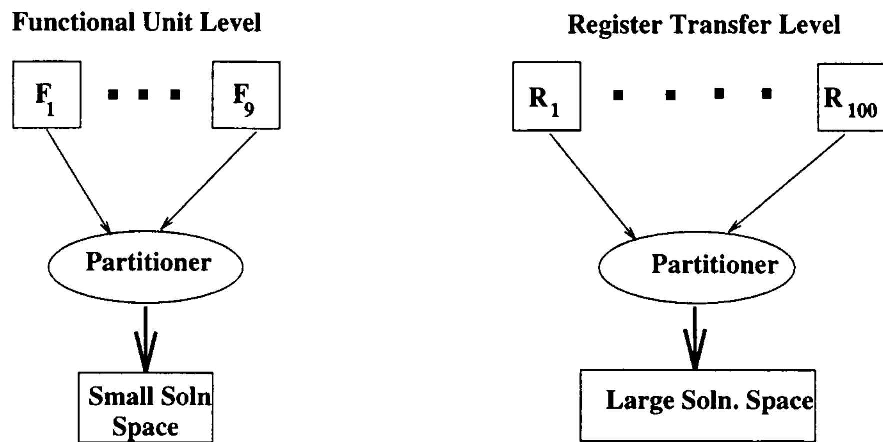

Figure 3.10: *Levels of Design Representation and Partitioning*

Table 3.1: *The time taken to generate all possible partitions*

Number of Units	Number of Partitions	Time to generate Partitions
9	21,147	fraction of seconds
10	115,975	,,
11	678,570	2 sec
12	4,213,597	9 sec
13	27,644,437	2 min.11 sec
14	190,899,322	8 min.52 sec
15	1,382,958,545	2 hrs. 10 min.
16	1,890,207,555	6 hrs. 49 min

just the time to generate the partitions and does not include the time to evaluate the partitions. The simulations indicate the exhaustive search is not feasible for designs with more than fifteen units.

Simulated Annealing

The simulated annealing algorithm was used to search the solution space of the designs at the RT level since it is not feasible to do an exhaustive search. The simulated annealing algorithm derives its name from its similarity to the process of annealing of solids. The configurations in the combinatorial systems corresponds to the states in the physical system. The minimum cost configuration corresponds to the ground state of the physical system. In the simulated annealing process first the temperature is increased to the melting point and then it is slowly cooled until the system freezes beyond which there is no further improvement in cost.

The simulated annealing algorithm used to solve the problem of partition

generation is shown in Figure 3.11 [25].

At each temperature step the partition is perturbed to obtain a new configuration or new partition. The algorithm starts with a initial random partition S. The cost of the partition $Cost(S)$ is evaluated and if this partition is not satisfactory then a new neighboring partition S_{new} is generated by perturbing the original partition. The difference in cost between the two partitions is denoted by ΔE. The new partition S_{new} is accepted if $\Delta E \leq 0$ and is used as the starting partition for the next temperature step. In the case of $\Delta E > 0$ the new partition can still be accepted with a certain probability. The probability of accepting the new partition which has worse cost than the old partition is given $P(\Delta E) = e^{-\Delta E/(k*T)}$ where k is the Boltzman's constant. The probability value is compared with a random number which is uniformly distributed between 0 and 1. If the probability calculated is greater than the random number the partition S_{new} is selected otherwise it is dropped. Several constraint satisfying partitions are generated by repeating these steps. The simulated annealing algorithm performs a random walk through the search space and helps to identify the neighborhood partitions satisfying the constraints. Thus it is feasible to identify the constraint satisfying partitions of the register transfer level design of the MicroSparc containing twenty three units which is not feasible with the exhaustive search.

```
     anneal()
1    S:Initial partition
2    T:Initial temperature
3    begin
4        while (T > Stopping Temperature) do
5        begin
6            while (constraint satisfying partition not obtained) do
7            begin
8            randomly perturb S to generate neighboring partition S_new
9            ΔE = Cost(S) - Cost(S_new)
10           P(ΔE) = min(1, e^{-ΔE/(k*T)})
11           if (rand(0,1) ≤ P(ΔE))
12                 S = S_new
13           end
14           output constraint satisfying partition
15       end
16       update T
17   end
```

Figure 3.11: *Simulated Annealing algorithm for Partitioning*

3.2 Early Analysis Considering System Performance

In the early analysis of the system considering packaging constraints described so far, cost was the criterion used to select the different partitions. The timing performance was calculated at the end of the design process for the partitions having the lowest cost values. There could be a improvement in performance of the design in the MCM implementation due to the reduction in interconnect delay of longer interconnect on the MCM substrate [6]. There is a great need to consider the timing performance early in the design process along with the packaging considerations. The delay information about the interconnect is available once the layout of the design is completed and static timing analysis is performed to obtain the delay values. The flow diagram of the early analysis considering system performance is shown in the Figure 3.12.

The first step towards determining the system delay performance of the split MicroSparc design is to find in the critical path in the monolithic design. The critical path is the longest path of the pipeline stage. MicroSparc has five pipeline stages namely instruction fetch (F), decode (D), execute (E), write (W) and result (R). The methodology chosen was to evaluate the logic and net delays in all the five pipeline stages and the longest pipeline stage is the critical path of the design.

The TACTIC (Timing Analysis and Critical Tuning for Integrated Circuits) tool in Epoch CAD tool is used to determine the logic and net delays in the design. TACTIC is a static timing analyzer for use on Epoch designs. It operates on gate level simulation models and requires no test vectors. TACTIC graphically

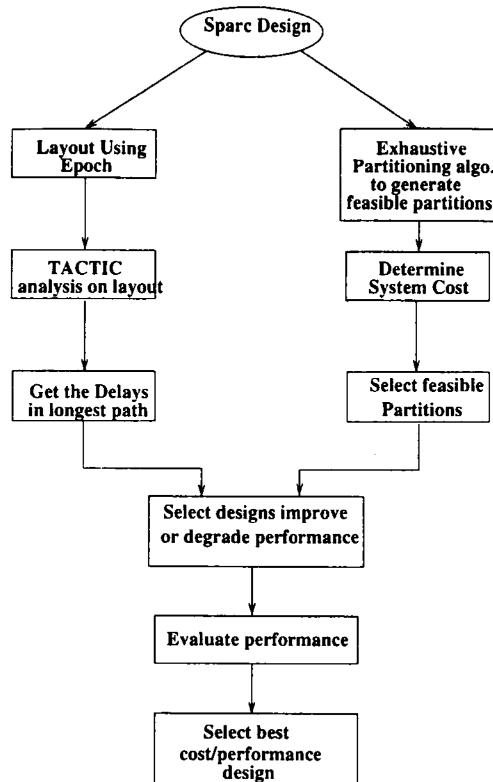

Figure 3.12: *The flow diagram of the Early Analysis considering System Performance*

displays critical paths, paths to, from and between specified nodes and clock paths.

TACTIC normally relies on SPICE-based lookup table delays values used in gate-level simulation. The lookup table values are highly accurate with respect to input slew rates, transistor impedance, and capacitive load but are unable to model the resistive effects of long routes. For such routes, TACTIC can use

lookup table values to determine the intrinsic delay of the functional modules and employ RC tree analysis using Penfield Rubenstein algorithm to calculate the delays to the various nodes on the net.

According the MicroSparc (Tsunami) specification the instruction fetch pipe line stage is as indicated in Figure 3.13.

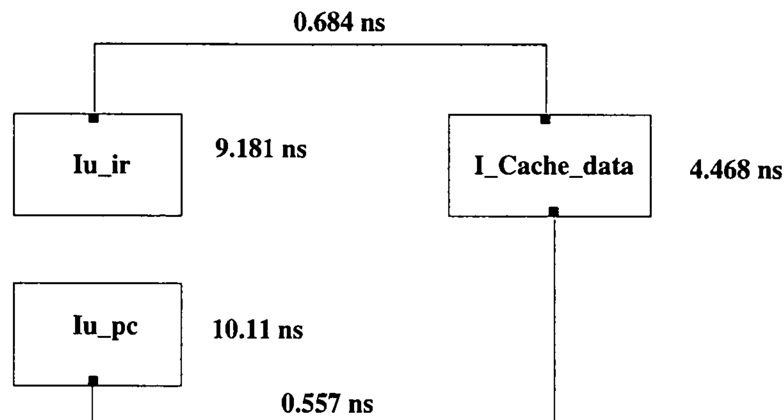

Figure 3.13: *The logic and net delays in the instruction fetch pipe line stage*

The TACTIC simulation was performed on the top level sparc design to determine the critical path in the MicroSparc design. The critical path to the nodes in the the fetch cycle path were determined and the logic delays of the individual functional units like icache, iu-pc were obtained. The RC analysis was performed on the nets in the critical and the net delays were determined. The TACTIC

simulation of the RC analysis of the critical nets is as shown in Figure 3.14.

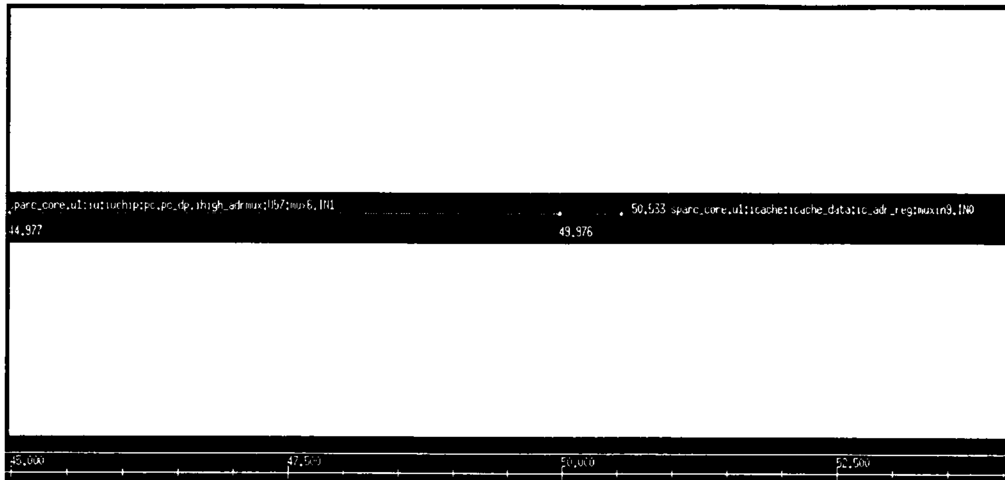

Figure 3.14: *The net delay in the fetch cycle using RC analysis*

The critical path analysis tells us that if icache, iu-pc and iu-ir are in the same chip then the performance of the chip remains the same as the monolithic design. There could be some performance improvement if the MicroSparc CPU is partitioned into at least 2 chips and the critical path is routed on the substrate which might take advantage of the faster MCM-D interconnects. The performance improvement or degradation depends upon the IC and MCM-D characteristics and interconnect length.

CHAPTER 4

IMPLEMENTATION

4.1 Early Cache Analysis

The flow diagram of the early analysis of various cache size alternatives is shown in Figure 4.1.

The design is captured as a schematic in Viewlogic's Viewdraw tool. The various cache sizes are given as attributes to the design schematic. At the early analysis stage the design information is available only at the functional unit level. In the cost driven early analysis we are looking for the lowest cost designs which satisfy the area and I/O constraints. The analysis is to be performed for single chip and multi-chip solutions of the design. In the multi-chip version the design functionality has to be partitioned into several chips to be packaged in MCM. At the functional unit level the partitioning solution space is small and there is a need to find the best candidate design since 50% or more of the design cost is committed at this stage [2]. Hence the partitioning solution space is searched exhaustively to find the best candidate. The partitioner is similar to the one described in [13] except for the following modifications. In the early analysis to evaluate the cache alternatives the main goal is to separate the memory and logic blocks in the design and group the memory and logic blocks to be realized

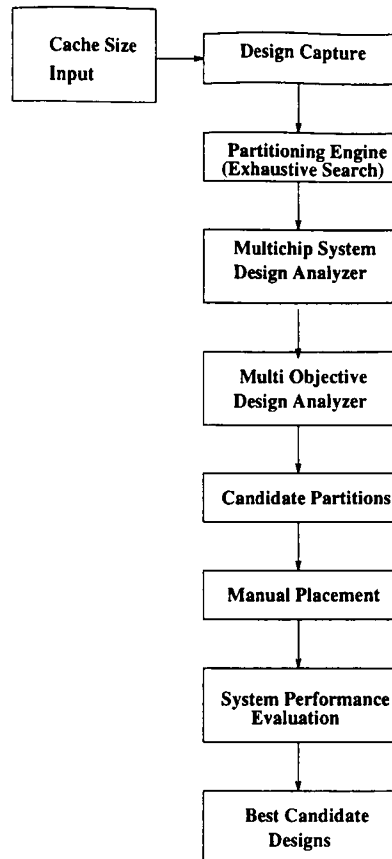

Figure 4.1: *Flow Diagram of the Early Cache Analysis*

as separate chips. This will result in a better cost system as the memory dies fabricated in a separate process are smaller in size [6]. The functional blocks are given a attribute in the design input to indicate their type, i.e. logic or memory. The partitioning engine allows only the group of logic or the group of memory elements to form a chip, it does not allow the grouping of logic and memory elements on the same chip. The generated partitions are checked to determine whether they satisfy the packaging constraints. The partitions that satisfy the constraints are taken into the next stage of the evaluation. The size of the die

for the considered MCM packaging option and the cost of the die are estimated using the estimation procedures described in [12].

The evaluation of the system consisting of multiple chips or dies on a single substrate is performed by the Multi-chip System Design Advisor (MSDA) tool [2]. The MSDA tool evaluate the system size, cost and noise of the system containing the valid partitions. The top few partitions are selected by the Multi Objective Design Advisor (MODA) [26] which evaluate the partitions according to the weights assigned to the different system performance parameters like cost, size and noise.

The selected partitions for the different cache alternatives considered are manually placed by the MSDA tool to estimate the interconnect length of the critical path of the design. The system timing evaluation is performed for the candidate designs as described in Chapter 3. The best design candidate is chosen depending upon the requirement of the application like lowest cost or highest performance/cost ratio system.

The partitioning engine is implemented in C. The interface between MSDA and the partitioning engine is written using AWK scripts. MODA is written in C and TCL. The early analysis tool runs on Unix platform.

Early Bond Pitch Analysis for Area-Array IC

The concept of design for packageability (DFP) was used in [13] in which several packaging and partitioning issues were considered at the early stages of the design

for the best cost MicroSparc MCM system. The bond pad pitch was kept constant in the above case study.

The early package analysis was extended to consider the impact of bond pad pitch on the MCM design. The multi-chip version of the MicroSparc CPU with flip-chip IC was analyzed using the early analysis tool. The bond pad pitch of the IC was varied from 150 to 400 microns in steps of 50 microns. The range and steps are the current manufacturable range in the industry. The MicroModule system's D500 MCM-D substrate with a merged via process was used for this analysis [27]. A "what if" type cost estimator was used to determine the low cost partitions for different choice of partitions [13]. The system cost consists of the cost of IC and substrate. The system cost, size, noise and power were estimated for the bond pitch varying from 150 to 400 micron. The results are reported in Chapter 5.

4.2 Early Analysis at RT Level

The flow diagram of the Simulated Annealing (SA) algorithm used to generate partitions for the early analysis is shown in Figure 4.2. The simulated annealing algorithm always takes a down hill step but some times it is allowed to take an uphill climb and this scheme has come to be known as Metropolis algorithm [28, pp. 444-455].

To make use of the Metropolis algorithm for the partitioning problem, the following elements are needed as described.

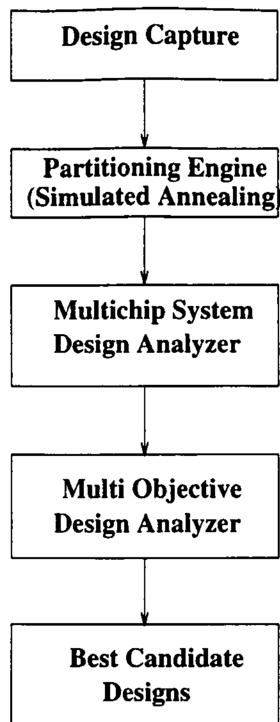

Figure 4.2: *Flow Diagram of the Early Analysis at RT level*

1. *Configuration.* The partitions are the different configuration in the SA algorithm. Each partition consists of several chips and each chip in turn contains several nodes representing the register transfer level units. A doubly linked list data structure is used to represent the partitions.

2. *Rearrangements.* The rearrangement or perturbation is achieved by first selecting a chip randomly from the partition and then selecting a random node from that chip. This node is deleted from the chip it belongs to and added to another randomly selected chip.

3. *Objective function.* The simulated annealing algorithm is implemented to

generate constraint satisfying partitions. Each chip is given a specific area and chip constraints. The algorithm selects the partitions which contain all the chips satisfying the constraints. The objective function is the sum of the deviation of the pins and area constraints for each chip.

4. *Annealing Schedule.* This basically tells how the temperature is lowered from higher to lower value. For the partition problem the temperature decrement factor is an user input. A value close to 1 for this factor means slow cooling and value close to 0 means fast cooling. For each value of the new temperature we allow for 100N reconfigurations or 10N successful configurations where N is the number of nodes in the graph. The successful configurations are the constraint satisfying partitions or the partitions accepted. The algorithm stops when the temperature reaches the or falls below the stopping temperature.

The simulated annealing algorithm is implemented in C. The code of the algorithm obtained from University of Cincinnati was modified to suit the early analysis application. The interface to the MSDA was written using AWK scripts.

4.3 Early Analysis Considering System Performance

The early analysis tool based on exhaustive partitioning algorithm evaluates the functional unit level design of the MicroSparc selects the chip partitions satisfying the area and pin constraints which are physical constraints that must

be met to fabricate the design. There could be a system delay improvement or degradation in the multi-chip version of the MicroSparc compared to the monolithic version only if the blocks i-cache, iu-pc and iu-ir are partitioned among two different chips. The multi-chip design of MicroSparc using separate memory process always partitions i-cache in one die and iu-ir, iu-pc in another die since partitioning algorithm does not allow the memory and logic functional units to be in the same die. The improvement in performance is more likely to happen in the MCM designs using separate memory process since the critical net between the cache and integer unit could be routed on the MCM substrate interconnect and could take advantage of the faster interconnects as described in Chapter 3.

There were about 200 valid partitions thus generated by the algorithm which satisfied the above constraints. The size of the i-cache unit fabricated in the combined memory and logic process could be reduced to halve its size if it is done on a separate memory process [29]. Thus the delay through the i-cache could also be reduced to halve its value in a combined memory process if it is fabricated in a separate memory process. The design candidates which offer lower design cost using the separate memory process technology were chosen and the critical path delay were evaluated using the methodology shown in Figure 4.3 and the look-up table of MCM-D interconnect delays.

The critical path delays in the instruction fetch pipe line stage was evaluated using the methodology discussed in Section 3.2.

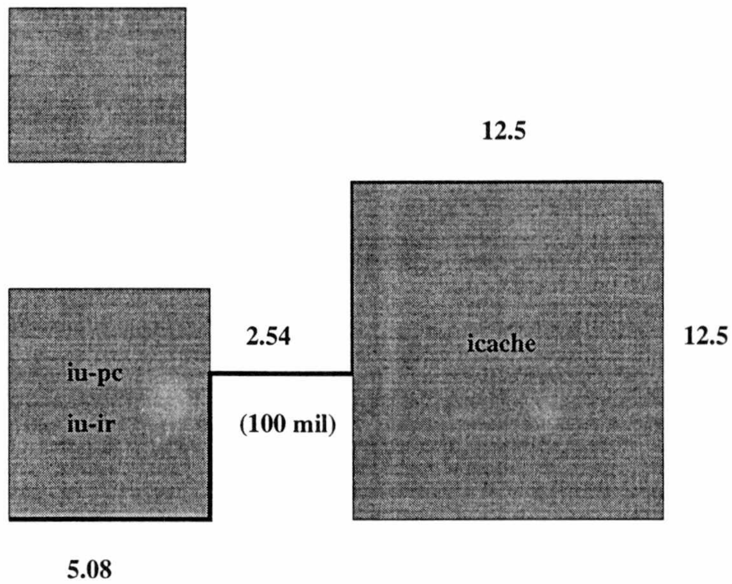

Figure 4.3: *The approach to evaluate the system delay performance*

CHAPTER 5

RESULTS

5.1 Early Cache Analysis

The early cache analysis was performed to evaluate the data/instruction cache size alternatives of the MicroSparc design described in Chapter 3. Both the combined logic and memory and separate logic/memory process were analyzed. The analysis was carried out with the assumption that the cost of the wafer and access time were the same in both the process, the separate memory process has double the density for cache implementation. Eight different candidate partitions on MCM-D substrate were evaluated for different cache sizes and the two process.

As described in Chapter 3 the overall system performance index term MIPS depends upon the CPI and cycle time. The ideal CPI ($CPI_{execution}$) measurement of the MicroSparc were provided by Sun Microsystems as 1.5. The real CPI values considering the memory hierarchy the first level of data and instruction cache were calculated using Eqns 3.4. These values are based on the Tsunami specification and cache miss rate figures provided by Sun Microsystems. Table 5.1 gives the real CPI values considering memory stall due to the memory hierarchy.

The average miss rate was computed with the assumption that 50% of memory references are to instruction and the other 50% to data. The miss penalty is the

Table 5.1: *The real CPI values considering the memory stalls due to cache-memory hierarchy.*

Data Cache	Instruction Cache	Average Miss rate	Average Memory access time (ns)	Real CPI
2K	4K	6.5%	33.3	2.07
4K	8K	4.8%	29.6	1.92
8K	16K	3.6%	27.2	1.82
16K	32K	2.6%	25.2	1.73

average of the miss penalties of the data and instruction caches. The hit time is assumed to be one cycle which amounts to 25ns at 40 MHz clock. The memory access per instruction was found to be 1.1 considering the maximum possible CPI improvement due to increase in cache size.

The processor cycle time was calculated based on the delay in the critical path as described in Chapter 3. The critical path is between TLB (Translation Lookaside Buffer) and Instruction Cache for the MicroSparc design. For the analysis carried in this work a pessimistic approach was adopted in which we assumed the critical path to be between TLB, Instruction Cache and IU (Integer Unit). This path is taken in the instruction pipeline during the instruction fetch stage.

The cycle time of the monolithic chip is assumed to be entirely due to the delay in the logic which is a pessimistic assumption done to simplify the analysis. For the MCM versions if the critical path is broken and runs as a inter chip connection on the substrate then the interconnect delay is added to the logic

delay to calculate the total system delays. The interconnect delays depend on the estimated die size and their relative placements on the MCM. The placement of the candidate partitions and the critical path is shown in Figure 5.1. Each die or chip on the substrate is assumed to have a JTAG (Joint Test Action Group) cells to manage the testing complexity. The JTAG cells have at least one 2 to 1 mux whose delay adds to the system delay. As the cache size increases it is assumed that the inherent memory access time varies, which is independent of the miss rate. MicroSparc contains 2K data cache and 4K instruction cache and it is assumed that for each additional 6K cache (total of instruction and data cache) the inherent access time is increased by 0.25 ns. The effective cycle time T_c is given by

$$T_c = T_{logic} + T_{extra-cache} + T_{jtag} + T_{buffer} + T_{substrate} \tag{5.1}$$

where T_{logic} is 25 ns, T_{jtag} is 0.5 ns, T_{buffer} and $T_{substrate}$ are calculated using SPICE simulations described in Chapter 3. The cycle time values for the various candidate partitions is given in Table 5.2. The MIPS performance index is calculated using the Eqn 3.1 and shown in Table 5.3. The contents and the normalized the cost/performance values of the candidate partitions using the combined logic and memory (LM) and separate memory (L/M) processes is shown in Table 5.4. The cost/performance values are normalized to the values of the monolithic design.

The relative MIPS performance index of the different design candidates is

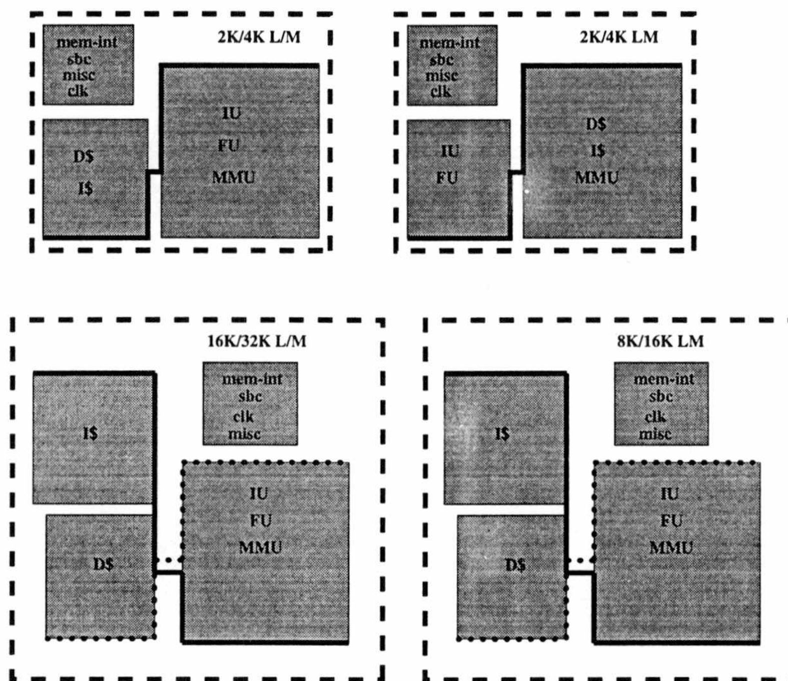

Figure 5.1: *Typical Placement of Dies on the MCM Substrate*

Table 5.2: *The cycle time values of the MCM designs*

Cache Size D$/I$	Cost ($)	Max. net length (mm)	Delay (ns)	No. of I/O	Cycle Time (ns)
2K/4K LM	57.45	21.3	0.53	1197	26.03
4K/8K LM	71.08	23.9	0.55	1197	26.30
8K/16K LM	95.34	28.2	0.59	1167	26.59
16K/32K LM	136.15	NA	NA	1171	NA
2K/4K L/M	81.48	23.5	0.54	1142	26.04
4K/8K L/M	81.48	23.5	0.54	1142	26.29
8K/16K L/M	85.02	25.1	0.56	1150	26.56
16K/32K L/M	95.34	28.2	0.59	1171	26.84

Table 5.3: *The Cost/MIPS performance index for the candidate designs*

Cache Size D$/I$	Cycle Time (ns)	CPI	MIPS	Cost ($)	Cost/MIPS
2K/4K Mono	25	2.07	19.32	400.05	20.71
2K/4K LM	26.03	2.07	18.56	57.45	3.10
4K/8K LM	26.30	1.92	19.80	71.08	3.59
8K/16K LM	26.59	1.82	20.66	95.34	4.61
2K/4K L/M	26.04	2.07	18.55	81.48	4.39
4K/8K L/M	26.29	1.92	19.79	81.48	4.12
8K/16K L/M	26.56	1.82	20.69	85.02	4.10
16K/32K L/M	26.84	1.73	21.54	95.34	4.43

Table 5.4: *The Cost/Performance Comparison of different Design Choices*

Cache Size D$/I$	Combined Logic and Memory Process (LM)		Separate Logic and Memory Process (L/M)	
	Contents	Cost/Perf. ($/MIPS)	Contents	Cost/Perf. ($/MIPS)
2K/4K	(monolithic): MEM-INT,SBC,CLK,MISC D$, I$, MMU,FU, IU	1.000	NA	
2K/4K	3 chips: D$, I$, MMU FU, IU MEM-INT,SBC,CLK,MISC	0.150	3 chips: D$, I$ FU, IU, MMU MEM-INT,SBC,CLK,MISC	0.212
4K/8K	3 chips: D$, I$, MMU FU, IU MEM-INT,SBC,CLK,MISC	0.173	3 chips: D$, I$ FU, IU, MMU MEM-INT,SBC,CLK,MISC	0.199
8K/16K	4chips: D$, FU, IU , MMU MEM-INT,SBC,CLK,MISC I$	0.223	3 chips: D$, I$ FU, IU, MMU MEM-INT,SBC,CLK,MISC	0.198
16K/32K	4chips: D$, FU, IU , MMU MEM-INT,SBC,CLK,MISC I$	N/A	4chips: D$ FU, IU, MMU MEM-INT,SBC,CLK,MISC I$	0.214

shown in Figure 5.2. There is a increase in relative performance as the cache size increases. The results show that there is no significant difference in the performance of MCM's realized in the combined or separate memory process. This is due to the fact that even though the two different process result in different partition sets the die sizes are almost the same and there is little difference in their cycle time.

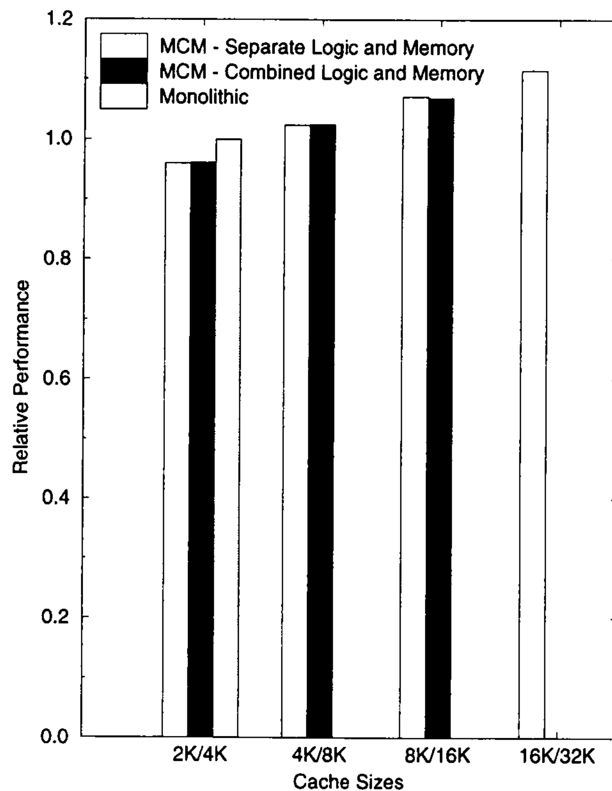

Figure 5.2: *The Relative Performance of the Candidate Designs*

The relative cost of the candidate designs are shown in Figure 5.3. All the

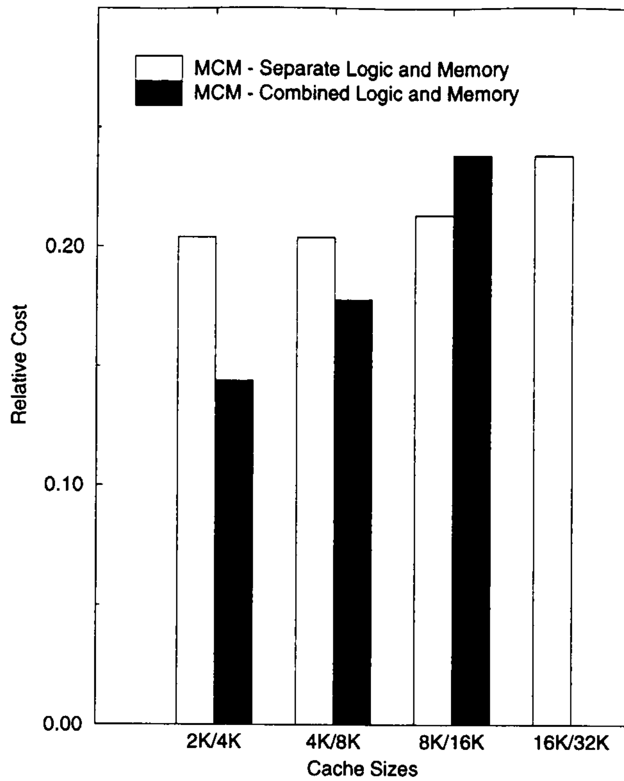

Figure 5.3: *The Relative Cost of the Candidate Designs*

MCM designs have a lower cost compared to the monolithic design. For higher cache sizes the separate memory process designs tend to be cheaper. Figure 5.4 shows the relative cost/performance index of the designs. The index for the combined logic and memory process follows a monotonic trend. This suggests that the cost/performance index gets worse as the cache size is increased. On the other hand the index for the separate memory process follows the inverted bell curve with a minimum value for the 8K/16K design. The best and the worst

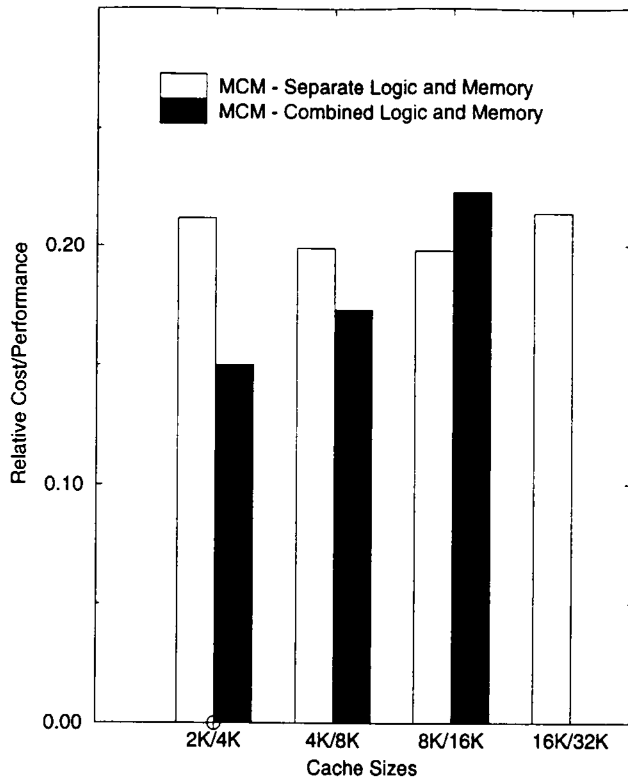

Figure 5.4: *The Relative Cost/Performance Index of the Candidate Designs*

cost/performance index among all design candidates considered is for 2K/4K and 8K/16K (combined logic and memory process) designs respectively. The designer can select one of the design candidates based on the application. For example for the lowest cost application 2K/4K combined logic and memory process design can be chosen. The 16K/32K separate memory process design is the choice for the highest performance.

5.1.1 Early Bond Pitch Analysis for Area-Array IC

The relative cost of the design candidates on an MCM system for the MicroSparc design for different values of bond pitch is shown in Figure 5.5. The number of chips in the MCM system for the bond pitch value ranging from 200 to 400 micron is shown in Figure 5.6. The system cost for design with bond pitch of 200 micron is assumed to be 1.0 p.u. The system cost decreases as the bond pad pitch is decreased from 400 to 250 micron. The reason for the decrease in cost is due to the fact that decreasing the bond pad pitch allows more number of I/O's (relaxes the I/O constraints), smaller die sizes and larger number of partition choices.

However decreasing the bond pad pitch from 250 to 200 micron increases the system cost. A careful investigation of this case revealed that as the bond pad pitch was reduced to 200 micron the number of tracks that can be routed between pads (P_{top}) in Equation 3.9 drops to 2 from 3 (P_{top} for 250 micron bond pitch). It is assumed the substrate technology (i.e. routing track pitch) is the same as the bond pad pitch is varied to determine its impact on the system design. This will reduce the number of I/O's that can be escape routed on the die which will in turn reduce the number of I/O's that can be place on the die. The net effect of this will constrain the partitioning algorithm from considering the partition with larger number of I/O's, smaller dies and lower cost. This is the case in which the difficulty in escape routing limits realization of the designs of lower system cost.

On the other hand increasing the bond pad pitch beyond 250 microns increases

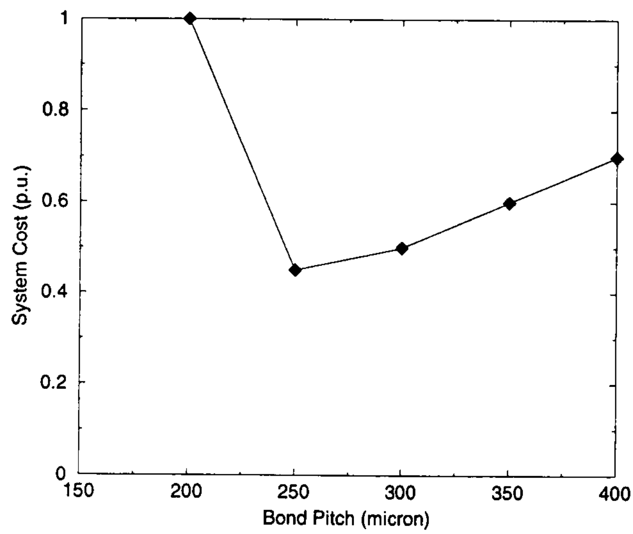

Figure 5.5: *The Relative Comparison of System Cost for for different values of Bond Pitch*

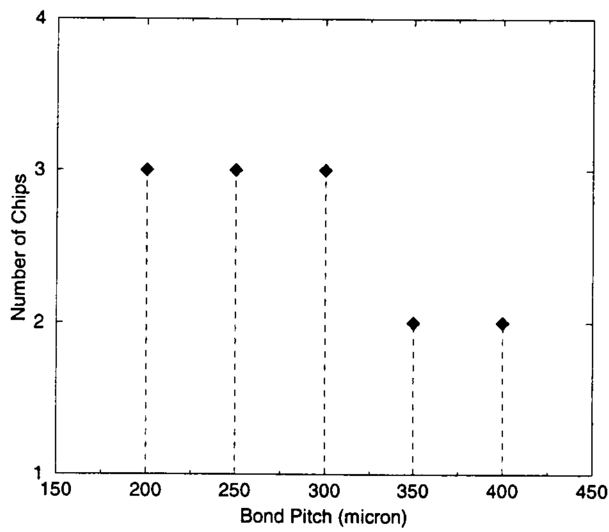

Figure 5.6: *The Relative Comparison of Number of Chips in the System for different values of Bond Pitch*

the die size of the I/O limited die and increases the system cost. This situation is expected as the bond pad pitch is increased beyond 250 micron it allows more tracks between bond pads which increases the number of I/O's that can be escape routed and placed on the die. The partitioner will allow partitions with higher number of I/O's on the die. This will more likely result in dies which are I/O limited ie the I/O's on the die determine the die size rather than the logic on the die.

The early bond pad pitch analysis results shows that the system designer has to consider two conflicting design criteria the escape routing difficulty and smaller bond pad pitch. Thus early analysis is vital in determining bond pad pitch which will result in the system with the lower cost which will be difficult to achieve if these decisions are postponed to the end of the design cycle. The results of this study suggest 250 micron bond pad pitch will realize the lowest cost system design for MicroSparc design with the Micro Module System D500 MCM-D substrate technology.

The relative size of the MCM system is shown in Figure 5.7. The early analysis results show that there is a increase is system size as the bond pitch is reduced. As the bond pad pitch is reduced to 250 from 350 micron the partitioner allows for chips with larger I/O count and larger number of chips in the partition. This increase in the overhead for each chip on the substrate and the increase in the routing area needed for the higher I/O counts on the chips increases the overall system size slightly. The increase in system size as the bond pad pitch is increased

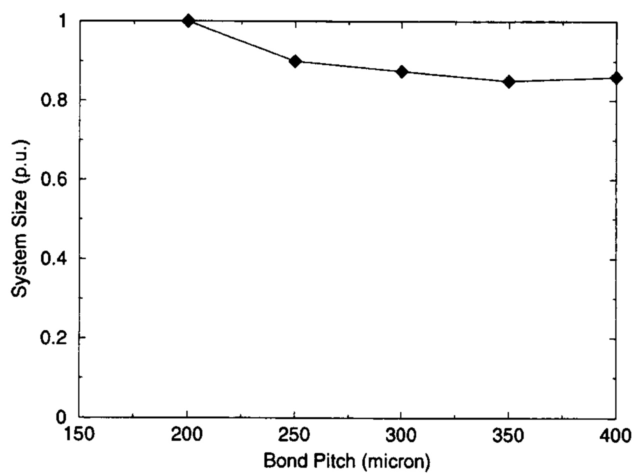

Figure 5.7: *The Relative Comparison of System Size for different values of Bond Pitch*

Table 5.5: *Results of Partitioning using Simulated Annealing at the Functional Unit level*

Chip	Pins	Area(mm^2)	Units	System cost($)
1	397	25.80	mem-intf, sbc	63.20
2	361	23.29	iu, clk-cntl	
3	203	20.84	fu, misc	
4	485	49.42	dcache, icache, mmu	

from 250 to 200 micron is mainly attributed to the larger size chips due to reduced number of I/O's limited by the escape routing difficulties.

The system power consumption and the relative simultaneous switching noise are shown in Figure 5.8 and Figure 5.9 respectively.

The power and the noise are dictated by the total number of I/O's in the system. The system with bond pitch of 250 micron has the largest total number of I/O's in the system and hence has the highest noise and power consumption.

5.2 Early Analysis at RT Level

To verify the correct implementation of the simulated annealing algorithm the results obtained for the design with nine functional units as shown in Table 5.5 was compared with the results got from the exhaustive search. The results from the SA are quite close to the results of the exhaustive search. The minor deviation of the results are due to the fact that SA guarantees a good solution not the best. The results were obtained after several runs of the algorithm with different choice of the annealing schedule. The good solution of the several runs is reported.

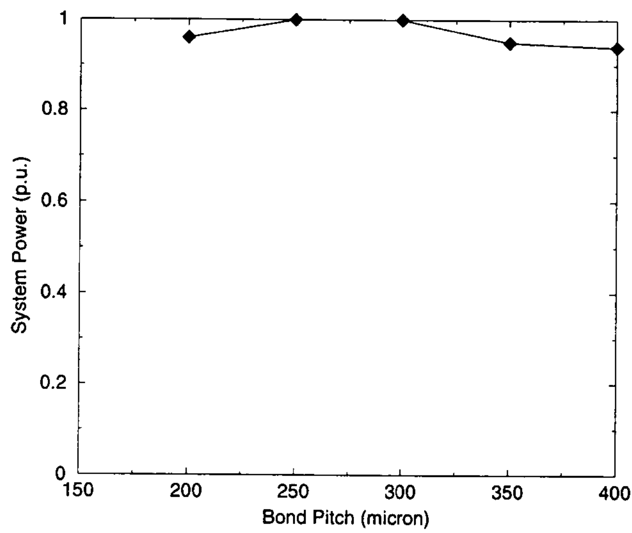

Figure 5.8: *The Relative Comparison of System Power for different values of Bond Pitch*

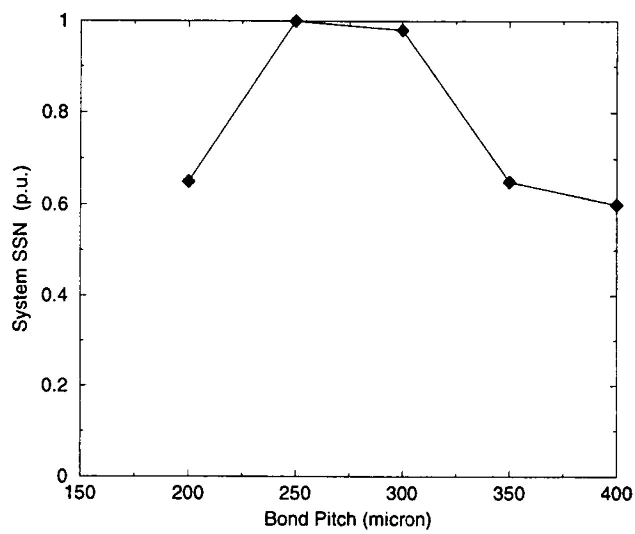

Figure 5.9: *The Relative Comparison of System Noise for different values of Bond Pitch*

Table 5.6: *Results of Partitioning using Simulated Annealing at the Register Transfer Level*

Chip	Pins	Area(mm^2)	RT-level Units	System cost($)
1	193	27.36	fp-rom, fp-fpc,jtag-cntl fp-exp, fp-frac, fp-ctl fp-rf, fp-rw	288.64
2	498	156.14	m-dcache, mir, mexec clock, mdcacheif, sbc misc, mpc, mdecode mc-tlb, dp-mmu, mregfile m-mmu-cache	
3	370	25.80	mr-icache, rl-memif	

Table 5.7: *Comparative results of the Early Analysis of MicroSparc for different design choices and processes.*

System	Process	Cost ($)	Delay (ns)	Cost/Perf.(p.u)
Monolithic	Combined logic and memory	400	25	1
MCM	Combined logic and memory	57.45	25.319	0.142
MCM	Separate logic and memory	81.48	23.165	0.185

The results of the early analysis of the MicroSparc design at the RT level is shown in Table 5.6.

5.3 Early Analysis Considering System Performance

The cost, delay, frequency and cost/performance of the MicroSparc design using the early analysis tool for the various design candidates is shown in Table 5.7. The monolithic design is considered as the base design against which the other two design choices are compared.

Figure 5.10 shows the system cost *vs* delay relationship for the three different

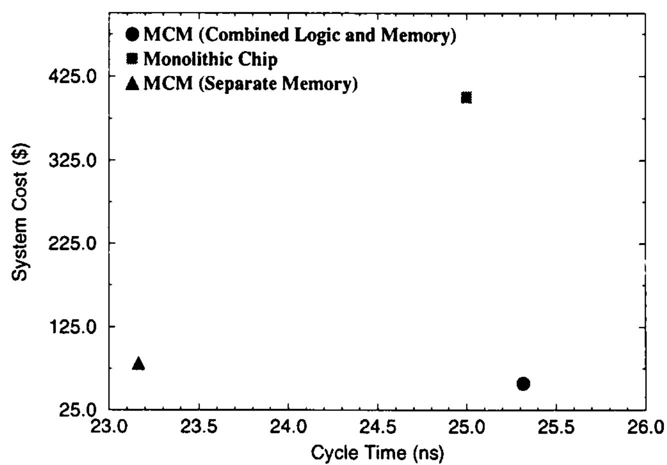

Figure 5.10: *Cost vs Performance Results of the Comparative Early Analysis for the MicroSparc Design.*

MCM/process choices considered for the MicroSparc design. They key observation found from this analysis was that the system cost varied over a wide range whereas the deviation in the delay was quite small. The early analysis considering system performance showed that about 8% performance improvement could be achieved using a separate memory and logic process for the MCM system. The combined memory/logic process is the best cost effective solution for the MicroSparc design but at the expense of a performance hit of 1.25 percent. The analysis also shows that MCM design choices are more cost effective than monolithic design choice for the MicroSparc design.

CHAPTER 6

SUMMARY, CONCLUSIONS AND FUTURE WORK

6.1 Summary

The procedure for the early analysis of VLSI systems under packaging considerations has been developed and implemented in this dissertation work. The early analysis tool developed around a user friendly interface was used to evaluate the inter-relationship between partitioning and packaging and determine the best system design considering cost, size and delays [13].

The early analysis tool was enhanced to analyze different cache sizes at the early stages of the design cycle for MicroSparc processor. The tool was also used to analyze the impact of separate memory and logic process and combined logic and memory process along with the different cache sizes, MCM packaging technology and partitioning [14].

The versatile nature of the tool is demonstrated by using the tool to perform an early bond pitch analysis to determine the best bond pitch for the MCM-D, flip-chip MicroSparc design [24].

The system delay performance estimation was fine tuned by using the Epoch Tactic timing analyzer. The delay estimates determined were used to evaluate more accurately the system delay performance of MicroSparc MCM designs on

separate / combined memory and logic process MCM packaging technology and partitioning.

The exhaustive partitioning algorithm for functional units of MicroSparc is used in the early analysis tool used in this work. There was also an attempt to use simulated annealing at the RT level of the design to perform the early analysis of the MicroSparc design.

6.2 Conclusion

The early analysis performed on the MicroSparc design using the tool developed in this work suggested that the three chip multi-chip design using flip-chip IC's and MCM-D substrate is the most cost effective design among the various design candidates considered for the MicroSparc design [13].

The early cache analysis showed that the separate memory and logic process made it feasible to design the MicroSparc design with larger cache sizes than the combined logic and memory process. The designs based on the separate process gave equivalent or better performance and cost/performance than the design candidates with smaller cache sizes [14].

The early bond pitch analysis performed using the tool concluded that 250 micron bond pitch is the best choice for the multi-chip MicroSparc designs using the flip-chip MCM-D packaging technology [24].

The early analysis methodology described in this work has been adapted and used to evaluate other RISC processor designs and found to suggest conclusions

identical to the one determined in this work [30].

The comprehensive benefit of the early analysis packaging methodology developed in this work is clearly demonstrated from the above conclusions.

6.3 Future Work

This work mainly focussed on the early analysis at the functional unit level using exhaustive partitioning algorithm, even though an attempt was made in this work to perform the early analysis at the Register Transfer (RT) level using a simulated annealing algorithm. Future work in the area of early analysis should consider RT level designs using simulated annealing and genetic algorithms.

The early analysis tool should be enhanced to handle net-lists other than Viewlogic.

The current work used this tool to evaluate the MicroSparc design. Future work should consider using this tool to evaluate other design candidates.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] David P. Lapotin et al., "Early Package Analysis:Considerations and Case Study", *Computer*, pp. 30–39, April 1993.

[2] MCC, Austin, Texas, *The MSDA Tradeoff Analysis Tool Users Guide - Volume I*, 1994.

[3] P. Sandborn, M. Abadir and C. Murphy, "The Tradeoff Between Peripheral and Area Array Bonding of Components in Multichip Modules", *IEEE Trans. on Components, Packaging and Mfg. Tech. - Part A*, vol. 17, no. 2, pp. 249–256, June 1994.

[4] B.M. Hahne, D.E. O'Brien and J.P. Krusius, "Package Architecture Design and Optimization Tool AUDiT:Version 4.2 for Inhomogenous Systems", in *Proceedings of 42nd Electronics Component Technology Conferenc*, 1992, pp. 326–331.

[5] James D. Roberts, Dr. Wayne W.M. Dai, "Early System Analysis of Cache Performance for RISC Systems: MCM Design Trade-Offs", Tech. Rep. UCSC-CRL-92-02, Univ. of California, Santa Cruz, June 1992.

[6] Paul D. Franzon et al., "System Design Optimization for MCM", in *IEEE Multi-Chip Module Conference*, 1995, pp. 138–143.

[7] R.R. Tummala and E.J. Rymaszewski, Eds., *Microelectronic Packaging Handbook*, Van Nostrand Reinhold, 1989.

[8] A.J. Blodgett and D.R. Barbour, "Thermal Conduction Module: A High-Performance Ceramic Package", *IBM Journal of Research and Development*, vol. 26, pp. 30–36, 1982.

[9] J.U. Knickerbocker et al., "IBM System/390 Air Cooled Alumina Thermal Conduction Module", *IBM Journal of Research and Development*, vol. 35, no. 3, pp. 330–341, May 1991.

[10] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, New York, 1979.

[11] Ram Vemuri, *Genetic Algorithms for Partitioning, Placement and Layer Assignment for Multichip Modules*, PhD thesis, University of Cincinnati, 1994.

[12] Karthikeyan Ramamurthi, "Impact of packaging technology on system partitioning", Master's thesis, University of Tennessee, Knoxville, 1995.

[13] P.Dehkordi, K.Ramamurthi, D.Bouldin, H.Davidson, and P.Sandborn, "Impact of Packaging Technology on System Partitioning: A Case Study", in *Proc. IEEE Multi-Chip Module Conference*, 1995, pp. 144–149.

[14] P. Dehkordi, K. Ramamurthi, D. Bouldin and H. Davidson, "Early Cost/Performance Cache Analysis of a Split MCM-Based MicroSparc CPU", in *Proc. IEEE Multi-Chip Module Conference*, 1996.

[15] A.I. Kayssi, K.A. Sakallah, R.B. Brown, and R.J. Lomax, "Impact of MCMs on System Performance Optimization", in *1992 IEEE International Symposium on Circuits and Systems*, 1992, pp. 919–922.

[16] David A. Patterson John L. Hennessy, *Computer Architecture A Quantative Approach*, Morgan Kaufmann Publisher, Inc., 1990.

[17] Neil H. Weste and Kamran Eshragin, *Principles of CMOS VLSI Design*, Addison Wesley, 1992.

[18] Wayne Wolfe, *Modern VLSI System Design*, Prentice Hall, 1994.

[19] H.B. Bakoglu, *Circuits, Interconnections and Packaging for VLSI*, Addition-Wesley, 1990.

[20] Paul Franzon Daryl Ann Doyle, Ed., *MultiChip Module Technologies and Alternatives*, Van Nostrand, 1993.

[21] P.D. Franzon and R.J. Evans, "A Multichip Module Design Process for Notebook Computers", *IEEE Computer*, pp. 41–49, April 1993.

[22] Martin Taylor, Daniel Helman, and Wayne Dai, "AMP, A Bus-based Packet Switch Design using MCM", in *Multichip Module Workshop*, 1991, pp. 25–32.

[23] T. Powell, "Delay estimation", Tech. Rep., University of Tennessee, Nov 1994.

[24] P. Dehkordi, K. Ramamurthi, D. Bouldin and H. Davidson, "Determination of Area-Array Bond Pitch for Optimum MCM Systems: A Case Study", in *Proc. IEEE Multi-Chip Module Conference*, 1997, pp. 8–12.

[25] R. Vemuri, N. Kumar, R. Vemuri, "Two randomized algorithms for multichip partitioning under multiple constraints", Tech. Rep. TM-ECE-DDE-94-36, Univ. of Cincinnati, 1994.

[26] Barry Tolnas, "Multi objective design advisor", Master's thesis, University of Tennessee, Knoxville, 1995.

[27] MicroModule Sytem Corp., "MCM-D Technology Kit, Rev. 1.01", Tech. Rep., MicroModule System Corp. Cupertino, CA, 1994.

[28] *Numerical Recipes in C*, Cambridge University Press.

[29] Paul D. Franzon and et.al., "Issues in Partitioning Integrated Circuits for MCM-D/Flip-Chip Technology", in *IEEE Multi-Chip Module Conference*, 1996, pp. 154–159.

[30] V. Garg, S. Lacy, D. Schimmel, D. Stogner, C. Ulmer, D. S. Wills, S. Yalamanchili, "Incorporating Multi-Chip Module Packaging Constraints into System Design", in *Proceedings of the 1996 European Design and Test Conference*, March 1996, pp. 508–513.

VITA

Karthikeyan Ramamurthi graduated from PSG College of Technology, Coimbatore, India with a Bachelor of Engineering in Electrical Engineering in 1984. He worked for a couple of years in the automotive and power electronics industries in India. He took up teaching position in his alma mater for a year before pursuing his graduate studies. He got his Master of Technology degree in Electrical Machines and Power Electronics from the Indian Institute of Technology, New Delhi in 1990. He joined the University of Tennessee, Knoxville, USA for higher studies in August 1991. He worked at the Power Electronics Application Center, Knoxville as Graduate Research Assistant from August 1991 through August 1993. He was a Graduate Teaching Assistant in the Department of Electrical and Computer Engineering during the fall semester of 1993. He worked as Graduate Research Assistant in the ARPA funded project on Design for Packageability from January 1994 to August 1996. He got his M.S. in Electrical Engineering from University of Tennessee, Knoxville in December 1995. He has been working as a Design Engineer in the Flash Products Division of Intel Corp., Folsom, CA, from August 1996.